DRAFT VERSION APRIL 30, 2021
Typeset using LATEX **twocolumn** style in AASTeX631

Perturbative Approach to Solve the Map-Making Equation

BAI-QIANG QIANG[1] AND KEVIN M. HUFFENBERGER[1]

[1] *Florida State University*

ABSTRACT

We present a method to solve map-making equation by reinterpreate messenger field method and using conjugate gradient algorithm. We separating noise covariance matrix N into two parts, white noise part and the remaining noise part as perturbation, with perturbation parameter η. Then we give an analytical expression for η, and show that it won't introduce extra computational cost compare to traditional conjugate gradient method, but give better result for $1/f$ frequency model.

Keywords: Some Keywords

1. INTRODUCTION

Map-making is an intermediate process between data collection and estimate various cosmological parameters. As the next generation CMB observations will have much higher resolution and generate more data, we need an efficient way to process the data. There are many map-making methods introduced in Tegmark (1997a). Currently, one the most commonly used method is COBE method.

Recently Elsner & Wandelt (2013) introduced a new method called messenger field to solve Wiener filter, and then this technique was being applied to map-making by Huffenberger & Næss (2018), and messenger field could outperform traditional conjugate gradient method, with proper cooling technique. It has been shown by Papež et al. (2018) this messenger field method is equivalent to applying a preconditioner to the original problem and introducing an extra cooling parameter λ, but whether this cooling parameter will boost performance compare to the (traditional) conjugate gradient method is still controversial . Here I give a detailed analysis of this parameter and show that it may improve performance under some circumstances, if we properly choose its values.

The map making procedure could be summarized in equation

$$\mathbf{d} = P\mathbf{m} + \mathbf{n} \tag{1}$$

where $\mathbf{d}$, P, $\mathbf{m}$, $\mathbf{n}$ are time-ordered data (TOD), pointing matrix, CMB map, and noise. Here we assume that the noise has zero mean $\langle \mathbf{n} \rangle = \mathbf{0}$, and noise covariance matrix could be written as $N = \left\langle \mathbf{n}\mathbf{n}^\dagger \right\rangle$.

As we can see the map making model Eq.(1) mathematically is a standard linear regression problem, with *design matrix* being pointing matrix P, and *regression coefficients* are $\mathbf{m}$. For COBE method, we estimate linear regression coefficients $\mathbf{m}$ with *generalized least square* (GLS) technique, since the noise $\mathbf{n}$ is *heteroscedastic*. The GLS minimize

$$\chi^2(\mathbf{m}) \equiv (\mathbf{d} - P\mathbf{m})^\dagger N^{-1} (\mathbf{d} - P\mathbf{m}). \tag{2}$$

and the estimated map $\hat{\mathbf{m}}$ is given by

$$\hat{\mathbf{m}} = \arg\min_{\mathbf{m}} \chi^2(\mathbf{m}) = \left(P^\dagger N P\right)^{-1} P^\dagger N^{-1} \mathbf{d} \tag{3}$$

Or rewrite it as

$$\left(P^\dagger N^{-1} P\right)\hat{\mathbf{m}} = P^\dagger N^{-1} \mathbf{d} \tag{4}$$

This is the map-making equation we need to solve. However, based on current computation power, it is impossible to solve $\hat{\mathbf{m}}$ by calculating $\left(P^\dagger N^{-1} P\right)^{-1} P^\dagger N^{-1} \mathbf{d}$ directly, since the noise covariance matrix N is sparse in frequency domain, and pointing matrix P is sparse in (time by pixel) domain. In experiments currently under design, there may be $\sim 10^{16}$ time samples and $\sim 10^9$ pixels, so these matrix inversions are intractable. Therefore we use Conjugate Gradient method, which is an iterative algorithm, to solve this map-making equation. For simplicity we fix the preconditioner being $M = P^\dagger P$ for all of calculations.

The structure of this paper is organized as follows. In section 2 we briefly introduce messenger field fixed point iteration method and preconditioned version. In section 3 we reinterpret this process and give an analysis on how to determine the parameters. Section 4 gives the noise power spectrum in our simulation, and Section 5 shows results. Finally, Section 6 we discuss this method's pro and con for possible future improvements.

2. MESSENGER FIELD METHOD

Messenger field method separate noise covariance matrix $N = \bar{N} + T$, with $T = \tau I$ and τ being the minimum eigenvalue of N. Then there is a cooling parameter λ such that $N(\lambda) = \bar{N} + \lambda T$, with initial λ being a very large number and final λ being 1.

After apply preconditioner $P^\dagger T^{-1} P$ to the map making equation Eq.(4), we would get:

$$\begin{aligned}\hat{\mathbf{m}} = &\left(P^\dagger T^{-1} P\right)^{-1} P^\dagger T^{-1} \left(T^{-1} + \bar{N}^{-1}\right)^{-1} \\ &\times \left[T^{-1} P \hat{\mathbf{m}} + \bar{N}^{-1} \mathbf{d}\right]\end{aligned} \tag{5}$$

To add cooling parameter λ, we need to change T to λT and N to $N(\lambda)$. Then we could rewrite it as a fixed point iteration form

$$\begin{cases} \mathbf{t}_i = \left((\lambda T)^{-1} + \bar{N}^{-1}\right)^{-1} \left[(\lambda T)^{-1} P \hat{\mathbf{m}}_i + \bar{N}^{-1} \mathbf{d}\right] \\ \hat{\mathbf{m}}_{i+1} = \left(P^\dagger (\lambda T)^{-1} P\right)^{-1} P^\dagger (\lambda T)^{-1} \mathbf{t}_i \end{cases} \tag{6}$$

This is fixed point iteration form of messenger field method. It's equivalent to solving map-making equation Eq.(4) with preconditioner $P^\dagger (\lambda T)^{-1} P = \tau^{-1} P^\dagger P$. This preconditioner is equivalent to preconditioner $M = P^\dagger P$, since multiply a constant won't change condition number. Therefore messenger field is solving modified map making equation

$$P^\dagger N(\lambda)^{-1} P \ \hat{\mathbf{m}} = P^\dagger N(\lambda)^{-1} \mathbf{d} \tag{7}$$

with preconditioner $M = P^\dagger P$. More detailed derivation could be found in Papež et al. (2018).

