

# ArtDeco: A beam-deconvolution code for absolute CMB measurements

E. Keihänen<sup>1</sup> and M. Reinecke<sup>2</sup>

<sup>1</sup> University of Helsinki, Department of Physics, P.O. Box 64, FIN-00014, Helsinki, Finland  
e-mail: elina.keihanen@helsinki.fi

<sup>2</sup> Max-Planck-Institut für Astrophysik, Karl-Schwarzschild-Str. 1, 85741 Garching, Germany  
e-mail: martin@mpa-garching.mpg.de

Preprint online version: November 19, 2012

## ABSTRACT

We present a method for beam-deconvolving cosmic microwave background (CMB) anisotropy measurements. The code takes as input the time-ordered data along with the corresponding detector pointings and known beam shapes, and produces as output the harmonic  $a_{Tlm}$ ,  $a_{Elm}$ , and  $a_{Blm}$  coefficients of the observed sky. From these one can derive temperature and Q and U polarisation maps. The method is applicable to absolute CMB measurements with wide sky coverage, and is independent of the scanning strategy. We tested the code with extensive simulations, mimicking the resolution and data volume of Planck 30GHz and 70GHz channels, but with exaggerated beam asymmetry. We applied it to multipoles up to  $l = 1700$  and examined the results in both pixel space and harmonic space. We also tested the method in presence of white noise.

**Key words.** methods: numerical – data analysis – cosmic microwave background

## 1. Introduction

Removing systematic effects plays an important role in the data analysis of modern high-sensitivity cosmic microwave background (CMB) experiments, such as the WMAP and Planck missions (Jarosik et al. 2011; Tauber et al. 2010). In this work we concentrate on one source of systematic effects, the beam asymmetry. We present an efficient beam deconvolution code, called artDeco. It can be applied to any absolute CMB experiment with sufficient sky coverage.

The required input consists of the time-ordered data (TOD), the corresponding detector pointings, and known beam shapes. The primary output of the code consists of the harmonic  $a_{Tlm}$ ,  $a_{Elm}$ , and  $a_{Blm}$  coefficients of the sky. From these one can derive temperature and Q and U polarisation maps.

This is the usual deconvolution map-making problem (see, e.g., Armitage & Wandelt 2004; Armitage-Caplan & Wandelt 2009; Harrison et al. 2011). Our method differs from the earlier works by an efficient reformulation of the deconvolution problem through heavy use of Wigner functions. The formulation is general, and makes no assumptions on the scanning pattern. Similar ideas have been studied by G. Prézeau, independently of us (private communication).

In the CMB literature the concept of map-making often refers to a procedure that involves removal of correlated noise. Methods for doing this have been treated in several papers (Keihänen et al. 2005, 2010; Poutanen et al. 2006; Ashdown et al. 2007a,b; Kurki-Suonio et al. 2009; Sutton et al. 2009). The methods discussed in these works construct pixelised temperature and Q, U polarisation maps, but do not attempt to correct for an asymmetric beam shape.

Our method steps in at the point where the correlated noise has already been cleaned from the data. The input TOD is assumed to include signal and uncorrelated noise only. We are assuming here that the correlated noise component can be removed

from the data to a sufficient degree prior to the beam deconvolution step. The noise removal step itself is affected by beam asymmetries because signal differences caused by beam mismatch can be falsely interpreted as noise. Fortunately, in destriping methods it is possible to strongly reduce this effect by masking out the regions with the strongest foreground contamination that generate most of the effect. Also, destriping methods produce as natural output the noise-cleaned TOD, which is the required input in deconvolution map-making. Accordingly, destriping methods are the natural counterparts of our deconvolution method.

Since the aim of this paper is to present a new deconvolution method, rather than assess the performance of any particular instrument, we tested the code with a somewhat idealised simulation. In particular, we ignored other systematic effects such as frequency response. On the other hand, we chose strongly asymmetric beam shapes to show the beam effects more clearly.

## 2. Beam deconvolution

### 2.1. Constructing the linear system

In this section we present the deconvolution problem using a formalism based on Wigner functions. We write the solution in a form that allows us to solve the problem numerically in an efficient way. Some details are left for Appendix A.

We start by writing the signal  $t_j$  received by a detector at a given time  $\tau_j$  as

$$t_j = \sum_{slmk} a_{slm} b_{slk}^* D_{mk}^{l*}(\omega_j) + n_j. \quad (1)$$

The  $a_{slm}$  ( $s = 0, \pm 2$ ) are harmonic coefficients that represent temperature and polarisation of the sky signal,  $b_{slk}$  are corresponding beam coefficients,  $n_j$  represents noise, and  $D_{mk}^l(\omega_j)$  is a Wigner

function that defines a rotation of the beam from a fiducial orientation at the north pole to its actual position and orientation. For exact definitions, see Appendix A.

We have adopted a compact notation where  $\omega_j$  denotes a combination of three angles  $\{\phi_j, \theta_j, \psi_j\}$ , which define the detector's orientation. Angles  $\theta_j$  and  $\phi_j$  define a point on the celestial sphere, and  $\psi_j$  defines the beam orientation. The angles can be interpreted as Euler rotation angles.

We perform a simple linear least-squares fit of  $a_{slm}$  to the data up to some multipole  $l_{\max}$ .

In general, a linear least-squares fit leads to an equation of the form

$$x = (A^\dagger C_n^{-1} A + C_s^{-1})^{-1} A^\dagger C_n^{-1} y, \quad (2)$$

where  $x$  is the vector of unknowns,  $y$  is the data vector,  $C_n$  is noise covariance, and  $C_s$  represents *a priori* information on the covariance of  $x$  (Wiener 1949). Matrix  $A$  relates the unknowns to the data through

$$y = Ax + n. \quad (3)$$

In the present case matrix  $A$  is

$$A_{slm}^j = \sum_k b_{slk}^* D_{mk}^{l*}(\omega_j) \quad (4)$$

and  $x$  represents  $a_{slm}$ .

We made two simplifying assumptions, which are not of fundamental nature, however. Firstly, we assume the noise is white with constant variance, and assign equal weights to all samples  $j$ . Secondly, we use no *a priori* information on the sky signal, i.e. the harmonic coefficients are assumed to be uncorrelated and can host a signal of infinite variance. Under these assumptions the covariance matrices drop out of the equation ( $C_n = \text{const}$ ,  $C_s^{-1} = 0$ ). The least-squares solution for  $a_{slm}$  is obtained by solving the linear system of equations

$$\sum_{s'l'm'} \sum_j A_{slm}^{j*} A_{s'l'm'}^j a_{s'l'm'} = \sum_j A_{slm}^{j*} t_j. \quad (5)$$

In the following we process Eq. (5) further to bring it into a form where it can be solved in an efficient way. We treat the right- and left-hand sides separately.

### 2.1.1. Right-hand side

The right-hand side yields

$$\sum_j A_{slm}^{j*} t_j = \sum_{kj} b_{slk} D_{mk}^l(\omega_j) t_j = \sum_k b_{slk} T_{mk}^l. \quad (6)$$

Here we define

$$T_{mk}^l = \sum_j D_{mk}^l(\omega_j) t_j. \quad (7)$$

The triplets  $\omega_j = \{\phi_j, \theta_j, \psi_j\}$  define the detector orientation and are provided as input to the code. To numerically evaluate the transform, we replace the sum over TOD samples by a sum over a three-dimensional grid as follows. We divide the space spanned by all possible triplets  $\omega = \{\phi, \theta, \psi\}$  into bins of equal volume and denote by  $t(\omega)$  the sum of values  $t_j$  falling into one bin. In  $\theta, \phi$  we use HEALPix<sup>1</sup> pixelisation (Górski et al. 2005), and in  $\psi$  we divide the space uniformly. Most elements of  $t(\omega)$  are zero, since a particular HEALPix pixel is typically observed only in a few orientations of the beam.

We call quantity  $t(\omega)$  the *3D signal map*, and quantity

$$T_{mk}^l = \sum_{\omega} D_{mk}^l(\omega) t(\omega) \quad (8)$$

the *Wigner transform of the signal map*.

Here  $\sum_{\omega}$  denotes a sum over all bins. Since most bins are empty, the sum needs to be carried out only over a subset of all  $\omega$  values. Expansion (8) can be evaluated numerically quite efficiently using FFTs and fast algorithms for the generation of Wigner functions.

Quantities  $t(\omega)$  and  $T_{mk}^l$  carry no memory of the order in which the sky was scanned. They just record which pixels of the sky were observed in which orientation of the beam.

### 2.1.2. Left-hand side

Consider then the matrix on the left-hand side of Eq. (5),

$$\sum_j A_{slm}^{j*} A_{s'l'm'}^j = \sum_{jkk'} b_{slk} D_{mk}^l(\omega_j) D_{m'k'}^{l*}(\omega_j) b_{s'l'k'}^*. \quad (9)$$

The multiplier matrix is too large to be inverted. Instead, we use the conjugate gradient (CG) iteration method to solve Eq. (5).

We pixelise the three-dimensional pointing space in the same way we did with the signal, and denote by  $n(\omega)$  the number of hits into bin  $\omega$ . In line with the definition of the 3D signal map, we call  $n(\omega)$  the *3D hit map*.

We replace the sum over samples  $j$  by a sum over bins and write

$$\sum_j D_{mk}^{l*}(\omega_j) D_{m'k'}^l(\omega_j) = \sum_{\omega} D_{mk}^{l*}(\omega) D_{m'k'}^l(\omega) n(\omega). \quad (10)$$

As the next step, we write the Wigner functions as

$$D_{mk}^l(\theta, \phi, \psi) = e^{-im\phi} d_{mk}^l(\theta) e^{-ik\psi}, \quad (11)$$

where  $d_{mk}^l$  are the reduced Wigner functions. Inserting this into (10), we obtain

$$\begin{aligned} & \sum_j D_{mk}^{l*}(\omega_j) D_{m'k'}^l(\omega_j) \\ &= \sum_{\theta} d_{mk}^l(\theta) d_{m'k'}^l(\theta) \sum_{\phi, \psi} e^{i(m-m')\phi} e^{i(k-k')\psi} n(\theta, \phi, \psi) \\ &= \sum_{\theta} d_{mk}^l(\theta) d_{m'k'}^l(\theta) N_{m-m', k-k'}^*(\theta), \end{aligned} \quad (12)$$

where we have defined

$$N_{mk}(\theta) = \sum_{\phi, \psi} e^{-im\phi} n(\theta, \phi, \psi) e^{-ik\psi}. \quad (13)$$

The sums over  $\theta, \phi, \psi$  are carried out over 3D bins. Substituting this back into (9) and rearranging, we bring the left-hand side of the deconvolution equation into the form

$$\begin{aligned} & \sum_{s'l'm'} \sum_j A_{slm}^{j*} A_{s'l'm'}^j a_{s'l'm'} \\ &= \sum_k b_{slk} \sum_{\theta} d_{mk}^l(\theta) \sum_{m'k'} N_{m-m', k-k'}^*(\theta) \sum_{l's'} d_{m'k'}^l(\theta) b_{s'l'k'}^* a_{s'l'm'}. \end{aligned} \quad (14)$$

This can be evaluated much more quickly than the original formula (9), since the summation over the long TOD vector has been replaced by an operation on the more compact  $N$  object. Also, the convolution operations in indices  $k$  and  $m$  can be carried out efficiently with FFTs.

The terms are written in the order in which they are applied in the code.

<sup>1</sup> <http://sourceforge.net/projects/healpix>

## 2.2. Formulation through $3j$ coefficients and preconditioner

Eq. (9) can be written in a yet more compact form.

