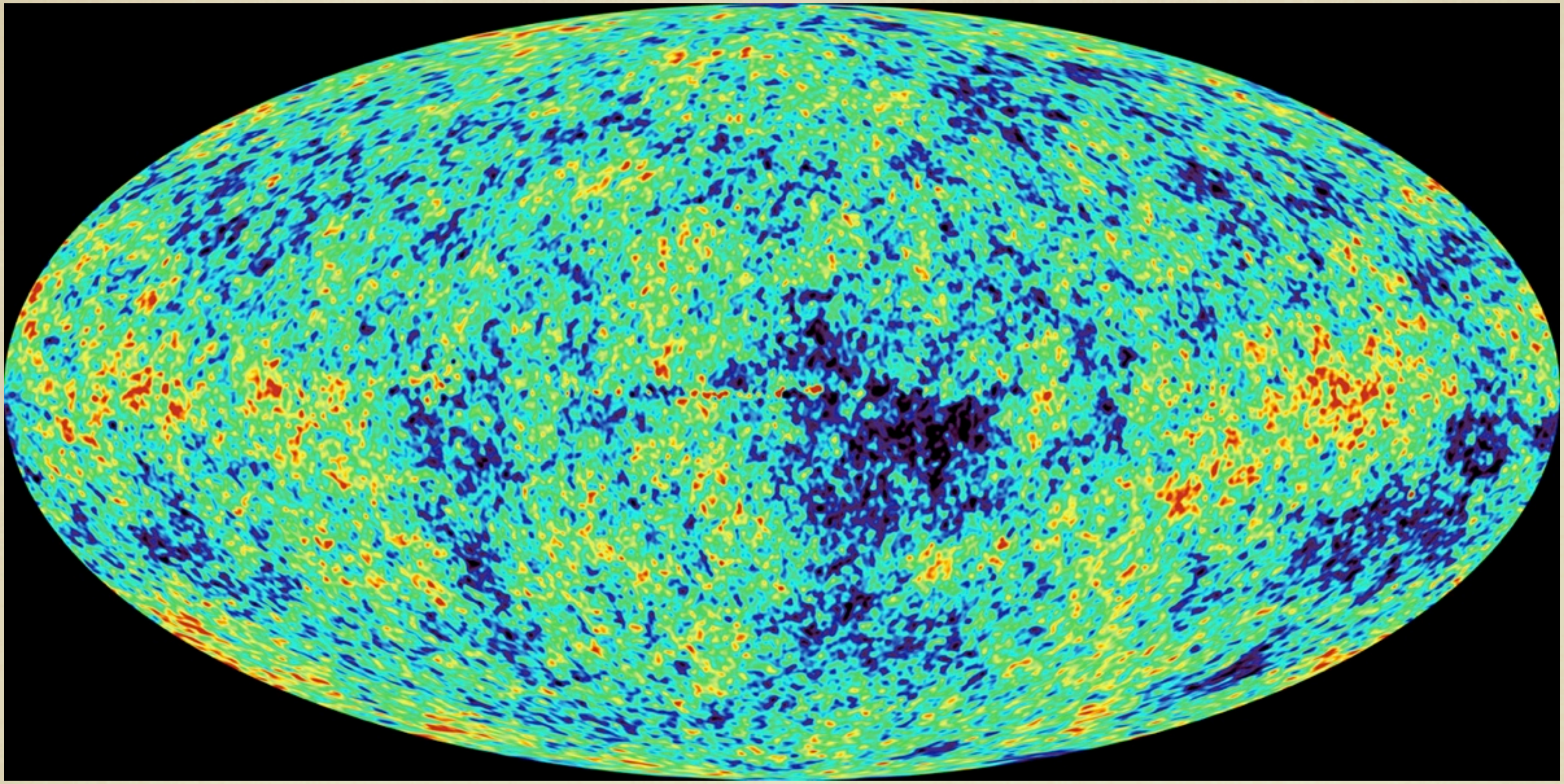
From Time Domain to Maps: principles and practices

Olivier Doré

CITA, University of Toronto

with a few slides from J. Borrill (NERSC)

How to produce these maps in an "optimal" manner?



