

# Mirage: A new iterative map-making code for CMB experiments

D. Yvon<sup>1,2</sup> and F. Mayet<sup>1,3</sup>

<sup>1</sup> CEA-CE Saclay, DAPNIA, Service de Physique des Particules, Bât. 141, 91191 Gif-sur-Yvette Cedex, France  
e-mail: yvon@hep.saclay.cea.fr

<sup>2</sup> Fédération de Recherche APC, Université Paris 7, Paris, France

<sup>3</sup> LPSC Grenoble, CNRS/IN2P3 and Université J. Fourier, 53 avenue des Martyrs, 38026 Grenoble Cedex, France

Received 19 December 2003 / Accepted 10 January 2005

**Abstract.** A major goal of CMB experiments is to obtain highly sensitive CMB maps in order to extract Spherical Harmonic Power Spectrum (SHPS) and cosmological parameters with unprecedented accuracy. We present a new map-making code (Mirage) based on a two-phase iterative algorithm, involving low frequency drift treatment, Butterworth high-pass filtering, and conjugate gradient method. This work was strongly motivated by Archeops CMB experiment data analysis, which we believe is a good test bench for future Planck Surveyor data analysis. Mirage was designed to be used for Planck data processing with a minimum amount of effort. A strong feature of Mirage is that it handles experimental problems in data, such as holes in the data stream, bright sources, and galaxy side effects, without jeopardising speed. The other advantage is its processing speed that allows Monte Carlo simulations of Archeops data processing to be run overnight on a single processor workstation. Algorithms are explained and systematic effects on SHPS investigated on various simulated data, including typical Archeops observational systematics. The code is now available at [http://www-dapnia.cea.fr/Telechargement/tel\\_mirage.php](http://www-dapnia.cea.fr/Telechargement/tel_mirage.php)

**Key words.** cosmic microwave background – cosmology: observations – submillimeter

## 1. Introduction

Current and forthcoming CMB anisotropy experiments, both space borne (Planck<sup>1</sup>, MAP<sup>2</sup>) and balloon borne, (e.g. Archeops<sup>3</sup> Benoît et al. 2003a,b,c), should allow for a major breakthrough in understanding the early Universe, in particular by estimating cosmological parameters with unprecedented accuracy.

Amongst the various steps of the CMB data analysis pipeline, from raw data treatment to angular power spectrum and eventually cosmological parameter extraction, the map-making phase is crucial enough to be handled with great care, in order not to induce systematics in the map, such as low frequency striping or galaxy side effects. Most map-making methods rely on the assumption that instrument noise is Gaussian to assure that the “optimal” map is a maximum likelihood estimate<sup>4</sup> and stationary, or at least piecewise stationary.

The basic formalism of map-making algorithms has already been described in great detail (e.g. Tegmark 1997; Hamilton 2003), but we recall the basic elements for convenience. Let  $t$  be the true sky convolved by the instrumental lobe, assumed to be symmetric,  $n$  the instrumental noise,  $A$  the pointing matrix, and finally  $d$  the observed data stream. Sky

patterns are stored in  $N_{\text{pix}}$  pixels. Assuming that the Time-Ordered Data (hereafter TOD) depends linearly on the map allows the fundamental relation to be written as

$$d = At + n. \quad (1)$$

As  $d$  and  $n$  involve  $N_{\text{sp}}$  samples, pointing matrix  $A$  is an  $N_{\text{sp}} \times N_{\text{pix}}$  matrix. The map-making method, if linear, can be expressed (Eq. (1)) as:  $s = Wd$ , where  $s$  is the reconstructed sky map and  $W$  the matrix describing the linear method. When computing the *error map*, one gets:  $s - t = [WA - I]t + Wn$  with  $I$  the identity matrix. It can be underlined that the property  $WA = I$  is highly desirable, because if true, the method is unbiased and the error map does not depend on the sky signal. This allows given  $W$  and a reliable noise model to statistically *subtract* known noise contributions in the analysis. The simplest map-making method is *coaddition*

$$W_{\text{coad}} = [A^t A]^{-1} A^t.$$

Operator  $A^t$  projects the timeline onto the sky map pixels, while operator  $A^t A$  counts the sample directed toward each pixel. This method is both very simple and fast, and it satisfies the property  $W_{\text{coad}} A = I$  and has been shown to minimise error map variance for white noise. Unfortunately, in most experiments, the noise is not observed to be white. Defining  $N = \langle nn^t \rangle$  as the noise covariance matrix, it has been shown in Tegmark (1997) that the optimal method is given by

$$W = (A^t N^{-1} A)^{-1} A^t N^{-1}. \quad (2)$$

<sup>1</sup> <http://astro.estec.esa.nl/Planck>

<sup>2</sup> <http://map.gsfc.nasa.gov/>

<sup>3</sup> <http://www.archeops.org>

<sup>4</sup> It can be noticed that even if the noise is not Gaussian the optimal map has the lowest residual variance.

This method satisfies the relation  $\mathbf{WA} = \mathbf{I}$ , minimises the chi-squared noise towards the reconstructed map, and is the maximum-likelihood estimate of true sky  $s$  if the probability distribution of  $\mathbf{n}$  is Gaussian. Tegmark (1997) shows it is *lossless*, i.e. retains all CMB information initially contained in the timeline. This algorithm (MADCAP Borrill 1999) has been successfully applied to several CMB data sets, such as COBE (Bennett et al. 1996), MAXIMA (Hanany et al. 2000) or BOOMERanG (de Bernardis et al. 2000). Nevertheless, inverting full matrix scales like  $N_{\text{pix}}^3$ , where  $N_{\text{pix}}$  is the number of pixels in the map, thus limits the interest of this method to rather *small* number of pixels in maps.

In the context of current and forthcoming experiments, for which the data set is extremely large ( $N_{\text{pix}} = 10^7$ ,  $N_{\text{sp}} = 8 \times 10^8$  for Archeops and  $N_{\text{pix}} = 5 \times 10^7$ ,  $N_{\text{sp}} = 10^9$  for Planck), several map-making codes have been developed and published, which involve either a direct ( $\mathbf{A}^t \mathbf{N}^{-1} \mathbf{A}$ ) inversion (MADCAP Borrill 1999), or an iterative inversion algorithm (Prunet 2000; Wright 1996; Natoli et al. 2001; Doré et al. 2001; Dupac et al. 2002). FFT, which scales only as  $N_{\text{sp}} \log N_{\text{sp}}$ , was the fundamental tool of these fast iterative map-making algorithms (Wright 1996) and is used intensively in Mirage. Recent implementation of some of the quoted algorithm improves this scaling to  $N_{\text{sp}} * \log L$  where  $L$  is the useful bandwidth of the noise, a constant factor.

The Mirage map-making presented in Sect. 3 is based on a two-phase iterative algorithm. In the first phase, a quick estimation of the sky map is evaluated, in which the low frequency drift is removed. This map is then used as input for the second phase, a conjugate gradient method.

## 2. Observational constraints

When applying a map-making method to real data, several items must be carefully taken into account. First, data streams do have gaps that should be filled properly. They may be induced by low level data treatment, such as the removal of glitches, i.e. high energy cosmic rays (mainly protons) hitting the bolometer. Instrumental noise in cryogenic experiments often presents low frequency systematics in which knee frequency may exceed the scanning frequency. As shown in the following simulations, scan-synchronous low frequency systematics project onto the map and cannot be minimised by the map-making algorithm. With ring-based scanning strategies (Archeops, Planck) annoying stripes show up on the sky maps.

