Understanding interferometric drag-free sensors in space using intelligent data analysis tools

Von der Fakultät für Mathematik und Physik der
Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades

Doktor der Naturwissenschaften
– Dr.rer.nat –

genehmigte Dissertation von

M.Sc. Anneke Monsky

geboren am 13. Dezember 1981
in Celle, Deutschland

2010

Für Papa.

Abstract

LISA, the Laser Interferometer Space Antenna, is a planned gravitational wave observatory in space. It is a joint ESA and NASA mission planned to be launched in 2018. Three spacecraft form an equilateral triangle with the armlength of 5 million kilometres, building a Michelson interferometer.

The technology demonstrator for LISA, LISA Pathfinder, itself a space mission to be launched in 2012. The main measurement concept of LISA is based on the ability to put a test mass into free fall and account for all residual forces that may act on it. LISA Pathfinder aims to verify this concept. An interferometer measures the distance between two free-falling test masses with pico-metre accuracy in the milli-Hertz range.

The LISA Pathfinder mission time is strictly limited. Experiments must be prepared in detail prior to the mission in order to maximise the mission's science output. In general the on-board experiments are not independent from each other, but the result of one may affect following experiments.

This situation leads to special demands on the data analysis activities, as only a robust and carefully tested tool shall be used for the quasi-online data analysis. As such the software tool 'LTPDA', which enables scientists to carry out the complete data analysis of the mission, was developed by the data analysis team as part of this thesis.

Every analysis carried out by using LTPDA can be traced back to the raw data used. With LTPDA it will be simple to pass analyses carried out for LISA Pathfinder to the mission time of LISA.

All results concerning the LISA Pathfinder mission presented in this thesis were carried out exclusively within LTPDA. In fact, the software tool is used for the complete data analysis of the LISA Pathfinder experiments at AEI.

In this thesis important laboratory experiments on the Engineering Model of the LISA Pathfinder optical bench are presented. These experiments are part of the experiment master plan carried out in preparation of the mission. The focus was on the development of data analysis tools in LTPDA. A number of significant noise sources in LISA Pathfinder are discussed in this thesis. These include: the angular motion of the LISA Pathfinder test masses coupling into the measurement, a special non-linear technical noise source, called sideband induced noise as well as the laser amplitude noise applying an additional force to the test masses. The work presented here highlights ready-to-use methods for analysing these noises.

The LISA Pathfinder experiment is simulated within so-called mock data challenges (MDCs). During the evolution of this thesis two of these (MDCs) have been carried out successfully. During the process of MDC1 one of the key components of the data analysis for LISA Pathfinder, the conversion of interferometer readout to test mass acceleration has been implemented (for the ideal, one-dimensional

model). The key point of MDC2 was to investigate a variety of parameter estimation methods and in this thesis the linear approach is presented.

The complexity of the simulated experiments is increased stepwise, such that the MDCs will become more realistic and it will be possible to test all mission experiments using enhanced MDC models. These MDCs have been proven to be a good instrument for driving the development of the analysis tools as well as forcing a good understanding of the experiment on all scientists involved.

The final project presented is the injection of non-Gaussian noise into LISACode. It is a first attempt of injecting a noise source measured in LISA Pathfinder into a LISA data simulator. This project is meant to serve as a catalyst for a more intense collaboration between the data analysis efforts done in LISA Pathfinder and those of LISA. The focus of the data analysis for the two missions is very different since LISA Pathfinder will be insensitive to gravitational wave signals. From the technical point of view, however, it is important to transfer the experiences made in LISA Pathfinder to LISA. This includes the sideband induced noise (SIN) presented in this thesis, which has been observed in the LISA Pathfinder interferometer readout and which has been well studied in the course of LISA Pathfinder data analysis. Further investigations on such realistic noise sources will improve the quality of the mock LISA data challenges (MLDCs) accomplished in preparation of the mission.

Zusammenfassung

LISA und LISA Pathfinder sind geplante Weltraummissionen der ESA und NASA. LISA (Laser Interferometer Space Antenna) ist ein geplanter Gravitationswellendetektor im All. Da es nicht möglich ist alle Schlüsseltechnologien für LISA in Laboren auf der Erde zu testen, wurde die Testmission LISA Pathfinder entwickelt.

Das Messprinzip, das in LISA Anwendung findet, basiert darauf den Abstand zweier frei-fallender Testmassen im Bereich von einigen milli-Hertz mit Picometer Genauigkeit interferometrisch zu vermessen. Eines der Hauptziele von LISA Pathfinder ist es, dieses Experiment zu verifizieren.

Die Missionszeit für LISA Pathfinder ist sehr begrenzt. Um den wissenschaftlichen Ertrag der Mission zu erhöhen, ist es daher wichtig, alle durchzuführenden Experimente im Vorfeld auszuarbeiten. Im Allgemeinen sind die jeweiligen Experimente nicht unabhängig voneinander. Das heißt, die Ergebnisse einer Messung kann die Wahl der Parameter für folgende Experimente beeinflussen.

Dieser Umstand stellt besondere Anforderungen an die Software zur Analyse der Missionsdaten. Für die quasi-online Analyse der Daten soll ein robustes und verlässliches Software-Tool verwendet werden. Dieses Tool stellt LTPDA dar. Die Entwicklung dauert an und fand maßgeblich im Rahmen dieser Arbeit statt. LTPDA ist ein gemeinschaftliches Projekt aller Wissenschaftler, die an der Datenanalyse für LISA Pathfinder beteiligt sind.

Hierbei ermöglicht die Verwendung von LTPDA, die Rückverfolgung aller Ergebnisse einer jeden Analyse bis hin zu den verwendeten Rohdaten. Mit LTPDA wird es demnach beträchtlich einfacher, Ergebnisse und Analysen LISA Pathfinder betreffend, bis in die Missionszeit von LISA weiterzureichen.

Tatsächlich wurden, alle Analysen von LISA Pathfinder Experimenten, die in dieser Arbeit gezeigt werden, mit LTPDA erstellt. Die Datenanalyse der Labordaten für LISA Pathfinder am AEI wird bereits vollständig mit LTPDA durchgeführt.

In dieser Arbeit werden wichtige Experimente am Engineering Model der optischen Bank von LISA Pathfinder präsentiert. Diese Experimente sind Teil des 'Experiment Master Plan' (EMP) und werden zur Vorbereitung der Mission und zum Test von Hardwarekomponenten durchgeführt. Der Fokus wird hierbei auf den erstellten Analyse-Tools liegen, die in dieser Form auch auf Missionsdaten von LISA pathfinder anwendbar sind. Die Beschreibung und Analyse wesentlicher Rauschquellen für LISA Pathfinder sind Bestandteil dieser Arbeit. Dies umfasst das Koppeln der Verkippung der Testmassen in die Abstandsmessug, eine besondere, technische, nicht-lineare Rauschquelle, das sogenannte *sideband induced noise* sowie das Laser-Amplitudenrauschen, welches eine zusätzliche Kraft auf die Testmassen ausübt. Herausgestellt werden hier besonders die entwickelten Methoden zur Analyse der Rauschquellen.

Das LISA Pathfinder Experiment wird durch sogenannte 'mock data challenges'

(MDCs) simuliert. Während der Entstehung dieser Arbeit wurden zwei dieser MDCs erfolgreich durchgeführt. In MDC1 wurden die Tools für eines der wichtigsten Experimente implementiert: das Umwandeln von Interferometerdaten in Testmassen-Beschleunigung (idealisiertes, eindimensionales Modell). Das Hauptaugenmerk in MDC2 waren verschiedene Methode zur Parameterschätzung. In dieser Arbeit wird die Methode der Linearisierung behandelt.

Die Komplexität der simulierten Experimente und die hierfür verwendeten Modelle werden stufenweise erhöht, sodass die Simulationen dem realen LISA Pathfinder Experiment zunehmend ähneln. Mit solch repräsentativen Modellen können dann alle Experimente des EMP simuliert und entsprechende Methoden der Datenanalyse entwickelt werden. MDCs haben sich als ein hervorragendes Instrument erwiesen die Entwicklung der Analyse-Software voranzutreiben. Ein wertvoller Nebeneffekt ist die Schulung der an den Simulationen für LISA Pathfinder beteiligten Wissenschaftler.

Abschließend, wird ein Projekt zur Injektion einer nicht-Gaussschen Rauschquelle in eine Simulation von LISA vorgestellt. Es ist der erste Ansatz eine in LISA Pathfinder beobachtete Rauschquelle in die Datenanalyse von LISA zu integrieren. Das vorgestellte Projekt soll als Katalysator für die Zusammenarbeit der sehr unterschiedlichen Lager der Datenanalyse für LISA und LISA Pathfinder dienen. Der große Unterschied rührt vor allem daher, dass LISA Pathfinder nicht-sensitiv für die Detektion von Gravitationswellen ist. Nichtsdestotrotz ist der eher technische Fokus der Datenanalyse für LISA Pathfinder in gleichem Maße für LISA von Bedeutung, und die gewonnenen Erfahrungen sollten für die zukünftige LISA Datenanalyse nutzbar gemacht werden. Dies schließt die besondere technische Rauschquelle namens 'Sideband Induced Noise' (SIN) ein, die sich als wichtiger Rauschbeitrag in der LISA Pathfinder Interferometrie herauskristallisiert hat. Diese Rauschquelle wurde im Rahmen dieser Arbeit detailliert analysiert. Das Fortführen der Einbindung realer Rauschquellen in die Simulationen von LISA wird die Qualität der zur Vorbereitung der Mission durchgeführten 'mock LISA data challenges' weiter verbessern.

Contents

List of Figures

List of Tables

List of acronyms

AEI	Albert Einstein Institut Hannover
AO	Analysis Object
AOM	acousto-optic modulators
ASD	amplitude spectral density
cov	covariance
DFACS	drag-free and attitude control system
DMU	data management unit
DWS	differential wavefront sensing
EM	engineering model
EMP	experiment master plan
ESA	European Space Agency
FFT	Fast Fourier Transform
FIR	finite impulse response
FM	flight model
FPGA	Field Programmable Gate Array
GW	gravitational waves
het	heterodyne
IIR	infinite impulse response
IFO	interferometer
IRS	inertial reference sensor
IS	inertial sensor
LAN	laser amplitude noise
LISA	Laser Interferometer Space Antenna
LMU	laser modulator unit
LPF	LISA Pathfinder
LSD	linear spectral density
LTP	LISA Technology Package
LTPDA	LTP data analysis tool
MB	measurement band
MBW	measurement bandwidth
MDC	mock data challenge
OB	optical bench
OBC	onboard computer
OMS	optical metrology system
OPD	optical pathlength difference
plist	parameter list
PSD	power spectral density
PZT	piezo-electric transducer

QPD	quadrant photodetector
RF	radio frequency
RIN	relative intensity noise
rms	root mean square
SBDFT	single-bin discrete Fourier Transform
S/C	spacecraft
std	standard deviation
SIN	sideband induced noise
SNR	signal-to-noise ratio
ssm	state space model
SVD	singular value decomposition
TDI	time delay interferometry
TF	transfer function
TM	test mass
URD	user requirement document
USO	ultra stable oscillator

1 Introduction

1.1 Gravitational waves

Space and time do not have an independent existence but are joined as *spacetime* and must be considered as measurable observables like any other physical quantity [1]. This is what Einstein's theory of Special Relativity [2] taught us, when published in 1905. Ten years later Einstein's field equation of General Relativity [3] replaced the Newtonian picture of Gravitation by a geometric one: massive bodies cause a curvature of spacetime. Changing mass distributions can cause indentations in spacetime propagating outwards at the speed of light. These indentations are often called *ripples* in spacetime. Figure 1.1 depicts a gravitational wave transmitted by a binary system as distortions in spacetime.
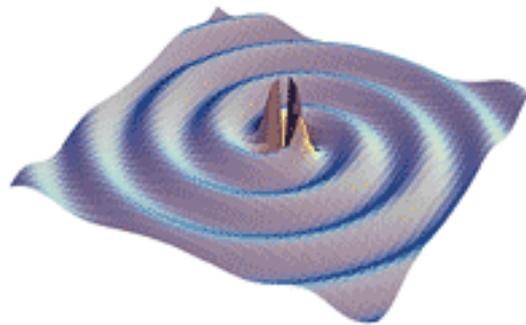
Figure 1.1: *Ripples* in spacetime: an artist's impression of gravitational waves transmitted by a system of two orbiting masses (binary system). Source: NASA

.

In 1918 Einstein published the first calculation of gravitational radiation [4] leading to the field equation that is today well known as the 'quadrupole formula' for gravitational wave (GW) emission. The quadrupole formula shows that gravitational waves arise from accelerated masses just as electromagnetic waves arise from charged particles [1]. The main reason for the difficulty of the observation of gravitational waves lies in the weakness of the interaction. Einstein's field equations predict that very large masses need to be subject to extreme accelerations to produce gravitational waves that are detectible by today's technology. Only astrophysical objects fulfil these requirements and as such the first indirect proof of the existence of gravitational waves has been found by studying binary neutron star systems. In 1975 Hulse and Taylor reported that the orbit of the 'Hulse-Taylor' pulsar 1913+16 is decaying with time, matching with high precision Einstein's equations and that part of the pulsar orbital energy is converted into gravitational radiation [5]. In 1993 they were awarded the Nobel prize in physics for this indirect proof of the existence

of gravitational waves. However, the weakness of the gravitational interaction still prevents gravitational waves from being detected directly.

Once gravitational waves are directly detectable they will provide a completely different way of exploring astrophysical objects. Electromagnetic radiation is strongly interacting with matter, which is the reason why it is so easily measurable. However, it thereby is also likely to be scattered or absorbed before reaching the detectors. In contrast the same reason that makes gravitational waves so difficult to detect, namely their weak interaction with matter, prevents them from being altered significantly during their travel through space. Hence, gravitational wave-astronomy will also, among others, aim to analyse gravitational radiation from the early Universe, when no electromagnetic radiation was present.

The spectrum of gravitational waves differs completely from the one of electromagnetic waves. It is impossible to produce an image of a gravitational wave source. Instead an analogy to sound waves works quite well. There exist sources transmitting gravitational waves of the same frequency range as sound waves. Hence, it is possible to translate certain gravitational wave signals into audible sound. Therefore results from gravitational wave astronomy are presentable as sound rather than in pictures.

Detection of gravitational waves   Gravitational waves produce oscillations in the distance between freely-floating masses. Figure 1.2 gives an impression of how gravitational waves *stretch* and *squeeze* matter or energy through which they propagate. Gravitational waves like electromagnetic waves are transverse, meaning they act in the plane perpendicular to their direction of propagation. Moreover, they oscillate such that a wave travelling through a plane containing two free-falling masses decreases their distance in one direction by the same amount as it increases the distance in the perpendicular direction. Hence there are two independent linear polarisations, called '+' and '×'. Figure 1.2 shows how a ring of freely-floating particles is deformed by a gravitational wave impinging perpendicularly. The upper row illustrates the effect of a purely '+' -polarised wave and the lower depicts the influence of a purely '×' -polarised wave on such an array of particles.

These changes in distance, δL, between the test masses are related to the dimensionless amplitude of the gravitational wave, h, and the total distance between the particles, L, via the equation:

$$\frac{1}{2}h = \frac{\delta L}{L}. \tag{1.1}$$

A world-wide network of ground based gravitational wave observatories aims to measure these changes in distance by means of laser interferometry. The total distance L is thereby given by the armlength of the interferometer. These ground-based detectors are sensitive to gravitational waves of relatively high frequencies between 10 Hz and 1 kHz or higher (see Figure 1.5). Seismic noise and length limitations prevent low frequencies from being accessible to ground-based detectors. The detection of sources emitting gravitational waves of lower frequencies will only possible from space.
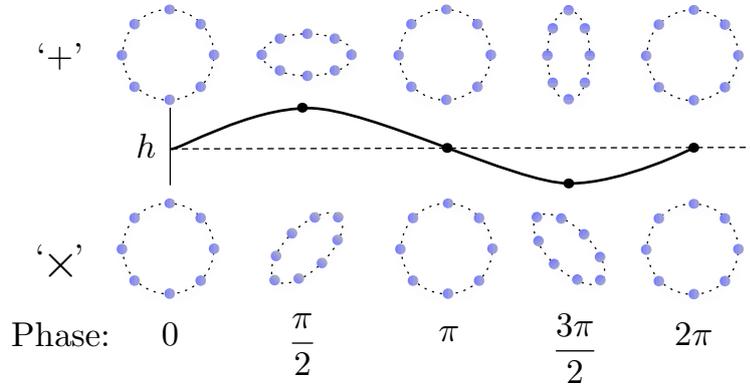
Figure 1.2: Illustration of the two polarisations of gravitational waves: the deformation of free-floating particles arranged as a ring is shown for both polarisations. The incident GW is perpendicular to the array of particles. The upper row shows how such a ring would be deformed by the gravitational wave that carries pure '+'-polarisation and the lower row shows the effect of a wave carrying pure '×'-polarisation. The amplitude of the GW is shown in phase steps of 90°

1.2 LISA

LISA, the Laser Interferometer Space Antenna, is a planned gravitational wave observatory in space. It is a joint ESA and NASA mission planned to be launched in 2020. Three spacecraft fly as an equilateral triangle with the armlength of 5 million kilometres, forming a Michelson interferometer with an additional arm. The third arm provides redundancy as well as independent information on both polarisations of the detected gravitational wave. The three spacecraft will travel in an Earth-like orbit around the sun while they remain in their triangular formation (see Figure 1.3). The plane of the triangle is inclined by an angle of 60° with respect to the plane of the solar system. The formation follows the Earth by an angle of 20°. Throughout the year the formation is passively maintained due to the choice of the individual spacecraft orbits so that the triangle rotates around its centre once per year. The orbits have been chosen to minimise the length changes between the spacecraft over the extended mission lifetime of 10 years.

A simplified scheme of one LISA spacecraft and the links to the remaining two spacecraft is shown in Figure 1.4. Each spacecraft carries two test masses (TM) and two telescopes. The test masses define the ends of the 5 million km long interferometer-arms. Their differential length changes are measured interferometrically using infrared lasers. The laser light propagates from one spacecraft to the other two via the telescopes and is reflected by one of the two test masses. Since diffraction causes great losses over the distances between the spacecraft, it is not possible to reflect the laser light directly back. Instead, the laser on the distant spacecraft is phase-locked to the incoming light and transmitted back. When the laser light arrives back at the original spacecraft, it is superimposed on another stable laser beam. The relative phase measurement gives information about the length
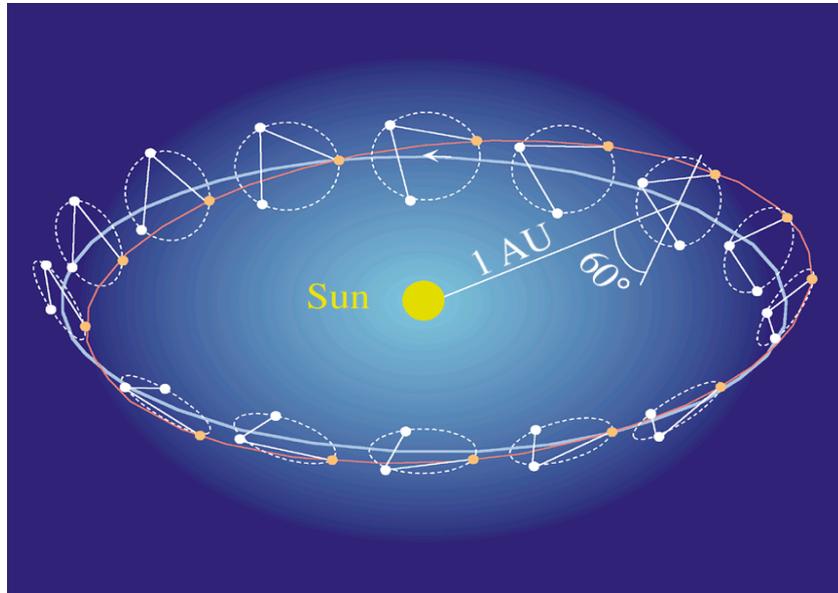
Figure 1.3: The heliocentric orbit of the LISA satellites. The plane of the triangle is inclined
at 60 to the ecliptic. The triangle appears to rotate once around its centre during
one revolution around the sun. Source: [6]

.
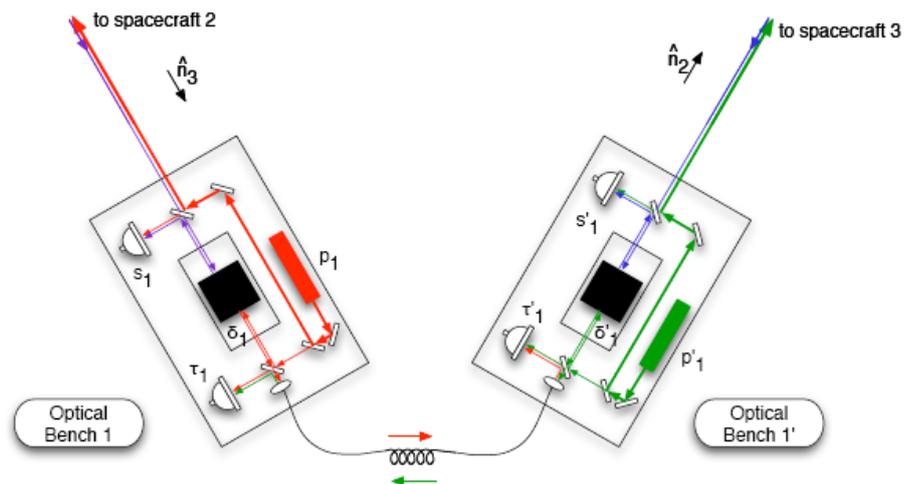
Figure 1.4: Schematic of the two optical benches onboard one LISA satellite. Indicated are
the phasemeters (s and τ), the lasers (p) and the test masses (δ). The picture
is taken from [7].

of that interferometer arm. The difference between the phase measurements for the two arms gives information about the relative changes in the two arms caused by the gravitational wave signal [8].

For the detection of gravitational waves the test masses, which reflect the laser light on the spacecraft need to be in free-fall within the measurement band of LISA. Outside this range as well as along the insensitive measurement axes, the position of the test masses inside each spacecraft is controlled by means of the inertial reference sensor (IRS). Details about these sensors can be found in [9].

1.3 Sensitivity of LISA

LISA's armlength has been chosen to achieve best sensitivity between about 0.1 mHz and 1 Hz. Thus, the frequency range covered by ground based detectors is completely different to that of LISA. Hence, these two kinds of observatories are sensitive to entirely different groups of gravitational wave sources. There exist various publications on gravitational wave sources and a nice overview of low frequency sources can be for example found in the Pre-Phase A Report for LISA [6]. Their discussion is beyond the scope of this thesis. Instead some examples for low frequency sources which LISA aims to detect shall be listed without further description: The coalescence of supermassive black holes (MBH) at the centres of merging galaxies, the radiation from compact binaries in our galaxy and the infall of small black holes, neutrons stars, and white dwarfs into MBH at galactic centres. The sensitivity of LISA reaches up to

$$h \approx 10^{-23}, \tag{1.2}$$

in the millihertz range. Hereby distance fluctuations, δL, measured over a period of one year and a signal-to-noise-ratio (SNR) of 5 are assumed. Figure 1.5 illustrates the complementarity of the ground-based interferometers and LISA using the example of the world's largest gravitational wave observatory, LIGO. The sensitivity (denoted on the y-axis) is plotted in terms of the dimensionless gravitational wave amplitude.

More on the LISA sensitivity is given in Chapter 5, where the introduction of non-Gaussian noise into the LISA data generator is presented.

LISA is one of the most challenging space missions planned for the next decades. While no completely new technologies need to be invented, the performance of existing technology must be improved and additional functionality is required. Some examples of fields this applies to are: micropropulsion, drag-free sensing and actuation and interferometric measurement systems. These are only a few examples of key technologies on which great demands are made. The most challenging task is thus the advancement and testing of the individual technologies as well as their assembly to the complete system of high precision technology. A great amount of technology to be developed for LISA has already progressed enormously.

Some of the technologies, like the drag-free sensing and actuation of test masses, cannot be tested on ground. Thus system level tests including these key technologies need to be done in space. This is the reason for the development of the technology demonstrator mission LISA Pathfinder [10].
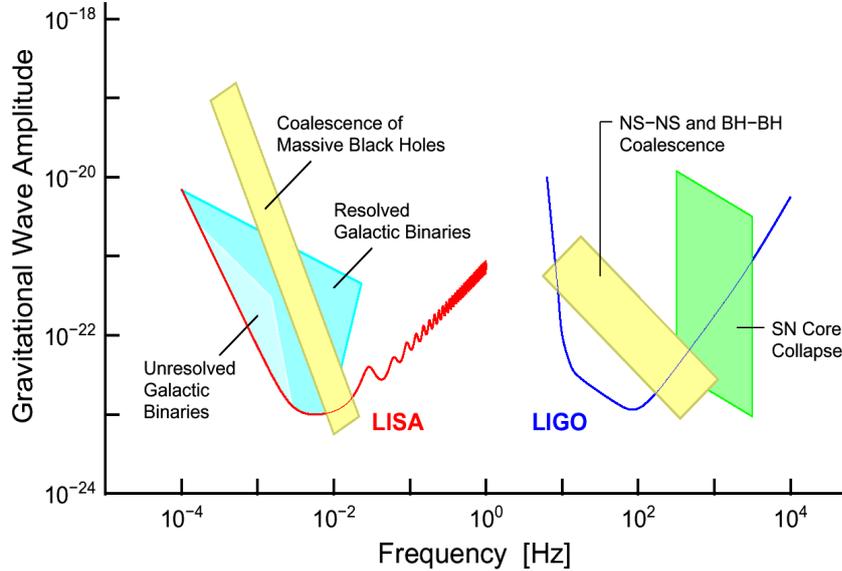
Figure 1.5: Sensitivity curves for LISA and LIGO and some sources that could be detected
by these gravitational wave observatories. For the LISA curve, a signal-to-noise-
ratio of 5, and averaging over one year and all possible directions and polarisa-
tion angles is assumed. Source: [6]

1.4 LISA Pathfinder

The technology demonstrator for LISA, LISA Pathfinder (LPF), is a space mission
to be launched in 2012. It carries two different payloads: the Disturbance Reduction
System (DRS) provided by NASA, and the LISA Technology Package (LTP) from
ESA. This thesis deals solely with the European experiment LTP. It comprises
two free-falling test masses, whose distance is measured interferometrically. LISA
Pathfinder can be regarded as the attempt to shrink one LISA arm to 38 cm in
order to realise it on one single satellite. The main science goal for LISA Pathfinder
is the verification of free-fall between the two test masses at a level of

$$a(f) = 3 \cdot 10^{-14} \mathrm{m\,s}^{-2} / \sqrt{\mathrm{Hz}} \left[1 + (\mathrm{f}/3\,\mathrm{mHz})^2 \right] \tag{1.3}$$

in a measurement band (MB) between $1\,\mathrm{mHz}$ and $30\,\mathrm{mHz}$ [11].

For LISA Pathfinder, flight hardware is being built and parts of the final assembly
have already been tested and thereby space-qualified at ESTEC, ESA. Testing at
the ESTEC facilities for space-qualification comprises a number of mechanical as
well as thermal and radiation tests. Figure 1.6 shows a photograph of the spacecraft
during the shock tests.

1.5 LISA Technology Package and on-orbit operation

The main payload of LISA Pathfinder is the LISA technology package (LTP). It
is composed of two vacuum containers carrying two test masses connected by an
optical bench. The test masses serve as end mirrors of the laser interferometer set

up on the optical bench. Figure 1.7(a) shows an artist's impression of LTP. The partly transparent 3D view reveals the two test masses that will be placed inside the vacuum chambers. The photograph in Figure 1.7(b) shows two dummy vacuum chambers inside the science module of the LPF spacecraft illustrating the assembly of LTP inside the spacecraft.
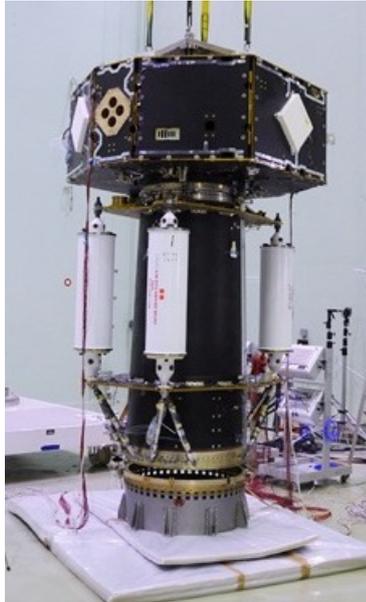
Figure 1.6: The LISA Pathfinder spacecraft composite. Science module on top of the propulsion module. Dummy tanks are attached to the propulsion moulde. The photograph has been taken during shock tests at ESTEC, The Netherlands, in October 2008. Source: [10]

In the previous section it has been pointed out that the need for a demonstrator mission arises from the numerous challenging technologies to be implemented in LISA that cannot be completely tested on-ground. The key technologies tested by accomplishing the LISA Pathfinder mission are:

Micro-Newton thrusters: Propulsion mechanism applying forces of the order of a few 100's of μN. They enable the spacecraft to remain centred on the test mass.

Inertial Sensor and actuator: A capacitive sensor, measuring distances between the electrode housing and the test mass and actuating the test mass position.

Drag-free and Attitude control: The control system of the two test masses inside the spacecraft and the control of the spacecraft itself.

Interferometry: The interferometer onboard LISA Pathfinder measures the distance between the test masses with picometre accuracy.

In the course of this introductory section the above terms will be explained further, however, for detailed technical discussions references are given. All future gravita-
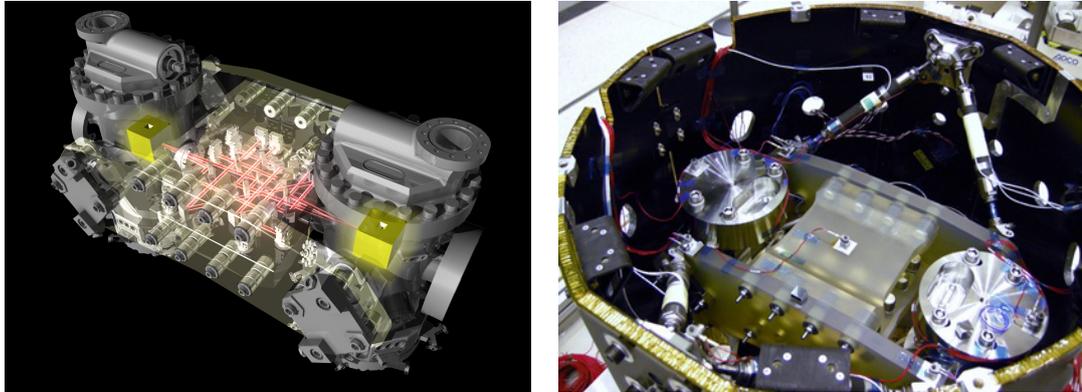
(a) Artist's impression of LTP.

(b) A photograph of the Science module of LTP.

