Map Making and Power Spectrum Estimation of the CMB Radiation

---

Julian Borrill
Lawrence Berkeley National Laboratory
Berkeley, California

James Demmel
Mathematics Department and Computer Science Division
University of California - Berkeley

Plamen Koev
Department of Mathematics
University of California - Berkeley

Planck Workshop on Image Processing

July 4-6, Pisa, Italy

OVERVIEW

- MAP MAKING
  - Why Preconditioned Conjugate Gradients Work?
  - Best Diagonal Preconditioner
  - Implementation of PCG - speed and parallelism

- WHY STOP THERE?
  POWER SPECTRUM ESTIMATION
  - The Monte Carlo Approach
  - Map Cutting - The Killer of All Matrix Structure

- CONCLUSIONS
- RECOMMENDATION FOR SCAN STRATEGIES

GIVEN:

- **Pointing Matrix** (10^9 **time samples** $\times 10^6$ **pixels)**
- **Inverse Noise TIME-TIME correlations** $10^9 \times 10^9$

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad T = \begin{bmatrix} 9 & 5 & 1 & & & & & & \\ 5 & 9 & 5 & 1 & & & & & \\ 1 & 5 & 9 & 5 & 1 & & & & \\ & 1 & 5 & 9 & 5 & 1 & & & \\ & & 1 & 5 & 9 & 5 & 1 & & \\ & & & 1 & 5 & 9 & 5 & 1 & \\ & & & & 1 & 5 & 9 & 5 & 1 \\ & & & & & 1 & 5 & 9 & 5 \\ & & & & & & 1 & 5 & 9 \end{bmatrix}$$

(Toeplitz, Banded)

- **Signal PIXEL-PIXEL Corellations** $S_i = H^T \cdot D_i \cdot H$ **(**H **- spherical harmonic tranforms,** D_i **- diagonal)**
- d_t **- TOD**

INVERSE PIXEL-PIXEL CORRELATIONS:

$$N^{-1} = A^T \cdot T \cdot A =$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 9 & 5 & 1 & & & & & & \\ 5 & 9 & 5 & 1 & & & & & \\ 1 & 5 & 9 & 5 & 1 & & & & \\ & 1 & 5 & 9 & 5 & 1 & & & \\ & & 1 & 5 & 9 & 5 & 1 & & \\ & & & 1 & 5 & 9 & 5 & 1 & \\ & & & & 1 & 5 & 9 & 5 & 1 \\ & & & & & 1 & 5 & 9 & 5 \\ & & & & & & 1 & 5 & 9 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

MAP MAKING

Is the process of solving

$$N^{-1}x = A^T T d_t, \quad \text{where} \quad N^{-1} = A^T T A$$

- **Structure Exploiting Direct Methods**
  e.g. Displacement Structure approach
  can't handle rectanglular matrices
- **Second Best: Iterative Methods**
  N^{-1} **- positive definite, so PCG**
- **PCG converges very fast:**
  about 30 iterations for rel. res. 10^{-4}

WHY PCG CONVERGES FAST?

- **Convergence of PCG** $\approx 1/\mathrm{cond}(N^{-1})$
- $\mathrm{cond}(N^{-1}) = \mathrm{cond}(A^T T A) \approx 10^4$
- $\mathrm{cond}(D^{-1/2} \cdot N^{-1} \cdot D^{-1/2}) \approx 10^2$,
  where $D = \mathrm{diag}(N^{-1})$ **≈PixelHitCount**
- **Both** D **and PixelHitCount work,**
  Which is mathematically "better"?
- **So a closer look at** $\mathrm{cond}(N^{-1}) = \mathrm{cond}(A^T T A)$

$$T = \begin{bmatrix}
9 & 5 & 1 & & & & & & \\
5 & 9 & 5 & 1 & & & & & \\
1 & 5 & 9 & 5 & 1 & & & & \\
 & 1 & 5 & 9 & 5 & 1 & & & \\
 & & 1 & 5 & 9 & 5 & 1 & & \\
 & & & 1 & 5 & 9 & 5 & 1 & \\
 & & & & 1 & 5 & 9 & 5 & 1 \\
 & & & & & 1 & 5 & 9 & 5 \\
 & & & & & & 1 & 5 & 9
\end{bmatrix}$$

  - $\mathrm{cond}(T) \approx 10^2$
  - **Band - property of instrument only (in theory)**
  - **Size = # time samples**
  - **cond**(T) **independent of size of** T
- **PLANCK – Same bolometers, same band, same cond**(T)

