Exp Astron (2014) 37:109–125
DOI 10.1007/s10686-014-9372-7

ORIGINAL ARTICLE

Improving Bayesian analysis for LISA Pathfinder using an efficient Markov Chain Monte Carlo method

Luigi Ferraioli · Edward K. Porter · Michele Armano · Heather Audley · Giuseppe Congedo · Ingo Diepholz · Ferran Gibert · Martin Hewitson · Mauro Hueller · Nikolaos Karnesis · Natalia Korsakova · Miquel Nofrarias · Eric Plagnol · Stefano Vitale

Received: 4 June 2013 / Accepted: 7 January 2014 / Published online: 2 February 2014
© Springer Science+Business Media Dordrecht 2014

Abstract We present a parameter estimation procedure based on a Bayesian framework by applying a Markov Chain Monte Carlo algorithm to the calibration of the dynamical parameters of the LISA Pathfinder satellite. The method is based on the Metropolis-Hastings algorithm and a two-stage annealing treatment in order to ensure an effective exploration of the parameter space at the beginning of the chain. We compare two versions of the algorithm with an application to a LISA Pathfinder data analysis problem. The two algorithms share the same heating strategy but with one moving in coordinate directions using proposals from a multivariate Gaussian distribution, while the other uses the natural logarithm of some parameters and proposes

L. Ferraioli (✉) · E. K. Porter · E. Plagnol
APC, Université Paris Diderot, CNRS/IN2P3, CEA/Ifru, Observatoire de Paris, Sorbonne Paris Cité, 10 Rue A. Domon et L. Duquet, 75205 Paris Cedex 13, France
e-mail: luigi.ferraioli@apc.univ-paris7.fr

M. Armano
SRE-OD ESAC, European Space Agency, Camino bajo del Castillo s/n, Urbanización Villafranca del Castillo, Villanueva de la Cañada, 28692 Madrid, Spain

H. Audley · I. Diepholz · M. Hewitson · N. Korsakova
Albert-Einstein-Institut, Max-Planck-Institut fuer Gravitationsphysik und Universität Hannover, Callinstr. 38, 30167 Hannover, Germany

G. Congedo · M. Hueller · S. Vitale
University of Trento and INFN, via Sommarive 14, 38123 Povo (Trento), Italy

F. Gibert · N. Karnesis · M. Nofrarias
Institut de Ciències de l'Espai, (CSIC-IEEC), Facultat de Ciències, Campus UAB, Torre C-5, 08193 Bellaterra, Spain

Present Address:
L. Ferraioli
ETH Zürich, Institut für Geophysik, Sonneggstrasse 5, 8092 Zürich, Switzerland

jumps in the eigen-space of the Fisher Information matrix. The algorithm proposing jumps in the eigen-space of the Fisher Information matrix demonstrates a higher acceptance rate and a slightly better convergence towards the equilibrium parameter distributions in the application to LISA Pathfinder data. For this experiment, we return parameter values that are all within $\sim 1\sigma$ of the injected values. When we analyse the accuracy of our parameter estimation in terms of the effect they have on the force-per-unit of mass noise, we find that the induced errors are three orders of magnitude less than the expected experimental uncertainty in the power spectral density.

Keywords Markov Chain Monte Carlo · LISA · eLISA · Gravitational waves · LISA Pathfinder · LISA technology package · LTP

1 Introduction

LISA Pathfinder [1–3] is a European Space Agency mission aiming to develop and test the key technology for the forthcoming space-based gravitational wave detectors [4]. A considerable amount of the LISA Pathfinder mission time will be dedicated to a series of experiments for the calibration of the model parameters of the spacecraft dynamics. Such procedures will be periodically repeated during the LISA Pathfinder mission, and similar procedures will be necessary on future space-based gravitational wave detectors in order to ensure that the sensitivity of the instrument is optimised. Since the instrument calibration has a direct impact on the scientific outcome of the mission, it is of fundamental importance to develop a powerful and solid procedure for parameter estimation that allows for an accurate calibration in a short amount of time.

The problem of the calibration of parameters describing the dynamics was already discussed in previous works focusing on LISA Pathfinder [5–7]. In Ref [5, 6] a frequentist approach to the problem was developed using both linear and non-linear parameter estimation techniques respectively. In Ref [7] a Bayesian framework was developed for the second LISA Pathfinder mock data challenge, based on a classic Metropolis-Hastings Markov Chain Monte Carlo (MCMC) scheme for parameter estimation[8, 9]. From the point of view of the present paper, the method presented in Ref [7] is comparable with the algorithm working with the direct physical parameters and proposing the MCMC jumps from a multivariate Gaussian distribution.

The MCMC algorithm is a stochastic method that is very useful for carrying out Bayesian inference. The algorithm allows us to easily sample from a sometimes unknown probability distribution. By generating an ergodic Markov chain, we can sample from the underlying target distribution. The longer the MCMC, the more statistically independent the chain is and the better we model this distribution.[1] In general MCMC algorithms are relatively easy to implement. While guaranteed to

[1]Once a chain has been run long enough and it begins to settle down to an equilibrium distribution, each point in the chain should reach a state where it is only dependent on the previous point.

converge to the target distribution, the convergence time of the MCMC is problem dependent and is virtually impossible to estimate. However, the convergence can be accelerated if we use proposal distributions for the chain which we believe to be close to the target distribution. MCMC algorithms are normally divided into two parts : a "burn-in" phase where we allow the chain to find an equilibrium state, and a true MCMC which we use afterwards for statistical purposes. The end of the burn-in phase is usually decided "by-eye" as it is problem specific.

We propose a Bayesian approach to the calibration of the parameters of the spacecraft dynamics that is based on Metropolis-Hastings MCMC. The Metropolis-Hastings MCMC is a reference method for parameter estimation in Bayesian Statistics (see for example Ref [10]) and in space-based Gravitational waves data analysis (for an overview, we refer the reader to Ref [11] and references therein) where it has demonstrated its flexibility and power in the estimation of parameters for a variety of GW sources [12] . We compare here two different algorithms for the construction of the Markov Chain. One works with the physical parameters and proposes MCMC jumps from a multivariate Gaussian distribution, the other uses a conversion from some of the physical parameters to their natural logarithm in order to calculate the Fisher Information Matrix. This improves the numerical stability for the inversion operation to obtain the covariance matrix, and proposes MCMC jumps in the eigen-space of the Fisher Information matrix. Both methods implement a double-stage annealing heat treatment in order to effectively explore the parameter space at the beginning of the chain.

The rest of the paper is laid out as follows. In Section 2 we provide details about the LISA Pathfinder dynamics, as well as an introduction to the experiment aimed at calibrating some of the parameters of the dynamics along the principal measurement axis. In Section 3 we develop the mathematical basis of our algorithm, while in Section 4 we report the results of the application of our methods to a model of LISA Pathfinder and we analyze the two methods in terms of acceptance rate, convergence to the equilibrium distribution and error on the test mass noise estimation caused by the uncertainty on the estimated parameters.