- Color codes temperature (intensity), here ±100μK
- Temperature traces gravitational potential at the time of recombination, when the Universe was 372 000 ±14000 years old
- The statistical analysis of this map entails detailed cosmological information

CMB analysis scheme

- The key estimator, i.e. the place where theory meets experiments, is the angular power spectrum since both the signal and the noise are Gaussian

$$\langle T_{p_1} T_{p_2} \rangle = \sum_{\ell} \frac{2\ell+1}{4\pi} C_\ell P_\ell \left(\cos \hat{n}_{p_1} \cdot \hat{n}_{p_2} \right)$$

- This set some basic experimental requirements: high resolution, low noise level, multi-frequency, redundancy, polarization

- Analysis Scheme

(1) Measurements are recored as the instrument scans the sky: time ordered data (TOD), d_t^{ν}

(2) Pre-process TOD (deglitch, decorrelate, pointing, calibration)

(3) From them we want to estimate the time domain noise properties

(4) We then want to deduce maps, x_p^{ν}, and their errors

(5) We want to characterize statistically these maps, e.g. power spectrum, bispectrum,... and its errors

(6) From these characteristics we want to infer cosmological constraints

Specific goals

- To cover
  - The basic mathematical formalisms
  - The algorithms and their scaling behavior
  - Some example implementation issues
- To consider how to extract the maximum of information from the data, subject to practical computational constraints
- To illustrate some of the computational issues faced when very analyzing large data sets (Planck, ACT, Spider)

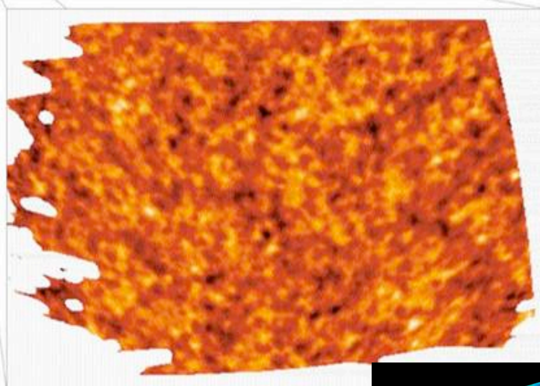
Few numbers

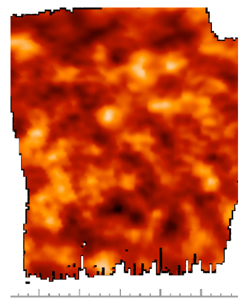
COBE

1992
10^3 pix.
7°

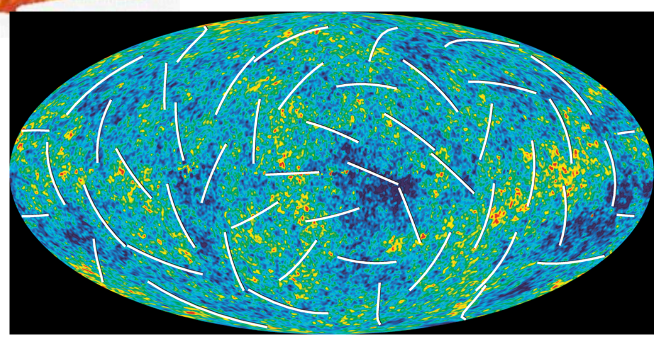
Boomerang 2001

1.5 10^5 pix.
7’

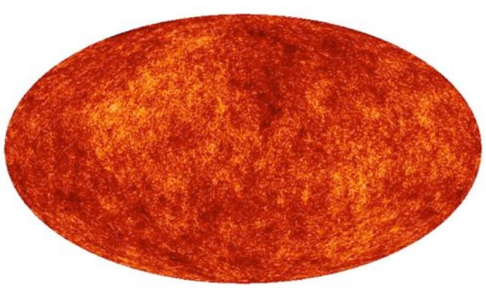




Maxima 2001

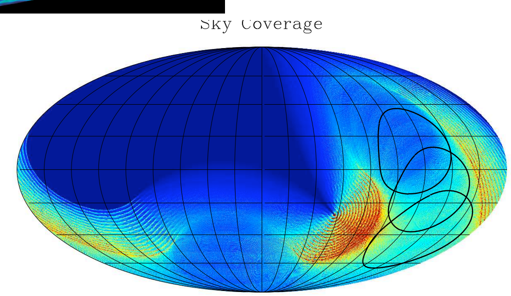


WMAP 2003-....
12 10^6 pix.
5’



PLANCK 2008

10^7 pix.
1.5’
54 channels



Spider 2009-2010
4 10^5 pix.
14’
3000 channels
8TB dataset

Why do we proceed this way?

