Draft version July 2, 2021
Typeset using LaTeX **twocolumn** style in AASTeX631

# Cosmic Microwave Background map-making solutions improve with cooling

Bai-Qiang Qiang (KMH: want chinese characters?)[1] and Kevin M. Huffenberger[1]

[1]*Department of Physics, Florida State University, Tallahassee, Florida 32306*

## ABSTRACT

In the context of Cosmic Microwave Background data analysis, we study the solution to the equation that transforms scanning data into a map. As originally suggested in "messenger" methods for solving linear systems, we split the noise covariance into uniform and non-uniform parts and adjusting their relative weight during the iterative solution. This "cooling" or perturbative approach is particularly effective when there is significant low-frequency noise in the timestream. A conjugate gradient algorithm applied to this modified system converges faster and to a higher fidelity solution than the standard conjugate gradient approach, for the same computational cost per iteration. We conclude that cooling is helpful separate from its appearance in the messenger methods. We give an analytical expression for the parameter that controls how gradually should change during the course of the solution.

*Keywords:* Computational methods — Cosmic microwave background radiation — Astronomy data reduction

## 1. INTRODUCTION

In observations of the Cosmic Microwave Background (CMB), map-making is an intermediate step between the collection of raw scanning data and the scientific analyses, such as the estimation of power spectra and cosmological parameters. Next generation CMB observations will generate much more data than those today, and so it is worth exploring efficient ways to process the data, even though, on paper, the map-making problem has long been solved.

The time-ordered scanning data is summarized by

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

where $\mathbf{d}$, $\mathbf{m}$, and $\mathbf{n}$ are the vectors of time-ordered data (TOD), the CMB sky-map signal, and measurement noise, and $P$ is the sparse matrix that encodes the telescope's pointing. Of several map-making methods (Tegmark 1997), one of the most common is the method introduced for the Cosmic Background Explorer (COBE, Janssen & Gulkis 1992). This optimal, linear solution is

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

where $\hat{\mathbf{m}}$ provides the generalized least squares minimization of the $\chi^2$ statistic,

$$\chi^2(\mathbf{m}) \equiv (\mathbf{d} - P\mathbf{m})^\dagger N^{-1}(\mathbf{d} - P\mathbf{m}). \tag{3}$$

Here we assume that the noise has zero mean $\langle \mathbf{n} \rangle = \mathbf{0}$, and noise covariance matrix $N = \langle \mathbf{n}\mathbf{n}^\dagger \rangle$ is diagonal in frequency space. Thus map-making is a standard linear regression problem. In the case where the noise is Gaussian, the COBE solution is also the maximum likelihood solution.

With current computation power, we cannot solve for $\hat{\mathbf{m}}$ by calculating $\left(P^\dagger N^{-1} P\right)^{-1} P^\dagger N^{-1}\mathbf{d}$ directly, since the $(P^\dagger N^{-1} P)$ matrix is too large to invert. The noise covariance matrix $N$ is often sparse in frequency domain and the pointing matrix $P$ is sparse in the time-by-pixel domain, and their product is dense. In experiments currently under design, there may be $\sim 10^{16}$ time samples and $\sim 10^9$ pixels, so these matrix inversions are intractable. We can use iterative methods, such as conjugate gradient descent, to avoid the matrix inversions, and execute each matrix multiplication in a basis where the matrix is sparse, using a fast Fourier transform to go between the frequency and time domain.

As an alternative to conjugate gradient descent, Huffenberger & Næss (2018) showed that the "messenger" iterative method could be adapted to solve the linear map-making system, based on the approach from Elsner & Wandelt (2013) to solve the linear Wiener filter. This technique splits the noise covariance into a uniform part and the remainder, and introduces an additional vector that represent the signal plus uniform noise. This messenger field acts as an intermediary between the signal and the data and has a covariance that

is conveniently sparse in every basis. Elsner & Wandelt (2013) also introduced a cooling scheme that takes advantage of the split covariance: over the course of the iterative solution, we adjust the relative weight of the two parts. Starting with the uniform covariance, the modified linear system gradually transforms to the final system, under the control of a cooling parameter. In numerical experiments, Huffenberger & Næss (2018) found that a map produced by the cooled messenger method converged significantly faster than for standard conjugate gradient methods, and to higher fidelity, especially on large scales.

Papež et al. (2018) showed that the the messenger field approach is equivalent to a fixed point iteration scheme, and studied its convergence properties in detail. Furthermore, they showed that the split covariance and the modified system that incorporates the cooling can be solved by other means, including a conjugate gradient technique, which should generally show better convergence properties than the fixed-point scheme. However in numerical tests, Papež et al. (2018) did not find benefits to the cooling modification of the map-making system, in contrast to findings of Huffenberger & Næss (2018).