These low frequency disturbances may be subtracted, if they turn out to be correlated to some other instrumental or observational data. If no useful decorrelation data is available, two options are left until a better idea shows up: either to filter the data, suppressing the noise components along with the signal at the corresponding scales in the sky map, or to make a priori assumptions about the noise properties, and subtract the best estimate of this noise component. In the case of the MIRAGE algorithm, the filtering option was chosen.

The effect of the Galaxy must then be taken into account. With the planets, it is the brightest object of the sky. When filtered, the rather large galactic signal, as compared to the

expected CMB one, induces ringing on the timelines that distort the power spectrum on various angular scales.

As a matter of fact, Monte Carlo simulation is compulsory for understanding data analysis systematic and correct for its effect. For instance, the MASTER method (Hivon et al. 2002) allows calibration of the unwanted effects on the CMB power spectrum, whether the instrumental noise, high-pass filtering, or any map-making induced alterations. This requires repetitive Monte Carlo TOD simulations, followed by map-making of these data streams. Any useful map-making algorithm must thus be fast enough to allow for massive Monte Carlo simulations, the next section presents the Mirage map-making code algorithms.

## 3. Mirage algorithms

### 3.1. General design options

We decided to develop two separated map-making algorithms, based on the same basic libraries and some specialised tools. The first algorithm, named MirageDC, computes a sky map without filtering the timeline. If the data does not display low frequency noise with knee frequency above the scanning (spinning) frequency, this is the algorithm to use. The second, named MirageAC filters the timeline using a parametrised Butterworth (Horowitz and Hill) high-pass filter and then computes the sky map. As expected, low frequency features in the timeline and in the map are filtered out.

Both algorithms work in two steps. The first is a fast estimation of the sky map in order to minimise the work left to the iterative optimisation, which is the most time-consuming step of the process. The quick estimation map is thus saved and feeds the second step of the iterative map optimisation based on a Conjugate Gradient iterator. The noise properties can be reliably extracted from the time ordered data, with no a priori. When the noise spectrum is complex with a strong low frequency component, this two step method ensures faster and smoother convergence of the algorithms. The optimised sky map is saved, then subtracted from the input timeline: the best estimate of the noise spectrum is computed from this noise timeline and saved.

Finally speed was a constant concern in writing this code. The code is not fully optimised, but all the chosen algorithms scale with the sample number  $N_{\text{sp}}$  as  $N_{\text{sp}} \log N_{\text{sp}}$  or less that are typical of Fast Fourier Transforms. These codes have been optimised for large sky coverage scanning strategies, such as Archeops and Planck, and obviously account for the strong signals induced by the Galaxy.

### 3.2. Gap filling

In order to compute a sky map, several variables are needed for each sample: the time (or sample number), the bolometer pointing Galactic latitude and longitude, and the bolometer signal, in physical unit on the sky. Most of the time the experiment also provides a quality indicator for the sample named flag. The flag is decoded according to the experiment: if the sample should not be kept, the pointing and bolometer signal are overridden

to undefined value. Such undefined samples belong mostly to relatively short sets of data, but induce sharp discontinuities in the timelines that prevent use of Fast Fourier Transforms.

To overcome this problem it is necessary to implement a gap-filling algorithm that has to be fast and has to preserve the spectrum of the timeline. Gap filling using linear prediction available in Numerical Recipes (Press et al.) provides satisfactory results, but its scales as  $N_{\text{gap}}^2$ . A simple mirroring technique has been chosen instead. Undefined bolometer signal samples are replaced by the mirror of the timeline taken at the last valid sample before the gap or the mirror of the timeline taken at the first valid sample after the gap. Sharp discontinuity (step) between the last (first) filled sample and the following (preceding) timeline may then happen. Continuity is restored by adding a linear interpolation of the step amplitude to the filled samples. We then retain the mirror option of minimum variance after subtraction of the interpolation between the last valid sample before gap and the first valid ample after gap. The algorithm computation time scales linearly with gap sample number. Additionally, using the simulated noise timeline with the spectrum in Fig. 2 with up to 20% undefined samples, the reconstructed noise spectrum after gap-filling was checked for statistical compatibility with the initial noise spectrum, thus behaving satisfactorily.

### 3.3. Conjugate gradient iterator

The Map making equation written as  $(A^t N^{-1} A) \tau = A^t N^{-1} d$ , where  $\tau$  is the vector of scanned sky pixels values to be computed, is a set of linear equations, where each observed pixel of the sky is an unknown. The Conjugate Gradient Method (CG, Barrett et al.) is an iterative numerical method used widely for solving systems of linear equations. Following Natoli et al. (2001), this method was chosen. Computation speed is improved by two methods:

1) Given an estimated sky  $s_0$  the map-making equation can be written as

$$(A^t N^{-1} A)(\tau - s_0) = A^t N^{-1}(d - A s_0).$$

When used with a reasonable initial sky guess, this variable change, suggested by Doré et al. (2001), removes most of the signal from the remaining timeline  $D = d - A s_0$  ( $A s_0$  is the timeline computed when the estimated sky  $s_0$  is *observed* through the pointing matrix. It allows the range of numbers involved in the computation to be cut thus improving numerical stability of the algorithm. It also minimises sharp transitions in the timeline data, when the line of sight crosses bright objects such as the Galaxy.  $D$  is now assumed to be dominated by noise and is used to compute the noise spectral density  $\Xi$ . Assuming the noise is stationary, Tegmark (1997) explains how to a very good approximation:

$$N^{-1} D = F^t [\Xi^{-1} F(D)]$$

where  $F$  ( $F^t$ ) is the Fast Fourier Transform (antitransform). Computation time scales as  $N_{\text{sp}} \log N_{\text{sp}}$  and the large Matrix  $N^{-1}$  do not need to be stored.

2) Convergence speed is improved by using a preconditioning matrix. A first choice of the preconditioning matrix is  $P = [\text{diag}(A^t N^{-1} A)]^{-1}$ , where the diag operator reduces  $A^t N^{-1} A$  to its diagonal part, that can be approximated by  $P = \nu_{\text{noise}}(A^t A)^{-1}$  square matrix where the diagonal element value  $P_{ii}$  is the noise variance divided by the number of measurements of the sky for pixel  $i$ . The map making equation is then solved in its final form:

$$P(A^t N^{-1} A)(\tau - A s_0) = P A^t N^{-1}(d - s_0).$$

The CG method requires both the initial residual  $A^t N^{-1}(d - A s_0)$  being provided and the operators working on the sky,  $P$  and  $A^t N^{-1} A$ . Then a stopping criterion is required. If quick estimation algorithms are well designed, little work is left to the Conjugate Gradient optimization, so we expect quick convergence. A first choice should be to stop the optimization when the optimization algorithm converges limited by machine precision. This shows when small oscillations in the map residuals are observed. But systematic errors on the Spherical Harmonic Power Spectrum (SHPS) due to sky pixelisation (Healpix pixel window functions, see Sect. 5.1) turn out to be enhanced by the CG iterator and poor scanning strategy redundancy. This effect is difficult to correct for, if the iteration number changes from one timeline to another. This is why we chose to stop the iteration at a predefined iteration number  $N_{\text{iter}}$ . In the following simulations we used  $N_{\text{iter}} = 5$ , as a working choice that minimizes computation time.