- The key reason is data compression as illustrated by the planck example

- Time-ordered data

  #samples=# detectors x sampling rate x duration

  ~ 70 x 200 Hz x 18 months

  ~ 6 10^{11} samples

- Pixelized sky maps (with Healpix)

  #Pixels = # components x sky fraction x $12N_{side}^2$

  ~ (3-6) x 1 x 12 x 4096^2

  ~ 6-12 10^8 pixels

- Power Spectrum

  #Bins = # Spectra x #multipoles / Bin resolution

  ~ 6 x (3 x 10^3) / 1

  ~ 1.8 10^4 multipoles

- We want to avoid any information loss at any stage

Likelihood Chain

- We will follow a Bayesian approach
- We are ultimately interested in the posterior Probability

$$\mathcal{P}(\Theta, C_\ell, x_p, N_{tt'} | d_t^\nu, I)$$

- Using the Bayes theorem, we can write

$$\begin{aligned}\mathcal{P}(\mathcal{T}|\mathcal{D},\mathcal{I}) &= \frac{\mathcal{P}(\mathcal{T}|\mathcal{I})\mathcal{L}(\mathcal{D}|\mathcal{I},\mathcal{T})}{\mathcal{P}(\mathcal{D}|\mathcal{I})} \\ &\propto \mathcal{P}(\mathcal{T}|\mathcal{I})\mathcal{L}(\mathcal{D}|\mathcal{I},\mathcal{T})\end{aligned}$$

- The estimators thus defined are <u>Maximum Likelihood estimators</u>

$$\begin{aligned}&\mathscr{P}(\Theta, N_{tt'}, \mathscr{C}_\ell, x_p^\nu \mid d_t^\nu, I) = \\ &\qquad \mathscr{L}(d_t^\nu \mid x_p^\nu, N_{tt'}, I)\, \mathscr{L}(x_p^\nu \mid \mathscr{C}_\ell, \Theta, N_{tt'}, I)\, \mathscr{L}(\mathscr{C}_\ell \mid \Theta, N_{tt'}, I) \\ &\qquad \times \quad \mathscr{P}(\Theta, N_{tt'} \mid I)/\mathscr{P}(d_t^\nu \mid I).\end{aligned}$$

- Adequate formulation but hide all the practical complexities... more on that later...

Formalism - I

- Consider data consists of noise and signal

$$d_t = n_t + s_t = n_t + A_{tp} s_p$$

- Pointing Matrix, A, encodes the weight of each pixel p in each time step t. In principle A encompass the full beam response as well as the calibration
- If we restrict ourselves to total power experiment (not differential like WMAP) and we aim at reconstructing the beam convolved sky, then

$$A_{tp} = \begin{cases} 1 & \text{if } \hat{n}_t \in \text{pixel } p \\ 0 & \text{otherwise} \end{cases}$$

- Note that in this case s is this "beam" and "pixel" smoothed

Formalism - II

- The statistical question we aim at answering is how to build the best estimate of the sky, $\hat{s}$, given the time stream, d_T
- Assuming Gaussian instrumental noise (the only assumption here), we can write this likelihood as

$$-2lnP(d|s) = n_{t_1}^T N_{t_1 t_2}^{-1} n_{t_2} + \mathrm{Tr}[\ln N_{t_1 t_2}]$$

- where the time domain noise covariance matrix is

$$N_{t_1 t_2} = < n_{t_1} n_{t_2} >$$

- Assuming that the noise is stationary and the noise correlation limited in time, we have

$$N_{t_1 t_2}^{-1} = f(|t_1 - t_2|) \quad \text{and} \quad f(|t_1 - t_2|) = 0 \quad \text{if} \quad |t_1 - t_2| > t_d$$

Noise estimation

- You need an approach to estimate the noise. It can be done as follows

$$N^{-1}(\Delta) = FFT^{-1}(\frac{1}{FFT[n_t])^2})$$

(1)Assume TOD is pure noise, i.e. $d_T=n_T$
(2)Solve for the map, $d_P \sim s_P$
(3)Subtract the evaluated signal from the data, $n_T=d_D-A_{TP}s_P$
(4)Iterate

- Shown to converge to be a slightly bias estimator (Ferreira & Jaffe 00, Prunet 01, Stompor 06)

Formalism-III

- Maximizing the previous likelihood over signals, we obtain the maximum likelihood map estimator

$$\hat{s}_p = (A^T_{t_1 p_1} N^{-1}_{t_1 t_2} A_{t_2 p_2})^{-1} A^T_{t_2 p_2} N^{-1}_{t_2 t_3} d_{t_3}$$

