

LISA Framework for Enhancing Gravitational Wave Signal Extraction Techniques

David E. Thompson¹ and Rajkumar Thirumalainambi²

¹*NASA Ames Research Center, MS 269-1, Moffett Field, CA 94035-1000*

²*QSS Group at NASA Ames, MS 269-2, Moffett Field, CA 94035-1000*

Abstract. This paper describes the development of a Framework for benchmarking and comparing signal-extraction and noise-interference-removal methods that are applicable to interferometric Gravitational Wave detector systems. The primary use is towards comparing signal and noise extraction techniques at LISA frequencies from multiple (possibly confused) gravitational wave sources. The Framework includes extensive hybrid learning/classification algorithms, as well as post-processing regularization methods, and is based on a unique plug-and-play (component) architecture. Published methods for signal extraction and interference removal at LISA frequencies are being encoded, as well as multiple source noise models, so that the stiffness of GW Sensitivity Space can be explored under each combination of methods. Furthermore, synthetic datasets and source models can be created and imported into the Framework, and specific degraded numerical experiments can be run to test the flexibility of the analysis methods. The Framework also supports use of full current LISA Testbeds, Synthetic data systems, and Simulators already in existence through plug-ins and wrappers, thus preserving those legacy codes and systems in tact. Because of the component-based architecture, all selected procedures can be registered or de-registered at run-time, and are completely reusable, reconfigurable, and modular.

Keywords: component-based exploration framework; LISA signal and noise identification methods; MLDC; intelligent systems classification techniques; regularization of time-series patterns

PACS: 95.75.-z; 95.55.Ym; 95.85.Sz; 95.75.Wx; 95.30.Sf.

INTRODUCTION

A team at NASA Ames is developing a Framework for benchmarking and comparing signal-extraction and noise-interference-removal methods that are applicable to interferometric Gravitational Wave (GW) detector systems. The main target is developing new concepts of science data analysis and statistical techniques useful for the extraction of signals from the proposed LISA spacecraft constellation TDI variables. This Framework will be open to the LISA research community for data analysis methods evaluation, exploration and comparison. In particular, it will support the Mock LISA Data Challenge (MLDC) (see <http://astrogravs.nasa.gov/docs/mldc/>, and [1,2,3]) by providing new search, feature discrimination, filtering, pattern classification, and regularization tools and methods to the MLDC participants and the scientific community generally. The Framework has applicability to signal extraction techniques-exploration from multiple gravitational wave sources, from in-spiraling compact binaries and supermassive black hole binaries, as well as from a variety of astrophysical sources against a cosmic gravitational radiation background. It could also support analysis of more cataclysmic events such as supernovae core collapse, observable at projected LIGO frequency ranges.

Coherent signals of GW within the TDI data variables require exploratory methods of isolation and noise identification. These methods ideally should include:

Data Stream Conditioning

- GW Filtering methods including Probabilistic Principal Component Analysis [or ICA] [4,5]; Kalman filtering; and statistical learning and pattern recognition algorithms [6];
- Kernel and/or Support Vector Machine Methods for time-series pattern recognition and feature classification [7,8]; a classified indexing of [possibly overlapping] parts of the time-series signal;

Signal Processing and Analysis in Feature Space

- specific appropriate Digital Fourier Transform methods [9], as well as Wavelet [10] and Radon Transform [11] analysis in Feature Space;
- coherent line removal through harmonics mappings [12];

Post-Processing and Evaluation/Estimation

- a variety of physics-based regularization methods, statistical learning, Kernel methods [7,8], and other knowledge-preserving smoothing techniques [13,14];
- function approximation or response surface construction methods that are similar mathematically to hybrid artificial neural networks [15];
- Hidden Markov Models, Markov Chains linked with Monte Carlo methods, and hybridized versions of these using Genetic Algorithms [16-20];
- parametric and non-parametric Bayesian estimation tools [6,16,17];

The LISA Framework project is adding many of the above additional pre-processing and post-processing features to the current types of analyses, both for individual research projects on noise identification and removal, and for larger test bed or simulation efforts for signal extraction. The LISA Framework is based on a unique plug-and-play architecture. This means that legacy codes that have been developed by others in the LISA research community can be used *as is* within this Framework. The Framework wraps the specific user codes (whether in C, C++, or Fortran) and thereby allows for transactions among the Framework's analysis, comparison, and visualization components. In addition, the LISA Framework should be able to support, through plug-ins, other test beds and simulators such as the TLA/LISA Testbed [21], Synthetic LISA [22,23], and LISA Simulator [24-28], while not necessarily being part of the Mission Critical Path.