In this paper, we show that the difference arose because the numerical tests in Papež et al. (2018) used much less low-frequency (or $1/f$) noise than Huffenberger & Næss (2018), and show that the cooling technique improves map-making performance especially when the low-frequency noise is large. This performance boost depends on a proper choice for the pace of cooling. Kodi Ramanah et al. (2017) showed that for Wiener filter the cooling parameter should be chosen as a geometric series. In this work, we give an alternative interpretation of the parameterizing process and show that for map-making the optimal choice (unsurprisingly) is also a geometric series.

In Section 2 we describe our methods for treating the map-making equation and our numerical experiments. In Section 3 we present our results. In Section 4 we interpret the map-making approach. Section 5 is our conclusion. In Appendix A we derive the prescription for our cooling schedule.

## 2. METHODS

### 2.1. *Parameterized Conjugate Gradient Method*

The messenger field approach introduced an extra cooling parameter $\lambda$ to the map-making equation, and solved the linear system with the alternative covariance $N(\lambda) = \lambda\tau I + \bar{N}$. The parameter $\tau$ represents the uniform level of (white) noise in the covariance, $\bar{N}$ is the balance of the noise covariance, and the parameterized covariance equals the original covariance when the cooling parameter $\lambda = 1$. In this work we find it more convenient to work with the reciprocal of cooling parameter $\eta = \lambda^{-1}$ which represents the degree of heteroscedasticity in the covariance

$$N(\eta) = \tau I + \eta\bar{N} \tag{4}$$

which leads to the same system of map-making equations. (This is because $N(\eta) = \lambda^{-1}N(\lambda)$ and the map-making equation (Eq. 5) is insensitive to to scalar multiple of the covariance since is appears on both sides.)

Papež et al. (2018) showed that the conjugate gradient method can be easily applied to the cooled map-making problem. In our notation, this is equivalent to iterating on the parameterized map-making equation

$$\left(P^\dagger N(\eta_i)^{-1}P\right)\hat{\mathbf{m}}(\eta_i) = P^\dagger N(\eta_i)^{-1}\mathbf{d}, \tag{5}$$

as we adjust the parameter through a set of levels $\{\eta_i\}$. Note that the estimated map $\hat{\mathbf{m}}(\eta)$ depends on $\eta$. If we write $\hat{\mathbf{m}}$ without the $\eta$ argument, that means the $\hat{\mathbf{m}}$ in Eq. (2), independent of $\eta$. In our numerical experiments, we confirm that the conjugate gradient approach is converging faster than the fixed point iterations suggested by the messenger map-making method in Huffenberger & Næss (2018). For simplicity we fix the preconditioner to $M = P^\dagger P$ for all of calculations.

When $\eta = 0$, the noise covariance matrix $N(0)$ is homoscedastic, and solution is given by simple binned map $\hat{\mathbf{m}}(0) = \left(P^\dagger P\right)^{-1}P^\dagger\mathbf{d}$, which can be solved directly.

Since the non-white part $\bar{N}$ is the troublesome portion of the covariance. We can think of the $\eta$ parameter as increasing heteroscedasticity of the system, adding a perturbation to the solution achieved at a particular stage, building ultimately upon the initial uniform covariance model. Therefore, this quasi-static process requires $\eta$ increase as $0 = \eta_0 \le \eta_1 \le \cdots \le \eta_{\rm final} = 1$, at which point we arrive at the desired map-making equation, and the solution $\hat{\mathbf{m}}(1) = \hat{\mathbf{m}}$.

We may iterate more than once at each intermediate $\eta_i$: we solve with conjugate gradient iterations using the result from previous calculation $\hat{\mathbf{m}}(\eta_{i-1})$ as the initial value, and move to next parameter $\eta_{i+1}$ when the norm of residual vector

$$\|\mathbf{r}(\eta_i)\| \equiv \left\|P^\dagger N(\eta_i)^{-1}P\,\hat{\mathbf{m}}(\eta_i) - P^\dagger N(\eta_i)^{-1}\mathbf{d}\right\| \tag{6}$$

per pixel is smaller than a tenth of the pixel variance given by the white noise level of the noise model,

$$\frac{\|\mathbf{r}(\eta_i)\|}{\text{totol pixel number}} < 0.1\sigma^2_{\rm pix}. \tag{7}$$

### 2.2. *Analytical expression for* $\{\eta_i\}$ *series*

The next question is how to appropriately choose these monotonically increasing parameters $\eta$. We also want to determine $\eta_1, \cdots, \eta_{n-1}$ before starting conjugate gradient iterations, because the time ordered data $\mathbf{d}$ is very large, and we do not want to keep it in the system memory during calculation. If we determine $\eta_1, \cdots, \eta_{n-1}$ before the iterations, then we can precompute the right-hand side of Eq. (5) for each $\eta_i$ and keep these map-sized objects in memory, instead of the entire time-ordered data.

In the appendix A, we show that a generic good choice for the $\eta$ parameters are the geometric series

$$\eta_i = \min\left\{ (2^i - 1)\frac{\tau}{\max(\bar{N}_f)},\ 1 \right\}, \tag{8}$$

where $\bar{N}_f$ are the eigenvalues of the non-uniform part of the covariance $\bar{N}$ under frequency representation. This is our main result. It tells us not only how to choose parameters $\eta_i$, 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 $\tau/\max(\bar{N}_f) \gg 1$. This tell us that we don't need to use parameterized method at all, because $\eta_0 = 0$ and $\eta_1 = \eta_2 = \cdots = 1$. This corresponds to the standard conjugate gradient method with simple binned map as the initial guess (as recommended by Papež et al. 2018).