- Simple Generalized χ^2 solution

- Its noise covariance properties are

$$N_{p_1 p_2} = (A^T_{t_1 p_1} N^{-1}_{t_1 t_2} A_{t_2 p_2})^{-1}$$

- Taken together, this is a complete description of the data

- Basically, MM problem is a simple inversion problem

Some Numbers

Symbol	Description	Planck
$\mathcal{N}_t$	Number of samples	5 x 10^{11}
$\mathcal{T}$	Noise bandwidth	O(10^4)
$\mathcal{N}_p$	Number of pixels	6 x 10^8
$\mathcal{N}_s$	Number of spectra	6
l_{max}	Maximum multipole	3 x 10^3
$\mathcal{N}_b$	Number of spectral bins	2 x 10^4
$\mathcal{N}_i$	Number of iterations	-
$\mathcal{N}_r$	Number of realizations	-

Computational Constraints

- 1 GHz processor running at 100% efficiency for 1 day performs $O(10^{14})$ operations.
- 1 Gbyte of memory can hold $O(10^8)$ element vector, or $O(10^4 \times 10^4)$ matrix, in 64-bit precision.
- You can read at 100MByte/s, i.e. 22hr for Spider data set
- Parallel (multiprocessor) computing increases the operation count and memory limits.
- Challenges to computational efficiency & scaling:
  - load balancing (work & memory)
  - data-delivery, including communication & I/O

Algorithms - I

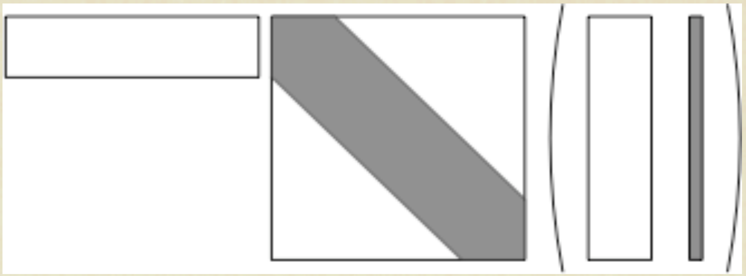
- We want to solve the system:

Equation	Naive Op. Count
$z_p = A^T_{tp} N^{-1}_{tt'} d_{t'}$	$\mathcal{N}_t^2$
$N^{-1}_{pp'} = (A^{\mathrm{T}}_{tp} N^{-1}_{tt'} A_{t'p'})$	$\mathcal{N}_t^2 \mathcal{N}_p$
$N_{pp'} = (N^{-1}_{pp'})^{-1}$	$\mathcal{N}_p^3$
$d_p = N_{pp'} z_{p'}$	$\mathcal{N}_p^2$

- Eg. $(5 \times 10^{11})2 \times (6 \times 10^{8}) \sim 2 \times 10^{32}$ for Planck...

ALGORITHMS - II

- Exploit the structure of the matrices... semi-brute force
  - Pointing Matrix is sparse
  - inverse noise correlation matrix is band-toeplitz



- Associated matrix-matrix & -Vector operations reduced from $N_T^2 N_P$ and N_T^2 to $N_T\tau$, e.g. $(5 \times 10^{11}) \times 10^{14} \sim 5 \times 10^{15}$ for Planck, i.e. 50 1GHz CPU for one year...
- MADCAP package by Borrill & Stompor 1999

Algorithms - III

- Replace explicit matrix inversion with an iterative solver (e.g. preconditioned conjugate gradient) using repeated matrix-vector multiplications

$$N^{-1}_{p_1 p_2} d^i_{p_2} = z_p$$

- reducing the scaling from N_P^3 to $N_I N_P^2$

- depends on the required solution accuracy and the quality of the preconditioner (white noise works well), e.g. $30 \times (6 \times 10^8)^2 \sim 10^{19}$ for Planck.

Algorithms - IV

- Leave the inverse pixel-pixel noise matrix in implicit form and use iterative method (Jacobi, PCG, Multigrid..)

$$\left(A^T_{t_1p_1} N^{-1}_{t_1t_2} A_{t_2p_2}\right) s_{p_2} = A^T_{t_3p_1} N^{-1}_{t_3t_4} d_{t_4}$$

- Now each multiplication takes $N_T\tau$ operations in pixel space or $N_T\log\tau$ in Fourier space, e.g. 30 x (5x10^{11}) x log 10^4 for Planck

- But this gives no information about the pixel-pixel matrix noise correlation matrix required for further stage, although you can fill the weight matrix easily

