Bayesian model selection for LISA pathfinder

Nikolaos Karnesis,[1,*] Miquel Nofrarias,[1] Carlos F. Sopuerta,[1] Ferran Gibert,[1] Michele Armano,[2] Heather Audley,[3] Giuseppe Congedo,[4] Ingo Diepholz,[3] Luigi Ferraioli,[5] Martin Hewitson,[3] Mauro Hueller,[6] Natalia Korsakova,[3] Paul W. McNamara,[7] Eric Plagnol,[8] and Stefano Vitale[6]

[1]*Institut de Ciències de l'Espai, (CSIC-IEEC), Campus UAB, Facultat de Ciències, Torre C-5, 08193 Bellaterra, Spain*
[2]*ESAC, European Space Agency, Camino bajo del Castillo s/n, Urbanización Villafranca del Castillo, Villanueva de la Canãda, 28692 Madrid, Spain*
[3]*Albert-Einstein-Institut, Max-Planck-Institut für Gravitationsphysik und Universität Hannover, Callinstrasse 38, 30167 Hannover, Germany*
[4]*Department of Physics, University of Oxford, Keble Road, Oxford OX1 3RH, United Kingdom*
[5]*ETH Zürich, Institut für Geophysik, Sonneggstrasse 5, 8092 Zürich, Switzerland*
[6]*Dipartimento di Fisica, Università di Trento and INFN, Gruppo Collegato di Trento, 38123 Povo, Trento, Italy*
[7]*European Space Technology Centre, European Space Agency, Keplerlaan 1, 2200 AG Noordwijk, The Netherlands*
[8]*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*
(Received 16 April 2013; published 13 March 2014)

The main goal of the LISA Pathfinder (LPF) mission is to fully characterize the acceleration noise models and to test key technologies for future space-based gravitational-wave observatories similar to the eLISA concept. The data analysis team has developed complex three-dimensional models of the LISA Technology Package (LTP) experiment onboard the LPF. These models are used for simulations, but, more importantly, they will be used for parameter estimation purposes during flight operations. One of the tasks of the data analysis team is to identify the physical effects that contribute significantly to the properties of the instrument noise. A way of approaching this problem is to recover the essential parameters of a LTP model fitting the data. Thus, we want to define the simplest model that efficiently explains the observations. To do so, adopting a Bayesian framework, one has to estimate the so-called Bayes factor between two competing models. In our analysis, we use three main different methods to estimate it: the reversible jump Markov chain Monte Carlo method, the Schwarz criterion, and the Laplace approximation. They are applied to simulated LPF experiments in which the most probable LTP model that explains the observations is recovered. The same type of analysis presented in this paper is expected to be followed during flight operations. Moreover, the correlation of the output of the aforementioned methods with the design of the experiment is explored.

I. INTRODUCTION

LISA Pathfinder (LPF) [1,2] is a European Space Agency mission that will serve as a technology demonstrator for a future space-based gravitational-wave observatory like eLISA [3]. The LPF mission will prove geodesic motion by monitoring the relative acceleration of two test masses in nominally free fall in the frequency band of 1 to 30 mHz. The main instrument onboard the LPF is the LISA Technology Package (LTP), which is a suite of experiments with the aim of measuring and characterizing the different contributions to the differential acceleration noise between the two test masses. This characterization is the main task of the data analysis team. To that end, dedicated experiments are going to be performed in order to estimate the unknown parameters of the system. And for that purpose, a number of parameter estimation methods and models of the LTP have been implemented [4–6]. The main question that arises is about the suitability of the different models implemented or, in simpler terms, which model can describe better the observations of the experiment.

The motivation to implement an algorithm that will help us classify our models is due to the nature of the LTP system. There has been a lot of work trying to understand the instrument, and the models implemented are based on theoretical and experimental measurements from test campaigns [7,8]. Selecting the most probable model is crucial for the analysis for two main reasons: first, the most suitable LTP model that describes the observed physical effects is chosen, and, second, overfitting situations are avoided, together with biased estimation of the parameters of the system.

One could use several criteria and algorithms that classify competing models, but in the end, working in a Bayesian framework, the main aim is to calculate the *Bayes factor*, which is a comparison between the *evidences* of the models [9,10]. The evidence is defined as the probability of the data $\vec{y}$ given the model, that is, the probability distribution on $\vec{y}$ that quantifies the predictive capabilities of the given model.

Most of the approaches are based upon the likelihood evaluated at the maximum and a penalty for the number of parameters in the model consisting of multiplying by the so-called *Occam factor*. The *Occam's razor* is the principle that states that the simplest hypothesis that explains the observations is the most favorable. In our case, we assume that we have to compare two different LTP models, model M_1 and a simpler one, M_2, over a data set $\vec{y}$. If M_1 is a more complex model, it presents more predictive capabilities than M_2, which translates to more disperse evidence over the data set $\vec{y}$. Thus, in the case in which the data are compatible with both models, M_2 will turn out to be more probable than M_1, without having to express any subjective dislike for complex models [10]. In other words, it is almost certain that the more parameters in a model the better the fit to the data. But taking it to the extreme, we can imagine a model with as many parameters as the data to fit. In that case, we have overparametrized the model while the aim is to include only the parameters that *substantially* improve it.

In this paper, we investigate several methods to compare competing LTP models giving emphasis to reversible jump Markov chain Monte Carlo techniques. The plan of the paper is as follows. In Sec. II, we make a brief introduction to Bayesian methods, and in Sec. III, the LTP experiments and models are thoroughly explained, and we investigate several applications of our Bayesian techniques to LTP experiments. In Sec. IV, we compare the available methods and discuss their output in connection with the experiment design. We end with a summary and conclusions in Sec. V. The codes for the different algorithms are integrated into the LISA Technology Package Data Analysis (LTPDA) MATLAB toolbox [11,12], which comes together with proper documentation and help for the user.

II. BACKGROUND ON BAYESIAN STATISTICS

An algorithm that automatically penalizes higher-dimensional models is the reverse jump Markov chain Monte Carlo (RJMCMC) algorithm. The RJMCMC method [13–17] is widely used when dealing with nested models, meaning that we need to compare a set of models in which simpler models are a subset of a more complicated one. In fact, the RJMCMC algorithm is the generalized case of Markov chain Monte Carlo (MCMC) methods that is capable of sampling the parameter space and at the same time jumping between models with different dimensionality.