3. PARAMETERIZED CONJUGATE GRADIENT METHOD

3.1. *Introduce the Idea*

The messenger field method introduced an extra cooling parameter λ to map-making equation, so we want to know how to choose this parameter. In Kodi Ramanah et al. (2017), they showed that for Wiener filter the cooling parameter should be chosen as a geometric series. In this work, we are going to give an alternative interpretation of the parameterizing process then show that for map-making equation the optimal choice for λ would also be a geometric series.

Based on previous analysis, we know that what messenger field method really does is parameterizing the map-making equation. Here to avoid confusion, we introduce another parameter η, such that the parameterized map-making equation is

$$P^\dagger N(\eta)^{-1} P \ \hat{\mathbf{m}} = P^\dagger N(\eta)^{-1} \mathbf{d} \tag{8}$$

The idea is that map-making equation Eq.(4) is hard to solve due to noise covariance matrix is sandwiched between $P^\dagger P$. But if noise covariance matrix N is proportional to identity matrix I, then its solution is given by simple binned map $\mathbf{m}_0 = \left(P^\dagger P\right)^{-1} P^\dagger \mathbf{d}$, which could be solved directly. We can parameterize the noise covariance matrix N with a parameter η, such that initially $\eta = \eta_i$ we have $N(\eta_i) \propto I$. In the end $\eta = \eta_f$ and $N(\eta_f) \propto N$, such that the final solution is what we want. We expect that the parameterized noise covariance matrix $N(\eta)$ would connect our initial guess $\hat{\mathbf{m}}_0$ and final solution $\hat{\mathbf{m}}$ as we change η from η_i to η_f, such that it would help improve convergence speed.

Now question is how to find $N(\eta)$ such that $N(\eta_i) \propto I$ and $N(\eta_f) \propto N$? Since the non-white noise part of N is the difficult portion, we could think of it as a perturbation term, which adds upon the white noise. Initially there is only white noise and solution is given by $\hat{\mathbf{m}}_0$, then we gradually add extra noise into this equation by changing η from 0 to 1. At the end when $\eta = 1$ we are solving equation Eq.(4).

Therefore we separate noise covariance matrix into two parts $N = \tau I + \bar{N}$ where τ is the minimum eigenvalue of N. Then we define $N(\eta) = \tau I + \eta \bar{N}$, with perturbation parameter η which satisfies $\eta_i = 0$ and $\eta_f = 1$.

Eq.(8) then becomes

$$\left(P^\dagger (\tau I + \eta \bar{N})^{-1} P\right) \hat{\mathbf{m}}(\eta) = P^\dagger \left(\tau I + \eta \bar{N}\right)^{-1} \mathbf{d} \tag{9}$$

We require the perturbation parameter η being monotonically increase series $0 = \eta_0 < \eta_1 < \cdots < \eta_n = 1$. For some specific η_m, we use conjugate gradient method to solve equation $\left(P^\dagger N(\eta_m)^{-1} P\right) \hat{\mathbf{m}}(\eta_m) = P^\dagger N(\eta_m)^{-1} \mathbf{d}$ with simple preconditioner $P^\dagger P$, and using $\hat{\mathbf{m}}(\eta_{m-1})$ as the initial value. The initial guess is $\hat{\mathbf{m}}(\eta_0) = \mathbf{m}_0 = \left(P^\dagger P\right)^{-1} P^\dagger \mathbf{d}$.

As you can see, η is the reciprocal of λ. The reason I switch to η in stead of keeping λ is that it would be easier for further derivations, and it's a different interpretation.

3.2. *Choosing perturbation parameters η*

The next question is how we choose these monotonically increasing parameters η. If we choose these parameters inappropriately, it would only makes it converge slower. Also we want to determine $\eta_1, \cdots, \eta_{n-1}$ before starting conjugate gradient iteration. That's because time ordered data $\mathbf{d}$ is very large, and we don't want to keep it in the system RAM during calculation. If $\eta_1, \cdots, \eta_{n-1}$ could be determined before the iterations, then we can first calculate $P^\dagger N(\eta)^{-1} \mathbf{d}$ for each η_m and store these map-sized objects in RAM, instead of the entire time-ordered data $\mathbf{d}$.

First let us try to find out our starting point η_1. What would be good value for η_1?

Here to simplify notation, I will use N_η to denote $N(\eta)$. The parameterized estimated map $\hat{\mathbf{m}}(\eta) = \left(P^\dagger N_\eta^{-1} P\right)^{-1} P^\dagger N_\eta^{-1}\mathbf{d}$ minimizes the parameterized

$$\chi^2(\mathbf{m},\eta) = (\mathbf{d} - P\mathbf{m})^\dagger N_\eta^{-1}(\mathbf{d} - P\mathbf{m}). \tag{10}$$

For some specific η value, the minimum χ^2 value is given by

$$\chi^2(\hat{\mathbf{m}}(\eta),\eta) = \left(\mathbf{d} - P\hat{\mathbf{m}}(\eta)\right)^\dagger N_\eta^{-1}\left(\mathbf{d} - P\hat{\mathbf{m}}(\eta)\right) \tag{11}$$

To further simplify the analysis, let's assume that the noise covariance matrix $N = \left\langle \mathbf{n}\mathbf{n}^\dagger \right\rangle$ is diagonal in the frequency domain.