Multiple similar types of Frameworks and Science Exploration Environments have been developed at NASA Ames, with several still in use. The development of LISA Framework actually represents a second instantiation of a project currently underway at NASA Ames, the Mission Control Technologies (MCT) Framework. The heritage and lessons-learned across multiple development methods, architectures, and interoperability of components provides a leg-up on scale feasibility for the LISA Framework project. The MCT Framework allows dynamic graphical composition of data and process representations, and it also facilitates interoperability between various components and with external applications. The MCT Framework allows registering and de-registering of analysis methods at runtime without restarting the whole project.

The LISA Framework is neither a testbed nor a simulator, *per se*. It is operationally defined by a given instance of its use. Both the architecture and use

scenarios are described below. Succinctly, the LISA Framework provides the new capabilities:

1. Capability of carrying out Sensitivity Analysis across multiple test beds, diverse FFT modules, and various data analysis methods;
2. LISA Framework is highly scalable and not restricted by number of plug-ins to be used by the scientist. In fact, current external codes and full test bed suites are automatically wrapped and integrated without interference with these legacy codes. The LISA Framework API (JAVA) is designed in such a way to be very generic and adaptive, hence independent of operating system;
3. LISA Framework is a component-based, plug-and-play architecture; hence, it is very easy to (re)configure, restart, and manage at runtime, for any of various scenarios for sensitivity analysis;
4. The flow of any scientist-driven use scenario within LISA framework is semantically rich and logically validated; it serves as a workbench for LISA science analysis and benchmarking of various algorithms and models.
5. A common messaging bus [Java/JMS], via plug-ins, is used for between-component communication as well as external communication with other tools and test beds, all within the LISA Framework.

DESCRIPTION AND USE OF THE LISA FRAMEWORK

The primary components of the LISA Framework are: the Framework and UI itself; the data analysis plug-ins; the graphics and visualization plug-ins; the sensitivity analysis component and plug-ins; and the metrics analysis plug-ins. The data retrieval and IO components facilitate messaging and other utilities for smooth operation and interface. *Yet to be customized* from MCT for the LISA Framework instance are the sensitivity estimator component and the specific data analysis plug-ins. LISA Framework is displayed in plug-and-play component architecture format in **Figure 1**.

To better understand this Framework, consider a typical example of how the Framework Architecture components can be organized to instantiate a particular analysis evaluation, as part of a greater exploration of sensitivity analysis. **Figure 2** provides an example process flow representation of such an exploration and analysis procedure.

A full time-series dataset represents the fundamental noise and signal embedded within TDI variables. The TDI variables are the time-delayed combinations of the basic interferometric measurements [1,2] that compare the frequencies (or phase-shifts) of the two lasers incoming from the 3x2 paired spacecraft. The Doppler measurements bear the imprint of the instrumental noises, astrophysical noise, and of the GW signals. In **Figure 2**, the raw source signals [simulated or experimental] are combined with one or a composite of noise models and with expected sensitivity data for the particular LISA configuration [probably derived from LISA Simulator experiments]. All these signals are mapped into one of a variety of classification schemes [pattern matching techniques], and in this example are put into several

classes. Classes may be intermediate in S/N, overlapping in time, or more pathological. A description of the basis for classification is provided back to the user.

LISA Framework Architecture

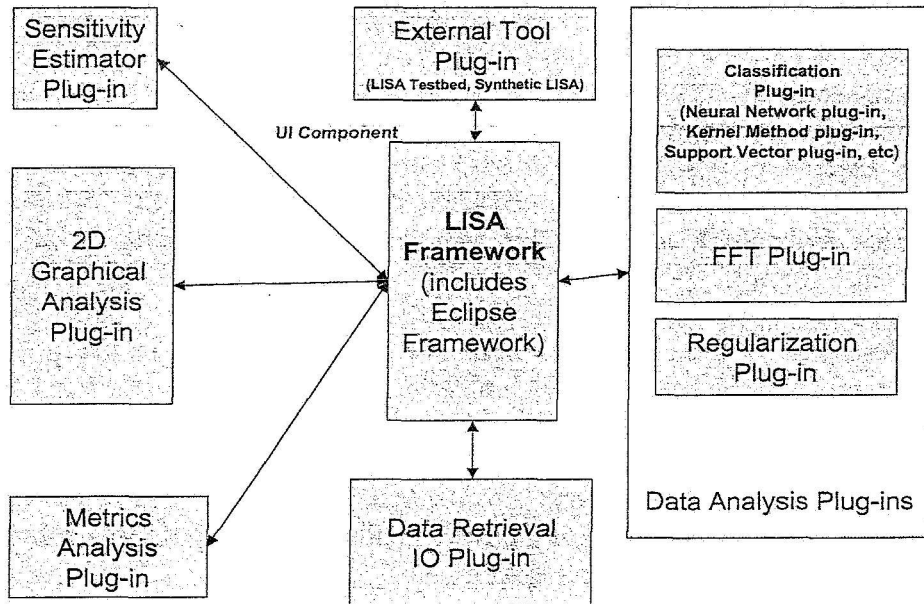


FIGURE 1. A schematic representation of the LISA Framework Architecture from which specific user scenario “projects” can be implemented according to the various “plug-ins” selected to instantiate an analysis process.