At this point, it would be convenient to describe the philosophy of the Bayesian framework and the Metropolis

algorithm and then move to the general case of the RJMCMC algorithm. The Bayes rule can be summarized by the equation

$$\pi(\vec{\theta}|\vec{y}) = \frac{\pi(\vec{y}|\vec{\theta})p(\vec{\theta})}{\pi(\vec{y})}, \qquad (1)$$

where $\vec{\theta}$ is a model parameter vector, $\pi(\vec{y}|\vec{\theta})$ is the *likelihood* of the parameters $\vec{\theta}$ over the data set $\vec{y}$, and $\pi(\vec{\theta}|\vec{y})$ and $p(\vec{\theta})$ are the *posterior* and the *prior* distributions of the parameters, respectively. Note that the *evidence* $\pi(\vec{y})$, or *marginal likelihood*, is often neglected in parameter estimation algorithms, as it serves only as a normalization constant:

$$\pi(\vec{\theta}|\vec{y}) \propto \pi(\vec{y}|\vec{\theta})p(\vec{\theta}). \qquad (2)$$

The Metropolis algorithm is one of the MCMC-type methods available that is widely used for parameter estimation purposes. It is based on sampling the parameter space by proposing new samples $\vec{\theta}_n$ and evaluating the likelihood at each step. By sampling the posterior distribution, probability distributions to the parameters to be estimated are assigned. It is certain that, given a large amount of steps in the parameter space, the Metropolis algorithm will converge to the set of parameters $\vec{\theta}_{\text{MAP}}$ that maximize the likelihood.

A. Calculating the Bayes factor

The evidence of a hypothesis X given the data set $\vec{y}$, $\pi_X(\vec{y})$, states the support for this hypothesis or, in other words, how much the data favors a given model. In our case, the hypothesis is the model implemented to describe the LTP system. Consequently, given two different models X and Y, the Bayes Factor B_{XY} is a comparison between the evidences of model X and model Y given by their ratio

$$B_{XY} = \frac{\pi_X(\vec{y})}{\pi_Y(\vec{y})}, \qquad (3)$$

where

$$\pi_K(\vec{y}) = \int \pi(\vec{\theta}_K, \vec{y})d\vec{\theta}, \qquad K = X, Y. \qquad (4)$$

This integral is extremely costly to evaluate, especially when the model becomes complicated with higher dimensionality. In the next sections, a selection of estimators of the Bayes factor is described. Most of the estimators evaluate directly the Bayes factor, while other techniques are used to estimate the evidence for each model. All of them are used to investigate our LPF models and mission experiments. In the end, we can draw conclusions about the competing models given the estimated value of the Bayes factor. If $B_{XY} < 1$, the evidence is negative, and the

observations support model Y. If $B_{XY} > 1$, the evidence is positive, and model X is more favourable than model Y.

B. Reversible jump Markov chain Monte Carlo algorithm

The RJMCMC algorithm is a robust and efficient tool to estimate the Bayes factor. It can be shown [13,15] that after a large number of iterations it will converge to the true value of the Bayes factor. The only drawback is the computational cost of the algorithm. When more than three models are being compared, meaning that many transdimensional moves have to be performed, a considerable amount of time is required for convergence. The algorithm implemented in this work is a special case of the *Metropolized Carlin and Chib* method [14]. More specifically, let us suppose that we have a total number K of models to compare given a data set $\vec{y}$. Then, the recipe for our RJMCMC method can be summarized in the following steps:

(1) Initialization: Choose an initial model k and the corresponding parameters $\vec{\theta}_k$.
(2) Apply the Metropolis algorithm for model k. This step is also called the "in-model step."
(3) Generate new $\vec{\theta}_{k'}$ from a multivariate Gaussian probability density function (PDF) and a random number $\rho \, \epsilon [0, 1]$ from a uniform distribution. This is the step in which we propose a new model k'.
(4) Calculate the acceptance ratio α',

$$\alpha' = \min\left[\frac{\pi(\vec{y}|\vec{\theta}_{k'})p(\vec{\theta}_{k'})g(u_{k'})}{\pi(\vec{y}|\vec{\theta}_k)p(\vec{\theta}_k)g(u_k)}|\mathbf{J}|, 1\right], \quad (5)$$

where $g(u)$ is the proposal distribution from where the "dimension matching" parameters u are drawn [18], and $|\mathbf{J}|$ is the Jacobian:

$$|\mathbf{J}| = \left|\frac{\partial(\vec{\theta}_{k'}, u_{k'})}{\partial(\vec{\theta}_k, u_k)}\right|. \quad (6)$$

(5) If $\rho < \alpha'$, we accept the new model k' with parameters $\vec{\theta}_{k'}$ and set $\vec{\theta}_k = \vec{\theta}_{k'}$.
(6) Iterate from step 2 until convergence.

The old set of parameters is connected to the new one by a well-defined function $\vec{\theta}_k = q(\vec{\theta}_{k'}, u)$ [and of course $\vec{\theta}_{k'} = q'(\vec{\theta}_k, u')$]. We use *independent* proposals, so $\vec{\theta}_k = q(\vec{\theta}_k, u) = u$ and $\vec{\theta}_{k'} = q'(\vec{\theta}_k, u') = u'$; thus, the Jacobian term in Eq. (5) is unity. The algorithm spends most of the time iterating "inside" the model that best describes the data. The RJMCMC method autopenalizes high-dimension models, also by taking into account the priors $p(\vec{\theta}_k)$ of each model k. They serve as an Occam factor integrated within the algorithm. Convergence is achieved if two main conditions are satisfied. First, the condition of *reversibility*,

which is stated in a simple way: The proposal function must be invertible, meaning that we can jump from the proposed parameters back to the current parameters. And second, we must satisfy the *dimension matching* condition, which in our case is always true since we use independent proposals in the acceptance ratio. After convergence has been achieved, a good approximation to the Bayes factor is given by [15,19–21]

$$B_{XY} = \frac{\text{of iterations in model X}}{\text{of iterations in model Y}}. \quad (7)$$

C. Other approximations

While the RJMCMC method directly estimates the Bayes factor, the other methods implemented in our work make an approximation to the evidence of each model. We implemented them as a cross-check for the RJMCMC method, but they also provide a certain freedom of choice, depending on the nature of the problem as well as the data available.