Let's first consider $\eta_1 = \eta_0 + \delta\eta = \delta\eta$ such that $\eta_1 = \delta\eta$ is very small quantity. Then the relative decrease of $\chi^2(\hat{\mathbf{m}}(0),0)$ from $\eta_0 = 0$ to $\eta_1 = \delta\eta$ is

$$-\frac{\delta\chi^2(\hat{\mathbf{m}}(0),0)}{\chi^2(\hat{\mathbf{m}}(0),0)} = \delta\eta\frac{1}{\tau}\frac{(\mathbf{d} - P\hat{\mathbf{m}}(0))^\dagger \bar{N}(\mathbf{d} - P\hat{\mathbf{m}}(0))}{\left(\mathbf{d} - P\hat{\mathbf{m}}(0)\right)^\dagger\left(\mathbf{d} - P\hat{\mathbf{m}}(0)\right)} \tag{12}$$

Here we put a minus sign in front of this expression such that it's non-negative.

Ideally, we want $\delta\chi^2(\hat{\mathbf{m}}(0),0) = \chi^2(\hat{\mathbf{m}}(1),1) - \chi^2(\hat{\mathbf{m}}(0),0)$, such that it would get close to the final χ^2 at next iteration. Here if we assume that initial χ^2 value $\chi^2(\hat{\mathbf{m}}(0),0)$ is much larger than final value $\chi^2(\hat{\mathbf{m}}(1),1)$, then we would expect $\left|\delta\chi^2(\hat{\mathbf{m}}(0),0)/\chi^2(\hat{\mathbf{m}}(0),0)\right| \approx 1^-$, strictly smaller than 1. To make sure it will not start too fast, we could set its upper bound equal to 1, $\delta\eta \max(\bar{N}_f)/\tau = 1$. This gives

$$\eta_1 = \frac{\tau}{\max(\bar{N}_f)} = \frac{\min(N_f)}{\max(N_f) - \min(N_f)} \tag{13}$$

Here N_f and $\bar{N}_f$ are the eigenvalues of N and $\bar{N}$ under frequency domain. If the condition number of noise covariance matrix $\kappa(N) = \max(N_f)/\min(N_f) \gg 1$, then $\eta_1 \approx \kappa^{-1}(N)$.

What about the other parameters η_m with $m > 1$? We could use a similar analysis, let $\eta_{m+1} = \eta_m + \delta\eta_m$ with a small $\delta\eta_m$, and set the upper bound of relative decrease equal to 1. See Appendix A for detailed derivation. We would get

$$\delta\eta_m = \min\left(\frac{\tau + \eta_m\bar{N}_f}{\bar{N}_f}\right) = \eta_m + \frac{\tau}{\max(\bar{N}_f)}. \tag{14}$$

Therefore

$$\eta_{m+1} = \eta_m + \delta\eta_m = 2\eta_m + \frac{\tau}{\max(\bar{N}_f)} \tag{15}$$

As we can see, $\eta_1, \cdots, \eta_n$ increase like a geometric series.

$$\eta_i = \min\left\{1,\ \frac{\tau}{\max(\bar{N}_f)}\left(2^i - 1\right)\right\} \tag{16}$$

Here we need to truncate the series when $\eta_i > 1$.

This is the main result. Eq.(16) tells us not only how to choose parameters η_i, but also when we should stop the perturbation, and set $\eta = 1$. For example, if noise covariance matrix N is almost white noise, then $\bar{N} = N - \tau I \approx 0$, and we would have $\frac{\tau}{\max(\bar{N}_f)} \gg 1$. This tell us that we don't need to use parameterized method at all, because $\eta_1 = 1$. Note that the vanilla conjugate gradient method with simple binned map as initial guess corresponds to choosing $\eta_0 = 0$ and $\eta_1 = \eta_2 = \cdots = 1$.

3.3. *Intuitive Interpretation of* η

In this section, let me introduce another way to understand the role of η. Our ultimate goal is to find $\hat{\mathbf{m}}(\eta = 1)$ which minimizes $\chi^2(\mathbf{m}) = (\mathbf{d} - P\mathbf{m})^\dagger N^{-1}(\mathbf{d} - P\mathbf{m})$. Since N is diagonal in frequency space, χ^2 could be written as a sum of all frequency mode $|(\mathbf{d} - P\mathbf{m})_f|^2$ with weight N_f^{-1}, such as $\chi^2(\mathbf{m}) = \sum_f |(\mathbf{d} - P\mathbf{m})_f|^2 N_f^{-1}$. N_f^{-1} is large when there is little noise at that frequency, and vice versa. Which means $\chi^2(\mathbf{m})$ would favor the low noise frequency mode over high noise ones. In other words the optimal map $\hat{\mathbf{m}}$ focusing on minimize the error $\mathbf{r} \equiv \mathbf{d} - P\mathbf{m}$ in the low-noise part.

After introducing η, we minimize $\chi^2(\mathbf{m},\eta) = (\mathbf{d} - P\mathbf{m})^\dagger N_\eta^{-1}(\mathbf{d} - P\mathbf{m})$. For $\eta = 0$, $N_{\eta=0}^{-1} \propto I$ and the estimated map $\hat{\mathbf{m}}(\eta = 0)$ does not prioritize any frequency mode. As we slowly increase η, we decrease the weight for the frequency modes which have large noise, and focusing minimizing error for low noise part. If we start with $\eta_1 = 1$ directly, which corresponds to the vanilla conjugate gradient method, then the entire conjugate gradient solver will only focusing on minimizing low noise part, such that χ^2 would converge very fast at low noise region, but relative slow on high noise part. However by introducing η parameter, we let the solver first treat every frequency equally. Then as η slowly increases, it gradually shifts focus to low noise part.

3.4. *Computational Cost*

To properly compare the performance cost of this method with respect to vanilla conjugate gradient method with simple preconditioner, we need to compare their computational cost at each iteration. The right hand side of parameterized map-making equation Eq.(8) could be computed before iterations, so it won't introduce extra computational cost. The most demanding part of conjugate gradient method is calculating $P^\dagger N^{-1} P\hat{\mathbf{m}}$, because it contains a Fourier transform of

$P\hat{\mathbf{m}}$ from time domain to frequency domain and an inverse Fourier transform of $N^{-1}P\hat{\mathbf{m}}$ from frequency domain back to time domain, which is order $\mathcal{O}(n\log n)$ with n being the length of time ordered data. If we change N^{-1} to $N(\eta)^{-1}$, it won't add extra cost, since both matrices are diagonal in frequency domain. Therefore the computational cost it the same for one step.

