Model-based component separation exploiting sparsity

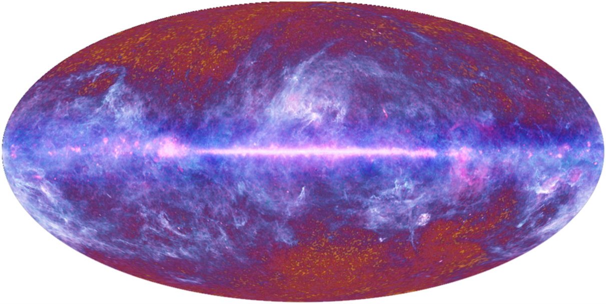
Mel Irfan
CosmoStat Lab, CEA

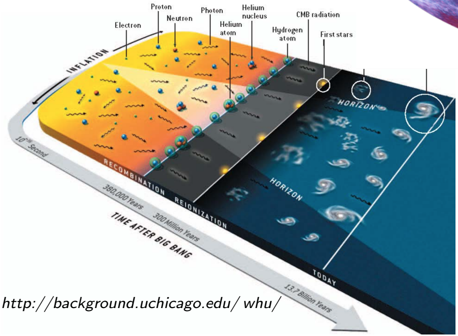
05/09/17

Overview

- Brief introduction to the CMB component separation
- How sparsity can help
- The problem at hand: thermal dust and the CIB
- The new methodology
- Validation on Planck simulation data

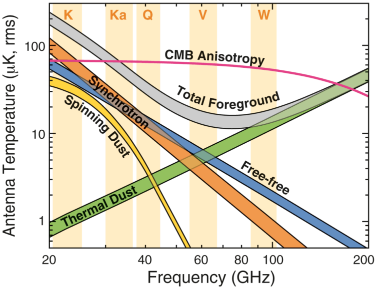
CMB component separation

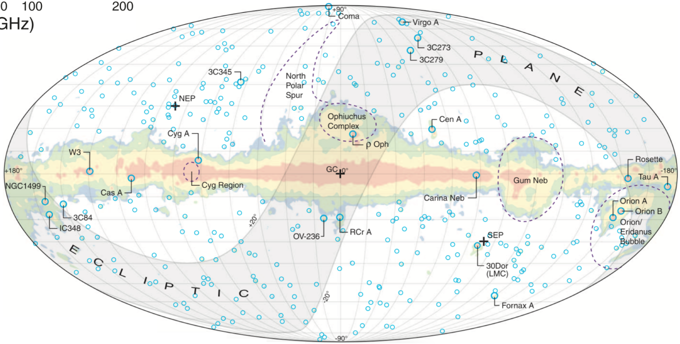




http://background.uchicago.edu/ whu/

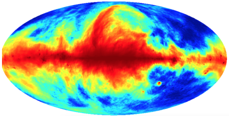
CMB component separation





Bennett et al. 2013

CMB component separation



Commander

Pixel space, parametric, priors

Haslam 1982

ILC (internal linear combination)

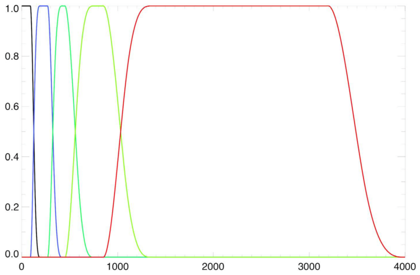
wavelet space, non-parametric, noise information - more to come

Sevem and SMICA

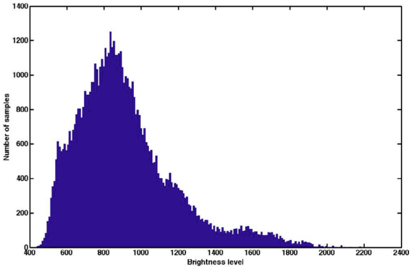
Recovery of just the CMB, no external data sets needed.