The first approximations we consider are the Laplace approximations. The Laplace approximations perform a comparison between the volume of the models in the parameter space and the volume of the uncertainty ellipsoid of the parameters [9]. This comparison is feasible if we assume that we work within a high signal-to-noise ratio (SNR) and therefore the posterior PDF is Gaussian near the maximum likelihood parameters, $\vec{\theta}_{MAP}$. Another requirement is that a sufficiently large number of samples must have been collected. Then, the evidence in Eq. (4) becomes

$$\pi_X(\vec{y}) \simeq (2\pi)^{D_X/2}|\mathbf{H}|^{1/2}\pi(\vec{\theta}_{MAP,X}|\vec{y}), \quad (8)$$

where D_X is the number of dimensions of model X and $\mathbf{H}$ is the Hessian matrix of the posterior. There are two main variations of the Laplace approximation. In the first one, we make use of the Fisher information matrix $\mathbf{F}$, calculated at $\vec{\theta}_{MAP}$, as an approximation to the expected covariance matrix [21,22]. Then, the evidence of the model becomes

$$\pi_X(\vec{y}) \simeq (2\pi)^{D_X/2}|\mathbf{F}|^{-1/2}\pi(\vec{\theta}_{MAP,X}|\vec{y}). \quad (9)$$

Of course, the main limitations of this method are associated with the confidence we have on the calculation of the Fisher matrix. Furthermore, as expected, the results appear to be poorer in comparison with the other methods as we move toward lower SNR areas. We can follow the notation of Ref. [21] and call this particular approximation the Laplace–Fisher (LF) approximation. Another well-known variation is the Laplace–Metropolis (LM) estimator of the marginal likelihood [9]. In this case, we use all necessary components for the calculation of the evidence from previous MCMC estimates. The parameters $\vec{\theta}_{MAP}$ are extracted from the chains of a MCMC parameter estimation run for the particular model, while we use the weighted

covariance matrix of the chains Σ, using a minimum volume ellipsoid (MVE) or a minimum covariance determinant estimator [23]. The MVE method has been implemented and integrated in the LTPDA toolbox. In this case, the evidence of model X can be written as

$$\pi_{\mathrm{X}}(\vec{\mathbf{y}}) \simeq (2\pi)^{D_{\mathrm{X}}/2} |\Sigma|^{1/2} \pi(\vec{\theta}_{\mathrm{MAP,X}}|\vec{\mathbf{y}}). \tag{10}$$

The LM method is considered to be a very reliable tool for the computation of the evidence of a model [9]. The third method we have used is the Schwarz–Bayes information criterion (SBIC) and is based on the following: After the assumption that the priors for each model follow a multivariate Gaussian PDF, the Schwarz criterion is defined as

$$\begin{aligned} S \simeq{}& \ln(\pi(\vec{\theta}_{\mathrm{MAP,X,}}|\vec{\mathbf{y}})) - \ln(\pi(\vec{\theta}_{\mathrm{MAP,Y}}|\vec{\mathbf{y}})) \\ & - 1/2(D_{\mathrm{X}} - D_{\mathrm{Y}})\ln(n), \end{aligned} \tag{11}$$

where D_{X} and D_{Y} are the dimensions of each model and n is the number of samples in the data. It can be proven that if $n \to \infty$ then $S \to B_{\mathrm{XY}}$, and the Schwarz criterion can be a good approximation to the Bayes factor. In fact, n must be chosen carefully so that $n = N_{\mathrm{eff}}$, where N_{eff} is the number of effective samples in the data that represent the growth of the Hessian matrix of the log likelihood [9].

D. Implementation

If the system's transfer function is $\mathbf{H}_s(\vec{\theta})$ and we consider known injection signals $\vec{\mathbf{x}}$, then the output of the system $\vec{\mathbf{y}}$ can be expressed as

$$\vec{\mathbf{y}} = \mathbf{H}_s(\vec{\theta})\vec{\mathbf{x}} + \vec{\mathbf{n}}, \tag{12}$$

where $\vec{\mathbf{n}}$ is the overall instrument noise and $h(\vec{\theta}) = \mathbf{H}_s(\vec{\theta})\vec{\mathbf{x}}$ is the response or template of the system. The noise in the LTP experiment can be approximated with Gaussian noise since we work in a high SNR regime and we can safely consider that any dependence on the system parameters is negligible. Consequently, the likelihood for a model with parameters $\vec{\theta}$ can be written as

$$\pi(\vec{\mathbf{y}}|\vec{\theta}) = C\exp\left[-\frac{1}{2}\langle\vec{\mathbf{y}} - h(\vec{\theta})|\vec{\mathbf{y}} - h(\vec{\theta})\rangle\right], \tag{13}$$

where the angular brackets denote the noise weighted inner product [24]

$$\langle\vec{a}|\vec{b}\rangle = 2\int_0^\infty [\tilde{a}^*(f)\tilde{b}(f) + \tilde{a}(f)\tilde{b}^*(f)]/S_n(f) \tag{14}$$

and where $S_n(f)$ is the power spectral density of the noise. For the proposal distribution appearing in the acceptance ratio of the Metropolis–Hastings algorithm, a multivariate Gaussian distribution is used. Although the choice of the

proposal distribution should not affect the final estimated parameter PDFs, it greatly affects the convergence speed of the chains to the target posterior distribution. The covariance of the multivariate Gaussian is calculated beforehand or during the search phase using the Fisher information matrix (FIM) [22]. In the case of the RJMCMC method, the FIM is calculated once for each model before the execution of the algorithm. It is computed numerically or analytically depending on the model. The LTP models are coded either in state-space format [25–27] or in pure analytical format in the acceleration domain [28].

The first step in the analysis is to apply a Fourier transformation to the data and estimate the power spectral density of the noise $\vec{\mathbf{n}}$, i.e., $S_n(f)$. As described in Ref. [4], the nature of the LTP experiment allows us to combine all the available information from different investigations. This means that we can combine the posterior distribution for each data set for each model. Then, we can apply MCMC analysis strategies to sample the posteriors and make use of any of the criteria described in Sec. II C or sample the joint parameter spaces with the RJMCMC method that was described in Sec. II B. Technically, since the RJMCMC algorithm can be seen as a generalized MCMC method, it performs a parameter search by simultaneously sampling the likelihood, and consequently it maps the joint posterior distribution and assigns PDFs to the parameters. The extra point is that we can estimate directly the ratio of the evidences of the models and test the hypothesis made for the data.

III. MODELING THE LTP EXPERIMENT

As a technology demonstrator of a space-based gravitational-wave observatory, the LPF mission will place two 2 kg test masses (TM_1 and TM_2) in free fall. The goal is to estimate the residual differential acceleration between TM_1 and TM_2 [6]. To minimize all external forces acting on TM_1 along the x axis, a drag-free control loop [30] has been designed. The coordinates of TM_2 are controlled via electrostatic actuators, and the spacecraft (SC) is controlled by micro-Newton thrusters. More specifically, the main components of the LPF mission are as follows:

(i) The gravitational reference sensor [31]. It consists of the test masses and the vacuum chamber around them. It is mounted with two identical electrostatic actuators to control the 6 degrees of freedom of each test mass and also to apply forces and torques to keep the test masses in free fall.

(ii) The optical metrology system [32] consists of the optical bench, its subsystems, and the processing computer. It performs the sensitive optical measurements of the positions of the test masses along the x axis. For the simplified one-dimensional model version, we consider two interferometer inputs $\vec{i}$ and two outputs $\vec{o}$ of the system. The measured displacements are x_1 and

x_{12}, the distance of TM$_1$ to the SC, and the distance between TM$_1$ and TM$_2$, respectively.