However our previous analysis is based on $\chi^2(\hat{\mathbf{m}}(\eta_i), \eta_i)$ which is evaluated at $\hat{\mathbf{m}}(\eta_i)$ the estimated map at η_i. So We should update η_i to η_{i+1} when $\mathbf{m} \approx \hat{\mathbf{m}}(\eta_i)$. How do we know this condition is satisfied? Since for each new η_i value, we are solving a new set of linear equations $A(\eta_i)\hat{\mathbf{m}} = \mathbf{b}(\eta_i)$ with $A(\eta_i) = P^\dagger N(\eta_i)^{-1}P$ and $\mathbf{b}(\eta_i) = P^\dagger N(\eta_i)^{-1}\mathbf{d}$, and we could stop calculation and moving to next value η_{i+1} when the norm of residual $||\mathbf{r}(\eta_i)|| = ||\mathbf{b}(\eta_i) - A(\eta_i)\mathbf{m}||$ smaller than some small value. Calculate $||\mathbf{r}(\eta_i)||$ is part of conjugate gradient algorithm, so this won't add extra cost compare to vanilla conjugate gradient method. Therefore, overall introducing η won't have extra computational cost.

4. NUMERICAL SIMULATIONS

To compare these algorithms, we need to do some simple simulation of scanning processes, and generate time ordered data from random sky signal. Our sky is a small rectangular area, with two orthogonal directions x and y, both with range from $-1°$ to $+1°$. The signal has first three stokes parameters (I, Q, U).

For the scanning process, our single telescope contains nine detectors, each has different sensitivity to polarization Q and U. It scans the sky with a raster scanning pattern and scanning frequency $f_{\text{scan}} = 0.1$ Hz sampling frequency $f_{\text{sample}} = 100$ Hz. The telescope scans the sky horizontally and then vertically, and then digitizes the position (x, y) into 512×512 pixel. This gives noiseless signal $\mathbf{s}$.

The noise power spectrum is given by

$$P(f) = \sigma^2\left(1 + \frac{f_{\text{knee}}^\alpha + f_{\text{apo}}^\alpha}{f^\alpha + f_{\text{apo}}^\alpha}\right) \tag{17}$$

Here we fixed $\sigma^2 = 10\ \mu\text{K}^2$, $\alpha = 2$ and $f_{\text{knee}} = 10$ Hz, and change f_{apo} to compare the performance under different noise models. Note that as $f_{\text{apo}} \to 0$, $P(f) \to \sigma^2\big(1 + (f/f_{\text{knee}})^{-1}\big)$, it becomes a $1/f$ noise model. The noise covariance matrix

$$N_{ff'} = P(f)\frac{\delta_{ff'}}{\Delta_f} \tag{18}$$

is a diagonal matrix in frequency space, where Δ_f is equal to reciprocal of total scanning time T.

Finally, we get the simulated time ordered data $\mathbf{d} = \mathbf{s} + \mathbf{n}$ by adding up signal and noise.

5. RESULTS

First let's compare the results with vanilla conjugate gradient method with simple preconditioner $P^\dagger P$. The results are showed in Figure.(1) for different kinds of noise power spectra. Here note that χ^2 in all figures are calculated based on Eq.(2) not $\chi^2(\mathbf{m}, \eta)$ in Eq.(10). The χ^2_{final} is calculated from perturbative conjugate gradient method with more intermediate η values, and more iterations after $\eta = 1$.

As we can see in the center graph in Figure(1), if the condition number of noise covariance matrix $\kappa(N)$ is small, and the noise is almost white noise, the performance between different these two methods is small. The vanilla conjugate gradient method converge faster, because its perturbation parameter goes to 1 at the first iteration, however for the perturbation method its η value will slowly reach 1 in few iterations as we can see in the third graph in Figure(1).

Notice that as we increase $\kappa(N)$, or equivalently decrease f_{apo}, the perturbation parameter η starts showing its benefits, as showed in Figure(2) and Figure(3). It outperforms the vanilla conjugate gradient method when $f_{\text{apo}} \approx 0$ and the noise power spectrum becomes the $1/f$ noise model, which usually is the intrinsic noise of instruments (Tegmark (1997b)).

Now let us compare the performance difference between choosing η parameters based on Eq.(16) and manually fixing number of η parameters n_η manually. We manually choose the η_i values using function `numpy.logspace(start=`$\ln(\eta_1)$`, stop=0, num=`n_η`, base=`e`)`. The results are showed in Figure(4), (5), and (6).

When $\kappa(N)$ is small, and Eq.(16) tells us that only a few η parameters are good enough, see the orange line in the last Figure(4). If unfortunately we choose n_η being large value, like 15 or 30, then it will ends up converge slowly, because it needs at least 15 or 30 iterations to reach $\eta = 1$, at least one iteration per η level.

On the other hand if $\kappa(N)$ is very large and the power spectrum is $1/f$ noise, we need more η parameters. If n_η is too small, for example $n_\eta = 5$ the red line in Figure(6), it may be better than the vanilla conjugate gradient method, but it is still far from optimal.