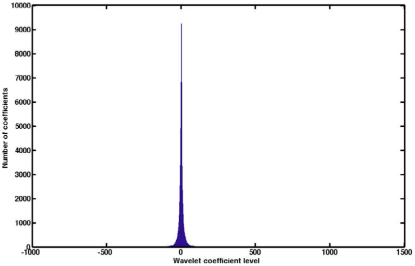
Goals for this work: provide foreground estimates as well as CMB
full sky, full resolution data
exploit sparsity in wavelet domain

Sparsity and the wavelet domain



Bobin et al. 2013





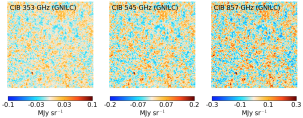
- Sparse: majority of signal is zero
- Spatially correlated source
- Wavelets filter in spherical harmonic domain (x-axis: ℓ)

Thermal dust and the CIB

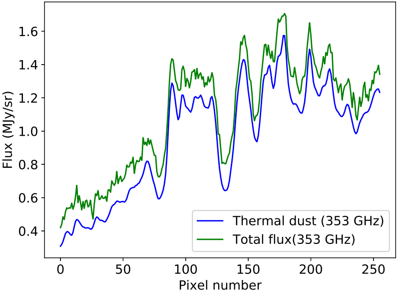
- *Planck* HFI: 217, 353, 545, 857 GHz
- Thermal dust and CIB

$$x_{\nu_i}^{\mathrm{dust}} = \tau_{353} \times B(T,\nu) \times \left(\frac{\nu}{353\,\mathrm{GHz}}\right)^{\beta}$$

Planck XLVIII 2016



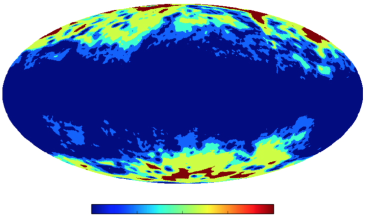
- Both BB, CIB flatter power spectrum
- Smoothing conundrum!



GNILC

$$\text{total flux} = \text{dust} + \underbrace{\mathrm{CIB} + \mathrm{CMB} + \mathrm{noise}}_{\text{nuisance, Gaussian approx}}$$

- Clever smoothing using nuisance estimates.



At each wavelet scale:

whitening $\mathbf{R}_{\mathrm{nus}}^{-1/2}\mathbf{R}_{\mathrm{tot}}\mathbf{R}_{\mathrm{nus}}^{-1/2}$

$\mathbf{v} \sim 1 \Rightarrow$ noise dominated $\rightarrow$ smooth

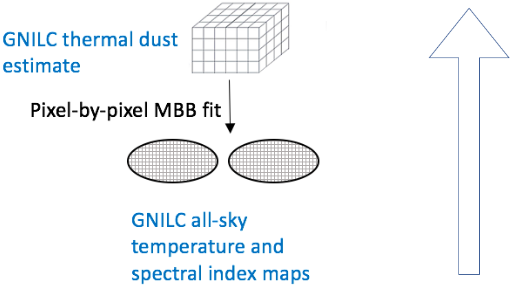
$\mathbf{v} >> 1 \Rightarrow$ signal dominated

min $||X - AS||^2$ for thermal dust map

a.k.a $||X - A\alpha\Phi||^2$

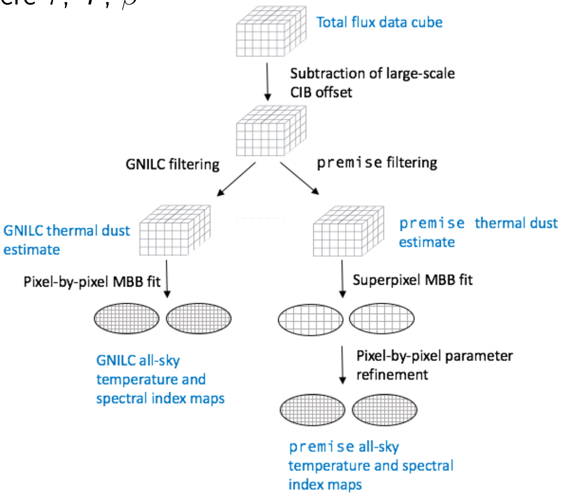
Objectives

- Don't want to smooth
- To remove sub-dominant CIB
- Don't want to complete a per pixel fit
- Parametric fit - not just dust