Given the pointing matrix  $A$ , operating  $A$  or  $A^t$  scales as  $N_{\text{sp}}$ , and the CPU time used by this Conjugate Gradient iterator scales as  $N_{\text{iter}} N_{\text{sp}} \log N_{\text{sp}}$ , thus being very efficient.

### 3.4. Classical “optimal” map-making: MirageDC

#### 3.4.1. A fast timeline slow-drift subtractor:

##### SlopeKillerMask

The goal of quick estimation algorithms is to quickly provide a rough but reliable estimation of the sky to feed the input the Conjugate Gradient method. Quick maps have to be good enough to ensure that when subtracted from the timelines, the residual is dominated by noise.

SlopeKillerMask assumes that the observing strategy describes large circles on the sky, the scanning strategy chosen by Archeops and Planck. The most dramatic features on a timeline are low frequency noise (see Fig. 3b), and the Galaxy. In the following, low frequency noise is assumed to have negligible power above the scanning (spinning) frequency. Then SlopeKillerMask masks Galactic signal, computes the low frequency component of the timeline, subtracts it, and then calculates the map by simple coaddition.

SlopeKillerMask first scans the timeline pointing, looking for the maxima in Galactic latitude to store them in memory. Data between these maxima are stored in a set named *Circles*. For each circle, samples pointing toward a bright region of the galaxy are flagged and discarded; bright Galactic regions are defined as brighter than a user-chosen threshold (typically 5 MJ/st) at a wavelength of 100  $\mu\text{m}$  using SFD Galactic model Finkbeiner et al. (1999). We fit a straight line over the valid

Bolometer signal sample values with time. *Offset* is finally stored as the value of the fitted line in the middle of the circle, which is a very robust estimation of the low frequency evolution of the timeline, provided the data has been properly flagged for glitches and cosmic rays before the map-making process. If not, this algorithm can be made insensitive to glitches by running the straight fit twice; after fitting the straightline, we compute the variance of the timeline minus the fitted line. We flagged samples that were 5 standard deviations from the fit. Doing so, we found that moderate and large amplitude glitches, as well as bright point sources, were flagged with adequate efficiency. We then computed the straight line fit once more and stored the fitted value in the middle of the circle.

The list of *Offset* was then used to compute the estimated slow drift component of the timeline by simple interpolation in the list, and this slow drift timeline is subtracted from the bolometer signal. A simple upgrade instead of interpolations would subtract a spline function of the offset list. The resulting timeline  $\mathbf{d}_{\text{sub}}$  is then projected onto the sky  $\mathbf{W}_{\text{coad}} \mathbf{d}_{\text{sub}} = \mathbf{s}_0$ , giving the fast rough estimation. Computation time scales as the timeline sample number and is found to be negligible as compared to the rest of the map-making process.

### 3.4.2. Iterative optimisation

The iterative optimisation of the map is a straightforward use of the Conjugate Gradient iterator described in Sect. 3.3. It uses the input timeline and the rough sky estimation to compute the optimised sky map. Five Conjugate Gradient cycles are used, and the residual modulus is typically divided by 100. Optimisation is indeed a time consuming step that scales as  $N_{\text{iter}} N_{\text{sp}} \log N_{\text{sp}}$ .

### 3.5. MirageAC

When low frequency noises happen to pollute data above the scanning frequency, the SlopeKillerMask algorithm does not produce adequate maps. MirageAC was designed to high-pass filter the timeline in order to suppress noise, as well as signal, under the frequency cutoff. The filtering tool is the fundamental building block of such a method. We chose to filter in the Fourier space handling edge effects, and used gap filling software to prepare the timelines. The main drawback of AC filtering is related to the Galaxy, because it is very luminous compared to the CMB. As expected, high-pass filtering induced ringing around large features. The way out of this problem is to subtract overly luminous objects from the timeline before filtering. But large low frequency noise prevents good estimation of the bright signal before filtering, so we designed an iterative algorithm to achieve a compromise.

#### 3.5.1. Quick map estimation: Rough filtered map-making

The input timeline is kept unmodified throughout the process. A copy of it is high-pass filtered and then projected onto a filtered sky map. The distribution variance of the filtered sky map

is computed outside the Galaxy. Filtered sky map pixels are then saved above a threshold of 2.5 standard deviation on bright object Map:  $\mathbf{s}_B$ . Making another copy of the input timeline, we first subtract the timeline obtained by observing the bright object Map:  $A^t \mathbf{s}_B$ . We filter the resulting timeline, project it, threshold it and add the result to the bright object Map. We loop on the process four times. Then in the last loop, we do not threshold the filtered sky map, but add it to the last bright object Map to compute the Rough Filtered Map, with noise.

This algorithm was shown to effectively remove low frequency noise and suppress ringing effect on the timeline and computed maps by a factor 20. Of course, the bright sky features (Galaxy) are high-pass filtered, in a way that depends on the scanning strategy. If this strategy is complex (Archeops), the bias induced on the maps can only be studied through simulations. This algorithm scales as  $N_{\text{sp}} \log N_{\text{sp}}$ , (dominated by the FFT transforms); it is thus significantly slower than the SlopeKillerMask algorithm and follows the same scaling as the iterative optimisation. An interesting alternative to this code is the *filtering without ringing* algorithm presented in (Amblard & Hamilton 2003). Tests need to be performed to compare respective performances.

### 3.5.2. Iterative optimisation in MirageAC

Iterative optimisation follows similar ideas. The Rough Filtered Map is *subtracted* from a copy of the input timeline. The resulting timeline, assumed to be dominated by noise, is processed by the Conjugate Gradient Iterator that computes a Correction Map. The optimised Sky Map is the sum of the Rough Filtered Map and the Correction Map.

## 4. Simulated timeline data

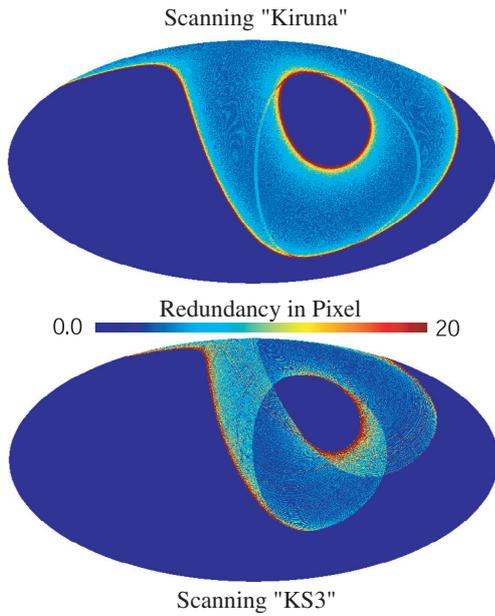
In order to test and optimize the map-making method, simulation tools were developed to generate realistic timelines.

### 4.1. Input sky maps

A CMB temperature anisotropy map was generated using CMBfast (Zaldarriaga et al. 2000) and Synfast (Gorski et al. 1998) codes. For these tests, a *standard* Cosmology model was used, namely:

$$\Omega_{\text{tot}} = 1, \Omega_{\Lambda} = 0.7, \Omega_{\text{M}} = 0.026, \Omega_{\text{B}} = 0.04, h_{100} = 0.65.$$

A galactic map was generated using a galactic dust emissivity model. Here the best fit extrapolation model of Finkbeiner et al. (1999) was used. This map is extrapolated from the 100  $\mu\text{m}$  emission and 100–240  $\mu\text{m}$  flux ratio maps taken from IRAS and COBE-DIRBE data. To take a finite instrumental resolution into account, the input map is convolved with Gaussian beam, assumed to be 10' *FWHM*. Optical frequency dependence on sky signal (dust, CMB, ...) is integrated on an optical filter assumed to be of top hat shape, with  $\Delta f/f = 30\%$ . Convolved CMB maps and galactic maps are then *observed*, given the pointing of the instrument, to obtain a *signal only* timeline.



