

## Cosmic Microwave Background map-making solutions improve with cooling

BAI-QIANG QIANG (KMH: WANT CHINESE CHARACTERS?)<sup>1</sup> AND KEVIN M. HUFFENBERGER <sup>1</sup>

<sup>1</sup>Department of Physics, Florida State University, Tallahassee, Florida 32306

### ABSTRACT

In the context of the Cosmic Microwave Background, we study the solution to the equation that transforms scanning data into a map. We show that splitting the noise covariance into two parts, as suggested by “messenger” methods for solving linear systems, is particularly effective when there is significant low-frequency noise in the timestream. A conjugate gradient algorithm applied to the modified system converges faster and to a higher fidelity solution than the standard approach, for the same computational cost per iteration. We give an analytical expression for the parameter that controls how gradually the non-uniform noise is switched on 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 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

$$\mathbf{d} = P\mathbf{m} + \mathbf{n} \quad (1)$$

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 mapmaking methods (Tegmark 1997a), one of the most common is the method introduced for the Cosmic Background Explorer (COBE, Janssen & Gulkis 1992). This optimal, linear solution is

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

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

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

Here we assume that the noise has zero mean  $\langle \mathbf{n} \rangle = \mathbf{0}$ , and noise covariance matrix could be written as  $N =$

$\langle \mathbf{n}\mathbf{n}^\dagger \rangle$ . We cast mapmaking as a standard linear regression problem. In case 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  $(P^\dagger N^{-1} P)^{-1} P^\dagger N^{-1} \mathbf{d}$  directly, since the  $(P^\dagger N^{-1} P)$  matrix is too large to invert. The noise covariance matrix  $N$  is 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. Therefore we use iterative methods, such as conjugate gradient descent, to avoid the matrix inversions, while executing 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 technique, Huffenberger & Næss (2018) showed that the “messenger method” could be adapted to solve the linear mapmaking 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, over the course of the iterative solution, it adjusts the relative weight of those two parts. Starting with the uniform covariance, the modified linear system gradually transforms to the final system via a cooling parameter. In numerical experiments, Huffenberger & Næss (2018) found that the large scales of map produced by the cooled messenger method converged significantly faster than for standard methods, and to higher fidelity.

Papež et al. (2018) showed that the splitting of the covariance in the messenger field approach is equivalent to a fixed point iteration scheme, and studied its convergence properties in detail. Furthermore, they showed that the modified system that incorporates the cooling scheme 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 linear system, in contrast to findings of Huffenberger & Naess (2018).

In this paper, we show that the difference arose because the tests in Papež et al. (2018) used much less low-frequency ( $1/f$ ) noise, and show that the cooling technique improves mapmaking 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 mapmaking equation and our numerical experiments. In Section 3 we present our results. In Section 4 we interpret the mapmaking approach and its computational cost. In Section 5 we conclude. In appendices we derive how we set our cooling schedule.

## 2. METHODS

### 2.1. Parameterized Conjugate Gradient Method

The messenger field approach introduced an extra cooling parameter  $\lambda$  to 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, 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 inverse cooling parameter  $\eta = \lambda^{-1}$  and define the covariance as

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

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

Papež et al. (2018) showed that the conjugate gradient method can be easily applied to parameterized map-

making equation by iterating on

$$P^\dagger N(\eta)^{-1} P \hat{\mathbf{m}} = P^\dagger N(\eta)^{-1} \mathbf{d} \quad (5)$$

as the cooling is adjusted. In our numerical experiments, we confirm that the conjugate gradient approach is converging faster than the fixed point iterations suggested by the messenger mapmaking method in Huffenberger & Naess (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 proportional to identity matrix  $I$ , and solution is given by simple binned map  $\mathbf{m}_0 = (P^\dagger P)^{-1} P^\dagger \mathbf{d}$ , which can be solved directly. From this starting point, the cooling scheme requires the inverse cooling parameter  $\eta$  increase as  $0 = \eta_0 \leq \eta_1 \leq \dots \leq \eta_{\text{final}} = 1$ , at which point we arrive at the desired mapmaking equation. For some intermediate  $\eta_i$ , we use the conjugate gradient method to solve equation  $(P^\dagger N(\eta_i)^{-1} P) \hat{\mathbf{m}}(\eta_i) = P^\dagger N(\eta_i)^{-1} \mathbf{d}$ , using  $\hat{\mathbf{m}}(\eta_{i-1})$  as the initial value. KMH: In this description, it is not totally clear whether you intend to update the eta after every iteration.