### 2.3. *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. (5) could be computed before iterations, since we have determined $\{\eta_i\}$ in advance, so it won't introduce extra computational cost. The most demanding part of conjugate gradient method is calculating its left hand side, because it contains a Fourier transform of $P\mathbf{m}$ from time domain to frequency domain and an inverse Fourier transform of $N(\eta_i)^{-1}P\mathbf{m}$ from frequency domain back to time domain, which is order $\mathcal{O}(n \log n)$ with $n$ being the length of time ordered data. Compare to traditional conjugate gradient method, we swap $N^{-1}$ with $N(\eta)^{-1}$, and it won't add extra cost, since both methods need a fast Fourier transfrom and inverse fast Fourier transform at one iteration. Therefore the computational cost it the same for one step.

In Appendix A our 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 $\eta_i$. So we should update $\eta_i$ to $\eta_{i+1}$ when the map from our calculation $\mathbf{m} \approx \hat{\mathbf{m}}(\eta_i)$. How do we know this condition is satisfied? Since for each new $\eta_i$ value, we are solving a new set of linear equations (Eq. 5), and we could stop calculation and moving to next value $\eta_{i+1}$ when the norm of residual $\|\mathbf{r}(\eta_i)\|$ is small, see Eq. (7). Calculate $||\mathbf{r}(\eta_i)||$ is part of conjugate gradient algorithm, so this won't add extra cost either. Therefore, overall introducing $\eta$ won't have extra computational cost.

### 2.4. *Numerical Simulations*

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

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

We model the noise power spectrum with

$$P(f) = \sigma^2\left(1 + \frac{f_{\rm knee}^\alpha + f_{\rm apo}^\alpha}{f^\alpha + f_{\rm apo}^\alpha}\right) \tag{9}$$

which is white at high frequencies, a power law below the knee frequency, and gives us the option to flatten the low-frequency noise below an apodization frequency (like in Papež et al. 2018). Note that as $f_{\rm apo} \to 0$, $P(f) \to \sigma^2(1 + (f/f_{\rm knee})^{-\alpha})$, and it becomes a $1/f$ noise model.

Dünner et al. (2013) measured the slopes of the atmospheric noise in the Atacama under different water vapor conditions, finding $\alpha = 2.7$ to 2.9. Here we fixed $\sigma^2 = 10\ \mu\mathrm{K}^2$, $\alpha = 3$, and $f_{\rm knee} = 10$ Hz, and change $f_{\rm apo}$ to compare the performance under different noise models.

The noise covariance matrix

$$N_{ff'} = P(f)\frac{\delta_{ff'}}{\Delta_f} \tag{10}$$

is a diagonal matrix in frequency space, where $\Delta_f$ is equal to reciprocal of total scanning time $T \approx 1.05 \times 10^4$ seconds. In our calculations we choose different combi-

[1] The source code and other information are available at https://github.com/Bai-Qiang/map_making_perturbative_approach

nation of $f_{\rm knee}$ and $f_{\rm apo}$, some of the power spectrum are shown in Figure 1.

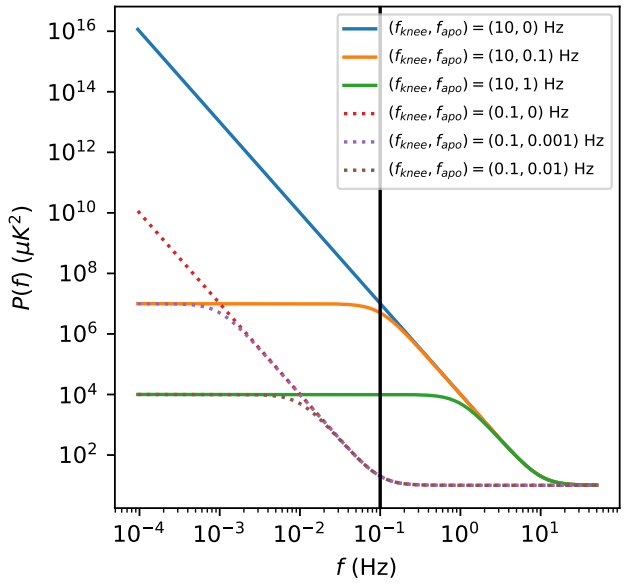


**Figure 1.** The noise power spectrum based on Eq. (9) with white noise level $\sigma^2 = 10$ $\mu$K$^2$ and low-frequency power-law slope $\alpha = 3$. We show two knee frequencies, $f_{\rm knee} = 10$ Hz (solid lines) and $f_{\rm knee} = 0.1$ Hz (dashed lines). For each knee frequency, we have show an unapodized spectrum ($f_{\rm apo} = 0$ Hz), and two apodized ones ($0.1 f_{\rm knee}$ and $0.01 f_{\rm knee}$). The vertical line shows our scanning frequency.