2 A test case for LISA Pathfinder mission

LISA Pathfinder is a controlled three-body system composed of two test masses and the enclosing spacecraft. One test mass is free-falling along the principal measurement axis and it is used as a reference for the drag-free controller of the spacecraft. The second test mass is actuated at very low frequencies (below 1 mHz) in order to follow the free falling test mass. This actuation scheme provides a measurement bandwidth of $1 \leq f \leq 100$ mHz in which both the test masses can be considered to be effectively free-falling. The system has two output channels along the principal measurement axis, which provide estimates of the displacement of the spacecraft relative to the free-falling test mass and of the relative displacement between the individual test masses. We focus our attention on the principal measurement axis dynamics for which we can build a model that contains at least the five key parameters reported in Table 1 of [6].

Table 1 Parameter descriptions and parameter range values used in the simulation and in the MCMC search. 'Simulation values' are the values used to produce the set of synthetic data, they represent the 'true' values for the parameters in our test. The 'search range' defines the extent of the uniform prior distribution used in the MCMC

Name	Description	Simulation value	Search range
A_{df}	Actuation gain of the drag-free control loop acting on spacecraft thrusters	1.05	[0.5, 1.5]
A_{lfs}	Actuation gain of the electrostatic suspension actuating the second test mass at low frequency	1.05	[0.5, 1.5]
$S_{\Delta 1}$	Interferometer cross-talk coefficient coupling the first channel with the differential channel	1×10^{-5}	$[-1 \times 10^{-3}, 1 \times 10^{-3}]$
$\lvert\omega_1\rvert^2$	Stiffness coupling the motion of the spacecraft to the motion of the first test mass	$1.4 \times 10^{-6}\,\mathrm{s}^{-2}$	$[1 \times 10^{-7}, 1 \times 10^{-5}]\,\mathrm{s}^{-2}$
$\lvert\omega_2\rvert^2$	Stiffness coupling the motion of the spacecraft to the motion of the second test mass	$2.2 \times 10^{-6}\,\mathrm{s}^{-2}$	$[1 \times 10^{-7}, 1 \times 10^{-5}]\,\mathrm{s}^{-2}$

We simulated two typical experiments [6] for LISA Pathfinder: one simulates the motion of the spacecraft with respect to the free-falling test mass and the other forces a movement of the suspended test mass with respect to the free-falling mass. The signal in the first experiment is obtained by forcing the drag-free controller of the spacecraft to change sinusoidally its set-point. This sends a command to the thrusters that produces a sinusoidal movement of the spacecraft with respect to the free-falling test mass. In the second experiment we force a sinusoidal bias in the set-point of the system that controls the low frequency actuation of the second test mass. As a consequence a sinusoidal voltage is applied by the actuators to the second test mass and a movement with respect to the free-falling test mass is induced. For the generation of our synthetic signals we adopted the standard scheme in which the output of the instrument can be written as $s(t) = h(t) + n(t)$, where $h(t)$ is the time-dependent noiseless output of the LISA Pathfinder model for the dynamics (the model is described in Ref [6]) and $n(t)$ is an additive noise contribution generated in accordance to the procedure described in Ref [13]. The noise $n(t)$ is a Gaussian distributed correlated noise that reproduces the current knowledge of the LISA Pathfinder noise budget [14], its power spectral density at the two output channels is reported in Fig. 1 together with the cross spectral density between the two channels.

This noise is principally composed of thruster noise in the channel sensing the displacement of the spacecraft relative to the free falling test mass (first channel) and of a mixture of capacitive actuation noise and interferometric sensing noise on

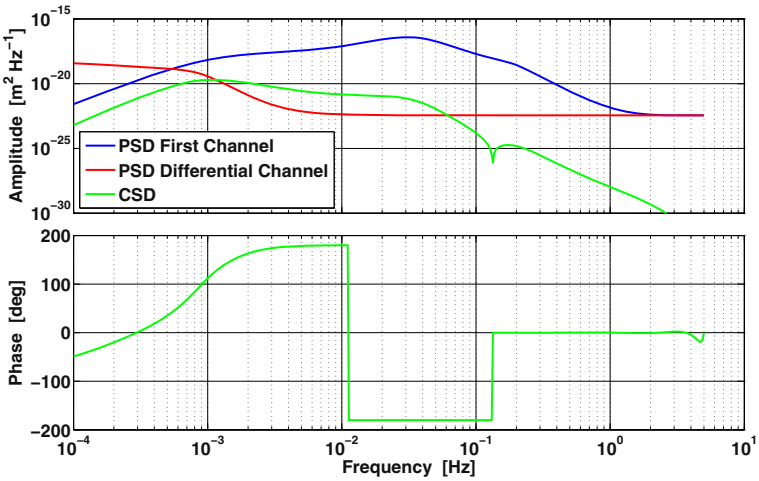

Fig. 1 Noise power spectral density (PSD) and cross spectral density (CSD) for the two output channels

the channel sensing the relative displacement between the test masses (differential channel). These two channels (which are displacement estimates at the interferometer output), need to be converted into force-per-unit-mass for the analysis of the force noise affecting the test masses. The conversion is based on a model of the spacecraft dynamics that is written in terms of the parameters described in Table 1. The calibration of the parameters of such a model therefore has a direct impact on the LISA Pathfinder test-mass noise budget, which is the main scientific target of the LISA Pathfinder mission.

In order to better understand our parameter estimation problem, we calculated the 5D likelihood surface over a uniform grid of points spanning the parameters ranges defined in Table 1. 2D slices of the likelihood surface are presented in Fig. 2 where we clearly see that there is one global solution over the 'search range'. The 'search range' for the parameters is defined from the prior knowledge of the system that principally comes from construction constraints and requirements.

In order to provide an idea of the sensitivity of the instrument to the parameters we report in Fig. 3 the system transfer functions for the parameters values used for the simulation (Table 1) and the values at the beginning of the Markov chain.