The product of two Wigner functions can be expressed in terms of  $3j$  symbols (Appendix A.3) as

$$D_{mk}^{l*}(\omega)D_{m'k'}^{l'}(\omega) = (-1)^{m'+k'} \sum_{l_2} (2l_2 + 1) D_{m'-m, k'-k}^{l_2}(\omega) \quad (15)$$

$$\times \begin{pmatrix} l & l' & l_2 \\ m & -m' & (m'-m) \end{pmatrix} \begin{pmatrix} l & l' & l_2 \\ k & -k' & (k'-k) \end{pmatrix}.$$

Substituting this into (10) and that into (9), we arrive at the formula

$$\sum_j A_{slm}^{j*} A_{s'l'm'}^j$$

$$= \sum_{kk'} (-1)^{m'+k'} b_{slk} b_{s'l'k'}^* \sum_{l_2} (2l_2 + 1) W_{m'-m, k'-k}^{l_2} \quad (16)$$

$$\times \begin{pmatrix} l & l' & l_2 \\ m & -m' & (m'-m) \end{pmatrix} \begin{pmatrix} l & l' & l_2 \\ k & -k' & (k'-k) \end{pmatrix}.$$

Indices  $l$  and  $l'$  run from 0 to  $l_{\max}$ . The sum over  $l_2$  covers the values for which  $|l-l'| \leq l_2 \leq l+l'$ . We have defined the *Wigner transform of the hit map* as

$$W_{mk}^l = \sum_{\omega} D_{mk}^l(\omega) n(\omega). \quad (17)$$

We use the formula (16) to explicitly construct the diagonal of the matrix, which serves as a preconditioner to speed up the CG iteration.

The formulation presented here is general, and makes no assumptions on the scanning pattern or on the sky coverage.

## 3. Implementation

### 3.1. User interface

The artDeco code takes two lists of input data objects containing the beam coefficients  $b_{slk}$  and the so-called 3D maps for the respective detector for all detectors that are to be processed simultaneously. The beam coefficients can be read from FITS files in the format that is also being used by the HEALPix package for storing spherical harmonic coefficients.

The 3D maps contain the quantities  $n(\omega)$  and  $t(\omega)$  for a given detector; they are created from time streams of detector pointings and sky signal, respectively, by another stand-alone module called *todtobin*. This module achieves the angular discretisation of the detector pointings by sorting them into pixels of a HEALPix map with a user-defined  $N_{\text{side}}$  parameter, which are in turn subdivided into  $n_{\psi}$  bins for the discretisation in  $\psi$ . The choice of  $N_{\text{side}}$  and  $n_{\psi}$  has direct consequences on the run-time of the deconvolver and the accuracy of its results: if high values are chosen, the 3D map will become very large, and the deconvolver will require more memory and CPU time. On the other hand, too low values will result in noticeable discretisation errors and inaccurate output  $a_{Xlm}$ . As a guideline,  $N_{\text{side}}$  should be chosen in such a way that the smallest features of all beams are well resolved at the angular resolution

$$\delta\alpha \approx 3500'/N_{\text{side}} \quad (18)$$

associated with this  $N_{\text{side}}$ . In a similar fashion,  $n_{\psi}$  should be large enough to resolve azimuthal features of the beams (assuming the beams are centred on a pole). For completely axisymmetric

beams,  $n_{\psi} = 1$  is sufficient. Generally,  $n_{\psi} = 10k_{\max}$  (where  $k_{\max}$  is the highest azimuthal multipole used to describe the beams) is a good choice.

The computational complexity of *todtobin* is approximately  $\mathcal{O}(n_{\text{samp}} \log n_{\text{samp}})$ , where  $n_{\text{samp}}$  is the total number of TOD samples, since its runtime is dominated by the sorting of the input TOD according to HEALPix pixel number and  $\psi$  bin. Consequently,  $N_{\text{side}}$  and  $n_{\psi}$  only have a significant effect on the size of the output 3D maps. Typical execution times for *todtobin* with several billion samples (as used for the experiments shown in this paper) are of the order of a few minutes.

ArtDeco takes several user-supplied parameters that influence its behaviour. Most important among these are  $l_{\max}$ , which specifies the maximum multipole up to which deconvolution is carried out, and  $k_{\max} \leq l_{\max}$ , which limits the maximum azimuthal multipole used for the beams. The explicit introduction of  $k_{\max}$  is advantageous since most beams can be expressed accurately even when using a  $k_{\max} \ll l_{\max}$  (even  $k_{\max} = 0$  if the beam has rotational symmetry and is unpolarised), and the reduction in  $k_{\max}$  reduces CPU and memory requirements dramatically.

ArtDeco's output consists of a set of  $a_{lm}$  in HEALPix-compatible format. Depending on whether the user requested polarisation, the file contains only  $a_{Tlm}$  or additional  $a_{Elm}$  and  $a_{Blm}$  coefficients.

### 3.2. Algorithm overview

The deconvolution algorithm has been implemented in C++, making use of the MPI library for parallelisation. The quantities that depend on  $\theta$ , such as  $d$  and  $N$ , are distributed to MPI tasks according to the  $\theta$  angle. This allows a fairly good load balance. Summation over  $\theta$ , when needed, is done efficiently through the collective `MPI_Reduce()` routine. The  $a_{slm}$  coefficients are distributed according to index  $m$ . The beam coefficients  $b_{slk}$  are fully present on all tasks.

The procedure of evaluating Eq. (14) can be summarised as follows. First we multiply the input  $a$  by the beam coefficients  $b$ . Then we perform a loop over index  $m'$ , inside which we broadcast the  $a_{s'l'm'}$  coefficients for the current  $m'$  to all processes, multiply by  $b$ , construct the reduced Wigner matrices for the local values of  $\theta$ , and accumulate over  $l'$ . The multiplication by matrix  $N$  is a convolution operation in indices  $m, k$  and can be performed efficiently using 2D FFTs. Each MPI task performs the multiplication by  $N$  independently for the local  $\theta$ . After that, we make another loop over index  $m$ , construct again the Wigner matrices, sum over  $\theta$ , collect the result according to  $m$  to the owner process, and multiply by the beam coefficients.

The solution is assumed to have converged sufficiently as soon as the residual of the current  $a_{slm}$  estimate is less than  $10^{-12}$  times the residual of a vector containing all zeroes.

To generate the  $3j$  symbols needed to construct the preconditioner, we use a C implementation of the recursive algorithm described by Schulzen & Gordon (1975).

For a more detailed technical description of the algorithm, see Appendix B. CPU and memory requirements of various runs are presented and discussed in Appendix C.

## 4. Simulation setup

The deconvolution algorithm was tested using mock data produced by the Planck simulation package (Level-S, Reinecke et al. 2006).

**Table 1.** Overview of the detector parameters

Name	Frequency	ellipticity	FWHM	$\psi_{\text{pol}}$	$\sigma_w(\mu\text{K})$
30 <sub>0,0</sub>	30GHz	1.7	32.2352'	-22.3°	514
30 <sub>0,1</sub>	30GHz	1.8	32.1377'	67.6°	514
30 <sub>1,0</sub>	30GHz	1.7	32.2352'	22.3°	514
30 <sub>1,1</sub>	30GHz	1.8	32.1377'	-67.6°	514
70 <sub>0,0</sub>	70GHz	1.7	13.0328'	22.2°	-
70 <sub>0,1</sub>	70GHz	1.8	12.9916'	-67.5°	-
70 <sub>1,0</sub>	70GHz	1.7	13.0328'	-22.2°	-
70 <sub>1,1</sub>	70GHz	1.8	12.9916'	67.5°	-

**Notes.** The first subscript denotes the horn, the second the number of the detector inside the horn.  $\sigma_w$  is the standard deviation of the Gaussian random numbers that are added to each TOD sample in simulations with white noise. Except for the ellipticities (which were exaggerated to stress-test our algorithm), the values are roughly comparable to those given in Tables 1 and 2 in ?. The absolute value of the individual  $\psi_{\text{pol}}$  is not relevant in the context of this paper, only their pairwise differences are.

#### 4.1. Mission parameters

We assumed a scanning strategy that caused the telescope's spin axis to perform two cycloidal excursions from the ecliptic plane per year with an amplitude of 7.5°; in azimuthal direction, the spin axis always pointed towards the Sun.

Time-ordered data were generated for a time span of 10 000 hours, corresponding to slightly more than two full sky surveys; their length is of the order of  $10^9$  samples.

#### 4.2. Detector geometry

For the first set of experiments, data were simulated for two horns, i.e. two pairs of detectors, which were modelled roughly following Planck's 30GHz channel. In particular, the detectors had an average FWHM of 32', the polarisation directions in each detector pair were rotated by about (but not exactly) 90° against each other, and the polarisation directions of the two horns were in turn shifted by 45°. Both horns followed the same scan path on the sky, although with a little time delay.

The beam shapes were assumed to be elliptical, with ellipticities of 1.7 and 1.8 for each detector in a pair, respectively. These values are considerably higher than in a typical experiment and were chosen to demonstrate the performance of the method even under unfavourable conditions.

In order to show the algorithm's performance at higher resolutions, another data set was produced using two horns sensitive at 70GHz and with an average FWHM of 13'.

Table 1 gives an overview over the detector properties.

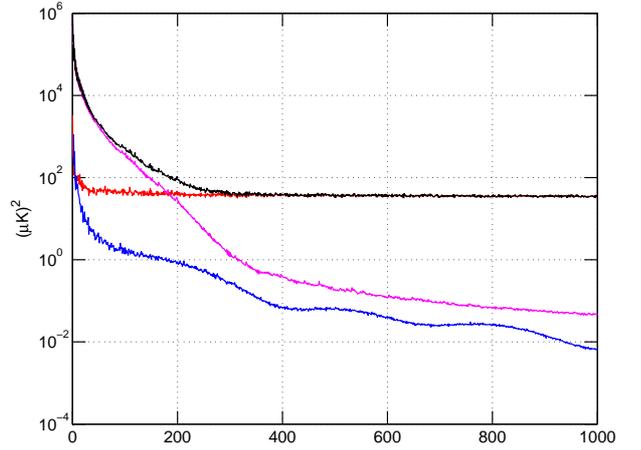
#### 4.3. Input data

To generate a theoretical power spectrum for the CMB emission, we used the CAMB<sup>2</sup> code with the cosmological parameters  $\Omega_b = 0.045$ ,  $\Omega_{\text{cdm}} = 0.255$  and  $\Omega_\lambda = 0.7$ . Using the HEALPix tool `syn_alm_cxx`, we created a random, unconstrained realisation of polarised  $a_{lm}$  from this power spectrum.

Emission maps of foreground components were generated using the Planck pre-launch Sky model<sup>3</sup> (PSM, Delabrouille et al. 2012), version 1.6.6. We chose to include

<sup>2</sup> <http://camb.info>

<sup>3</sup> <http://www.apc.univ-paris7.fr/~delabrou/PSM/psm.html>



**Fig. 1.** Spectra of the 30GHz input sky components: CMB (blue), diffuse foregrounds (purple), point sources (red), and total (black).

the contributions from galactic infrared emission (galactic synchrotron radiation, free-free emission, as well as thermal and spinning dust).

The signal from strong point sources was taken into account as well; to achieve this, a PSM-generated catalogue of point sources was directly expanded into spherical harmonic coefficients, using a sufficiently high cutoff value  $l_{\text{max}}$  that the beam response was negligible at this scale.

Polarised emission was disabled for all contributions except for the CMB, which makes it very easy to detect leakage of the strong temperature signal in the galactic plane into the polarisation signal.

We neglected bandpass effects and assumed an identical delta frequency response for all detectors.

The power spectra of the signal components at 30 GHz are plotted in Fig. 1.

#### 4.4. Noise

In some of our simulations, detector noise was added to the idealised sky signal. Since the deconvolution method requires correlated noise to be removed from the time-ordered data beforehand, we only added Gaussian white noise with an appropriately chosen  $\sigma$  to the simulated samples (see Table 1).

#### 4.5. 3D map generation

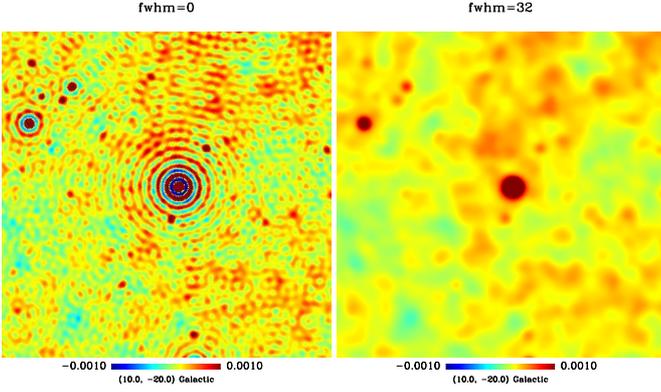
The mock TOD were processed with the `todtobin` tool to generate the 3D input maps required by the deconvolver. We set  $N_{\text{side}} = 1024$  for the 30GHz case, and  $N_{\text{side}} = 2048$  for 70GHz;  $n_\psi$  was always set to 256.