(iii) The current design of the propulsion system for LPF is based on cold gas [2] thrusters. The main function of the thrusters is to maintain the reference test masses in free fall conditions in the measured bandwidth.

(iv) Finally, the drag-free and attitude control system (DFACS) calculates all forces and torques acting on the SC and test masses and computes the commanded forces/response of the system in order to maintain the nominal free fall of TM$_1$.

The LTP is a complicated instrument composed by a plethora of subsystems and coupled control loops. During the mission, there will be dedicated experiments with the aim of characterizing the different noise contributions and couplings. In the following section, the LTP models implemented for system identification experiments are described. We present simulations of the planned system identification experiments and use all techniques described to perform model selection.

A. LTP system dynamics

For the purposes of this work, we consider a simplified version of the LTP operating in the so-called main science mode, and we confine the system dynamics to the one-dimensional case. There are two approaches in modeling the dynamics of the three-body system: The first one, shown in Fig. 1, is to fully describe the dynamics together with the controllers in a state-space format. In this case, it is very convenient to represent the model as a closed loop system, due to closed-loop dynamics and controllers of the instrument. The white boxes represent standalone state-space models (SSMs), altogether *assembled* to form the final LTP experiment [25]. Each submodule has injection $u(t)$ and output $y(t)$ ports and can be represented as

$$\dot{x}(t) = A \times x(t) + B \times u(t),$$
$$y(t) = C \times x(t) + D \times u(t), \tag{15}$$

where $x(t)$ are the *states* of the system, A is the state matrix, B is the input matrix, C is the output matrix, and D is the feedthrough matrix [25]. This form of implementation has numerous advantages, like modularity and flexibility in modeling the LPF mission. For example, the user is able to combine any given SSM module and use a custom noise model for each particular subsystem.

The second approach is to represent the dynamics in an analytical way in the acceleration domain [28], which is
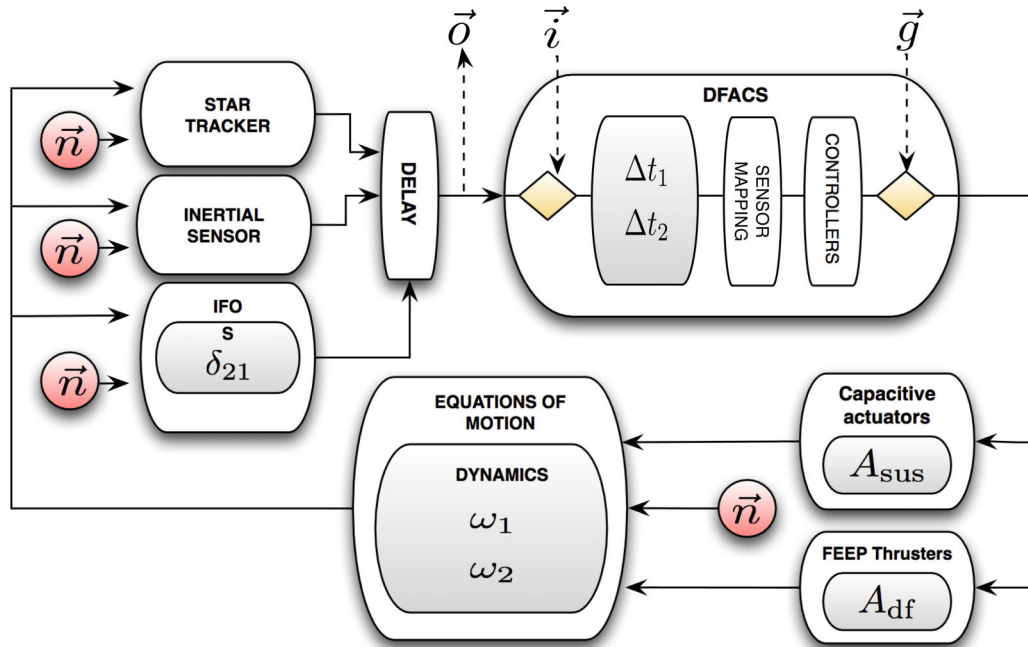

FIG. 1 (color online).   Schematics of a simplified LTP SSM. It is composed by smaller SSMs of the various subsystems, each one a standalone SSM, here represented with white boxes. The rhombi represent the main injection inputs to the system. We use mainly the first injection port, which represents the interferometric inputs to the system, while the second rhombus stands for the capacitance actuators injection ports. The symbol $\vec{n}$ represents noise contributions from various sources, and finally the parameters to fit, for the sake of convenience, are located in the gray boxes inside of the respective SSMs. The interferometer signals $\vec{i}$ are injected to the controllers (DFACS), in which the commanded forces are generated and applied through the capacitance actuators and the thrusters of the spacecraft. In the last two of experiments of Ref. [29] "out-of-loop" forces $\vec{g}$ are applied to the three bodies (TMs and SC) of the system. The resulting movement of the bodies is monitored by the inertial sensor, the star tracker, and the interferometer. Here, the interferometer output is denoted as $\vec{o}$.

described in Sec. III C. A more in-depth investigation on the analysis in the acceleration domain and its practical advantages is to appear soon.

For the purposes of this paper, we simulate the experiments as explained in Refs. [4,5,25,29,33]. Then, the complete system is controlled by optical readouts that measure the positions of the test masses, x_1 and x_{12}, where the x axis is defined by the line joining them. To perform parameter estimation, we inject sinusoidal signals to each interferometric channel alternately or direct forces to the three bodies of the system (TMs and SC) and measure the response of the system. For these types of experiments, we can define the time series arranged as vectors,

$$\vec{o} = \begin{pmatrix} o_{x_1} \\ o_{x_{12}} \end{pmatrix}, \qquad \vec{i} = \begin{pmatrix} i_{x_1} \\ i_{x_{12}} \end{pmatrix}, \qquad \vec{n} = \begin{pmatrix} n_{x_1} \\ n_{x_{12}} \end{pmatrix}, \quad (16)$$

$$\vec{g} = \begin{pmatrix} g_1 - g_{SC} \\ g_2 - g_1 \end{pmatrix}, \qquad (17)$$

where $\vec{o}$ is the measured signals vector, $\vec{i}$ is the interferometer injection signals vector, $\vec{g}$ is the commanded forces signals vector, and $\vec{n}$ is the overall noise for both channels. Here, g_1 and g_2 are the commanded forces applied to TM_1 and TM_2, respectively while g_{SC} is the force applied to the SC. The noise contributions of each LTP subsystem are denoted in Fig. 1 as $\vec{n}$. It can be instrumental or read-out noise, or it may originate from external sources (like solar radiation).