Wright 96, Prunet et al. 00, Doré et al. 01, Natoli et al. 01, Stompor et al. 02, Ashdown et al. 06

Jacobi solver

- To solve for $$Ax = b$$

$$\sum_{j=1}^{n} a_{ij} = b_i$$

- If we assume all the x_j ($i \neq j$) are fixed then we can solve for x_i

$$x_i = (b_i - \sum_{j \neq i} a_{ij} x_j)/a_{ii}$$

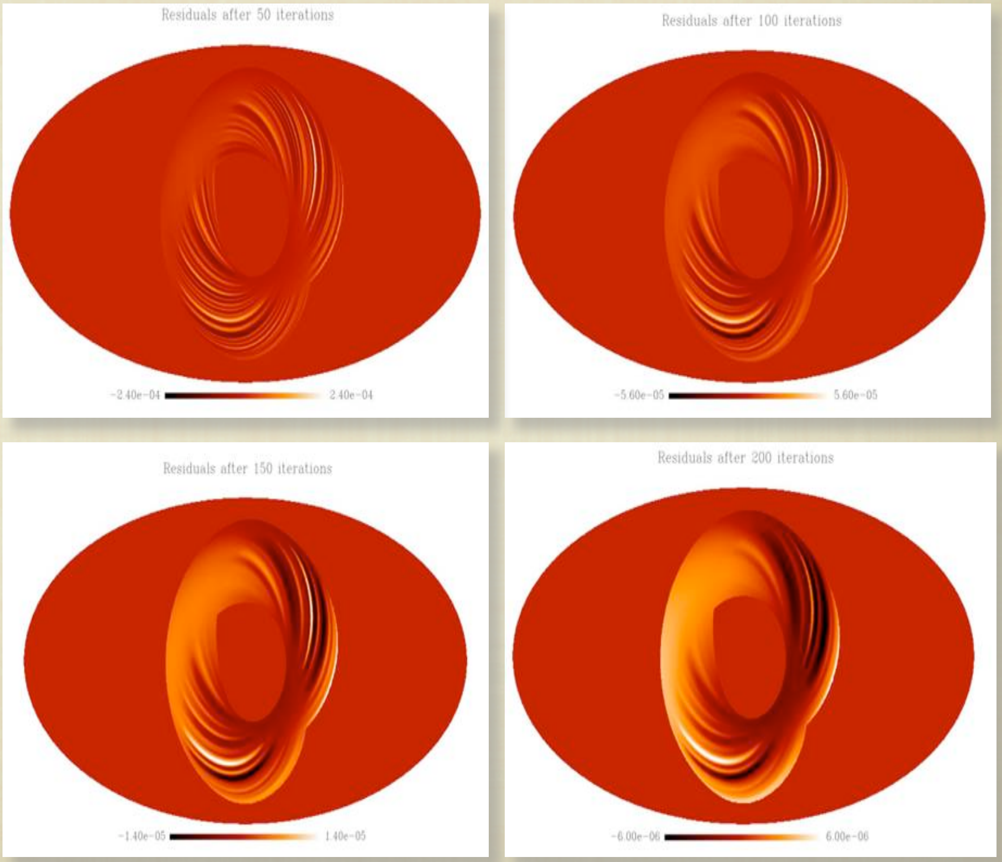
- Which suggests the following iterative scheme

$$x_i^{(k)} = (b_i - \sum_{j \neq i} a_{ij} x_j^{(k-1)})/a_{ii}$$

- Order of operations irrelevant

Which iterative methods?