## 5. Results

### 5.1. Constructing a sky map

The primary output of the deconvolution code consists of the  $a_{Xlm}$  ( $X=T,E,B$ ) coefficients of the sky, up to  $l_{\text{max}}$ . From these one can construct the usual temperature and Q,U polarisation maps through harmonic expansion of the coefficients. We did this with the `alm2map_cxx` tool of the HEALPix package.

For comparison, we also produced binned maps directly from the TOD. A binned map is constructed by assigning each



**Fig. 2.** Ringing due to insufficient smoothing. A map constructed from the input  $a_{Tlm}$  coefficients with  $l_{\max}=800$  with no smoothing (*left*) or smoothed to FWHM=32' (*right*). The cut-off of the spectrum at  $l_{\max}$  leads to strong ringing effects around a point source.

TOD sample completely to the pixel containing the centre of the beam. The binning operation can be formally written as

$$m = (P^T C_n^{-1} P)^{-1} P^T C_n^{-1} y. \quad (19)$$

Here  $P$  is a pointing matrix,  $C_n$  is white noise covariance, and  $m$  is a (I,Q,U) map triplet.

Binned maps are distorted by beam effects if the beam is asymmetric. In particular, the polarisation component of a binned map is contaminated by temperature leakage through beam shape mismatch. We aim at demonstrating that deconvolution can reduce these undesired effects.

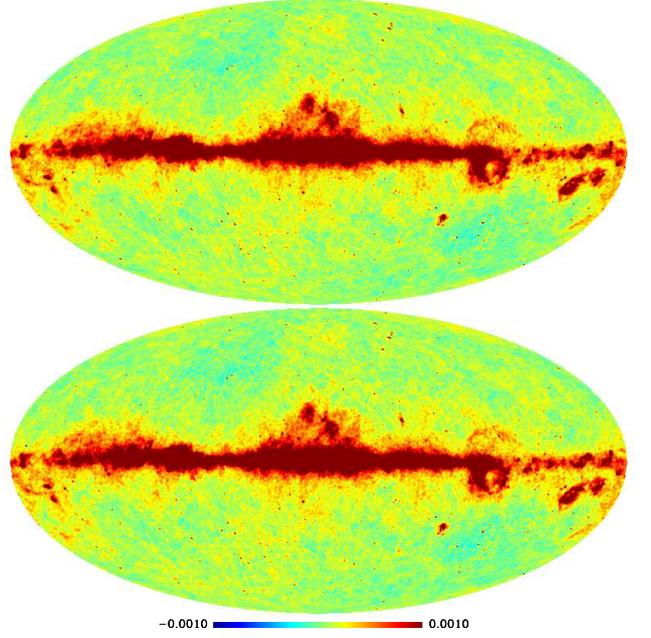
Since our simulation includes point sources, the input  $a_{Xlm}$  coefficients are not band-limited. However, we can only solve the coefficients up to some limited  $l_{\max}$ . This leads to ringing artefacts, if a map is constructed directly from the raw  $a_{Xlm}$  coefficients. Ringing is particularly prominent around point sources and other sharp features.

Ringing is caused by the sharp cut-off of the spectrum at  $l_{\max}$ , and it occurs even if the deconvolved coefficients are fully correct up to  $l_{\max}$ . For this reason it is necessary to smooth the coefficients with a symmetric Gaussian beam prior to constructing a map. The required amount of smoothing depends on the spectrum of the underlying sky, and on  $l_{\max}$ . The highest  $l_{\max}$  that one can solve in turn depends on the beam width. In other words, it is not possible to recover structures significantly smaller than the beam itself. This gives the following rule of thumb: the required smoothing width is given by the smallest features of the original beam.

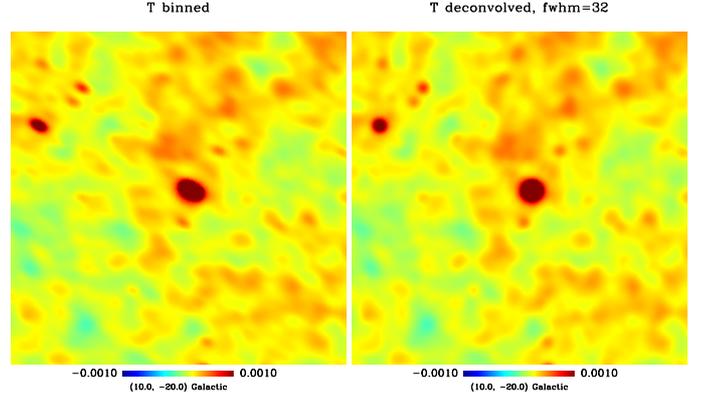
In Fig. 2 we demonstrate the ringing around a point source. We show an image of a point source, constructed from the input  $a_{Xlm}$ , but including only coefficients up to  $l_{\max}=800$ . Ringing artefacts are evident. On the right we show the same region of the sky after the map is smoothed with a FWHM=32' Gaussian symmetric beam. Smoothing removes the ringing, but naturally also smoothes out the smallest structures in the map.

## 5.2. Signal-only simulations with elliptical beam

We analyse first the noise-free 30GHz simulation in detail. A detailed description of the simulation was given in Section 4. We ran the code with various combinations of parameters  $l_{\max}$ ,  $k_{\max}$ , and analysed the results both in pixel space and in harmonic space. We included data from all four detectors and considered both temperature and polarisation.



**Fig. 3.** 30GHz temperature map without noise: binned (*top*), and deconvolved and FWHM=32' smoothed (*bottom*).



**Fig. 4.** Zoom into the 30GHz temperature map without noise: binned (*left*) and deconvolved (*right*). Shown is a 1000'x1000' patch of the sky. The deconvolved map can be compared to the map constructed from the input  $a_{Tlm}$  (right panel of Fig. 2).

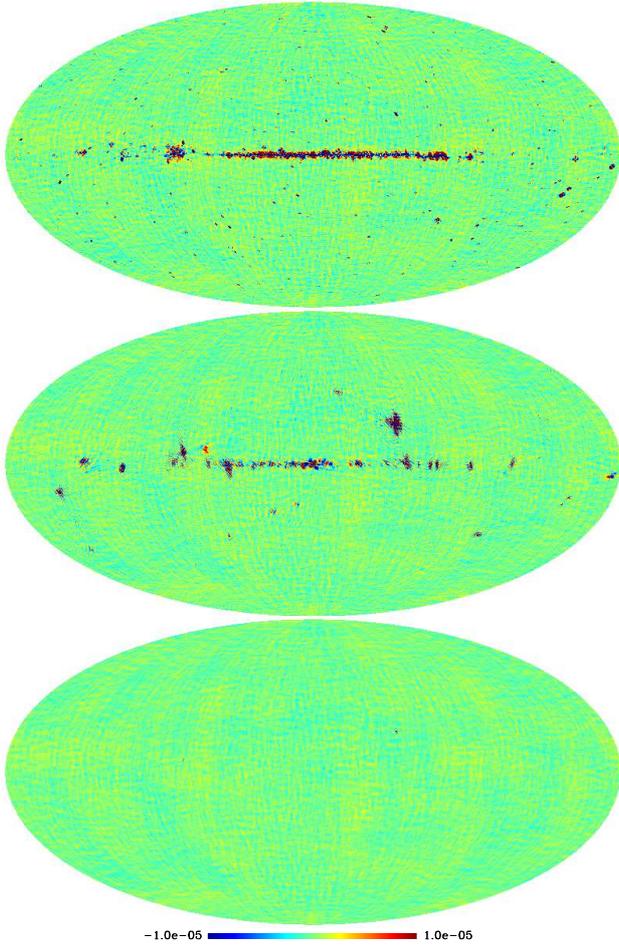
### 5.2.1. Temperature map

In Fig. 3 we show a full-sky Mollweide projection of the binned temperature map together with that of the deconvolved map. We used deconvolution parameters  $l_{\max}=800$ ,  $k_{\max}=6$ .

We chose the value FWHM=32' as the baseline smoothing width for the 30GHz maps. This corresponds to the mean width of the input beam. In some cases it was necessary to apply even stronger smoothing.

The difference between the binned and the deconvolved map does not appear dramatic in the full-sky map image, but becomes evident when we zoom into a point source. Figure 4 shows a randomly selected strong point source below the galactic plane. While the source in the binned map is clearly elongated due to beam shape, it appears circular in the deconvolved map.

The deconvolved image can be compared to the right panel of Fig. 2, where we have shown the same patch of the sky constructed from the input  $a_{Xlm}$  coefficients. The maps are nearly identical.



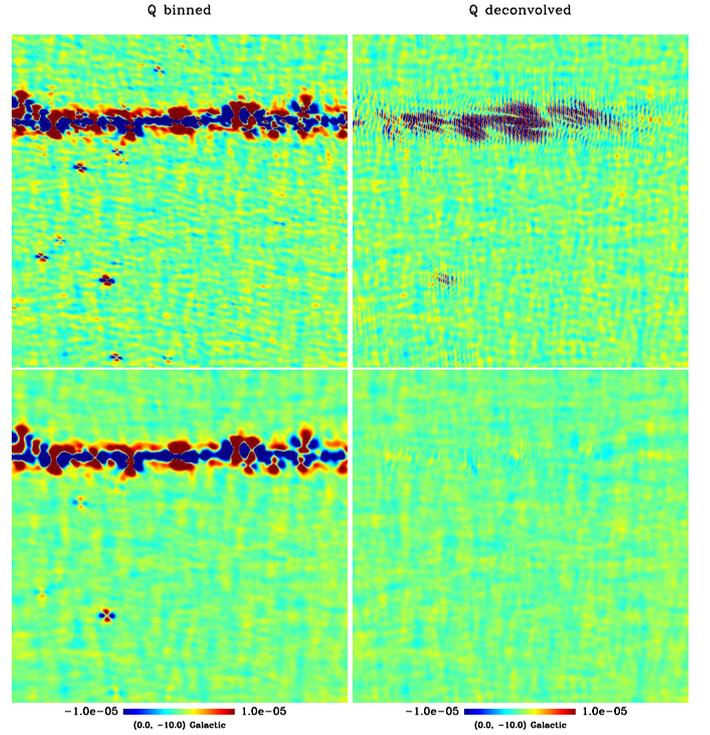
**Fig. 5.** 30GHz Q polarisation maps, in absence of noise: binned (*top*), and deconvolved with FWHM=32' (*middle*), or FWHM=45' (*bottom*) smoothing. The whole galactic structure is a fake polarisation signal arising from beam mismatch.

### 5.2.2. Polarisation maps

The beam effects show up more dramatically in the polarisation maps (Fig. 5). The temperature signals recorded by two detectors differ due to the different beam shapes. The binning procedure interprets this erroneously as polarisation signal. This leads to a leakage of temperature signal into the polarisation maps. Since our simulation did not include polarised foregrounds, the whole galactic structure seen in the binned map is temperature leakage through beam mismatch.

We show deconvolved maps with two levels of smoothing below the binned map. Smoothing with a FWHM=32' beam is not sufficient to fully remove the galactic residual, but if we smooth the map further to FWHM=45', the residual disappears almost completely.

In Fig. 6 we show a zoom into the Q polarisation map, just below the galactic plane. In the binned map we see a strong galactic residual, and typical “four-leaf clover” structures that arise from temperature leakage at the location of a point source. We show the deconvolved map again for two levels of smoothing: FWHM=32' and FWHM=45'. The additional smoothing removes the clover structure nearly completely. To show that this is not only an effect of more aggressive smoothing, we smoothed the binned map further with a Gaussian beam of width FWHM=32', which, combined with the 32' of the original beam, gives a total smoothing of 45'. This map is shown below the un-



**Fig. 6.** Zoom into the 30GHz Q polarisation map without noise: binned (*left*) and deconvolved (*right*). The maps were smoothed to resolution FWHM=32' (*top*), or to 45' (*bottom*). We show a 2500'x2500' patch of the sky. The horizontal structure is spurious signal due to temperature leakage from the galaxy.

smoothed binned map. The additional smoothing has little effect on the galactic residual.