6. FUTURE PROSPECTS AND CONCLUSION

As you may have noticed in Figure(5) and Figure(6), the perturbation parameter based on Eq.(16) is more than needed, especially for $1/f$ noise case. From the last graph in Figure(6) we notice that Eq.(16) gives us $n_\eta \approx 40$, however based on χ^2 result in Figure(6) $n_\eta \approx 30$ or even $n_\eta \approx 15$ is good enough. Also, for the nearly-white-noise case, we could certainly choose

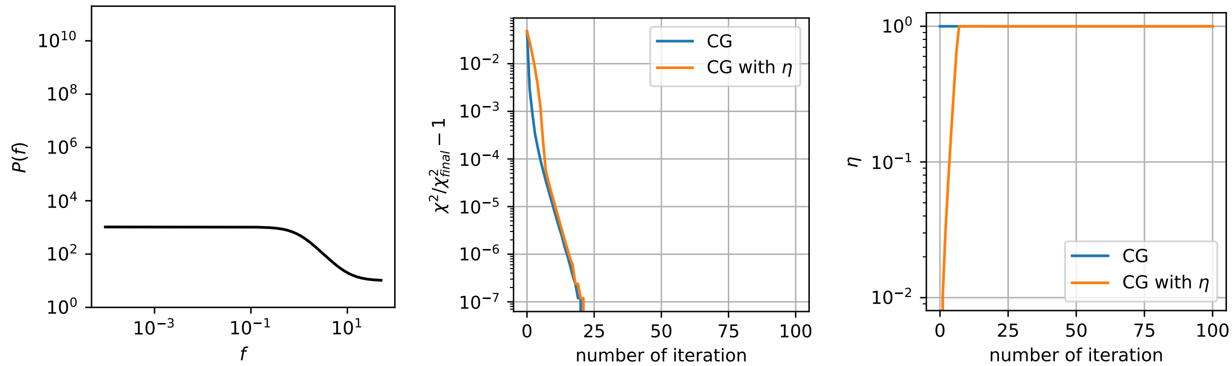

Figure 1. The left graph shows the noise power spectrum Eq.(17) with $f_{\rm apo} \approx 0.99$ and $\kappa(N) = 10^2$. The center one shows the $\chi^2(\mathbf{m})/\chi^2_{\rm final} - 1$, with $\chi^2(\mathbf{m})$ calculated based on Eq.(2). The right one shows the η value for each iteration. For vanilla conjugate gradient method η always equal to 1, so it's a horizontal line at $\eta = 1$.

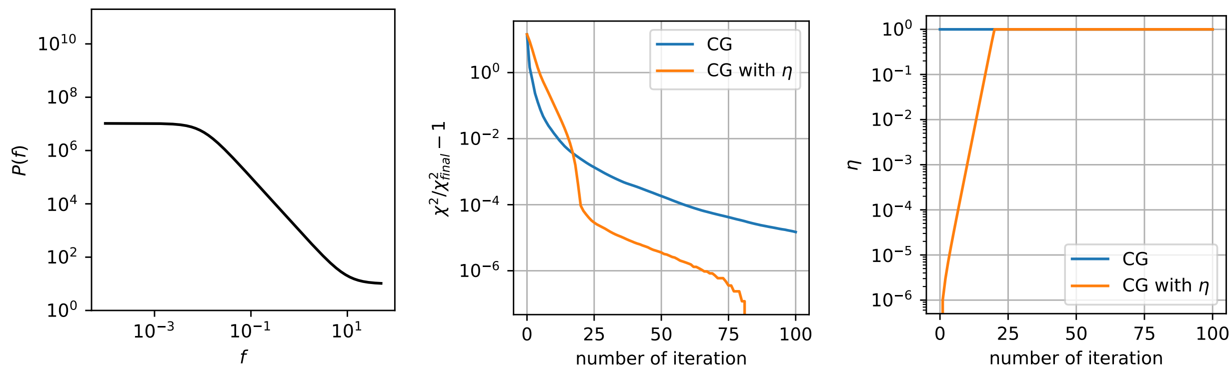

Figure 2. The figure shows results for $f_{\rm apo} \approx 9.8 \times 10^{-3}$ and $\kappa(N) = 10^6$.

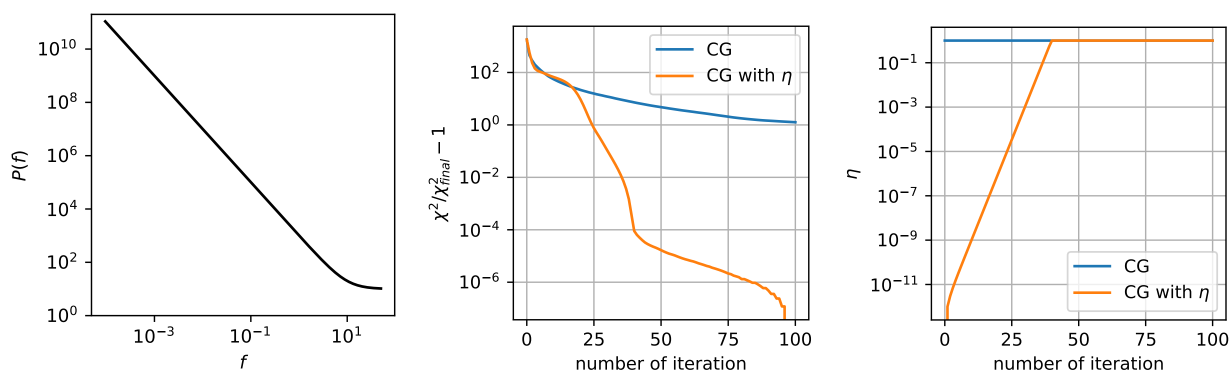

Figure 3. The figure shows results for $f_{\rm apo} \approx 9.8 \times 10^{-6}$ and $\kappa(N) = 10^{12}$.

$n_\eta = 1$ such that $\eta_1 = 1$ which corresponds to vanilla conjugate gradient method, based on χ^2 result in Figure(4). However Eq.(16) gives us $n_\eta \approx 6$, see the last last graph in Figure(4), even though it does not make the final χ^2 result much different at the end.