Example Process Flow

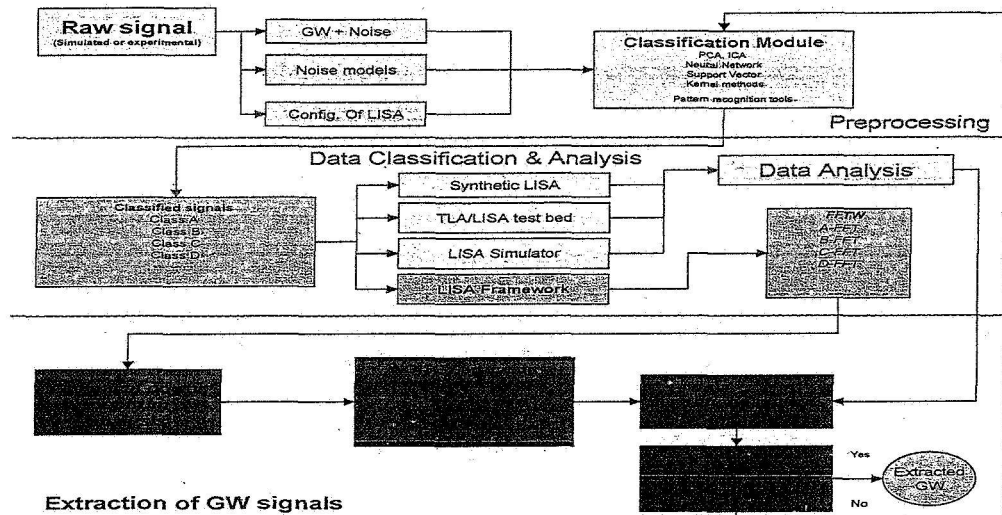


FIGURE 2. An example representation of a process flow for analysis using LISA Framework, while also accommodating additional LISA simulators and their data analysis features for inter-comparison.

Simple or complicated, and multiple types of FFTs can be selected to operate on the separated noise and signal classes instead of applying one full-blown FFT technique to the entire time-series. This classification allows a lengthy time-series to be divided into manageable, possibly overlapping, time-series and thereby achieve justifiable mapping of analysis to signal. Notice, however, that *classification does not alter the input time-series signal itself*; it only indexes parts of the signal according to expectations. The LISA Framework supports not only new classification algorithms, but still provides segmentation algorithms based on prior information similar to the work of Umstätter, et al [17]. After *classification* and *pre-conditioning*, a user can launch the new data streams into the TLA Testbed or into Synthetic LISA and their standard data analysis methods will proceed within those methods. *Or* the user can continue to build analysis with the components in LISA Framework. As seen in **Figure 2**, LISA Framework analysis then proceeds to call the appropriate FFTW modules for each section of the partitioned/classified data stream. This would typically be followed by analysis of the power spectrum, in frequency or phase feature space. The classification pedigree of the signals would be appended, so that possible anticipated or undiscovered noise sources can be identified or analyzed further. *Post-processing*, smoothing, and Tikhonov Regularization all aid in this discovery. A decision is made as to what knowledge has been gained, and how to implement the next cycle of the numerical experiment. New classification tools and FFTs can be selected, and then multiple sensitivity trials can be built up and mapped into a general sensitivity space for comparison and evaluation.

CONCLUSION

The unique LISA Framework offers the following critical capabilities. It supports analysis of gravitational wave sources and interference in simulated and real-time. It is adaptable to any data analysis algorithms (as components) within LISA experimentation, and it is modular and reconfigurable to analyze a variety of mission data types. It allows users to extract noise from signals across new partitioned (classified) regions of the frequency spectrum. It can integrate any of several available Testbeds and Simulators, through wrapped plug-ins at run-time, providing enhanced sensitivity analysis and exploration of sensitivity-space, while preserving those legacy codes as is. The Framework is configured for a default template analysis and can be modified during run-time to analyze performance of various algorithms as well as model complexity. And finally, LISA Framework will provide better understanding of fundamental gravitational physics through new intuition development and methods exploration in preparation for the LISA spacecraft constellation launch.

REFERENCES

- [1]. Armstrong, J.W., F.B. Estabrook, and M. Tinto (2003) Time delay interferometry. CQG, 20, S283-289.
- [2]. Tinto, M., F.B. Estabrook, and J.W. Armstrong (2002) Time-delay interferometry for LISA. PRD, 65, p.082003-1 – 082003-12.