Figure 1.7: The LISA Technology Package (LTP), including the two test masses inside the vacuum chambers and the optical bench between them. (a) Partly transparent view (artist's impression) revealing the test masses inside their vacuum enclosures. The test masses serve as end mirrors for the laser interferometer on the optical bench between them. (b) This photograph shows two dummies of the two vacuum chambers which will contain the proof masses. It was taken at ESTEC during the acoustic tests in 2008. Source [10]
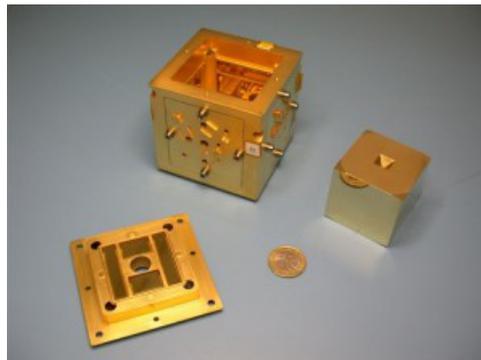
Figure 1.8: Each vacuum chamber contains a test mass enclosed in an electrode-housing box. The photograph depicts the electrode housing with one test mass. The cubic mass, 4.6 cm on a side, weights approximately 2 kg and is made from a gold-platinum alloy. The housing is made of molybdenum and sapphire coated with gold.
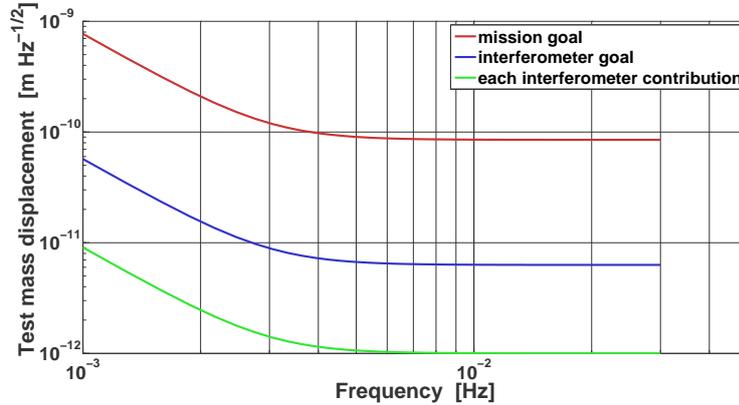
Figure 1.9: The LISA Pathfinder mission requirements. The upper most curve represents the requirement on displacement noise due to force noise on the test masses for the entire mission. The curve below indicates the requirement on the interferometric measurement. The lowest curve is not an official requirement but rather a guideline for noise contributions of individual noise sources of the OMS.

tional wave detectors in space will benefit from the system level testing in space of these key technologies.

LTP aims to measure the geodesic motion of the two test masses. However, the test masses cannot be completely shielded from external and internal forces. External forces may arise, for instance, from solar radiation pressure. Examples for internal disturbances are magnetic forces, residual charges and temperature fluctuations. Hence, in order to realise so-called *drag-free* motion of the test masses, a control system is needed to reduce the influence of these disturbances.

This control system is the Drag-Free and Attitude Control System (DFACS): it is implemented to counteract the disturbances, forces and torque applied on the spacecraft to maintain the free-floating conditions on the main test mass. The control laws running on the spacecraft On-Board Computer (OBC) use the test mass position measurements to compute the necessary forces to apply to the spacecraft to maintain a steady position and orientation of the test mass. These forces are applied by the Micro-Newton thrusters that move the spacecraft around the test mass. The systems that measure the test mass position are the inertial sensors and the optical metrology system (OMS) [12]. The inertial sensors measure the position of each test mass with respect to their electrode housing, depicted in Figure 1.8. The OMS comprises two interferometers: the so-called X1 interferometer measures the distance between one test mass and the spacecraft, whereas the X12 interferometer measures the distance between the two test masses.

LISA Pathfinder's operational phase will last six months. Out of these, 90 days of the complete mission time are reserved for measurements done with LTP. After that 60 days for DRS follow and finally a shared mission time for both payloads of 30 days is foreseen. Every day of the mission must be well planed in advance. The measurements to be done are defined in detail in the so-called Experiment Master Plan (EMP). The aim is to achieve the requirement plotted in Figure 1.9. The Figure shows three requirement curves: the mission goal, the requirement on

the interferometric measurement and a guideline for the requirements on each instrument contributing to the optical metrology system (OMS). For requirements on individual system components see also [13]. The requirements are plotted as test mass displacement but could as well be given as equivalent acceleration noises on the test masses. In order to achieve these requirements analytic analysis and simulations were done for various noise sources. These analyses are well documented and comprise for example:

- test mass acceleration noise [14],

- thruster noise [14],

- interferometer measurement noise [15],

- laser frequency noise [16],

- noise due to optical path length difference [17].

In the course of the LTP mission time the noise sources are to be minimised in order to achieve the target performance of the experiment. The experiments related to the various noise sources are defined in the above documents, which belong to the EMP but certain details about the course of action will depend on the results of the corresponding measurements.

The main scientific experiment to be accomplished with LISA Pathfinder is the measurement of the geodesic motion of the two test masses of LTP. For this measurement a certain setup of the hardware inside LTP is needed. In the following, the arrangement of the test masses and the measurements needed, along the sensitive x-axis, for the performance of the experiment are given. The procedure described in the following applies to an ideal, one-dimensional case, where no forces acting along the non-sensitive axis are included.

- Test mass 1 is free-floating within its housing, which means there is no control force from the Inertial Sensor (IS) applied to the test mass. The spacecraft is made to follow its motion.

- The position of the free-floating test mass (TM1) relative to the spacecraft is measured interferometrically, via the so-called X1 interferometer.

- The position of test mass 2 is controlled electro-statically by means of the Inertial Sensors.

- The distance between the two test masses is measured by means of the so called X12 interferometer. This signal is used to control the position of test mass 2 and thereby TM2 is made to follow TM1.

This setup is called the *science mode* and it is the most sensitive measurement mode of LTP. To implement this mode, however, a series of actions is necessary between the time of arrival of the spacecraft in its orbit and the start of the experiments in science mode. One reason for this is that during launch the test masses are fixed by the so-called caging mechanism in there housings to withstand the mechanical

load during launch. Only after the spacecraft reaches its final orbit, the test masses are released and their position pre-aligned for the interferometrical measurement. The complete procedure is documented in the Experimental Master Plan mentioned above.

In science mode test mass 1 is kept in free-fall. Its position with respect to the spacecraft is known from the interferometrical measurement. The spacecraft is made to follow the motion of test mass 1 by means of the drag-free control loop, which controls the Micro-Newton thrusters. Test mass 2 is directly controlled via the electro-static control loop to follow the motion of the spacecraft. This control is realised with low bandwidth to keep the noise introduced by the electro-static actuator small. In this way the motion of test mass 1 is drag-free and test mass 2 serves as reference for the measurement of this motion. The system that combines all of these control laws is the DFACS, mentioned above.

The extraction of the differential residual acceleration of the test masses by accounting for all control forces applied, is one example for a variety of tasks that need to be prepared from the side of the data analysis for LISA Pathfinder. A general overview to the field of data analysis for LTP will be given in the next section.

1.6 Data Analysis - Characterisation of LTP

LTP will measure the free-fall between two test masses. The baseline for the sensitivity of this measurement is of the order of $3 \cdot 10^{-14}$ m $/(\mathrm{s}^2\sqrt{\mathrm{Hz}})$ in a measurement bandwidth between $1\,\mathrm{mHz}$ and $30\,\mathrm{mHz}$.

It was pointed out earlier in this introduction that LTP can be regarded as single LISA arm downscaled from 5 Million km to 38 cm. LTP is completely set up within one spacecraft. Also it has already been stated that the sensitivity to gravitational waves scales with $\delta L/L$, with L being the length of the optical path between the test masses. Hence LTP is effectively insensitive to gravitational waves [18].

As such the objective of its data analysis is very different from the one of LISA and the ground based observatories, where the search of gravitational wave signals is the focal point of all considerations. The challenges for the LTP data analysis arise mainly from the limited mission time for LTP (about 90 days), which is why it is very important for the mission's success that every experimental run to be carried out is very well planned in advance.

In preparation of the data analysis for LTP the work presented in this thesis can be divided into two different parts:

- The data analysis of ongoing laboratory experiments and tests of engineering and flight models of the LTP hardware. Especially the analysis of experiments from the engineering model (EM) of the optical bench for the LTP interferometer.

- Implementing and simulating the experiments that are to be processed during the mission. This is done by the so called Mock Data Challenges, which are discussed in detail in Chapter 4.

In the course of fulfilling these tasks, reliable tools for the data analysis for the mission are implemented. The data analysis for the LISA Technology Package onboard LISA Pathfinder is a major topic of this thesis.

2 The data analysis infrastructure for the LISA Technology Package

The LISA Pathfinder mission time is strictly limited. Experiments must be prepared in detail prior to the mission in order to maximise the mission's science output. The task of planning every experiment in advance becomes even more challenging when taking into account that the experiments are not independent from each other, but the result of one is likely to affect following experiments.

This situation leads to special demands on the data analysis activities, as only a robust and carefully tested tool shall be used for the quasi-online data analysis. By quasi-online post-processing of obtained data on-ground is meant. Results of this may affect decisions for the experiments to follow. It becomes evident that the LISA Pathfinder mission needs a robust and flexible data analysis tool which enables scientists to carry out the complete data analysis of the mission.

In late 2006 the development of such a data analysis tool started, and the idea of an object-oriented LTPDA (LISA Technology Package Data Analysis) MATLAB toolbox formed. In this chapter LTPDA will be presented: its main concept will be exposed, the main functionality will be pointed out and some special features will be highlighted. [19] also gives some detailed information on LTPDA and a general overview about the data analysis for LTP. The LTPDA development is a joint effort between the AEI in Hannover and the University of Trento, with about ten people working full time at the moment.

2.1 Requirements on the data analysis for LTP

LTPDA will be delivered to ESA and will be tested according to ESA standards. Thus a detailed set of requirements on the software tool have been formulated and are documented in official technical reports: the user requirements are defined in the User Requirement Document (URD) [20] and based on this is the Software Requirements Document [21]. In this context discussions will be limited to the most important points leading to the concept that is finally implemented.

The defined task was to develop a comprehensive data analysis environment which contains all analysis algorithms necessary to completely characterise the LISA Pathfinder experiment. It was clear from the beginning that there will be non-programming experts among the scientists performing the online data analysis of the mission. As such the software tool should require only a short training period.

LTPDA is implemented as a MATLAB toolbox and will in the following be denoted simply as toolbox. The main reason for this choice of a commercial software environment is that it comes with a large collection of professionally implemented algorithms useful for all sorts of data analysis. Moreover, a commercial software

like MATLAB does not need to undergo official testing. As such original MATLAB functionality is used wherever possible to limit the amount of necessary testing. Moreover the mathematical oriented MATLAB scripting language with its large function library simplifies common data analysis tasks compared to other programming languages like C.

Another major aim was to develop the toolbox such that every analysis is fully reproducible and completely traceable back to the raw data used. This is especially important for sharing results among a team of scientists, where everybody works on the same data like in LISA Pathfinder and also for maintaining the readability of the results until the mission time of LISA. Analysis results will be comprehensible and easily comparable. As such, these results are not simple files containing experimental data. They are objects having a certain structure, which makes it possible to store all kinds of information.

It has been agreed that the user of the developed toolbox should be able to do the following things when obtaining a result of a data analysis accomplished with the same toolbox:

- Find out about all steps that were involved in the production of the final result, including

    - information on the raw data that was involved (date, channel, time segment, time of retrieval).

    - all operations that have been applied to modify the raw data e.g which functions were applied.

- Re-do all operations provided she or he has access to the raw data and the toolbox (LTPDA).

- Modify any of the applied operations to obtain a new result.

2.2 The concept of the Analysis Object

The concept of the Analysis Object (AO) arose from the above requirements on the software tool. The first document dealing with the concept is [22], which still applies for the general overview of Analysis Objects. Detailed up-to-date information can be found in the manual of LTPDA [23]. It has a structure providing the possibility to not only store the data resulting from an analysis but also various information about it. Examples for fields included in the architecture of the Analysis Object are:

- a description,

- information on the raw data that has been used,

- a history of all operations on the data that have been performed and

- the type of the resulting data, e.g. time domain data or frequency domain data.

Many more fields exist and they partly depend on the type of the resulting data. For example it makes not much sense to speak of the number of seconds of a frequency data series. As such, different data classes exist for example for frequency and time dependent data. These are `fsdata` and `tsdata` respectively. The purpose of classes is discussed later in this chapter. Figure 2.1 reveals the first layer of the structure of an Analysis Object.
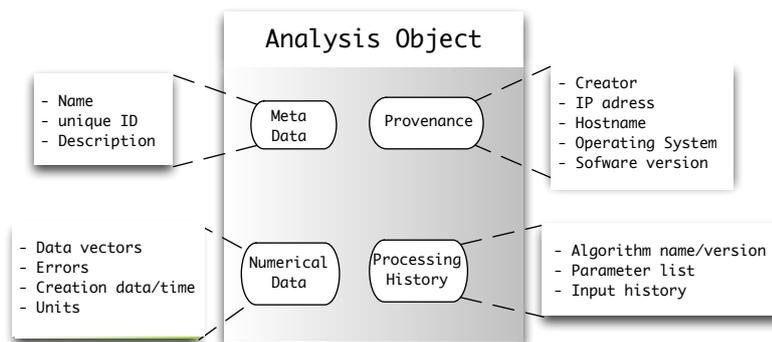
Figure 2.1: Illustration of the structure of an Analysis Object. It contains various fields for combining data and useful information about it.

2.3 Object Orientated Programming

For the realisation of the concept of Analysis Objects it is most convenient to chose an object oriented developing environment. In Object Orientation the relation of data and algorithms is represented by objects. Thereby it is possible to construct reusable components and put them together as modules of a complex program. This is an effective way to decompose a complex project into smaller components which may be easier to understand and which are reusable within the project. Consequently the management of such complex software projects gets simplified significantly. Today object oriented programming is of great importance for the development of comprehensive Applications.

Classes A software tool can represent all kinds of different categories. These can be physical objects like animals or vehicles or virtual objects like financial markets. In object orientation these categories are represented by classes. States are properties and methods are operations of these classes. Objects are instances of classes and they are constructed according to the fixed behaviour of this class.

The basic architecture of a software tool needs to be elaborated carefully in advance to the development. Hence it must be defined what type of data elements are needed and which operations in the form of methods need to be implemented. All desired information on data, depending on the type of object the data belongs to, can then be stored as properties.

Different methods belonging to different classes can have the same name. Defining a method with the name of an already existing one is called *overloading*. Which

method finally gets called, depends on the class the object belongs to. This is a key part of the development of LTPDA, where existing MATLAB functions like *plus* and *minus* are overloaded for Analysis Objects. They could be completely redefined, but of course it is easier for the user to keep the methods behaviour the same as far as possible.

Also in MATLAB all variables belong to a certain class. For instance double values belong to the MATLAB class *double*. In MATLAB it is possible to have user defined classes and objects. As such the class `ao` has been defined together with a set of classes for objects containing different types of data. At present five different types of data objects exist in LTPDA:

tsdata: For storing time dependent data (time-series).

fsdata: For storing frequency dependent data, like e.g. spectra or transfer functions.

cdata: For storing constants or any non-dependent values.

xydata: For storing two types of any kind of data that depend on each other.

xyzdata: For storing three types of any kind of data that depend on each other.

All of the above objects are part of Analysis Objects. They represent sub-objects of the Analysis Object.

It shall be noted that once an object is defined with all its properties including the data, the user will deal only with objects and functions. The objects are treated like common variables whose metadata is kept hidden from the user unless the user calls the appropriate field.

2.4 The MATLAB Toolbox - LTPDA

LTPDA stands for LISA Technology Package Data Analysis and is the name of the MATLAB toolbox developed for the data analysis of LISA Pathfinder. It implements the concept of Analysis Objects described above by means of object orientated programming. At present the project comprises:

- around 2,500 source code files,

- about 100,000 lines of code,

- approximately 60,000 lines of documentation.

The toolbox is completely integrated within MATLAB which includes the help pages being part of the MATLAB manual. Figure 2.2 shows a screen shot of one of the introductory help topics inside MATLAB. On the page shown, the hierarchical scheme of the classes of the toolbox is introduced. The help can as well be found online at [23].

Figure 2.3 shows the most important classes containing the most important methods. Emphasised are the so-called user classes, the user interacts only with these classes directly. The call of non user-classes is happening hidden from the user. Thus
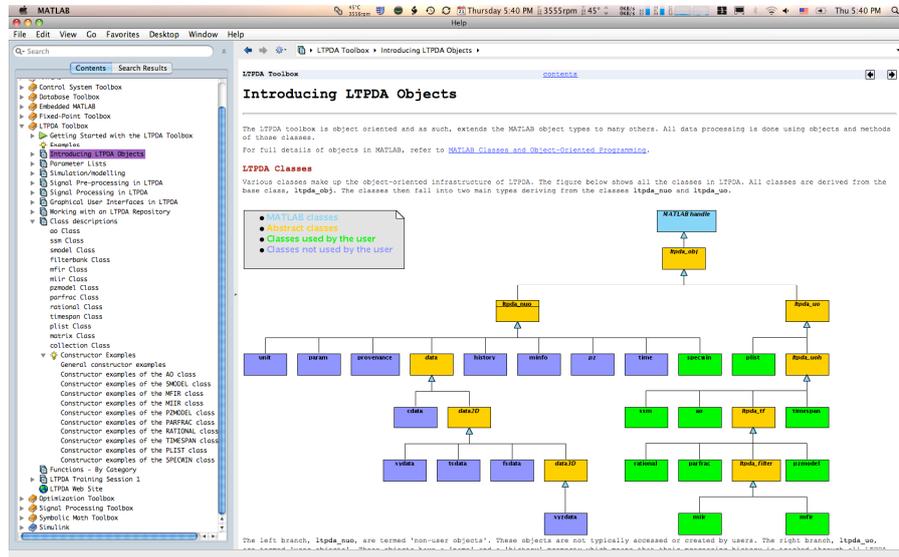
Figure 2.2: Screenshot of the help of LTPDA inside MATLAB. Shown is the introductory topic about its classes. The task-bar on the left hand side shows the help topics for every toolbox installed in MATLAB.
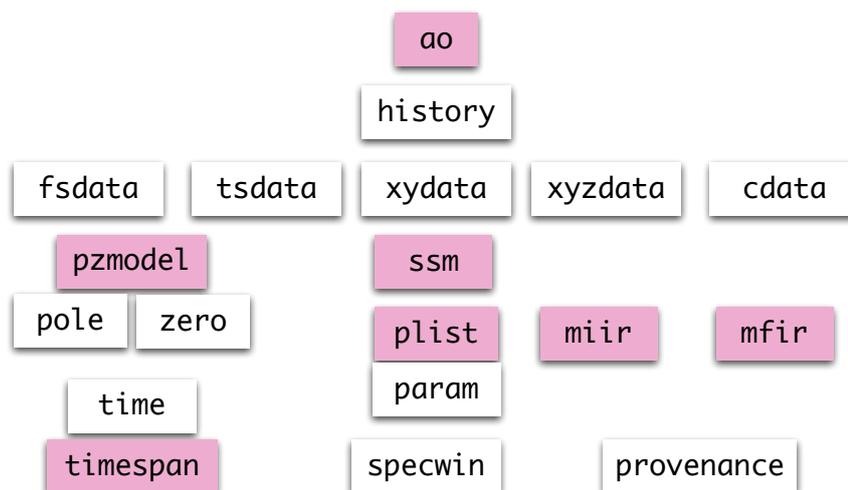
Figure 2.3: LTPDA classes, the list is not complete but only the most important classes are shown. Coloured are the classes the user has access to.

the architecture of LTPDA is well structured by numerous classes without compli-
cating its usage. The majority of methods belong to the class `ao` which means they
act on Analysis Objects and give Analysis Objects out. Figure 2.4 depicts the list
of methods only from the `ao`-class.


```
Methods for class ao:

Contents          delayEstimate      gapfillingoptim   ln                    noisegen1D        setDy               sumjoin
abs               demux              ge                log                   noisegen2D        setFs               svd
acos              det                get               log10                 noisegen_vpa      setMdlfile          svd_fit
angle             detrend            getdof            lpsd                  norm              setName             t0
ao                dft                gt                lscov                 normdist          setPlotinfo         table
asin              diag               heterodyne        lt                    nsecs             setProcinfo         tan
atan              diff               hist              ltfe                  offset            setT0               tdfit
atan2             display            hypot             ltp_ifo2acc           phase             setUUID             tfe
bin_data          dopplercorr        ifft              ifo2acc               phasetrack        setX                timedomainfit
bsubmit           downsample         ifo2acc           md5                   plot              setXY               times
cat               dropduplicates     imag              mdc1_cont2act_utn     plus              setXunits           timeshift
char              dsmean             index             mdc1_ifo2acc_fd       polyfit           setY                transpose
cohere            dx                 integrate         mdc1_ifo2acc_fd_utn   power             setYunits           type
complex           dy                 interp            mdc1_ifo2acc_inloop   psd               setZ                uminus
compute           eig                interpmissing     mdc1_ifo2cont_utn     psdconf           sign                unwrap
confint           eq                 inv               mdc1_ifo2control      pwelch            simplifyYunits      update
conj              eqmotion           iplot             mdc1_x2acc            rdivide           sin                 upsample
consolidate       evaluateModel      iplotyy           mdc2_ifo2acc_fd       real              smallvec_coef       validate
conv              exp                isprop            mdc2_ifo2control      rebuild           smallvector_lincom  var
convert           export             isvalid           mean                  report            smallvectorfit      viewHistory
copy              fft                join              median                resample          smoother            whiten1D
corr              fftfilt            lcohere           metropolis1D          residue           sort                whiten2D
cos               filtSubtract       lcpsd             metropolis2D          retarded_delay    spectrogram         x
cov               filter             le                min                   rms               spikecleaning       xcorr
cpsd              filtfilt           len               minus                 sDomainFit        split               xfit
created           find               linSubtract       mode                  save              sqrt                xunits
creator           firwhiten          lincom            mpower                scale             std                 y
csvexport         fixfs              linedetect        mrdivide              search            straightLineFit     yunits
ctranspose        fngen              linfit_ri         mtimes                select            string              zDomainFit
curvefit          fs                 linlsq            ne                    setDescription    submit              zeropad
delay             gapfilling         lisovfit          noisegen              setDx             sum

Static methods:

SETS              getBuiltInModels   getInfo           initObjectWithSize    update_struct
VEROUT            getDefaultPlist    getNDSchannelList  retrieve
```

Figure 2.4: The complete set of methods of the `ao` class inside LTPDA. It is the largest
class of the toolbox containing most of the methods.

The implementation of methods is realised in different ways:

Methods are newly developed: This applies mostly to functions that make use of
the different fields of the AO. For example the *select* method lets the user
select a segment of data of a certain time or frequency span, depending on
the type of data and the user's needs.

Methods are implemented in C: MATLAB offers the possibility to embed C-code
in the software via a so-called *wrapper*. This type of implementation is re-
stricted to methods where C-code offers a great speed advantage over MAT-
LAB. This is for example true for the *logarithmic power spectral density* (lpsd)
method [24].

Methods are overloaded: One advantage in using MATLAB as development plat-
form is the great amount of ready implemented and tested functions. Essential
functions like operators (plus, minus, *ect.*), see also Figure 2.5, are overloaded
and extended to work with Analysis Objects.

Figure 2.5 shows a selection of methods in LTPDA grouped in categories named
after their purpose of application. There are the methods for:

Spectral analysis: Producing frequency dependent data.

Spectral		Pre-processing		Operators
psd	lpsd	resample	select	+,-,^,*,/,.*,./,.^,'
tfe	ltfe	downsample	split	abs, sin, cos, tan,
cohere	lcohere	upsample	whiten	exp, ln, log, sqrt,
fft		filter	delay	min, max, real, imag

Figure 2.5: A selection of LTPDA methods from three different categories of required operations.

Pre-processing: Preparing data for further analysis.

Operators: All common operators need to be implemented. Most of the operators are overloaded MATLAB methods.

Functionality and usage of LTPDA   The reason why existing MATLAB functions need to be overloaded for the use in LTPDA is the special structure of Analysis Objects. Methods of the class `ao` only deal with AOs rather than with variables. They accept AOs as input and also produce AOs as output. How the methods deal with AOs internally is hidden from the user and in principle the usage is not any different from dealing with plain variables. The following example shall give an idea about the usage of LTPDA.

```
>> a1 = ao(5); % Creates an AO of type cdata with a constant value of 5
>> a2 = ao(4);
>> a3 = a1+a2  % adds up the data inside the two AOs (4+5)
───────────── ao 01: (a1+a2) ─────────────

       name:  (a1+a2)
       data:  9
              ─────────── cdata 01 ─────────────
                  y:  [1x1], double
                 dy:  [0x0], double
             yunits:  []
              ───────────────────────────────────

       hist:  ao / plus / Id : plus.m, v1.452009/08/3117 : 09 : 44ingoExp
    mdlfile:  empty
description:
       UUID:  3f5e7e9e−3042−4569−9411−a5d520d7fbf5
       ───────────────────────────────────────
```

In the example two Analysis Objects are created. The first having the constant value 5 and the second contains the value 4. Once the objects are created they can be treated as variables and, like in the given example, simply be added together. In fact the method that is called is an overloaded method for *plus* that deals with Analysis Objects.

Additional information can be assigned to all Analysis Objects. Some fields contain standard values, in the above example the resulting object `a3` is named after

the operations done (a1+a2). Information will be carried along during operations. For instance, physical units should be attached to all data in newly created Analysis Objects and will be computed correctly with subsequent operations.

Figure 2.6 illustrates a method call where the name *method* refers to any possible method: In general the behaviour of methods is configured via so-called parameters. These parameters consist of a name ('key') and a value and they are collected in so-called parameter lists (`plist`). For every method it is well documented what sets of parameters are applicable.

Figure 2.6: A method call in LTPDA. The name 'method' can be replaced by any method name of LTPDA and 'pl' stands for the *parameter list*. Methods of LTPDA deal with Analysis Objects containing the data to be processed. Their behaviour is configured via parameter lists by which the user can choose different options that may be implemented inside the algorithm.

Figure 2.7 depicts how the methods keep given information on Analysis Objects during operations. All information needed to trace a given result back to the raw data is kept. The complete set of information necessary is called the **history** of an object.
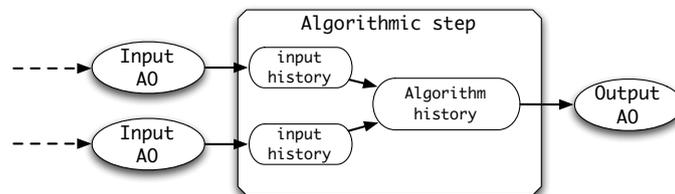
Figure 2.7: Illustration of how the so called `history` of an Analysis Object is tracked during operations. The `history` object of every input AO is called by every algorithm applied and each algorithm writes the input `history` plus its own processing `history` into the newly created AO.

Every algorithm working on an object in LTPDA adds an entry to the `history` of that object. The entry contains:

- the algorithm name,

- the algorithm version,

- the parameter list that was used by the algorithm,

- the names of the input objects and

- the `history` entries of the input object.

In this way a tree of `history` entries is built as an object passes through the individual operations of an analysis. The `history` entry is an object (or instance) of the `history` class and it is part of the Analysis Object. The processing tree can be extracted from the `history` object to be viewed and it can be used to rebuild the object or to alter the processing steps of the analysis. This fulfils the above listed requirements of traceability and reproducibility of any result of the data analysis for LISA Pathfinder.

The diagram in Figure 2.8 shows the processing `history` of an Analysis Object. It is to be read from the top to the bottom: two AOs are created, they contain time-series data following a random distribution. The spectrum of each of the objects is computed by the method `lpsd` and finally the two objects are added together. The additional information in the boxes represents all of the parameter list entries that can be set when applying a function. For example `T0` is the start time of a time-series inside an AO, which can be set.
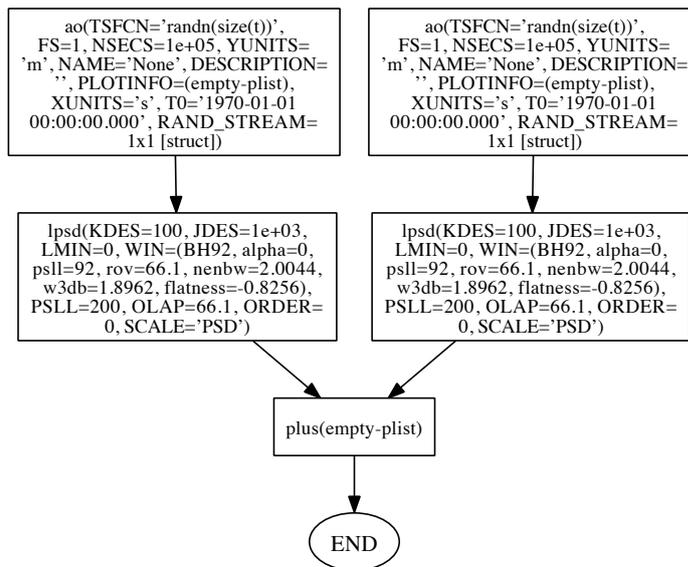
Figure 2.8: History tree for an Analysis Object. It is to be read from top to the bottom: Two AOs are created and contain a random time-series. After that there power spectral density is computed and finally they are added together. Each block represents an algorithm applied and contains all important configuration parameters used. The diagram was generated automatically by the method `viewHistory()` [25].

A `history` tree like the one shown is generated automatically by applying the method `viewHistory()` on the object, which creates a pdf document (if the appropriate third party software is installed [25]). These diagrams can become arbitrarily complicated just like the analysis process applied. The `history` object makes it possible to rebuild the entire analysis script that was used to achieve the analysis result.

Testing   Since LTPDA will be the official mission tool for data analysis it must be reliable and therefore its functionality must be well tested. Testing is done on different levels [19]:

Unit level tests: For every method of the toolbox one or more unit tests are implemented. This is a MATLAB file that tests two different aspects of the according algorithm:

>   **the syntax:** This test assures that the algorithm can be called as described in the documentation.

>   **the algorithm:** In this test, the result returned by the algorithm is checked (against the result from the appropriate MATLAB function or against applicable test data).

System level tests: For these tests blocks of several algorithms connected in various ways are implemented. These blocks form compatibility tests of the functions among each other.

User tests: Daily use of the software tool by various scientists are an effective way of testing the existing methods on all levels as well as revealing the need for new algorithms. In this context, especially the *Mock Data Challenges* and the laboratory analysis, which are discussed in detail in Chapter 4 and Chapter 3 respectively, shall be highlighted.

Acceptance test: this is a formal test campaign carried out by the ESA. Its aim is to assure that the software is capable of performing its role in the mission.

2.5 An example for a special tool of LTPDA: Generation of time-series with a prescribed spectrum - The Franklin noise-generator

The toolbox comprises a large collection of algorithms. This collection is a mixture of algorithms adopted from MATLAB, functions originally implemented for ground-based gravitational wave detectors (especially GEO600) and newly implemented methods by the LTPDA team and several other people (for every function the credits are included in the manual). For this section, one contribution of special interest is picked out and presented.

We want a method to generate Gaussian random noise with a given spectral density. Such time-series are needed, for example, to generate test data sets for programs that compute spectral densities, or as inputs for various simulations.

One way to realise this is to apply digital filters (FIR or IIR) to white input noise. This approach is effectively implemented for the generation of 'multichannel' noise with a given cross spectral density. 'Multichannel' transfer functions are identified by an automatic fit procedure based on a modified version of the vector-fitting algorithm (see z-Domain Fit in the user manual for LTPDA [23] for further details on the algorithm). Partial fraction expansion of 'multichannel' transfer functions and

the implementation of filter state initialisation avoid the presence of any unwanted *warm-up period.*

A different approach is implemented in LTPDA as the Franklin noise-generator. It produces spectral densities according to a given pole/zero model, meaning an analytical transfer function, and does not require any warm-up period. Franklin's noise generator is a method to generate arbitrarily long time-series with a prescribed spectral density. The algorithm is based on a paper by J. N. Franklin written in 1956 [26].

The algorithm is based on the propagation of a state vector, $\vec{y}$. The state vector is a random vector with a prescribed covariance matrix. For this method random vectors with two prescribed covariance matrices are required: one for initialisation, $\vec{y}_0$, and one for the noise contributions, $\vec{\varepsilon}$, to each time step.