For the sake of simplicity, we use the one-dimensional version of the models. The set of parameters that we recover from the system is $\{\omega_1, \omega_2, \Delta t_1, \Delta t_2, A_{df}, A_{sus}, \delta_{21}\}$: ω_1 and ω_2 stand for the electrostatic stiffness of the test masses to the surroundings; Δt_1 and Δt_2 are time delays; A_{df} and A_{sus} the capacitance and thruster gains, respectively; and δ_{21} denotes the cross-coupling between the two interferometric channels.

As described in Refs. [29,33], during the planned mission experiments, sinusoidal signals of different frequencies are going to be injected to the control loop, simulating displacements of the test masses. Besides the LTP dynamics system, the data analysis team has modeled noise sources for the various subsystems of the LPF mission based on theoretical predictions or characterization of each element from on-ground test campaigns. The main noise contributions come from the thrusters, the interferometric readouts, and the capacitance actuators [34]. We can simulate synthetic noise for any particular experiment and perform parameter estimation exercises in order to test the system identification algorithm's readiness. A MCMC search of the parameter space for such a simulated run returns satisfactory results as shown in Table I.

By using the LTPDA machinery, synthetic noise can be generated, and the response of a known system can be

TABLE I.   MCMC parameter estimation results from simulated experiments. The experiments being analyzed here are the two first described in Refs. [4] and [29].

Parameter	Real value	Estimated $\pm\sigma$
ω_1	1.3×10^{-6}	$(-1.2999 \pm 0.0002) \times 10^{-6}$
ω_2	1.9×10^{-6}	$(-1.8999 \pm 0.0002) \times 10^{-6}$
A_{df}	0.82	0.8201 ± 0.00025
A_{sus}	1.08	$1.080004 \pm 3 \times 10^{-6}$
Δt_1	0.2	$0.20020 \pm 5 \times 10^{-5}$
Δt_2	0.2	$0.2003 \pm 2 \times 10^{-4}$
δ_{21}	0.0004	$0.0004001 \pm 1 \times 10^{-7}$

simulated. For the following study cases, the LTP default parameters were always set to the same values given in Table I. It should be noted that there is no reason to have any strong prior belief about the true values of the parameters. Therefore, it is reasonable to assume uniform priors for all the tests.

B. Application to a simplified LTP model

To demonstrate a first application to the LTP, we can investigate a model selection case that was first encountered during a data analysis exercise [33]. These types of exercises are organized by the data analysis team with the aim of testing the developed tools in more realistic scenarios. In this case, the "true" values of the parameters of the system to identify were totally unknown. In that situation, the data analysis team was able to perform parameter estimation, but it was noted that the fit was improved when two extra parameters were introduced. These parameters are the guidance delays, Δt_1 and Δt_2. They are simply time delays to the application of the guidance signals (see Fig. 1) due to operations of the data management unit, which is the onboard computer controlling the LTP experiment. The final robustness of the fit indicates that the new parameters substantially improve the model.

This problem can be better studied by reducing it to a model selection problem and can be tested with synthetic data sets. The simulated data source is a LTP system with the default characteristics given in Table I with the exception of the delays, which were set to $\Delta t_1 = \Delta t_2 = 0$. With this configuration, we assume a true system in which the application signals are applied instantaneously. For our first try out, we injected "fake" interferometric displacements, while for the second investigation, we used the same structure of injections but with a much lower SNR ($\sim$5).

To determine the importance of the two extra parameters, we have to verify which model describes better the particular data set: the seven-parameter model X with parameters $\vec{\theta}_X = \{\omega_1, \omega_2, \Delta t_1, \Delta t_2, A_{df}, A_{sus}, \delta_{21}\}$ or the five-parameter model Y with parameters $\vec{\theta}_Y = \{\omega_1, \omega_2, A_{df}, A_{sus}, \delta_{21}\}$. While both models, X and Y, are capable of explaining the observations, we expect the simpler model Y to be more favorable from a RJMCMC output since the extra
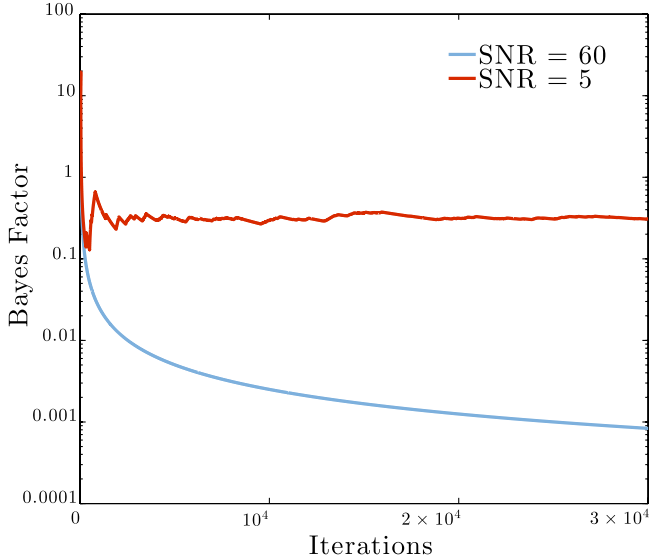
FIG. 2 (color online). First 3×10^4 iteration of the RJMCMC output when comparing a seven- and a five-dimensional LTP model (models X and Y, respectively). Since the models are not competitive when the SNR = 60, the blue line tends asymptotically to zero.

parameters Δt_1 and Δt_2 are not significant for the data. The evolution of the Bayes factor for such an investigation for two different levels of SNR is shown in Fig. 2.

The results we obtain for all the approximations verify that the simpler model Y is much more probable than the more complicated model X. For the particular experiments proposed in Ref. [29], in which the SNR is high, the Bayes factor computed tends to zero. In fact, for the case of the RJMCMC method, there is no single iteration inside the more complex model. This changes dramatically depending on the nature of the problem and, of course, as we show in Sec. IV A, on the SNR. In Table II, we present only the low-SNR experiment, for the sake of comparison between the methods. Each method seems to favor the simpler model, but they are not in total agreement between them. This is to be expected, as the SNR of this investigation is very low and the approximations of the evidence become more sensitive. For this particular case of the injections, the models are not competitive, and therefore the resulting estimated Bayes factor is extremely small. For a direct comparison, the RJMCMC algorithm requires more than 10^8–10^9 iterations.

TABLE II. Results for the guidance delay investigation with the low-SNR experiment. See the text for details.

Method	B_{XY}
RJMCMC	0.309
LF	0.124
LM	0.078
SBIC	0.768

C. More realistic applications

During the last years, a series of data analysis exercises has been performed. These so-called operational exercises are organized by the LPF data analysis team with the aim of training the scientists and engineers for the upcoming flight operations. Data generated by the European Space Agency (ESA) simulator are used, and real-life shifts and duties are rehearsed. The ESA simulator for the LPF is an industry-developed software that is considered the most accurate simulator to date. The scientific and organizational results are valuable since they reproduce situations and data analysis challenges that are expected to occur during flight operations.