The non-white part  $\bar{N}$  is the troublesome portion of the covariance, and we can think of the  $\eta$  parameter as turning it on slowly, adding a perturbation to the solution achieved at a particular stage, building ultimately upon the initial uniform covariance model.

### 2.2. Choice of inverse cooling parameters $\eta$

The next question is how we choose these monotonically increasing parameters  $\eta$ . If we choose them inappropriately, the solution converge slowly, because we waste effort converging on the wrong system. We also want to determine  $\eta_1, \dots, \eta_{n-1}$  before starting conjugate gradient iterations. 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  $\eta_1, \dots, \eta_{n-1}$  before the iterations, then we can precompute the right-hand side  $P^\dagger N(\eta)^{-1} \mathbf{d}$  for each  $\eta_i$  and keep these map-sized objects in memory, instead of the entire time-ordered data.

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

$$\eta_i = \min \left\{ (2^i - 1) \frac{\tau}{\max(\bar{N}_f)}, 1 \right\}, \quad (6)$$

where  $\bar{N}_f$  is the frequency representation of the non-uniform part of the covariance. This is the 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

169  $\tau/\max(\bar{N}_f) \gg 1$ . This tell us that we don't need to  
 170 use parameterized method at all, because  $\eta_0 = 0$  and  
 171  $\eta_1 = \eta_2 = \dots = 1$ . This corresponds to the standard  
 172 conjugate gradient method with simple binned map as  
 173 the initial guess (as recommended by Papež et al. 2018).

### 2.3. Numerical Simulations

174  
 175 To compare these algorithms, we need to do some sim-  
 176 ple simulation of scanning processes, and generate time  
 177 ordered data from random sky signal.<sup>1</sup> Our sky is a  
 178 small rectangular area, with two orthogonal directions  
 179  $x$  and  $y$ , both with range from  $-1^\circ$  to  $+1^\circ$ . The signal  
 180 has first three stokes parameters ( $I, Q, U$ ).

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

189 The noise power spectrum is given by

$$P(f) = \sigma^2 \left( 1 + \frac{f_{\text{knee}}^\alpha + f_{\text{apo}}^\alpha}{f^\alpha + f_{\text{apo}}^\alpha} \right) \quad (7)$$

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

$$N_{ff'} = P(f) \frac{\delta_{ff'}}{\Delta_f} \quad (8)$$

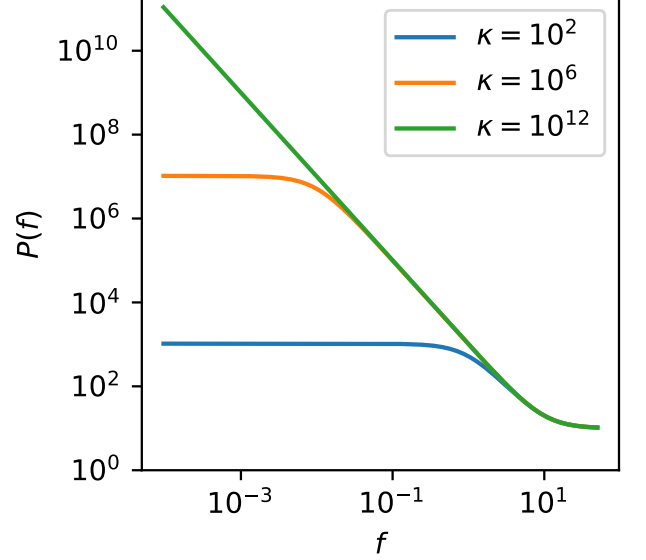
195 is a diagonal matrix in frequency space, where  $\Delta_f$  is  
 196 equal to reciprocal of total scanning time  $T$ . In our  
 197 calculations we choose the  $f_{\text{apo}}$  such that the condition  
 198 numbers  $\kappa$  are  $10^2$ ,  $10^6$ , and  $10^{12}$ . The corresponding  
 199 power spectrum are shown in Figure(1).

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

202 **KMH: Compare to the noise power spectrum of Papez.**  
 203 **Remark how little  $1/f$  is in their test. What is the effect**  
 204 **of changing the noise slope?**

## 3. RESULTS

205  
 206 First let's compare the results with vanilla conjugate  
 207 gradient method with simple preconditioner  $P^\dagger P$ . The  
 208 results are showed in Figure.(2) for different kinds of



**Figure 1.** The noise power spectrum based on Eq.(7) with  $\sigma^2 = 10 \mu\text{K}^2$ ,  $\alpha = 2$  and  $f_{\text{knee}} = 10$  Hz. And fixing the condition number  $\kappa$  of noise covariance matrix Eq.(8) by choosing  $f_{\text{apo}}$ .