**Fig. 1.** Mollweide projections of the two scanning strategies used in this study. **a)** Kiruna scanning strategy for the Archeops mission. The label Kiruna corresponds to the theoretical sky coverage of a 25-hour Archeops Flight. This sky coverage is homogeneous with timelines cutting each other at wide angles. No gaps in timelines are simulated. **b)** Sky coverage of the third Archeops Kiruna Scientific flight (KS3): a *real life* example.

#### 4.2. Scanning strategies

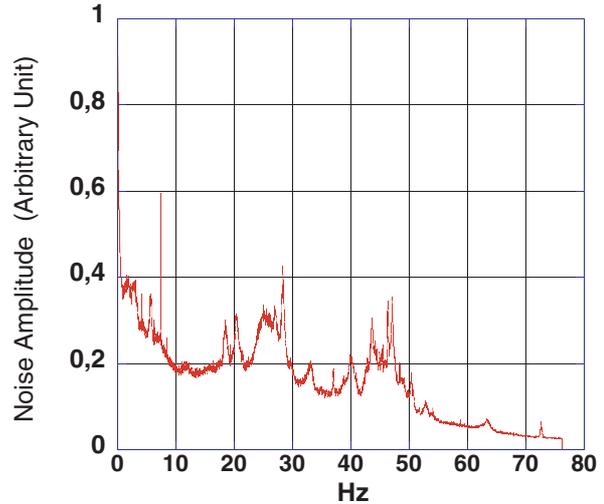
Two scanning strategies are being used. Their Mollweide projections of sky coverage are shown in Fig. 1. The first one is a *theoretical* Kiruna 24-h scanning strategy for the Archeops mission, which was limited to  $9 \times 10^6$  samples to allow quick tests on our workstation. This is a best case sky coverage because it is homogeneous, with timelines cutting each other at wide angles and no data gaps. The second copies an Archeops third Kiruna Scientific Flight (KS3) bolometer pointing file, including data gaps and inhomogeneous sky coverage, which is an example of *real-life* data.

#### 4.3. Noise models

Our models assume that noise properties are stationary. Noise timelines are then added to the *signal only timeline* to make input timelines of the map making. Two kinds of noise spectrum are being used. The first one is the classical  $1/f$  and white noise spectrum parametrised as:

$$\text{NPd}(f) = e_n \times (1 + f_{\text{knee}}/f) \times f / (0.002f_{\text{knee}} + f)$$

where  $e_n$  is the high frequency *flat noise* power spectrum density and  $f_{\text{knee}}$  the  $1/f$  knee frequency of the spectrum. The term  $f / (0.002f_{\text{knee}} + f)$  has been added to avoid divergence of the integral of Noise Power Spectral density at zero frequency. In the following, this noise spectrum is used unless otherwise specified. The second model shown in Fig. 2 uses the noise spectrum of what would be a strongly polluted experiment that displays wide noise spikes, in addition to a moderate  $1/f$  noise. Though



**Fig. 2.** Noise spectrum of what would be a strongly polluted experiment, displaying wide noise spikes, added to moderate  $1/f$  noise excess.

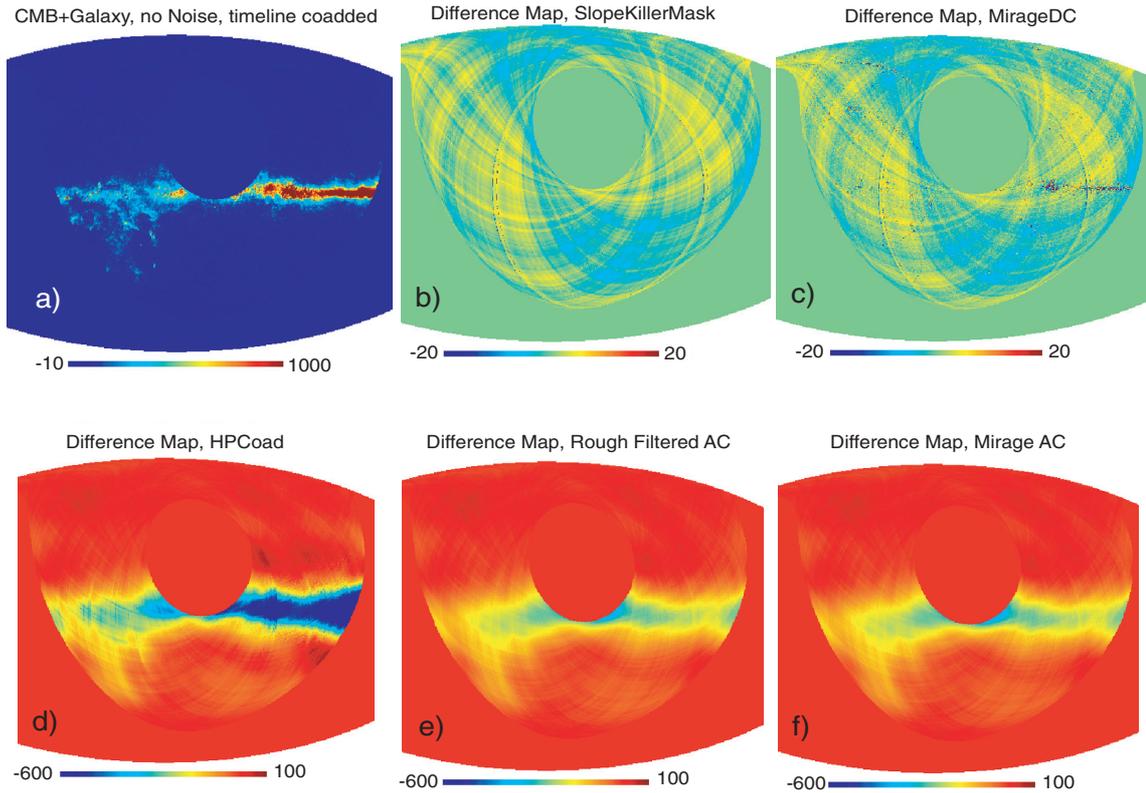
such an experiment is not likely to detect much of the CMB signal, the heavy structure of the noise spectrum is a tough test of the robustness of the algorithms presented in this paper.

The Mirage code may then be tested with various physical inputs (CMB, galaxy), various noise level as well as various scanning strategies. The entire simulation tools are developed in C++ within the framework of SOPHYA (Ansari et al. 2002).

#### 4.4. Mirage algorithms on simulated data

As a first test of systematics induced by our algorithms, we computed noise-free timelines by reading a Galaxy + CMB anisotropy sky map. We then processed these timelines by Mirage algorithms and subtracted the input sky map to produce difference maps. Figure 3 shows that SlopeKillerMask algorithm induces little distortion, while MirageDC optimiser somehow minimises the galaxy's brightest pixels. MirageAC quick estimation and optimiser do induce some residual ringing that distorts the galactic map, but this ringing is low enough to allow CMB anisotropy  $C_\ell$  estimation, if we mask the galaxy area.