PREMISE

- Parameter Recovery Exploiting Model Informed Sparse Estimates
- Requires model - here τ, T, β



Filtering and Super-pixels

- Use GNILC filtering BUT

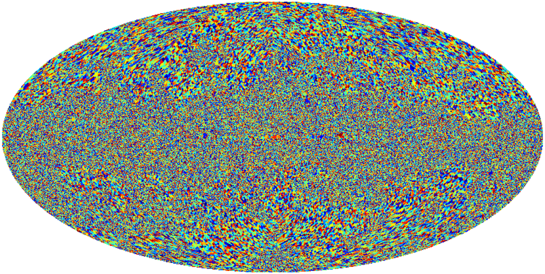
  Marcenko-Pastur distribution

  Penalise in favour of sparsity

$$\min\ (\ ||X - A\alpha\Phi||^2 + \lambda||\alpha||_1^1\ )$$

- Accurate and fast parameter estimates from fit - τ, T, β

χ^2_{red} in wavelet domain!



Refinement

- Low resolution informed initial guesses (64 times faster than pix-by-pix)
- T and β refinement - normalisation factor subject to degeneracies

$$x_{\nu_i}^{\mathrm{dust}} = \tau_{353} \times B(T, \nu) \times \left(\frac{\nu}{353\,\mathrm{GHz}}\right)^{\beta}$$

- Gradient descent at each pixel (until convergance)

$$\beta_n / T_n = \beta_0 / T_0 + \rho \times \Delta((\mathrm{Data} - \mathrm{model})\,\mathrm{w.r.t}\,\beta\,\mathrm{and}\,\mathrm{T})$$

- Threshold all-sky β and T estimates in wavelet domain
- Template for normalisation

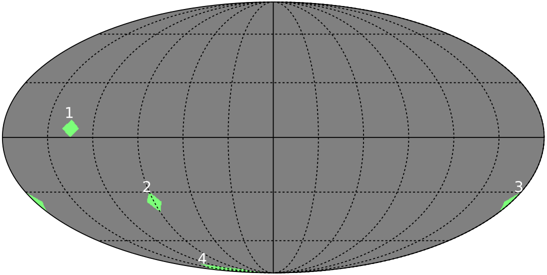
$$\tau_{353} = \frac{X_{857}}{B(T{,}857\,\mathrm{GHz}) \times \left(\frac{\nu}{353\,\mathrm{GHz}}\right)^{\beta}}$$

Validation

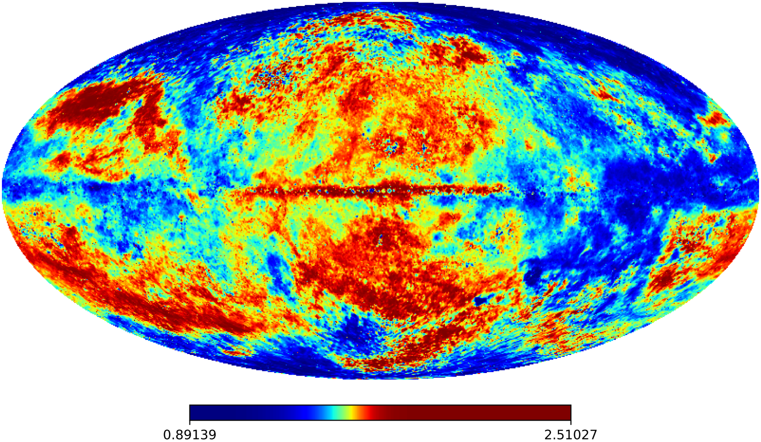
- Simulated data - known parameters

$$x_{\nu_i}^{\text{dust}} = x_{353\,\text{GHz}}^{\text{ffp8}} \times \frac{B(T,\nu)}{B(T,353\,\text{GHz})} \times \left(\frac{\nu}{353\,\text{GHz}}\right)^{\beta}$$