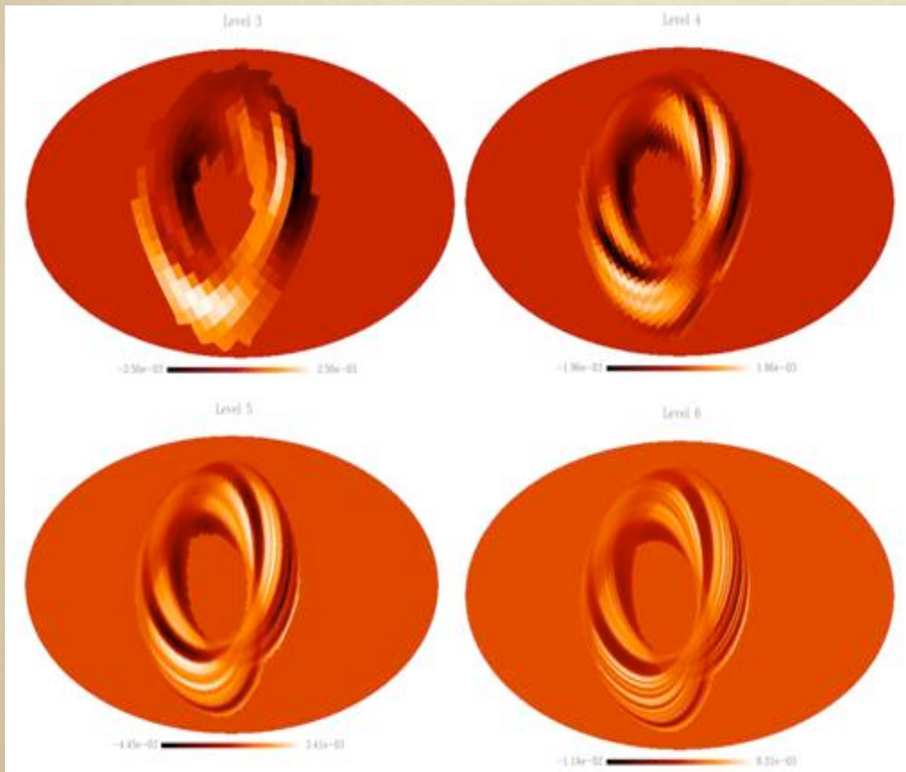
- <u>Jacobi</u> method:
  - the most standard
  - Guaranteed convergence....
  - but slow...
- But why so slow?

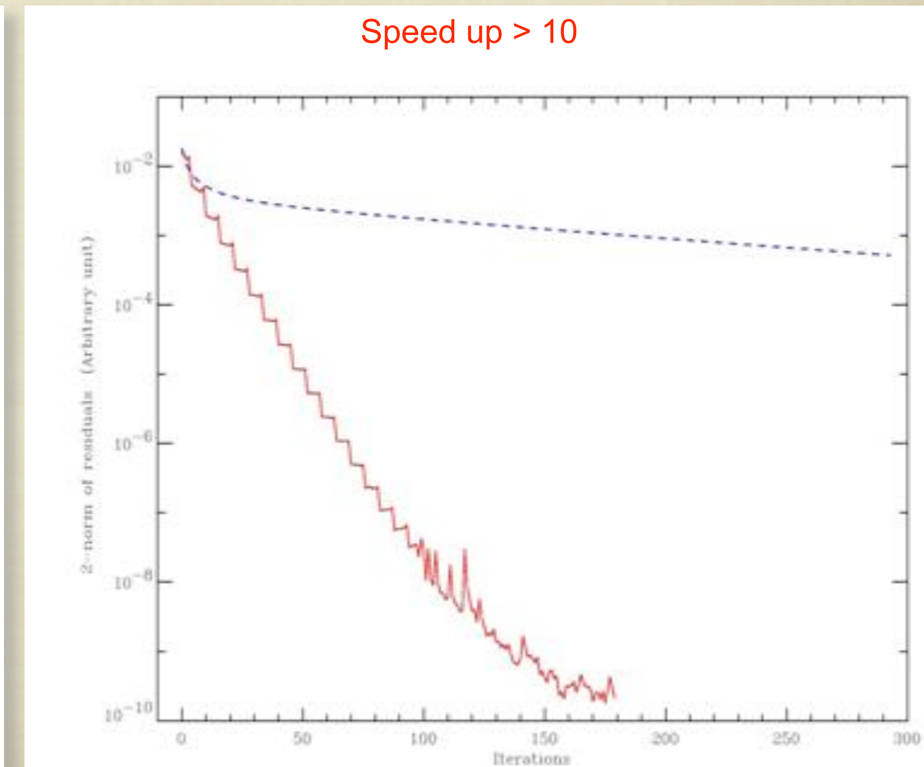


- Archeops simulations
- Slow large scale convergence

Multigrid method?

