Issues In The Analysis Of Very Massive CMB Data Sets

Julian Borrill
Computational Cosmology Center, Berkeley Lab
& Space Sciences Laboratory, UC Berkeley

Collaborators

Reese Baird, Christopher Cantalupo & Theodore Kisner,
Berkeley Lab/UC Berkeley

Radek Stompor,
Laboratoire Astroparticule et Cosmologie, Paris

Andrew Jaffe,
Imperial College, London

The BOOMERanG, MAXIMA, MAXIPol,
Planck, EBEx & PolarBear Teams

The Berkeley Lab NERSC & CRD Staff

CMB Data Analysis Overview

- Despite the existence of simple and exact expressions for making maximum likelihood maps and power spectra, their scaling with the cube of the number of pixels means that all current and future CMB data analysis necessarily involves approximations.
- Some methods are exact in some unrealizable limit, others are fundamentally inexact.
- Using these approximations, the dominant scaling is driven by manipulations of the time-ordered data.

Time-Stream Manipulations

- Analysis
  - Pre-processing
  - Noise-estimation
    - Frequency/time domain
    - Pixel/harmonic domain
  - Map-making

- Simulation
  - Experimental design & development
  - Quantification of pixel/harmonic space approximations
    - Simulation - e.g. Keating et al
    - Analysis - e.g. power spectrum estimation

Time-Ordered Data Volumes

Experiment	Duration	Freq. (GHz)	Sampling (Hz)	Time-Streams	TSamples
WMAP	9 yr	23 - 94	7.8 - 19.5	40	0.2
Planck	2 yr	30 - 857	32.5 - 172	74	0.6
EBEx	2 wk	150 - 410	400	1402	0.7
Spider	3 wk	96 - 220	200	2112	0.8
SPTpol	3 x $^1/_2$ yr	90 - 220	200	1000	9.5
PolarBear	2 yr*	150 - 220	200	2548	32.1
QUIET	3 yr	40 - 90	100	11200	106.0
EPIC	1 yr	30 - 250	10	4096	1.3
PILOT	4 yr	30 - 150	100	364	4.6
Feed Farm	1 yr	150 - 300	100	1959	6.2
EPIC-LC	2 yr	30 - 300	1000	2878	181.5
EPIC-INT	4 yr	30 - 850	1000	1620	204.4
EPIC-CS	4 yr	30 - 500	1000	1520	191.7

$$2 \times 10^{11} < \mathcal{N}_t < 2 \times 10^{14}$$

Computational Costs

- CPU cycles
  - No more than linear/log-linear in $\mathcal{N}_t$
  - Prefactor: 1 - 100
  - Low computational efficiency: 1 - 10%
  - 30 min - 6 yrs on a single 1 Gflop/s core
    - Massive parallelism essential

- Disk space
  - Single precision tod: 1 TB - 1 PB
  - Typically O(100x) required
  - 2 min - 1 day over a 10 GB/s IO subsystem
    - Stage data from archival storage
    - Minimize IO

Analysis Resources

- 10-year history of CMB data analysis at NERSC.
- Recognized as key facility by Weiss task-force.
- Each year, O(100) users from O(10) experiments.
  - 1,000,000 MPP-hours/year for suborbital expts
  - 2,000,000 MPP-hours/year for Planck
  - 50,000 MPP-hour startup this year for CMBpol
- NASA/DOE IAA guarantees Planck time; sets precedent for future satellite missions.
- Currently upgrading to 40,000 core system
  - 2,000,000,000 MPP-hours/year
  - Plenty of room for allocation growth
  - New challenges to exploit this degree of parallelism

Analysis Tools - I

- Microwave Anisotopy Dataset mapmaker (MADmap)
  - Maximum likelihood solution of map-making equation
  - Preconditioned conjugate gradient solver
  - Massively parallelized (run up to 16,000 way)

- M3 data abstraction layer
  - Analysis codes should not be concerned with
    - Data format
    - Data distribution
    - IO optimization
  - Processes request data; M3 uses XML data description to parse & fill requests.