213 noise power spectra. Here note that  $\chi^2$  in all figures are  
 214 calculated based on Eq.(3) not  $\chi^2(\mathbf{m}, \eta)$  in Eq.(??). The  
 215  $\chi^2_{\text{min}}$  is calculated from perturbative conjugate gradient  
 216 method with more intermediate  $\eta$  values, and more it-  
 217 erations after  $\eta = 1$ .

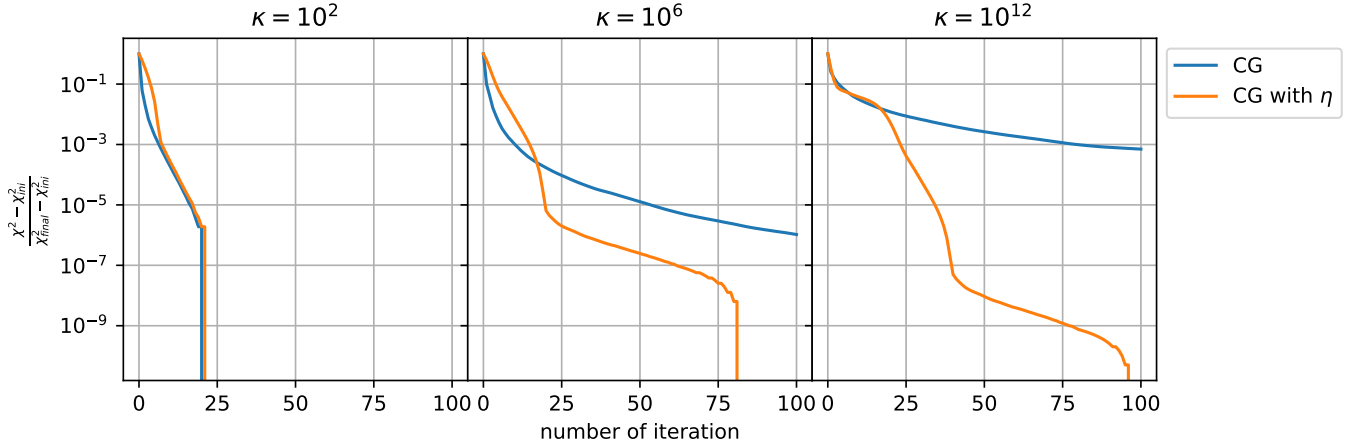
218 As we can see in the left graph in Figure(2), when  
 219 the condition number of noise covariance matrix  $\kappa(N)$   
 220 is small, the performance between different these two  
 221 methods are small. The vanilla conjugate gradient  
 222 method converge faster, because its perturbation pa-  
 223 rameter goes to 1 at the first iteration, however for the  
 224 perturbation method its  $\eta$  value will slowly reach 1 in  
 225 about ten iterations.

226 Notice that as we increase  $\kappa(N)$ , or equivalently de-  
 227 crease  $f_{\text{apo}}$ , the perturbation parameter  $\eta$  starts showing  
 228 its benefits, as showed in the second and third graph in  
 229 Figure(2). It outperforms the vanilla conjugate gradient  
 230 method when  $f_{\text{apo}} \approx 0$  and the noise power spectrum be-  
 231 comes the  $1/f$  noise model, which usually is the intrinsic  
 232 noise of instruments (Tegmark (1997b)).

233 Now let us compare the performance difference  
 234 between choosing  $\eta$  parameters based on Eq.(A6)  
 235 and manually fixing number of  $\eta$  parameters  $n_\eta$   
 236 manually. We manually choose the  $\eta_i$  values us-  
 237 ing function `numpy.logspace(start=ln( $\eta_1$ ), stop=0,`  
 238 `num= $n_\eta$ , base= $e$ )`. The results are showed in Figure(3).

239 When  $\kappa(N)$  is small, and Eq.(A6) tells us that only a  
 240 few  $\eta$  parameters are good enough, see the orange line  
 241 in the first Figure(3), where we have  $\sim 10$   $\eta$  levels. If

<sup>1</sup> The source code and other information are available at [https://github.com/Bai-Qiang/map\\_making\\_perturbative\\_approach](https://github.com/Bai-Qiang/map_making_perturbative_approach)



**Figure 2.** These three figures show the  $\frac{\chi^2(\mathbf{m}) - \chi^2_{\min}}{\chi^2_{\min} - \chi^2_{\text{ini}}}$  changes for each iteration under different noise covariance matrix with condition number being  $10^2$ ,  $10^6$ , and  $10^{12}$ .