As a second test, Fig. 4 shows Mollweide projections of sky maps computed by Mirage algorithms using noisy simulated timelines. Timelines were generated using a Galaxy + CMB anisotropy map, with  $1/f$ , and white noise. Mirage algorithms processed these timelines and we subtracted the input sky map to produce difference maps. Figure 4a shows the difference map obtained if we use the simplest of all mapmaking algorithms, using simple projections and averages of sky pixels measurements (coaddition). We used a timeline with a noise knee frequency lower than the scanning redundancy frequency. The map is dramatically dominated by noise-induced lines following the scanning path called *stripes*. Figure 4b shows the difference map computed using the SlopeKillerMask algorithm to remove low frequency drifts followed by coaddition. SlopeKillerMask, as well as MirageDC Optimiser, produces maps with no stripe, and the noise level is lower where the



**Fig. 3.** Mollweide projections of sky maps computed by Mirage algorithms from simulated timelines: **a)** the Signal (CMB anisotropy + Galaxy) map to be reconstructed; **b) to f)** difference maps: maps processed by Mirage algorithms running on simulated timelines with CMB + Galaxy signal only, with the Signal map subtracted. These maps show and sometimes enhance the systematics induced by algorithms processing the timelines with signals of large amplitude (Galaxy map brightest pixel’s value is 13 000). We observe that SlopeKillerMask **b)** induces very little distortion, while MirageDC **c)** tends to minimise the Galaxy brightest pixels. **d)** shows the difference map computed by high-pass filtering and coaddition of the timeline. Large systematic shows in and around the galaxy area due to the high-pass filter and its side-effect:ringing. **e)** and **f)** show the efficiency of MirageAC and the rough filtered map making algorithms to minimise distortion and ringing in maps. Still, large distortions do remain and MirageAC is of little use for mapping the Galaxy or other foreground, but turns out to be very useful for CMB  $C_\ell$  estimation when data involve noise with high frequencies and strong  $1/f$  features.

scanning frequency gives higher redundancy. Figures 4c et 4d show difference maps computed from a timeline with a noise knee frequency above the scanning redundancy frequency. Figure 4c shows the optimised map computed by MirageDC. The map shows strong pollution by striping, which is due to low frequency noise that is synchronous with observations and therefore projects on the sky. Figure 4d shows a difference map processed by the optimised MirageAC. Note the change in scales and noise with scanning strategy redundancy. To remove striping in Fig. 4c, we chose a MirageAC high-pass filter cutting frequency equal to one tenth of the noise knee frequency. The striping is dramatically reduced, though still noticeable. Raising the high-pass frequency allowed us to suppress stripes completely but at the same time filtered the underlying CMB signal. Thus the choice of the filter cutting frequency is a compromise.

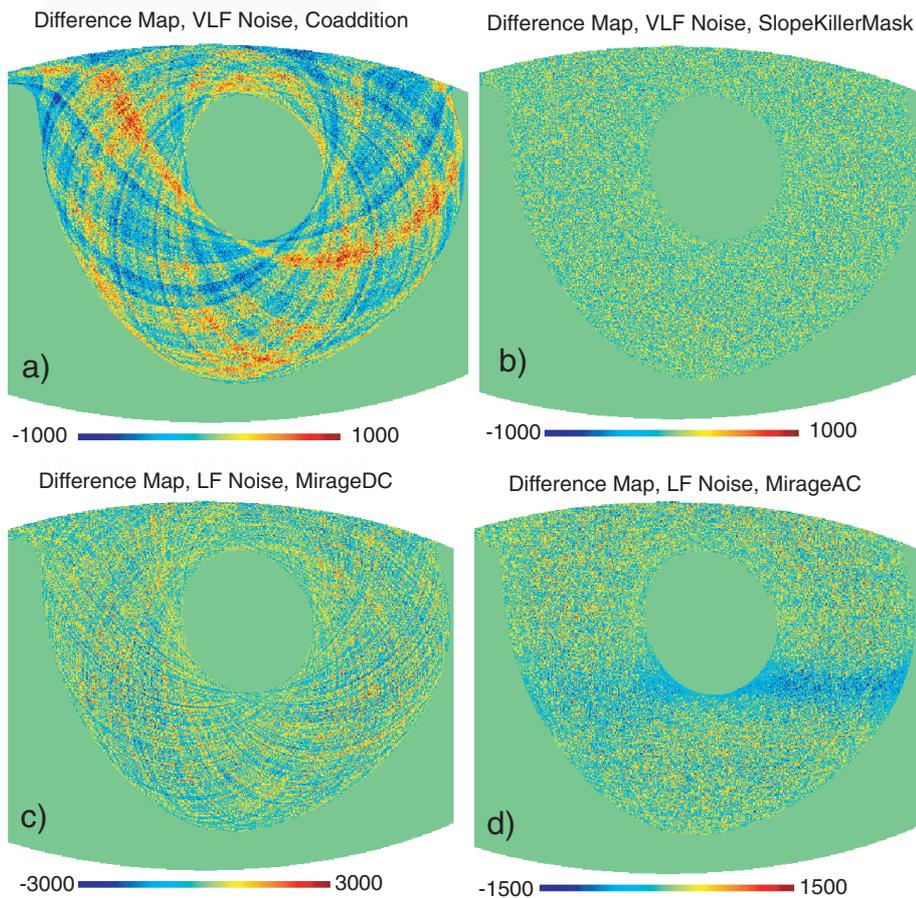
## 5. Errors on SHPS induced by Mirage algorithms

All data presented in the following are simulations. This work follows Hivon et al. (2002), and then Amblard & Hamilton (2003) within the Archeops collaboration where biases on SHPS induced by coaddition-based algorithms were studied in

detail. In this section we explain methods used within Mirage optimisation algorithms to minimise its systematic errors. We quantify the algorithm’s efficiencies and residual biases. The SHPS is calculated simply by expansion of the map in spherical harmonics, using the anaFast algorithm of Gorski et al. (1998) and then calculating the Pseudo  $C_\ell$ . No attempt was made to de-correlate  $C_\ell$  values due to limited sky coverage.

### 5.1. Sky pixelisation systematic errors

In the following, Healpix pixelisation and associated algorithms (Gorski et al. 1998) are often used. When a map is used as an intermediate step before computing power spectrum, the first systematic error to enter is the well known “pixelisation” effect. Healpix’s analytically computed pixel window function allows one to correct the effect of non-zero pixel size in evaluating of the SHPS. It is found that a healpix resolution factor  $N_{\text{side}} \geq 512$  must be used to compute SHPS with acceptable distortion up to  $\ell = 600$ . It turns out that the Conjugate Gradient optimiser manipulates data in the form of pixelised maps at each iteration. We observed that the pixel window effect is amplified when the number of iterations increases. The



**Fig. 4.** Mollweide projections of difference sky maps computed by Mirage algorithms from simulated timelines. The signal map simulates the CMB anisotropy and the Galaxy. **a)** to **d)** Are difference maps processed by algorithms running on simulated timelines with Signal and  $1/f$  + white noise. VLF stands for Very Low Frequency, the noise knee frequency is below scanning redundancy frequency. LF stands for Low Frequency, and the noise knee frequency is chosen above scanning redundancy frequency. **a)** Shows the results of simple coaddition on Very Low Frequency Noise. **b)** SlopeKillerMask and MirageDC algorithms are very efficient over very low frequency noise. **c)** On low frequency noise, MirageDC performs better than SlopeKillerMask, but fails as expected to remove stripes from maps. **d)** MirageAC algorithms are necessary to remove stripes if the timeline is contaminated by low frequency noise, but they induce residual ringing around the galaxy area that will have to be masked for CMB  $C_\ell$  estimation.

natural remedy would be to pixelise the sky at a higher resolution. But in real life data (Archeops), as well as for our simulations, increasing the healpix resolution factor to 1024 would result in little measurement redundancy. The large sky coverage associated with short flight duration prevents us from using a healpix resolution factor larger than 512. In order to be able to correct for the effect of the pixel window function, we decided to run the CG optimiser for a user-defined number of iterations. In the following all maps were pixellised using resolution factor  $N_{\text{side}} = 512$ . We then compared the additional distortions induced by the algorithms.