3 Bayesian analysis and Markov Chain Monte Carlo

Bayesian analysis has become a very useful tool in the field of space based gravitational waves [15–27], and has also been applied to LISA Pathfinder data analysis [7]. One of the advantages of Bayesian analysis is that it provides a direct method of calculating the marginalised posterior density function (PDF) for each of the parameters via Bayes' theorem

$$p(\boldsymbol{\lambda}|s) = \frac{\pi(\boldsymbol{\lambda})p(s|\boldsymbol{\lambda})}{p(s)}, \tag{1}$$

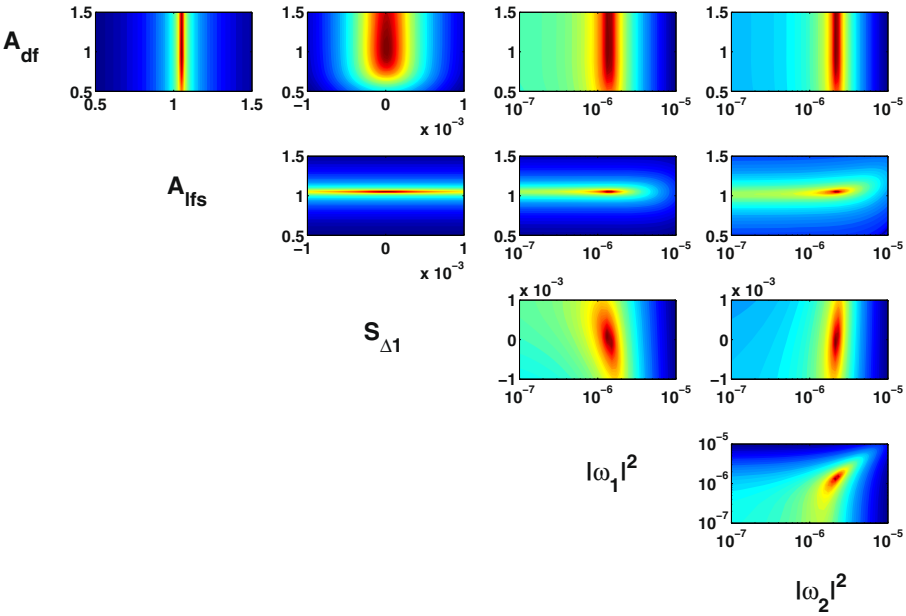

Fig. 2 2D slices in the 5D likelihood surface for the current parameter set. Note that some of the axes are in log-scale while others are in a linear scale. We clearly see that over the parameter ranges, the surface contains a single global solution

where $p(\boldsymbol{\lambda}|s)$ is the posterior density function (PDF) for the solution $\boldsymbol{\lambda}$ given the data $s(t)$, $\pi(\boldsymbol{\lambda})$ denotes the prior probability of the parameters and $p(s|\boldsymbol{\lambda})$ is the likelihood, which we will define below. The constant in the numerator is the Bayesian evidence and is defined by

$$p(s) = \int d\boldsymbol{\lambda}\, \pi(\boldsymbol{\lambda})\, p(s|\boldsymbol{\lambda}), \tag{2}$$

which, as can be seen, is independent of the solution set $\boldsymbol{\lambda}$ and is unimportant for our study.

While the previous works cited above have used sophisticated variants of the MCMC to search over a wide parameter space, in this study, as we can see from Fig. 2, we are dealing with a simplified surface. Even starting at the boundaries of our parameter priors, we can see that the surface contains a unimodal solution. Because of this we can use a straightforward MCMC.

Starting with a detector output $s(t) = h(t) + n(t)$, where $h(t)$ is the induced signal and $n(t)$ is the noise in the detector, we generate a template $h(t; \boldsymbol{\lambda}_i)$ by choosing a random starting point in the parameter space $\boldsymbol{\lambda}_i$, and then draw from a proposal distribution in order to propose a jump to another point in the space $\boldsymbol{\lambda}_{i+1}$. To compare both points, we evaluate the Metropolis-Hastings ratio

$$H = \frac{\pi(\boldsymbol{\lambda}_{i+1})\, p(s|\boldsymbol{\lambda}_{i+1})\, q(\boldsymbol{\lambda}_i|\boldsymbol{\lambda}_{i+1})}{\pi(\boldsymbol{\lambda}_i)\, p(s|\boldsymbol{\lambda}_i)\, q(\boldsymbol{\lambda}_{i+1}|\boldsymbol{\lambda}_i)}, \tag{3}$$

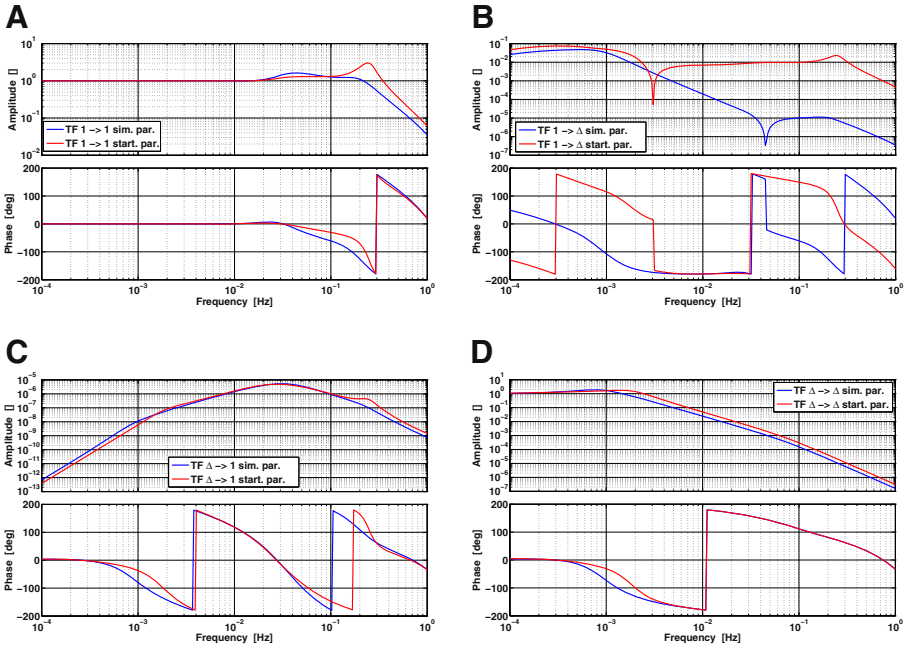

Fig. 3 This Figure displays the system input—output transfer functions for the parameter values used for the simulation data (Table 1) and the values at the starting of the Markov chain. *Blue line* refers to simulation parameters while *red line* refers to chain starting values. **A** Transfer function between the first experiment input and the first output channel. **B** Transfer function between the first experiment input and the differential output channel. **C** Transfer function between the second experiment input and the first output channel. **D** Transfer function between the second experiment input and the differential output channel

where we now define $p(s|\boldsymbol{\lambda})$ as the likelihood by

$$p(s|\boldsymbol{\lambda}) \equiv \mathcal{L}(\boldsymbol{\lambda}) = C\, e^{-\langle s-h(\boldsymbol{\lambda})|s-h(\boldsymbol{\lambda})\rangle/2}, \tag{4}$$

where C is a normalization constant. We are unconcerned by this constant as we ultimately are interested in the likelihood ratio between the original and proposed points, hence cancelling the constant. The angular brackets above denote the noise-weighted inner product of the two real functions $s(t)$ and $h(t)$

$$\langle h\,|s\,\rangle = 2\int_0^\infty \frac{df}{S_n(f)}\,[\tilde{h}(f)\tilde{s}^*(f) + \tilde{h}^*(f)\tilde{s}(f)], \tag{5}$$

where

$$\tilde{h}(f) = \int_{-\infty}^{\infty} \mathrm{d}t\, h(t) e^{2\pi \imath f t} \tag{6}$$

is the Fourier transform of the time domain waveform $h(t)$ and an asterisk denotes a complex conjugate. The quantity $S_n(f)$ is the one-sided noise spectral density of the detector. The quantity $q(\boldsymbol{\lambda}|\boldsymbol{y})$ is the transition kernel or proposal distribution used for jumping from $\boldsymbol{\lambda}_i$ to $\boldsymbol{\lambda}_{i+1}$. Once a jump is proposed, it is then accepted with probability $\alpha = \min(1, H)$, otherwise the chain stays at $\boldsymbol{\lambda}_i$. While we are sure that the MCMC

will eventually converge to the global solution, the timescale of this convergence is unclear. So, since we can never be sure that our starting values are close to the true values, it can take a long time for the chain to "walk" to the global maximum. A common way of decreasing this travel time is to heat the likelihood surface using a technique called simulated annealing. This has the effect of lowering and fattening structures on the surface. We will describe this method in more detail at a later stage.