242 unfortunately we choose  $n_\eta$  being large value, like 15 or  
 243 30, then it will ends up converge slowly, because it needs  
 244 at least 15 or 30 iterations to reach  $\eta = 1$ , at least one  
 245 iteration per  $\eta$  level.

246 On the other hand if  $\kappa(N)$  is very large and the power  
 247 spectrum is  $1/f$  noise, we need more  $\eta$  parameters. If  
 248  $n_\eta$  is too small, for example  $n_\eta = 5$  the green line in last  
 249 Figure(3), it may be better than the vanilla conjugate  
 250 gradient method, but it is still far from optimal.

## 251 4. DISCUSSION

### 252 4.1. Intuitive Interpretation of $\eta$

253 **KMH: most of this is pretty similar to discussion in**  
 254 **Huffenberger and Naess. The last paragraph is new.**

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

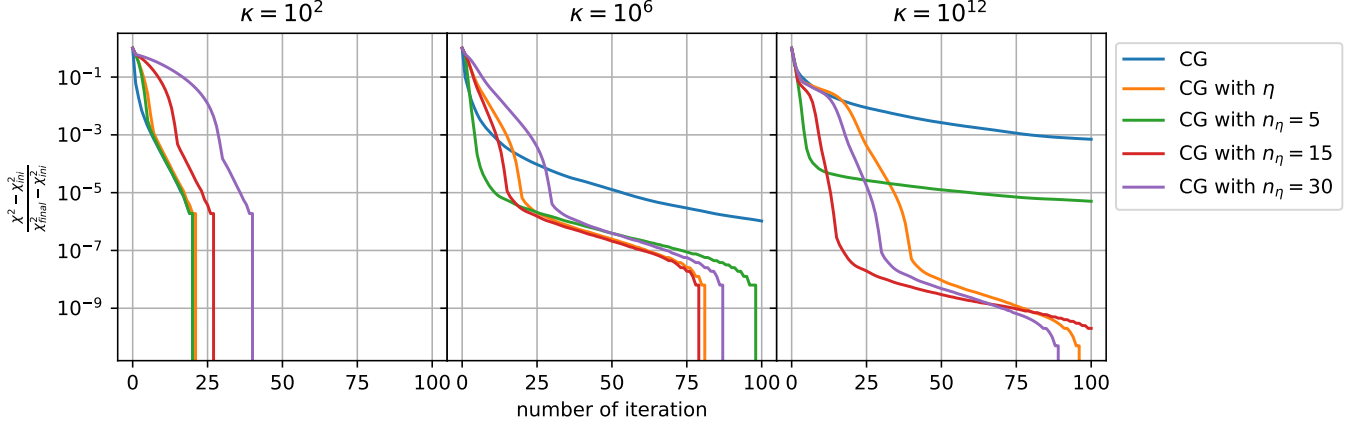
266 After introducing  $\eta$ , we minimize  $\chi^2(\mathbf{m}, \eta) = (\mathbf{d} -$   
 267  $P\mathbf{m})^\dagger N_\eta^{-1}(\mathbf{d} - P\mathbf{m})$ . For  $\eta = 0$ ,  $N_{\eta=0}^{-1} \propto I$  and the esti-  
 268 mated map  $\hat{\mathbf{m}}(\eta = 0)$  does not prioritize any frequency  
 269 mode. As we slowly increase  $\eta$ , we decrease the weight  
 270 for the frequency modes which have large noise, and fo-  
 271 cusing minimizing error for low noise part. If we start  
 272 with  $\eta_1 = 1$  directly, which corresponds to the vanilla  
 273 conjugate gradient method, then the entire conjugate  
 274 gradient solver will focus most on minimizing the low  
 275 noise part, such that  $\chi^2$  would converge very fast at low

276 noise region, but slowly on high noise part. **Since it**  
 277 **focus on low noise part only, it may be stuck at some**  
 278 **local minimum point.** However by introducing  $\eta$  param-  
 279 eter, we let the solver first treat every frequency equally.  
 280 Then as  $\eta$  slowly increases, it gradually shifts focus from  
 281 the highest noise to the lowest noise part. **KMH: I feel**  
 282 **what this is missing is why the high-noise modes get**  
 283 **stuck though.**

284 If we write the difference between final and  
 285 initial  $\chi^2$  value as  $\chi^2(\hat{\mathbf{m}}(1), 1) - \chi^2(\hat{\mathbf{m}}(0), 0) =$   
 286  $\int_0^1 d\eta \frac{d}{d\eta} \chi^2(\hat{\mathbf{m}}(\eta), \eta)$ , and use Eq.(B7). We note that  
 287 when  $\eta$  is very small, the  $\frac{d}{d\eta} \chi^2(\hat{\mathbf{m}}(\eta), \eta)$  would have rel-  
 288 atively large contribution from medium to large noise  
 289 region, comparing to large  $\eta$ . So introducing  $\eta$  might  
 290 improve the convergence of  $\chi^2$  at these regions, because  
 291 the vanilla conjugate gradient method only focuses on  
 292 the low noise part and it may have difficulty at these  
 293 regions.