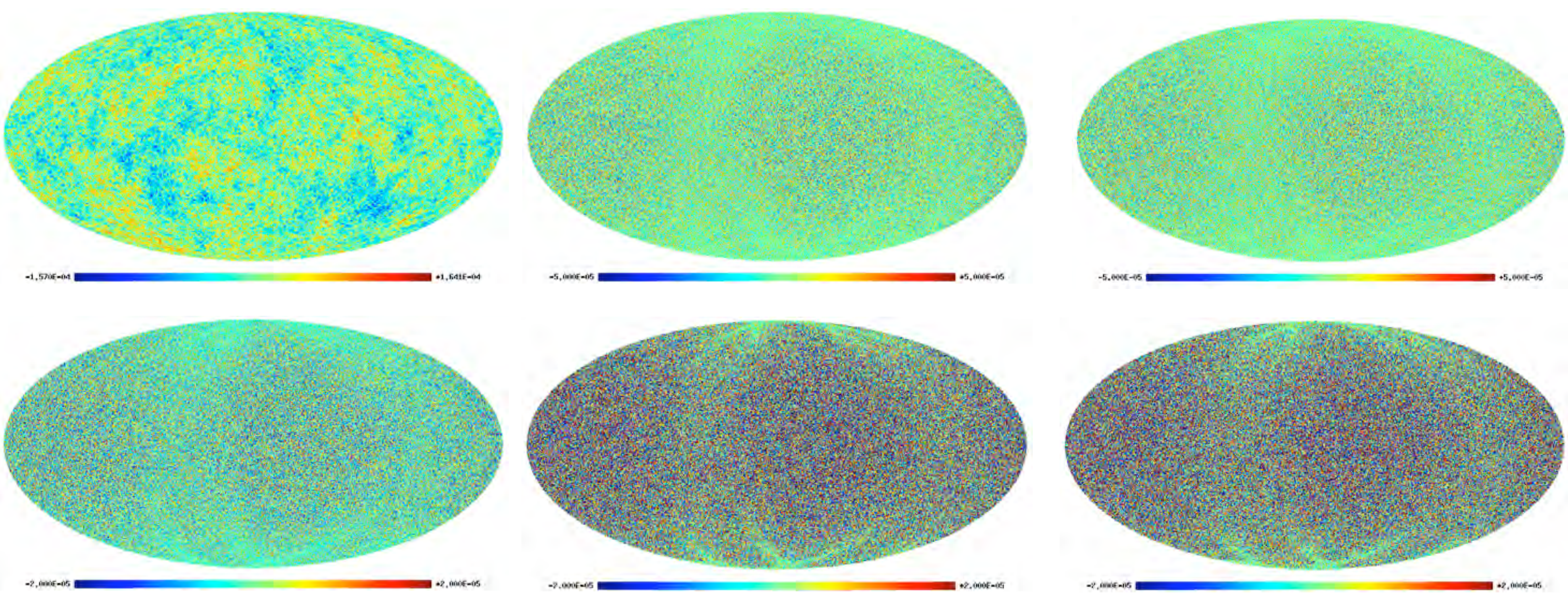
Analysis Tools - II

- Generalized Compressed Pointing (GCP)
  - Store/read sparse boresight (or similar) pointing & polarization modulation parameter(s).
  - Compute dense detector pointing/weights on the fly
    - Interpolation/rotation implies optimal parallelism
- On-The-Fly Simulation (OTFS)
  - Generate simulations on demand to avoid IO cost
  - General capabilities
    - Scan from a beam-smoothed map
    - Add noise with arbitrary spectrum
  - Experiment-specific plug-ins