### 5.2.3. Harmonic space

We proceed to analyse the deconvolution results in harmonic space. We computed the TT, EE, and TE spectra of the deconvolved  $a_{Xlm}$  coefficients and compared them to the corresponding input spectra. The spectrum is constructed as

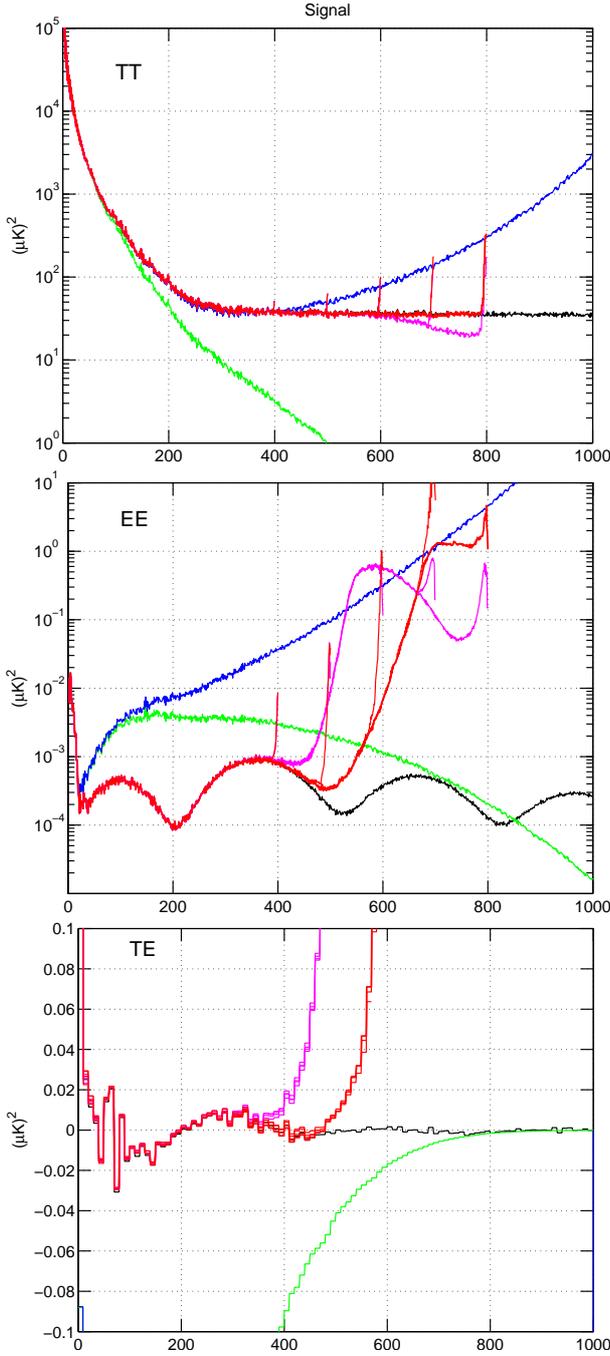
$$C_l^{XY} = \frac{1}{2l+1} \sum_{m=-l}^l a_{Xlm}^* a_{Ylm},$$

where  $X$  and  $Y$  stand for  $T$  and  $E$ .

We ran the code for all combinations of  $k_{\max}=4,6$  and  $l_{\max}=400,500,600,700,800$ . Results for different combinations are shown in Fig. 7. We applied no smoothing to the coefficients.

We observe that the optimal value for  $k_{\max}$  is coupled to  $l_{\max}$ . Results for  $k_{\max}=4$  and  $k_{\max}=6$  begin to diverge only above  $l=400$ . The results for different  $l_{\max}$  with  $k_{\max}=6$  are practically on top of each other, except for the highest multipoles near  $l_{\max}$ , which are clearly distorted. When constructing a map, we cut out the last 20 multipoles before making the harmonic expansion.

For comparison we also show the spectrum obtained from a harmonic expansion of the binned map, which we computed using the anafast tool of the HEALPix package. This decreases rapidly as a function of  $l$  as a result of beam smoothing. To correct for the smoothing, we divided the spectrum by the window function of a symmetric Gaussian beam with FWHM=32'. This corrects for the mean smoothing and allows one to recover the correct spectrum up to  $l_{\max}=300$ . Above that, the result begins to deviate from the input spectrum due to beam ellipticity, which



**Fig. 7.** Noise-free 30GHz TT (*top*), EE (*middle*), and TE (*bottom*) spectra. Shown are the input (*black*) and the spectra constructed from the  $a_{Xlm}$  coefficients from deconvolution. Deconvolution results are shown for  $k_{\max}=4$  (*purple*) and  $k_{\max}=6$  (*red*), and for  $l_{\max}=400, 500, 600, 700, 800$ . The best-case results ( $l_{\max}=800, k_{\max}=6$ ) are plotted with a thicker linetype. For comparison we show also the spectrum obtained from harmonic expansion of a naive binned map (*green*). The spectrum of the binned map decreases rapidly with  $l$  because the binning procedure inherently produces a beam-smoothed map. The *blue* line (TT and EE) is obtained by dividing the latter by the window function of a symmetric Gaussian beam with  $\text{FWHM}=32'$ . The TE spectra have been binned over ten multipoles to reduce scatter and to make the plot more readable.

cannot be corrected for by application of a window function only.

Deconvolution recovers the TT input spectrum well up to  $l=800$ . At higher multipoles there is little information left in the signal, and the convergence properties of the code become poor. The weaker EE and TE spectra are recovered up to  $l=400$ . Owing to temperature leakage, the binned map only gives the correct EE spectrum for the ten first multipoles in EE, and the TE spectrum is useless.

We did one run with parameters  $l_{\max}=600, k_{\max}=8$  (not shown), to see if it would help to recover the EE spectrum to even higher multipoles. The results, however, did not differ significantly from the  $l_{\max}=600, k_{\max}=6$  case.

#### 5.2.4. Reconstruction error of harmonic coefficients

A comparison between the reconstructed spectrum and the input spectrum is not a fully reliable test of the correctness of the deconvolution result. It is possible to construct a set of harmonic coefficients that provides the correct spectrum, although the individual coefficients are incorrect.

To assess directly the accuracy of the  $a_{Xlm}$  coefficients, we constructed a quantity we call the error spectrum. It is computed as follows. We take the difference between the  $a_{Xlm}$  coefficients we obtain from deconvolution and the corresponding input coefficients  $a_{Xlm}^0$ . As before,  $X$  stands for T or E. We compute the spectrum of this difference as

$$C_{Xl}^{\text{err}} = \frac{1}{2l+1} \sum_{m=-l}^l (a_{Xlm} - a_{Xlm}^0)^* (a_{Xlm} - a_{Xlm}^0).$$

The resulting spectrum is a measure of the error in the reconstruction of the  $a_{Xlm}$  coefficients, as a function of scale.

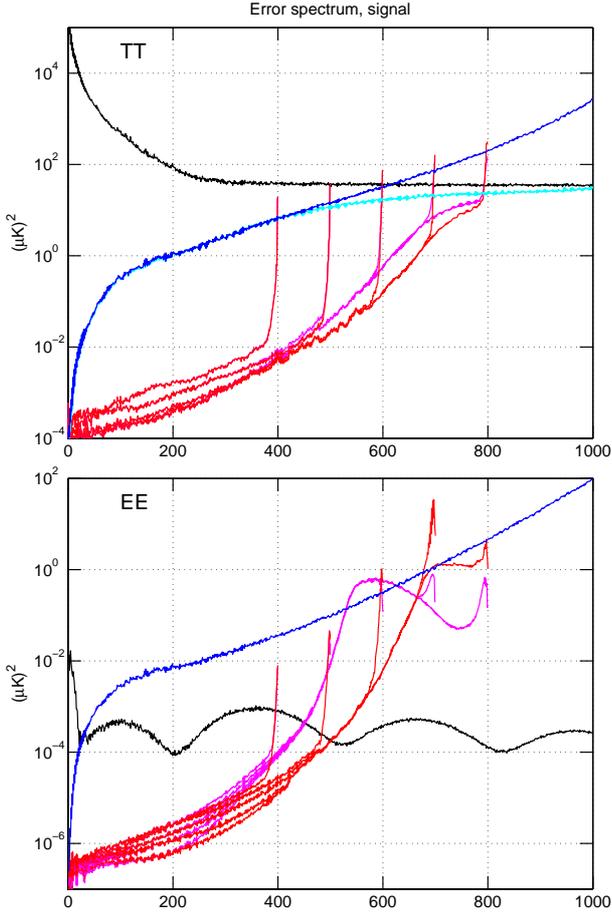
We show the error spectra of the 30GHz runs for T and E coefficients in Fig. 8. For comparison, we show the input TT or EE spectrum in the same figure. The  $a_{Xlm}$  coefficients are recovered with good accuracy when the error spectrum is well below the input spectrum. The meaning of the blue line is the same as in Fig. 7, that is, we compute the harmonic expansion of the binned map and divide them by the window function of a symmetric Gaussian beam ( $\text{FWHM}=32'$ ).

We made another comparison with an ideal window function, which we computed as the ratio of the uncorrected spectrum of the binned map, and the input spectrum (green and black lines in the top panel of Fig. 7). The error spectrum for this is shown in light blue in Fig. 8 (TT spectrum only). By construction, applying the ideal window function to the spectrum of the binned map returns exactly the correct input spectrum. The error spectrum, however, is not improved with respect to the result obtained with Gaussian window function, which shows that the failure of the binning procedure to recover the correct spectrum above  $l = 300$  in Fig. 7 was not just a result of a poorly chosen window function.

The error spectrum reveals that the deconvolved  $a_{Xlm}$  coefficients are more accurate than those obtained by harmonic expansion of the binned map already at the lowest multipoles.

#### 5.3. Simulations with noise

The first simulation was quite idealised, since the data were noise-free. We made another simulation, where we added white noise to the TOD. In other aspects the simulation was identical

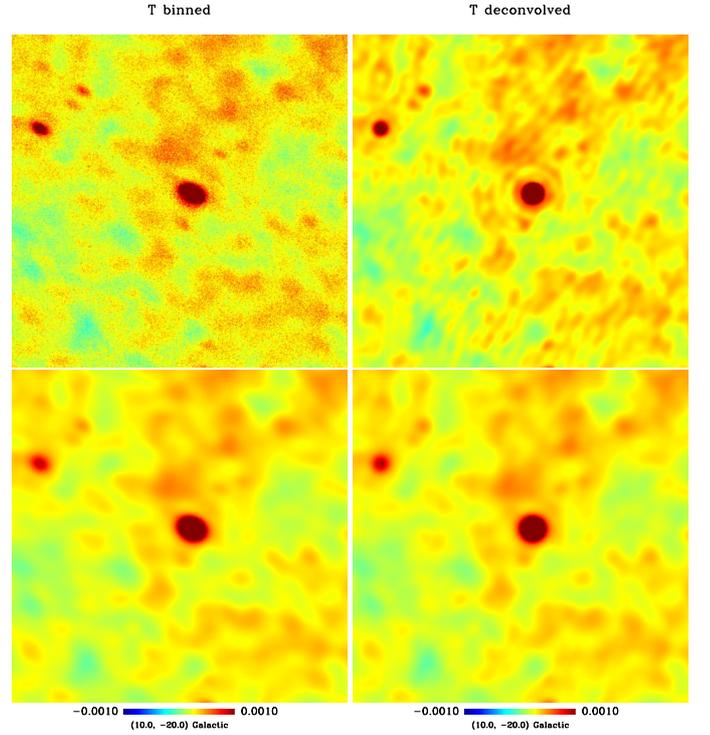


**Fig. 8.** 30GHz error spectrum, for T (*top*) and E (*bottom*), without noise. See main text for definition. The deconvolution results are shown for the same combinations of parameters as in Fig. 7. The input spectrum is shown in *black*. Also shown is the error spectrum for a harmonic expansion of the binned map, corrected with a window function of a symmetric Gaussian beam (*blue*), or (T only) with an ideal window function (*light blue*). The ideal window function was defined as the ratio of the input spectrum and the spectrum of the uncorrected binned map.