- [3]. Tinto, M., F.B. Estabrook, and J.W. Armstrong (2004) Time-delay interferometry with moving spacecraft arrays. PRD, **69**, p.082001-1 – 082001-10.
- [4]. Sathyaprakash, B.S. (1998) Filtering gravitational waves from supermassive black hole binaries. p.110-119; In Folkner, W.M. ed., Second International LISA Symposium [LISA 2]. AIP Conference Proceedings no. 456. Amer. Institute of Physics. 243p.
- [5]. Chui, C.K. and G. Chen (1999) Kalman Filtering with Real-Time Applications. 3rd edit., Springer. 229p.
- [6]. Berger, J.O. (1985) Statistical Decision Theory and Bayesian Analysis. 2nd edition; Springer. 617p.
- [7]. Cristianini, N. and J. Shawe-Taylor (2000) An Introduction to Support Vector Machines and other Kernel-Based Learning Methods. Cambridge Univ. Press. 189p.
- [8]. Shawe-Taylor, J. and N. Cristianini (2004) Kernel Methods for Pattern Analysis. Cambridge Univ. Press. 492p.
- [9]. Frigo, M. and S.G. Johnson (2005) The design and implementation of FFTW3. Proc IEEE, v.93, no.2, p.216-231.
- [10]. Hayama, K. (2005) Wavelet method to reduce binary confusion noise. CQG, **22**, S527-S530.
- [11]. Helgason, S. (1999) The Radon Transform, 2nd ed. Series, Progress in Math, vol. 5. Birkhauser, Boston and New York. 188p.
- [12]. Sintes, A.M. and B.F. Shutz (1998) Removal of interference from external coherent signals. In Folkner, ed., *ibid.* p.135-140.
- [13]. Hastie, T., R. Tibshiriani, and J. Friedman (2001) The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer. 533p.
- [14]. Wahba, G (1990) Spline Models for Observational Data. SIAM. CBMS-NSF Conf. Series in Appl. Math., v.59. 169p.
- [15]. Duda, R.O., P.E. Hart, and D.G. Stork (2001) Pattern Classification. 2nd edition; John Wiley & Sons, Inc. 654p.
- [16]. Umstätter, R., N. Christensen, M. Hendry, R. Meyer, V. Simha, J. Veitch, S. Vigeland, and G. Woan (2005a) Bayesian modeling of source confusion in LISA data. PRD, **72**, 022001-1 – 022001-15.
- [17]. Umstätter, R., N. Christensen, M. Hendry, R. Meyer, V. Simha, J. Veitch, S. Vigeland, and G. Woan (2005b) LISA source confusion: identification and characterization of signals. CQG, **22**, S901-S911.
- [18]. Norris, J.R. (1997) Markov Chains. Cambridge Univ. Press. 237p.
- [19]. Cornish, N.J., and J. Crowder (2005) LISA data analysis using Markov chain Monte Carlo methods. PRD, **72**, p.043005-1 – 043005-15.
- [20]. Crowder, J., N.J. Cornish, and J.L. Reddinger (2006) LISA data-analysis using genetic algorithms. PRD, **73**, p.063011-1 – 063011-11.
- [21]. Finn, L.S., M.J. Benaquista, S.L. Larson, and L.J. Rubbo (2006) The testbed for LISA Analysis project. WhitePaper; and arxiv.org/abs/gr-qc/0602019, v.1, 5 Feb 2006. 5p.
- [22]. Vallisneri, M. (2005) Synthetic LISA: simulation time delay interferometry in a model LISA. WhitePaper, and arxiv.org/abs/gr-qc/0407102, v.3, 5 Jan 2005. 18p.
- [23]. Shaddock, D.A., B. Ware, R.E. Spero, and M. Vallisneri (2004) Postprocessed time-delay interferometry for LISA. PRD(rapid), **70**, 081101-1 – 081101-5.
- [24]. Cornish, N.J. and L.J. Rubbo (2003) LISA response function. PRD, **67**, 022001-1 – 022001-7
- [25]. Larson, S.L., W.A. Hiscock, and R.W. Hellings (2000) Sensitivity curves for spaceborne gravitational wave interferometers. PRD, **62**, p.062001-1 – 062001-10.
- [26]. Larson, S.L., R.W. Hellings, and W.A. Hiscock (2002) Unequal arm space-borne gravitational wave detectors. PRD, **66**, p.062001 – 062001-7.
- [27]. Hellings, R. (2001) Elimination of clock jitter noise in spaceborne laser interferometers. PRD, **64**, p.022002-1 – 022002-12.
- [28]. Cornish, N.J. and S.L. Larson (2003) LISA data analysis: Doppler demodulation. CQG, **20**, S163-S170.