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

## 3. RESULTS

We first compare the vanilla conjugate gradient method with simple preconditioner $P^\dagger P$ versus conjugate gradient with our perturbed linear system. Figure (2) shows the $\chi^2(\mathbf{m})$ results for 1/f noise model ($f_{\rm apo} = 0$) with different knee frequencies. Note that $\chi^2$ in all figures are calculated based on $\chi^2(\mathbf{m})$ in Eq. (3) not $\chi^2(\mathbf{m}, \eta)$ in Eq. (A1). And the $\chi^2_{\rm min}$ is calculated from parameterized conjugate gradient method with 100 $\eta$ values, and it stops when the norm of residual $\|\mathbf{r}(1)\|$ per pixel is smaller than $10^{-10}$, or after 1000 iterations. From Figure (2) we can see for $1/f$ noise model, when $f_{\rm knee} \gtrsim 10 f_{\rm scan}$ the parameterized method starts showing advantage over vanilla conjugate gradient method.

In Figure (3) we fixed $f_{\rm knee} = 10$ Hz, and change $f_{\rm apo}$. When $f_{\rm apo}$ is much smaller than $f_{\rm knee}$ the parameterized conjugate gradient method would performs better. As we increase $f_{\rm apo}$ while fix $f_{\rm knee}$, eventually these two methods perform similar.

If we look at the power spectrum in Figure (1), when $f_{\rm knee}$ is small or $f_{\rm apo}$ is large there are not many large scale low-frequency noise. So we conclude that by introducing $\eta$ parameter could improve perform when there are large low noise contribution.

We also tried different $\alpha$ values. For $\alpha = 2$, the conclusion is the same as $\alpha = 3$. When $\alpha = 1$, there are not many low-frequency noise, the vanilla conjugate gradient is preferred, except some cases with very large knee frequency like $f_{\rm knee} = 100$ Hz and $f_{\rm apo} = 0$ would favor parameterized method. In Papež et al. 2018, the $\alpha = 1$ and the noise power spectrum is apodized at $0.1 f_{\rm knee}$, which corresponds to $f_{\rm apo} \approx 0.1 f_{\rm knee}$, and their knee frequency is the same as scanning frequency, so $f_{\rm knee} = f_{\rm scan} = 0.1$ in our cases. In their case there are not many low-frequency noise, and we confirm that vanilla conjugate gradient method would converge faster.

## 4. DISCUSSION

### 4.1. *Intuitive Interpretation of $\eta$*

There is another way to interpretate the role of $\eta$. Our ultimate goal is to find $\hat{\mathbf{m}}(1)$ which minimizes $\chi^2(\mathbf{m})$ in Eq. (3). Since $N$ is diagonal in frequency space, $\chi^2$ could be written as a sum of all frequency mode $|(\mathbf{d} - P\mathbf{m})_f|^2$ with weight $N_f^{-1}$, such as $\chi^2(\mathbf{m}) = \sum_f |(\mathbf{d} - P\mathbf{m})_f|^2 N_f^{-1}$. The weight is large for low noise frequency mode (small $N_f$), 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 $\boldsymbol{\varepsilon} \equiv \mathbf{d} - P\mathbf{m}$ in the low-noise part.

After introducing $\eta$, we minimize $\chi^2(\mathbf{m}, \eta)$ in Eq. (A1) instead. For $\eta = 0$, $N^{-1}(0) \propto I$ the system is homoscedastic and the estimated map $\hat{\mathbf{m}}(0)$ does not prioritize any frequency mode. As we slowly increase $\eta$, we decrease the weight for the high noise modes, 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 focus most on minimizing the low noise part, such that $\chi^2$ would converge very fast at low noise region, but slowly on high noise part. It may be stuck at some local minimum point and hard to get to global minimum. However by introducing $\eta$ parameter, we let the solver first treat every frequency equally, then as $\eta$ slowly increases, it gradually give more focus to the lowest noise part.

### 4.2. *Other $\eta$ Choices*

Now let us compare the performance difference between choosing $\eta$ parameters based on Eq. (8) and fixing