The input to the method is a pole/zero model defining the desired spectrum of the time-series to generate, the sampling frequency f_s and the number of seconds to be generated. The generator operates on a real state-vector, $\vec{y}$, of length N which is maintained between invocations. It produces samples of the time-series in equidistant steps $T = 1/f_s$. The sequence of operations is as follows:

- initialising the state vector:

$$\vec{y}_0 = \mathbf{T}_{\text{init}} \cdot \vec{r},\tag{2.1}$$

- propagate the state vector:

$$\vec{y}_i = \mathbf{E} \cdot \vec{y}_{i-1} + \mathbf{T}_{\text{prop}} \cdot \vec{r},\tag{2.2}$$

- generate the final time-series:

$$\vec{x}_i = \vec{a} \cdot \vec{y}_i,\tag{2.3}$$

where $\vec{r}$ is a vector of independent normal Gaussian random numbers and $\mathbf{T}_{\text{init}}$, $\mathbf{E}$ and $\mathbf{T}_{\text{prop}}$ are real matrices and $\vec{a}$ is a real vector. The matrices are determined once by the algorithm from the given transfer function. The mathematical basis of the algorithm has been published by Joel Franklin in 1965 in [26] and in [27] the main concept as well as the implementation is described comprehensively. The key of the implementation is the computation of the matrices $\mathbf{T}_{\text{init}}$, $\mathbf{E}$ and $\mathbf{T}_{\text{prop}}$. It is described in [27], where the first implementation of the method is documented. This initial implementation has been coded in the programming language C by Gerhard Heinzel and has been transformed to MATLAB code for the integration into LTPDA as part of this work. The implementation will be described at the end of this section, where all matrices required for the individual processing steps listed above will be constructed. After that those steps will become less abstract. After this, the usage of the noise generator within LTPDA is shown and some significant examples will be presented.

2.5.1 Implementation

The implementation of the algorithm is taken from [26] and [27]. In this section the computation of the matrices $\mathbf{T}_{\mathrm{init}}$, $\mathbf{E}$, $\mathbf{T}_{\mathrm{prop}}$ and the vector $\vec{a}$ is presented. As mentioned in the previous section, these are the key components of Franklin's algorithm for generating time-series with prescribed spectra.

The starting point is always a mathematical description of the desired spectrum.

$$H(s) = \frac{a_0 + a_1 s + \cdots + a_m s^m}{b_0 + b_1 s + \cdots + b_{n-1} s^n}, \quad \text{with} \quad n > m, \tag{2.4}$$

where $s = i\omega = 2\pi i f$ is the Laplace variable and this transfer function is conceptually applied to the LSD in $1/\sqrt{Hz}$ of white input noise.

The denominator of the transfer function to compute is denoted by b and the numerator by a. Franklin's method starts with the computation of the propagation matrix $\mathbf{E}$. At first a matrix, $\mathbf{A}$, is set up using the b-coefficients of the input transfer function:

$$\mathbf{A} = \mathbf{A_{ij}} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & 0 \\ \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & \ddots & 1 \\ -b_0 & -b_1 & -b_2 & \cdots & -b_{n-1} \end{bmatrix} \quad \text{or} \tag{2.5}$$

$$\mathbf{A_{ij}} = \begin{cases} 1 & : \quad j = i+1, \\ -b_j & : \quad i = n-1, \quad (i = 0, \ldots, n-1; \quad j = 0, \ldots, n-1). \\ 0 & : \quad else, \end{cases} \tag{2.6}$$

where $n + 1$ is the number of b-coefficients in the denominator of the transfer function. This is due to the fact that here the counting of array entries starts at 0. It shall be noted that the b-coefficients are normalised to $b_n = 1$ beforehand. According to [26] $\mathbf{E}$ is given by

$$\mathbf{E} = \exp(\mathbf{A} \cdot \mathrm{T}) = \mathbf{E_{ij}} \quad (\mathrm{i} = 0, \ldots, \mathrm{n}-1; \quad \mathrm{j} = 0, \ldots, \mathrm{n}-1), \tag{2.7}$$

where $T = 1/f_{\mathrm{s}}$ the input sampling frequency.

Two instances of random vectors with prescribed covariance matrix are need for the final generation of the time-series. They are produced from generic vectors by multiplication with a lower triangular matrix $\mathbf{T}$. The resulting random vector has a covariance given by

$$\mathbf{C} = \mathbf{T} \cdot \mathbf{T^{\mathsf{T}}}. \tag{2.8}$$

Thus, once $\mathbf{C}$ has been found, the 'generator matrix' $\mathbf{T}$ can be found by decomposition, for example using a Cholesky decomposition an algorithm that is described in standard textbooks on Linear Algebra (for example, in Numerical Recipes [28]).

The next step of the implementation will provide the random vector generator matrix, $\mathbf{T}_{\mathrm{init}}$, for initialisation of the random vector with prescribed covariance matrix. This is the first of the two instances mentioned above. The resulting matrix $\mathbf{T}_{\mathrm{init}}$ will later be used only once by the generation algorithm for initialising the

---

Algorithm 1: Recipe for filling the matrix $\mathbf{B_{ij}}$ necessary for computing the random vector generator matrix for initialisation $\mathbf{T}_{\text{init}}$. The symbol $\leftarrow$ represents a variable allocation

---

for $i = 0$ to n-1 **do**
    if i is even **then**
        $j_0 \leftarrow \text{i}/2$
        $\text{s} \leftarrow (-1)^{j_0}$
        $\text{j} \leftarrow j_0$
        for $k = 0$ to n **do**
            $\mathbf{B_{ij}} \leftarrow \text{s} \cdot b_k$
            $\text{k} \leftarrow \text{k} + 2$
            $\text{s} \leftarrow \text{-s}$
            $\text{j} \leftarrow \text{j} + 1$
    else
        $j_0 \leftarrow (\text{i} + 1)\,/2$
        $\text{s} \leftarrow (-1)^{(j_0+1)}$
        $\text{j} \leftarrow j_0$
        for $k = 1$ to n **do**
            $\mathbf{B_{ij}} \leftarrow \text{s} \cdot b_k$
            $\text{k} \leftarrow \text{k} + 2$
            $\text{s} \leftarrow \text{-s}$
            $\text{j} \leftarrow \text{j} + 1$

---

state vector, $\vec{y}$. Analogue to Franklin's formulae we begin by setting up a matrix $\mathbf{B} = \mathbf{B_{ij}}$ using the recipe given in Algorithm 1.

For n = 4 the result from the recipe given in Algorithm 1 is as follows

$$\mathbf{B} = \begin{bmatrix} b_0 & -b_2 & b_4 & 0 \\ 0 & b_1 & -b_3 & 0 \\ 0 & -b_0 & b_2 & -b_4 \\ 0 & 0 & -b_1 & b_3 \end{bmatrix}. \tag{2.9}$$

For the next step the linear set of equations

$$\mathbf{B} \cdot \vec{m} = \vec{k} \tag{2.10}$$

must be solved for the unknowns $\vec{m} = (m_0, \cdots, m_{n-1})^{\mathsf{T}}$ and $\vec{k} = (0, \cdots, 1/2)^{\mathsf{T}}$.

Using the resulting $\vec{m}$, the matrix $\mathbf{C}_{\text{init}}$ can be filled according to the following rules.

$$C_{\text{init}} = \begin{cases} (-1)^{(i-j)/2} \cdot m_{(i+j)/2} & \text{if } (i+j) \text{ is even,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.11}$$

For n = 4 the matrix $\mathbf{C}_{\text{init}}$ reads:

$$\mathbf{C}_{\text{init}} = \begin{bmatrix} m_0 & 0 & -m_1 & 0 \\ 0 & m_1 & 0 & -m_2 \\ -m_1 & 0 & m_2 & 0 \\ 0 & -m_2 & 0 & m_3 \end{bmatrix}. \tag{2.12}$$

Finally the desired matrix $\mathbf{T}_{\text{init}}$ is obtained by decomposing $\mathbf{C}_{\text{init}}$ according to Equation 2.8 such that

$$\mathbf{C}_{\text{init}} = \mathbf{T}_{\text{init}} \cdot (\mathbf{T}_{\text{init}})^{\mathsf{T}}, \tag{2.13}$$

using, for example, the Cholesky decomposition mentioned above [28].

There is a second instance of random vectors with prescribed covariance matrix needed. It will be used for the propagation of each step in the final noise generation method. This covariance matrix is called $\mathbf{C}_{\text{prop}}$. It is set up similar to the previously described procedure for $\mathbf{C}_{\text{init}}$ above. The covariance matrix will again be decomposed according to

$$\mathbf{C}_{\text{prop}} = \mathbf{T}_{\text{prop}} \cdot (\mathbf{T}_{\text{prop}})^{\mathsf{T}}, \tag{2.14}$$

To start with, the non-symmetric $N \times N$ matrix $\mathbf{D}$ is to be constructed. The corresponding recipe is given again as pseudo-code in Algorithm 2.

Algorithm 2: Recipe for filling the matrix $\mathbf{B}_{\mathbf{ij}}$ necessary for computing the random vector generator matrix for initialising $\mathbf{T}_{\text{init}}$. The symbol $\leftarrow$ represents a variable allocation

---

for $i = 0$ to N -1 **do**
$\quad$ **for** $j = 0$ to N-1 **do**
$\quad\quad$ $\mathbf{D}_{\mathbf{ij}} \leftarrow 0$
for $i = 0$ to n-1 **do**
$\quad$ **for** $j = 0$ to n-1 **do**
$\quad\quad$ **for** $k = 0$ to n= -1 **do**
$\quad\quad\quad$ $\mathbf{D}_{\mathbf{g(i,j),g(j,k)}} \leftarrow \mathbf{D}_{\mathbf{g(i,j),g(j,k)}} + \mathbf{A}_{\mathbf{ik}}$
$\quad\quad\quad$ $\mathbf{D}_{\mathbf{g(i,j),g(j,k)}} \leftarrow \mathbf{D}_{\mathbf{g(i,j),g(i,k)}} + \mathbf{A}_{\mathbf{jk}},$

---

with $\mathbf{A}_{\mathbf{ij}}$ being defined in Equation 2.6 and the indexing function g:

$$g(i,j) = \begin{cases} \frac{i(i+1)}{2} + j & \text{if } i \geq j \\ \frac{j(j+1)}{2} + i & \text{otherwise.} \end{cases} \tag{2.15}$$

The matrix $\mathbf{D}$ is of size

$$2 \cdot n + n \; : \; 2,$$

where ':' denotes the 'division algorithm'. Thus, for $n = 4$, $\mathbf{D}$ becomes a 10×10

matrix. The so constructed matrix reads:

$$
\mathbf{D} = \begin{bmatrix}
0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\
-b_0 & -b_1 & 0 & -b_2 & 0 & 0 & -b_3 & 1 & 0 & 0 \\
0 & -b_0 & -b_1 & 0 & -b_2 & 0 & 0 & -b_3 & 1 & 0 \\
0 & 0 & 0 & -b_0 & -b_1 & -b_2 & 0 & 0 & -b_3 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & -2b_0 & -2b_1 & -2b_2 & -2b_3
\end{bmatrix} . \tag{2.16}
$$

Using this matrix, the linear set of equations

$$
\mathbf{D} \cdot \vec{p} = \vec{q} \tag{2.17}
$$

needs to be solved for $\vec{p}$, while $\vec{q}$ is filled according to:

$$
q_{g(i,j)} = \begin{cases} (\mathbf{E_{n-1,n-1}})^2 - 1 & \text{if } i = n - 1, \\ \mathbf{E_{i,n-1}} \cdot \mathbf{E_{j,n-1}} & \text{otherwise,} \end{cases} \tag{2.18}
$$

with $\mathbf{E_{ij}}$ from Equation 2.7. After solving the system 2.17 for $\vec{p}$, the elements of $\vec{p}$ are used to fill the covariance matrix C_{prop} using the indexing function $g(i,j)$:

$$
\mathbf{C}_{\mathrm{prop}} = p_{g(i,j)}. \tag{2.19}
$$

And finally the random vector generator matrix for propagation $\mathbf{T}_{\mathrm{prop}}$ can be found according to Equation 2.14.

Once all matrices have been computed, for the final generation of the time-series with a prescribed spectrum, the process is rather simple. It was stated earlier in this section that the final generator comprises three steps. The steps shall be repeated here for clarity:

$$
\begin{aligned}
\vec{y}_0 &= \mathbf{T}_{\mathrm{init}} \cdot \vec{r} & \text{initialisation of the state vector,} \\
\vec{y}_{\mathrm{i}} &= \mathbf{E} \cdot \vec{y}_{\mathrm{i}-1} + \mathbf{T}_{\mathrm{prop}} \cdot \vec{r} & \text{propagation of the state vector,} \\
\vec{x}_{\mathrm{i}} &= \vec{a} \cdot \vec{y}_{\mathrm{i}} & \text{generation of the final time-series.}
\end{aligned} \tag{2.20}
$$

$\vec{r}$ is a random vector of length n. It is generated by the built-in MATLAB random number generator producing Gaussian random numbers from a normal distribution with zero mean and unity variance. For every invocation for a new time step, a new set of n uncorrelated Gaussian random numbers is created. The random number vector acts on the state vector $\vec{y}$ to produce one time step of the final time-series per invocation.

2.5.2 Functionality and performance of the noise generator

The formula of the transfer function was given in Equation 2.4 on page 24. For the noise generator in LTPDA this must be given as a pole/zero model. The factors

for transforming poles and zeros to the form of $H(s)$ given in Equation 2.4 are as follows.

A single real pole at frequency f corresponds to the factor

$$1/\left(1 + \frac{s}{2\pi f}\right). \tag{2.21}$$

A complex pole (conjugated pair) at frequency f with the quality factor Q corresponds to the factor

$$1/\left(1 + \frac{s}{2\pi f Q} + \frac{s^2}{(2\pi f)^2}\right). \tag{2.22}$$

Accordingly, a single real zero at frequency f corresponds to the factor

$$\left(1 + \frac{s}{2\pi f}\right) \tag{2.23}$$

and a complex zero (conjugated pair) at frequency f with the quality factor Q corresponds to the factor

$$\left(1 + \frac{s}{2\pi f Q} + \frac{s^2}{(2\pi f)^2}\right). \tag{2.24}$$

For more information on pole/zero models, see the LTPDA manual [23].

The algorithm will transform the model to the transfer function of the form given in Equation 2.4. Thus the requirement $n > m$ means that the model must consist of more poles than zeros. This can always be achieved by adding out of band poles if necessary.

Once the model is defined the user needs to specify also the sampling frequency, f_s and the number of seconds, nsecs, of the time-series to compute. In LTPDA the implementation of this method is realised within the constructor of an Analysis Object. Thus, the method `ao` can be called with a parameter list (see previous section) containing the above listed information and the result is an Analysis Object containing a time-series with the prescribed spectrum. An example in MATLAB for using LTPDA for the generation of a time-series with a prescribed spectrum is given in the following.

```
% The gain of the transfer function is put to 1.
g = 1;
% defining the pole/zero model
pzm = pzmodel(g,{1e-6 1e-6 0.8 0.8},{1e-5 1e-5 5e-2});
% The parameter list for the noise generator is filled
% with all necessary information.
pl = plist('fs',100,...
'pzmodel',pzm,'nsecs',100000);
% The ao-constructor is called with the defined paramter list
% and the time-series is generated.
ng = ao(pl);
```

The noise generator produces a time-series of arbitrary length. The variable, ng, from the above call, for example contains a time-series of 10000 samples ($f_s \times$ nsecs).

However, to evaluate the result, the spectral content of that time-series is to be compared to the input target spectrum. As such the linear spectral density (LSD) of the generated time-series is computed using the logarithmic power spectral density algorithm, `lpsd`, [24] implemented in LTPDA. Figure 2.9 shows the plot of the analytical function computed from the input pole/zero model of the above example together with the linear spectral density of the generated time-series.
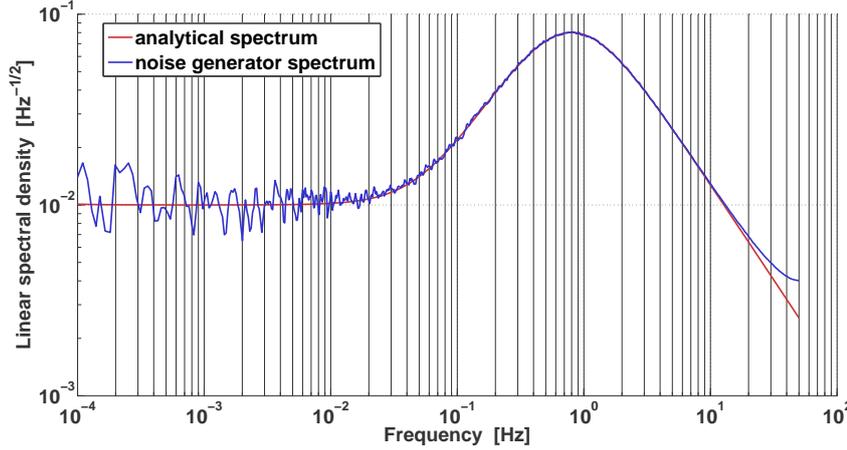
Figure 2.9: Numerical power spectral density of the generated time-series compared to the analytical function defined by the input pole/zero model.

At high frequencies the curve for the spectrum of the noise generator includes aliasing [27]. This causes the projection of signal components at frequencies above the *Nyquist frequency*

$$f_{\mathrm{Ny}} = fs/2 \qquad (2.25)$$

into the *Nyquist band*

$$0 \le f \le f_{\mathrm{Ny}}. \qquad (2.26)$$

Aliasing is an unavoidable effect yet it is computable and could be included in the computation of the analytical spectrum to match the two curves.

The pole/zero model used here is well suited for demonstrating the usage of the tool. However, for such rather simple examples conventional methods may be sufficient. The results of two more complicated examples will be presented in the following to demonstrate the performance of the noise generation based on Franklin's method. These examples are adapted from [27]. It is worth mentioning that for the pole/zero model of the next two examples the default MATLAB double precision would not be sufficient for the computation of the matrices. For computing the time-series that lead to the spectra shown in Figure 2.10 and 2.11 a higher precision was needed. Note however that once the matrices are computed the propagation of the time-series does not require the high precision.

In MATLAB arbitrary precision can be obtained by using the symbolic math toolbox which is as well required for some other applications of LTPDA. It provides the 'variable precision arithmetic' library with which the user can define the desired number of digits for every variable in symbolic math mode. While double precision

floating point provides a relative precision of 16 decimal digits, the number of digits can be set for the use of the noise generator and for the examples presented here the number of digits has been set to 32. In case of insufficient precision, the algorithm will fail to invert certain matrices and an error occurs.

The first spectrum represents a measurement from the Engineering Model (EM) of the LTP interferometer that will be described in Chapter 3. The assembling of the pole/zero model is omitted here for readability. Instead the poles and zeros defining the desired spectrum are listed in Table 2.1 and 2.2. Also the MATLAB calls are not presented anymore since they would not add any information. For both examples the generated time-series consist of 10 Million samples. Depending on the machine this may take around 20 seconds in MATLAB. In Figure 2.10 the resulting linear spectral density is plotted again together with the analytical function.

Table 2.1: Pole/zero model for a laboratory measurement of the LTP Engineering model for the interferometer, described in Chapter 3. The individual poles and zeros are defined by a frequency and a quality factor Q. Pole or zeros without a given Q are real. The gain of the transfer function is $g = 4.37 \times 10^3$ and the sampling frequency has been chosen to be $f_s = 100$.

Poles		Zeros	
f [Hz]	Q	f [Hz]	Q
10^{-6}		687.61×10^{-6}	2.211
10^{-6}		972.81×10^{-6}	
179.90×10^{-6}	2.156	11.14×10^{-3}	
218.46×10^{-3}		638.61×10^{-3}	557.6×10^{-3}
7.909	906	7.414	4.62
11.125	200	11.125	5.79
6.626	907.64×10^{-3}	973×10^{-6}	
16.4	5.047		
139×10^{-6}			

The next example is made artificially difficult to test the algorithm. Several peaks occur in the spectrum as can be seen in the plot in Figure 2.11. The spectrum computed from the generated time-series fits well the analytical spectrum. For low frequencies the computation of the spectrum gets costly. The generated time-series underlying the spectrum plotted in Figure 2.11 is plotted in Figure 2.12.

2.5.3 Comparison with conventional methods.

What is called a conventional method here is using a filter (FIR or IIR) with the desired transfer function and apply it to a time-series of white input noise. One advantage of the algorithm presented here over this conventional method is the precision with which the spectral content of the produced time-series matches the input spectrum. Another great advantage is the absence of any warm-up period. The common method of filtering using recursive IIR filters requires a certain period
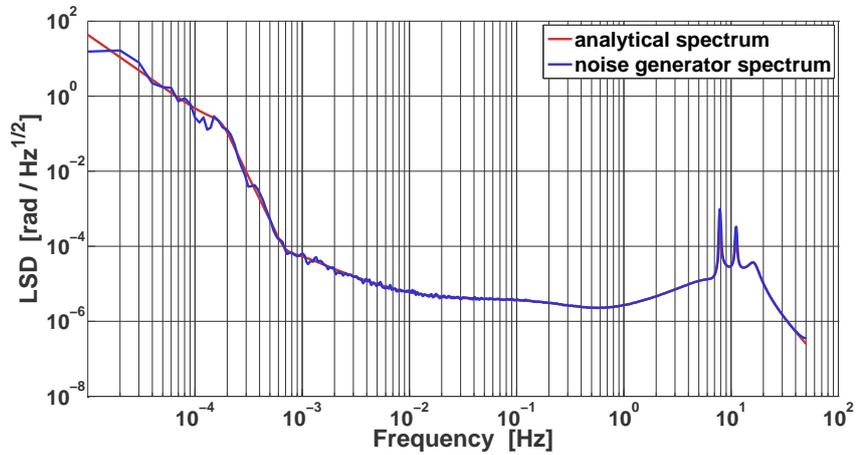
Figure 2.10: Example for an LTP interferometer spectrum. The peaks in the spectrum are also modelled. The time-series underlying the noise generator spectrum can only be computed with a precision higher than MATLAB standard double. Here 32 digits have been used instead of 16.
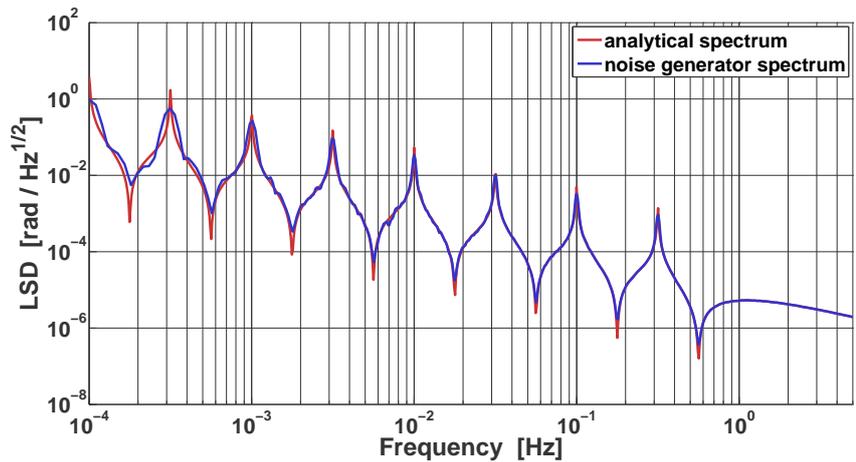
Figure 2.11: Generated time-series for artificially complicated spectrum in Figure 2.11. The time-series can only be computed with a precision higher than MATLAB standard double. Here 32 digits have been used instead of 16.
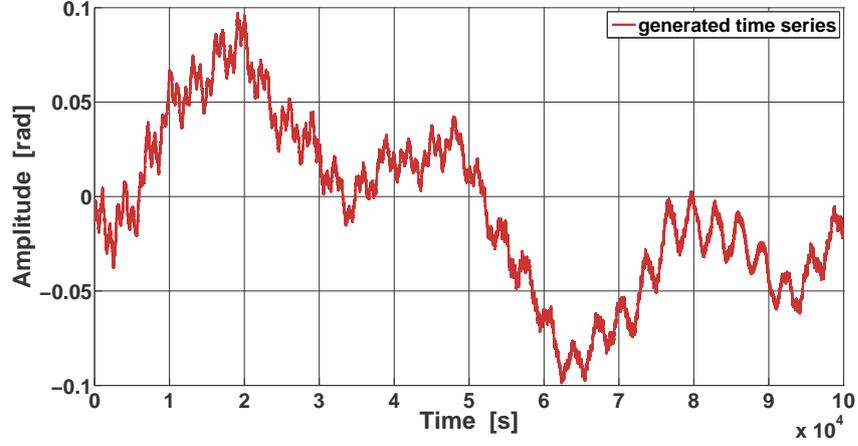
Figure 2.12: Generated time-series underlying artificially complicated spectrum containing many peaks and a large range of frequencies where poles and zeros are located. The time-series can only be generated with a precision higher than MATLAB standard double. Here 32 digits have been used instead of 16.

Table 2.2: Pole-zero model for an artificially difficult spectrum. The individual poles and zeros are defined by a frequency and a quality factor Q. Poles or zeros without a given Q are real. The gain of the transfer function is $g = 1$ and the sampling frequency is $f_s = 100$.

Poles		Zeros	
f [Hz]	Q	f [Hz]	Q
10^{-5}	100	1.778×10^{-5}	100
3.162×10^{-5}	100	5.623×10^{-5}	100
10^{-4}	100	1.778×10^{-4}	100
3.162×10^{-4}	100	5.623×10^{-4}	100
10^{-4}	100	1.778×10^{-3}	100
3.162×10^{-3}	100	5.623×10^{-3}	100
0.01	100	1.778×10^{-2}	100
3.162×10^{-2}	100	5.623×10^{-2}	100
0.1	100	1.778×10^{-1}	100
3.162×10^{-1}	100	5.623×10^{-1}	100

before the internal memory settles to a stationary state [27].

The advantage of a method involving filtering is, on the other hand, that it is usually simpler, both in implementation and computation in terms of processing power. However, the matrices for Franklin's method need to be computed only once and after that the processor power needed is similar to the method of filtering.

2.6 Conclusion

The object oriented programming concept has paved the way for LTPDA to become the official analysis software tool for the entire LISA Pathfinder mission. Since LTPDA is a comprehensive and reliable tool it has become popular beyond the preparation of the data analysis for LISA Pathfinder. It is not only used extensively for the data analysis of the LISA Pathfinder component tests but for all kinds of laboratory experiments throughout scientific groups around Europe. These numerous users, especially the scientists, working in the lab on LPF experiments, greatly accelerated the development process.

A first version of LTPDA has been delivered to ESA in 2009. As such it has passed numerous tests, listed in this section. The third version is planned to contain all tools necessary to analyse the mission data and it is scheduled to be delivered in 2010.

Numerous tools have been newly developed for LTPDA or have been integrated from other projects. One such tool has been picked out to be presented in this section; the Franklin noise generator, a useful tool for generating a time-series with a prescribed spectrum. It has been implemented in MATLAB and integrated into LTPDA. It turns out to have a couple of advantages over the conventional method of filtering a time-series with a transfer function of the desired spectrum.

3 Characterisation of the optical metrology system

The optical metrology system (OMS) is one key part of the LISA Technology Package. It comprises the system units responsible for the measurement of the distance between the two test masses.

In this chapter key points of the laboratory data analysis on the OMS that has been accomplished in preparation of the mission will be presented. And the focus will lie on the development of analysis tools and algorithms in LTPDA, the software tool for the mission which was introduced in the previous chapter. The system will be introduced to an extent that makes it possible to understand the data analysis accomplished. Further information on the optical metrology system can be found, for example in [29], [30] and [31].

All experiments presented in this chapter have been performed on the Engineering Model (EM) of the optical bench for LISA Pathfinder. This optical bench was designed at the AEI Hannover and constructed at the Rutherford Appelton Laboratories in collaboration with the LISA Pathfinder groups from the University of Glasgow and the AEI Hannover.

All analyses presented have been carried out completely using LTPDA (see Chapter 2).

3.1 Introduction to the OMS

The optical metrology system comprises

- four interferometers situated on one optical bench,

- the laser and laser modulator (laser assembly),

- the Phasemeter and

- the Data Management Unit (DMU).

For understanding the purpose of the individual units as well as their interaction, the main measurement principle will be explained. At first, the LISA Pathfinder interferometry will be introduced and moreover, the optical bench with its four interferometers will be presented. Finally the process of the phase readout accomplished via the phasemeter and process by the DMU is explained.

3.1.1 The LISA Pathfinder interferometry

The displacement and attitude of the two test masses on-board LISA Pathfinder is measured interferometrically. The setup realising the length measurement comprises

a heterodyne Mach-Zehnder laser interferometer, a phasemeter and the DMU which together transform the measured signals to the desired length measurement.

Heterodyne in contrast to homodyne interferometry makes use of two laser beams featuring a frequency difference leading to a beat signal at the detector. Mathematically an expression for the interference of two beams at a beam splitter can be derived by combining their electromagnetic fields [31, 32]. The fields can be described as

$$E_1(t) = A \cdot e^{i(\omega_1 t + \phi_1)} \tag{3.1}$$

$$E_2(t) = A \cdot e^{i(\omega_2 t + \phi_2)}, \tag{3.2}$$

where A denotes the amplitude (for simplicity here the same amplitude is assumed for both beams), ω the angular frequency and ϕ the phase. The intensity of the two interfering beams that will be measured as photocurrent at the detector is given by

$$I = |E_1 + E_2|^2 \tag{3.3}$$

$$= A \left(1 - c \, \cos(\omega_{\mathrm{het}} t + \phi(t))\right), \tag{3.4}$$

where the interferometer contrast, c, has been introduced for taking imperfections in the interferometer into account. Furthermore A is the average photocurrent of the heterodyne signal, $\omega_{\mathrm{het}} = \omega_1 - \omega_2$ is the angular heterodyne frequency and $\phi = \phi_1 - \phi_2$ denotes the phase difference.

With $\omega_1 \neq \omega_2$ the time dependency of Equation 3.4 is kept. The controlled time dependency results in a dynamic range for the interferometer of multiples of the laser frequency. For comparison, in homodyne interferometry the predefined operating point limits the dynamic range to a fraction of an interference fringe [31, 30, 32, 29].

The phase difference ϕ contains the main measurement, the optical change ΔL:

$$\phi = \frac{2\pi}{\lambda} \Delta L, \tag{3.5}$$

where λ is the average wavelength of the two laser beams and ΔL is the difference in length of the two interferometer arms.

For the LTP setup the two frequency shifted laser beams are realised by the so-called laser modulation: one laser beam is split into two beams via a beam splitter. Subsequently a frequency difference between the two beams is introduced by making them pass through individual Acousto-Optic Modulators (AOM) operating at 80 MHz $\pm$ 0.5 kHz. The resulting frequency shift is of the order of 1 kHz.

This so-called modulation of the laser beam is physically separated from the ultra stable optical bench. It takes place on the 'modulation bench' and the beams are transferred, via optical fibres, to the optical bench. The setup of the modulation bench is illustrated in Figure 3.1.

The optical bench comprises four interferometers.

The reference interferometer: It is shown in Figure 3.2(a). It is situated completely on the optical bench and as such detects common disturbances of all interferometers. By subtracting its signal from a different interferometer signal
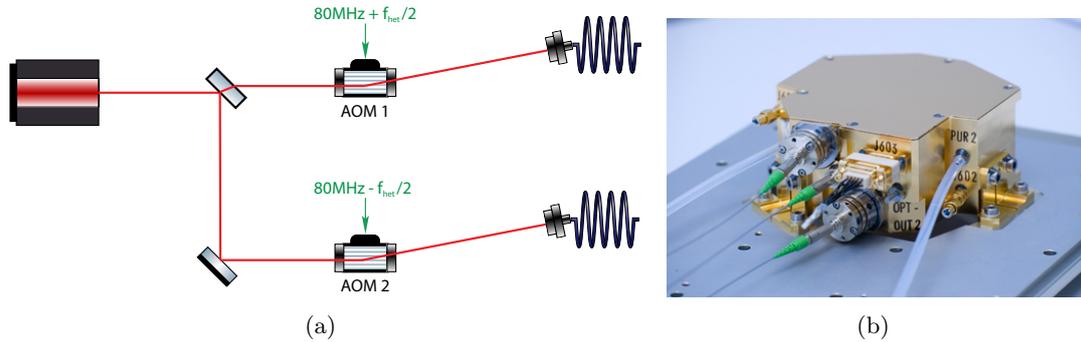
Figure 3.1: (a): Schematic of the LTP modulation bench: One laser beam is split into two beams. Both beams pass through AOMs that operate at 80 MHz with a slight mismatch yielding the heterodyne frequency. The output beams feature a frequency difference of about 1 kHz needed for the heterodyne interferometry on the ultra stable optical bench (not included in the picture). (b) The flight hardware of the laser modulator unit as provided by Kayser Threde (D) / Max Planck Albert Einstein Institute. Copyright: Kayser Threde.

these disturbances are removed in the resulting signal. The common disturbances are mainly introduced from the more unstable modulation bench. The optical bench itself is made of Zerodur® and therefore rather insensitive to the main effect causing mechanical instability - thermal fluctuations.