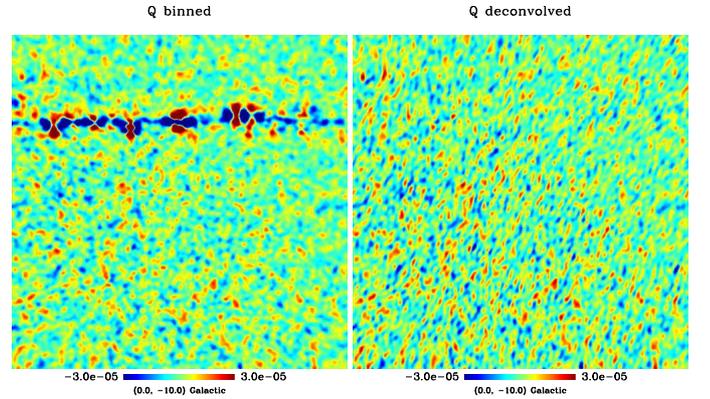
to the first one. No correlated noise was added. We are implicitly assuming that correlated noise components, if present, were removed to sufficient accuracy prior to deconvolution.

As expected, noise made the deconvolution process more difficult, and the code did not converge for the highest values of  $l_{\max}$ . We reached full convergence for the combinations  $l_{\max}=600$ ,  $k_{\max}=6$ , and  $l_{\max}=700$ ,  $k_{\max}=4$ . Case  $l_{\max}=700$ ,  $k_{\max}=6$  did not converge anymore.

Fig. 9 shows a zoom into a point source, the same as in Fig. 4, in presence of white noise. The deconvolution parameters were  $l_{\max}=600$ ,  $k_{\max}=6$ . The deconvolved map smoothed with the fiducial FWHM=32' is contaminated by distorted noise. Also, we see some ringing around the point source due to the moderately low  $l_{\max}$ . We therefore smoothed the map further to FWHM=45', which was enough to remove the distortion. For comparison we also smoothed the binned map to comparable resolution by applying a smoothing by another FWHM=32' to it. This, combined with the original beam, gives a total resolution of roughly FWHM=45'. These two maps are shown in the bottom row. Even after the additional smoothing, the point source in



**Fig. 9.** Zoom into the 30GHz temperature map in presence of noise: binned (*left*), and deconvolved (*right*). In the *top* row we show the unsmoothed binned map and the deconvolved map with FWHM=32' smoothing. The deconvolved map is distorted by ringing and amplification of noise. In the *bottom* row we show the same maps smoothed further to FWHM=45'. The additional smoothing removes the distortion.



**Fig. 10.** Zoom into the 30GHz Q polarisation map: binned (*left*) and deconvolved (*right*) in presence of noise. The maps were smoothed to FWHM=45'. As in the noiseless case, the smoothed deconvolved map does not exhibit the spurious leakage signal in the galactic plane.

the binned map is elongated. Deconvolution restores the circular shape of the source.

Figure 10 shows a zoom into the Q polarisation map. We show the same patch of the sky as in Fig. 6. The raw binned map (not shown) is fully noise-dominated. We show maps smoothed to FWHM=45'. Again, deconvolution removes the galactic leakage that is apparent in the binned map.

Figure 11 shows the spectra constructed from the deconvolved  $a_{Xlm}$  coefficients for the parameter combinations for

which the code converged fully. Again we also show the spectrum of the binned map, and the same corrected for a symmetric Gaussian beam with  $\text{FWHM}=32'$ . Deconvolution recovers the true TT spectrum up to nearly  $l = 600$ , after which noise begins to dominate. The spectrum of the binned map begins to deviate already below  $l = 400$ . In the case of the EE spectrum, both deconvolution and binning are unable to recover the true spectra except for the very lowest multipoles. In the case of the TE spectrum, deconvolution performs considerably better than binning, and the result follows the input spectrum up to  $l = 300$ , although it is noisy.

Finally, in Fig. 12 we show the error spectra in presence of noise. Deconvolution recovers the  $a_{Tlm}$  coefficients well up to  $l = 600$ . In the case of  $a_{Elm}$  coefficients, the error is well above the input spectrum for all but the lowest multipoles.

It is a well-known and often quoted fact that deconvolution tends to amplify noise at high multipoles. We see this clearly in the upper deconvolved map in Fig. 9. The effects can, however, be suppressed by applying a sufficiently strong smoothing to the map, as we see from the lower panels of the same figure.

We can look at the amplification of noise also in harmonic space (Fig. 11). While the TT spectrum of the binned map (green line) continues to fall until it levels out at  $l = 900$  (not seen in the plot), the deconvolution spectrum begins to rise above  $l = 600$ . Note, however, that for the whole multipole range where we have a deconvolution result, the extra power in the deconvolved map is still below that in the binned map corrected for a symmetric beam (blue line).

In the EE spectrum, the power of the deconvolved map exceeds that in the binned map corrected for a symmetric beam, around  $l = 300$ . This is in the domain where we have full noise domination in any case.

#### 5.4. 70GHz simulation

To test the performance of the code at higher multipoles, we generated one noise-free simulation mimicking the data volume of the Planck 70GHz channel. The used elliptic beams have an average FWHM of  $13'$ ; for more parameters see Table 1. We ran the deconvolution code with parameters  $l_{\text{max}}=1700$ ,  $k_{\text{max}}=6$ .

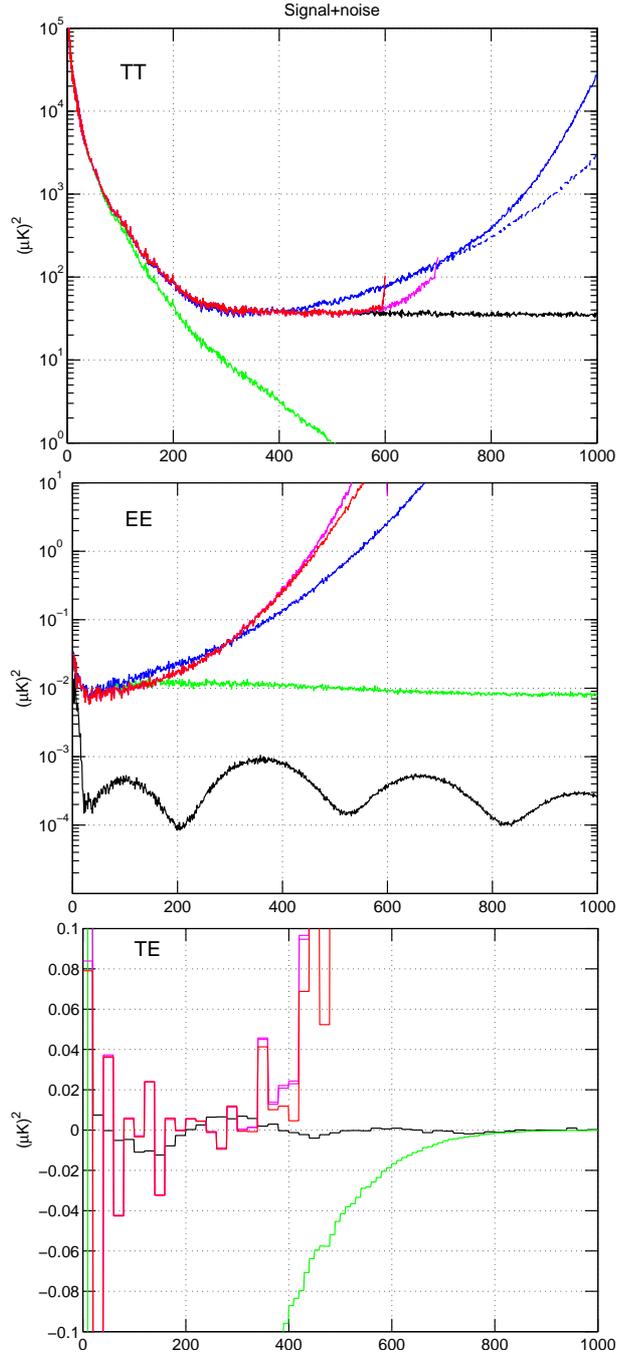
Temperature and polarisation maps constructed from the  $a_{Xlm}$  with resolution  $\text{FWHM}=13'$  (T), or  $\text{FWHM}=18'$  (Q) are shown in Figs. 13 and 14 together with the corresponding binned maps. We show the same region of the sky as with 30GHz simulations. Again, deconvolution has worked well, returning the circular shape of the point source, and removing the galactic leakage in the Q polarisation map.

In Fig. 15 we show the TT and EE error spectra for the 70GHz simulation. The  $a_{Tlm}$  are recovered well nearly up to  $l = 1700$ , and the  $a_{Elm}$  coefficients well above  $l = 1000$ . Thus we have shown that, with sufficient computational resources, our deconvolution method can be applied to realistic data sets of volume and resolution of the Planck 70GHz channel.

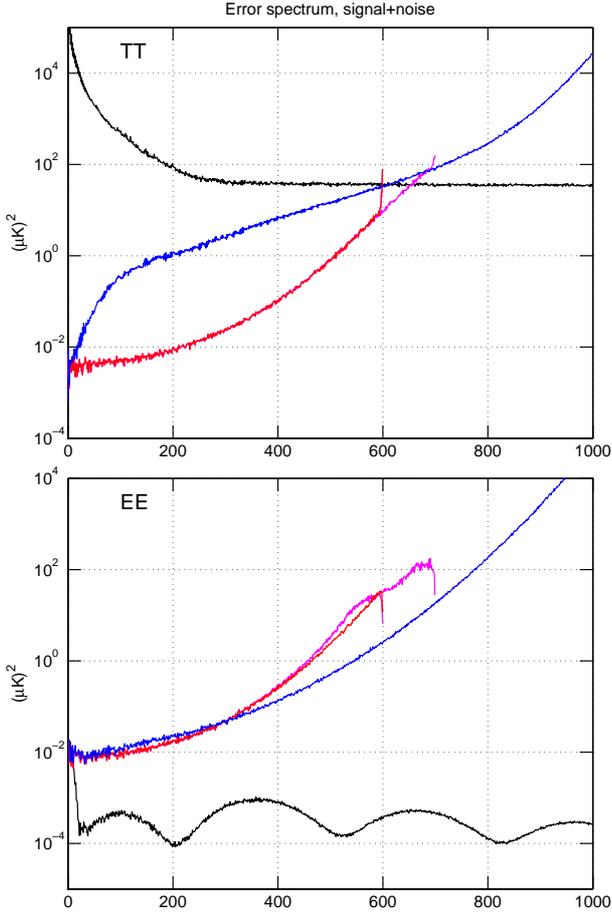
#### 5.5. Convergence

The convergence criterion we had set in our code is rather strict. We require that the square of the residual vector has fallen below a fraction of  $10^{-12}$  of its initial value.

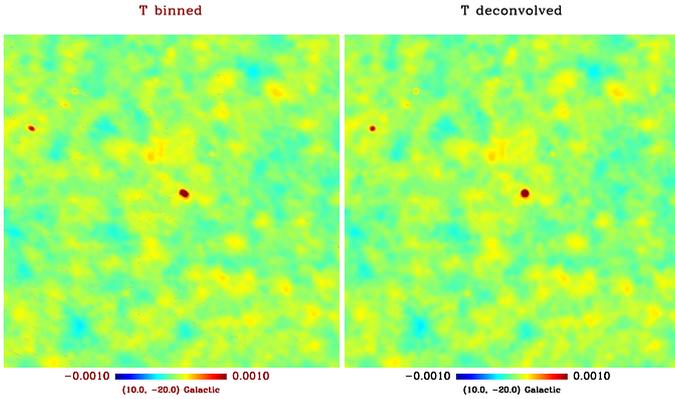
To see if a less strict criterion would help in reducing the computation time, we studied the convergence of the solution. We dumped the  $a_{Xlm}$  coefficients after every 50 steps, and computed the error spectrum with respect to the input coefficients.



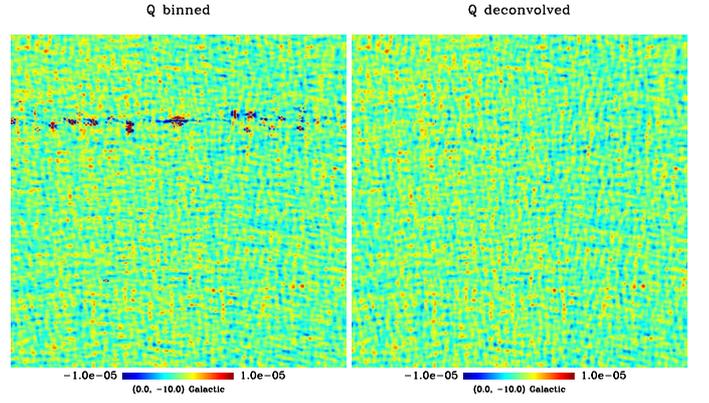
**Fig. 11.** 30GHz TT, EE, and TE spectra with white noise included. The TE spectra have been binned over 20 multipoles. The deconvolution results are shown for three combinations of parameters:  $k_{\text{max}}=4$  and  $l_{\text{max}}=600,700$  (purple), and  $k_{\text{max}}=6, l_{\text{max}}=6$  (red). The spectrum of a naive binned map is shown in green. The blue line shows the spectrum of the binned map corrected for a symmetric beam. The corresponding noise-free result is shown with dashed linetype in the TT plot, to indicate the region where noise begins to dominate over signal.