### 294 4.2. Computational Cost

295 To properly compare the performance cost of this  
 296 method with respect to vanilla conjugate gradient  
 297 method with simple preconditioner, we need to com-  
 298 pare their computational cost at each iteration. The  
 299 right hand side of parameterized map-making equation  
 300 Eq.(5) could be computed before iterations, so it won't  
 301 introduce extra computational cost. The most demand-  
 302 ing part of conjugate gradient method is calculating  
 303  $P^\dagger N^{-1} P \hat{\mathbf{m}}$ , because it contains a Fourier transform of  
 304  $P \hat{\mathbf{m}}$  from time domain to frequency domain and an in-  
 305 verse Fourier transform of  $N^{-1} P \hat{\mathbf{m}}$  from frequency do-  
 306 main back to time domain, which is order  $\mathcal{O}(n \log n)$   
 307 with  $n$  being the length of time ordered data. If we  
 308 change  $N^{-1}$  to  $N(\eta)^{-1}$ , it won't add extra cost, since



**Figure 3.** The blue line and the orange line are the same as Figure.(2). For three extra lines, we fix the number of  $\eta$  parameter  $n_\eta$  manually. Instead of using Eq.(A6), we use `numpy.logspace(start=ln( $\eta_1$ ), stop=0, num= $n_\eta$ , base=e)` to get all  $\eta$  parameters.

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  $\eta_i$ . So We should update  $\eta_i$  to  $\eta_{i+1}$  when  $\mathbf{m} \approx \hat{\mathbf{m}}(\eta_i)$ . How do we know this condition is satisfied? Since for each new  $\eta_i$  value, we are solving a new set of linear equations  $A(\eta_i)\hat{\mathbf{m}} = \mathbf{b}(\eta_i)$  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  $\eta_{i+1}$  when the norm of residual  $\|\mathbf{r}(\eta_i)\| = \|\mathbf{b}(\eta_i) - A(\eta_i)\hat{\mathbf{m}}\|$  is 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  $\eta$  won't have extra computational cost.

## 5. CONCLUSIONS

**KMH: some of this future prospects should move to discussion** As you may have noticed in the second and third Figure(3), the perturbation parameter based on Eq.(A6) is more than needed, especially for  $1/f$  noise case. For the case  $\kappa = 10^{12}$ , we notice that based on Eq.(A6) it gives us  $n_\eta \approx 40$ , however from  $\chi^2$  result in the last Figure(3)  $n_\eta \approx 30$  or even  $n_\eta \approx 15$  is good enough. Also, for the nearly-white-noise case, we could certainly choose  $n_\eta = 1$  such that  $\eta_1 = 1$  which corresponds to vanilla conjugate gradient method, based on  $\chi^2$  result in first Figure(3). However Eq.(A6) gives us  $n_\eta \approx 6$ , even though it does not make the final  $\chi^2$  result much different at the end.

Is it possible to further improve the analysis, such that it produces smaller  $n_\eta$ ? Let's examine how we get  $\eta_i$  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.(A2). For  $\eta \neq 0$ , the upper bound is

$$\delta\eta \frac{\hat{\mathbf{r}}_\eta^\dagger N(\eta)^{-1} \bar{N} N(\eta)^{-1} \hat{\mathbf{r}}_\eta}{\hat{\mathbf{r}}_\eta^\dagger N(\eta)^{-1} \hat{\mathbf{r}}_\eta} \leq \frac{\delta\eta}{\eta + \frac{\tau}{\max(N_f) - \tau}} \quad (9)$$

with  $\mathbf{r}_\eta = [1 - P(P^\dagger N(\eta)^{-1}P)^{-1}P^\dagger N(\eta)^{-1}]\mathbf{d} \equiv \mathcal{P}_\eta \mathbf{d}$ . 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 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  $(P^\dagger N(\eta)^{-1}P)^{-1}$ . Note that we have to determine all of  $\eta$  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,  $\left| \frac{d}{d\eta} \chi^2(\hat{\mathbf{m}}(\eta), \eta) / \chi^2(\hat{\mathbf{m}}(\eta), \eta) \right|$  won't get close to the upper bound  $1/\left(\eta + \frac{\tau}{\max(N_f) - \tau}\right)$ . Based on the analysis in Section(4.1), for small  $\eta$  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  $\eta$  values. Which means if we could somehow know the frequency distribution of  $\mathbf{r}_\eta$ , we could tighten the boundary of  $\left| \frac{d}{d\eta} \chi^2(\hat{\mathbf{m}}(\eta), \eta) / \chi^2(\hat{\mathbf{m}}(\eta), \eta) \right|$ , and get larger  $\delta\eta$  value. This should make  $\eta$  goes to 1 faster, and yields the fewer  $\eta$  parameters we need.