The frequency stabilisation interferometer: This interferometer is shown in Figure 3.2(b). It measures frequency fluctuations in the laser beam via an intentional pathlength difference and can therefore be used for a laser frequency stabilisation.

The measurement interferometers: Two of the four interferometers are often indexed by an 'M' for indicating the measurement interferometers. They are designed to detect displacements of both test masses. In the laboratory experiment where the Engineering Model of the LTP optical bench is used, the test masses are replaced by piezo-actuated mirrors. The piezos can be used to change the longitudinal as well as the angular position of the mirrors.

   The test mass 1 interferometer: The so-called X1 interferometer. It measures the displacement of test mass 1 with respect to the optical bench. In Figure 3.3(a) the optical path of this interferometer is shown.

   The differential interferometer: It is also called the X12 interferometer and measures the distance between the two test masses. It can be seen from Figure 3.3(b) that this is a direct measurement rather than the result from a measurement of the individual positions of the two test masses.

3.1.2 The data processing

Quadrant photodiodes are used for the detection of the interference signal. The measured photo currents are converted into voltages by transimpedance amplifiers

(a) The reference interferometer   (b) The frequency interferometer
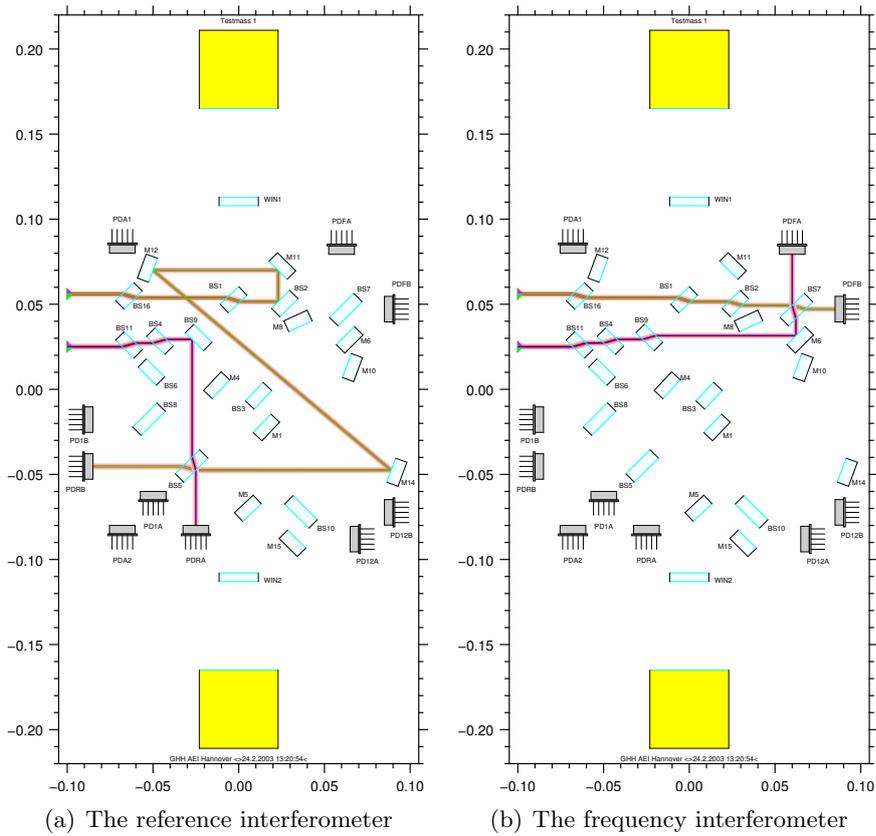
Figure 3.2: Schematic of the reference (a) and frequency (b) interferometers. The optical
          path is shown individually for each interferometer but the optical bench is
          constructed such that all four interferometers (see Figure 3.3) are situated on
          one base plate. See caption of Figure 3.3 on next page also.

(a) The X1 interferometer   (b) The X12 interferometer
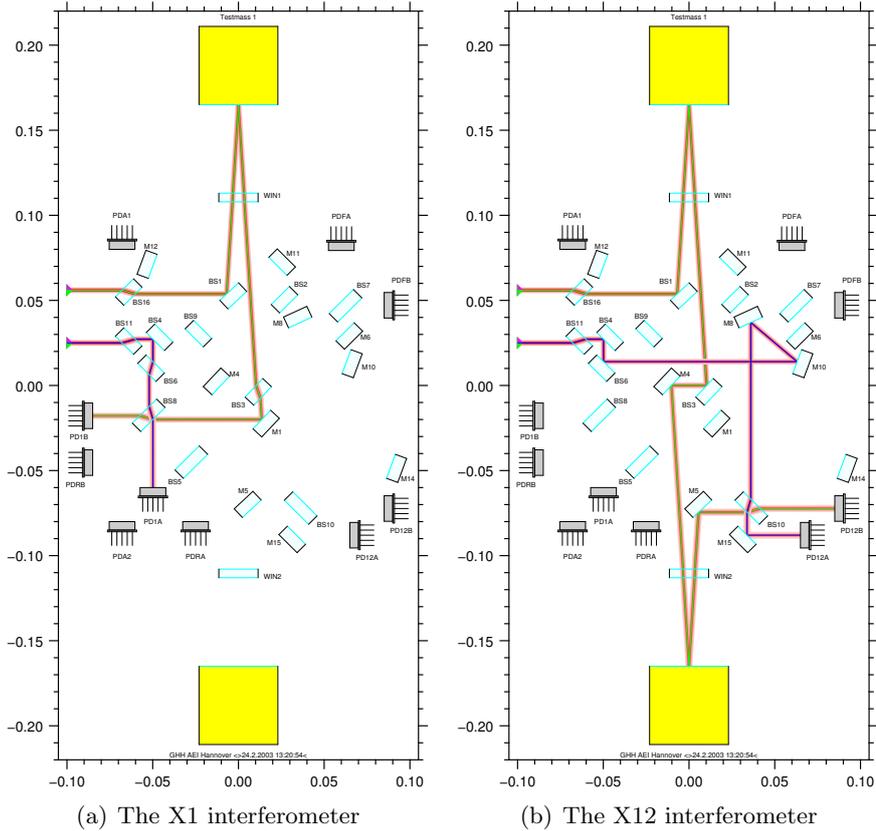
Figure 3.3: The schematic of the two measurement interferometers situated on the optical bench for LTP are shown. In a) the X1 interferometer is shown and in b) the differential interferometer X12 is presented. The optical path is shown individually for each interferometer but the optical bench is constructed such that all four interferometers are situated on one base plate. The laser beams are transferred to the optical bench via fibre optics from the modulation bench (not included here). The baseplate and the components are made of Zerodur and the bench was constructed using hydroxide-catalysis bonding techniques [30]. In the laboratory experiments the test masses are replaced by piezo-actuated mirrors.

which is fed into the phasemeter. The role of the phasemeter is to perform a 'single bin discrete Fourier transform' (SBDFT) on the digitised input signals (the original interference signals). Single bin here means that the Fourier transform for each segment containing N samples is only computed at the frequency bin corresponding to the heterodyne frequency, f_{het}. The result is a complex vector representing the amplitude of the photodiode signal at f_{het}. A simplified representation of the real and imaginary parts of the vector is as follows.

$$\Re\{F(f_{\text{het}})\} = \frac{1}{N} \sum_{i=0}^{N-1} \cos\left(2\pi f_{\text{het}} \frac{i}{f_{\text{samp}}}\right) \cdot x_i(t_i), \tag{3.6}$$

$$\Im\{F(f_{\text{het}})\} = \frac{1}{N} \sum_{i=0}^{N-1} \sin\left(2\pi f_{\text{het}} \frac{i}{f_{\text{samp}}}\right) \cdot x_i(t_i), \tag{3.7}$$

where x_i is the amplifier output at time t_i. The DC mean value of the signal is calculated according to

$$DC = \frac{1}{N} \sum_{i=0}^{N-1} x_i(t_i). \tag{3.8}$$

These computations are implemented in the Field Programmable Gate Array (FPGA) based phasemeter for every quadrant of each photodiode.

The final data processing steps are implemented inside the Data Management Unit (DMU). Here the signals computed by the phasemeter are converted to the longitudinal phase and alignment signals.

Longitudinal measurement   The longitudinal phase ϕ is defined as:

$$\phi = arg(F_{\sum}) + n2\pi, \tag{3.9}$$

where $F_{\sum}$ represents the sum of the relevant quadrants of the photodiode signals and the term $n2\pi$ results from a phasetracking algorithm [33, 29].

In Equation 3.5 the general dependency of the measured phase, ϕ, and the pathlength difference, ΔL, was given. On the optical bench of LTP, four interferometers with four different optical paths are situated. The individual pathlengths are as indicated in Figure 3.4. Thus the following phases shall be introduced:

$$\Delta_{\text{F}} = \frac{2\pi}{\lambda}(L_1 - L_2) \tag{3.10}$$

$$\Delta_{\text{R}} = \frac{2\pi}{\lambda}(L_{1\text{R}} - L_{2\text{R}}) \tag{3.11}$$

$$\Delta_{\text{M}} = \frac{2\pi}{\lambda}(L_{1\text{M}} - L_{2\text{M}}), \tag{3.12}$$

where the different variables L_x represent the pathlengths of the respective interferometer arms. A more detailed description of these can be found in Chapter 2 of [15]. It follows that the given phases Δ_x represent pathlength fluctuations in the associated optical paths. The phase Δ_{F} refers to the pathlength difference in the common feed paths for all interferometers. This originates mostly from fluctuations
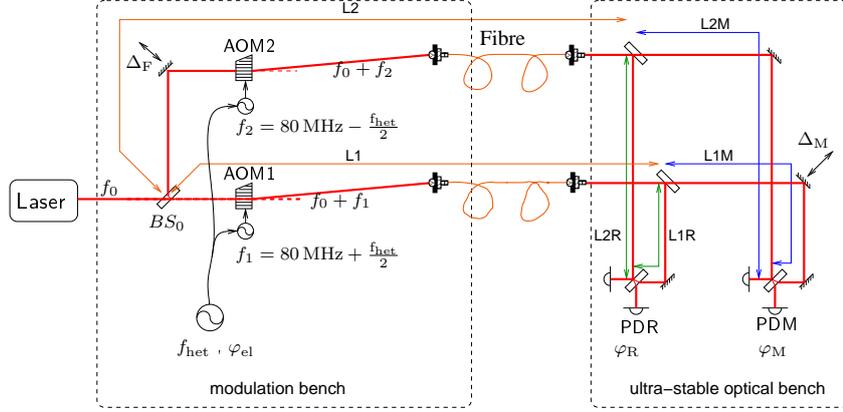
Figure 3.4: Schematic of the heterodyne interferometer used in LISA Pathfinder. The optical pathlengths used in Equations 3.10 - 3.12 are indicated by L_1, L_2, L_{1R}, L_{2R}, L_{1M} and L_{2M}.

of the fibres guiding the laser beams from the unstable modulation bench to the ultra stable optical bench. The fluctuations Δ_R and Δ_M refer to the stable reference interferometer and the two stable measurement interferometers respectively. From these relations the primary observables can be written as:

$$\phi_R = \Delta_F + \Delta_R, \tag{3.13}$$
$$\phi_M = \Delta_F + \Delta_M. \tag{3.14}$$

From these computed longitudinal phases, the main interferometric measurement is obtained. The reference phase, ϕ_R, is subtracted from the other phases to cancel the environmental noise which is present in all measurements, resulting in the "Ψ" variables. For test mass 1 the phase measurement of the displacement with respect to the optical bench is denoted by Ψ_1 and the differential phase measurement is denoted by Ψ_{12}.

$$\Psi_1 = \phi_1 - \phi_R, \tag{3.15}$$
$$\Psi_{12} = \phi_{12} - \phi_R. \tag{3.16}$$

Finally the phase measurements, Ψ_1 and Ψ_{12}, can be transformed to length measurements. For this operation the incident angle of the measurement beam on the test masses has to be taken into account. Within this thesis the length measurement will be denoted by o but often it is also called x.

$$o_1 = x_1 = \frac{\lambda}{4\pi \cos\alpha}\Psi_1 = \frac{\lambda}{4\pi \cos\alpha}(\phi_1 - \phi_R), \tag{3.17}$$
$$o_{12} = x_{12} = \frac{\lambda}{4\pi \cos\alpha}\Psi_{12} = \frac{\lambda}{4\pi \cos\alpha}(\phi_{12} - \phi_R), \tag{3.18}$$

where α represents the incident angle.

Alignment measurement In addition to the longitudinal displacement of the test masses described above, alignment measurements give information on the orientation of the test masses in two degrees of freedom. Figure 3.5 shows the coordinate
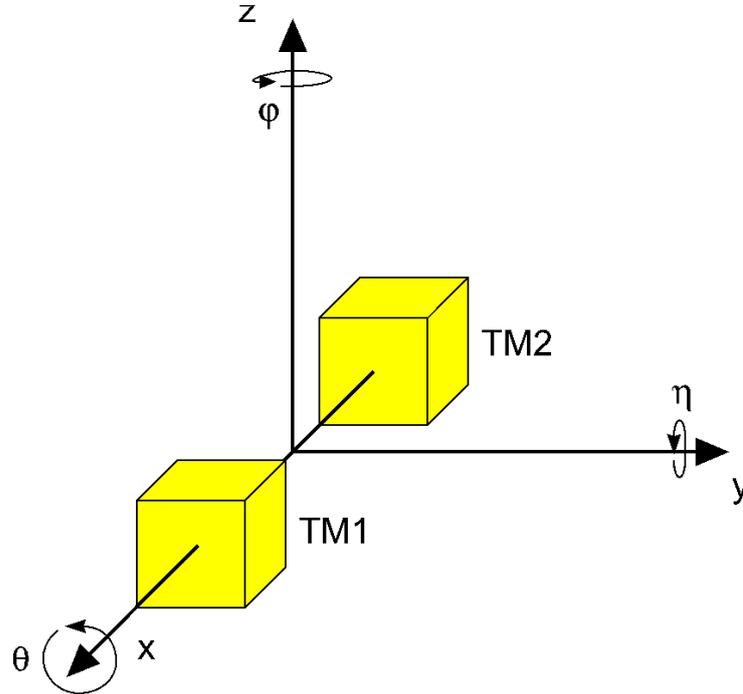
Figure 3.5: Coordinate frame for LISA Pathfinder

frame for LISA Pathfinder and illustrates the orientation of the test masses. The angles φ and η are measured while the interferometer is insensitive to θ. These measurements are as well obtained from the outputs of the phasemeter. Two different approaches lead to the angular motion of the test masses:

- Estimating the beam position on the quadrant photodiode. This is done from the DC measurement of the differential light power contribution on the individual quadrants.

- The so-called Differential Wavefront Sensing (DWS). This more sensitive method is explained in detail in for example [30]. Here only the basic principle will be mentioned: The method is based on the analysis of the wavefronts of the beams arriving at the photodiodes. The beam corresponding to the reference interferometer is assumed to be fixed. The wavefront of a measurement beam is analysed with respect to the one of the reference beam and the differential phases between the different halves of the quadrant photo diode correspond to the angular displacement of the corresponding test mass.

The DWS signals are about 10 times more sensitive than the DC alignment signals [29]. Estimating the beam position by analysing the light power on the individual quadrants of the photodiode is very robust. Sensible results are obtained with low light power impinging the photodiode. Hence it is well suited for the initial alignment of the interferometer [29].

3.2 Angular noise subtraction

It was described above that the interferometer measurements give information on the longitudinal as well as on the angular displacement of the test masses. In fact, these two measurements are correlated: The angular motion of the test masses couples into the longitudinal measurement. The reason for the coupling of the angular motion of the test masses lies in the imperfect alignment of laser beam at the centre of rotation of the test masses.

The aim of the experiment described in this section is to probe the impact of angular test mass movement on the sensitivity of the interferometric measurement. For this, simulations resulting in spectral predictions of the test mass angular noise expected on orbit, were performed [34]. By means of 'Franklin's noise generator' (see Chapter 2 and [27]) time series matching these simulated spectra were generated. As such the generated data represents a time dependent angular movement of the test masses. These time series served as input signal for the piezo actuated end-mirrors of the interferometer representing the test masses. Finally interferometric measurements were carried out using the DWS signals, described above, for measuring the angular displacement. Figure 3.6 shows the spectra of the measured angular motion together with the spectral prediction from the simulations accomplished beforehand. Both spectra are in good agreement.
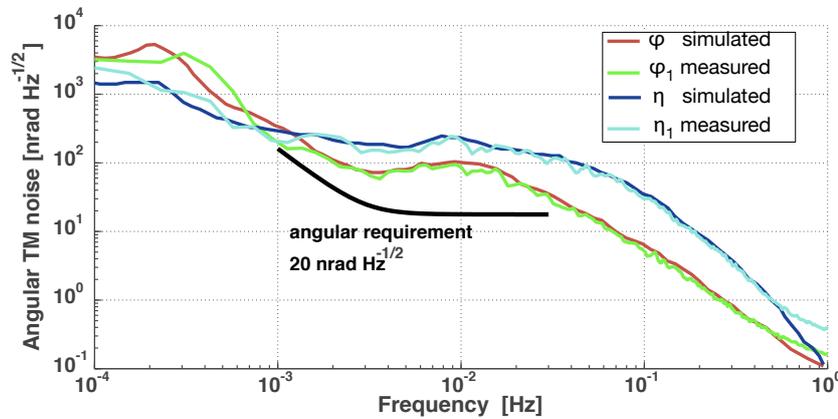
Figure 3.6: Angular test mass noise. Noise with a prescribed spectrum from a simulation was applied to the test masses. The measurements from the differential wavefront sensing (DWS) agree with the desired simulated spectrum. Shown is the angular measurement of test mass 1 but the same noise is applied to both test masses.

At the end of this section the results of the noise subtraction are plotted in Figure 3.9 together with the sensitivity curve of the longitudinal measurement. It can be seen that the sensitivity is lowered by a significant amount.

In this section a method for removing this residual angular noise from the measured data will be presented. The focus will be on the algorithm development for the software tool LTPDA, presented in Chapter 2.

3.3 Development of the algorithm for angular noise subtraction in LTPDA

The angular test mass motion couples into the longitudinal measurement. For removing the noise originating from this displacement, a linear fit is applied to compute the coupling factors explicitly. These coupling factors represent the relation between the noise introduced and the final longitudinal measurement. As such they are used to identify the noise contribution and subtract it from the measured longitudinal phase. The procedure has been published in [35].

The longitudinal measurement can be expressed as:

$$\Psi_{\mathrm{m}} = \Psi_{\mathrm{true}} + \Psi_{\mathrm{ang}}, \tag{3.19}$$

where Ψ_{m} is the measured longitudinal phase, Ψ_{true} is the true longitudinal contribution to the measurement and Ψ_{ang} represents the angular noise coupling into the longitudinal measurement.

A measurable correlation between the angular and longitudinal displacement of the test masses, however, exists only in a limited frequency region of the measurement. As such the linear model for the correlation entering the fitting procedure is also only applicable in this region. Therefor, the data is bandpass-filtered in the region of interest prior to the fitting procedure. All data streams involved in the fitting procedure are filtered such that the frequencies irrelevant for the angular noise contribution are suppressed. The frequency response for the applied bandpass filter is plotted in Figure 3.7. The filter is constructed such that frequencies in the region between $f_1 = 3 \times 10^{-3}$ Hz and $f_2 = 3 \times 10^{-2}$ Hz are passing unsuppressed. In this region the residual angular test mass noise is dominating the longitudinal measurement.
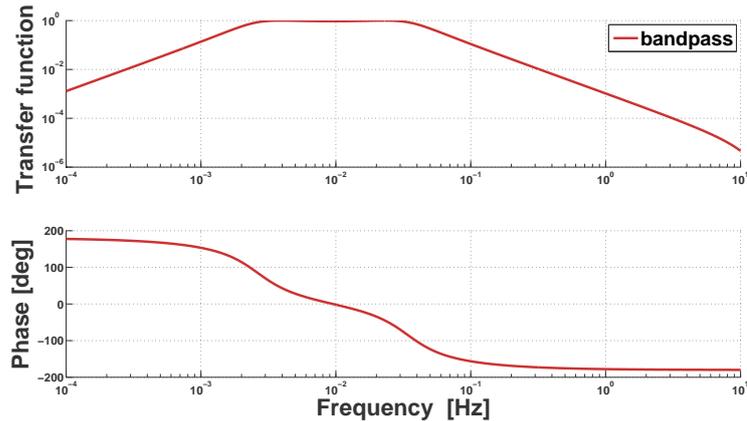
Figure 3.7: Frequency response of the bandpass filter applied to the interferometer measurement data.

In the following the rotation of the test masses will be described by the angles η and φ as illustrated in Figure 3.5. Hereby the rotations for the individual test masses will be denoted by η_1 and φ_1 for TM1 and by η_{12} and φ_{12} for the differential rotation. The linear model describing the residual angular test mass noise is different for

the two interferometer read-out channels. In the measurement of test mass 1 only the TM1 angles φ_1 and η_1 are involved whereas all four angles couple into the differential measurement. The resulting equations for the linear model to be fitted to the filtered data reads for the two output channels respectively:

$$\Psi_{\mathrm{ang}_1} = \alpha_1 \cdot \varphi_1 + \beta_1 \cdot \eta_1, \tag{3.20}$$

$$\Psi_{\mathrm{ang}_{12}} = \alpha_2 \cdot \varphi_1 + \beta_2 \cdot \eta_1 + \gamma_2 \cdot \varphi_{12} + \delta_2 \cdot \eta_{12}, \tag{3.21}$$

where α_i, β_i, γ_2 and δ_2 are the coupling factors for the angles of rotation. Measurements of the angles are taken by the interferometer and the coupling factors are to be estimated by the linear fitting routine.

The applied estimation method is the 'linear least squares method'. MATLAB supplies a function for this, called `lscov` and it was overloaded to work as an LTPDA method (see Chapter 2). The LTPDA method is called `lscov` as well.

The linear least squares method is the subject of many data analysis textbooks like [36] and various ways of implementation are discussed in the collection of numerical methods in [28]. Also the main principle is introduced in Chapter 4, while the special case of 'Singular value decomposition' (SVD) is presented in detail in Section 4.6.4. Nevertheless, the basic idea will be repeated here briefly to present the application to the given problem, the estimation of the coupling factors of the angular test-mass motion into the longitudinal measurement.

The task of a fitting algorithm based on the general least squares method is to construct a so-called merit function according to the given model. This function relates the measurements, y_i with the model denoted by f which depends on the unknown parameters, $\vec{\theta}$.

$$r(\vec{x}, \vec{y}, \vec{\theta}) = \sum_i (y_i - f(x_i, \vec{\theta}))^2. \tag{3.22}$$

The model function for Ψ_{ang} was defined in Equation 3.21. The optimal parameter set, θ_{opt}, minimises the residual function. As such the defined merit function r needs to be minimised to find the optimal parameter set. In the following equation, the expression 'argmin' indicates the minimum of the function for the parameter set $\vec{\theta}$.

$$\vec{\theta}_{\mathrm{opt}} = \underset{\vec{\theta}}{\mathrm{argmin}}(r(\vec{x}, \vec{y}, \vec{\theta})). \tag{3.23}$$

In the case that the model $f(x_i, \vec{\theta}))^2$ is linear in its parameters $\vec{\theta}$, the task of minimisation reduces to solving a linear system of equations.

$$\mathbf{A}\vec{\theta} = \vec{b}, \tag{3.24}$$

where $\vec{\theta}$ stands for the optimal parameter set to be estimated, $\vec{b}$ represents the measurement of the longitudinal phase Ψ and the matrix $\mathbf{A}$ is called the 'design matrix'. It contains the angular measurements. For the case of the differential longitudinal measurement Ψ_{12} these are the DWS signals of all four angles: φ_1, η_1, φ_{12} and η_{12}. Thus, Equation 3.24 relates the angular measurements to the longitudinal

measurement via the desired coupling factors $\vec{\theta}$. Solving this system for $\vec{\theta}$ provides the optimal parameter set.

This is done by using the LTPDA method `lscov`. For solving the system a number of methods exist. One of them is the above mentioned SVD (see Section 4.6.4). The method used by `lscov` is by default the *cholesky* decomposition [28], which was mentioned in Section 2.5.

For the task at hand, the input data for the function `lscov` in LTPDA are the following:

$\varphi_1, \eta_1, \varphi_{12}$ and η_{12}:   The measured DWS signals of the four angles (see Figure 3.5).

Ψ_{12}:   The differential longitudinal phase measurement.

An example for the function call in LTPDA for computing the coupling factors for the differential longitudinal test mass motion is as follows:

```
%% Loading data
psi_meas = ao('psi12.txt');
phi1 = ao('phi1.txt');
eta1  = ao('eta1.txt');
phi12 = ao('phi12.txt');
eta12 = ao('eta12.txt');
%% Constructing filter
bp = mfir(plist('type', 'bandpass', 'order', 1, ...
'gain', 1,'fs',phir.fs, 'fc', [0.4e-3 1]);
%% Bandpass filtering
psi_measf = filter(psi_meas,bp);
phi1f     = filter(phi1,bp);
eta1f     = filter(eta1,bp);
phi12f    = filter(phi12,bp);
eta12f    = filter(eta12,bp);
%% Function call for linear parameter estimation.
c = lscov(phi1f, eta1f, phi12f, eta12f ,psi_measf);
```

In the code shown above `phi1`, `eta1`, `phi12` and `eta12` are Analysis Objects containing the time series corresponding to φ_1, η_1, φ_{12} and η_{12} respectively and `psi_meas` is an AO corresponding to the time series of Ψ_{12}.

The result of the above call are the coupling factors α_2, β_2, γ_2 and δ_2. They are stored in the AO, c, together with subsidiary information like the individual standard deviation and the covariance matrix [36] (see also Section 4.6.4).

After the coefficients have been estimated, they can be inserted into the equation for the contribution of the angular displacement measurement, Equation 3.21. However, in contrast to the the linear fitting routine, where the bandpass filtered data for the angular and longitudinal measurements have been used, the computation of Ψ_{ang} is done using the original, unfiltered data. In LTPDA the linear combination of the computed coupling factors and the angular measurement is performed by the function `lincom`. The inputs to the function are the variable c, containing the coupling factors and the unfiltered angular measurements. With the result of Ψ_{ang} the longitudinal phase measurement can be corrected by simply subtracting the

angular contribution in time domain. From Equation 3.19 it follows:

$$\Psi_{\text{true}} = \Psi_{\text{measured}} - \Psi_{\text{ang}}, \tag{3.25}$$

In LTPDA the corresponding code is the following:

```
%% Linearly combine fitted coefficients, c, with unfiltered data
psi_ang = lincom(phi1, eta1, phi12, eta12, c);
%% Subtracting noise contribution from measurement
psi_true = psi_meas - psi_ang;
```

In the above code **psi_ang** represents Ψ_{ang} and **c** is an AO containing the fitted coupling factors. Figure 3.8 illustrates the procedure of the estimation and final subtraction of the angular noise.
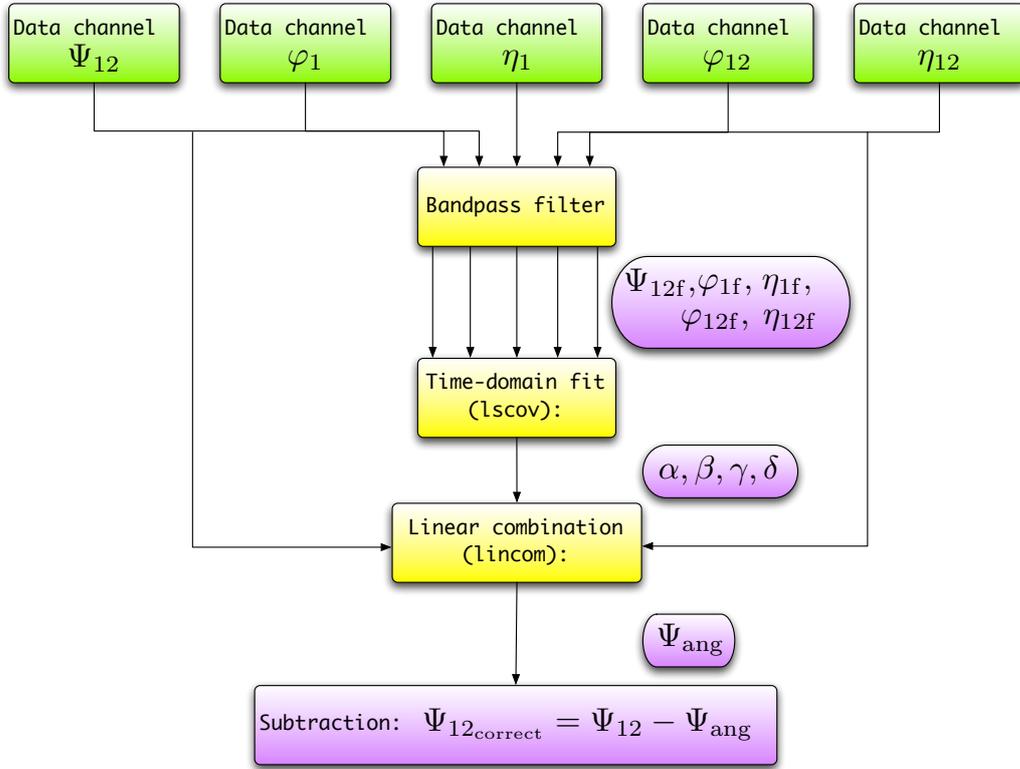
Figure 3.8: Illustration of the angular noise subtraction in LTPDA. The purple boxes represent the results of the respective method. All variables (purple and yellow) are Analysis Objects containing time series data. The yellow boxes depict LTPDA methods and the final subtraction is a simple operation on Analysis Objects to be treated like any common variable.

Experimental results The sensitivity curves with angular noise injected and the one where the noise has been subtracted, Ψ_{true} are plotted in Figure 3.9. For comparison a measurement, where no noise was applied to the test masses is plotted

as well. It can be seen that the corrected longitudinal measurement reaches approximately the same level as the reference measurement for the region where the angular noise impacts and the requirement for the interferometer is met.
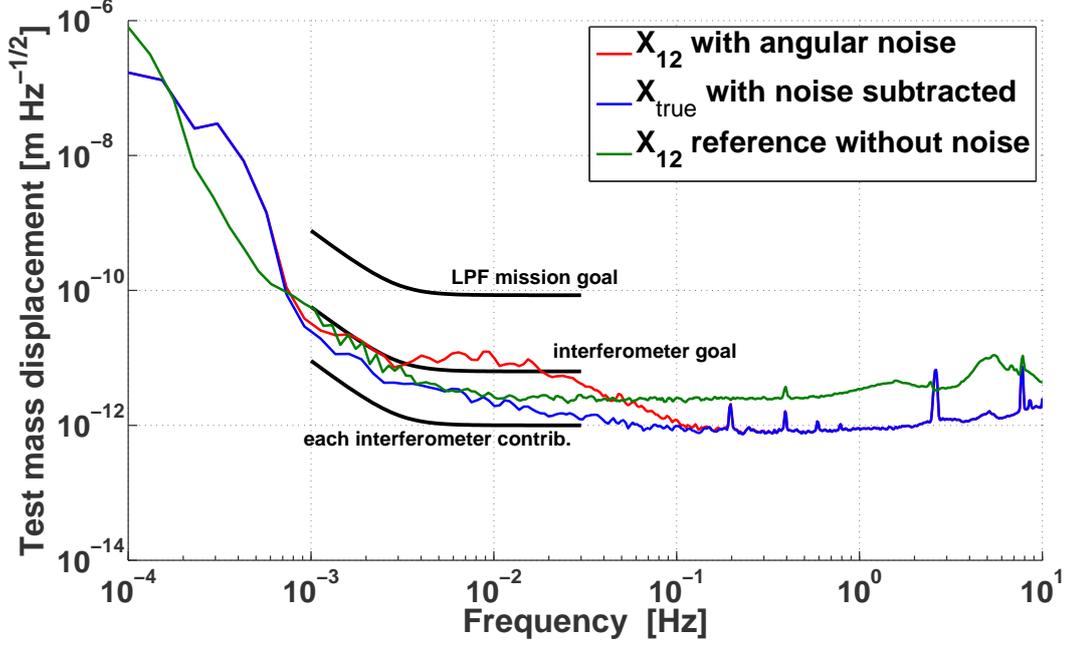
Figure 3.9: The spectral densities of the longitudinal phase measurements converted to meters. The red curve represents the measurement with injected angular noise on the test masses. The blue curve is the spectral density of the corrected measurement. The green curve belongs to a reference measurement, where no angular noise was injected.