THE POINTING MATRIX A

– **Has orthogonal columns:**

$$A^T A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} = D_1^2.$$

– AD_1^{-1} **- orthogonal**

– **Equivalent to making all pixels "equally" important**

– **Equivalent to diagonal preconditioning with** D_1

- D_1 – **"correct" diagonal preconditioner because**

$$\text{cond}((D_1^{-1}A^T)T(AD^{-1})) = \text{cond}(T) \approx 10^{-2}$$

- **Saves 1 iteration from PCG.**

IMPLEMENTING PCG

- **Converges fast, bottleneck – multiplication by N^{-1}**
- **Issues:**
  - **Can't store vector of size 10^9 in RAM - need parallelism to avoid disk access**
  - **Exploit structure to gain speed**

$$N^{-1}v = A^T \cdot T \cdot A \cdot v$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 9 & 5 & 1 & & & & & & \\ 5 & 9 & 5 & 1 & & & & & \\ 1 & 5 & 9 & 5 & 1 & & & & \\ & 1 & 5 & 9 & 5 & 1 & & & \\ & & 1 & 5 & 9 & 5 & 1 & & \\ & & & 1 & 5 & 9 & 5 & 1 & \\ & & & & 1 & 5 & 9 & 5 & 1 \\ & & & & & 1 & 5 & 9 & 5 \\ & & & & & & 1 & 5 & 9 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \cdot v.$$

- $w = Av$ **=observing ... OK**
- $u = Tw$ **... band Toeplitz product**
- $A^T u$ **=repointing on the sky ... OK**

BAND TOEPLITZ MATRIX-VECTOR PRODUCT

- **Band Matrix-Vector**
  cost $O(\text{band} \cdot n)$
- **Embed in** $(2n-1 \times 2n-1)$ **Circulant Matrix**
  $= (\text{FFT})^T \cdot D \cdot (\text{FFT})$
  - **parallel FFTs?**
  - **cost** $O(2n \log 2n)$
- **Better: Embed in** $(n+\text{band}) \times (n+\text{band})$ **Circulant Matrix**

$$\begin{bmatrix}
9 & 5 & 1 & & & & & & 5 & 1 \\
5 & 9 & 5 & 1 & & & & & & 5 \\
1 & 5 & 9 & 5 & 1 & & & & & \\
 & 1 & 5 & 9 & 5 & 1 & & & & \\
 & & 1 & 5 & 9 & 5 & 1 & & & \\
 & & & 1 & 5 & 9 & 5 & 1 & & \\
 & & & & 1 & 5 & 9 & 5 & 1 & \\
 & & & & & 1 & 5 & 9 & 5 & 1 \\
5 & & & & & & 1 & 5 & 9 & 5 \\
1 & 5 & & & & & & 1 & 5 & 9
\end{bmatrix}$$

  cost $O((\text{band} + n) \log(\text{band} + \text{n}))$

- **Can tweak size slightly to make size=**$2^a 3^b 5^c 7^d$

Best: Block-Toeplitz. Instead of:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{bmatrix} = \begin{bmatrix} 9 & 5 & & & & & & & \\ 5 & 9 & 5 & & & & & & \\ & 5 & 9 & 5 & & & & & \\ & & 5 & 9 & 5 & & & & \\ & & & 5 & 9 & 5 & & & \\ & & & & 5 & 9 & 5 & & \\ & & & & & 5 & 9 & 5 & \\ & & & & & & 5 & 9 & 5 \\ & & & & & & & 5 & 9 \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ w_7 \\ w_8 \\ w_9 \end{bmatrix}$$

Do this:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ * \end{bmatrix} = \begin{bmatrix} 9 & 5 & & & & 5 \\ 5 & 9 & 5 & & & \\ & 5 & 9 & 5 & & \\ & & 5 & 9 & 5 & \\ & & & 5 & 9 & 5 \\ 5 & & & & 5 & 9 \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} * \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{bmatrix} = \begin{bmatrix} 9 & 5 & & & & 5 \\ 5 & 9 & 5 & & & \\ & 5 & 9 & 5 & & \\ & & 5 & 9 & 5 & \\ & & & 5 & 9 & 5 \\ 5 & & & & 5 & 9 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ w_5 \\ w_6 \\ w_7 \\ w_8 \\ w_9 \end{bmatrix}$$

BLOCK TOEPLITZ MULTIPLICATION

- **Cost** $O(n \log \text{band})$
  (vs $O(n \log n)$**)**
- **Embarrassingly parallel**
- **Can select** 2^k **size blocks**
- **FFTW more efficient for** $n = 65536$ **than** $n = 10^9$
- **FFTW on consecutive vectors faster than consecutive FFTs**
- **Experimental result:**
  Optimal Block Size ≈ 65536 **for bandwidth of** $T = 10000$**.**