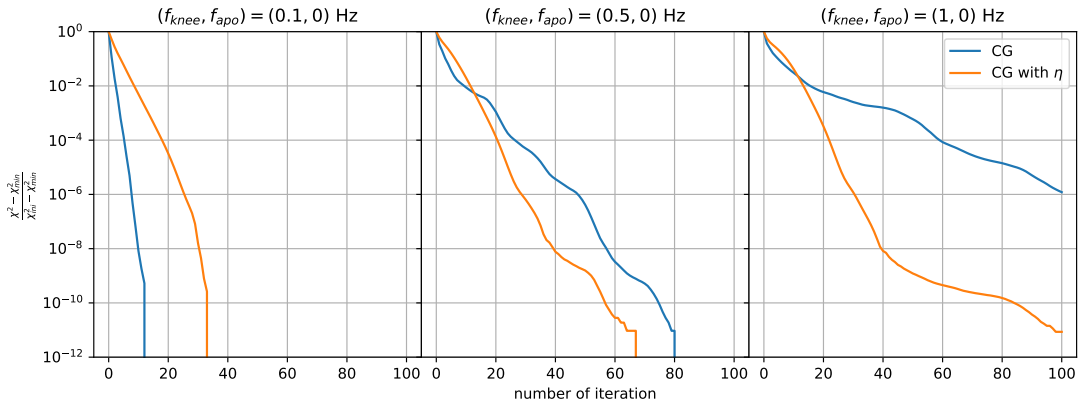


**Figure 2.** Here we show the $\chi^2(\mathbf{m})$ with respect to number of iterations. The vertical axis is rescaled such that all curves start from 1. The map-making equation (Eq. 2) minimize the $\chi^2(\mathbf{m})$, so the curve which goes down fast and get close to zero at the end is the preferred method. In this figure we are comparing traditional conjugate gradient method labeled as *CG* (blue line) with parameterized conjugate gradient labeled as *CG with* $\eta$ (orange line) under different $1/f$ noise model (fixed $f_{\rm apo} = 0$ Hz but different $f_{\rm knee}$). As we can see here when $f_{\rm knee} \gtrsim 10\, f_{\rm scan} = 1$ Hz, there are significant amount of low-frequency noise and the parameterized conjugate gradient method starts showing advantages.

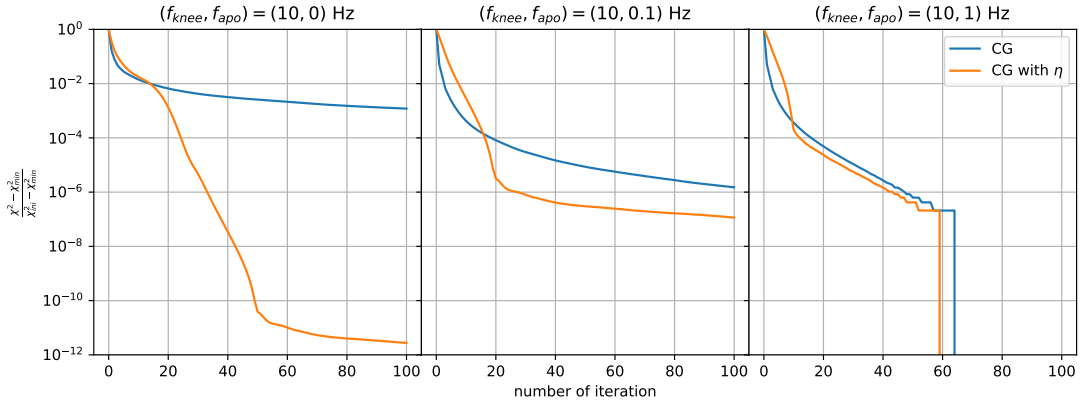


**Figure 3.** The vertical and horizontal axes are the same as in Figure (2), and also compare traditional conjugate gradient method labeled as *CG* (blue line) with parameterized conjugate gradient method labeled as *CG with* $\eta$ (orange line). But here we fix the knee frequency $f_{\rm knee} = 10$ Hz, and change apodized frequency $f_{\rm apo}$. When $f_{\rm apo}$ is much smaller than $f_{\rm knee}$, there are more low-frequency noise and parameterized conjugate gradient method is better than traditional ones.

number of $\eta$ parameters $n_\eta$ manually then choose the $\eta_i$ from a geometric series

$$\eta_i = \eta_1^{\frac{n_\eta - i}{n_\eta - 1}} \quad \text{with } i = 1, 2, \cdots, n_\eta \tag{11}$$

The results are showed in Figure (4).

In some cases the $\eta$ series determined by Eq. (8) is ideal (the first graph in Figure 4), in other cases Eq. (8) gives too many $\eta$ values such that it is not optimal (the second and third graph in Figure 4). So we need to find a way to improve Eq. 8.

### 4.3. *Future Prospects*

In Appendix A, we determine $\delta\eta_m$ value based on the least upper bound of $-\delta\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)/\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)$, and choose $\delta\eta_m$ such that the least upper bound is equal to 1. The reason we use this upper bound instead of using

$$\delta\eta_m = -\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m) / \frac{\mathrm{d}}{\mathrm{d}\eta}\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m) \tag{12}$$

directly, is that we don't want to keep the time ordered data $\mathbf{d}$ in system memory. In Figure (5) we can see if we use Eq. (12), indeed it can improve performance. Especially for the third graph where the power spectrum does not have lots of low-frequency noise. This performance could get close to vanilla conjugate gradient method when there is no significant amount of low-frequency noise, which overcomes the shortcomings of parameterized conjugate gradient method. Therefore, to further improve this method, we need to find more accurate expression for Eq. (A7).