Also notice that the  $\eta$  values determined from Eq.(A6) are not dependent on any scanning information, it only depends on noise power spectrum  $P(f)$ , or noise covariance matrix  $N$ . In Appendix C we would show two examples with same parameters as in Figure(3) except scanning frequency  $f_{\text{scan}}$ . It turns out the  $\eta$  values should somehow depends on scanning scheme. Again that's be-



cause when we determine the upper bound we treated  $\mathbf{r}_\eta$  as an arbitrary vector, such that we lose all information related to scanning scheme in the pointing matrix  $P$ .

**KMH: We need some discussion of the things that haven't yet been demonstrated with the PCG, like multiple messenger fields. Has the Kodi-Ramanah dual messenger field scheme been demonstrated in a PCG scheme by Papez?**

Even though the perturbation parameter  $\eta$  get from Eq.(A6) 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  $\eta_{i+1}$ .

Also this analysis of  $\eta$  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. Better preconditioners would also lead to improvements.

BQ and KH are supported by NSF award 1815887.

## APPENDIX

### A. THE SEQUENCE OF INVERSE COOLING PARAMETERS

We know that the initial inverse cooling parameter  $\eta_0 = 0$ . 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  $\hat{\mathbf{m}}(\eta)$ ,

$$\chi^2(\hat{\mathbf{m}}(\eta), \eta) = (\mathbf{d} - P\hat{\mathbf{m}}(\eta))^\dagger N_\eta^{-1} (\mathbf{d} - P\hat{\mathbf{m}}(\eta)) \quad (\text{A1})$$

We restrict to the case that the noise covariance matrix  $N$  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

$$-\frac{\delta\chi^2(\hat{\mathbf{m}}(0), 0)}{\chi^2(\hat{\mathbf{m}}(0), 0)} = \delta\eta \frac{1}{\tau} \frac{(\mathbf{d} - P\hat{\mathbf{m}}(0))^\dagger \bar{N} (\mathbf{d} - P\hat{\mathbf{m}}(0))}{(\mathbf{d} - P\hat{\mathbf{m}}(0))^\dagger (\mathbf{d} - P\hat{\mathbf{m}}(0))} \quad (\text{A2})$$

Here we put a minus sign in front of this expression such that it's non-negative. **KMH: This statement is kind of non trivial... I think you need to explain more the step  $(\tau I + \delta\eta \bar{N})^{-1} \approx \tau I - \delta\eta \bar{N}$ . (I think that's right, even if you don't restrict to the frequency diagonal case.)**

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  $\chi^2$  at next iteration. Here if we assume that initial  $\chi^2$  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  $|\delta\chi^2(\hat{\mathbf{m}}(0), 0)/\chi^2(\hat{\mathbf{m}}(0), 0)| \approx 1^-$ , strictly smaller than 1. To make sure it will not start too fast, we could set its upper bound equal to 1,  $\delta\eta \max(\bar{N}_f)/\tau = 1$  **KMH: explain this reasoning a bit better.** This gives

$$\eta_1 = \frac{\tau}{\max(\bar{N}_f)} = \frac{\min(N_f)}{\max(N_f) - \min(N_f)} \quad (\text{A3})$$

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  $\eta_m$  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 B for detailed derivation. We would get

$$\delta\eta_m = \min \left( \frac{\tau + \eta_m \bar{N}_f}{\bar{N}_f} \right) = \eta_m + \frac{\tau}{\max(\bar{N}_f)}. \quad (\text{A4})$$

Therefore

$$\eta_{m+1} = \eta_m + \delta\eta_m = 2\eta_m + \frac{\tau}{\max(\bar{N}_f)} \quad (\text{A5})$$

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

$$\eta_i = \min \left\{ 1, \frac{\tau}{\max(\bar{N}_f)} (2^i - 1) \right\} \quad (\text{A6})$$