3.4 Characterisation of the optical pathlength difference

The computation of the longitudinal measurement was derived in the introduction to this chapter. The pathlength fluctuations were introduced and will be repeated here for the sake of readability:

$$\phi_{\mathrm{R}} = \Delta_{\mathrm{F}} + \Delta_{\mathrm{R}}, \tag{3.26}$$

$$\phi_{\mathrm{M}} = \Delta_{\mathrm{F}} + \Delta_{\mathrm{M}}, \tag{3.27}$$

$$\Psi_{\mathrm{M}} = \phi_{\mathrm{M}} - \phi_{\mathrm{R}}. \tag{3.28}$$

where ϕ_{M} denotes the two measurement phases. In the ideal case the optical pathlength difference, Δ_{F}, cancels when computing Ψ_{M}. Further noise studies, however, showed that the Ψ_{M} variables are not completely independent of the optical path fluctuation Δ_{F} [15]. The resulting 'sideband induced noise' is the noise source investigated in this section. A model for the noise source will be presented from which we will find the coupling factors to the sensitivity measurement by applying a linear fit similar to the angular noise analysis presented in the previous section.

3.4.1 OPD stabilisation

The studies of the sideband induced noise revealed a coupling of the optical path fluctuation Δ_F into the phase measurement Ψ_M. The investigations on the sideband induced noise in the LISA Technology Package (LTP) has first been published in [37] in 2006. It has been found to be an important noise source for LTP and as such an on-orbit experiment is planned to measure and eventually subtract it from the phase measurement.

In this section a method of subtracting this noise source from the output channel will be presented. However if possible, minimising the noise during the measurement should always be the preferred method over estimating coupling factors and subtracting a noise source. For the sideband induced noise it is possible to suppress the noise driven by Δ_F by stabilising the optical pathlength difference. This is done by phase locking the reference phase to the electronic reference signal ϕ_el. A measurement with stabilised OPD is not investigated in this thesis. The sideband induced noise is instead suppressed by reducing another important factor for this noise, the electronic cross-talk. More detailed explanations on the stabilisation loops can be found in [29] and [30]. While the particular noise mechanism described here is unique to LTP, the ubiquitous noise caused by ghost beams and stray light often has similar characteristics and can be treated similarly, such that these methods may also be applied to other interferometers.

3.4.2 Theory of sideband induced noise (SIN)

It is explained in detail in [15] that the dependency of Ψ on the optical path fluctuation originates in the presence of electrical sidebands introduced by the RF driving signals of the acoustic optical modulators (AOM) used for the generation of the heterodyne signal (as explained in Section 3.1.1). In fact the electrical sidebands cause optical sidebands on the signals measured at the photodetectors. A detailed discussion on how the electrical sidebands convert into optical sidebands is as well given in [15] and will be omitted here. These optical sidebands, on the other hand, produce a spurious interferometer signal, called the sideband induced noise.

The total error due to sidebands can be expresses analytically by the following equation:

$$\delta\Psi = \left\{ b_0 \sin\left(\frac{\phi_\mathrm{R} + \phi_\mathrm{M}}{2}\right) + b_1 \cos\left(\frac{\phi_\mathrm{R} + \phi_\mathrm{M}}{2}\right) \right\} \cdot \sin\left(\frac{\phi_\mathrm{M} - \phi_\mathrm{R}}{2}\right) \tag{3.29}$$

$$+ \left\{ c_0 \sin(\phi_\mathrm{R} + \phi_\mathrm{M}) + c_1 \cos(\phi_\mathrm{R} + \phi_\mathrm{M}) \right\} \cdot \sin(\phi_\mathrm{M} - \phi_\mathrm{R}). \tag{3.30}$$

This analytical function is linear in its parameters and hence a linear fitting routine can be applied for the analysis. The procedure is shown at the end of this section. The error consists of two main contributions:

- The contribution $\phi_\mathrm{M} - \phi_\mathrm{R}$ represents the pathlength changes on the ultra-stable optical bench, which change only slowly. On orbit this corresponds to the varying test mass position, which is expected to be slow as well.

- The contributions from the raw phases: ϕ_R and ϕ_M vary rapidly with the fluctuating optical pathlength difference, Δ_F (see Equations 3.26 and 3.27).

Two different phase measurements will be analysed in terms of sideband induced noise:

A free running performance experiment: An experiment under standard laboratory conditions, where no OPD stabilisation is active. Hence ϕ_R is fluctuating rapidly and so the sideband induced noise limits the measurement sensitivity. The measurement analysed in this thesis was performed in 2005 and was chosen for demonstration purposes because of its significance in terms of the sideband induced error.

An OPD scan: Scanning the OPD means changing the optical pathlength Δ_F in a controlled fashion. For the measurement analysed in this thesis this is achieved by injecting a triangular scanning signal to the piezo. The measurement under investigation in this thesis was taken during the test campaign for the Engineering Model of the laser and laser modulator at AEI. It was measured under idealised conditions with the focus on minimising electrical sidebands caused by electrical cross-talk for example due to cables which cause the sideband induced noise. The conditions might be unrealistic when it comes to the on-orbit experiment but they provide a good indicator for the efficiency of minimising electronic cross-talk in terms of sideband induced noise.

A detailed discussion about the Engineering Model laser and laser modulator test campaign in terms of sideband induced noise is given in the technical document [17].

Fast phase shifts in ϕ_R, either originating from rapid OPD fluctuations or an OPD scan, introduce an additional error due to the Doppler effect. This error term will have to be subtracted from the measurement before further analyses on the sideband induced error can be carried out. The method of subtracting will be discussed in the following.

3.4.3 General remarks on segment selection and the presentation of results

The aim of the analysis presented here is the investigation of the dependency of the measurement phase Ψ_M on the OPD fluctuation Δ_F. As such, in the time domain the effect under investigation is best visualised by plotting Ψ_M against $\phi_R = \Delta_F + \Delta_R$.

Figure 3.10 shows a plot of the measurement phase, Ψ_1, versus the reference phase, ϕ_R. The measurement shown comprises a couple of hours during which Ψ_M is drifting naturally. This effect overlaps the periodic sideband induced error and leads to an ambiguous dependency of ϕ_R. As such the measurement does not represent a function in the mathematical sense, since not all values of Ψ_1 are unique co-values of ϕ_R. For this reason the measurement must be split into segments of well defined functions in order to analyse the relation between the phase measurements properly. The segment splitting is done in the time domain, in parallel for every phase measurement. For the segments two important points are required:

- The segment must be long enough to comprise a variation of at least 2π in ϕ_R.
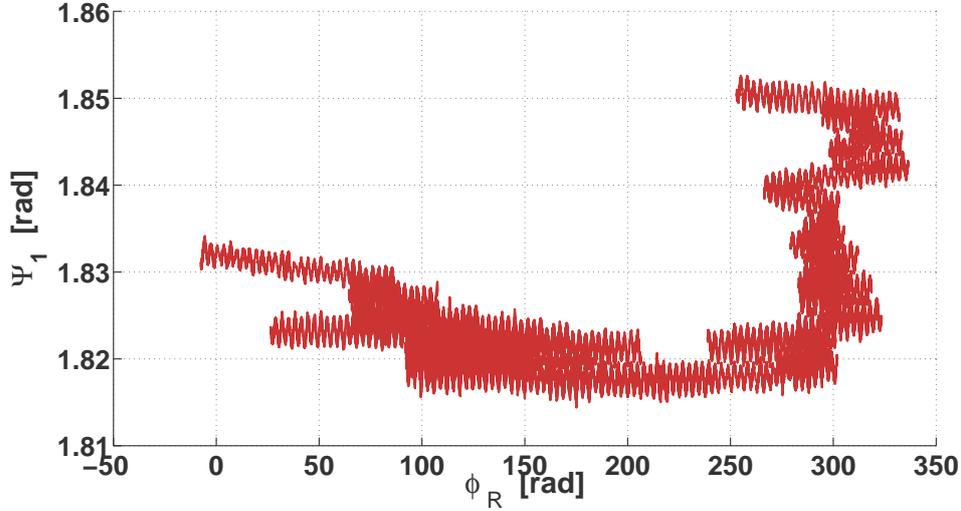
Figure 3.10: $\Psi_1 = \phi_1 - \phi_R$ is plotted as a function of $\phi_R = \Delta_F + \Delta_R$. The plot shows a measurement of a few hours. The natural fluctuation of Ψ_1 has no simple ϕ_R dependence [15].

- The segment must be short enough to allow a linear approximation of the dependency of Ψ_1 on ϕ_R [15].

In the analyses presented in the following a set of segments fulfilling the requirements above is selected from the data and each is fitted to the given model. The results are averaged and the coefficients obtained are inserted back into the model to compute the noise contribution which can then be subtracted from the data.

3.4.4 Doppler correction

The fast phase shifts introduced by the fast OPD fluctuations cause a Doppler shift (see Chapter 9 of [15]). This is due to the fact that the single-bin discrete Fourier transform (SBDFT) as stated in Section 3.1.2 requires the heterodyne frequency, f_{het}, to be centred in the chosen output bin. The rapid OPD fluctuations result in a fluctuation of f_{het} within the bin which resembles a Doppler shift [31] and causes a dterministic error in the phase measurement.

Prior to analysing the sideband induced error of the measurement of the test mass displacement, the error term caused by this Doppler shift must be removed. This is because the Doppler shift has the same effect on the data as the spurious sidebands and would distort the analysis. Fortunately, the Doppler error term is known analytically and can be removed by subtraction from the raw data. The error term depends only on the absolute value of the phase and the Doppler-shift, δ, which is numerically approximated as

$$\delta_i = \frac{\phi_{i+1} - \phi_{i-1}}{4\pi}, \tag{3.31}$$

where i is a counter for the data points. In fact δ represents the time derivative of ϕ. The Doppler induced error term applies likewise to each raw phase, i.e ϕ_M and

(a) Ψ_1

(b) Ψ_{12}
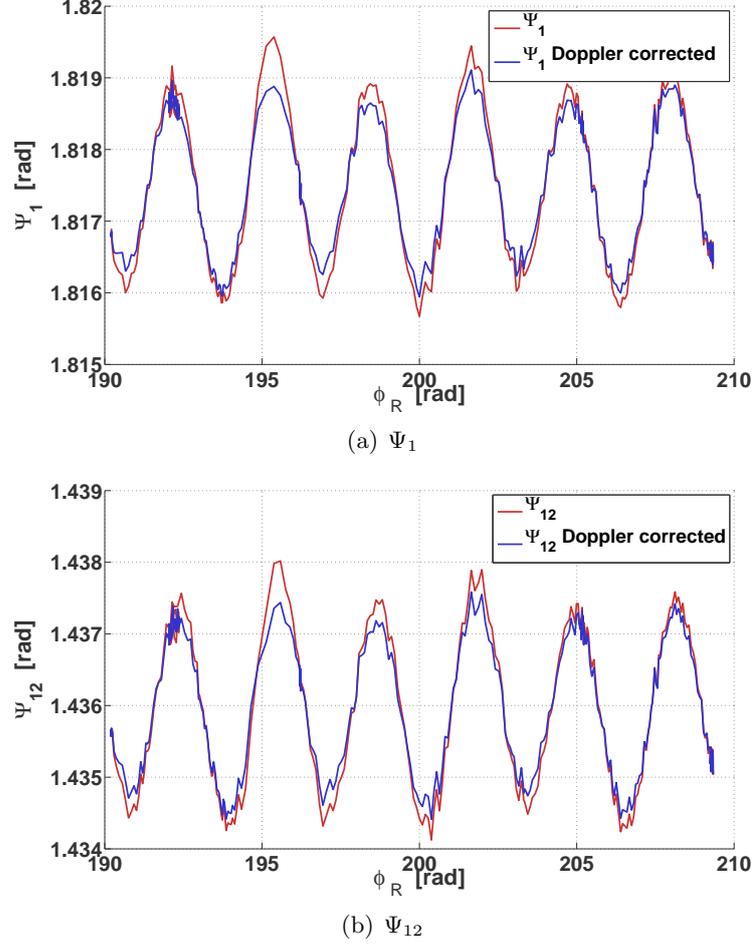
Figure 3.11: Plotted is $\Psi_M = \phi_M - \phi_R$ as a function of ϕ_R. Before (red) and after (blue) the Doppler correction was applied. The measurement was carried out without stabilising the optical pathlength difference (OPD). The Doppler effect originates from the fast fluctuations of Δ_F.

ϕ_R. It will be computed on a grid of $k \times k$ points. In the case presented here $k = 50$ is the bin size. The Doppler contribution is approximated by the following formula taken from [15]:

$$\phi_{\text{Doppler}} = -\frac{1}{2k}\delta\sin(2\phi) + \frac{1}{4k^2}\delta^2\sin(2\phi) + \frac{1}{8k^2}\delta^2\sin(4\phi). \qquad (3.32)$$

The Doppler correction is implemented in LTPDA under the name `dopplercorr`. The method computes the error term according to the equation defined above and subtracts it from the input raw phase. This procedure is applied to all phases relevant for the planned analysis, *i.e.* ϕ_M and ϕ_R. The resulting respective raw phase is:

$$\phi_{\text{corr}} = \phi - \phi_{\text{Doppler}}. \qquad (3.33)$$

The measurement that will be analysed in the following has been performed without OPD stabilisation and under ordinary laboratory conditions. This means

that indeed fast OPD fluctuations are present. In Figure 3.11, the effect of the Doppler correction on the described relation is shown. The amplitude of the signal with respect to the reference phase ϕ_R is reduced. In fact the amplitude of the signal is the amplitude of the OPD noise plus the error term introduced by the Doppler effect. Hence if the Doppler induced error would not be removed from the data, it would distort the result.

The Doppler correction can always be applied to the data because its contribution is zero if no Doppler shift is present.

3.4.5 The estimation of the sideband induced noise (SIN) and the correction of the measurement data

After this post-processing stage, the measured phases can be analysed in terms of the sideband induced errors. In this section results of both experiments with and without OPD stabilisation, will be presented. The measurement performed without stabilisation was performed in 2005. More recent measurements exist but the chosen one is well suited for demonstration. A second experiment with active OPD and frequency stabilisation has been accomplished as part of the tests of the Engineering Model of the laser modulator at AEI in 2009.

As explained above, the first step of the procedure is the Doppler correction of all phase measurements involved (ϕ_R and ϕ_M). After this we can approximate the sideband induced noise term using the model given in Equation 3.30. The model is linear in its parameters and hence a linear fitting routine can be applied to fit the measurement to the model. As for the angular noise fit in the previous section, the LTPDA method `lscov`, is used. It performs a linear least squares estimation. The method description is given in [23] and a brief summary on the linear least squares method was already given in Section 4.6.4.

The complete model the data is fitted to comprises the natural drift of ϕ_M with ϕ_R and the model given in Equation 3.30. It reads

$$
\Psi_M = \overbrace{a_0 + a_1 \cdot \phi_R + a_2 \cdot \phi_R^2}^{\text{natural evolution of } \phi_R}
$$
$$
+ \left\{ b_0 \cdot \sin\left(\frac{\phi_R + \phi_M}{2}\right) + b_1 \cdot \cos\left(\frac{\phi_R + \phi_M}{2}\right) \right\} \cdot \sin\left(\frac{\phi_M - \phi_R}{2}\right)
$$
$$
+ \left\{ c_0 \cdot \sin(\phi_R + \phi_M) + c_1 \cdot \cos(\phi_R + \phi_M) \right\} \cdot \sin(\phi_M - \phi_R), \qquad (3.34)
$$

where the terms associated with the natural evolution of ϕ_R are necessary for the fit to converge to the correct result since for this the model need to reflect the measurement very well. The result of the fit will be the coupling factors a_0, a_1, a_2, b_0, b_1, c_0 and c_1. The error term has two amplitudes that must be examined separately; the first order amplitude is given by:

$$
\varepsilon_1 = \sqrt{b_0^2 + b_1^2} \qquad (3.35)
$$

and the amplitude of the second order error term yields

$$
\varepsilon_{12} = \sqrt{c_0^2 + c_1^2}. \qquad (3.36)
$$

The a-coefficients are not relevant for the amplitude of the error term, but only describe the natural evolution of ϕ_R linearly.

Figures 3.12(a) and 3.12(b) show a selected segment of $90\,s$ from the end of the phase measurement of Ψ_1 and Ψ_{12} respectively. They are plotted against the reference phase, ϕ_R. The fit is also shown and it can be seen that it is a good representative of the measurement.

(a) Ψ_1
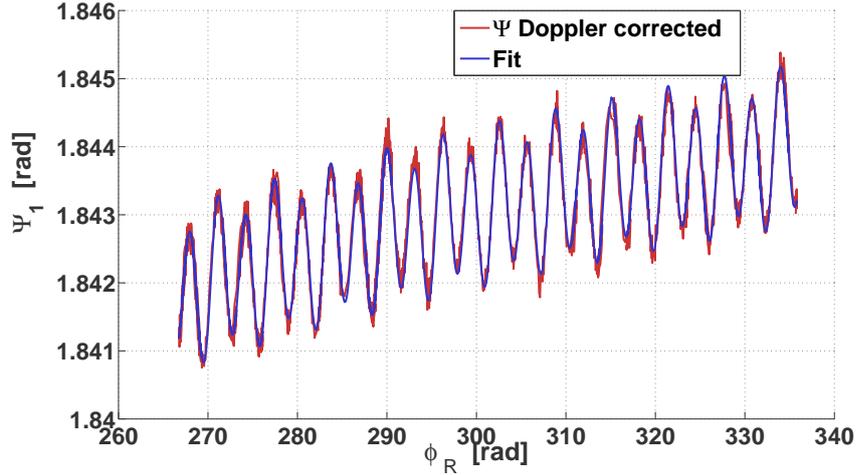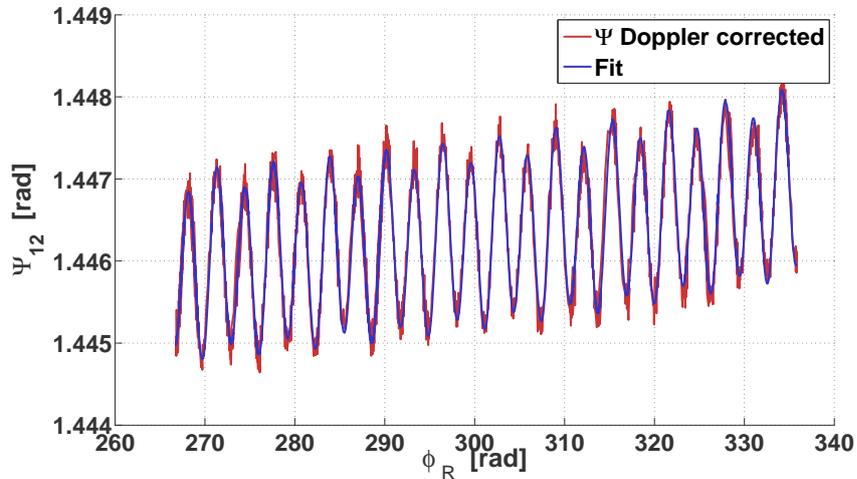
(b) Ψ_{12}

Figure 3.12: Fit of sideband induced noise. Plotted are Ψ_1 (a) and Ψ_{12} (b) as a function of ϕ_R. The fit was performed on the Doppler corrected data. The measurement was carried out without stabilising the optical pathlength difference (OPD). The Doppler effect originates from the fast fluctuations Δ_F. The length of the segment shown is $90\,s$.

The residuals of the data shown in Figure 3.12 and the corresponding fits are shown in a histogram in Figure 3.13. The relative residuals are given in percentage. They are normally distributed as expected.
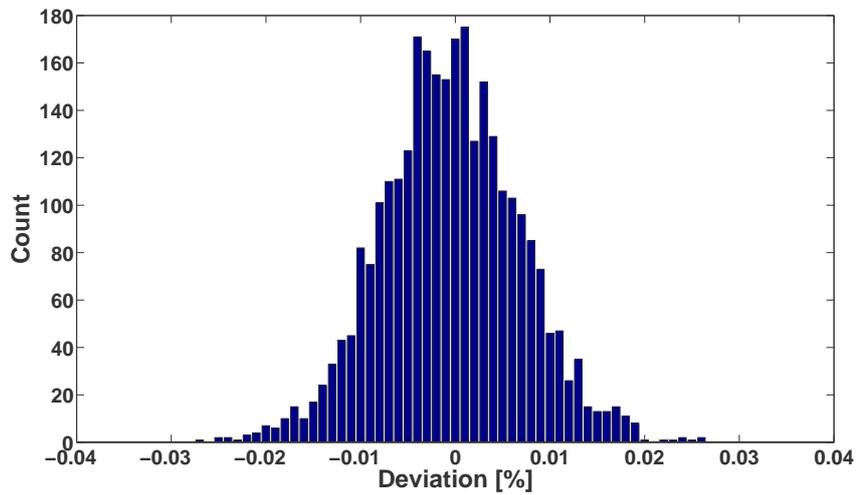
(a) Residuals for the fit of Ψ_1
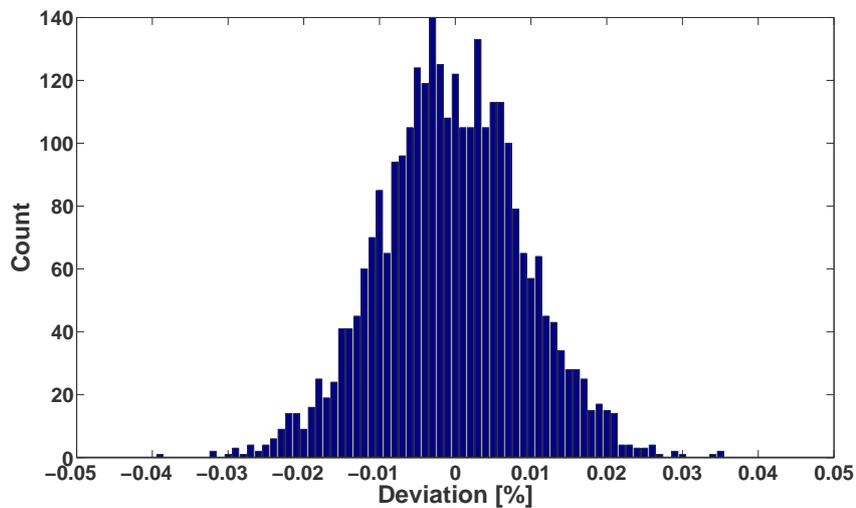
(b) Residuals for the fit of Ψ_{12}

Figure 3.13: Histogram of the residuals of the fit of the sideband induced noise. The plot
refers to the fit shown in Figure 3.12. The counts of the data points are plotted
as a function of the deviations in percentage. a) refers to Ψ_1 and b) refers to
Ψ_{12}. Both functions follow a normal distribution as expected.

This procedure is repeated for two more segments: one is taken from the beginning of the measurement and is $50\,\mathrm{s}$ long and another one is $90\,\mathrm{s}$ long and has been taken from the end of the measurement. The resulting amplitudes of the first and second order sidebands as well as their average are given in Tables 3.1 and 3.2 for Ψ_1 and Ψ_{12} respectively.

Table 3.1: Sideband amplitude derived indirectly from the phase measurement of Ψ_1. Given are the combined first order amplitudes and in the last column the combined second order amplitudes are shown.

Time segment [s]	ε_1 [rad]	
	1st order	2nd order
1660-1710	$2.26 \times 10^{-4} \pm 1.86\%$	$12.68 \times 10^{-4} \pm 0.27\%$
3540-3550	$2.25 \times 10^{-4} \pm 4.98\%$	$13.08 \times 10^{-4} \pm 0.69\%$
5580-5670	$3.11 \times 10^{-4} \pm 1.20\%$	$10.67 \times 10^{-4} \pm 0.31\%$

Table 3.2: Sideband amplitude derived indirectly from the phase measurement of Ψ_{12}. Given are the combined first order amplitudes and in the last row the combined second order amplitudes are shown.

Time segment [s]	ε_{12} [rad]	
	1st order	2nd order
1660-1710	$1.85 \times 10^{-4} \pm 3.22\%$	$12.67 \times 10^{-4} \pm 0.27\%$
3540-3550	$2.64 \times 10^{-4} \pm 5.61\%$	$13.03 \times 10^{-4} \pm 0.74\%$
5580-5670	$2.49 \times 10^{-4} \pm 1.80\%$	$10.69 \times 10^{-4} \pm 0.29\%$

The averaged results are then inserted into the model to obtain the sideband induced noise (SIN) contribution and subtracted from the original phase. Figure 3.14 shows the spectral density of the original measurement phase converted into test mass displacement in metres, which is then called X_m. This is plotted together with the data from which the SIN noise contribution was subtracted. The sensitivity of the corrected test mass displacement is significantly increased by the subtraction. The fact that the corrected measurement still does not reach the requirement of the interferometer is of minor importance since the measurement was taken in 2005 and different noise sources might have had an impact on the experiment. Nevertheless, it shall be noted that stabilising the OPD and reducing electronic cross-talk which result in a suppression of the amplitude of optical sidebands are the preferred method for increasing the measurement sensitivity.

It was stated before that, if possible, suppressing a noise during a measurement, should be the preferred method over estimating and subtracting it from the noisy measurement. This is the approach for the experiment presented in the following.

As mentioned above, the analysis of the OPD scan in terms of sideband induced noise will be investigated. The aim of this experiment was to find an upper limit for the amplitudes, ε, of the sidebands introducing the noise. To find such upper limits, the origin of the noise source (as stated in [15]) the electrical sidebands resulting
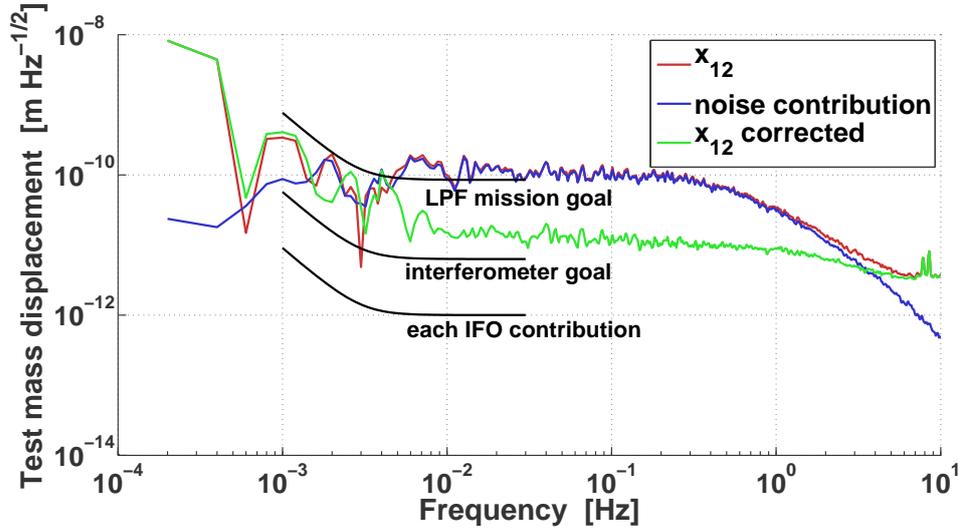
Figure 3.14: X_{12} (red) is the differential test mass displacement in meters. It was taken without OPD stabilisation. The blue curve shows the estimated sideband induced noise (SIN) contribution and the green curve represents the test mass displacement from which the SIN contribution was subtracted. Measurement data from 2005.

from electronic cross-talks has been minimised by using a setup that resembles the one on-board LISA Pathfinder. That is connecting units directly instead of feeding through a vacuum chamber. The procedure to be applied is the same as before: At first the data is corrected for the Doppler shift introduced by the rapid OPD scan and then the remaining error is estimated by applying the linear fit.

Figure 3.15 shows the variation of ϕ_R over one complete measurement.

The measurements of the OPD scan for the EM test campaign of the laser and laser modulator were of length of around 200-300 seconds. For this reason the spectral densities of these measurements will not be computed but the focus of the data analysis will be on the estimation of the sideband amplitudes, ε_M.

Figure 3.16 shows one segment of the time series of the phase measurement. It was selected using the same criteria as before. Again the phase is plotted against the reference phase to make the dependency of the measurement phase on the reference phase visible. The plot shows the measurement phase Ψ_{12}, before and after the Doppler correction was applied. The periodicity of Ψ_{12} in ϕ_R is significantly reduced.

In Figure 3.17 the reference phase, Ψ_{12}, (after the correction of the Doppler error) is plotted against the reference phase to visualise the residual sideband induced error term. In this curve the amplitude of the error term is not clearly visible anymore. The fitted curve is shown as well. The result of the fit is given in Table 3.3. For completeness also the result for Ψ_1 is given which was not explicitly plotted before. The numerical results represent what could be suspected from the plot: the statistical variances of the estimated amplitudes are mostly of the same order of magnitude as the amplitudes themselves. Hence, the sideband induced error term in this case is too low to obtain a reliable fitting result. Nevertheless the obtained
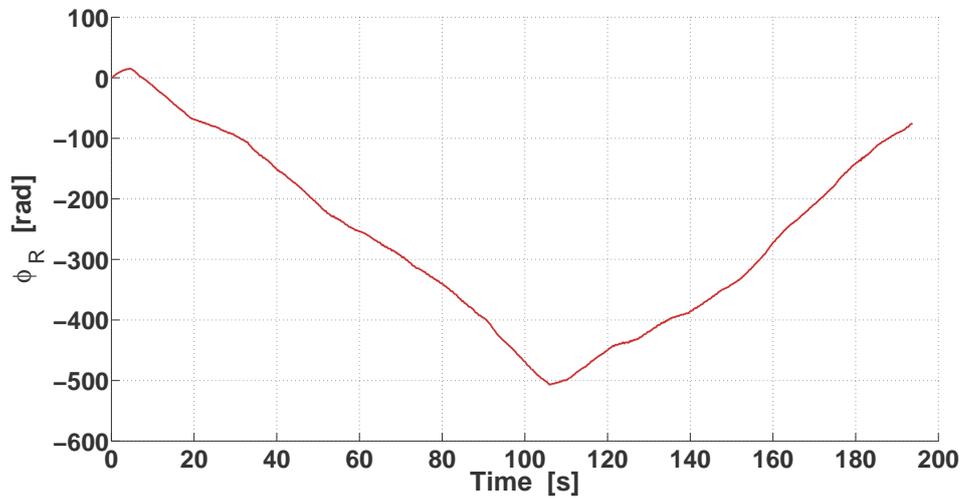
Figure 3.15: Variation of ϕ_R during an OPD scan of the Engineering Model test campaign of the laser.
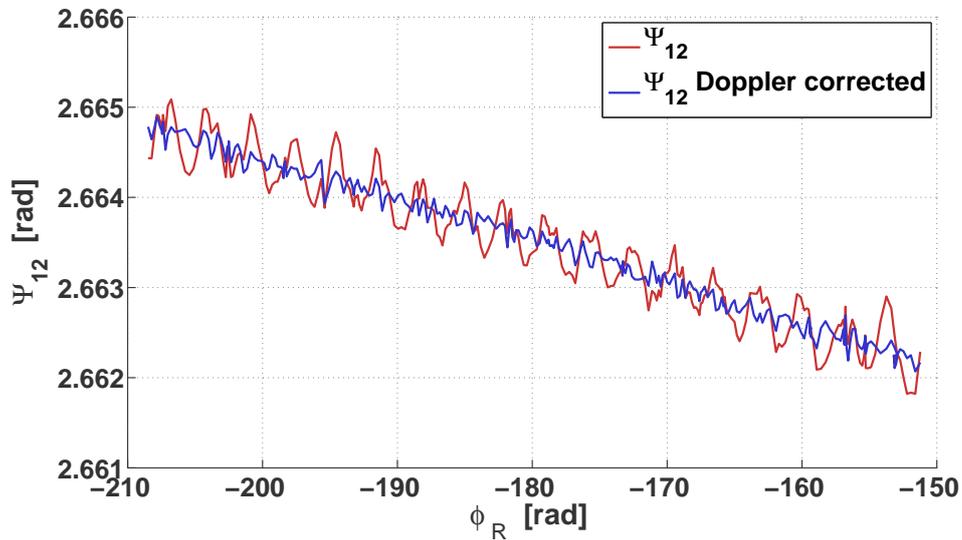
Figure 3.16: Doppler correction of an OPD scan experiment during the Engineering Model test campaign of the laser and laser modulator. Ψ_{12} is plotted before (red) and after (blue) the removal of the Doppler effect.

values for the amplitudes can be seen as an upper limit for the true amplitudes of the noise term. But even these upper limits are one order of magnitude lower than the ones estimated for the free-running OPD measurement, where the electronic cross-talks were not reduced. The upper limits should, however, not be inserted in the model for the noise source for subtraction of the measurement phase.
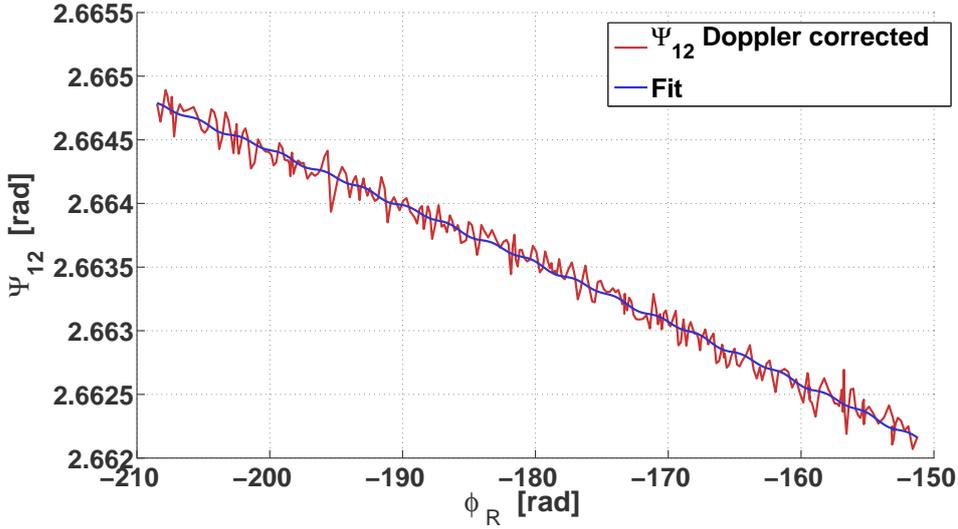
Figure 3.17: Fit of Ψ_{12} for measuring the amplitudes of the sideband induced noise term. The red curve shows the measurement after the Doppler correction was applied. The blue curve represents the function resulting from the linear fitting routine.