## 5.2. MirageDC

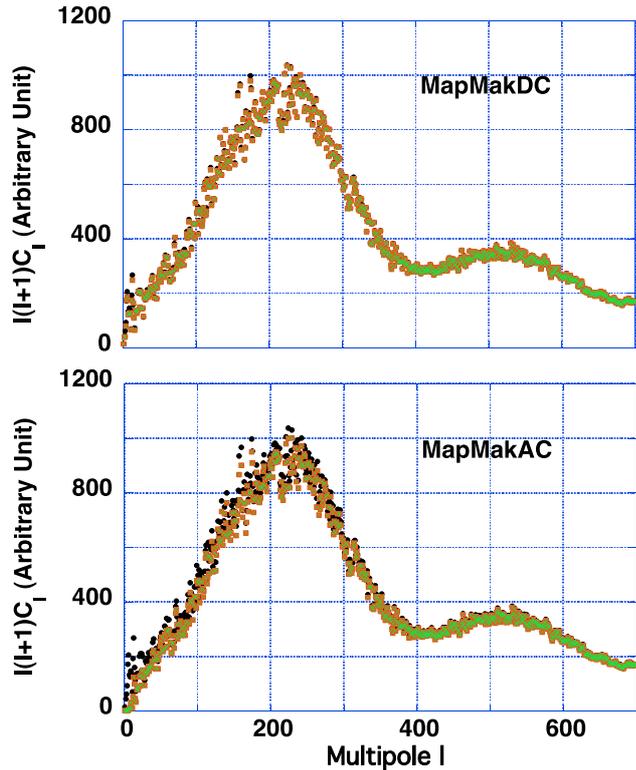
### 5.2.1. Filtering function

This estimates the distortions induced by the MirageDC when used to process noiseless timelines. The input timeline is prepared from the CMB map using the Kiruna 24-Hour scan sequence. No noise was added. We ran the MirageDC to compute

3 maps, a simple coaddition, the SlopeKillerMask quick estimation, and the iteratively optimised maps. These are, as expected, so similar that they can only be compared through their SHPS. Figure 5 shows the SHPS of the input convolved CMB map over the SHPS of the 2 output maps. No distortion on the SHPS is seen.

### 5.2.2. Noise suppression efficiency

The three map making algorithms mentioned above have all been tested and found to be linear with respect to their inputs. The simplest way to quantify the noise suppression efficiency is to run Mirage on a *Noise only timeline*. In fact, the lower the noise spectral amplitude, the better the suppression. In order to compare efficiency of different algorithms, we ran Mirage on the *Noise only timeline* using 3 kinds of noise spectra and 2 kinds of scanning strategies. The noise labelled VLF is a  $1/f$  + white noise with a knee frequency of 0.1 Hz. The noise spectrum labelled LF is a  $1/f$  + white noise with a knee frequency of 5 Hz. Finally the *bad* noise spectrum is the one presented in



**Fig. 5.** **a)** Upper panel: the Spherical Harmonic Power Spectrum of MirageDC computed maps. All the maps have been pixelised with a Healpix resolution parameter  $N_{\text{side}} = 512$ . Black dots are the SHPS of the reference CMB map with suppression of pixels outside the observed sky. Red dots are the SHPS of the map computed from a *CMB only* timeline using the SlopeKillerMask algorithm. Green dots are the SHPS extracted from the same timeline with the MirageDC iterative algorithm. No SHPS distortion is observed. **b)** Lower panel: the same but for the MirageAC algorithms. Black dots are for the SHPS of the input CMB map, red dots plot the SHPS of the GalaxMapMak computed map, and green dots are the SHPS extracted from MirageAC optimised maps.

Fig. 2. The computed SHPS are shown in Fig. 6. Since noise spectra are very different in amplitude and scanning strategies of very different redundancy and sky coverage, the  $Y$  axis amplitudes should not be compared between different windows. The most relevant information is the relative efficiency of the various algorithms. Figure 6 shows the most relevant results of these simulations. We see in Figs. 6a and 6b that simple coaddition is the worst algorithm. If noise excess happens only at very low frequency, a simple *Slope Killer* followed by coaddition is very efficient in suppressing noise, and the conjugate gradient iterative optimisation does not improve the results. If the noise excess happens to have a knee frequency above the spinning frequency, MirageDC Conjugate Gradient optimisation significantly improves the results for both scanning strategies. Finally if the noise spectrum display strong peaks spread all over the spectrum, the CG optimisation remains the best of the MirageDC algorithms. If we now look closely at the reconstructed SHPS spectrum from the MirageDC CG algorithm, we notice a spread in 3 lines of the SHPS, though at a much lower level than with simpler algorithms.

### 5.2.3. Is MirageDC the final word?

As shown above, MirageDC is efficient. The worse the noise spectrum, the more efficient the algorithm compared to simpler algorithms. But when the low frequency noise knee frequency happens to exceed the spinning frequency, excess spin synchronous noise projects on the sky producing stripes on the maps and the SHPS line pattern. Though MirageDC minimises the noise variance in the map optimally, it is not satisfactory for the physicist, because the line pattern of the pseudo- $C_\ell$  spectrum will hinder our attempts to recover the primordial CMB SHPS at all multipoles. Some other hypothesis is required to improve our map-making process. Having noticed that a strong low frequency noise will overwhelm the CMB SHPS at low frequency, we chose to high-pass filter the timelines, thereby sacrificing the low multipole SHPS information, in order to remove the striping: this is MirageAC. The hope is to recover the high multipole SHPS information.