An outcome of these data challenges was the realization of the necessity to change or manipulate the model of the dynamics of the system shortly after receiving telemetry from the satellite, thus facing once again model selection problems. Expressing the dynamics in the acceleration domain [28], one could write a simplified model for the differential acceleration, like

$$a_{12} = \left[\frac{d^2}{dt^2} + \omega_2^2\right] x_{12} + (\omega_2^2 - \omega_1^2) x_1$$
$$- A F_2 + A F_1, \tag{18}$$

where a_{12} is the differential acceleration and F_1 and F_2 denote the applied forces on the first and second test masses, respectively. The real motion $\vec{x}$ of the test masses can be approximated by the delayed interferometer read outs:

$$\vec{x} = \begin{pmatrix} x_1 \\ x_{12} \end{pmatrix} = \begin{pmatrix} o_1(t - \tau_{\text{ifo}}) \\ o_2(t - \tau_{\text{ifo}}) \end{pmatrix}. \tag{19}$$

The parameters appearing in Eqs. (18) and (19) are the stiffnesses of the two test masses ω_1 and ω_2, the interferometer read out delay τ_{ifo}, and the gains of the capacitance actuators A_{sus} (here represented as A for simplicity). Note that for this first approximation we have assumed identical actuators A for both test masses. The commanded forces g_1 and g_2 are available as telemetry, but the *real* applied forces on the three bodies are to be determined by the measurements and the analysis itself. For example, in the real data stream, there might be additional delays or even filtering of the applied forces coming from the controllers, so in reality a gain A might be proven to be frequency dependent: $A(f)$. This situation will appear in the received telemetry, and we need the means to disentangle those two physical effects in a quantitative way.

For the particular simulated data set, the model was not able to remove all the injected signals, and even for the simple case of Eq. (18), this led to a biased estimate for the parameters. Apart from a simple time delay on the commanded signals, there might be another process that causes a difference between the commanded g_1 and g_2 and the actual applied forces F_1 and F_2. For a first

approximation of such a process, we assume a single real pole filter, filtering the time series of the applied force on the second test mass. For this investigation, we can propose two models in which $AF_i(t) = Ag_i(t - \tau_C)$ and $AF_i = A(\frac{f_o}{f - jf_o})g_i$, respectively. Here, τ_C denotes the actuators time delay. In the end, we can apply the model selection methods to these two models: the first one, X, in which the applied forces are time delayed and the second one, Y, in which the forces are frequency dependent (filtered by a single real pole filter). The calculated Bayes factor between those two models is

$$\mathrm{BF}_{\mathrm{XY}} = \frac{\pi_{\mathrm{X}}(\vec{\mathbf{y}})}{\pi_{\mathrm{Y}}(\vec{\mathbf{y}})} = 9.9478 \times 10^{-11}, \qquad (20)$$

clearly indicating that the most probable process on the forces is the one described by model Y. The real pole was estimated to be $f_o = 1.963 \pm 0.001$ Hz and was confirmed for the complete set of system identification experiments. The estimated acceleration residuals for both models can be seen in Fig. 3.

The same analysis strategy applies, of course, to more complicated versions of the analytical models, in which the number of parameters increases with the more terms of the equation that are added, as happens in the following more realistic case. As already described in Sec. III, the two test masses are controlled with identical actuators. In reality, a small misbalance between the electrostatic actuators surrounding each test mass might be present. If we introduce this "asymmetry" to the system, immediately for the simple case of Eq. (18), we can increase the dimensionality of the model by four parameters:
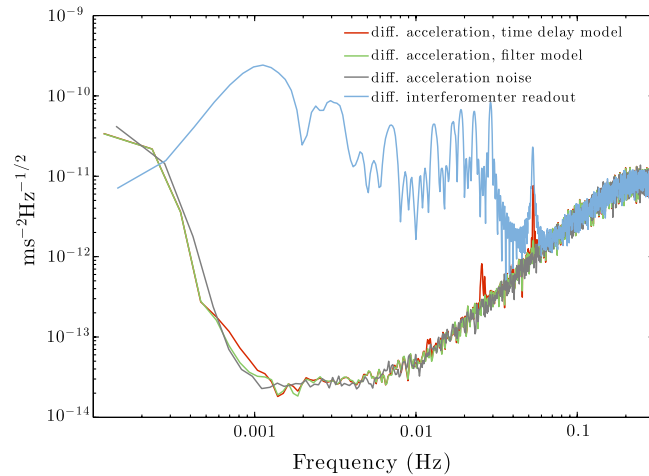
FIG. 3 (color online). Power spectra of simulated differential acceleration between the two test masses. The gray curve represents the reference noise measurement, while the light blue curve is the differential interferometer read out. The value of the computed Bayes factor can be confirmed with the comparison of the equivalent estimated residual acceleration for each model. The differential interferometer read out is also plotted for comparison.
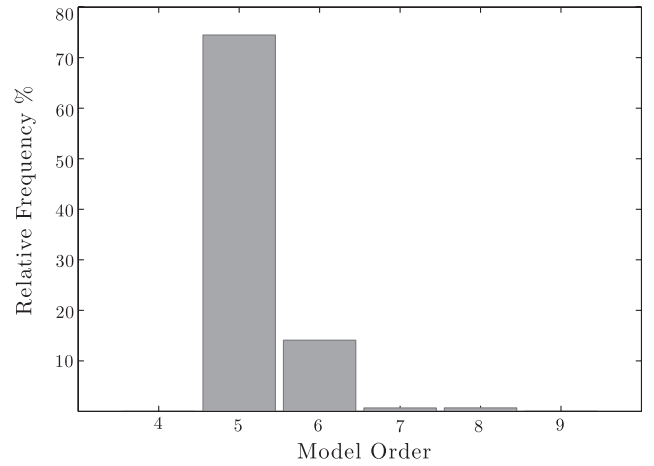
FIG. 4. A RJMCMC run on a set of nested LTP models. There is a clear preference for the five-dimensional model for the given data set. The data were produced with a "perfect" model in which the two respective actuators were identical.

$$a_{12} = \left[\frac{d^2}{dt^2} + \omega_2^2\right]x_{12} + (\omega_2^2 - \omega_1^2)x_1$$
$$- A_2\left(\frac{f_{o2}}{f - jf_{o2}}\right)g_2(t - \tau_{C2})$$
$$+ A_1\left(\frac{f_{o1}}{f - jf_{o1}}\right)g_1(t - \tau_{C1}). \qquad (21)$$

These parameters are gains, delays, and filters that are different for the actuator of each test mass. Theoretically, the highest in dimensions model of Eq. (21) can describe the observations, but the problem to solve appears to be overparametrized. A solution is to generate a set of nested models under the highest in dimensions of Eq. (21) and apply the RJMCMC algorithm. The result of such a run is shown in Fig. 4, and it reveals the most favorable model and consequently the underlying procedure that describe best the physical system. For the particular simulation, we can verify that the five-dimensional model is the best, concluding that no asymmetry in the hardware of the LTP is present. This changed in the following data challenge, in which the Bayes factor between the simple model of Eq. (18) and a six-parameter asymmetric one is greater than 10^6, clearly supporting the correct higher-dimension model in which $A_1 \neq A_2$.