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

## B. UPPER BOUND FOR $\eta$

We want to find the upper bound for  $-\delta\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)/\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)$ . **KMH: I think you need more details on the steps of this derivation.** First let's calculate  $\frac{d}{d\eta}\chi^2(\hat{\mathbf{m}}(\eta), \eta)$

$$\begin{aligned} \frac{d}{d\eta}\chi^2(\hat{\mathbf{m}}(\eta), \eta) &= \frac{\partial}{\partial\eta}\chi^2(\hat{\mathbf{m}}(\eta), \eta) \\ &= \frac{\partial}{\partial\eta}(\mathbf{d} - P\hat{\mathbf{m}}(\eta))^\dagger N(\eta)^{-1}(\mathbf{d} - P\hat{\mathbf{m}}(\eta)) \\ &= -(\mathbf{d} - P\hat{\mathbf{m}}(\eta))^\dagger N(\eta)^{-1} \bar{N} N(\eta)^{-1} (\mathbf{d} - P\hat{\mathbf{m}}(\eta)) \\ &= -\mathbf{r}^\dagger(\eta) N(\eta)^{-1} \bar{N} N(\eta)^{-1} \mathbf{r}(\eta). \end{aligned} \quad (\text{B7})$$

where the first line comes from,  $\chi^2(\hat{\mathbf{m}}(\eta), \eta)$  is minimum  $\chi^2$  value for certain  $\eta$ , 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

$$\begin{aligned} -\frac{\delta\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)}{\chi^2(\hat{\mathbf{m}}(\eta_m), \eta_m)} &= \delta\eta_m \frac{\mathbf{r}^\dagger N(\eta_m)^{-1} \bar{N} N(\eta_m)^{-1} \mathbf{r}}{\mathbf{r}^\dagger N(\eta_m)^{-1} \mathbf{r}} \\ &\leq \delta\eta_m \max \left( \frac{\bar{N}_f}{\tau + \eta_m \bar{N}_f} \right) \end{aligned} \quad (\text{B8})$$

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

If we set the upper bound  $\delta\eta_m \max \left( \frac{\bar{N}_f}{\tau + \eta_m \bar{N}_f} \right) = 1$ ,<sup>2</sup> and then we get

$$\delta\eta_m = \min \left( \frac{\tau + \eta_m \bar{N}_f}{\bar{N}_f} \right) = \eta_m + \frac{\tau}{\max(\bar{N}_f)}. \quad (\text{B9})$$

## C. OTHER CASES

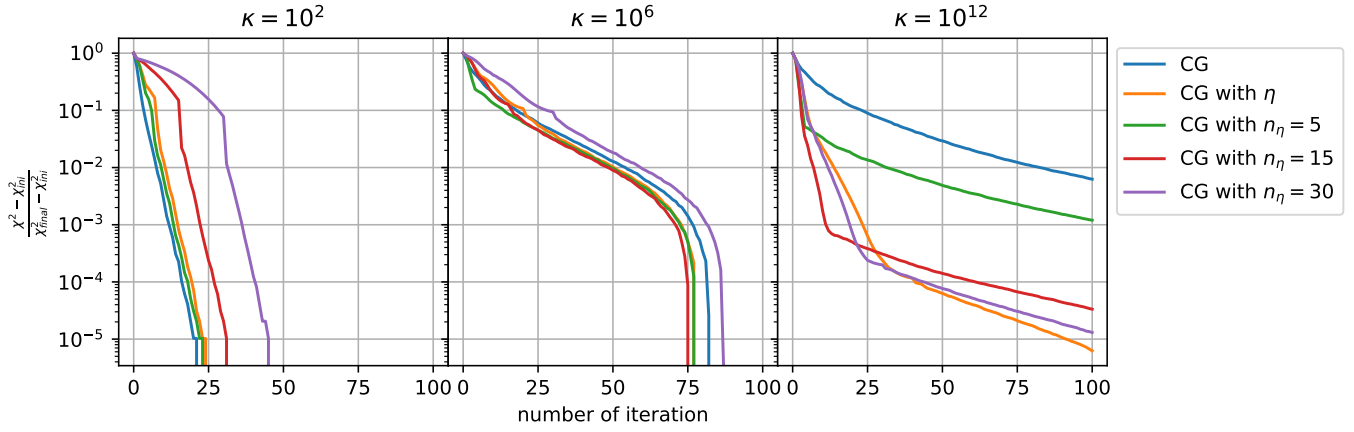
Since the  $\eta$  values determined from Eq.(A6)

$$\eta_i = \min \left\{ 1, \frac{\tau}{\max(\bar{N}_f)} (2^i - 1) \right\} \quad (\text{A6})$$