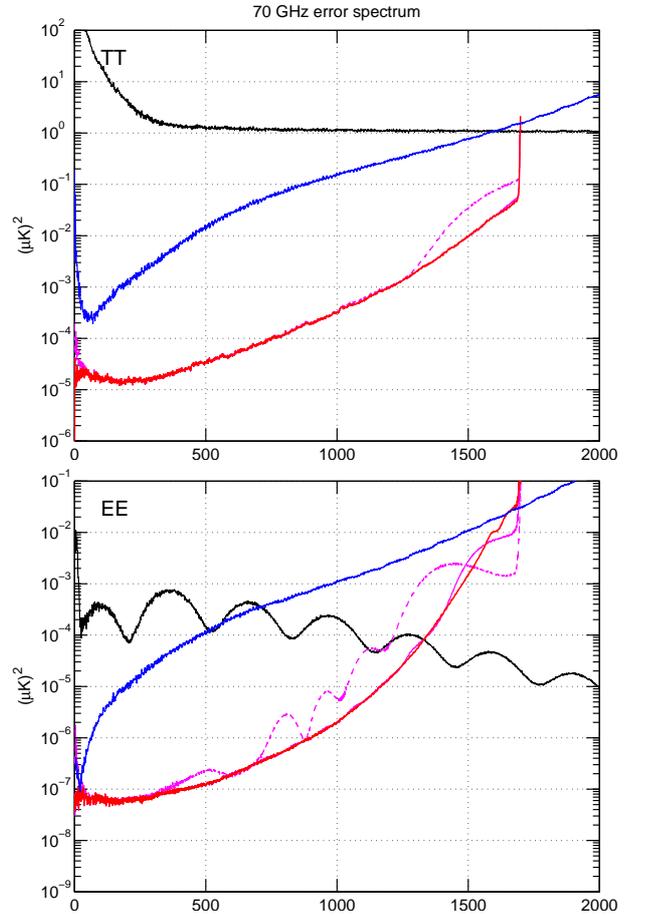
**Fig. 12.** 30GHz error spectra in presence of white noise. The purple and red lines correspond to the same combinations of deconvolution parameters as in 11. The blue line shows the error spectrum for the harmonic expansion of a binned map, corrected for a symmetric beam with FWHM=32'.



**Fig. 13.** 70GHz temperature map without noise: binned (left) and deconvolved (right). The deconvolved map was smoothed to resolution FWHM=13', corresponding to the mean width of the actual beam. Shown is the same patch of sky as in Fig. 4.



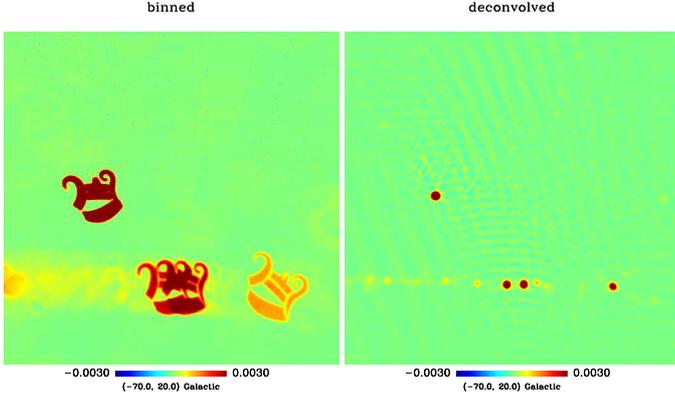
**Fig. 14.** 70GHz Q polarisation map without noise: binned (left) and deconvolved (right). Both maps were smoothed to final resolution FWHM=18'. Shown is the same patch of sky as in Fig. 6. Deconvolution removes the spurious galactic signal.



**Fig. 15.** 70GHz TT (top) and EE (bottom) error spectra, without noise. The blue line represents harmonic expansion of the binned map, corrected for a symmetric beam with FWHM=13'. The fully converged deconvolution result ( $l_{\max}=1700$ ,  $k_{\max}=6$ ), is shown by a red line. We also show the error after 200 (purple dashed) and 500 (purple solid) iteration steps.



**Fig. 16.** Unconventional beam pattern



**Fig. 17.** Sky seen through a D-beam: binned (*left*) and deconvolved (*right*) temperature map. The beam transforms the image of a point source into an image of the beam. Deconvolution recovers the original shapes of the sources. The size of the shown patch is  $83.3^\circ$  squared. The horizontal structure near the bottom of the plots is the accumulation of weak point sources in the galactic plane.

We show the TT and EE spectra after 200 and 500 iteration steps in Fig. 15, along with the final results. We observe that the solution changes only very little after the first few hundred iterations, and 200 steps already give a significant improvement over the harmonic expansion of the binned map. In fact, after the first 100 iteration steps we can see no visible improvement in the reconstructed sky map. We could thus reduce the computation time by a large factor if we used a more relaxed convergence criterion and would not significantly lose accuracy.

### 5.6. Complicated beam shape

To demonstrate the power of the deconvolution method, we made yet another simulation with an unconventional D-shaped beam (D for “deconvolution”), which is shown in Fig. 16. The linear dimension of the beam was approximately  $12^\circ$  squared. For demonstration purposes we included only point sources and considered temperature only. For the binning process into the 3D map, we chose  $N_{\text{side}} = 1024$  and  $n_\psi = 256$ . Deconvolution parameters were  $l_{\text{max}} = 500$ ,  $k_{\text{max}} = 30$ , and only a single full-sky survey was simulated. The comparatively high value of  $k_{\text{max}}$  was required to properly take into account the complicated azimuthal structure of the beam.

A map binned directly from the TOD is shown in the left panel of Fig. 17. The beam transforms the image of a point source into a D-shaped pattern, resembling the beam. The deconvolved map is shown on the right; it was smoothed by a symmetric Gaussian beam with  $\text{FWHM}=1^\circ$  to reduce ringing. Even

with this complicated beam shape we are able to recover the underlying point sources.

It is interesting that in this case it was sufficient to smooth the map with a beam of  $\text{FWHM}=1^\circ$  to obtain a very satisfactory map, although the full dimension of the beam is much larger.

## 6. Conclusions

We have presented an efficient beam deconvolution code, designed for absolute CMB experiments, and tested it on simulated data. The code is released under the terms of the GNU General Public License and can be obtained from <http://sourceforge.net/projects/art-deco/>.

We looked at maps constructed from the recovered  $a_{Tlm}$ ,  $a_{Elm}$ , and  $a_{Blm}$  coefficients, and examined the coefficients directly in harmonic space. These reflect different aspects of the solution. We compared the results to a harmonic expansion of a binned map. We have also shown that with sufficient computational resources, we can extend the method to  $l_{\text{max}}=1700$ , which would be sufficient for the Planck 70GHz channel.

In absence of noise we could recover the  $a_{Tlm}$  coefficients to a high accuracy, and remove the effects of beam asymmetry. When white noise was added, we were able to reach a lower value of  $l_{\text{max}}$ , but the advantage of deconvolution over binning was still clear.

In the case of polarisation, deconvolution worked well in absence of noise. We were able to almost completely remove the temperature leakage due to beam mismatch. When noise was added, results were less clear. Deconvolution removed the visible galactic residual that arises from beam mismatch, but in harmonic space deconvolution did not seem to bring a clear benefit over binning. The recovered  $a_{Elm}$  coefficients were dominated by noise at nearly all multipoles, but this was true for both deconvolved and binned maps.

Our simulation used beams that were highly asymmetrical and also had a strong mismatch between them, which leads to a significant polarisation leakage. We did this to show the beam effects more clearly, but at the same time we also made the deconvolution problem very challenging. The accuracy we can expect when the code is applied to a real experiment strongly depends on the beam shapes of the particular experiment.

Some topics for future study can be mentioned. We have set a very strict convergence criterion for the CG iteration. Our convergence study indicates that a much more relaxed criterion could be sufficient for practical purposes. A future improvement would be to define a more suitable convergence criterion. This could reduce the computation time by a significant factor.

We applied the method to a simulation data set that provides full sky coverage. Our code is built on a formalism that makes no assumptions on the sky coverage. In practice, good convergence requires nearly full coverage. This is intuitively evident, since we are solving the harmonic coefficients, which necessarily represent the complete celestial sphere. If the input data only cover a part of the sky, the coefficients cannot be well constrained. A straightforward extension of the method would be to insert a prior that constrains the solution in the region which is not covered by the data, but this is beyond the scope of this paper and is a subject for future study.

Yet another topic for further study is determining the noise bias present in the TT and EE spectra at high multipoles.

Though the simulations used in this work mimic some aspects of the Planck mission, the parameters used do not reflect the properties of the actual instrument. The results presented in this paper are thus not representative of the sensitivity of the

Planck experiment, and the authors do not represent the Planck collaboration in this context.

## Appendix A: Definitions

In this appendix we present definitions and conventions used in the paper. We follow the conventions of Varshalovich et al. (1988).

### A.1. Harmonic expansion of the CMB radiation field

The CMB temperature and polarisation fields can be presented by spin-weighted harmonic functions with spin  $s = 0, \pm 2$ . Spin  $s = 0$  represents the temperature field, while components  $s = \pm 2$  represent polarisation.

The temperature field on the celestial sphere is expanded as

$$T(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^l a_{0lm} Y_{lm}(\theta, \phi). \quad (\text{A.1})$$

The Q and U Stokes fields, which describe linear polarisation, can be written as

$$(Q \pm iU)(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^l a_{\pm 2lm} Y_{lm}. \quad (\text{A.2})$$

Here  ${}_s Y_{lm}$  are spin-weighted harmonic functions, and  $a_{slm}$  are the corresponding (complex) coefficients. The harmonic functions and coefficients obey the symmetry relation

$$\begin{aligned} {}_s Y_{lm}(\theta, \phi) &= (-1)^{m-s} Y_{l,-m}^*(\theta, \phi) \\ a_{slm} &= (-1)^{m-s} a_{-s, l, -m}^* \end{aligned} \quad (\text{A.3})$$

The harmonic functions are related to the *Wigner functions*  $D_{sm}^l(\theta, \phi, \psi)$  through

$${}_s Y_{lm}(\theta, \phi) = F_l D_{m, -s}^{l*}(\phi, \theta, 0), \quad (\text{A.4})$$

where

$$F_l = \left( \frac{2l+1}{4\pi} \right)^{1/2}. \quad (\text{A.5})$$

It is convenient to define further

$$\begin{aligned} a_{Tlm} &= a_{0lm} \\ a_{Elm} &= -\frac{1}{2}(a_{2lm} + a_{-2lm}) \\ a_{Blm} &= -\frac{1}{2i}(a_{2lm} - a_{-2lm}). \end{aligned} \quad (\text{A.6})$$

The CMB radiation field is expected to be a statistically isotropic Gaussian spin 0,2 field. The statistical properties of the harmonic expansion coefficients are then fully characterised by four spectra  $C_l^T, C_l^E, C_l^B, C_l^C$ , such that

$$\begin{aligned} \langle a_{Tlm}^* a_{Tl'm'} \rangle &= C_l^T \delta_{ll'} \delta_{mm'} \\ \langle a_{Elm}^* a_{El'm'} \rangle &= C_l^E \delta_{ll'} \delta_{mm'} \\ \langle a_{Blm}^* a_{Bl'm'} \rangle &= C_l^B \delta_{ll'} \delta_{mm'} \\ \langle a_{Tlm}^* a_{El'm'} \rangle &= C_l^C \delta_{ll'} \delta_{mm'}. \end{aligned} \quad (\text{A.7})$$

### A.2. Wigner functions and rotations on a sphere

The harmonic coefficients transform under rotations as follows.