IV. USING THE BAYES FACTOR FOR EXPERIMENT OPTIMIZATION

In this last section, we explore the capability of the implemented model selection framework, not only as a set of tools for data analysis purposes but also as a way to evaluate the efficiency of the experiments we are planning to run in the satellite. As we are going to show, by comparing different models under different input signal conditions, we can safely determine the best range of parameters that define our
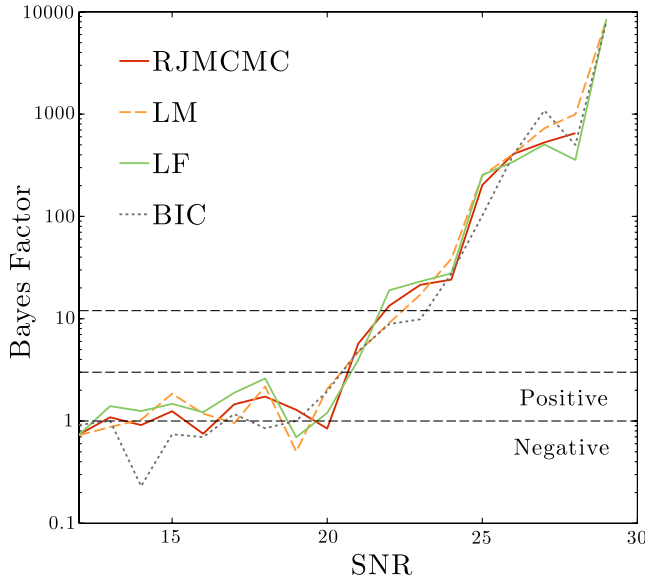
FIG. 5 (color online). The Bayes factor as a function of SNR computed using different methods.

experiments or verify the injection frequencies that maximize the information extracted from the system.

A. Bayes factor as function of the SNR

It has been shown [21,35] that there is a dependence of the Bayes factor output on the SNR regime of the investigations. This, of course, holds true in the case of the LTP, as can be seen in Fig. 5. This figure was created by simulating LTP experiments for each value of the SNR, while the injection signals were single-frequency ($f = 0.01$ Hz) sinusoidal inputs into the system. The LTP models under comparison were quite similar with the exception of a different realization of the response model of the thrusters. It is clear that above the critical value of the SNR $= 21$ the results obtained with the different techniques are consistent and in good agreement. Below that value of the SNR, we cannot make clear decisions about the competing models, as the wrong model is showing preference, or we poorly approximate the Bayes factor.

Although this SNR limit varies, as expected, depending on the type of investigation and model, the current result is already providing an estimation of the required power of the injection signals that we need to consider in the LTP experiment. This information and the method used here will be of particular interest for the design of in-flight experiments for the LPF mission.

B. Bayes factor as function of the injection frequencies

Furthermore, for system identification experiments, as in the case of the LTP, the computed evidence of a model depends on the design of the experiment itself. The information obtained from the system differs depending on the injection frequencies. An interesting study is to

explore in detail this relation. A four-dimensional model (X) and a five-dimensional model (Y) are examined, given different injection frequencies. More precisely, since the difference between the models is the cross-coupling δ_{21} as shown in Fig. 1, which describes the signal leakage from the first to the differential interferometric channel, we examine the Bayes factor given different injection frequencies to the first channel, while keeping constant the injection to the differential channel ($f_{i_{x12}} = 0.2$ Hz). The SNR of this experiment is kept at the "low" value of 28.

The data-generation model is mounted with a perfect interferometer ($\delta_{21} = 0$), and model X is the same as the one used to produce the data, while model Y is the one with the extra parameter δ_{21}. The expected outcome of this exploration is that if the system is more sensitive to the δ_{21} parameter at some particular frequencies we must detect an increase in the Bayes factor that underlines a more clear decision toward the correct model.

In Fig. 6, we can see the corresponding Bayes factor vs the injected frequencies to the first channel. Given the low SNR of the investigation, while model X should be more favorable, a preference for the more complex model Y is shown for a certain set of frequencies. This result is a clear indicator of the set of preferred frequencies that can be injected to the system for its characterization given the current configuration and SNR. Indeed, injections around $f \simeq 0.0006$ and $f \simeq 0.05$ Hz promote the identification of the correct model, while injections at both the high- and low-frequency limits, together with $f \simeq 0.01$ Hz, may induce the analysis into an error. This frequency
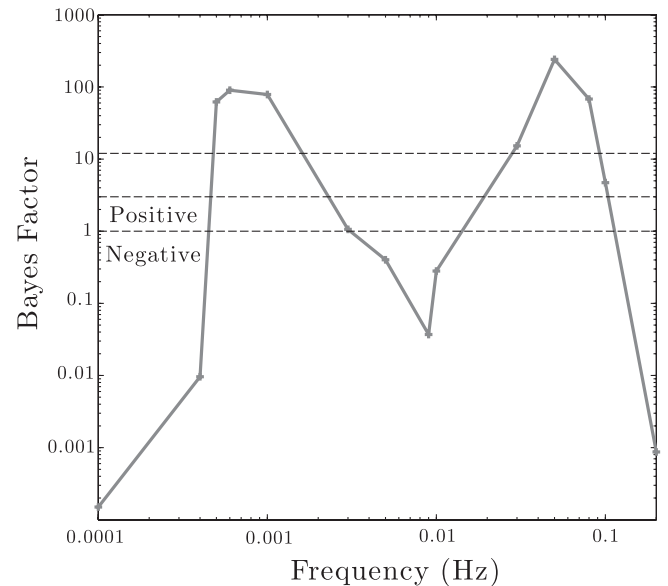

FIG. 6. The Bayes factor as a function of the injection frequencies in the first interferometric channel. The computed evidence of model X is stronger when the sinusoidal injection signals have frequencies around $f_{i_{x_1}} \simeq 0.0006$ Hz and $f_{i_{x_1}} \simeq 0.05$ Hz, respectively.

dependence must be associated to the sensitivity of the experiment to a given parameter, the parameter δ_{21} in this particular case. This observed dependency, when considering a more realistic model, will be of particular interest in the selection of injection signals for the experiments to be run in flight.