Table 3.3: Sideband amplitudes derived indirectly from the phase measurement of Ψ_1 during the test campaign of the Engineering Model of the laser and laser modulator. Given are the combined first order amplitudes and in the last row the combined second order amplitudes are shown.

Measurement phase	ε [rad]	
	1st order	2nd order
Ψ_1	$2.15 \times 10^{-5} \pm 60.47\%$	$2.73 \times 10^{-5} \pm 25.27\%$
Ψ_{12}	$1.20 \times 10^{-5} \pm 40.83\%$	$3.62 \times 10^{-5} \pm 28.45\%$

It shall be noted again that the measurement on the Engineering Model of the laser and laser modulator have been performed under idealised conditions to minimise the electronic cross-talk causing the sideband induced noise. As such, the results can be regarded as upper limits for the combined amplitudes of the first and second order sidebands. The two different measurements analysed in this section illustrate well that the contribution of the sideband induced noise term can be lowered significantly by reducing electronic cross-talk.

In LISA Pathfinder an on-orbit experiment on the estimation of the sideband induced noise term is planned. The analysis presented here has been completely carried out within LTPDA, demonstrating that the analysis tools for these experiments are in place.

3.5 Laser amplitude noise

In this section the effect of laser amplitude noise on the interferometric measurement will be discussed. For this, a realistic assumption based on laboratory results will be made for the noise source. The noise will be injected into a one-dimensional simulation of LTP to establish the noise contribution to the test-mass displacement which is the interferometer output. The work presented here is based on technical document written by Martin Hewitson [38].

Fluctuations of the amplitude of the laser beam used in LTP cause a change in momentum transfer on the test masses. This change results in force noise on the test-masses coupling directly into test mass motion. Establishing a method for measuring the noise contribution of laser amplitude noise to the interferometric measurement is important since this measurement is planned to be carried out on-board LISA Pathfinder. As such it is part of the experiment master plan whose development is performed in parallel to the preparation of the data analysis of the mission.

As the experiment involves test mass motion, it cannot simply be performed on ground. Instead, the experiment is simulated using a simulator developed in LTPDA. The one-dimensional model to be simulated for the purpose of the noise projection of laser amplitude noise is discussed in detail in Chapter 4. There all parameters and controllers involved are explained. The model is based on the so-called science mode, where the position of TM1 is measured by the X1 interferometer (see Figure 3.3(a)) and fed back to the drag-free controller. The differential displacement is measured by the X12 interferometer (see Figure 3.3(b)) and it is controlled using the electrostatic suspension controller, C_{sus}.

It is possible to suppress the noise source by stabilising the laser amplitude. A discussion on stabilisation loops in the OMS can be found in [29, 30].

3.5.1 Building the noise model for the input laser amplitude noise

The laser amplitude fluctuations lead to a change in the power of the laser beams hitting the test masses. This power can be extracted from the measurement of relative intensity noise (RIN) using for example the frequency interferometer in LTP (see Figure 3.2(b)). Here it is important to note that the frequency interferometer measures the relative intensity noise of both beams whereas only beam 1 hits the test masses. One possibility for determining the RIN of the beam hitting the test masses is to assume equal noise in both beams and dividing the measured RIN by $\sqrt{2}$. Another possible approach is to turn off one beam and measure the RIN separately from the measurement of the test mass position. More possibilities can be thought of but in this thesis the second option is considered. Figure 3.18 shows the transfer function of the assumed RIN used as the noise-shape filter for white noise in the simulation.

The noise will be injected into the simulation as force noise on the test masses. The force resulting from photon momentum transfer is given by

$$F = \frac{2P}{c}, \tag{3.37}$$
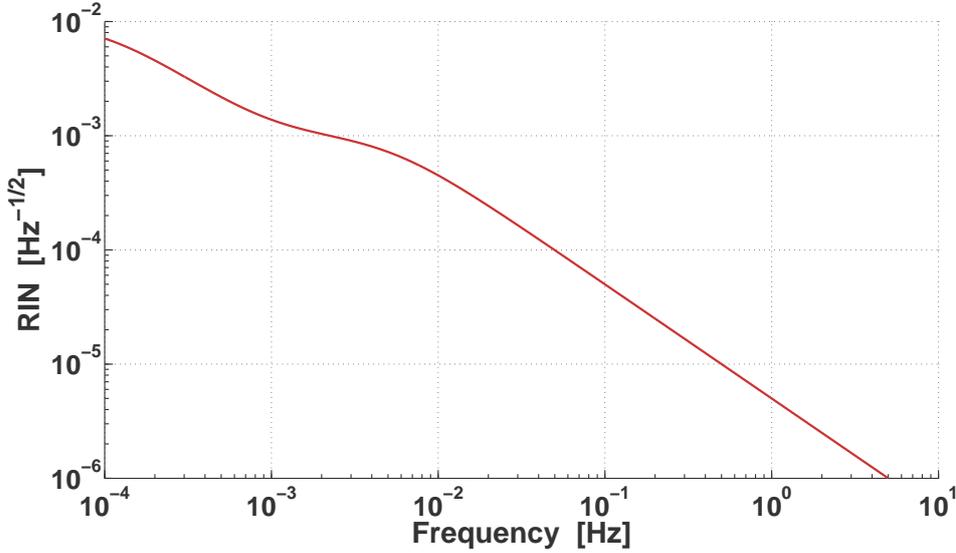
Figure 3.18: Transfer function for the model of relative intensity noise in the laser beam
            used in LTP. It is based on a free-running measurement performed on the
            Engineering Model of the laser assembly.

where P is the power and c is the speed of light. This model is a simplified one; for
example it is assumed here that the reflectivity of the test masses is equal to one.

To convert the dimensionless relative intensity noise plotted in Figure 3.18 to
power we have to multiply by the laser beam power that impinges the test masses.
Assuming a perfect beam splitter the laser beam power divides on the two test masses
as follows:

- 2 mW hit TM1 and

- 1 mW hits TM2.

For stimulating the system a sinusoid modulation signal is injected. For the am-
plitude we chose about 10% of the force noise, hence it is set to 0.1 mW. Since
the beam is split after it hits TM1, the amplitude of the signal on TM1 is 0.1 mW
whereas the amplitude of the signal on TM2 is 0.05 mW. The modulation frequency,
f_m, is set to 10 mHz.

$$F_1 = \frac{2}{c} \cdot [P_1 + A_1 \sin(2\pi f)] \tag{3.38}$$

$$F_2 = -\frac{2}{c} \cdot [P_2 + A_2 \sin(2\pi f)] , \tag{3.39}$$

where the opposite signs originate from the coordinate system of LTP. It was shown
earlier in this chapter in Figure 3.5. The forces act in opposite directions and push
the test masses apart. This results in an increasing differential displacement, o_{12},
measured by the interferometer. From the arguments listed above, the following set

of parameters is inserted into the individual equations for the force noises:

$$P_1 = 2 \times 10^{-3} \quad [\text{W}]$$
$$P_2 = 1 \times 10^{-3} \quad [\text{W}]$$
$$A_1 = 1 \times 10^{-4} \quad [\text{W}]$$
$$A_2 = 5 \times 10^{-5} \quad [\text{W}]$$
$$f_{\text{m}} = 1 \times 10^{-2} \quad [\text{Hz}]$$

$$(3.40)$$

The resulting differential force noise contributing to the differential test mass displacement, o_{12}, is given by the sum of the individual force noises:

$$F_{\text{diff}} = F_1 + F_2. \tag{3.41}$$

In Figure 3.19 the modulated differential force noise is plotted. The modulation signal is clearly visible at 10 mHz. The figure shows the force noise as it is injected into the system. The modelling of F_{diff} as described above is done in the frequency domain. After that a filter is created from the resulting transfer function and it used to filter a time series of white noise. Finally the time-series created is injected into the system.
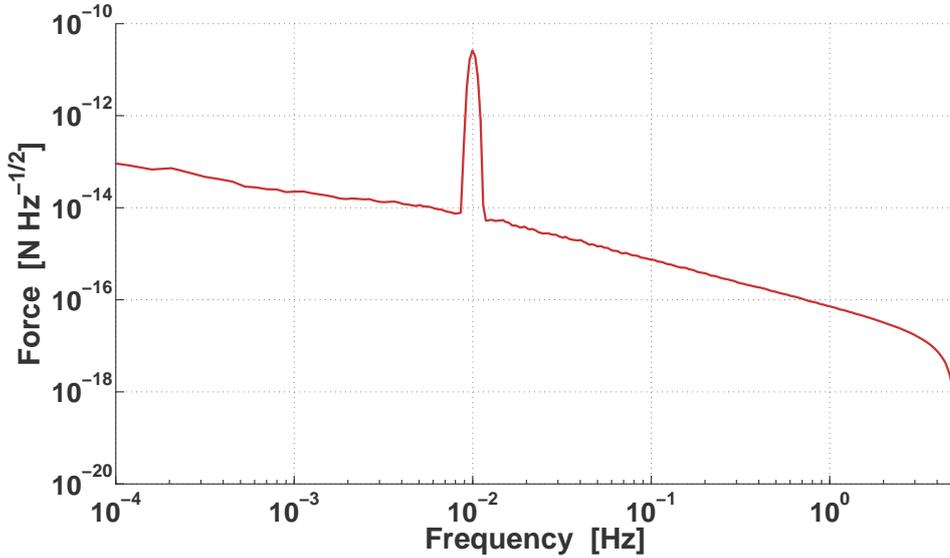
Figure 3.19: Linear spectral density of the modelled force noise due to laser amplitude fluctuations.

3.5.2 Simulation of a simplified model of LTP with injected laser amplitude noise

The model used for the simulation is a one-dimensional state space model (ssm) implemented in LTPDA. The method of modelling the system in state space is

useful for complex systems like LTP. However, state space will not be discussed further here since the development of the model in ssm is not part of this thesis. It is rather the tool used to accomplish the noise projection to be presented.

In addition to the modelled laser amplitude noise, the following noise sources have been injected into the system:

- thruster noise,

- test mass noise,

- interferometer measurement noise.

The origin of these noise sources is discussed in Chapter 4. They are the most obvious noises to be included. In Figure 3.20 the two output channels of the interferometer, o_1 and o_{12} simulated using all noises listed including the laser amplitude noise are plotted. It can be seen that the modulated laser amplitude noise has no contribution in the o_1 channel but it couples into into the differential channel, o_{12}.
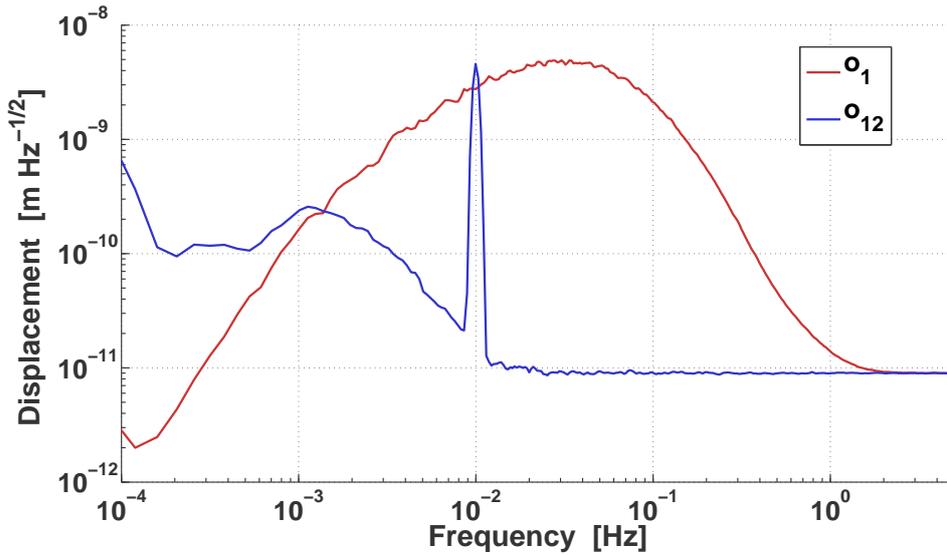
Figure 3.20: Interferometer measurement channels simulated with modulated laser frequency noise. The modulation couples only into the differential channel, o_{12}.

3.5.3 Noise projection

The aim of the work presented here is to establish a method for measuring the contribution of laser amplitude noise to test mass motion. This is done by performing a noise projection of the modelled laser amplitude noise on the differential interferometer output, o_{12}. The noise projection is carried out by performing the following steps:

1. Measure the input laser amplitude noise, F_{diff} (see Figure 3.19).

2. Convert F_{diff} to in-loop displacement noise, D by

$$D(f) = \frac{\text{PSD}(F_{\text{diff}})}{(s^2 + \omega_2^2) + C_{\text{sus}}},\qquad(3.42)$$

where $s = 2\pi i f$ is the Laplace variable, $\omega_2^2 = -2 \times 10^{-6}\ s^{-2}$ and C_{sus} represents the electronic suspension loop controller.

3. Measure the differential interferometer output, o_{12} (see Figure 3.20).

4. Compute the transfer function, T, from the injected force noise, F_{diff}, to the differential test mass displacement, o_{12}:

$$T = \frac{D(f_{\text{m}})}{\tilde{o}_{12}},\qquad(3.43)$$

where $\tilde{o}_{12}$ denotes the DFT at f_{m} and D has been computed explicitly at the modulation frequency, f_{m}.

5. The projection of laser amplitude noise to differential test mass displacement, R_{12} is given by the multiplication of the absolute value of the coupling factor, T, with the amplitude spectral density of the simulated differential displacement, o_{12}.

$$R_{12} = ASD(o_{12}) \times |T|.\qquad(3.44)$$

The resulting noise contribution, R_{12}, is plotted together with the simulated displacement noise, o_{12}, in Figure 3.21. The injected signal at 10 mHz is clearly visible in both spectral densities.
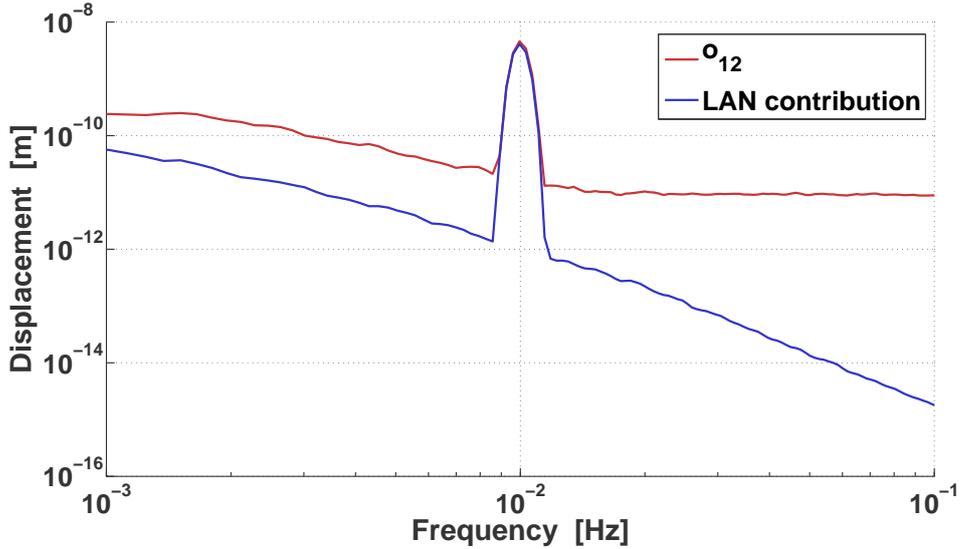
Figure 3.21: Noise projection of laser amplitude noise, R_{12}, into differential displacement noise, o_{12}.

3.5.4 Future investigations

We established a method to estimate the coupling of laser amplitude noise into the measurement of the differential displacement of the test masses in LTP. For this a newly developed simulator based on a state space description of the system was used. The results shown in this section represent the first attempt of simulating this noise source using the state space simulator.

The simulator will become increasingly realistic with the approaching launch date of LISA Pathfinder. Finally it will be possible to simulate the complete system of LTP in three dimensions and with all variable parameters. In the near future for example input noises maybe adjusted to be more realistic. Also the model used for the conversion of laser amplitude noise to force noise on the test masses will be extended to take into account important physical quantities like the reflectivity of the test masses. When the coupling factor is known these additional parameters could as well be estimated from the noise projection. Following this scheme, it may be also possible to track the test mass reflectivity and the efficiency of the photodiodes over the mission life time. Since these two parameters have the same effect on the measurement performed it could be difficult to distinguish them but it should be possible by including the measurements of all appropriate photodiodes.

3.6 Conclusion

In this chapter important laboratory experiments on the Engineering Model of the LISA Pathfinder optical bench were presented. These experiments are part of the experiment master plan prepared in the preparation of the mission.

The first experiment investigated was the angular noise subtraction, where simulated angular jitter was injected into the measurement. The coupling coefficients that relate the the angular noise to the longitudinal measurement were found and then used to subtract this noise from the data. It was shown that the sensitivity can be improved significantly by this method.

After that the optical pathlength difference (OPD) was introduced and two different measurements were analysed in terms of sideband induced noise (SIN). The first measurement comprises several hours, where the OPD was not stabilised. The measurement was taken under normal laboratory conditions. Here the sideband induced noise was estimated and then subtracted like shown in the first experiment.

The second measurement was also performed without stabilising the OPD but here the electronic sidebands responsible for the sideband induced noise were intentionally minimised. It shall be noted that the electronic system of the engineering and flight models has been optimised carefully in terms of sidebands because of the early comprehension that this will be an important noise source.

The comparison of the estimations of the amplitudes of the two measurements shows that even without stabilising the OPD it is possible to suppress the SIN significantly by reducing electronic sidebands in LTP.

The last project presented was the estimation of the coupling of laser amplitude noise into the differential displacement of the test masses. A realistic relative intensity noise was forming the bases of the input amplitude noise. A simplified model

for the conversion of amplitude noise to force noise on each test mass was introduced and then a noise projection was carried out. This noise projection shows the contribution of the laser amplitude noise to displacement noise.

The focus of this chapter was on the development of data analysis tools in LT-PDA, the software tool to be used for the on-board data analysis. The work presented here highlights ready-to-use methods within this comprehensive data analysis tool. The close collaboration between the data analysis of the laboratory experiments and the development of LTPDA has proven to be a very effective way of development. As such the effort will be continued throughout the preparation of the experiment master plan as well as during all hardware tests accomplished at the AEI.

4 Data Analysis for the Mission

4.1 Introduction

The main measurement concept of LISA is based on the ability to put a test mass into free fall and account for all residual forces that may act on it. LISA Pathfinder aims to verify this concept. An interferometer measures the distance between two free-falling test masses with pico-meter accuracy in the milli-Hertz range. The LISA Pathfinder mission time is strictly limited, therefore experiments must be prepared in detail prior to the mission in order to maximise its science output. All experiments that are to be carried out during the mission are collected and defined in the Experimental Master Plan (EMP). The preparation of the data analysis for LISA Pathfinder and this Experimental Master Plan go hand in hand aiming to

- assure that all experiments necessary to characterise the mission will be well understood and realisable,

- verify before launch the ability of analysing the complete mission data.

Daily data analysis during hardware development and testing on ground are crucial, but this alone will not ensure that all necessary parts needed for the analysis of the mission are in place. For this reason major components from the Experimental Master Plan are simulated in theoretical scenarios defined by the data analysis groups. The theoretical scenarios had to be prepared carefully in order to accomplish them successfully. These preparations and the first simulations performed are the subject of this chapter. The course of action as reflected by the individual sections is described in the following.

1. To simplify international collaboration, scientists involved in the data analysis for LISA Pathfinder agreed on one model of the dynamics of the two test masses onboard the satellite. This model is a simplification of the actual experiment and will be expanded as the development of the algorithms for the analysis, as well as the understanding of the complete experiment, advances. Thus the development of the simulation of LTP started with the definition of a one-dimensional model of LTP in the so-called science mode. In this mode test mass 1 is kept drag-free which means the spacecraft follows the motion of test mass 1 (TM1), and test mass 2 (TM2) moves such that the distance between test mass 1 and test mass 2 is kept constant. The technical details of this mode, as well as the mathematical description of the motion of the test masses, will be described in Section 4.2.

2. Once a common model has been defined, it needs to be verified. Since all future models will probably be based on this first one, and since it will serve

as a reference for future simulations, it must be very reliable. The verification of the model is done by the collaboration of two sub-groups of the LTP data analysis group who have their own independent understanding of the defined model. One group runs a simulation based on this model and delivers the obtained data to the other group, which analyses it according to its own understanding of the defined model. The result is a consolidated understanding of the physics of this simplified but fundamental model and an agreement on language and procedures for the future collaboration. The main organisational structure of having two sub-groups is kept throughout all simulations, whereas the scientists involved in each group may of course change during the project. The details about the process of verification of the simplified model for LTP are given in section 4.3.

3. The third step on the way to simulating LTP is performing an analysis defined in the Experimental Master Plan on data from the simulations - the 'conversion to acceleration'. The position of the two test masses is measured by means of the interferometer. Due to residual forces, like solar pressure, gravitational couplings or un-commanded thruster force, the test mass position changes continuously. Differentiating the change in position of the test masses twice in principle leads to their acceleration. However, additional controller forces are applied to keep the test masses inside the measurement range. Thus, in order to compute the test mass acceleration resulting from residual force noises, the controller forces need to be accounted for, which slightly complicates the procedure.

   The data to be simulated (that is the data that will be analysed as described in this chapter) are the interferometer measurements.

These three steps have been published under name of 'The first mock data challenge for LISA Pathfinder' [39].

The test mass accelerations are an important science output of the mission since they can be directly related to the residual force noises on the test masses, whose measurement is a major goal of LISA Pathfinder. In section 4.5 the results of this important experiment are presented. Simulating certain experiments from the experimental master plan, even with a simplified model of the experiment, enormously promotes the development of the data analysis tools.

The most recent experiment of this kind that has been performed, aims to estimate certain model parameters. During the mission some parameter values might not be known exactly and in order to proceed (for example, with the 'conversion to acceleration') these parameter values are to be estimated first. The algorithms necessary to accomplish the estimation of parameter values need essential for the mission and their development will be a very important outcome of this experiment. The last part of this chapter (Section 4.6) deals with a couple of methods implemented to perform this task.

The work introduced above and described in detail in this chapter has been done within the framework of so-called Mock Data Challenges (MDCs). In this context the term 'Mock Data' describes data obtained by performing simulations of the defined models and experiments, and 'challenge' comprises all kinds of possible

analysis based on the definitions made. While the data originates from a purely theoretical scenario, the tools developed for the analysis are the actual ones that will be used for the mission, which in our case is LTPDA. So far, two of these MDCs have been performed successfully. Figure 4.1 shows the way the MDCs are accomplished.
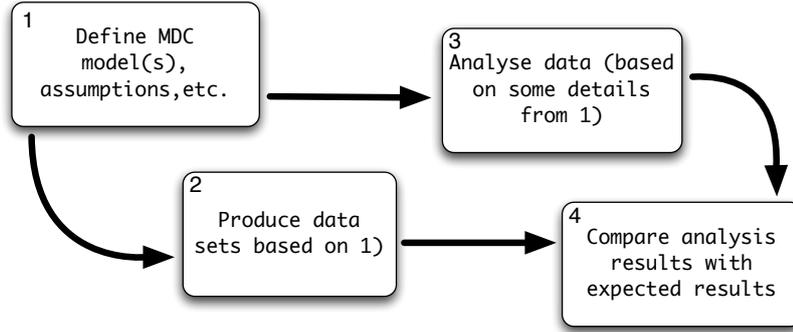
Figure 4.1: Illustration of the work flow in MDCs for LTP: Data sets are produced (2) according to a previously defined model (1). This data is analysed by making use of some information incorporated into the model definition. Finally the result from the analysis is compared to the expectation from the process of data generation (4).

4.2 A simplified model of the LTP dynamics

The model underlying the first couple of Mock Data Challenges describes a simplified setup of the LISA Pathfinder experiment where the two test masses move only along the sensitive x-axis and no cross-talk from other axes is included. The measurement principle is illustrated in Figure 4.2. It shows a schematic of the spacecraft (S/C) including the two test masses (TM1 and TM2), the capacitive sensors, the micro Newton thrusters and the controllers, as well as all forces and interactions acting along the sensitive axis. The spacecraft has a mass of $M = 475$ kg and the two test masses are assumed to have equal mass $m_1 = m_2 = m = 1.96$ kg.

The model is based on the so-called science mode. Test mass 1 moves, since it experiences residual force noises. The spacecraft is made to follow this motion of test mass 1 by controlling the thrusters. To keep the distance between the two test masses constant, test mass 2 is made to follow the motion of the spacecraft with low bandwidth. The position of test mass 1 is measured with respect to the spacecraft and since the control of the spacecraft via the micro-Newton thrusters is noisy, the measurement is dominated by exactly this thruster noise. In the differential measurement (that is, the position of the two test masses with respect to each other), the effect of the thruster noise mostly cancels out and a measure of the residual test mass force noises is obtained. These relationships of measurement and noises can be extracted from the graph in Figure 4.7 of Section 4.5, where the matter is discussed in more detail.

The data used as input error signals to the control loops are the two interferometer measurement channels: The measurement of the position of test mass 1 with respect to the spacecraft and the measurement of the differential position of the two test masses, referred to as $\vec{o}_1$ and $\vec{o}_{12}$ respectively. Two control loops are illustrated in Figure 4.2:

- The position of the spacecraft relative to test mass 1 is controlled by the micro Newton thrusters according to the measurement of $\vec{o}_1$. This combination is called the drag-free control loop.

- The electrostatic suspension loop comprises the capacitive actuators controlling the position of test mass 2 relative to test mass 1 by means of the interferometer output $\vec{o}_{12}$ [40].

Analysing all forces acting on the system shown in Figure 4.2 will lead to the equations of motion of the two test masses. These equations form the recipe to derive the acceleration of the two test masses from their position measured by the interferometer. The equations of motion can be found by setting up the Lagrangian equations of the system and solving the Euler-Lagrange equations of the two test masses. The Lagrangian is defined as the kinetic energy of the system, T, minus its potential energy, V, which can be written as $L = T - V$. For the simplified model of LTP the Lagrangian reads:

$$L = \underbrace{\frac{m}{2}\dot{x}_1^2 + \frac{m}{2}\dot{x}_2^2}_{\text{kinetic energy}} - \underbrace{\frac{k_1}{2}x_1^2 - \frac{k_2}{2}(x_2 - x_1)^2 - \frac{k_3}{2}x_2^2}_{\text{potential energy}}, \qquad (4.1)$$

where m is the mass of each TM, x_1 and x_2 are the test mass positions; k are the spring couplings between bodies: k_1 acts between the spacecraft and test mass 1, k_2 acts between the S/C and test mass 2 and k_3 acts between the two test masses.

The term $\frac{k_2}{2}(x_2 - x_1)^2$ represents the gravitational coupling between the test masses, with k_2 being the spring constant. From this, the Euler-Lagrange equations are found for test mass 1 and test mass 2 respectively:

$$\text{external force on TM1: } F_1 = \frac{\partial L}{\partial x_1} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}_1}\right). \qquad (4.2)$$

$$\text{external force on TM2: } F_2 = \frac{\partial L}{\partial x_2} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}_2}\right). \qquad (4.3)$$

In an inertial frame, Equations 4.2 and 4.3 would be equal to zero. In this document, however, the LTP experiment is defined in a non-inertial frame, and forces acting on the spacecraft are defined to be external. This leads to a total force on test mass 1 and test mass 2 which is non-zero.

The view of the spacecraft being a non-inertial frame is only a matter of definition. It is however a very convenient definition since all coordinates necessary to describe the motion of the test masses can be defined with respect to an origin inside the spacecraft. Other definitions of the system will result in the same equations of motion provided the correct coordinate transformations are made.