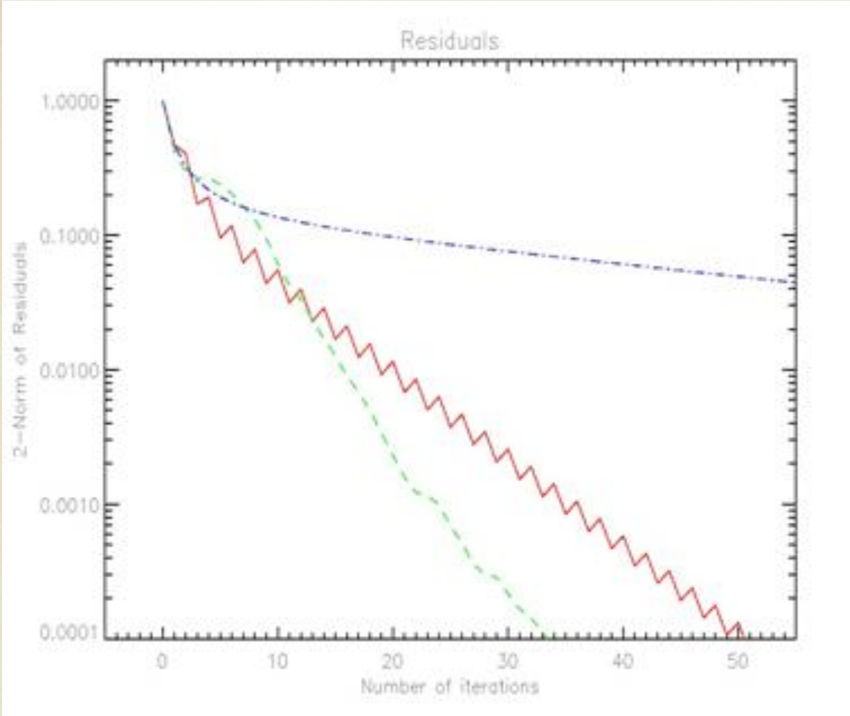
- This suggests to solve for the maps at lower resolution where we can solve for the maps exactly by iterating many times and inject the solution at higher resolution
- Natural recursive jacobi algorithm





Doré et al. 01

Preconditioned Conjugate Gradient



- PCG is guaranteed to converge
- Actually faster in terms of # iterations and also in terms of time per iteration

Preconjugate Conjugate Gradient

$$Ax = b$$

- We define the residuals at an iteration k as

$$r^{(k)} = b - Ax^{(k)}$$

- We update the solution x^k by moving in the direction p^k

$$x^{(k)} = x^{(k-1)} + \alpha_k p^{(k)}$$
$$\alpha_i = r^{(i-1)T} r^{(i-1)} / (p^{(i)T} A p^{(i)})$$

- The residuals are updated as

$$r^{(k)} = r^{(k-1)} - \alpha_k A p^{(k)}$$

- The direction p^k (and amplitude α) are orthonormal to all previous Ap^{k-1}

$$p^{(i)} = r^{(i)} + \beta_{i-1} p^{(i-1)}$$
$$\beta_i = r^{(i)T} r^{(i)} / r^{(i-1)T} r^{(i-1)}$$

- The new choice x^i belongs to x^0+span{r^0+...$A^{i-1}r^0$} and minimizes

$$(x^{(i)} - \hat{x})^T A (x^{(i)} - \hat{x})$$

- Well studied methods and converges as A condition number

Extension to Polarization

- For polarization data the signal can be written in terms of the i, q & u Stokes parameters and the angle α of the polarizer relative to some chosen coordinate frame

$$s_t = \frac{1}{2}(i_t + \cos 2\alpha_t q_t + \sin 2\alpha_t u_t) = A_{tp} \begin{pmatrix} i \\ q \\ u \end{pmatrix}_p$$

where A_{tp} now has 3 non-zero entries per row.

- We need at least 3 observation-angles to separate i, q & u.

Extension to systematics

- If the data includes a sky-asynchronous contaminating signal (eg. MAXIMA's chopper-synchronous signal)

$$d_t = n_t + A_{tp}x_p + B_{tq}x_q = n_t + \left[A_{tp}B_{tq}\right]\begin{pmatrix} s_p \\ x_q \end{pmatrix}$$

- This can be extended to any number of contaminants, including relative offsets between sub-maps

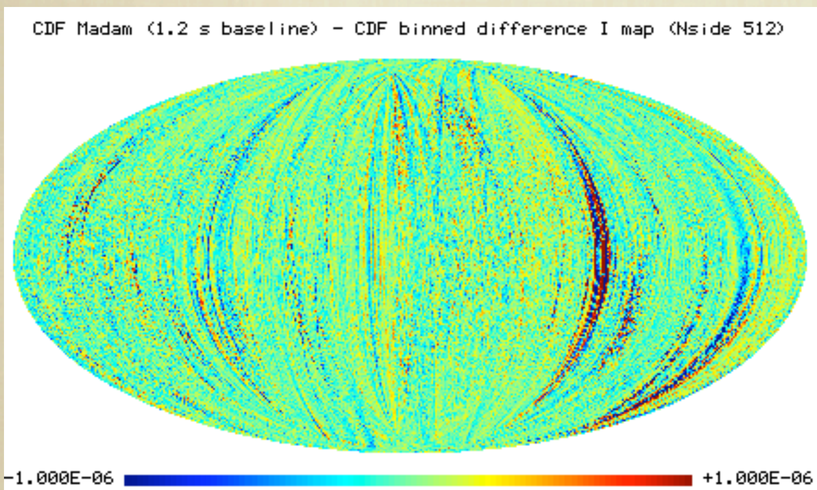
- Easy extension to (correlated) multi-detectors system (see e.g. Blast maps, Patanchon et al. 07)