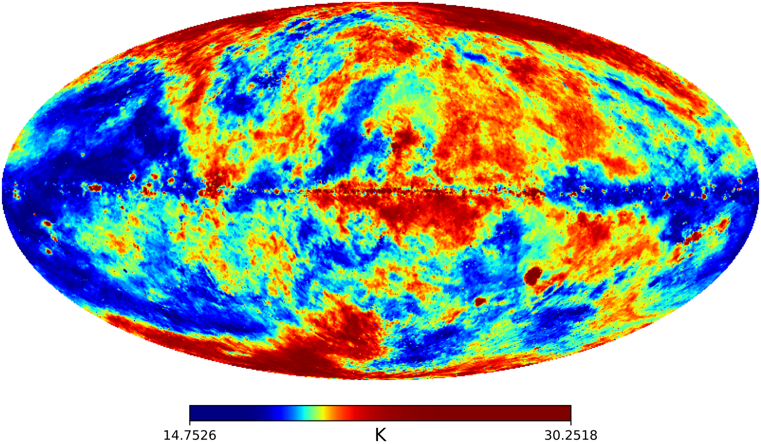
- Comparison with GNILC



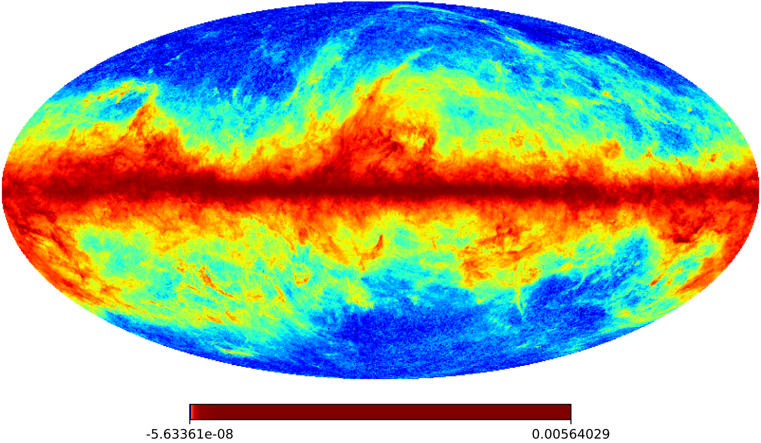
Full-sky β estimate - 5 arcmin



Full-sky T estimate - 5 arcmin

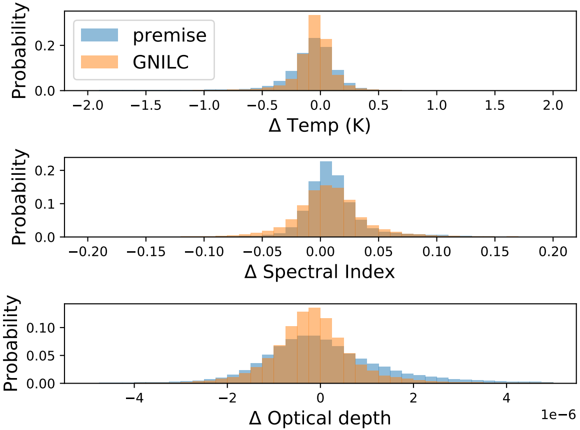


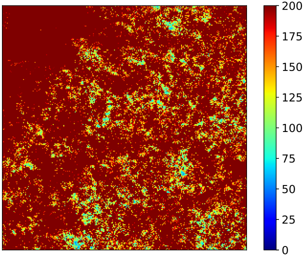
Full-sky τ_{353} estimate - 5 arcmin



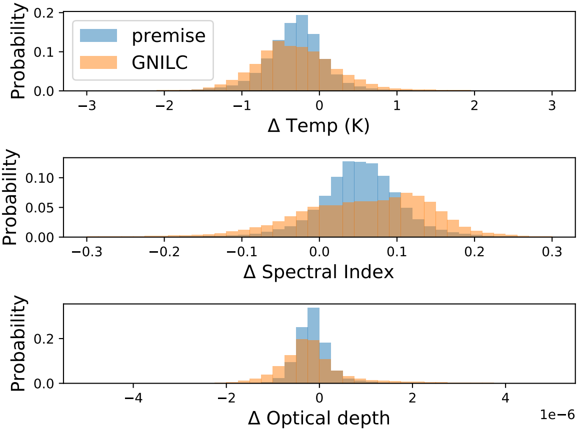
Value	Medium %Δ	1σ %Δ	2σ %Δ	3σ %Δ
Temperature	1.7	2.8	8.0	16.5
Spectral index	3.4	5.7	15.4	25.6
Optical depth (353 GHz)	3.7	7.2	31.2	77.0

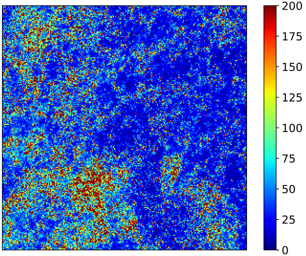