For the LISA Pathfinder experiments, we expect to have an extremely high signal-to-noise ratio (SNR) ($\sim 10^5 - 10^6$). In this case the detector output has the form $s(t) \approx h(t)$. For this high SNR limit, it is expected that the errors in the parameter estimation will have a Gaussian probability distribution given by

$$p(\Delta\boldsymbol{\lambda}) = \sqrt{\frac{\Gamma}{2\pi}} e^{-\frac{1}{2}\Gamma_{\mu\nu}\Delta\lambda^{\mu}\Delta\lambda^{\nu}}, \tag{7}$$

where $\Gamma_{\mu\nu}$ is the Fisher information matrix (FIM)

$$\Gamma_{\mu\nu} = \left\langle \frac{\partial h}{\partial \lambda^{\mu}} \middle| \frac{\partial h}{\partial \lambda^{\nu}} \right\rangle, \tag{8}$$

and $\Gamma = \det(\Gamma_{\mu\nu})$. For the LISA Pathfinder project, there is no closed-form solution to the Fisher matrix calculation as it is not possible to take the derivatives of the template with respect to the parameters, nor evaluate the integral analytically. Therefore, the derivatives in the inner product need to be calculate numerically using a one-sided difference algorithm. Also, as the templates are expensive to generate (on the order of seconds), we update the FIM calculation every 100 iterations during the annealing scheme and as we believe that the FIM should remain almost constant at the global solution, every 5000 iterations during the MCMC.[2]

While we have seen that the likelihood surface is highly simplified compared to other gravitational wave problems, in that there exists a single mode solution, our goal here is to present an algorithm that can be started from any point in the parameter space. In order to accelerate the convergence of the chain to the global peak, we use a mixture of annealing methods as presented in [15, 16]. This first means re-writing the likelihood as

$$\mathcal{L}(\boldsymbol{\lambda}) = C \exp\left(-\frac{1}{2\beta}\langle s - h(\boldsymbol{\lambda}) | s - h(\boldsymbol{\lambda})\rangle\right), \tag{9}$$

where we introduce a heat scaling β. The first annealing scheme used is called thermostated annealing. While the SNR for the experiments are large, and even though there is no chance of getting stuck on a secondary solution in this particular problem, it is difficult to choose the initial annealing temperature by hand. If the temperature is too high, all structures on the likelihood surface are completely flattened and the MCMC aimlessly explores a flat surface. Therefore, as it is again costly to generate templates, we do not want to waste computer cycles by choosing the initial temperature too high. On the other hand, in the case of a multimodal solution, if the

[2]The update of the Fisher Information Matrix is performed in order to update the proposal distribution that is a multivariate Gaussian which covariance matrix is provided by the inverse of the Fisher Information Matrix.

temperature is too low, the chain can get stuck on a secondary maximum and never reach the global solution. Therefore, we allow the algorithm to chose its own initial annealing temperature according to

$$\beta = \begin{cases} 1 & 0 \leq \mathrm{SNR} \leq \mathrm{SNR}_T \\ \left(\frac{\mathrm{SNR}}{\mathrm{SNR}_T}\right)^2 & \mathrm{SNR} > \mathrm{SNR}_T \end{cases}, \tag{10}$$

which ensures that once we attain a SNR of greater than a threshold SNR, SNR_T, the effective SNR never exceeds this value. The working principle of such a method is that below the threshold SNR_T, the algorithm behaves as a standard MCMC. Therefore it spends all of its time conducting local exploration. However once it finds something interesting, i.e. $\mathrm{SNR} > \mathrm{SNR}_T$, the surface is heated according to (10). In our algorithm we run the thermostated annealing for the first 2000 iterations. For this problem, we choose our threshold SNR upon which we start injecting heat into the system as follows. We calculate the SNR at the boundary as defined by our priors, and pick an arbitrary value for β, which defines SNR_T. Therefore as the solution progresses more and more controlled heat is injected into the system, allowing the MCMC to effectively control its own environment, and converge faster. As previously stated, the SNRs involved in this problem are very large. If we take the boundary values for the parameters, as given in Table 1, the SNR at the boundary is 1.097×10^6, while at the injected values we have a SNR of 1.176×10^6. To choose the threshold SNR, we arbitrarily demand an initial value of $\beta = 1000$. Solving for the threshold SNR in the above expression then provides a threshold value of $\mathrm{SNR}_T \sim 3.5 \times 10^4$. After this we use a second annealing scheme, which is standard simulated annealing, to cool the surface down. To do this we use an annealing of the form

$$\beta = \begin{cases} 10^{\xi\left(1-\frac{i}{T_c}\right)} & 0 \leq i \leq T_c \\ 1 & i > T_c \end{cases}, \tag{11}$$

where i represents the chain index, and T_c represents the number of chain points during the cooling schedule. For this study, $\xi = \log_{10}\left(\beta_{\max}^{TA}\right)$ is the initial simulated annealing heat-index defined by the maximum temperature at the end of the thermostated annealing phase $\beta_{\max}^{TA}$. Again, as there is no risk of getting stuck on secondary maxima, we cool down the surface much quicker that we normally would for other problems. Here we set the cooling schedule equal to $T_c = 3000$ iterations.

One of the things that we wanted to test in this study was the difference between moving in eigen-directions and coordinate directions when we update the parameters. In Ref [15, 16] the proposed updates were done in the following manner. Once the Fisher matrix is calculated, we find the eigenvalues E_μ and eigenboldsymboltors $V_{\mu\nu}$. We then update the parameters in the eigen-directions according to

$$\lambda^\mu \to \lambda^\mu + \left(\sqrt{\frac{\beta}{D}}\right) \sum_{\nu=1}^{5} \frac{V_{\mu\nu}}{\sqrt{E_\nu}} n_\nu \tag{12}$$

where D is the dimensionality of the problem and $n_\nu = \mathcal{N}_\nu(0, 1)$ is a boldsymboltor of normal distributed random numbers with zero mean and unit variance. The factor of $1/\sqrt{D}$ ensures that the total jump in the parameter space has a length of $\sim 1\,\sigma$, while the heat factor β ensures that while the temperature is high, we take larger steps in the chain and thus more quickly explore the parameter space.