THE POWER SPECTRUM ESTIMATION

$$\text{maximize} \quad f(c) = d^T \cdot D \cdot d + \text{Tr}[\ln D]$$

$$D = \mathcal{N} + S = \mathcal{N} + c_1 S_1 + c_2 S_2 + \cdots + c_l S_l$$

c_i **- multipole powers we want**

$l = 20..1000$

d **- pixel map**

$\mathcal{N}$ **- noise correlations of the "cut map":**

$$N = \left[\begin{array}{c|c} \mathcal{N} & * \\ \hline * & N_c \end{array}\right]$$

- **Newton's Method:**

$$\frac{\partial f}{\partial c_i} = \frac{1}{2}\left(d^T D^{-1} S_i D^{-1} d - \text{Tr}\left[D^{-1} S_i\right]\right)$$

$$\frac{\partial^2 f}{\partial c_i \partial c_j} = -d^T D^{-1} S_i D^{-1} S_j D^{-1} d + \frac{1}{2}\text{Tr}\left[D^{-1} S_i D^{-1} S_j\right]$$

NEWTON'S METHOD CONTINUED

- **Newton's Method:**

$$\frac{\partial f}{\partial c_i} = \frac{1}{2}\left(d^T D^{-1} S_i D^{-1} d - \mathrm{Tr}\left[D^{-1} S_i\right]\right)$$

$$\frac{\partial^2 f}{\partial c_i \partial c_j} = -d^T D^{-1} S_i D^{-1} S_j D^{-1} d + \frac{1}{2}\mathrm{Tr}\left[D^{-1} S_i D^{-1} S_j\right]$$

- **We need quantities like $D^{-1}z$ and Tr$(D^{-1}S_i)$, Tr$(D^{-1}S_iD^{-1}S_j)$**
- **Monte Carlo Estimator for Tr(A)**

$$\mathrm{Tr}(A) =< y^T A y >, \qquad y = \{\pm 1\}, \text{ uniformly distributed}$$

- **Works for Tr$(D^{-1}S) =< yD^{-1}(S_i y) >$**
- **Rough Tr(A) OK**
- **All comes down to $D^{-1}z$, i.e. solving**

$$(\mathcal{N} + S)x = z \quad \text{i.e.} \quad (\mathcal{N} + c_1 S_1 + ... + c_l S_l)x = z$$

- **(Preconditioned) Conjugate Gradients**
- **$S_i x$ in $O(n \log^2 n)$ - on processor $i = 1, ..., l$**
- **$\mathcal{N}x$ ... we don't even have N or N^{-1} explicitly**

SOLVING $(\mathcal{N} + S)x = z$

$$N = \left[\begin{array}{c|c} \mathcal{N} & * \\ \hline * & N_c \end{array}\right]$$

- **If we don't cut maps ($\mathcal{N} = N$) solve**

$$(I + N^{-1}S)x = N^{-1}z$$

  using PCG. Easy

- **Computing $\mathcal{N}x$ if we cut maps using just $N^{-1} = A^T T A$:**
  - **We solve $N^{-1}\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} x \\ 0 \end{bmatrix}$ using PCG (same as in the map-making)**
  - **Equivalent to:**

$$\begin{bmatrix} u \\ v \end{bmatrix} = N \cdot \begin{bmatrix} x \\ 0 \end{bmatrix} = \left[\begin{array}{c|c} \mathcal{N} & * \\ \hline * & N_c \end{array}\right] \begin{bmatrix} x \\ \\ 0 \end{bmatrix}$$

  - **Therefore $u = \mathcal{N}x$**

- **Convergence reasonable cond$(\mathcal{N} + S) \approx 10^4$, 200 or so iterations**

- **no known preconditioner for $(\mathcal{N} + S)$**

CONCLUSIONS

- **Overall cost $O(N_t \log \text{band} \cdot l^2)$ ... almost optimal provided PCGs converge fast**
- **Can handle 10^9 timesamples in days vs hundreds of years**
- **Recommendations for Scan Strategies:**
  - **Observe all pixels the same number of times**
    - **Good or bad!**
    - **Map making OK either way**
    - **Keeps cond$(\mathcal{N} + S)$ small**
  - **Design instruments with smaller cross-corellation of timesamples (i.e. make T more diagonal)**

FUTURE WORK:

- **Fast spherical harmonic transforms**

SLIDES AT:

`http://www.math.berkeley.edu/~plamen/pisa.ps`