Assume we are given the coefficients  $a_{slm}$  in one coordinate system. We then rotate the coordinate system, as described by Euler angles  $\phi, \theta, \psi$ :

1. Rotate by angle  $\psi$  around the  $z$ -axis (in positive direction).
2. Rotate by angle  $\theta$  around the *initial*  $y$ -axis.
3. Rotate by angle  $\phi$  around the *initial*  $z$ -axis.

The rotation is equivalent to rotating first by angle  $\phi$  around the  $z$ -axis, then by angle  $\theta$  around the *new*  $y$ -axis, and finally by angle  $\psi$  around the *new*  $z$ -axis. The harmonic coefficients in the rotated coordinate system are given by

$$a'_{slm} = \sum_{m'} a_{slm'} D_{m'm}^{l*}(\phi, \theta, \psi), \quad (\text{A.8})$$

where  $D_{m'm}^l$  are Wigner functions.

A Wigner function may be split into a product of three terms, each of which only depends on one of the angles:

$$D_{mk}^l(\theta, \phi, \psi) = e^{-im\phi} d_{mk}^l(\theta) e^{-ik\psi}. \quad (\text{A.9})$$

Functions  $d_{mk}^l(\theta)$  are the *reduced Wigner functions*. They have the property

$$d_{mk}^l(0) = \delta_{mk}. \quad (\text{A.10})$$

They are real-valued and obey the symmetry relation

$$d_{mk}^l(\theta) = (-1)^{m+k} d_{-m, -k}^l(\theta). \quad (\text{A.11})$$

The symmetry relation for the full Wigner functions is

$$D_{mk}^l(\phi, \theta, \psi) = (-1)^{m+k} D_{-m, -k}^{l*}(\phi, \theta, \psi). \quad (\text{A.12})$$

### A.3. Wigner expansion

Wigner functions  $D_{mk}^l(\phi, \theta, \psi)$  for integer values of  $l, m, k$  are defined in domain  $V$ :

$$V: \quad 0 \leq \phi \leq 2\pi, \quad 0 \leq \theta \leq \pi, \quad 0 \leq \psi \leq 2\pi. \quad (\text{A.13})$$

Domain  $V$  can be interpreted as the space of Euler angles. The volume element in domain  $V$  is  $dV = \sin \theta d\theta d\phi d\psi$ , and  $V$ 's total volume is  $8\pi^2$ .

Any finite function  $f(\phi, \theta, \psi)$  defined in domain  $V$  may be expanded in a series of Wigner functions as

$$f(\phi, \theta, \psi) = \sum_{l=0}^{\infty} \sum_{m=-l}^l \sum_{k=-l}^l c_{mk}^l D_{mk}^l(\phi, \theta, \psi), \quad (\text{A.14})$$

where the coefficients  $c_{mk}^l$  are given by

$$c_{mk}^l = \frac{2l+1}{8\pi^2} \int_0^{2\pi} d\phi \int_0^{\pi} \sin \theta d\theta \int_0^{2\pi} d\psi f(\phi, \theta, \psi) D_{mk}^{l*}(\phi, \theta, \psi). \quad (\text{A.15})$$

Expansion (A.14) can be understood as the three-dimensional equivalent of the spherical harmonics expansion.

In particular, a product of two Wigner functions  $D_{mk}^{l*}(\omega) D_{m'k'}^{l'}(\omega)$  can be expanded as a series of Wigner functions as

$$\begin{aligned} & D_{mk}^{l*}(\omega) D_{m'k'}^{l'}(\omega) \\ &= \sum_{l_2 m_2 k_2} D_{m_2 k_2}^{l_2}(\omega) \frac{2l_2+1}{8\pi^2} \int D_{mk}^{l*}(\omega') D_{m'k'}^{l'}(\omega') D_{m_2 k_2}^{l_2*}(\omega') d\omega'. \end{aligned} \quad (\text{A.16})$$

An integral over a product of three Wigner functions is given by Wigner  $3j$  symbols as

$$\int D_{mk}^{l*}(\omega') D_{m'k'}^{l'}(\omega') D_{m_2k_2}^{l_2*}(\omega') d\omega' \quad (\text{A.17})$$

$$= 8\pi^2 (-1)^{m'+k'} \begin{pmatrix} l & l' & l_2 \\ m & -m' & m_2 \end{pmatrix} \begin{pmatrix} l & l' & l_2 \\ k & -k' & k_2 \end{pmatrix}$$

(Varshalovich et al. 1988).

The following properties of the  $3j$  symbols help in restricting the range of summation. A  $3j$  symbol is non-vanishing

$$\begin{pmatrix} l_1 & l_2 & l_3 \\ k_1 & k_2 & k_3 \end{pmatrix} \neq 0 \quad (\text{A.18})$$

only if the two conditions

$$k_1 + k_2 + k_3 = 0 \quad (\text{A.19})$$

and

$$|l_1 - l_2| \leq l_3 \leq l_1 + l_2 \quad (\text{A.20})$$

are fulfilled. We can thus eliminate the summation over  $k_2$  and  $m_2$ . Product (A.16) becomes

$$D_{mk}^{l*}(\omega) D_{m'k'}^{l'}(\omega) = (-1)^{m'+k'} \sum_{l_2} (2l_2 + 1) D_{m'-m, k'-k}^{l_2}(\omega) \quad (\text{A.21})$$

$$\times \begin{pmatrix} l & l' & l_2 \\ m & -m' & (m'-m) \end{pmatrix} \begin{pmatrix} l & l' & l_2 \\ k & -k' & (k'-k) \end{pmatrix}.$$

## A.4. Beam

### A.4.1. Detector signal and definition of beam coefficients

Consider a detector measuring CMB temperature and linear polarisation.

We assume now, very generally, that the signal recorded by a detector, apart from noise, is proportional to the  $a_{slm}$  coefficients. We define the beam coefficients  $b_{slk}$  as follows. When the detector is at some (predefined) fiducial position and orientation (for instance at the north pole), the signal detected in the absence of noise is

$$t_{\text{fid}} = \sum a_{slm} b_{slm}^* \quad (\text{A.22})$$

It is clear from the definition that the beam coefficients must contain all information not only of the beam shape, but also of the possible polarisation sensitivity and orientation.

We may then rotate the beam to another position and orientation, as described in Section A.2. The rotation brings the beam onto location  $\theta, \phi$  in spherical coordinates, in an orientation defined by angle  $\psi$ . The harmonic coefficients transform under rotations according to (A.8). The signal seen by the rotated detector is then

$$t = \sum_{slmk} a_{slm} b_{slk}^* D_{mk}^{l*}(\phi, \theta, \psi). \quad (\text{A.23})$$

In the following we derive expressions for the beam coefficients in some special cases.

### A.4.2. Non-polarised beam

Consider first a detector with an ideal beam, beam width equal to zero, and no sensitivity to polarisation. The signal detected by the beam is directly given by (A.1). We see readily, by comparing (A.1) and (A.22), that if the chosen fiducial beam position is at the north pole ( $\theta = 0, \phi = 0$ ), the beam coefficients must be

$$b_{slm} = {}_s Y_{lm}^*(0, 0) \delta_{s0} = F_l \delta_{s0} \delta_{m0}. \quad (\text{A.24})$$

A general non-polarised beam with sensitivity  $g(\theta, \phi)$  gives for the  $s = 0$  components

$$b_{0lm} = \iint g(\theta, \phi) {}_0 Y_{lm}^*(\theta, \phi) \sin \theta d\theta d\phi \quad (\text{A.25})$$

$$= F_l \iint g(\theta, \phi) D_{m0}^l(\phi, \theta, 0) \sin \theta d\theta d\phi$$

and  $b_{slm} = 0$  for  $s = \pm 2$ .

### A.4.3. Polarisation measurements

A polarisation-sensitive detector requires careful treatment, because the Stokes parameters Q and U are not defined at the pole. The beam coefficients can still be defined in a meaningful way, as we show in the following.

Consider again a beam at a fiducial position and orientation at the north pole. A general beam may have different polarisation sensitivity at different locations. We therefore define the beam shape  $g(\theta, \phi)$  separately for temperature and polarisation.

The signal detected may be written in a general form as

$$t = \iint \sin \theta d\theta d\phi \left\{ g_T(\theta, \phi) T(\theta, \phi) + g_P(\theta, \phi) [Q(\theta, \phi) \cos(2\psi(\theta, \phi)) + U(\theta, \phi) \sin(2\psi(\theta, \phi))] \right\}. \quad (\text{A.26})$$

Here  $\psi(\theta, \phi)$  is the *local* direction of polarisation sensitivity, measured as an angle from the local meridian. Functions  $g_T(\theta, \phi)$ ,  $g_P(\theta, \phi)$ , and  $\psi(\theta, \phi)$  together define a general beam.

We proceed to calculate the beam coefficients. We write the trigonometric functions in exponential form, to obtain

$$t = \iint \sin \theta d\theta d\phi \left\{ g_T(\theta, \phi) T(\theta, \phi) + \frac{g_P(\theta, \phi)}{2} [(Q - iU)(\theta, \phi) e^{i2\psi(\theta, \phi)} + (Q + iU)(\theta, \phi) e^{-i2\psi(\theta, \phi)}] \right\}. \quad (\text{A.27})$$

We can now insert the harmonic expansions (A.1) and (A.2) into (A.27) and read the beam coefficients

$$b_{0lk} = \iint g_T(\theta, \phi) {}_0 Y_{lk}^*(\theta, \phi) \sin \theta d\theta d\phi$$

$$b_{\pm 2lk} = \frac{1}{2} \iint g_P(\theta, \phi) {}_{\pm 2} Y_{lk}^*(\theta, \phi) e^{\pm i2\psi(\theta, \phi)} \sin \theta d\theta d\phi. \quad (\text{A.28})$$

We then use definition (A.4) and write the harmonic functions in terms of Wigner functions. In terms of the reduced Wigner functions we have

$${}_s Y_{lk}^*(\theta, \phi) = F_l e^{-ik\phi} d_{k,-s}^l(\theta). \quad (\text{A.29})$$

The  $\psi$ -term can be transferred inside the Wigner function. We may thus write for a general polarised beam

$$b_{0lk} = F_l \iint g_T(\theta, \phi) D_{k,0}^l(\phi, \theta, 0) \sin \theta d\theta d\phi$$

$$b_{\pm 2lk} = \frac{F_l}{2} \iint g_P(\theta, \phi) D_{k,\mp 2}^l(\phi, \theta, \psi(\theta, \phi)) \sin \theta d\theta d\phi. \quad (\text{A.30})$$

The above formulas for the beam coefficients are completely general as long as the beam response is linear.

We now make the simplifying assumption that the polarisation sensitivity is “in the same direction” everywhere on the beam. By “same direction” we mean that the direction of polarisation sensitivity at an arbitrary location is obtained by rotating the polarisation direction vector at the north pole to the desired location along a meridian. The local polarisation orientation angle then becomes

$$\psi(\theta, \phi) = \psi_{\text{pol}} - \phi, \quad (\text{A.31})$$

where  $\psi_{\text{pol}}$  is the deviation of polarisation direction from  $\phi = 0$  at the north pole.

At this point it is convenient to write the Wigner functions in terms of the reduced Wigner functions (A.9). The coefficients for a beam with unique polarisation direction become

$$\begin{aligned} b_{0lk} &= F_l \int \sin \theta d\theta d_{k,0}^l(\theta) \int g_T(\theta, \phi) e^{-ik\phi} d\phi \\ b_{\pm 2lk} &= \frac{F_l}{2} e^{\pm i2\psi_{\text{pol}}} \int \sin \theta d\theta d_{k,\mp 2}^l(\theta) \int g_P(\theta, \phi) e^{-i(k\pm 2)\phi} d\phi. \end{aligned} \quad (\text{A.32})$$

We readily see that for a symmetric beam the only non-vanishing coefficients are those with  $k + s = 0$ . Especially, for a perfect beam with zero width and perfect polarisation sensitivity ( $g_P = g_T$ ) we obtain

$$\begin{aligned} b_{0lk} &= F_l d_{k,0}^l(0) = F_l \delta_{k,0} \\ b_{\pm 2lk} &= \frac{F_l}{2} d_{k,\mp 2}^l(0) e^{\pm i2\psi_{\text{pol}}} = \frac{F_l}{2} e^{\pm i2\psi_{\text{pol}}} \delta_{k,\mp 2}. \end{aligned} \quad (\text{A.33})$$

We have assumed normalisation  $\int g_T(\Omega) d\Omega = 1$ .