## 5.3. MirageAC

### 5.3.1. Filtering function

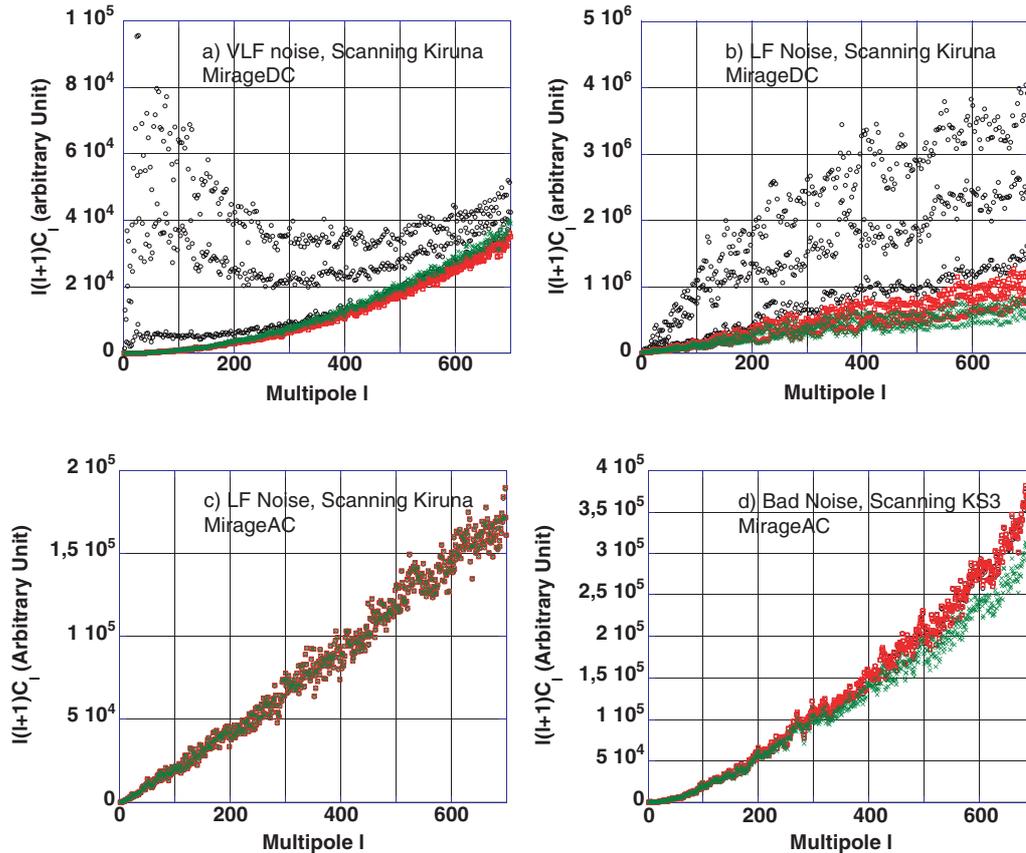
For this test we use the same timelines as those prepared for Sect. 5.2.1. We run the MirageAC, with a high-pass cutting frequency of 0.3 Hz and a 4 order Butterworth filter. Figure 5b presents the SHPS obtained from the 3 output maps, as well as the SHPS computed from coaddition of the same timeline. It can be seen that the 3 algorithms induce very similar SPHS systematics that on SHPS are significant at low multipole values. This systematic is induced by the high-pass filtering of the timelines and must be evaluated and corrected using simulations.

### 5.3.2. Noise suppression efficiency

Map-making AC is used to process the same timelines as in Sect. 5.3.1. Some results are presented in Figs. 6c and 6d. We observed that when the  $1/f$  + white noise knee frequency is lower or close to the High-Pass cutting frequency, Mirage AC efficiently suppresses the  $1/f$  noise contribution to SHPS as expected. The 3 outputs maps provide the same SHPS spectrum, even for LF noise, if the scanning strategy is homogeneous (Kiruna). But if the scanning strategy is inhomogeneous (e.g. Archeops), conjugate gradient optimisation does improve the result. Finally, if the noise spectrum has many peaks, the Conjugate Gradient optimisation improves the result at high multipole values (6d). Archeops data analysis showed that MirageAC is very effective when the noise spectrum rises at high frequencies. This happens when deconvolution of bolometer thermal constant low-pass filtering is necessary.

### 5.3.3. MirageAC and the Galaxy, validation of the method

The Galaxy is the largest and brightest non-solar-system object of the sky in the 100 GHz band. This reality induces two additional problems for the evaluation of SHPS using sky maps.



**Fig. 6.** The SHPS computed from maps calculated with the MirageDC and MirageAC algorithms. The timelines used included no signal, were generated from 3 different noise spectra, and projected on the sky following two scan sequences called Kiruna and KS3 (see Fig. 1). **a)** The black lines were computed by simple coaddition, the red lines were the result of the SlopeKillerMask algorithm, the green were the output of iterative optimisation using the Conjugate Gradient method. For noise that peaks below the scanning redundancy, both MirageDC algorithms do the job well. For knee frequencies above the scanning frequency **b)**, MirageDC optimiser improves the results compared to those from SlopeKillerMask algorithm as expected, but residual stripes in maps split the SHPS curve in 3 branches. **c)** and **d)** SHPS computed from MirageAC maps algorithms working on the same low frequency noise timelines. The black lines were computed by high-pass filtering and coaddition of the timelines; the red lines were the result of the GalaxMapMak algorithm; the green were the output of iterative optimisation using the Conjugate Gradient method. **c)** Mirage AC greatly improves the SHPS compared to **b)** and we notice that the 3 algorithms show very similar performances for low frequency dominated noise. But if the noise spectrum and/or the scanning strategy are more complex **d)**, iterative optimisation of maps improves the noise rejection.

Easiest to understand is that after a large signal, any high-pass filter rings over any given period of time. The ringing biases the map over an area that must be masked before SHPS estimation. This effect was taken care of from the start in the MirageAC algorithm.

The second effect is a bias in the noise spectrum evaluation (Amblard & Hamilton 2003). The power radiated by the Galaxy varies continuously as the scan moves through it. For noise estimation, the quick sky signal estimation is subtracted from the noise timeline. This sky signal estimation is pixelised with finite resolution, the same as for the output maps ( $N_{\text{side}} = 512$  for this work). When the scan crosses a quickly varying region of the Galaxy, the timeline samples digitizes a fast-changing signal, and the sky estimation averages this signal over the pixel size. When subtracted, the resulting timeline is contaminated by spurious bipolar features of pixel size, which consequently distorts the noise spectrum. This biased noise spectrum drives the Conjugate Gradient optimisation

algorithm to minimise some of the CMB features. To correct this behavior, we assume that the slope of Galactic signal is strongly correlated with the Galaxy intensity. Before computing the timeline noise spectrum (sky subtracted), we ignore samples pointing toward a galactic area brighter than 10 MJy/sr. The timeline is then restored with the gap-filling algorithm, and the *corrected* noise power spectrum is computed.

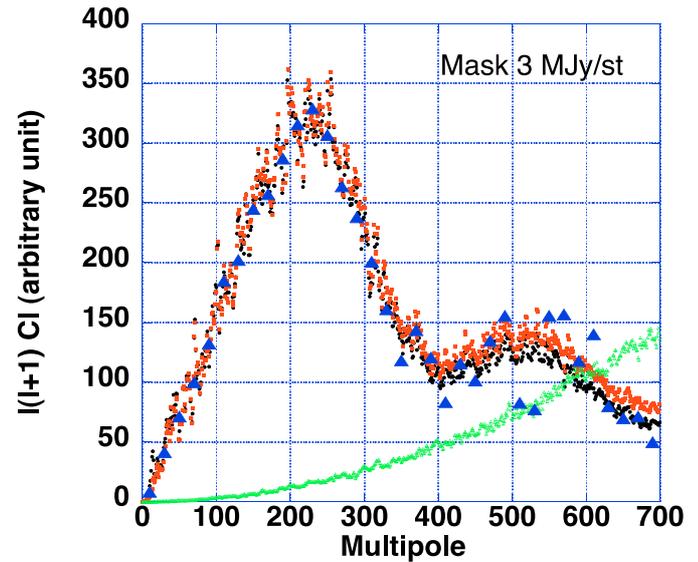
A final test was performed to check for additional bias and to validate Mirage AC. The simulated timelines were prepared, using the sum of a simulated CMB map and the Milky Way map observed with the Archeops scanning pattern. Noise was generated using a  $1/f$  + white noise spectrum. The same noise generation was used on two timelines, one with Sky + Noise1 (*Signal*) and one with noise only (*Noise1*). In addition, a second independent noise timeline (called *Noise2*) was generated with the same noise statistical properties. The High-pass filter cutting frequency and slope were set at 0.3 Hz and 4 poles. In

the output maps, the areas where IRAS 100  $\mu\text{m}$  flux was larger than 3 MJ/sr were set to zero.