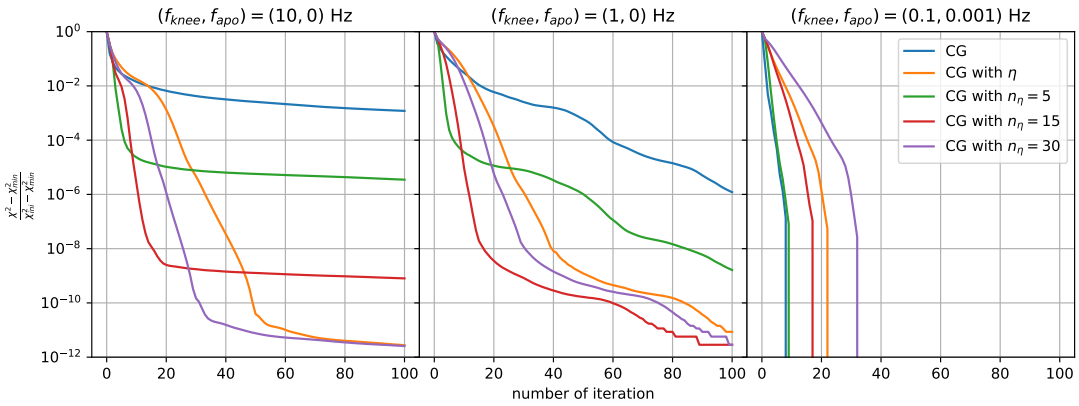


**Figure 4.** The horizontal and vertical axes are the same as in the previous figure. The blue line and the orange line are traditional conjugate gradient method and parameterized conjugate gradient method. For three extra lines, we fix the number of $\eta$ parameter $n_\eta$ manually. The $\eta$ series are determined by Eq. (11). The first graph shows in some cases the $\eta$ series given by Eq. (8) is ideal, but the second and third graph show that Eq. (8) may ends up too many $\eta$ which yields slower convergence.

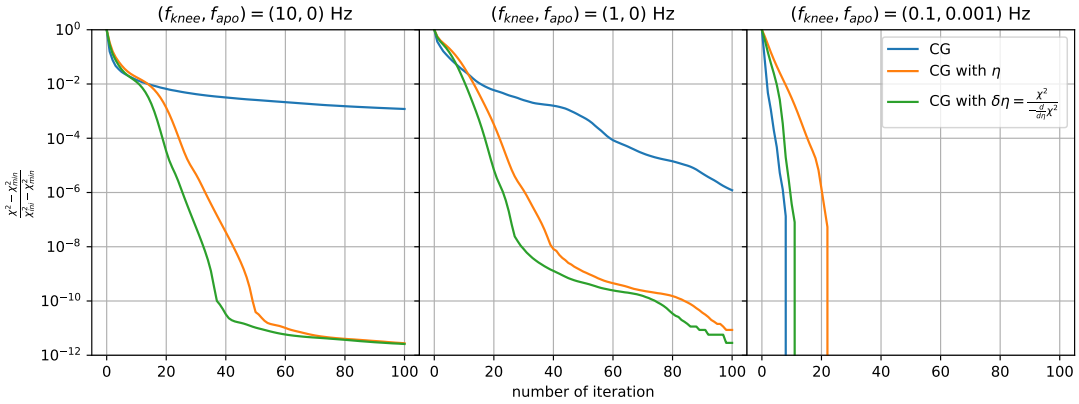


**Figure 5.** The blue line and orange line is the same as in Figure (4) for reference. The extra green line shows the result when $\delta\eta_m$ is determined from Eq. (12) not from Eq. (8). This shows that if we could update based on exact expression Eq. (12), it could converge even faster. Especially in the third graph it would overcome the shortcomings of parameterized conjugate gradient method.

## 5. CONCLUSIONS

We presented a parameterized conjugate gradient method with parameter $\eta$ based on the idea of messenger field separating the white noise out of noise covariance matrix. Then we gave an analytical expression for $\eta$ series, and showed that this method would not introduce extra computational cost than traditional conjugate method.

We tested this method under different power spectrum both apodized and non-apodized. The results showed that this method is faster than traditional conjugate gradient method when there are significant amount of low-frequency noise. But it could be further improved if we could get more accurate estimation for Eq. (12), either before iteration or without using time ordered data during iteration.

Also note that we fixed preconditioner as $M = P^\dagger P$ during our calculation, this parameterizing process could be applied to any preconditioner and possibly improve performance when there are significant amount of low-frequency noise.

Papež et al. (2018) showed that the messenger field method solving Wiener filter problem introduced by Elsner & Wandelt (2013) could also be written as parameterized conjugate gradient algorithm. Then Kodi Ramanah et al. (2017) introduced dual messenger field method to Wiener filter. If applying our idea to Wiener filter problem, hopefully, it may also bring improvements.

BQ and KH are supported by NSF award 1815887.