A_{df} $[\frac{N}{kg}]$ - thruster force per unit mass
A_N $[\frac{N}{kg}]$ - acceleration noise on spacecraft
$k_1 x_1$ $[N]$ - spring coupling between TM1 and S/C
$k_3 x_2$ $[N]$ - spring coupling between TM2 and S/C
A_1, A_2 $[\frac{N}{kg}]$ - acceleration noises on TM1 and TM2
C_{df} $[s^{-2}]$ - drag-free controller
C_{sus} $[s^{-2}]$ - electrostatic suspension controller
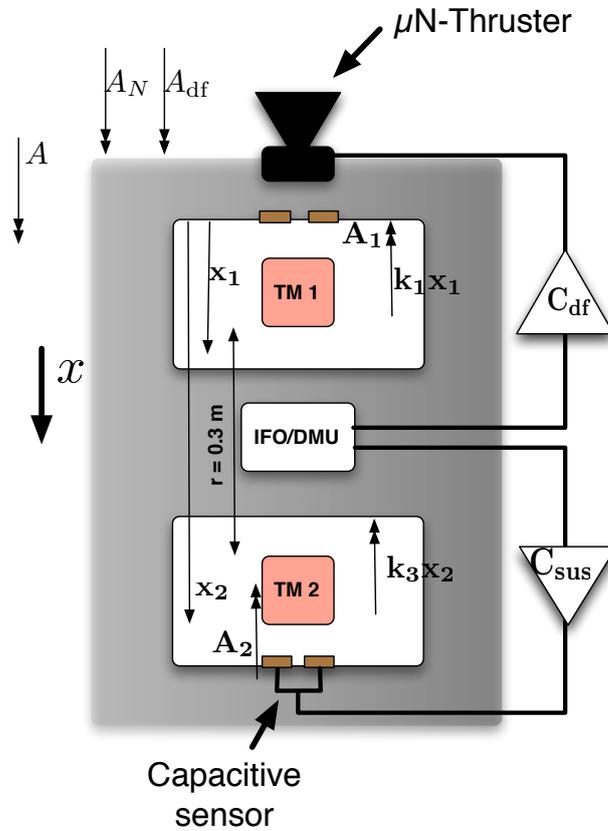
Figure 4.2: Top: List of definitions for variables depicted in the figure below including their units. Bottom: Simplified model of the LTP dynamics: The spacecraft in grey houses two test masses. The thruster (black) moves the S/C via the controller C_{df}, and the capacitive sensor (brown) acts on test mass 2, where the movement is controlled by C_{sus}. The arrows of A and x indicate the positive direction of forces (per unit mass) and distances reflected in Equations 4.4 to 4.10. x_1 and x_2 are the distances between the Spacecraft and test mass 1 and test mass 2 respectively. The block IFO/DMU represents the interferometer and the Data Management Unit. More detailed information on the DMU can be found in [12]

The external forces on the test masses (F_1 and F_2) arise from the total force on the spacecraft. To derive the total forces on the test masses, first the total force on the spacecraft must be found. The mass of the spacecraft is denoted by M, and m is the mass of each test mass. The forces acting on the spacecraft are:

- the force applied by the thrusters: MA_{df},

- additional force noises like solar radiation pressure and un-commanded thruster noise: MA_{N} and

- forces arising from the back reaction of the spring coupling of the test masses: $-k_1x_1$ and $-k_3x_2$ on the S/C.

The total force on the spacecraft can then be written as:

$$F_{\mathrm{SC}} = MA_{\mathrm{sc}} = MA_{\mathrm{df}} + MA_{\mathrm{N}} - k_1x_1 - k_3x_2. \tag{4.4}$$

The thruster force, MA_{df}, arises from the drag-free loop reading the output o_1 of the interferometer and feeding back to the micro-Newton thrusters to make the spacecraft follow test mass 1. Therefore it reads:

$$F_{\mathrm{df}} = MA_{\mathrm{df}} = -o_1C_{\mathrm{df}}M, \tag{4.5}$$

with C_{df} representing the transfer function of the drag-free controller and o_1 the first interferometer output channel.

External forces on the two test masses arise from the pseudo force from the spacecraft, due to working in the non-inertial frame ($-mA_{\mathrm{sc}}$). Additional forces (mA_1) may arise from local disturbances, for example from magnetic fields. This leads to the expression for the force on test mass 1:

$$F_1 = mA_1 - mA_{\mathrm{sc}} = mA_1 - mA_{\mathrm{df}} - mA_{\mathrm{N}} + \mu k_1x_1 + \mu k_3x_2, \tag{4.6}$$

where $\mu = m/M$ is the reduced mass.

Inserting Equation 4.5 into Equation 4.6 leads to:

$$F_1 = mA_1 + m\ o_1C_{\mathrm{df}} - mA_{\mathrm{N}} + \mu k_1x_1 + \mu k_3x_2. \tag{4.7}$$

The same pseudo force ($-mA_{\mathrm{sc}}$) has to be taken into account in the equation of motion for test mass 2. Additionally a force from the electrostatic actuator (mA_{sus}) acts on test mass 2, which leads to the following expression for the total force on test mass 2:

$$F_2 = mA_2 + mA_{\mathrm{sus}} - mA_{\mathrm{df}} - mA_{\mathrm{N}} + \mu k_1x_1 + \mu k_3x_2. \tag{4.8}$$

In the electrostatic suspension loop the differential interferometer output o_{12} is measured and fed back to control the position of test mass 2. The force arising from this reads:

$$F_{\mathrm{sus}} = mA_{\mathrm{sus}} = o_{12}C_{\mathrm{sus}}m, \tag{4.9}$$

where C_{sus} is the transfer function of the electrostatic suspension controller. Now Equation 4.8 can be rewritten as:

$$F_2 = mA_2 + m\, o_{12}C_{\text{sus}} - mA_{\text{df}} - mA_{\text{N}} + \mu k_1 x_1 + \mu k_3 x_2. \qquad (4.10)$$

Having found the net-forces acting on the two test masses (Equations 4.7 and 4.10), it is possible to set up the equations of motion for test mass 1 and test mass 2 from Equations 4.2 and 4.3. For this the right hand side of Equations 4.2 and 4.3 need to be solved using the Lagrangian from Equation 4.1.
For test mass 1 this reads:

$$mA_1 + m\, o_1 C_{\text{df}} - mA_{\text{N}} + \mu k_1 x_1 + \mu k_3 x_2 = -k_1 x_1 + k_2(x_2 - x_1) - m\ddot{x}_1, \quad (4.11)$$

and for test mass 2 we get:

$$mA_2 + m\, o_{12}C_{\text{sus}} - mA_{\text{df}} - mA_{\text{N}} + \mu k_1 x_1 + \mu k_3 x_2 = -k_2 x_2 + k_2 x_1 - k_3 x_2 - m\ddot{x}_2. \quad (4.12)$$

The following modifications lead to the final equations of motion of the test masses given in Equation 4.13 and Equation 4.14:

1. division by the mass of the test masses m,

2. substitution of $x_{12} = x_2 - x_1$,

3. substitution of the so-called *stiffness* terms: $\omega_1^2 = \frac{k_1}{m}$, $\Gamma = \frac{k_2}{m}$, $\omega_2^2 = \frac{k_3}{m}$ and

4. moving the equations to Laplace domain, with $s = i\omega$ acting as the Laplace variable for easier calculation of the derivatives.

$$\begin{aligned}
\left[s^2 + \omega_1^2(1+\mu) + \mu\omega_2^2\right] x_1 + \left[\mu\omega_2^2 - \Gamma\right] x_{12} &= -o_1 C_{\text{df}} + A_{\text{N}} - A_1, \quad (4.13)\\
\left[s^2 + 2\Gamma + \omega_2^2\right] x_{12} + \left[\omega_2^2 - \omega_1^2\right] x_1 &= -o_{12}C_{\text{sus}} + A_1 - A_2. \quad (4.14)
\end{aligned}$$

From this point on, terms involving the mass of the test masses divided by the mass of the spacecraft, μ, or the gravitational coupling between the test masses (Γ) will be neglected since they are small. It makes sense to move to matrix notation to preserve readability. The following matrices are obtained by appropriate grouping of all variables:

- the test mass coordinates:

$$\vec{q} = \begin{bmatrix} x_1 \\ x_{12} \end{bmatrix}, \qquad (4.15)$$

- the interferometer output:

$$\vec{o} = \begin{bmatrix} o_1 \\ o_{12} \end{bmatrix}, \qquad (4.16)$$

- the acceleration noises:

$$\vec{g}_{\text{n}} = \begin{bmatrix} A_1 - A_{\text{N}} \\ A_2 - A_1 \end{bmatrix}, \qquad (4.17)$$

- the controllers:

$$\mathbf{C} = \begin{bmatrix} C_{\mathrm{df}} & 0 \\ 0 & C_{\mathrm{sus}} \end{bmatrix} \quad \text{and} \tag{4.18}$$

- the dynamics of the system:

$$\mathbf{D} = \begin{bmatrix} (s^2 + \omega_1^2) & 0 \\ (\omega_2^2 - \omega_1^2) & (s^2 + \omega_2^2) \end{bmatrix}, \tag{4.19}$$

Using the above made definitions, Equations 4.13 and 4.14 can be combined to the following relationship:

$$\mathbf{D}\vec{q} = -\mathbf{C}\vec{o} - \vec{g}_{\mathrm{n}}. \tag{4.20}$$

Equation 4.20 relates the TM coordinates to the dynamics of the model of LTP and states that this is equal to the multiplication of the control matrix with the measured output minus the acceleration noises. The controller matrix, $\mathbf{C}$, comprises all terms for delays and gains. Both controller transfer functions include well defined delays, Ξ_{df}, for the drag-free loop and Ξ_{sus} for the suspension loop, as well as an actuator coefficient G_{df} and G_{sus} respectively (for further information on the delays see [41]). The matrices for the actuation ($\mathbf{G}$), the time delays ($\boldsymbol{\Xi}$) and the controller transfer functions ($\mathbf{H}$) are defined as follows:

$$\boldsymbol{\Xi} = \begin{bmatrix} \Xi_{\mathrm{df}} & 0 \\ 0 & \Xi_{\mathrm{sus}} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} G_{\mathrm{df}} & 0 \\ 0 & G_{\mathrm{sus}} \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} H_{\mathrm{df}} & 0 \\ 0 & H_{\mathrm{sus}} \end{bmatrix}. \tag{4.21}$$

The complete controller matrix reads:

$$\mathbf{C} = \begin{bmatrix} C_{\mathrm{df}} & 0 \\ 0 & C_{\mathrm{sus}} \end{bmatrix} = \boldsymbol{\Xi}\,\mathbf{G}\,\mathbf{H}. \tag{4.22}$$

The last part missing from the complete mathematical description of the dynamics of this model of LTP is the relationship between the interferometer data, $\vec{o}$, and the actual position of the test masses $\vec{q}$. The interferometer can of course only measure the position of the test masses with a finite precision limited by the so-called sensing noise $\vec{o}_{\mathrm{n}}$:

$$\vec{o}_{\mathrm{n}} = \begin{bmatrix} o_{\mathrm{n}_1} \\ o_{\mathrm{n}_{12}}. \end{bmatrix} \tag{4.23}$$

Moreover, the measurement involves a cross-talk between the two interferometer channels, δ, leading to the cross-talk matrix $\mathbf{S}$:

$$\mathbf{S} = \begin{bmatrix} 1 & 0 \\ \delta & 1 \end{bmatrix}. \tag{4.24}$$

The expression for the interferometer output $\vec{o}$ yields:

$$\vec{o} = \mathbf{S}\vec{q} + \vec{o}_{\mathrm{n}}, \tag{4.25}$$

The cross-talk, δ, originates from interferometer imperfections. It is not a noise source but a constant coupling factor. As such, the measurement of the position of

test mass 1 couples into the differential interferometer output, o_{12}, in addition to the actual measurement of $x_1 - x_{12}$, by the coupling factor δ. Thus, if both test masses are moving in common mode such that their separation remains constant, the second interferometer channel detects a spurious signal, o_{12}, proportional to the change in position of test mass 1. It shall be noted that the zero entry in the matrix is always true and as such not stated as a variable here. That is, there is no such coupling from the position of test mass 2 to the measurement of o_1.

With the full mathematical description of the dynamics of the test masses considering the simplifications described above, it is finally possible to set up an expression for the test mass accelerations. Combining Equations 4.20 and 4.25 leads to an expression for the acceleration of the test masses separating the well known parameters like the dynamics, the interferometer cross-talk, the controller functions, and the interferometer output data (left hand side of Equation 4.26) from the unknown ones that are the noise sources (right hand side of Equation 4.26). The analysis of the system in terms of the here presented mock data challenges (MDCs) is fully described by the following equation:

$$[\mathbf{DS^{-1} + C}]\vec{o} \ = \ \mathbf{D \ S^{-1}} \ \vec{o}_{\mathrm{n}} - \vec{g}_{\mathrm{n}}. \tag{4.26}$$

The right hand side of 4.26 will be denoted as *predicted acceleration*:

$$\vec{a}_{\mathrm{pred}} = \mathbf{D \ S^{-1}} \ \vec{o}_{\mathrm{n}} - \vec{g}_{\mathrm{n}}. \tag{4.27}$$

and the left hand side of Equation 4.26 will be denoted as *measured acceleration*:

$$\vec{a}_{\mathrm{meas}} = [\mathbf{DS^{-1} + C}]\vec{o} \tag{4.28}$$

Note that Equation 4.27 does not contain any simulated data. It consists only of parameters for which realistic assumptions have been made when the simulated data was generated. These will be unknown when LISA Pathfinder is producing real data, but the theoretical assumptions made for the data generation are used here for predicting the test mass acceleration. Thus $\vec{a}_{\mathrm{pred}}$ is computed by applying all information about the noise sources to the known dynamical model.

The result of Equation 4.28 gives $\vec{a}_{\mathrm{meas}}$. All variables in this equation will initially assumed to be known to some finite accuracy and precision:

- the dynamics, $\mathbf{D}$, from analysing the model,

- the IFO cross coupling, $\mathbf{S}$, from simulations,

- the controllers, $\mathbf{C}$, by definition and

- the IFO output, $\vec{o}$, from the measurement.

The dynamics, the interferometer cross-coupling and eventual controller actuation factors (see Equation 4.21) can be adapted according to the results of the on-orbit experiments. These experiments and the estimations of those parameters are the subject of Section 4.6. The expressions for the accelerations of the test masses derived here are the basis of all simulations of the LTP experiment already accomplished and of the ones still to be done. The complexity of the model will increase

further with the approach of the launch date of the mission. Thereby the model
will become more and more realistic. For instance, the one-dimensional experiment
needs to be extended to three dimensions. All computations accomplished in this
chapter consider a purely one-dimensional model, where all forces only act along
the sensitive x-axis such that resulting displacements also only affect this axis. In
reality residual forces act in all dimensions such that the applied control forces
will need to compensate for them. However, the one-dimensional model represents
the perfect scenario where cross-talks from other axes are minimised. Therefore in
principle the complete analysis presented in this thesis holds for such a scenario.

4.3 Validation of the model

The expressions derived above will be used throughout the LISA Pathfinder com-
munity for the preparation of the data analysis of the mission. Consequently, the
description of this first model of the LISA technology package needs to be verified.
Data from the first Mock Data Challenge (MDC1) is used for this verification pro-
cess. The data was generated by a sub-group of the data analysis group from the
University of Trento. The data generation procedure itself is not a subject of the
work presented here, but documented in [42].

The first step on the way to verifying the model is to recover the simulated
interferometer data. Together with the generated data, all details regarding quan-
tities relevant for the generation process were provided. Using the model that the
scientists involved agreed on, it is possible to compute an analytic expression for
the interferometer output in the frequency domain. Finally, inserting the provided
details about the relevant quantities will lead to the theoretical function of the
interferometer data in Fourier space.

The analytical expression for the model computed in the following will thus be
validated by comparing the 'power spectral densities' (PSDs) of the simulated time
series of the interferometer data with its computed analytical expression.

The model is considered as being validated when both the data generation group
and the data analysis group agree on one set of equations. Once the results match
and the analytical expression can be regarded as correct, it will be used for further
analysis such as the analytical function of the TM acceleration discussed in Section
4.4.

These analytical expressions play an important role especially in the beginning
of the process of the data analysis. Once they are validated they can be used as a
reference for the results obtained from actually analysing the simulated data.

Thus the task to fulfil in this section is to obtain a verified, analytical expression
for the TM accelerations that will represent the reference for the final data analysis
result, the 'conversion to acceleration' discussed in Section 4.5. From Equation 4.26
it follows that the interferometer data, $\vec{o}$, can be described as:

$$\vec{o} = [\mathbf{DS^{-1}} + \mathbf{C}]^{-1} \quad [\mathbf{DS^{-1}}\vec{o}_\mathrm{n} - \vec{g}_\mathrm{n}]. \tag{4.29}$$

This equation shows that for finding the function describing the interferometer
output, it is necessary to know all information used to generate the data, including

analytical expressions for the various noise sources summed up in $\vec{g}_n$. The frequency dependent functions describing the noise sources are communicated for MDC1 and can be found in [42]. Equation 4.29 needs to be solved to obtain an expression representing the PSDs of o_1 and o_{12}, since these can be computed directly from the data of MDC1 such that these two results can be compared. The first term of Equation 4.29 can be expressed as

$$[\mathbf{DS}^{-1} + \mathbf{C}]^{-1} = \begin{bmatrix} \frac{1}{(s^2+\omega_1^2)+C_{\mathrm{df}}} & 0 \\ \frac{-\beta}{((s^2+\omega_1^2)+C_{\mathrm{df}})((s^2+\omega_2^2)+C_{\mathrm{sus}})} & \frac{1}{(s^2+\omega_2^2)+C_{\mathrm{sus}}} \end{bmatrix}, \quad (4.30)$$

with $\beta = (\omega_2^2 - \omega_1^2) - \delta(s^2 + \omega_2^2)$. And the second term of Equation 4.29 can be expressed as:

$$\mathbf{DS}^{-1}\vec{o}_{\mathrm{n}} - \vec{g}_{\mathrm{n}} = \begin{bmatrix} (s^2 + \omega_1^2)o_{\mathrm{n}_1} + A_{\mathrm{N}} - A_1 \\ \beta o_{\mathrm{n}_1} + (s^2 + \omega_2^2)o_{\mathrm{n}_{12}} + A_1 - A_2 \end{bmatrix}. \quad (4.31)$$

Introducing the closed loop transfer functions

$$S_{\mathrm{df}} = \frac{(s^2 + \omega_1^2)}{(s^2 + \omega_1^2) + C_{\mathrm{df}}} \quad (4.32)$$

and

$$S_{\mathrm{sus}} = \frac{(s^2 + \omega_2^2)}{(s^2 + \omega_2^2) + C_{\mathrm{sus}}}, \quad (4.33)$$

Solving Equation 4.29 leads to the following expressions for $\vec{o}$:

$$o_1 = S_{\mathrm{df}}o_{\mathrm{n}_1} + \frac{S_{\mathrm{df}}}{(s^2 + \omega_1^2)}(A_{\mathrm{N}} - A_1), \quad (4.34)$$

$$o_{12} = o_{\mathrm{n}_1}\left[\beta\frac{S_{\mathrm{sus}}}{(s^2 + \omega_2^2)}(1 - S_{\mathrm{df}})\right] + o_{\mathrm{n}_{12}}S_{\mathrm{sus}} - A_2\frac{S_{\mathrm{sus}}}{(s^2 + \omega_2^2)} \quad (4.35)$$

$$+ A_1\left[\frac{S_{\mathrm{sus}}}{(s^2 + \omega_2^2)}\left(\beta\frac{S_{\mathrm{df}}}{(s^2 + \omega_1^2)} + 1\right)\right] - A_N\left(\beta\frac{S_{\mathrm{df}}S_{\mathrm{sus}}}{(s^2 + \omega_1^2)(s^2 + \omega_2^2)}\right). \quad (4.36)$$

For later comparison of the data it is useful to define the power spectral densities of the interferometer outputs. To verify that the results of the prediction of $\vec{o}$ are the same as the simulated mock data, 'power spectral densities' (PSD) of both will be compared graphically (Figure 4.4). The PSDs of o_1 and o_{12} are defined as:

$$\mathrm{PSD}(\mathrm{o}_1) = \langle \mathrm{o}_1 \cdot \mathrm{o}_1^* \rangle. \quad (4.37)$$

$$\mathrm{PSD}(\mathrm{o}_{12}) = \langle \mathrm{o}_{12} \cdot \mathrm{o}_{12}^* \rangle. \quad (4.38)$$

From this the 'amplitude spectral density' (ASD) can be calculated as:

$$\mathrm{ASD} = \sqrt{\mathrm{PSD}}. \quad (4.39)$$

Furthermore, the spectra of the force noises ($A_2 - A_1, A_N, o_n$) are also given as combinations of underlying ASDs from the data generation team which makes it impossible to recover all information about the variable itself, because the phase

information is missing. Moreover, some terms are statistically independent and their expectation value is defined to be zero. For instance: $\langle o_{n1} g_{n1} \rangle = 0$. Thus, the computation of the PSDs must be done analytically rather than simply using a computer algorithm for the complex conjugate of $\vec{o}$.

$$\mathrm{PSD}(o_1) = |S_{\mathrm{df}}|^2 |o_{n_1}|^2 + \frac{|S_{\mathrm{df}}|^2}{|(s^2 + \omega_1^2)|^2}(|A_1|^2 + |A_{\mathrm{N}}|^2), \tag{4.40}$$

$$\begin{aligned}
\mathrm{PSD}(o_{12}) = {} & |o_{n_1}|^2 \left[|\beta|^2 \frac{|S_{\mathrm{sus]}}|^2}{|(s^2 + \omega_1^2)|^2} \left(1 + |S_{\mathrm{df}}|^2 - 2Re\{S_{\mathrm{df}}\} \right) \right] \\
& + |o_{n_{12}}|^2 |S_{\mathrm{sus}}|^2 + |A_2|^2 \frac{|S_{\mathrm{sus}}|^2}{|(s^2 + \omega_2^2)|^2} \\
& + |A_1|^2 \frac{|S_{\mathrm{sus}}|^2}{|(s^2 + \omega_2^2)|^2} \left(|\beta|^2 \frac{|S_{\mathrm{df}}|^2}{|(s^2 + \omega_1^2)|^2} + 1 + 2Re\left\{ \beta \frac{S_{\mathrm{df}}}{(s^2 + \omega_1^2)} \right\} \right) \\
& + |A_{\mathrm{N}}|^2 \frac{|\beta|^2 |S_{\mathrm{df}}|^2 |S_{\mathrm{sus}}|^2}{|(s^2 + \omega_1^2)|^2 |(s^2 + \omega_2^2)|^2}. \tag{4.41}
\end{aligned}$$

Equation 4.40 describes the generation of the output data, o_1, as follows:
First the transfer function of the dynamics is applied to the sum of the force noises on test mass 1, A_1 and the force noise on the spacecraft, A_{N}. The resulting signal gets added to the measurement noise of the first interferometer channel o_{n_1}. After that the transfer function of the closed loop drag-free controller is applied to the signal which results in the power spectral density of o_1. Predicting o_{12} is more complicated, because here both o_1 and o_{12}, couple into the result. The result for o_{12} is given in Equation 4.41. 4.3 represents exactly the same content as expressed by Equation 4.40 but as a signal flow diagram.

Figure 4.4 shows two curves:

- the solid lines represent the analytical reference of the interferometer outputs, o_1 and o_{12}, computed above and

- the dashed lines are the ASDs computed directly from the simulated interferometer data that is to be analysed in the next section.

The noise sources assumed for the data generation are given in [43]. The interferometer noise, o_n has been assumed to be 3 pm/$\sqrt{Hz}$ above 10 mHz for both interferometers. The force noises can be read directly from Figure 4.7. For better visualisation of the differences of the analytical references and the simulated data the ratios are plotted in Figure 4.5.

These results show the consistency of the model of the LTP dynamics with the model that was originally used for the data generation. For both interferometer outputs the curves for the predicted and the generated results agree very well. Small deviations for frequencies in the sub-milihertz range most likely arise from averaging only very few points for the spectral estimation. For the spectrum of o_1 deviations also arise around 0.1 Hz, which may arise from differences in controllers used for simulating the data and computing the analytical function.

Finally all deviations are very small, which of course did not come for free. In reality the understanding of the defined model was naturally not the same from the
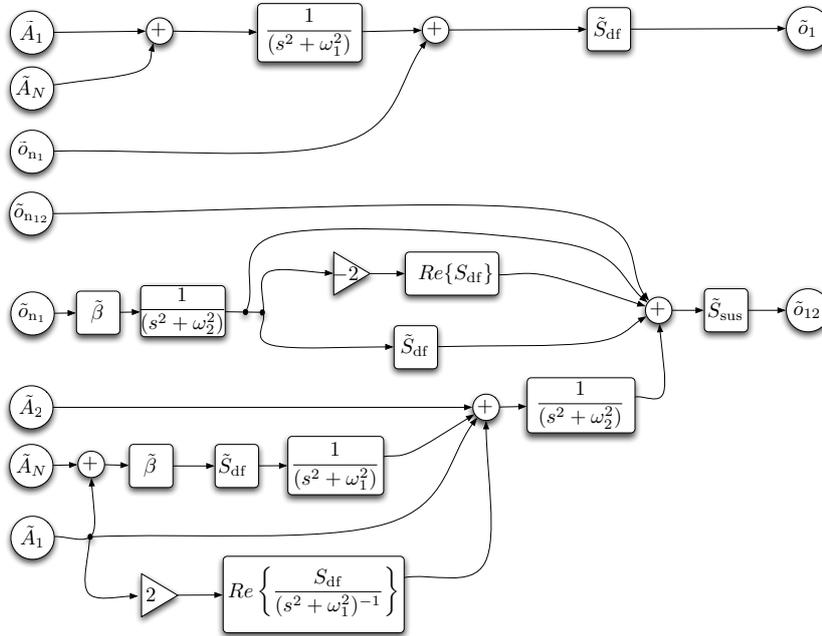
Figure 4.3: Flow diagram for recovering simulated interferometer data of the simplified dynamical model for LTP analytically. The tilde above the variables indicate that the quantities are given in the Fourier domain.
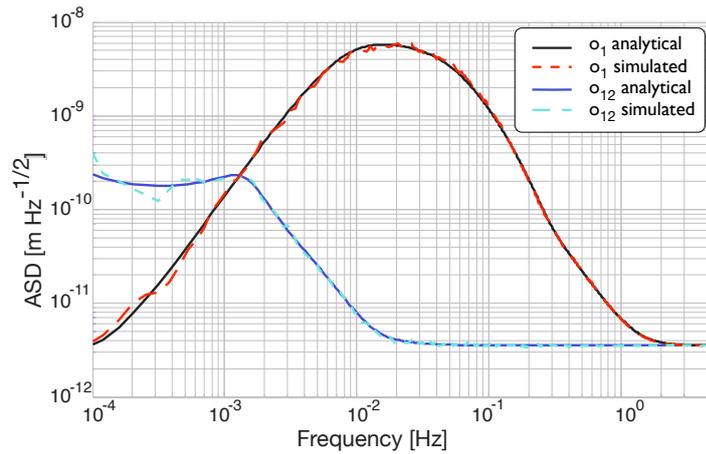

Figure 4.4: Amplitude spectral densities of the simulated interferometer data and its analytical prediction. The simulated data is equal to the interferometer measurement noise of 3 pm/$\sqrt{Hz}$ above 10 mHz.

(a) o_1 (b) o_{12}
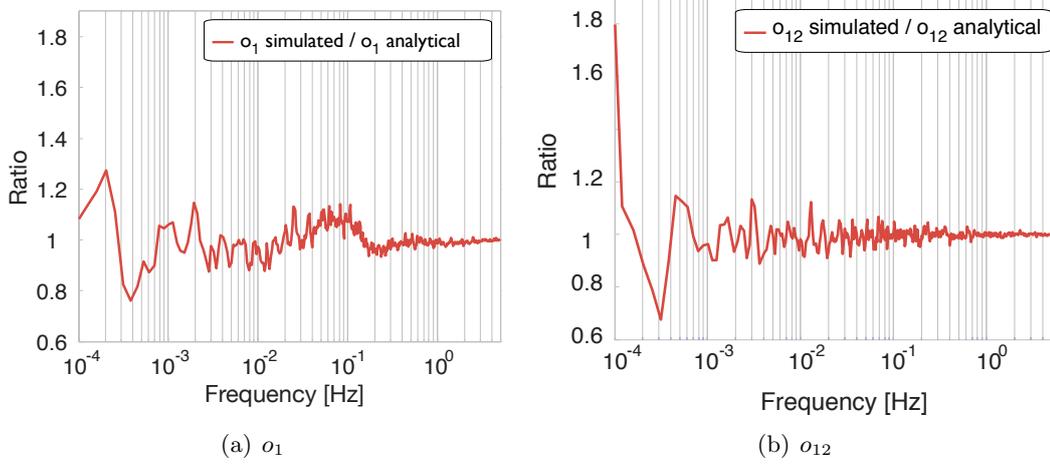
Figure 4.5: Ratio of the amplitude spectral density (ASD) of the simulated interferometer data in MDC1 and the analytical expression of the ASD of the interferometer data.

start for all scientists involved. Instead what is summarised in this section is the result of an intensive process of discussion and error hunting among both sub-groups. The result presented here forms the basis of a close and efficient collaboration between all groups working on the LISA Pathfinder mission.

From this point on, the computations regarding the test mass accelerations are practically straightforward, since the tools describing the model are all verified and in place and only need to be rearranged to compute the results described in the following two sections.

4.4 Computation of analytical interferometer signals

Since it has now been demonstrated that the formalism used to analyse the model is correct, the next step is to compute the expected results from the data analysis. This will provide the reference for the results obtained from converting the generated mock data back to acceleration. These references are given by the right hand side of Equation 4.26:

$$\vec{a} = \mathbf{D}\vec{o}_{\mathrm{n}}\,\mathbf{S^{-1}} + \vec{g}_{\mathrm{n}}. \tag{4.42}$$

The PSD of $\vec{a}$ is defined as:

$$\mathrm{PSD(a_1)} = |(s^2 + \omega_1^2)|^2|o_{\mathrm{n_1}}|^2 + |A_1|^2 + |A_{\mathrm{N}}|^2, \tag{4.43}$$

$$\mathrm{PSD(a_{12})} = |\beta|^2|o_{\mathrm{n_1}}|^2 + |(s^2 + \omega_2^2)|^2|o_{\mathrm{n_{12}}}|^2 + |A_2|^2 + |A_1|^2. \tag{4.44}$$

These equations are as well visualised in the form of a flow diagram in Figure 4.6.

Figure 4.7 shows the predictions for the accelerations of test mass 1, a_1, and the differential acceleration between the test masses, a_{12}, respectively. The spectra of the noise sources described earlier in this document are plotted for comparison.
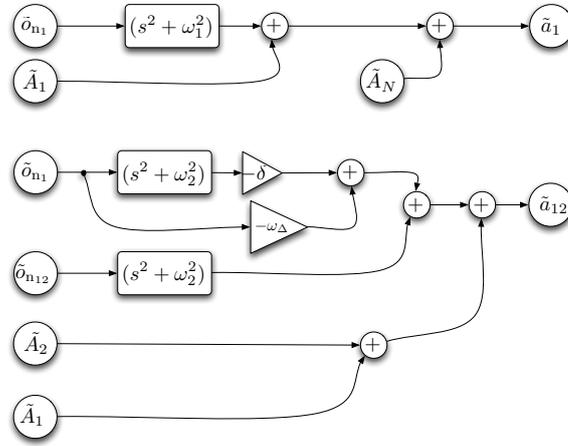
Figure 4.6: Visualisation of Equations 4.43 and 4.44. Depicted is the computation of an analytical expression representing the acceleration of the test masses in Fourier space. The spectra of the involved noise sources are well defined for the purpose of the performed simulations. The $\sim$ indicates that the respective variable is in the Fourier domain.
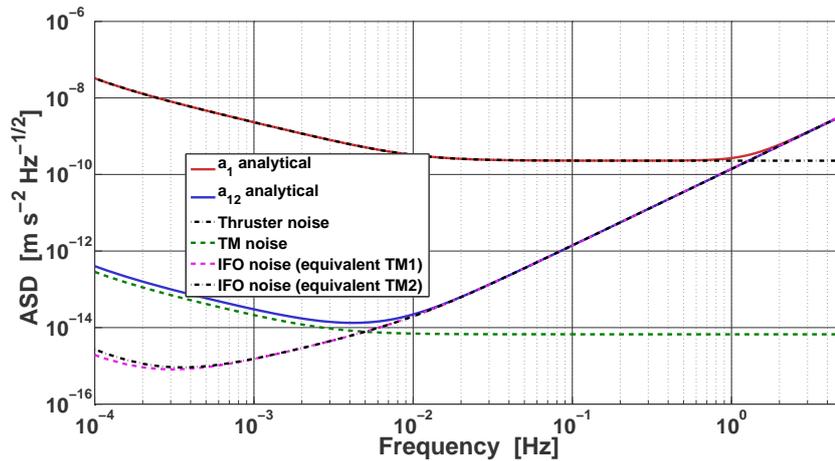
Figure 4.7: The impact of the different force noises on the predicted accelerations of test mass 1 and the differential acceleration of both test masses. The interferometer noise was transformed to represent an acceleration by multiplying it with the stiffness term of the corresponding test mass.

Figure 4.7 reflects very well the measurement principle of the science mode aiming to recover the residual accelerations of the test masses and thereby the acceleration noises acting on them.