In this work we have constructed the beam coefficients according to (A.32), but the assumption of unique polarization direction is not crucial for the deconvolution method. A more general beam presentation is given by (A.30).

## Appendix B: Methods used to accelerate computation

### B.1. Load balancing

In an MPI-parallel code, good load balancing among the individual tasks is crucial. As was mentioned in Section 3.2, some quantities are distributed across tasks depending on their colatitude  $\theta$  or the index  $m$ . Since the computational cost of the algorithm is not independent of  $\theta$  and  $m$ , but rather an unknown, smoothly varying function of these two numbers, it is advisable not to assign sequential ranges of those indices to individual MPI tasks, because choosing index ranges that distribute the load evenly is very hard in that scenario.

To avoid this complication, we adopted a “round robin” strategy for data distribution. Assuming  $n$  MPI tasks numbered 0 to  $n - 1$ , we assign to a particular task  $i$  the  $\theta$  indices  $\theta_i, \theta_{i+n}, \theta_{i+2n}, \dots$ , and deal with the index  $m$  in analogous fashion. This results in near-perfect load balancing.

### B.2. Convolution optimisation

A significant part of total CPU time is spent in the FFTs necessary for the convolution with the quantity  $N$  (see section 3.2). Fortunately, several measures can be taken to decrease the resource requirements. First of all, we are making use of the highly

optimised FFTW<sup>4</sup> library (Frigo & Johnson 1998), which provides close to optimal performance for FFTs of specific size.

Furthermore, it is important to notice that the execution time for an FFT of length  $n$  depends sensitively on the prime factorisation of  $n$ ; in general, an  $n$  composed of only small prime factors is much preferable.

Since we are performing a convolution operation, we have some freedom in the choice of array dimensions; they can be increased and zero-padded if desired. In our case, the minimal array dimensions are  $4l_{\text{max}} + 1$  and  $4k_{\text{max}} + 1$ , respectively; if necessary, we increase both of them to the next larger number, which is a product of the prime factors 2, 3, and 5 only. Depending on the prime factorisation of the original array dimensions, this can decrease the run-time for this part of the code by integral factors.

Finally, the quantities that are to be convolved exhibit the symmetry

$$V_{i,k} = V_{n_i-i, n_k-k}^* \quad \text{for } i, k > 0,$$

which allows us to employ the specialised “real-valued” FFT, resulting in another substantial speedup and a size reduction of the precomputed Fourier transform of  $N$  (see Section B.5).

### B.3. Efficient computation of the reduced Wigner functions

During code execution, the reduced Wigner functions  $d_{mk}^l(\theta)$  are required twice in every conjugate gradient iteration step. Storing them in memory is prohibitively expensive, so it is important to regenerate them efficiently whenever needed. For this purpose we used code originally presented by Prézeau & Reinecke (2010), which was extended to make use of the SSE2 instruction set present in all modern Intel-compatible CPUs. We also used the symmetry properties of the  $d$  matrix to avoid redundant computations.

### B.4. Shortcuts for symmetrical beams

If a beam used for deconvolution is symmetric with respect to a 180° rotation around its axis (which is true for elliptical beams, for example), all its  $b_{slm}$  coefficients for odd  $m$  vanish by definition. This fact is used to skip unnecessary computations as well as save space by not storing any array elements that are known to be zero. It also reduces the minimum second dimension of the convolution arrays from  $4k_{\text{max}} + 1$  to  $2k_{\text{max}} + 1$ , saving even more time on the FFT operations and reducing memory consumption.

### B.5. Optional precomputation

The convolution step mentioned in section 3.2 requires the Fourier transform of array  $N$ , which is constant throughout a run. The code permits one to precompute and store this quantity, which reduces total CPU time (because only two FFT operations are then needed for each convolution step, instead of three). However, especially for runs with high  $l_{\text{max}}$  and  $k_{\text{max}}$  this increases the total memory consumption considerably. So for situations where the available main memory is insufficient to store this precomputed array, we provide the option of recomputing the Fourier-transformed  $N$  whenever necessary, approximately halving the memory consumption of the code, but also slowing it down by 5-15 percent. For the 70GHz runs described in this paper, we had to make use of this space-saving feature.

<sup>4</sup> <http://fftw.org>

**Table C.1.** Resource usage for various deconvolution runs

Name	$l_{\max}$	$k_{\max}$	$T_{\text{wall}}$	$T_{\text{setup}}$	$T_{\text{iter}}$	$N_{\text{iter}}$	Mem
30a	400	6	10:07	1:13	3.9	140	6.6
30b	500	6	25:49	1:20	5.5	270	8.1
30c	600	4	1:24:57	1:07	5.1	995	6.4
30d	600	6	1:08:53	1:35	6.9	591	9.6
30e	600	8	1:40:40	2:24	10.3	573	11.9
30f	700	6	3:41:13	1:48	8.5	1567	11.3
30g	800	6	5:10:41	1:56	10.6	1759	12.8
D	500	30	10:31:38	8:34	20.3	1842	17.6
70a	1700	6	63:51:56	15:27	78.3	2923	28.8
70x	1700	6	9:43:05	5:58	11.8	2955	—
70y	1700	6	5:25:32	7:20	6.6	2922	—

**Notes.**  $T_{\text{wall}}$  denotes the total wall clock time of the run,  $T_{\text{setup}}$  is the time required for the code startup (i.e. data input, computing the RHS, Wigner transform of the hit count and preconditioner),  $T_{\text{iter}}$  gives the average wall clock time for a single CG step in seconds,  $N_{\text{iter}}$  is the required number of iterations until convergence, and “Mem” is the total memory usage in GB. Please note the additional comments in the text.

### B.6. Supplying an initial guess

While not strictly a performance optimisation, the code allows the user to supply an initial guess from which to start the CG iteration, which is then used instead of a vector of zeroes. This is particularly helpful if one tries to compute series of deconvolutions on the same input data set, but with different parameters. For example, if a solution for a deconvolution problem with a given  $l_{\max}$  and  $k_{\max}$  has already been obtained, it can be used as a good starting point for other calculations with higher  $l_{\max}$  and/or  $k_{\max}$ . In any case, the convergence criterion will still be evaluated with respect to an all-zero  $a_{slm}$ , not the supplied guess.

## Appendix C: Resource usage

Most computations discussed in this paper were performed on a computer with 64GB of RAM and 24 Intel Xeon E7450 cores. Since this computer is used interactively, we only ran our calculations on 16 of these cores and ensured that interactive usage did not exceed the capability of the remaining 8 cores. Nevertheless, the measured timings contain small uncertainties and should be interpreted as upper limits.

Table C.1 gives an overview of the parameters of the various runs and their impact on resource usage. Runs 30a-g are the 30GHz calculations discussed in Sect. 5.2; runs 70a, 70x, and 70y are related to the 70GHz high-resolution study (Sect. 5.4), and run “D” is the deconvolution with the D-shaped beam presented in Sect. 5.6. For all runs except the D-beam, the beam was assumed to be symmetrical with respect to a 180° rotation around its axis. Also, for all runs except the 70GHz ones the FFT of array  $N$  was cached (see Section B.5).

For the 70GHz run on the test machine described above (run 70a), each CG iteration step needed significantly more time than for the calculations using the 30GHz data set; in combination with the high number of iterations, this leads to an inconveniently long computation time. As a consequence, deconvolution runs of this magnitude are better suited for execution on massively parallel computers. We therefore repeated the computation on the Louhi supercomputer of CSC (Cray XT4/XT5)

on 128 and 512 processor cores, respectively. The corresponding performance figures are listed as runs 70x and 70y in Table C.1; memory consumption was not measured for these runs.

The computational power of a single CPU core is roughly the same on both computers used for our tests. Looking at the total wall clock time for the 70GHz runs, it becomes obvious that the scaling from 16 to 128 cores is close to ideal, but becomes much worse when going from 128 to 512 cores. In this range, MPI communication and redundant computations begin to dominate. For deconvolution problems with larger  $l_{\max}$ , we expect that this transition will occur at a higher number of MPI tasks.

The numbers of iterations required for the 70GHz runs are not identical, as one might intuitively expect. This is because in an MPI-parallel code some computation steps – most notably summations and other reductions of quantities residing on all tasks – are not carried out in a strictly defined order, but depend on the relative timing of the tasks during such an operation, which is nondeterministic. Since floating-point algebra with finite precision is non-associative, the results of such operations can differ slightly from run to run, causing the conjugate gradient solver to take different paths towards the solution, but of course ultimately solving the same numerical problem.

Compared to earlier publications on deconvolution map-making, it is evident that both memory and CPU requirements of the presented algorithm are quite modest and allow deconvolution up to values of  $l_{\max}$  that were prohibitively expensive up to now. Making use of massively parallel computing clusters, we expect that the algorithm can be applied to data sets with even higher resolution than those presented here.

*Acknowledgements.* Some of the results in this paper have been derived using the HEALPix (Górski et al. 2005) package. We acknowledge the use of the Planck Sky Model, developed by the Component Separation Working Group (WG2) of the Planck Collaboration. MR is supported by the German Aeronautics Center and Space Agency (DLR), under program 50-OP-0901, funded by the Federal Ministry of Economics and Technology. EK is supported by Academy of Finland grant 253204. This work was granted access to the HPC resources of CSC made available within the Distributed European Computing Initiative by the PRACE-2IP, receiving funding from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement RI-283493. We thank CSC – the IT Center for Science Ltd. (Finland) – for computational resources. We thank Torsten Enßlin for constructive discussions and feedback, Valter Lindholm for help in using the Planck Sky Model, and Ronnie James Dio for the D-beam.

## References

- Armitage, C. & Wandelt, B. D. 2004, *Phys. Rev. D*, 70, 123007
- Armitage-Caplan, C. & Wandelt, B. D. 2009, *ApJS*, 181, 533
- Ashdown, M. A. J., Baccigalupi, C., Balbi, A., et al. 2007a, *A&A*, 471, 361
- Ashdown, M. A. J., Baccigalupi, C., Balbi, A., et al. 2007b, *A&A*, 467, 761
- Delabrouille, J., Betoule, M., Melin, J.-B., et al. 2012, *ArXiv e-prints*
- Frigo, M. & Johnson, S. G. 1998, in *Proc. 1998 IEEE Intl. Conf. Acoustics Speech and Signal Processing*, Vol. 3 (IEEE), 1381–1384
- Górski, K. M., Hivon, E., Banday, A. J., et al. 2005, *ApJ*, 622, 759
- Harrison, D. L., van Leeuwen, F., & Ashdown, M. A. J. 2011, *A&A*, 532, A55
- Jarosik, N., Bennett, C. L., Dunkley, J., et al. 2011, *ApJS*, 192, 14
- Keihänen, E., Keskitalo, R., Kurki-Suonio, H., Poutanen, T., & Sirviö, A.-S. 2010, *A&A*, 510, A57
- Keihänen, E., Kurki-Suonio, H., & Poutanen, T. 2005, *MNRAS*, 360, 390
- Kurki-Suonio, H., Keihänen, E., Keskitalo, R., et al. 2009, *A&A*, 506, 1511
- Poutanen, T., de Gasperis, G., Hivon, E., et al. 2006, *A&A*, 449, 1311
- Prézeau, G. & Reinecke, M. 2010, *ApJS*, 190, 267
- Reinecke, M., Dolag, K., Hell, R., Bartelmann, M., & Enßlin, T. A. 2006, *A&A*, 445, 373
- Schulten, K. & Gordon, R. G. 1975, *Journal of Mathematical Physics*, 16, 1961
- Sutton, D., Johnson, B. R., Brown, M. L., et al. 2009, *MNRAS*, 393, 894
- Tauber, J. A., Mandolesi, N., Puget, J.-L., et al. 2010, *A&A*, 520, A1
- Varshalovich, D. A., Moskalev, A. N., & Khersonskii, V. K. 1988, *Quantum Theory of Angular Momentum* (World Scientific Publishers)

Wiener, N. 1949, Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications, Technology press books in science and engineering (Technology Press of the Massachusetts Institute of Technology)