OTFS - Planck 217GHz Example

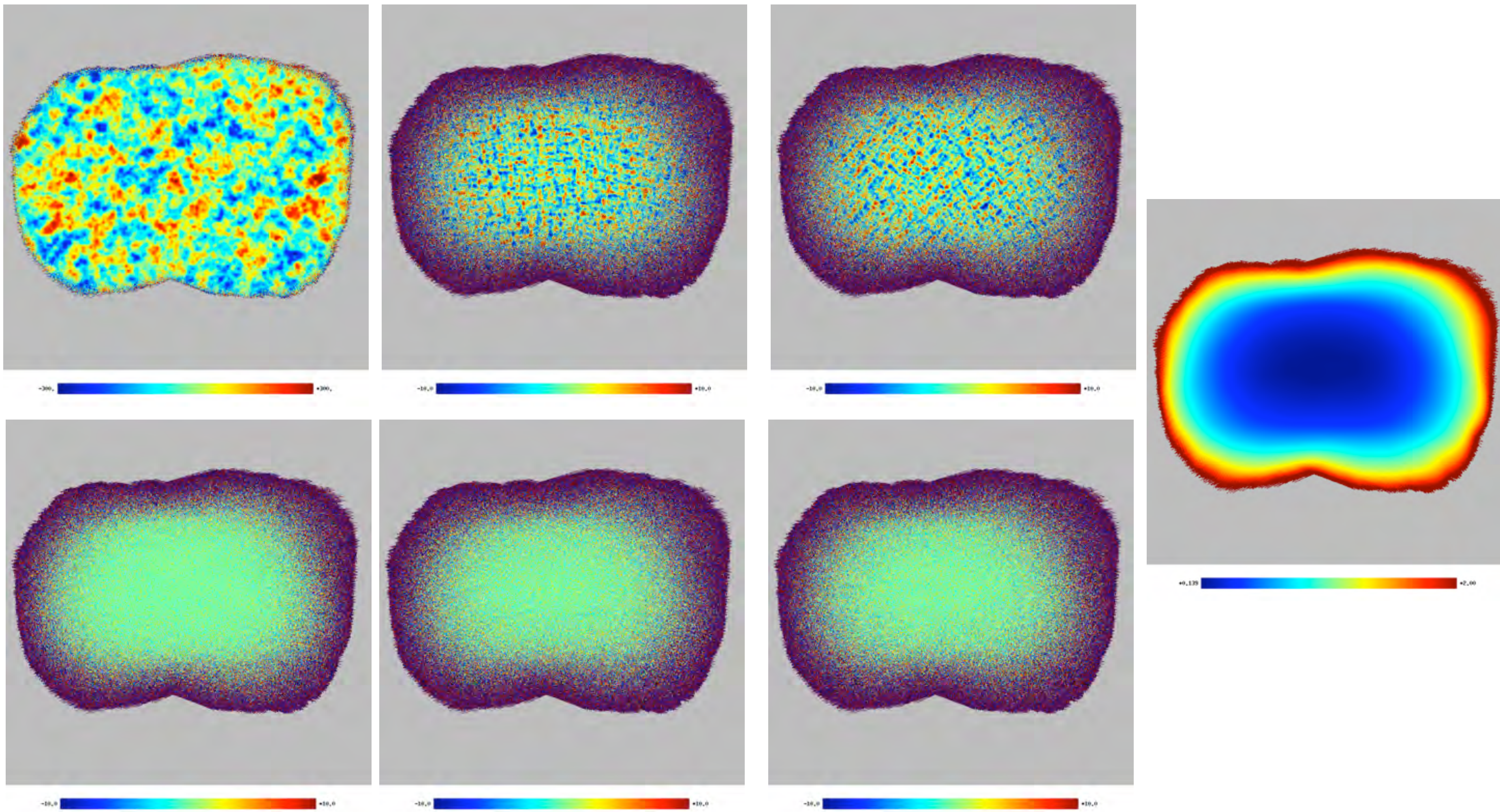
MADmap-ing 12 detectors for 1 year in 15 min on 4196 cores.

(cf. Seaborg 6000 core/4 hour "heroic computation")



OTFS - EBEx LDB Example

MADmap-ing 721 detectors for 14 days in <1hr on 8192 cores.



OTFS Issues

- Generating large numbers of uncorrelated Gaussian randoms quickly and reproducibly across very many processes:
  - Use independent streams from a single pseudo-random sequence.
  - Associate a stream with each stationary piece (SP) of the time stream.
- Distributing data over very many processes load-balanced & algorithmically optimally:
  - Randoms generation prefers complete SPs on each process.
  - GCP prefers the same interval from all timestreams to be on the same process

Data & Code Management

- Data management
  - Locating data sets
  - Staging from archive to disk (including clean-up)
  - Building data description, e.g. M3 XML
  - Data ownership by experiment not user

- Code management
  - Common user environment across systems
  - Traceable provenance & history
  - System-specific optimization
  - Access control within shared/community resource
  - Modules system provides this

Summary

- B-mode CMB experiments require very massive data sets to achieve sufficient S/N.
- Data management and analysis become:
  - Significant computational challenge
    - Bandwidth, cycles, storage, IO
  - Significant source of systematic errors
    - Analysis approximations
    - Algorithmic, implementation & numerical errors
  - Significant source of human error
    - Data & code provenance, validation & verification.
- Massively parallel computing resources are essential & available, but work needs to be done on scaling and data/task automation.