MirageAC processed the 3 optimised maps. In order to test for additional bias in map-making, we first subtracted the Noise1 map from Signal Map and computed the SHPS. Though unrealistic in real life, this was a tough test of the method bias, because we expected to recover the input signal without noise or distortion. Figure 7 shows the results with black dots the “CMB only” SHPS, processed with MapMakAC, pixelised at same resolution and with the same sky coverage. Red dots are the recovered SHPS. Green triangles show the SHPS of the Noise1 map, *divided by 10*. We see that the recovered SHPS follows the CMB SHPS closely with a small excess at multipoles larger than 400. No excess power is seen when processing a noiseless Galaxy+CMB timeline. This excess power is found to depend on the noise amplitude and is due to a small error when the noise spectrum is extracted from the data (Amblard & Hamilton 2003); the quick-estimation maps are contaminated by some residual noise. When subtracted from the input timeline, the noise spectrum is then underestimated. This bias on noise amplitude is small (2% for this simulation), and can be easily corrected by Monte Carlo simulations making it negligible compared to the statistical noise shown in Fig. 8 (blue triangles). In this way, we subtracted the Noise2 SHPS from the Signal Map SHPS. The reconstructed SHPS (best estimation of experimental CMB SHPS) is noisy, and must be rebinned (blue triangles). We see that at multipoles larger than 400, where bias becomes significant, scattering of recovered SHPS around the CMB SHPS is much larger.

## 6. Conclusion

Two algorithms have been developed in order to optimise the sky maps computed from noisy timelines. Conjugate Gradient optimisation is efficient at minimising noise in a timeline. CG is effective on timelines displaying a complex noise spectrum and inhomogeneous sky coverage. If data is contaminated by low frequency systematics with a knee frequency higher than the scanning frequency, MirageDC, though effective at reducing noise on a SHPS, does not succeed in removing stripes on the computed maps. This problem is shared by all “Optimal” map-making software (Borrill 1999; Prunet 2000; Wright 1996; Natoli et al. 2001). The MirageAC algorithm does the job beautifully, but at the cost of suppressing the power spectrum at low multipoles. This effect must be corrected using Monte Carlo simulations. In this way, MirageAC can be seen as an improvement of the MASTER map-making method (Hivon et al. 2002; high-passed coadded timeline) with *optimal* processing of medium and high frequency noise features (lines and bumps in noise spectrum). Both algorithms handled experimental data efficiently, including data gaps and bright objects, and required very little CPU time. Mirage neither supports nor needs parallel processing. On a single processor Compaq Alpha 667 MHz workstation, MirageDC processes a  $10^7$ -sample timeline in 20 s for the quick estimation and 40 s per CG iteration.



**Fig. 7.** SHPS computed from four maps from Mirage-AC-processed timelines. Simulated timelines were generated according to the text. Map pixels pointing toward areas of IRAS 100  $\mu\text{m}$  flux greater than 3 MJ/sr have been set to zero. Black dots show the SHPS of the CMB only map, pixelised at the same resolution, and with the same sky coverage. Red dots present the SHPS of the Signal map, after subtraction of the Noise1 map. Green triangles show the SHPS of the Noise1 map, divided by 10. The recovered SHPS follows closely the CMB SHPS with a small excess at multipoles greater than 400. Blue triangles present the rebbinned Signal Map SHPS after subtraction of the “Noise2” SHPS.

Finally Mirage (AC algorithm) has been intensively used for an improved estimation of Archeops CMB power spectrum using a cross-spectrum method (Tristram et al. 2004).

*Acknowledgements.* The authors wish to thank the ARCHEOPS collaboration for using, criticising, and motivating the development of Mirage software. C. Magneville and the SOPHIA team supported us by discussions and providing the mathematical libraries used by Mirage. J. Rich, J.-Ch. Hamilton, J.-P. Pansart, C. Yeche, R. Teyssier, J. Macias-Perez, M. Tristram, F.-X. Désert and the anonymous Referee meticulously read and commented on this manuscript. Finally O. Doré and J.-Ch. Hamilton introduced us to the subject in the early stages of this work.

## References

- Amblard, & A., Hamilton, J.-Ch. 2003, A&A, 417, 1189
- Ansari, R., Aubourg, E., Le Meur, G., et al. <http://www.sophya.org>
- Barrett, R., Berry, M., Chan, T. F., et al., Template for the solution of Linear Systems: Building Blocks for Iteratives Methods, see also <http://www.netlib.org/templates/Templates.html>
- Bennett, C. L., Banday, A., Gorski, K. M., et al. 1996, ApJ, 464, L1
- Benoît, A., Ade, P., Amblard, A., et al. 2003, A&A, 399, 25
- Benoît, A., Ade, P., Amblard, A., et al. 2003, A&A, 399, 19
- Benoît, A., Ade, P., Amblard, A., et al. 2003, A&A, 424, 571
- Borrill, J. [arXiv:astro-ph/9911389], see also <http://www.nersc.gov/borrill/cmb/madcap.html>
- De Bernardis, P., Ade, P. A. R., Bock, J. J., et al. 2000, Nature 404, 955

- Doré, O., Teyssier, R., Bouchet, F. R., et al. 2001, *A&A*, 374, 358, see also <http://ulyse.iap.fr/cmbsoft/mapcumba/>
- Dupac, X., Giard, M. 2002, *MNRAS*, 330, 497
- Finkbeiner, D. P., Davis M., Schlegel, D. M., et al. 1999, *ApJ*, 524, 867, see also <http://astron.berkeley.edu/davis/dust/>
- Gorski, K. M., Hivon, E., & Wandelt, B. 1998, Proc. MPA/ESO Conf. on Evolution of Large-Scale Structure: from Recombination to Garching, 2–7 August 1998, ed. A. J. Banday, R. K. Sheth & L. Da Costa [arXiv:astro-ph/9812350], see also <http://www.eso.org/science/healpix>
- Hamilton, J.-Ch. 2003, *C. R. Phys.*, 4, 871
- Hanany, S., Ade, P., Balbi, A., et al. 2000, *ApJ*, 545, L5
- Hivon, E., Gorski, K. M., Netterfield, C. B., et al. 2002, *ApJ*, 567, 2
- Horowitz, P., & Hill, W., *The Art of Electronics* (Cambridge University Press), 2nd edition, 269
- Natoli, P., de Gasperis, G., Gheller, C., Vittorio N., 2001, *A&A*, 372, 346
- Press, W., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P., *Numerical Recipes in C: The Art of Scientific Computing* (Cambridge University Press)
- Prunet, S. 2000 [arXiv:astro-ph/0006052]
- Tegmark, M. 1997, *Phys. Rev. D*, 56, 4514; *ApJ*, 480, L87
- Tristram, M., Patanchon, M. G., Macias-Perez, J. F., et al. 2004 [arXiv:astro-ph/0501427], *A&A*, submitted
- Wright, E. L. 1996 [arXiv:astro-ph/9612006]
- Zaldarriaga, M., Seljak, U. 2000, *ApJS*, 129, 431