# APPENDIX

## A. THE DERIVATION OF $\eta$ SERIES

We know that the initial degree of heteroscedasticity $\eta_0 = 0$, which means the system is homoscedastic. What would be good value for the next parameter $\eta_1$? To simplify notation, we use $N_\eta$ to denote $N(\eta) = \tau I + \eta\bar{N}$. For some specific $\eta$ value, the minimum $\chi^2$ value is given by the optimized map $\hat{\mathbf{m}}(\eta) = \left(P^\dagger N_\eta^{-1} P\right)^{-1} P^\dagger N_\eta^{-1}\mathbf{d}$, which minimizes

$$\chi^2(\mathbf{m}, \eta) = \big(\mathbf{d} - P\mathbf{m}\big)^\dagger N_\eta^{-1}\big(\mathbf{d} - P\mathbf{m}\big). \tag{A1}$$

with $\eta$ being fixed. We restrict to the case that the noise covariance matrix $N$ is diagonal in the frequency domain, and represent the frequency-domain eigenvalues as $N_f$.

Let us first consider $\eta_1 = \eta_0 + \delta\eta = \delta\eta$ such that $\eta_1 = \delta\eta$ is very small quantity, $\delta\eta \ll 1$. Since $\hat{\mathbf{m}}(\eta)$ minimizes $\chi^2(\mathbf{m}, \eta)$ with $\eta$ being fixed, we have $\frac{\partial}{\partial\hat{\mathbf{m}}}\chi^2(\hat{\mathbf{m}}(\eta), \eta) = 0$, and using chain rule

$$\frac{\mathrm{d}}{\mathrm{d}\eta}\chi^2(\hat{\mathbf{m}}(\eta), \eta) = \frac{\partial}{\partial\eta}\chi^2(\hat{\mathbf{m}}(\eta), \eta) = -(\mathbf{d} - P\hat{\mathbf{m}}(\eta))^\dagger N_\eta^{-1}\bar{N}N_\eta^{-1}(\mathbf{d} - P\hat{\mathbf{m}}(\eta)) \tag{A2}$$

Then the fractional decrease of $\chi^2(\hat{\mathbf{m}}(0), 0)$ from $\eta_0 = 0$ to $\eta_1 = \delta\eta$ is

$$-\frac{\delta\chi^2(\hat{\mathbf{m}}(0), 0)}{\chi^2(\hat{\mathbf{m}}(0), 0)} = -\delta\eta\frac{\frac{\mathrm{d}}{\mathrm{d}\eta}\chi^2(\hat{\mathbf{m}}(0), 0)}{\chi^2(\hat{\mathbf{m}}(0), 0)} = \delta\eta\frac{1}{\tau}\frac{(\mathbf{d} - P\hat{\mathbf{m}}(0))^\dagger\bar{N}(\mathbf{d} - P\hat{\mathbf{m}}(0))}{\big(\mathbf{d} - P\hat{\mathbf{m}}(0)\big)^\dagger\big(\mathbf{d} - P\hat{\mathbf{m}}(0)\big)} \tag{A3}$$

Here we put a minus sign in front of this expression such that it's non-negative, and use $N_{\eta=0} = \tau I$ at the second equality. Since it is hard to analyze $\mathbf{d} - P\hat{\mathbf{m}}$ under frequency domain, we treat it as an arbitrary vector, then the least upper bound is given by

$$-\frac{\delta\chi^2(\hat{\mathbf{m}}(0), 0)}{\chi^2(\hat{\mathbf{m}}(0), 0)} \leq \frac{\delta\eta}{\tau}\max(\bar{N}_f) \tag{A4}$$

where $\max(\bar{N}_f)$ is the maximum eigenvalue of $\bar{N}$. We want $\big|\delta\chi^2(\hat{\mathbf{m}}(0), 0)\big| = \chi^2(\hat{\mathbf{m}}(0), 0)) - \chi^2(\hat{\mathbf{m}}(\eta_1), \eta_1)$ to be large such that it converge fast. Let's say $\chi^2(\hat{\mathbf{m}}(\eta_1), \eta_1)$ is much smaller than $\chi^2(\hat{\mathbf{m}}(0), 0)$, or $\chi^2(\hat{\mathbf{m}}(\eta_1), \eta_1) \ll \chi^2(\hat{\mathbf{m}}(0), 0)$. We would expect

$$-\frac{\delta\chi^2(\hat{\mathbf{m}}(0), 0)}{\chi^2(\hat{\mathbf{m}}(0), 0)} = 1 - \frac{\chi^2(\hat{\mathbf{m}}(\eta_1), \eta_1)}{\chi^2(\hat{\mathbf{m}}(0), 0)} \approx 1^- \tag{A5}$$

The upper bound is strictly smaller than 1. Combining Eq. (A4) and Eq. (A5) we can choose $\delta\eta$ such that the least upper bound is equal to 1. Thus we have

$$\eta_1 = \frac{\tau}{\max(\bar{N}_f)} = \frac{\min(N_f)}{\max(N_f) - \min(N_f)}. \tag{A6}$$