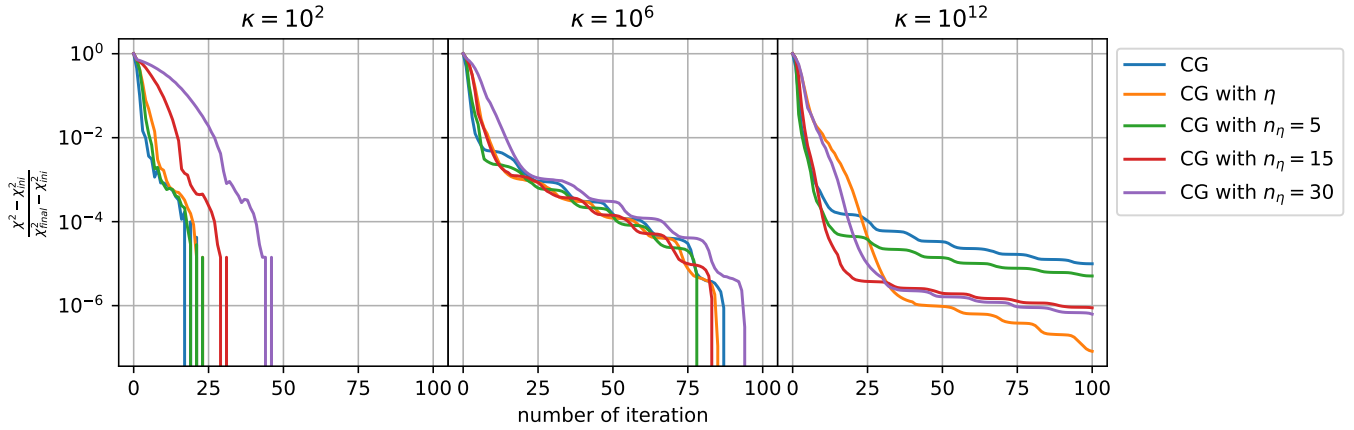
are not dependent on any scanning information, it only depends on noise power spectrum  $P(f)$ , or noise covariance matrix  $N$ . Figure.(4) and Figure.(5) show two examples with same parameters as in Figure.(3) except scanning frequency  $f_{\text{scan}}$  (also we need to change  $f_{\text{apo}}$  to fix condition number), in Figure.(4) it scans very slow and in Figure.(5) it's very fast. In these two cases under  $1/f$  noise model, our  $\eta$  values based on Eq.(A6) are better than manually selected values. Based on these two results we know, the  $\eta$  values should somehow depends on scanning scheme.

## REFERENCES

<sup>2</sup> 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.(A6) is geometric series, only a few  $\eta_m$  values won't satisfy this condition.



**Figure 4.** In this case all parameters are the same as Figure.(3) except  $f_{\text{scan}} = 0.001$ , and corresponding  $f_{\text{apo}}$  to fix the condition number. **KMH:** I'm mixed up about what is changing here. If you change the scan frequency only, I don't see why the apodization should have to change. Since this has the scan slower than the knee, this seems like a lot like fig 3, although the convergence is slower it seems.



**Figure 5.** In this case all parameters are the same as Figure.(3) except  $f_{\text{scan}} = 10$ , and corresponding  $f_{\text{apo}}$  to fix the condition number. **KMH:** I wonder what a scan faster than the knee would make.

460 Elsner, F., & Wandelt, B. D. 2013, A&A, 549, A111,  
 461 doi: [10.1051/0004-6361/201220586](https://doi.org/10.1051/0004-6361/201220586)  
 462 Hufenberger, K. M., & Naess, S. K. 2018, The  
 463 Astrophysical Journal, 852, 92,  
 464 doi: [10.3847/1538-4357/aa9c7d](https://doi.org/10.3847/1538-4357/aa9c7d)  
 465 Janssen, M. A., & Gulkis, S. 1992, in NATO Advanced  
 466 Science Institutes (ASI) Series C, ed. M. Signore &  
 467 C. Dupraz, Vol. 359 (Springer), 391–408

468 Kodi Ramanah, D., Lavaux, G., & Wandelt, B. D. 2017,  
 469 MNRAS, 468, 1782, doi: [10.1093/mnras/stx527](https://doi.org/10.1093/mnras/stx527)  
 470 Papež, J., Grigori, L., & Stompór, R. 2018, A&A, 620, A59,  
 471 doi: [10.1051/0004-6361/201832987](https://doi.org/10.1051/0004-6361/201832987)  
 472 Tegmark, M. 1997a, ApJL, 480, L87, doi: [10.1086/310631](https://doi.org/10.1086/310631)  
 473 —. 1997b, PhRvD, 56, 4514,  
 474 doi: [10.1103/PhysRevD.56.4514](https://doi.org/10.1103/PhysRevD.56.4514)