Is it possible to further improve the analysis, such that it produces smaller n_η? Let's examine how we get η_i series. Remember that we determine $\delta\eta$ value based on the upper bound of $-\delta\chi^2(\hat{\mathbf{m}}(\eta),\eta)/\chi^2(\hat{\mathbf{m}}(\eta),\eta)$, in Eq.(12). For $\eta \neq 0$, the upper bound is

$$\delta\eta \frac{\hat{\mathbf{r}}_\eta^\dagger N(\eta)^{-1}\bar{N}N(\eta)^{-1}\hat{\mathbf{r}}_\eta}{\hat{\mathbf{r}}_\eta^\dagger N(\eta)^{-1}\hat{\mathbf{r}}_\eta} \leq \frac{\delta\eta}{\eta + \frac{\tau}{\max(N_f)-\tau}} \tag{19}$$

with $\mathbf{r}_\eta = \left[1 - P\left(P^\dagger N(\eta)^{-1}P\right)^{-1}P^\dagger N(\eta)^{-1}\right]\mathbf{d} \equiv \mathcal{P}_\eta \mathbf{d}$. To get the upper bound we treated $\mathbf{d} - P\hat{\mathbf{m}}(\eta)$ as an arbitrary vector in frequency domain, since we don't know

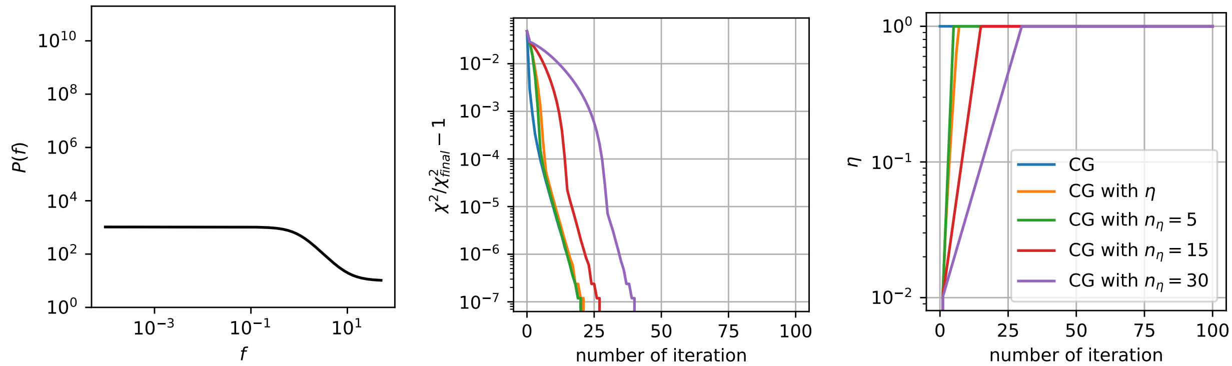

Figure 4. Same as Figure(1) with extra manually chosen n_η results.

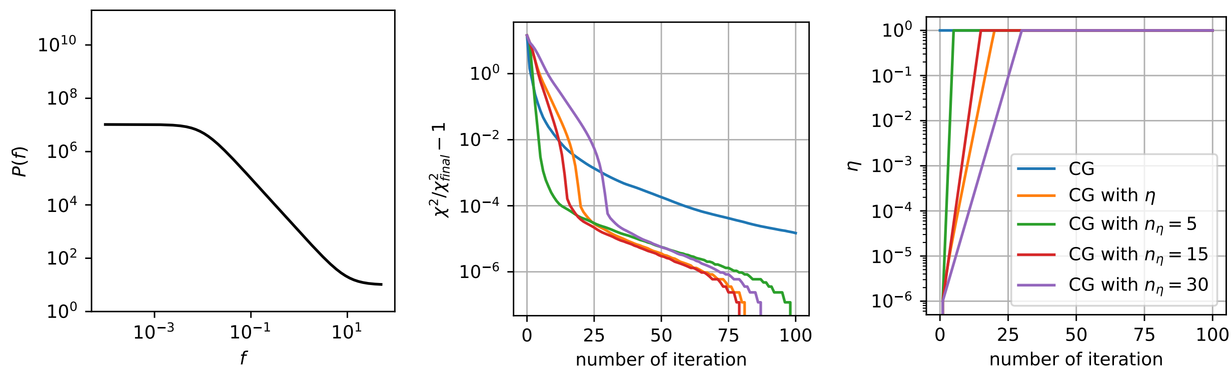

Figure 5. Same as Figure(2) with extra manually chosen n_η results.

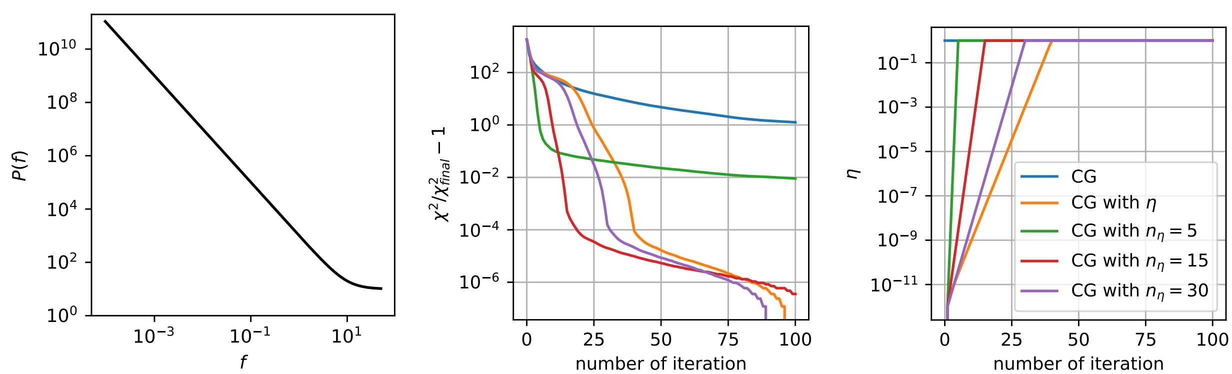

Figure 6. Same as Figure(3) with extra manually chosen n_η results.