Here $N_f$ and $\bar{N}_f$ are the eigenvalues of $N$ and $\bar{N}$ in the 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 $\eta_m$ with $m > 1$? We use a similar analysis, letting $\eta_{m+1} = \eta_m + \delta\eta_m$ with a small $\delta\eta_m \ll 1$. First, let us find the least upper bound

$$-\frac{\delta\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)}{\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)} = \delta\eta_m\frac{(\mathbf{d} - P\hat{\mathbf{m}}(\eta_m))^\dagger N_{\eta_m}^{-1}\bar{N}N_{\eta_m}^{-1}(\mathbf{d} - P\hat{\mathbf{m}}(\eta_m))}{\big(\mathbf{d} - P\hat{\mathbf{m}}(\eta_m)\big)^\dagger N_{\eta_m}^{-1}\big(\mathbf{d} - P\hat{\mathbf{m}}(\eta_m)\big)} \tag{A7}$$

$$\leq \delta\eta_m\ \max\left(\frac{\bar{N}_f}{\tau + \eta_m\bar{N}_f}\right) \tag{A8}$$

The upper bound in the second line is a little bit tricky. Both matrix $\bar{N}$ and $N_{\eta_m}^{-1}$ can be simultaneously diagonalized in frequency space. For each eigenvector $\mathbf{e}_f$, the corresponding eigenvalue of the matrix on the numerator $N_{\eta_m}^{-1}\bar{N}N_{\eta_m}^{-1}$ is $\lambda_f = \bar{N}_f(\tau + \eta_m\bar{N}_f)^{-2}$, and the eigenvalue for matrix on the denominator $N_{\eta_m}^{-1}$ is $\gamma_f = (\tau + \eta_m\bar{N}_f)^{-1}$. Their eigenvalues are related by $\lambda_f = [\bar{N}_f/(\tau + \eta_m\bar{N}_f)]\gamma_f$. For any vector $\mathbf{v} = \sum_f \alpha_f \mathbf{e}_f$, we have

$$\frac{\mathbf{v}^\dagger N_{\eta_m}^{-1}\bar{N}N_{\eta_m}^{-1}\mathbf{v}}{\mathbf{v}^\dagger N_{\eta_m}^{-1}\mathbf{v}} = \frac{\sum_f \alpha_f^2\lambda_f}{\sum_f \alpha_f^2\gamma_f} = \frac{\sum_f \alpha_f^2\gamma_f\bar{N}_f/(\tau+\eta_m\bar{N}_f)}{\sum_f \alpha_f^2\gamma_f} \leq \max\left(\frac{\bar{N}_f}{\tau+\eta_m\bar{N}_f}\right). \tag{A9}$$

Again assuming $\chi^2(\hat{\mathbf{m}}(\eta_{m+1}), \eta_{m+1}) \ll \chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)$, which we expect it to be satisfied for $\eta_m \ll 1$. That is because if $\eta \approx 1$, then $\chi^2(\hat{\mathbf{m}}(\eta), \eta)$ would also close to the minimum $\chi^2$ which means the next $\chi^2$ value would be close to current one. Since the final result Eq. (A13) is a geometric series, only the last few $\eta_m$ values fail to satisfy this condition. Similarly, we could set the least upper bound equal to 1. Then we get

$$\delta\eta_m = \min\left(\frac{\tau+\eta_m\bar{N}_f}{\bar{N}_f}\right) = \eta_m + \frac{\tau}{\max(\bar{N}_f)}. \tag{A10}$$

Therefore

$$\eta_{m+1} = \eta_m + \delta\eta_m = 2\eta_m + \frac{\tau}{\max(\bar{N}_f)} \tag{A11}$$

The final term $\tau/\max(\bar{N}_f) = \eta_1$ becomes subdominant after a few terms, and we see that the $\eta_m$ increase like a geometric series. If written in the form $\eta_{m+1} + \tau/\max(\bar{N}_f) = 2\big(\eta_m + \tau/\max(\bar{N}_f)\big)$ it's easy to see that for $m \geq 1$, $\eta_m + \tau/\max(\bar{N}_f)$ forms a geometric series

$$\eta_m + \frac{\tau}{\max(\bar{N}_f)} = \left(\eta_1 + \frac{\tau}{\max(\bar{N}_f)}\right)2^{m-1} = \frac{\tau}{\max(\bar{N}_f)}2^m \tag{A12}$$

where we used $\eta_1 = \tau/\max(\bar{N}_f)$. Note that $m = 0$ and $\eta_0 = 0$ also satisfy this expression and we've got final expression for all $\eta_m$

$$\eta_m = \min\left\{1,\ \frac{\tau}{\max(\bar{N}_f)}(2^m - 1)\right\} \tag{A13}$$

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

## REFERENCES


Dünner, R., Hasselfield, M., Marriage, T. A., et al. 2013, ApJ, 762, 10, doi: 10.1088/0004-637X/762/1/10

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

Janssen, M. A., & Gulkis, S. 1992, in NATO Advanced Science Institutes (ASI) Series C, ed. M. Signore & C. Dupraz, Vol. 359 (Springer), 391–408

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. 1997, ApJL, 480, L87, doi: 10.1086/310631