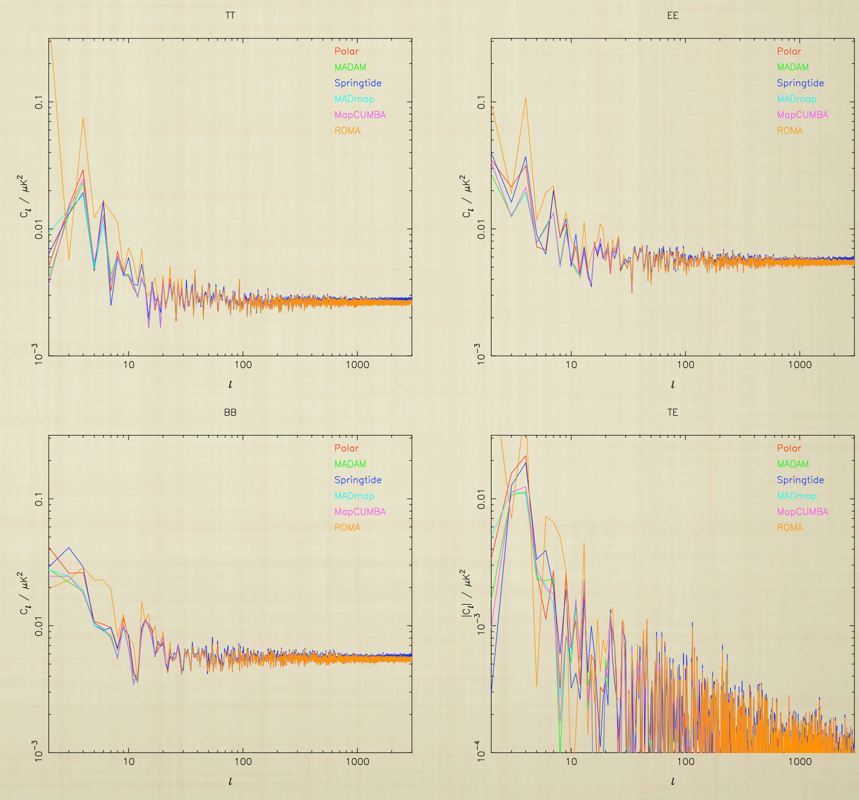
$$\begin{pmatrix} d_1^1 \\ \vdots \\ d_{t_n}^1 \\ d_1^2 \\ \vdots \\ d_{t_n}^2 \\ \vdots \end{pmatrix} = \begin{pmatrix} n_1^1 \\ \vdots \\ n_{t_n}^1 \\ n_1^2 \\ \vdots \\ n_{t_n}^2 \\ \vdots \end{pmatrix} + \begin{bmatrix} A_{tp}^1 \\ A_{tp}^2 \\ \vdots \end{bmatrix} x_p$$

Shortcuts?

- E.g. destriping for Planck
- Alternative description of the data

$$
\begin{aligned}
d_t &= A_{tp} x_p + n_t \\
d_t &= A_{tp} x_p + B_{tq} c_q + n_t
\end{aligned}
$$





Ashdown et al. 06

Conclusions

- The map-making problem is well defined formally but challenging practically
- Optimal solution is expansive but :
  - The maximum-likelihood map only can be calculated in $N_I N_T \log(\tau)$ operations - $O(10^{14})$ for Planck.
  - The sparse inverse pixel-pixel correlation matrix can be calculated in $N_T \tau$ operations - $O(10^{15})$ for Planck.
  - A single column of the pixel-pixel correlation matrix can be calculated (just like a map) in $N_I N_T \log(\tau)$ operations - $O(10^{14})$ for Planck.
- they are always short-cuts (high-pass filtering and simple coaddition, destriping, etc.) not necessarily too unoptimal

Fin