how to calculate $\mathcal{P}_\eta$ for $\eta \neq 0$, and it's hard to analyze the projection matrix $\mathcal{P}_\eta$ in frequency space, as it contains $\left(P^\dagger N(\eta)^{-1} P\right)^{-1}$. Note that we have to determine all of η value before calculation, because we don't want to keep the time ordered data in system RAM, so we need to somehow analytically analyze $\mathcal{P}_\eta$, and its behavior in frequency space. Unless $\mathbf{r}_\eta$ almost only has large noise modes, $\left|\frac{\mathrm{d}}{\mathrm{d}\eta}\chi^2(\hat{\mathbf{m}}(\eta),\eta)/\chi^2(\hat{\mathbf{m}}(\eta),\eta)\right|$ won't get close to the upper bound $1/\left(\eta + \frac{\tau}{\max(N_f)-\tau}\right)$. Based on the analysis in Section(3.3), for small η the estimated map $\hat{\mathbf{m}}(\eta)$ does not only focusing on minimizing error $\mathbf{r}_\eta$ at low noise region. So we would expect that there would be a fair amount of low noise modes contribution in $\mathbf{r}_\eta$ especially for the first few η values. Which means if we could somehow know the frequency distribution of $\mathbf{r}_\eta$, we could tighten the boundary of $\left|\frac{\mathrm{d}}{\mathrm{d}\eta}\chi^2(\hat{\mathbf{m}}(\eta),\eta)/\chi^2(\hat{\mathbf{m}}(\eta),\eta)\right|$, and get larger $\delta\eta$ value.

This should make η goes to 1 faster, and yields the fewer η parameters we need.

Also notice that the η values determined from Eq.(16) are not dependent on any scanning information, it only depends on noise power spectrum $P(f)$, or noise covariance matrix N. In Appendix B we would show two examples with same parameters as in Figure(6) except scanning frequency f_{scan}.It turns out the η values should somehow depends on scanning scheme. Again that's because when we determine the upper boundwe treated $\mathbf{r}_\eta$ as an arbitrary vector, such that we lose all information related to scanning scheme in the pointing matrix P.

Even though the perturbation parameter η get from Eq.(16) are not the most optimal, it still performs much better than traditional conjugate gradient method under $1/f$ noise scenario without adding extra computational cost. The only extra free parameter added is to determine whether the error at current step $\mathbf{r}(\eta_i) = ||\mathbf{b}(\eta_i) - A(\eta_i)\mathbf{m}||$ is small enough such that we advance to next value η_{i+1}.

Also this analysis of η value also explains why cooling parameters $\lambda = 1/\eta$ in messenger field are chosen to be geometric series or `logspace` used in Huffenberger & Næss (2018).

All of the calculation are using simple preconditioner $P^\dagger P$, but the entire analysis is independent of preconditioner. Using better preconditioners, it would also have improvements.

APPENDIX

A. UPPER BOUND FOR η

We want to find the upper bound for $-\frac{\delta\chi^2(\hat{\mathbf{m}}(\eta_m),\eta_m)}{\chi^2(\hat{\mathbf{m}}(\eta_m),\eta_m)}$ First let's calculate $\frac{\mathrm{d}}{\mathrm{d}\eta}\chi^2(\hat{\mathbf{m}}(\eta),\eta)$

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\eta}\chi^2(\hat{\mathbf{m}}(\eta),\eta) &= \frac{\partial}{\partial\eta}\chi^2(\hat{\mathbf{m}}(\eta),\eta) \\
&= \frac{\partial}{\partial\eta}\big(\mathbf{d} - P\hat{\mathbf{m}}(\eta)\big)^\dagger N(\eta)^{-1}\big(\mathbf{d} - P\hat{\mathbf{m}}(\eta)\big) \\
&= -(\mathbf{d} - P\hat{\mathbf{m}}(\eta))^\dagger N(\eta)^{-1}\bar{N}N(\eta)^{-1}(\mathbf{d} - P\hat{\mathbf{m}}(\eta)) \\
&= -\mathbf{r}^\dagger(\eta)N(\eta)^{-1}\bar{N}N(\eta)^{-1}\mathbf{r}(\eta).
\end{aligned} \tag{A1}
$$

where the first line comes from, $\chi^2(\hat{\mathbf{m}}(\eta),\eta)$ is minimum χ^2 value for certain η, therefore $\frac{\partial}{\partial\mathbf{m}}\chi^2(\mathbf{m},\eta)\Big|_{\mathbf{m}=\hat{\mathbf{m}}(\eta)} = 0$. So the third line we only take partial derivative with respect to $N(\eta)^{-1}$. The last line we define $\mathbf{r}(\eta) = \mathbf{d} - P\hat{\mathbf{m}}(\eta)$.

The upper bound is given by

$$
\begin{aligned}
-\frac{\delta\chi^2(\hat{\mathbf{m}}(\eta_m),\eta_m)}{\chi^2(\hat{\mathbf{m}}(\eta_m),\eta_m)} &= \delta\eta_m \frac{\mathbf{r}^\dagger N(\eta_m)^{-1}\bar{N}N(\eta_m)^{-1}\mathbf{r}}{\mathbf{r}^\dagger N(\eta_m)^{-1}\mathbf{r}} \\
&\le \delta\eta_m \max\left(\frac{\bar{N}_f}{\tau + \eta_m\bar{N}_f}\right)
\end{aligned} \tag{A2}
$$

For the last line, both matrix $\bar{N}$ and $N(\eta_m)^{-1}$ can be simultaneously diagonalized in frequency space. For each eigenvector $\mathbf{e}_f$, the corresponding eigenvalues of the matrix $N(\eta_m)^{-1}\bar{N}N(\eta_m)^{-1}$ are $\lambda_f = \bar{N}_f(\tau + \eta_m\bar{N}_f)^{-2}$, and the eigenvalues for matrix $N(\eta_m)^{-1}$ are $\gamma_f = (\tau + \eta_m\bar{N}_f)^{-1}$. Their eigenvalues are related by $\lambda_f = \frac{\bar{N}_f}{\tau+\eta_m\bar{N}_f}\gamma_f$. For vector $\mathbf{r} = \sum_f \alpha_f\mathbf{e}_f$, we have $\frac{\mathbf{r}^\dagger N(\eta_m)^{-1}\bar{N}N(\eta_m)^{-1}\mathbf{r}}{\mathbf{r}^\dagger N(\eta_m)^{-1}\mathbf{r}} = \frac{\sum_f\alpha_f^2\lambda_f}{\sum_f\alpha_f^2\gamma_f} = \frac{\sum_f\alpha_f^2\gamma_f\bar{N}_f/(\tau+\eta_m\bar{N}_f)}{\sum_f\alpha_f^2\gamma_f} \le \max\left(\frac{\bar{N}_f}{\tau+\eta_m\bar{N}_f}\right)$.