To move in coordinate directions, we first invert the Fisher matrix to obtain the variance-covariance matrix, i.e.,

$$\mathbf{C} \equiv C_{\mu\nu} = (\Gamma_{\mu\nu})^{-1}. \tag{13}$$

As $\mathbf{C}$ is a real symmetric matrix, we then use a Cholesky decomposition to write the covariance matrix as $\mathbf{C} = \mathbf{L}\mathbf{L}^*$, where $\mathbf{L}^*$ is the Hermitian conjugate or adjoint matrix of $\mathbf{L}$, and $\mathbf{L}$ is a unique lower triangular matrix. We should note here that for a real symmetric matrix the Hermitian conjugate is equivalent to the matrix transpose. We then generate a boldsymboltor of zero mean, unit variance random Gaussian values $\boldsymbol{n} = n_\nu$, and we update the parameter values according to

$$\lambda^\mu \to \lambda^\mu + \sqrt{\beta}(\mathbf{L}.\mathbf{n}), \tag{14}$$

where the update is along the coordinate directions for the parameters.

In the analysis conducted in this article, we combine the information provided by two LISA Pathfinder output channels and two separate experiments. Since we assume that the outputs of two distinct experiments are independent, the total likelihood for a number N_e of experiments is just the sum of the likelihoods from each experiment, i.e.

$$\mathcal{L}_{tot} = \sum_{i=1}^{N_e} \mathcal{L}_i. \tag{15}$$

The inner product in (5) can be generalized in a straightforward way to the case of multiple output channels by

$$\begin{aligned} &\langle \mathbf{s} - \mathbf{h}(\boldsymbol{\lambda}) \,|\mathbf{s} - \mathbf{h}(\boldsymbol{\lambda})\,\rangle = \ldots \\ &2\int_0^\infty [\tilde{s}_1 - \tilde{h}_1(\boldsymbol{\lambda}), \ldots, \tilde{s}_{Nc} - \tilde{h}_{Nc}(\boldsymbol{\lambda})]\Sigma_n^{-1}(f) \begin{bmatrix} \tilde{s}_1 - \tilde{h}_1(\boldsymbol{\lambda}) \\ \vdots \\ \tilde{s}_{Nc} - \tilde{h}_{Nc}(\boldsymbol{\lambda}) \end{bmatrix}^* \mathrm{d}f + c.c. \end{aligned} \tag{16}$$

Here $\Sigma_n(f)$ is the cross-spectral matrix, which contains the power spectral density at frequency f on the diagonal and the cross-spectral density on the off-diagonal terms, Nc is the number of output channels.

4 Results

Our study involves two different types of MCMC. For the movement in eigen-directions, we choose to work with the parameter set:

$$\lambda^\mu = \left\{A_{\mathrm{df}}, A_{\mathrm{lfs}}, S_{\Delta 1}, \ln\,|\omega_1|^2, \ln\,|\omega_2|^2\right\}. \tag{17}$$

Table 2 Simulated and recovered parameter values from the eigen-direction MCMC. The final column gives the standard deviation scaled offset from the true value

Parameter	Simulation value	Mean	Standard deviation	$(\lambda_R - \lambda_C)/\sigma_C$
A_{df}	1.05	1.049445	4.7356×10^{-4}	1.17
A_{lfs}	1.05	1.050002	3.3727×10^{-6}	−0.58
$S_{\Delta 1}$	10^{-5}	9.9314×10^{-6}	1.5669×10^{-7}	0.44
$\|\omega_1\|^2$	1.4×10^{-6}	1.40004×10^{-6}	1.8048×10^{-10}	−0.23
$\|\omega_2\|^2$	2.2×10^{-6}	2.20011×10^{-6}	1.5974×10^{-10}	−0.69

By working in the natural logarithm of the parameters $\left\{|\omega_1|^2, |\omega_2|^2\right\}$ we lower the condition number of the Fisher matrix, making it numerically more stable for inversion purposes. This matrix is still not as stable as we would like due to the small values of the parameter $S_{\Delta 1}$.[3] However, as this parameter can also have negative values we cannot use its logarithm. For the coordinate direction chains, we use the physical parameter values. Each chain was run for 2×10^5 iterations, with 5,000 iterations of annealing. The execution time for each chain is roughly 12 hours using a single core of a 16 cores E5640 2.66 GHz Intel(R) Xeon(R) processor. The simulated experiment includes two system identification investigations of 4 hours each. The LISA Pathfinder scheduled mission length is six months, during this period a densely scheduled program of experiments will be executed with the aim of completely characterise the system in free-fall condition. The typical length for an experiment is 24 hours, each experiment can involve several output channels sampled at 1 Hz. In some special cases, e.g. long noise measurements to assess noise stability and low frequency behaviour, a single experiment can last more than one day. To confirm the relative simplicity of the problem, starting with injected values of $\{1.885, 2.0399, -0.00746, 9.135 \times 10^{-6}, 6.327 \times 10^{-6}\}$, both chains converged to the global maximum within 3000 iterations. However, just to be safe, we will label the first 8,000 chain points as "burn-in". These points will be rejected when it comes to constructing a statistical analysis from the chains. These initial values correspond to a separation of approximately $\{10^3, 10^5, 10^4, 10^4, 10^4\}\sigma$ from the injected values, where σ here is the standard deviation calculated from the chain and presented later in Table 2. In Fig. 4 we plot the instantaneous acceptance rates for both the eigen-direction chain (blue-solid) and the coordinate direction chain (orange-dashed). We can see that we gain roughly a factor of two by using jumps in eigen-directions rather than coordinate directions with the chains having final acceptance rates of 63 % and 31 % respectively. This means that to independently sample the same posterior density, the coordinate direction chain would need to be twice as long as the eigen-direction chain.

In Figs. 5 and 6 we plot the convergence of the means and standard deviations of each parameter from the two MCMC methods. We can see by comparing both

[3]We are in log space in order to have the parameters of the same order of magnitude. In the case of S_Δ, since it can go negative, we can not use the logarithm and the parameter keeps its standard value of the order 10^{-5} that is considered small when compared with 1.