Region 1 - High SNR



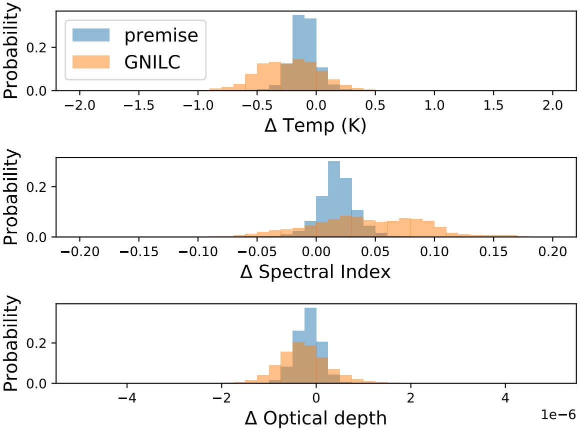


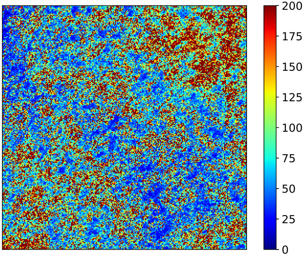
Region 2 - Medium SNR



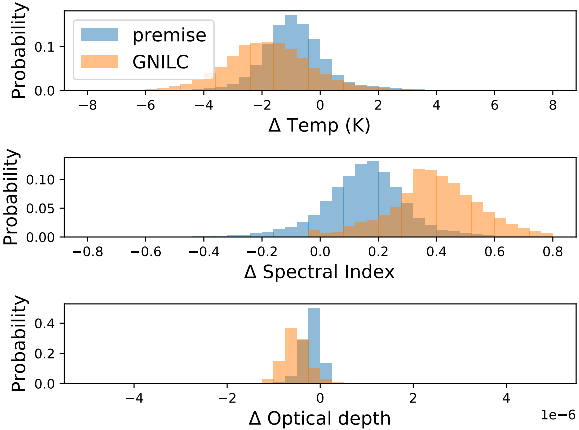


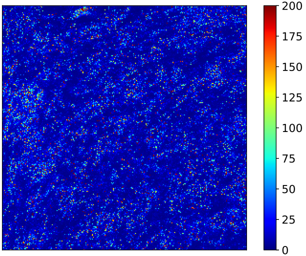
Region 3 - Medium SNR





Region 4 - Low SNR





Conclusion

- Recovery of model parameters: full sky (varying signal to noise) at full resolution
- Sparsity in place of smoothing
- Fast ($\sim$ 2 days)
- Improvement for all but the largest signal to noise regions