If we set the upper bound $\delta\eta_m\max\left(\frac{\bar{N}_f}{\tau+\eta_m\bar{N}_f}\right) = 1$, [1] and then we get

$$
\delta\eta_m = \min\left(\frac{\tau + \eta_m\bar{N}_f}{\bar{N}_f}\right) = \eta_m + \frac{\tau}{\max(\bar{N}_f)}. \tag{A3}
$$

[1] Here we also assumed that $\chi^2(\hat{\mathbf{m}}(\eta_m),\eta_m) \gg \chi^2(\hat{\mathbf{m}}(1),1)$, which we expect it to be satisfied for $0 \simeq \eta_m \ll 1$. Since final result Eq.(16) is geometric series, only a few η_m values won't satisfy this condition.

B. OTHER CASES

Since the η values determined from Eq.(16)

$$\eta_i = \min\left\{1,\ \frac{\tau}{\max(\bar{N}_f)}\left(2^i - 1\right)\right\} \tag{16}$$

are not dependent on any scanning information, it only depends on noise power spectrum $P(f)$, or noise covariance matrix N. Figure(7) and Figure(8) show two examples with same parameters as in Figure(6) except scanning frequency f_{scan}, in Figure(7) it scans very slow and in Figure(8) it's very fast. In these two cases our η values based on Eq.(16) are better than manually selected values. Based on these two results we know, the η values should somehow depends on scanning scheme.

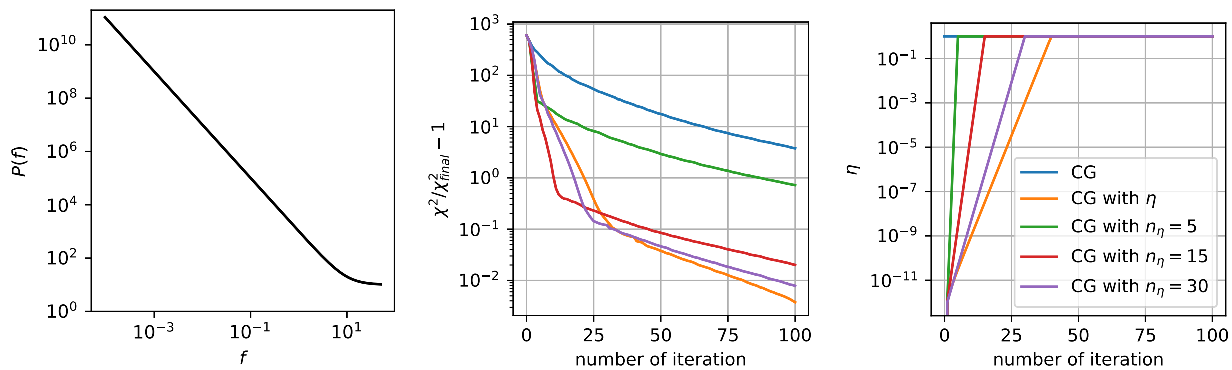

Figure 7. In this case all frequencies are the same as Figure(6) except $f_{\mathrm{scan}} = 0.001$.

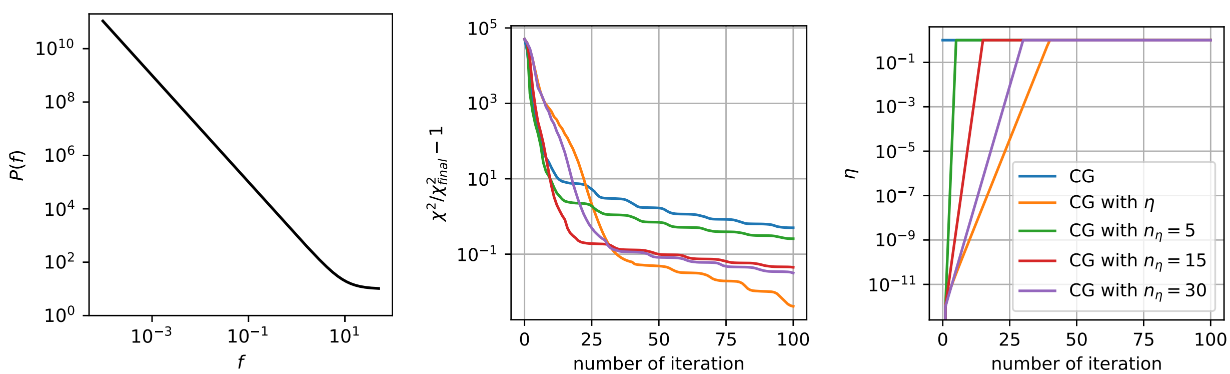

Figure 8. In this case all frequencies are the same as Figure(6) except $f_{\mathrm{scan}} = 10$.

REFERENCES

Elsner, F., & Wandelt, B. D. 2013, A&A, 549, A111, doi: 10.1051/0004-6361/201220586

Huffenberger, K. M., & Næss, S. K. 2018, The Astrophysical Journal, 852, 92, doi: 10.3847/1538-4357/aa9c7d

Kodi Ramanah, D., Lavaux, G., & Wandelt, B. D. 2017, MNRAS, 468, 1782, doi: 10.1093/mnras/stx527

Papež, J., Grigori, L., & Stompor, R. 2018, A&A, 620, A59, doi: 10.1051/0004-6361/201832987

Tegmark, M. 1997a, ApJL, 480, L87, doi: 10.1086/310631

—. 1997b, PhRvD, 56, 4514, doi: 10.1103/PhysRevD.56.4514