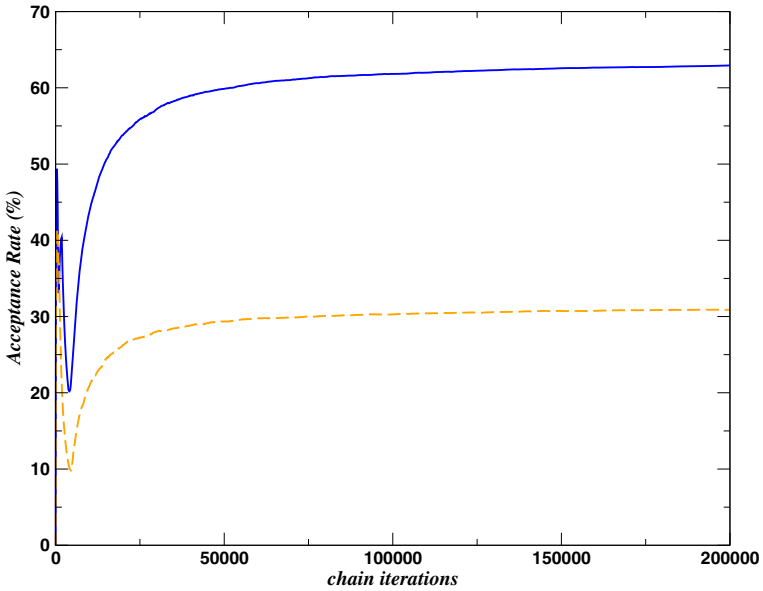

Fig. 4 The instantaneous acceptance rate for two Markov chains moving in eigen-directions (*blue-solid*) and coordinate directions (*orange-dashed*). We observe a factor of two improvement by moving in eigen-directions, as opposed to moving in coordinate directions

methods that they have a similar rate of convergence. We should point out here that we would not expect both chains to find exactly the same value. If we were to run N chains, we would expect the chains to produce N values all roughly within $\pm 1\sigma$ of the true value. Our criterion of evaluation here is how quickly the chains flatten with no large fluctuations. Although after 2×10^5 iterations, it is still difficult to prefer one method over the other. This is because of the fact that even though we are dealing with a simplified problem, it still takes 10^5 iterations for the Metropolis-Hastings algorithm to begin to converge. This suggests that for full convergence, the chains would need to be run for longer. However, it is still clear that we are finding the global peak very quickly.

In Fig. 7 we plot the marginalised, normalised histograms from the MCMC. On the x-axis we plot the true value offset scale $(\lambda_R - \lambda_C)/\sigma_C$, where λ_R is the true injected parameter value, λ_C is the chain value and σ_C is the standard deviation as calculated from the chain. We can see that both chains give us distributions very close to the true values. If we look closer, we can however see that the coordinate direction distributions have a higher kurtosis due to the lower acceptance rate. In all cases, the distributions are almost Gaussian with a skewness of $\sim 10^{-2}$. While there may seem to be no reason to prefer one method over the other, we will focus on the results from the eigen-direction chain due to its higher acceptance rate. We present the results in full in Table 2, but we quote here that the chain returned mean values for each parameter that were within $\{1.17, -0.58, 0.44, -0.23, -0.69\}\sigma$ of the true answer.

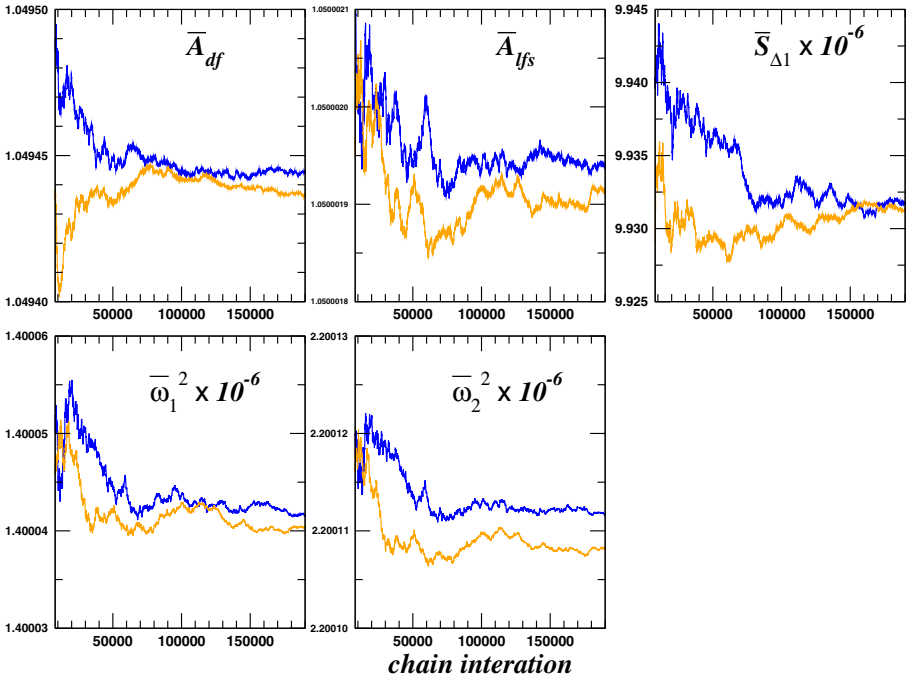

Fig. 5 The instantaneous parameter means for two Markov chains moving in eigen-directions (*blue*) and coordinate directions (*orange*). The two methods show similar rate of convergence and despite we are dealing with an unimodal problem it takes 10^5 samples to have a reasonable convergence. The means are calculated using a sliding window, so each value of the mean is updated as the number of iterations of the chain is increased

One of the aspects of the problem that we can also investigate from the MCMC is the correlations between parameters. Using the chains, we calculate the variance-covariance matrix using

$$C_{\mu\nu} = \frac{1}{N-1} \sum_{i=1}^{N} (x_\mu - \overline{x})(y_\nu - \overline{y}), \tag{18}$$

where $N = 192,000$ is the number of chain points (ignoring the 8,000 burn-in points at the beginning of the 2×10^5 iterations). An overbar denotes the mean of the parameter as calculated from the chain. Once we have the variance-covariance matrix, we can calculate the matrix of standard deviations and correlation coefficients according to

$$D_{\mu\nu} = \begin{cases} \sqrt{C_{\mu\nu}} & \mu = \nu \\ C_{\mu\nu}/\sqrt{\sigma_\mu \sigma_\nu} & \mu \neq \nu \end{cases}, \tag{19}$$

where $\sigma_\mu = \sqrt{C_{\mu\mu}}$ are the standard deviations of the marginalised chains for each parameter. The numerical values for the matrix, where the order of the rows and