V. CONCLUSIONS

We have implemented three different methods to compare competing models of the LTP experiment onboard the LPF mission: the RJMCMC algorithm, the Laplace approximations, and the Bayes information criterion. The results from each method seem to be in agreement, but the output strongly depends on the expected SNR and, of course, on the models under investigation. Considering the LPF mission planned experiments, the SNR is high enough to safely use any of all the available techniques, but probably the most computationally demanding methods will be used for offline analysis to confirm our first computations.

The RJMCMC algorithm (together with the Laplace methods and the Bayes information criterion) employed in this work has been integrated in the LTPDA toolbox as part of the LPF data analysis software. The RJMCMC algorithm is by far the most computationally costly, but at the same time, it is the more suitable one when we compare more than two nested models or we work with inputs with a low SNR. The Laplace–Metropolis and the Laplace–Fisher methods are reliable when we work in the high-SNR regime, but they also require significant computing time, especially when one has to use outlier detection methods to estimate the weighted covariance matrix. On the other hand, the Laplace–Fisher approximation is limited by the use of the Fisher information matrix, which for the case of LTP state-space models is computed numerically.

Moreover, an attempt to associate the output of the aforementioned methods with the actual system identification experiment has been made. We have used different experiment setups to demonstrate that the Bayes factor depends not only on the SNR but also on the injection frequencies to the system.

The developed algorithms were successfully applied to model selection problems for the LPF data analysis for the first time. Two different cases of LTP model selection problems have been investigated over data sets that were produced by both the LTPDA and the ESA simulator. For the first case, we have considered an easy case of five- and seven-dimensional state-space models, in which the importance of the extra two parameters was examined. These two extra parameters are time delays caused by the LPF hardware, and they can be characterized as essential parameters of the model. For the second case, we explored the most suitable dimensionality of analytic models. There, the simplest model that described efficiently the observations was recovered, excluding the more complicated ones that caused overfitting issues. This type of analysis is expected to be performed during operations due to the broad spectrum of possible applications, like identifying external disturbances that result in forces applied to the three-body system.

ACKNOWLEDGMENTS

---

[1] F. Antonucci *et al.*, Classical Quantum Gravity **29,** 124014 (2012).

[2] P. McNamara *et al.*, in *Proceedings of the 9th LISA Symposium, 2012, Bibliothéque Nationale de France, Paris*, edited by G. Auger, P. Binétruy, and E. Plagnol, ASP Conference Series Vol. **467** (Astronomical Society of the Pacific, San Francisco, 2013), p. 5.

[3] P. A. Seoane *et al.* (eLISA Collaboration), arXiv:1305.5720.

[4] M. Nofrarias, C. Röver, M. Hewitson, a. Monsky, G. Heinzel, K. Danzmann, L. Ferraioli, M. Hueller, and S. Vitale, Phys. Rev. D **82,** 1 (2010).

[5] G. Congedo, L. Ferraioli, M. Hueller, F. De Marchi, S. Vitale, M. Armano, M. Hewitson, and M. Nofrarias, Phys. Rev. D **85,** 122004 (2012).

[6] F. Antonucci *et al.*, Classical Quantum Gravity **28,** 094006 (2011).

[7] F. G. Cervantes *et al.*, in *Proceedings of the 9th LISA Symposium, 2012, Bibliothéque Nationale de France, Paris*, edited by G. Auger, P. Binétruy, and E. Plagnol, ASP Conference Series Vol. **467** (Astronomical Society of the Pacific, San Francisco, 2013), p. 141.

[8] H. Audley *et al.*, Classical Quantum Gravity **28,** 094003 (2011).

[9] R. E. Kass and A. E. Raftery, J. Am. Stat. Assoc. **90,** 773 (1995).

[10] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms* (Cambridge University Press, Cambridge, England, 2003).

[11] M. Hewitson *et al.*, Classical Quantum Gravity **26,** 094003 (2009).

[12] LTPDA Matlab toolbox (2005), http://www.lisa.aei-hannover.de/ltpda/.

[13] P. J. Green, Biometrika **82**, 711 (1995).

[14] H. F. Lopes and M. West, Statistica Sinica **14**, 41 (2004).

[15] P. Dellaportas, J. J. Forster, and I. Ntzoufras, Stat. Comput. **12**, 27 (2002).

[16] A. Stroeer and J. Veitch, Phys. Rev. D **80**, 064032 (2009).

[17] R. Umstätter, N. Christensen, M. Hendry, R. Meyer, V. Simha, J. Veitch, S. Vigeland, and G. Woan, Phys. Rev. D **72**, 022001 (2005).

[18] S. J. Godsill, J. Comput. Graph. Stat. **10**, 230 (2001).

[19] T. B. Littenberg and N. J. Cornish, Phys. Rev. D **80**, 063007 (2009).

[20] F. Bartolucci, L. Scaccia, and A. Mira, Biometrika **93**, 41 (2006).

[21] N. Cornish and T. Littenberg, Phys. Rev. D **76**, 1 (2007).

[22] M. Vallisneri, Phys. Rev. D **77**, 1 (2008).

[23] P. J. Rousseeuw and M. Leroy Annick, *Robust Regression and Outlier Detection* (Wiley, New York, 1987).

[24] L. Barack and C. Cutler, Phys. Rev. D **69**, 082005 (2004).

[25] M. Nofrarias *et al.*, *State Space Modelling and Data Analysis Exercises in LISA Pathfinder*, edited by G. Auger, P Binetruy, and E. Plagnol, Astronomical Society of t he Pacific Conference Series Vol. **467** (2013), p. 161,

http://www.aspbooks.org/a/volumes/article_details/?paper_id=35130.

[26] M. Hewitson, M. Diaz-Aguilo, and A. Grynagier, Tech. Rep. S2-AEI-TN-3069, 2011 (unpublished).

[27] A. Grenadier and M. Weyrich, M.A. thesis, Institute of Flight Mechanics and Control, University of Stuttgart, Stuttgart, 2008.

[28] G. Congedo, R. Dolesi, M. Hueller, S. Vitale, and W. J. Weber, Phys. Rev. D **88**, 082003 (2013).

[29] S. Vitale, Tech. Rep. S2-UTN-TN-3045, 2007 (unpublished).

[30] W. Fichter, P. Gath, S. Vitale, and D. Bortoluzzi, Classical Quantum Gravity **22**, S139 (2005).

[31] R. Dolesi *et al.*, Classical Quantum Gravity **20**, S99 (2003).

[32] G. Heinzel *et al.*, Classical Quantum Gravity **22**, S149 (2005).

[33] M. Nofrarias, L. Ferraioli, G. Congedo, M. Hueller, M. Armano, M. Diaz-Aguiló, A. Grynagier, M. Hewitson, and S. Vitale, J. Phys. Conf. Ser. **363**, 012053 (2012).

[34] F. Antonucci *et al.*, Classical Quantum Gravity **29**, 124014 (2012).

[35] J. Veitch and A. Vecchio, Phys. Rev. D **78**, 022001 (2008).