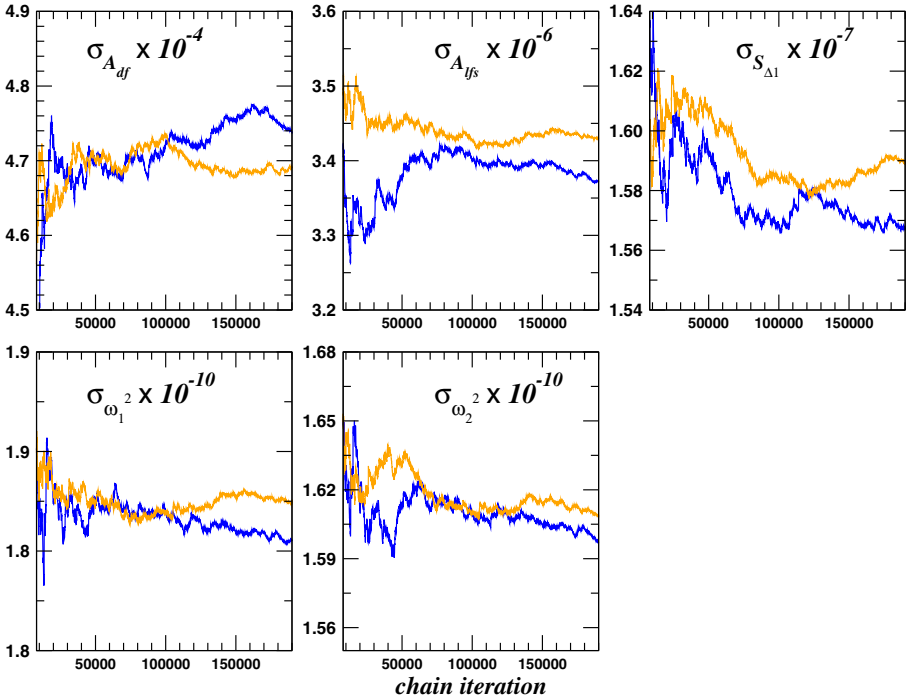

Fig. 6 The instantaneous parameter standard deviations for two Markov chains moving in eigen-directions (*blue*) and coordinate directions (*orange*). They are calculated using the same sliding procedure adopted for the means in Fig. 5

columns correspond to $\{A_{\rm df}, A_{\rm lfs}, S_{\Delta 1}, |\omega_1|^2, |\omega_2|^2\}$ are

$$D_{\mu\nu} = \ldots \begin{pmatrix} 4.736\times10^{-4} & -9.921\times10^{-3} & 1.662\times10^{-3} & -6.891\times10^{-3} & -1.547\times10^{-3} \\ & 3.373\times10^{-6} & 1.259\times10^{-2} & 6.697\times10^{-1} & 7.659\times10^{-1} \\ & & 1.567\times10^{-7} & -2.267\times10^{-1} & 8.097\times10^{-3} \\ & & & 1.805\times10^{-10} & 8.789\times10^{-1} \\ & & & & 1.597\times10^{-10} \end{pmatrix}. \tag{20}$$

We can see that the parameter set is mostly uncorrelated. However, we do see strong correlations between $A_{\rm lfs}$ and the two parameters $(|\omega_1|^2, |\omega_2|^2)$, i.e. 0.67 and 0.77, and between $|\omega_1|^2$ and $|\omega_2|^2$ themselves, i.e. 0.88. Such values are in agreement with the Cramer-Rao bound matrix up to the second decimal digit.

It is interesting here to quantify the accuracy of our result in terms of the LISA Pathfinder test mass noise budget. As discussed above, the parameters of the dynamics play an important role in the analysis of the test mass noise, since they are used for the calibration of the model that converts the displacement noise to force-per-unit-mass. On the basis of such a model we calculated the noise response to the

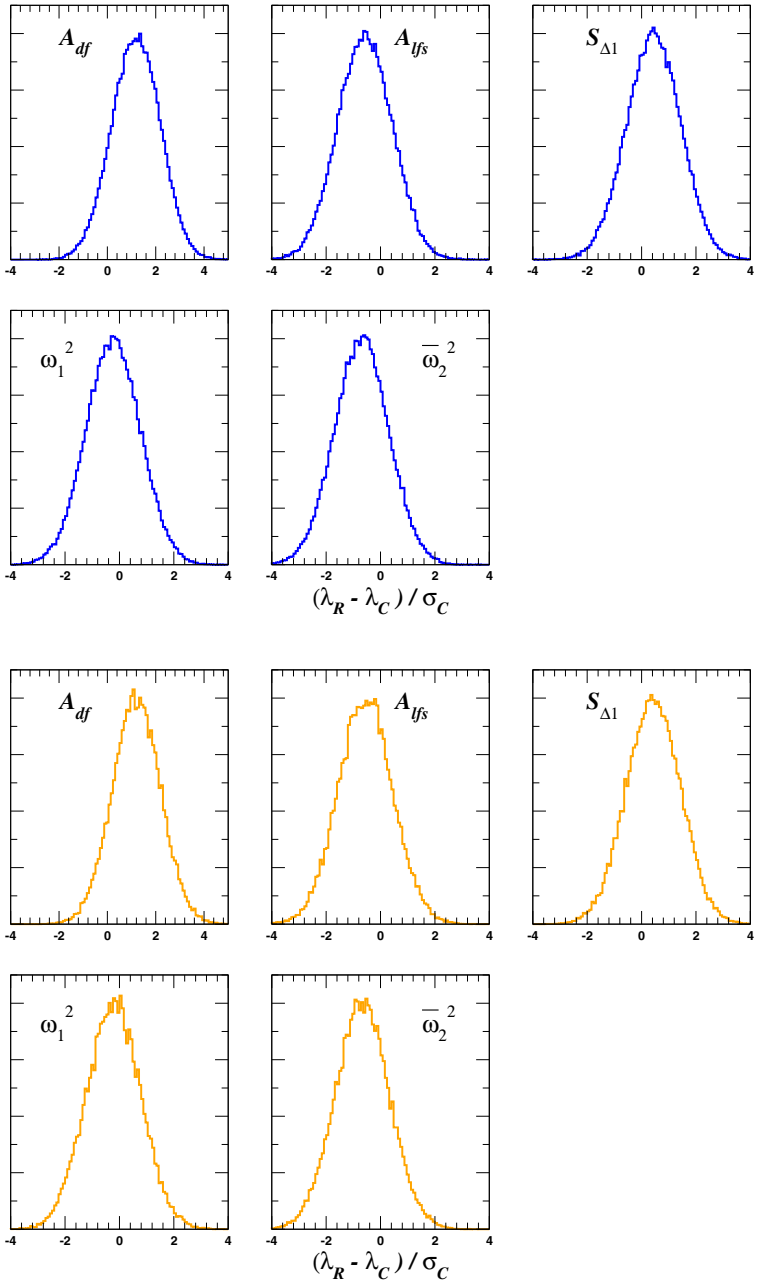

Fig. 7 This Figure displays the marginalised 1D histograms from the chains moving in eigen-directions (*left*) and coordinate directions (*right*). The x-axis origin is set at the true value of the parameters and the offset is the scaled difference between the true value λ_R and the chain value λ_C, divided by the standard deviation of the chain σ_C. Both chains give distributions very close to the true values but from a closer observation the coordinate direction distributions have a higher kurtosis due to the lower acceptance rate

Table 3 Test mass noise response to parameters variations. The response of the noise as a function of each parameter is linear $n(\nu) = A\nu + B$, therefore we obtained the slope A by a linear fit to the data. The noise variation Δn at 1σ is obtained multiplying A by the sample standard deviation of the parameters obtained from the chain

λ	A	Δn at 1σ [ms^{-2}Hz$^{-1/2}$]
A_{df}	1.9×10^{-28}	9×10^{-32}
A_{lfs}	6.9×10^{-15}	2×10^{-20}
$S_{\Delta 1}$	3.5×10^{-16}	5×10^{-23}
$\lvert\omega_1\rvert^2$	2×10^{-12}	4×10^{-22}
$\lvert\omega_2\rvert^2$	1.2×10^{-11}	2×10^{-21}

parameters as reported in Table 3. This data is obtained calculating the noise-per-unit-mass on an uniform grid of values for the parameters. The grid is centered around the simulation 'true value' and extends over 10σ, where σ is the sample standard deviation obtained by the chain. The response of the noise as a function of each parameter is linear $n(\nu) = A\nu + B$, therefore we obtained the slope A reported in Table 3 by a linear fit to the data. The noise change Δn at 1σ is obtained by multiplying A by the values of σ obtained by the chain. As can be seen in Table 3, even combining the results for all the five parameters we obtain $\Delta n \sim 2 \times 10^{-21}$[ms^{-2}Hz$^{-1/2}$]. Even assuming that we use 6 months data (the full mission duration) this value is three orders of magnitude lower than the expected experimental uncertainty (2.3×10^{-18} [ms^{-2}Hz$^{-1/2}$]). For this calculation we calculated the power spectral density with the well known Welch's overlapped segment averaging method, using a 4 sample Blackman-Harris window [28] and 50 % segment overlapping. A reasonably long LISA Pathfinder noise run can last for 3 days, and the current estimation for the noise level is 7.2×10^{-15} [ms^{-2}Hz$^{-1/2}$] at 1mHz [14]. Using this information in calculating the expected uncertainty, we obtain a value of 1.4×10^{-16} [ms^{-2}Hz$^{-1/2}$]. While if we assume a 6 month long data run, we obtain an experimental uncertainty of 2.3×10^{-18} [ms^{-2}Hz$^{-1/2}$], where the uncertainty is simply calculated as the value for the power spectral density divided by the number of averaging segments.

5 Conclusions

We have presented a study of two different MCMC methods for an experiment with LISA Pathfinder. While both methods are based on the Metropolis-Hastings algorithm, and both use the Fisher matrix to generate the covariance matrix, they differ in their treatment of the parameters and the way in which they move in the parameter space. In one case, we work with the physical parameter values and make MCMC jumps in the coordinate directions. For the other chain, we generate a more numerically stable Fisher matrix by working in the logarithm of two of the five parameters. This lowers the condition number of the matrix and makes it more stable during the inversion process to obtain the covariance matrix. We demonstrate that while

the experiment chosen for this study is quite simple, in that a single global solution exists in the likelihood surface, we gain roughly a factor of two in acceptance rate by moving in eigen-directions rather than in coordinate directions. Our chains both end up with parameter estimations that are very close to the true values. This is to be expected due to the extremely high SNR of the signal. However, as we are conducting a Bayesian inference, we are mostly interested in the full posterior distribution for each parameter. In this case, due to the higher acceptance rate, it is clear that calculating a more stable Fisher matrix and moving in eigen-directions is an optimal choice. The accuracy of the recovered parameters has been analyzed in terms of the effect they have on the force-per-unit-mass test mass noise estimate. We demonstrated that the error introduced by the parameters estimated from the MCMC chain is in any case more than three orders of magnitude less than the expected experimental uncertainty in the power spectral density.

Acknowledgment This research was supported by the Centre National d'Études Spatiales (CNES).

References

1. Armano, M., et al.: Class. Quantum Gravity **26**(9), 094001 (2009)
2. Antonucci, F., et al.: Class. Quantum Gravity **28**(9), 094001 (2011)
3. McNamara, P., Vitale, S., Danzmann, K.: On behalf of the LISA Pathfinder science working team. Class. Quantum Gravity **25**(11), 114034 (2008)
4. Amaro-Seoane, P., et al.: arXiv:1201.3621 (2012)
5. Antonucci, F., et al.: Class. Quantum Gravity **28**(9), 094006 (2011)
6. Congedo, G., Ferraioli, L., Hueller, M., De Marchi, F., Vitale, S., Armano, M., Hewitson, M., Nofrarias, M.: Phys. Rev. D **85**(12), 122004 (2012)
7. Nofrarias, M., Röver, C., Hewitson, M., Monsky, A., Heinzel, G., Danzmann, K., Ferraioli, L., Hueller, M., Vitale, S.: Phys. Rev. D **82**(12), 122002 (2010)
8. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: J. Chem. Phys. **21**(6), 1087 (1953)
9. Hastings, W.K.: Biometrika **57**(1), 97 (1970)
10. Hitchcock, D.B.: Am. Stat. **57**(4), 254 (2003)
11. Porter, E.K.: Gravitational Wave Notes (Online Journal) **01** (2009)
12. Babak, S., Baker, J.G., Benacquista, M.J., Cornish, N.J., Crowder, J., et al.: Class. Quantum Gravity **25**, 184026 (2008)
13. Ferraioli, L., Hueller, M., Vitale, S., Heinzel, G., Hewitson, M., Monsky, A., Nofrarias, M.: Phys. Rev. D **82**(4), 042001 (2010)
14. Antonucci, F., et al.: Class. Quantum Gravity **28**(9), 094002 (2011)
15. Cornish, N.J., Porter, E.K.: Phys. Rev. D **75**(2), 021301 (2007)
16. Cornish, N.J., Porter, E.K.: Class. Quantum Gravity **24**(23), 5729 (2007)
17. Gair, J.R., Porter, E.K.: Class. Quantum Gravity **26**, 225004 (2009)
18. Feroz, F., Gair, J.R., Hobson, M.P., Porter, E.K.: Class. Quantum Gravity **26**, 215003 (2009)
19. Babak, S., Gair, J.R., Porter, E.K.: Class. Quantum Gravity **26**, 135004 (2009)
20. Cornish, N.J., Porter, E.K.: Class. Quantum Gravity **23**, 761 (2006)
21. Cornish, N.J.: Class. Quantum Gravity **28**, 094016 (2011)
22. Littenberg, T.B., Cornish, N.J.: Phys. Rev. D **80**, 063007 (2009)
23. Cornish, N.J., Littenberg, T.B.: Phys. Rev. D **76**, 083006 (2007)
24. Crowder, J., Cornish, N.J.: Phys. Rev. D **75**, 043008 (2007)
25. Del Pozzo, W., Veitch, J., Vecchio, A.: Phys. Rev. D **83**, 082002 (2011)
26. Van der Sluys, M., et al.: Bull. Am. Astron. Soc. **38**, 992 (2006)
27. Petiteau, A., et al. Phys. Rev. D **81**, 104016 (2010)
28. Harris, F.: Proc. IEEE **66**(1), 51 (1978)