As mentioned earlier in this chapter, the measurement of the acceleration of test mass 1, a_1, is dominated by the thruster noise A_N up to a frequency of about 1 Hz. Consequently, at these frequencies one cannot distinguish between test mass movement and spacecraft movement, which is consistent with our model because the spacecraft follows the motion of test mass 1 by applying a force via the micro Newton thrusters. Force noise acting on the thrusters should directly couple into the measurement of the acceleration of test mass 1. The residual acceleration noises, however, can only be extracted from the measurement of the differential acceleration a_{12}. The plot in Figure 4.7 shows that at Fourier frequencies up to $10\,\mu\text{Hz}$ the test mass noise is measured. At higher frequencies the interferometer sensing noise dominates the measurement. In Figure 4.7 two curves are plotted for the interferometer noise, one corresponding to each test mass. Unlike all other noise sources plotted, the sensing noise is not an acceleration noise but is a measure of distance. In order to make it comparable in the given context it is multiplied with the dynamics of the corresponding test mass and thereby corresponds to an acceleration. For test mass 1 the curve represents:

$$\frac{1}{(s^2 + \omega_1^2)}\, o_n \tag{4.45}$$

and for test mass 2 curve was computed by

$$\frac{1}{(s^2 + \omega_2^2)}\, o_n. \tag{4.46}$$

In fact the two curves only differ by the values for the corresponding stiffness terms and as such almost lie on top of each other.

4.5 Conversion of interferometer data to test mass acceleration

In the previous section the test mass acceleration expected from the known spectra of present noise sources has been derived to obtain a reference for the actual analysis of the generated data. Just as for the verification of the model in the first section of this chapter, the conversion of interferometer data to test mass acceleration is done exemplarily using data generated in the course of the first mock data challenge. Their spectra are plotted in Figure 4.4. The aim of the analysis of the simulated data is to extract an estimate of the differential acceleration of the two test masses. From those the residual acceleration noises can be calculated. Just like for the performance of this experiment on actual mission data, only the dynamics of the system and the controller transfer functions must be known for computing the test mass accelerations. The left hand side of Equation 4.26 gives the recipe of how to extract the TM acceleration, $\vec{a}$, from the interferometer output, $\vec{o}$:

$$\vec{a} = [\mathbf{DS^{-1}} + \mathbf{C}]\vec{o}, \tag{4.47}$$

which in principle states to differentiate the interferometer output twice and correcting for the measurement channel coupling as well as for the commanded forces. The resulting test mass accelerations are as follows:

$$
\begin{bmatrix} a_1 \\ a_{12} \end{bmatrix} = \begin{bmatrix} (s^2 + \omega_1^2)S_{\mathrm{df}}^{-1} & 0 \\ \omega_\Delta^2 + (s^2 + \omega_2^2) \cdot (-\delta) & (s^2 + \omega_2^2)S_{\mathrm{sus}}^{-1} \end{bmatrix} \begin{bmatrix} o_1 \\ o_{12} \end{bmatrix}. \qquad (4.48)
$$

The procedure is illustrated in Figure 4.8. The output of the second interferometer measurement channel, $\vec{o}_{12}$, couples into the calibration of the differential test mass acceleration, which is the reason for the pipeline of a_{12} being more complicated than that of a_1.
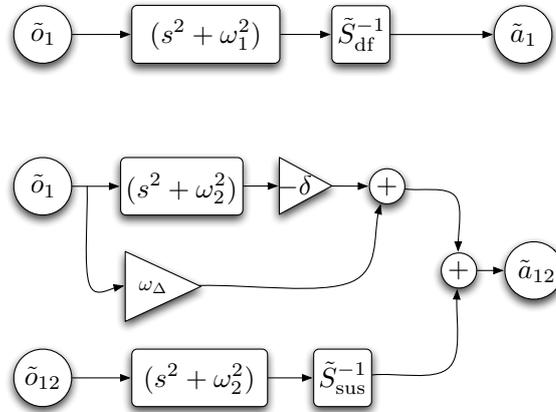
Figure 4.8: Data analysis of the interferometer output data. In the case of analysis in the Fourier domain, the boxes represent transfer functions, in case of time-domain analysis corresponding filters are defined.

The data analysis of the simulated interferometer data can be done in the Fourier domain as well as in the time-domain. In the case of the analysis in the time-domain, the elements in Figure 4.8 are generated using filters. Figure 4.9 shows the test mass accelerations which have been found by analysing eleven sets of simulated interferometer data and averaging the results. The error curves correspond to the standard deviation of the displayed mean and are indicated as dashed lines. The solid line represents the theoretical curve computed above using Equations 4.43 and 4.44. To make the differences better visible the ratios of analytical prediction and analysis of the simulated data are plotted in Figure 4.10. It can be seen that the results of the data analysis agrees very well with the theoretical predictions derived from analysing the dynamical model of LTP.

There are small deviations:

- At low frequencies these arise from the estimation of the spectral densities. At low frequencies a larger number of data points is needed to obtain a good spectral estimation.
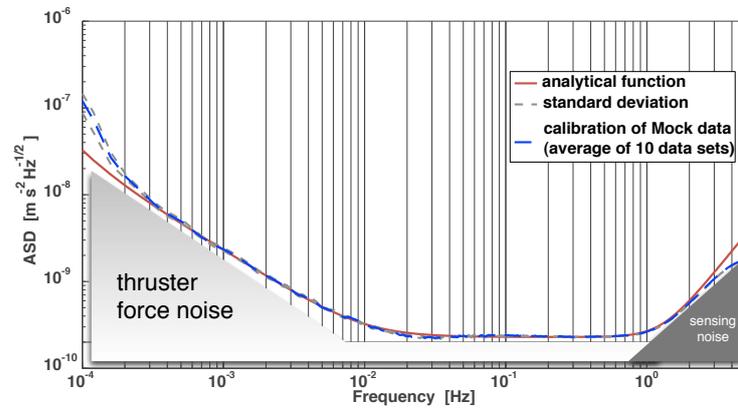
(a) Calibrated test mass 1 acceleration noise, a_1.
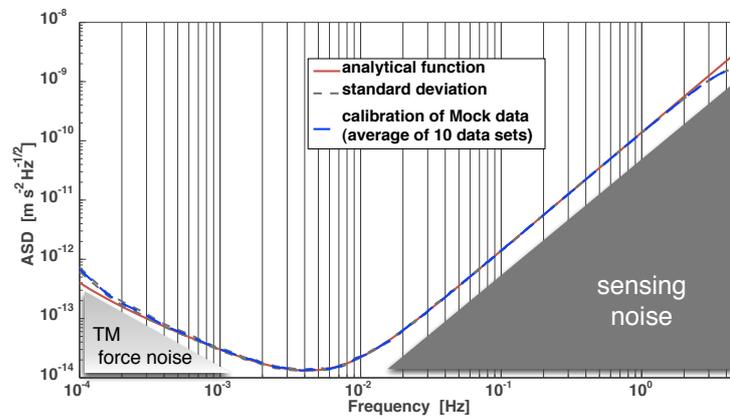
(b) Calibrated differential test mass 1 acceleration noise, a_{12}.

Figure 4.9: Calibration of interferometer data to test mass acceleration: plotted is average of the data analysis of 10 data sets together with the analytical prediction computed in Section 4.4. The grey dashed lines represent the standard deviation.
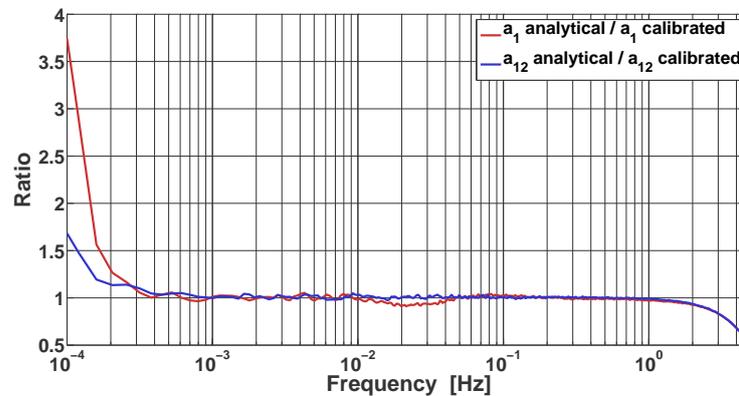
Figure 4.10: Ratio of the analytical prediction of the test mass acceleration and the calibration of the simulated data.

- The discrepancy between reference and analysis above 1 Hz arises from the method of numerical differentiation used in the time-domain analysis. The method is discussed in [44].

- The very small deviations around 10 mHz for a_1 arise from slightly different controller transfer functions for the analysis and for the generation of the data. For the ongoing investigations of MDC2 these have been corrected.

Analysing the data in the Fourier domain means that all elements in Figure 4.8 are transfer functions, which is indicated by the $\sim$. This means that initially the PSD of $\vec{o}$ is estimated and after that the transfer functions are applied. From Equation 4.48 and Figure 4.8, the following equations follow for the result of the data analysis in the frequency domain:

$$\mathrm{PSD}(a_1) \;=\; |o_1|^2 \frac{|(s^2 + \omega_1^2)|^2}{|S_{\mathrm{df}}|^2}, \tag{4.49}$$

$$\mathrm{PSD}(a_{12}) \;=\; |o_1|^2|\beta|^2 + |o_{12}|^2 \frac{|(s^2 + \omega_2^2)|^2}{|S_{\mathrm{sus}}|^2}$$
$$+\; \beta \frac{(s^2 + \omega_2^2)^*}{S_{\mathrm{sus}}^{\;*}} - o_{1,\Delta} + \beta^* \frac{S\omega_2}{S_{\mathrm{sus}}} o_{1,\Delta}\,, \tag{4.50}$$

where $o_{1,\Delta} = \langle o_1\, o_{12}^* \rangle$ is the cross correlation.
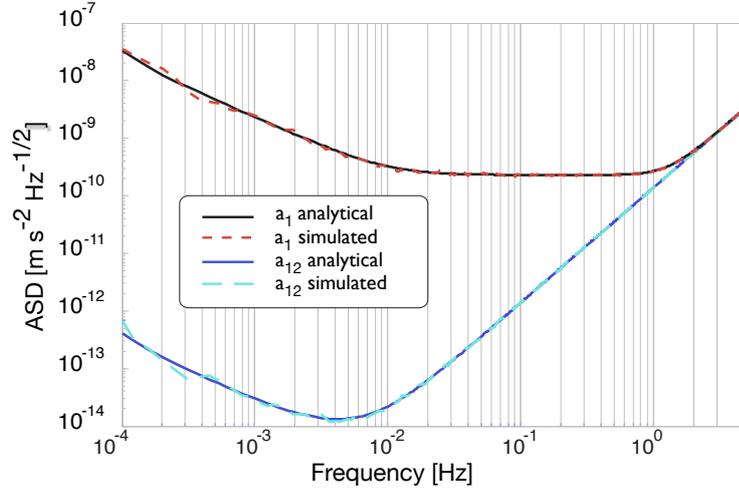
Figure 4.11: Calibration of one set of interferometer data to test mass acceleration in the frequency-domain. The analytical prediction computed in Section 4.4.

The resulting functions for $\vec{a}$ in the frequency-domain are plotted in 4.11. For this task only one set of data was analysed, instead of averaging all 10 interferometer data sets like for the time-domain analysis. They agree with the results of the time-domain analysis (Figure 4.9), apart from the deviations above 1 Hz which arise from the numerical differentiation. From the results shown it can be said that the theoretical predictions are in very good agreement with the outcome of the data analysis.

Along with this analysis a set of LTPDA tools has been developed. They are ready to be used on any data simulated according to the here defined model of LTP. With an expansion of the dynamics e.g. taking into account cross-talks from other axes, the tools shall be expanded accordingly.

Up to this point, all results shown involving simulated data, are done on data generated in the course of MDC1. Any data generated according to the model described here (see Section 4.2) can now be analysed using the developed tools. This of course presumes the knowledge of all the parameters in Equations 4.49 and 4.50.

The question of finding the parameters, which might not be known precisely enough is the subject of the following section and the topic of the second Mock Data Challenge (MDC2).

4.6 Parameter estimation

To perform the tasks discussed in the previous section (calibration of interferometer data to test mass acceleration), all parameters describing the dynamical model underlying the simulated or real data must be known. Uncertainties in the parameter values will of course propagate into the data analysis result, that is the acceleration of the test masses.

In reality, during the mission some parameters might not be known exactly. Therefore, before the interferometer data can, for example, be converted to test mass acceleration, these parameters need to be estimated. The observeability of the parameters, which depend on the dynamical system improves when the system is stimulated by appropriate input signals. Naturally the next Mock Data Challenge (MDC2) is designed in the following way:

Like in MDC1 two sub-groups are formed: one generating data and the other one analysing the simulated data.

Data generation The simulation comprises a set of experiments where signals are injected into the system and the output is generated.

Data analysis The data analysis task is to find certain unknown parameter values from the generated data. Given the injected input signals the transfer functions can be measured and the parameters can be approximated by appropriate fitting algorithms.

Different approaches for the actual parameter estimation (non-linear, linear and Bayesian) have been investigated by the data analysis group of LTP. In the scope of the work presented here, the focus lies on the linear analysis of the system. In this section the method with which the system is stimulated and how the estimation of the desired parameters is realised will be explained. Finally, together with the obtained results, the applicability of the linear approximation of the simplified model of LTP shall be discussed.

4.6.1 Defining the simulation procedure

During the mission as well as in the course of simulations, the stimulation of the dynamical system can be realised by injecting stimulus signals at the input of the controllers (C_{df} and C_{sus}) by adding them to the error signal. Ideally, when cross-talks from other axes are minimised, the injection introduces a longitudinal motion of the two test masses within the respective loop bandwidth. In fact, the dynamical model underlying the simulations discussed in this thesis (see Section 4.2) represents this ideal one-dimensional case. From the resulting output data and the injected input signals, the transfer functions are measured. Figure 4.12 illustrates how the system is stimulated by the injected signals $o_{\mathrm{i}1}$ and $o_{\mathrm{i}2}$.
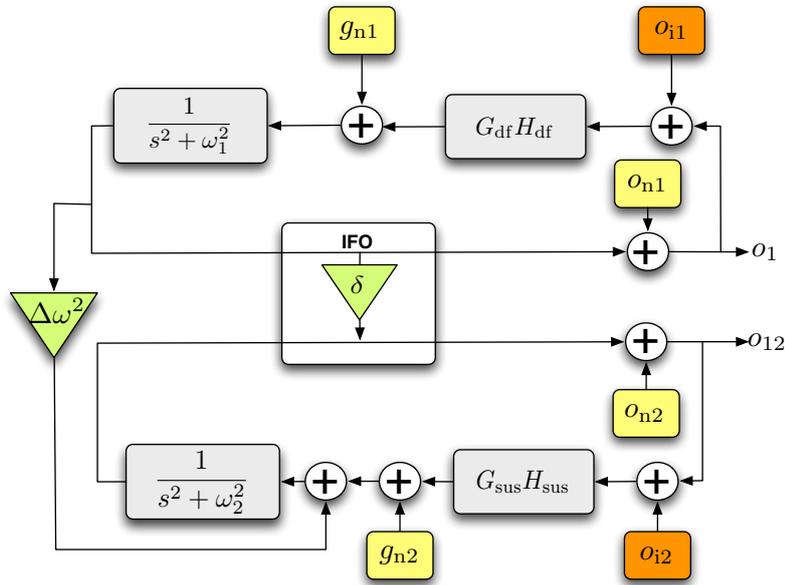
Figure 4.12: Extension of Figure 4.2: the simplified model of the LTP dynamics including input signals $o_{\mathrm{i}1}$ and $o_{\mathrm{i}12}$ stimulating the system.

The diagram in Figure 4.12 depicts very well that the model comprises two control loops which are coupled via two terms:

- the differential stiffness of the test masses, $\Delta\omega = \omega_2^2 - \omega_1^2$ and

- the interferometer cross-talk, δ.

The two control loops each consist of a controller term and a term describing the dynamics of the two test masses respectively. The drag-free loop controls the position of the spacecraft with respect to test mass 1 and the electro-static suspension loop controls the position of test mass 2 relative to TM 1 as discussed in the description of the science mode in Section 4.2.

In the Laplace domain a transfer function is defined as the ratio of the output of a system to the input of a system. Given an input function $X(s)$, and an output

Figure 4.13: Schematic of the definition of the transfer function.

function $Y(s)$ (Figure 4.13), the transfer function $T(s)$ is defined as:

$$T(s) = \frac{Y(s)}{X(s)}. \tag{4.51}$$

In the simulations for the investigations on parameter estimation methods described here, the inputs to the system are always o_{i1} and o_{i12} (see Figure 4.12). Within the simulations performed, data sets that include the input signals are provided by the data generation team[1]. As this data mimics the interferometer output as in the previous sections, it will again be denoted as o_1 and o_{12} for the two interferometer output channels.

With the known input signals and the provided simulated data sets the appropriate transfer functions are measured according to Equation 4.51. Some parameters of the model are not communicated to the analysis team and instead have to be estimated from the measured transfer functions by suitable fitting routines. For this purpose the measured transfer functions need to be fitted to the theoretical models derived from the equations of the dynamics for LTP. To derive this model, Equation 4.20 from Section 4.2 is modified to include the input signals $\vec{o}_i$:

$$\mathbf{D}\vec{q} = -\mathbf{C}(\vec{o} + \vec{o}_i) - \vec{g}_n. \tag{4.52}$$

This leads to the needed expression for the interferometer output $\vec{o}$. Equation 4.29 becomes

$$\vec{o} = [\mathbf{D}\mathbf{S}^{-1} + \mathbf{C}]^{-1} \quad [-\mathbf{C}\vec{o}_i - \vec{g}_n + \mathbf{D}\mathbf{S}^{-1}\vec{o}_n]. \tag{4.53}$$

Now the interferometer output from Equation 4.53 can be expressed as a combination of a set of transfer functions, each acting on one of the three types of inputs to the system, the interferometer sensing noise, $\vec{o}_n$, the residual acceleration noises $\vec{g}_n$ and the input signals o_i:

$$\vec{o} = \mathbf{T}(\tilde{\theta}) \ \vec{o}_i + \mathbf{T}_n(\tilde{\theta}) \ \vec{g}_n + \mathbf{T}_{on}(\tilde{\theta}) \ \vec{o}_n, \tag{4.54}$$

where $\tilde{\theta}$ denotes the model parameters to be estimated.

The parameter estimation procedures discussed in this thesis represent the very first attempt of finding unknown model parameters from simulated LTP interferometer data. As such, the model is kept simple by not taking into account the parameter dependent noise terms $\vec{g}_n$ and $\vec{o}_n$ (see Figure 4.12 and Section 4.2). Instead the signals $\vec{o}_i$ are assumed to be large enough to dominate the interferometer output. It will be shown in Section 4.6.5 that the parameter dependence of the

---

[1]The data generation team for parameter estimation simulations are: Miquel Nofrarias (AEI) and Luigi Ferraioli (UTN).

noise terms cannot be neglected and an appropriate method to deal with this kind of noise will be presented.

The system simplifies to

$$\vec{o} = \mathbf{T}(\tilde{\theta}) \ \vec{o}_{\mathrm{i}} + \vec{n}, \tag{4.55}$$

where the parameter dependent noise terms get summarised into the single term, $\vec{n}$. The matrix, $\mathbf{T}(\tilde{\theta})$, represents all possible transfer functions from the introduced signals. It reads:

$$\mathbf{T}(\tilde{\theta}, \mathbf{f}) = \left[\begin{array}{c} -\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}(f)}{(s^2 + \omega_1^2) + G_{\mathrm{df}}H_{\mathrm{df}}(f)} \\ -\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}(f)(\omega_1^2 - \omega_2^2 + \delta(s^2 + \omega_2^2))}{((s^2 + \omega_1^2) + G_{\mathrm{df}}H_{\mathrm{df}}(f))((s^2 + \omega_2^2) + G_{\mathrm{sus}}H_{\mathrm{sus}}(f))} \\ 0 \\ -\dfrac{G_{\mathrm{sus}}H_{\mathrm{sus}}(f)}{(s^2 + \omega_2^2) + G_{\mathrm{sus}}H_{\mathrm{sus}}(f)} \end{array} \right]. \tag{4.56}$$

From this matrix the transfer functions of the system from each of the two inputs $\vec{o}_{\mathrm{i}}$ to the output of the two measurement channels are found as

$$\left[\begin{array}{c} o_1 \\ o_{12} \end{array} \right] = \left[\begin{array}{cc} T_{11} & 0 \\ T_{12} & T_{22} \end{array} \right] \left[\begin{array}{c} o_{\mathrm{i}1} \\ o_{\mathrm{i}2} \end{array} \right] + \left[\begin{array}{c} n_1 \\ n_2 \end{array} \right]. \tag{4.57}$$

This construct of transfer functions based on the basic model of the LTP dynamics and the two input signals injected at the controllers represents the model for the parameter estimation algorithms. In Section 4.7 the transfer functions for each experiment performed to simulate interferometer data computed via this matrix are used. The model including all transfer functions is completely implemented in LTPDA, enabling everybody with access to the simulated data, to easily analyse the data according to the presented formalism. With future runs of simulations (MDCs) the complexity of the model will increase and the implemented transfer functions will of course be enhanced accordingly.

The model parameters whose values are not communicated by the data generation team and thus need to be estimated are:

ω_1^2 - the stiffness of TM1,

ω_2^2 - the stiffness of TM2,

G_{df} - the controller gain of the drag free loop,

G_{sus} - the controller gain of the electrostatic suspension loop and

δ - the interferometer cross-talk.

Finally the true values must be communicated to evaluate the results of the parameter estimation accomplished by the data analysis team. Thus they can be presented here and are listed in Table 4.1. The set of unknown parameters will in the following be denoted as $\vec{\theta}$.

Table 4.1: Unknown parameters and their correct values, which are to be estimated.

Parameter	value
G_{df}	0.8
G_{sus}	1.15
ω_1^2	-1.1×10^{-6}
ω_2^2	-2.2×10^{-6}
$\omega_{\mathrm{match}}^2$	-2.4×10^{-6}
δ	1.35×10^{-4}.

As mentioned above and reflected in the derived equations (Equation 4.55), the parameter dependence of $\vec{n}$ will not be considered. Nevertheless, the noise plays an important role in the parameter estimation procedure. Prevalent fitting routines require white Gaussian noise as input data, but $\vec{n}$ is non-white and parameter dependent. A common approach to solving the former but not the latter is applying so called 'whitening filters' to whiten the noise and then apply the fitting routine to the data with the whitened noise. Regarding the parameter dependence, the signal is assumed to be significantly larger than the noise and the issue can be neglected. However, the two most obvious issues of whitening the described simulated interferometer data are pointed out in the following:

1. In the here discussed simulated interferometer data as well as in future mission data the two noise terms n_1 and n_2 of the two interferometer channels are highly correlated. For finding appropriate whitening filters this correlation must be taken into account which makes the process rather complex. A so-called two-dimensional whitening tool has been implemented for this purpose. It is part of the LTPDA toolbox, in whose manual [23] further information on the algorithm can be found.

2. Data and noise are inseparable and therefore applicable whitening filters, are applied to both. Therefore the model must be filtered by the exact same whitening filters as the data such that the procedure of noise whitening only changes the noise term while the relation between data and model stays the same.

For whitening two correlated signals an advanced algorithm has been implemented. The function is called `whiten2D` and it is included in the LTPDA Toolbox. It was developed by Luigi Ferraioli from the University of Trento. For further information on this algorithm see [23]. The mathematical description of the relationship between the whitened data, the whitening filters as found by `whiten2D` and the transfer functions of the model are carried out in Section 4.6.5. To minimise unknown issues,

data without the noise terms $\vec{o}_\mathrm{n}$ and $\vec{g}_\mathrm{n}$ has been generated and instead white noise of comparable magnitude has been added to it in every experiment. The analysis with aiming the estimation of parameters is in the following sections always done first on the data with white noise, before the final parameter estimation is done on the original data, where the more realistic noise terms, $\vec{o}_\mathrm{n}$ and $\vec{g}_\mathrm{n}$, are included and whitened.

4.6.2 Experiments

There exist a couple of documents [45, 42], suggesting a variety of experiments necessary to maximise the science output of the mission and hence its value as technology demonstration for LISA. The experiments performed for the simulations, which are the subject of this thesis are inspired by these but are not completely the same. The focus of the work presented here lies in the implementation and investigation of suitable fitting routines rather than probing every suggested experiment. Nevertheless, the investigations on the different kinds of parameter estimation techniques should reveal how well the performed experiments are suited for measuring certain parameters.

The simulation performed to investigate and implement different methods of parameter estimation on the example of the presented LTP setup, comprises a set of three experiments. For each experiment a set of interferometer data, $\vec{o}$, is generated and will be analysed in order to estimate the values of the parameters listed at the end of the previous section. For this purpose, the data will be fitted to the theoretical expressions of the corresponding transfer functions defined by Equation 4.57. The analysis of the three different experiments and corresponding transfer functions will allow a comprehensive evaluation of the linear parameter estimation method introduced here.

In the following, the experiments will simply be denoted as Experiment 1 to 3. Figures 4.12, 4.22 and 4.19 illustrate the setup of each experiment accordingly. They shall be described here only briefly:

Experiment 1 : Two signals, o_i1 and o_i2, are injected into the system.

Experiment 3 : Only the signal o_i1 is injected into the system.

Experiment 2 : Like Experiment 3, but the stiffness parameters ω_1 and ω_2 are set to the same value which is renamed to the parameter ω_match.

The experiments presented are described by a document accompanying the generated data sets [46]. The document includes all necessary information for the data analysis team to analyse the given data. In Section 4.7 each experiment is discussed in detail and the results of important investigations are presented. The focus of the analysis lies in the investigations of applicable parameter estimation methods, in particular how well the model can be approximated by a linearisation, in order to estimate the parameters using a straightforward linear fitting routine.

The relevant transfer functions for the above listed experiments result from Equation 4.56 and are plotted in Figure 4.14. Moreover in Figure 4.14 arrows indicate the frequencies of interest. In each experiment the system is stimulated at certain

frequencies and the output is measured. As presented earlier (in Equation 4.51) the measured output over the input signal yields the transfer function. The frequencies that are indicated by the arrows in the plot of the transfer functions will be listed later for each experiment individually in the corresponding tables (Tables 4.4, 4.6 and 4.8 in Section 4.7) for the description of every input signal.
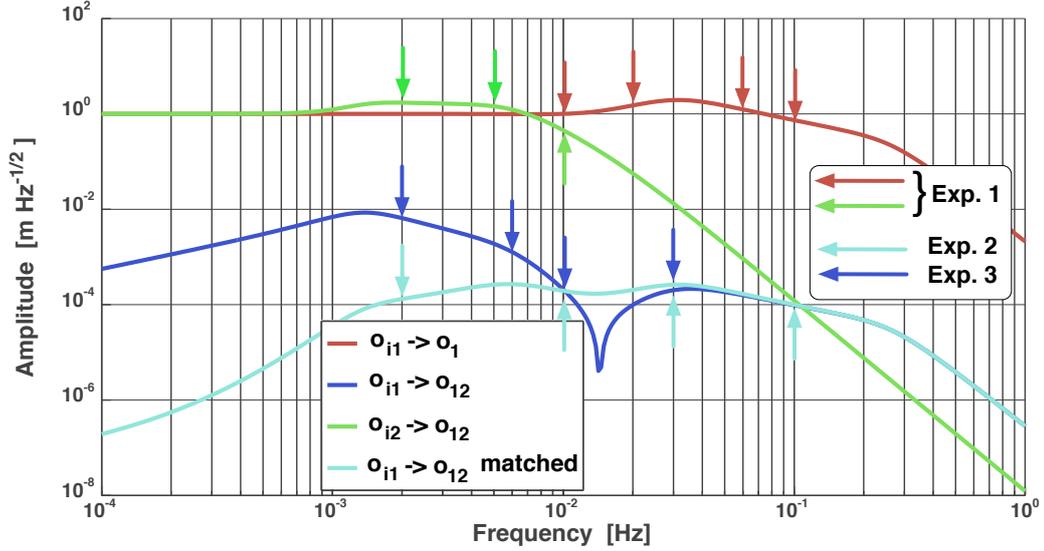
Figure 4.14: Transfer functions $\mathbf{T}(\vec{\theta})$ from stimulus signals o_{i1} and o_{i2} to interferometer data o_1 and o_{12} the last curve represents Experiment 2, where ω_1 and ω_2 are set to ω_{match}.

4.6.3 Linearisation of the model

The unknown values of the model parameters are to be found by comparing the model given in the matrix of Equation 4.56 with the simulated data. This can be done with the 'Least-squares method'. General information on this method is given in several textbooks on data analysis like [36] and [47]. The main principle is that the optimal parameter set is found by minimising the so-called *merit* or *residual* function, that represents the difference between the theoretical model and the data. The residual function is defined as:

$$r(\vec{x}, \vec{y}, \vec{\theta}) = \sum_i (y_i - f(x_i, \vec{\theta}))^2, \tag{4.58}$$

where $\vec{\theta}$ is the set of unknown parameters, $\vec{y}$ are the measurements and $f(x_i, \vec{\theta})$ is the model function defined in Equation 4.56. The optimal parameter set, θ_{opt}, minimises the residual function, r. In the following equation, the expression $\underset{\vec{\theta}}{\mathrm{argmin}}$ indicates the minimum of the function for the parameter set $\vec{\theta}$.

$$\vec{\theta}_{\mathrm{opt}} = \underset{\vec{\theta}}{\mathrm{argmin}}(r(\vec{x}, \vec{y}, \vec{\theta})), \tag{4.59}$$

Thus, the parameter can be found by finding the minimum of the above residual function. This is the principle non-linear least squares fitting routines are based on. In Section 4.7 the residual function is plotted for all data sets of the three experiments under investigation (see Figures 4.18(a), 4.24(a) and 4.21(a)). The residual function has been computed for a set of parameter values, $\vec{\theta_0}$, around the true values of $\vec{\theta}$, and the minimum should always lie at the true parameter value.

The model $f(x_i, \vec{\theta})$ is non-linear in its parameters, which makes it necessary to use iterative non-linear parameter estimation methods to find the optimal parameter set $\vec{\theta}_{\text{opt}}$ that minimises the residual function $r(\vec{x}, \vec{y}, \vec{\theta})$.

The focus of this thesis, however, is the linear analysis of the model. A common approach in estimating parameters of non-linear functions is to linearise the function first and then use a straightforward, well understood, linear fitting algorithm. A non-linear model can in principle be approximated by a linear expression using the Taylor expansion and considering only the first term:

$$f(\vec{\theta}, \vec{x}) \approx f(\vec{\theta_0}) + f_{\text{lin}}(\vec{\theta_0}, \vec{\theta}), \tag{4.60}$$

$$f(\vec{\theta}, \vec{x}) \approx f(\vec{\theta_0}) + \sum \frac{\partial f}{\partial \theta_k}(\vec{\theta_0}, \vec{x})\Delta\theta_k, \tag{4.61}$$

$$f(\vec{\theta}, \vec{x}) - f(\vec{\theta_0}) \approx \sum \frac{\partial f}{\partial \theta_k}(\vec{\theta_0}, \vec{x})\Delta\theta_k, \tag{4.62}$$

where $\vec{\theta}$ are the unknown parameters and $\vec{\theta_0}$ are a first estimation for each unknown value. These first estimations may be known, for example, from a measurement or a theoretical evaluation. The above approximation holds only for a certain interval around the unknown parameter. The interval depends on the function f and is given by $\vec{\theta} - \vec{\theta_0}$, where $\vec{\theta}$ represents the correct values of the parameters used during the simulation. Those values are given in Table 4.1. The right hand side of Equation 4.62 is an expression that is linear in $\Delta\vec{\theta}$ and thus can be fitted by a linear fitting routine. In contrast to the non-linear approach stated in Equation 4.59 the simulated data denoted by y_i cannot simply be fitted to the linear model f_{lin}. Instead, projecting the statement made by Equation 4.58 to the linear case yields

$$r_{\text{lin}}(\Delta\vec{\theta}) = \sum_i^N \left([y_i - f(x_i, \vec{\theta_0})] - f_{\text{lin}}(x_i, \Delta\vec{\theta}) \right)^2,$$

$$r_{\text{lin}}(\Delta\vec{\theta}) = \sum_i^N \left([y_i - f_i(\vec{\theta_0})] - \sum_k^M \frac{\partial f}{\partial \theta_k}(\vec{\theta_0}, \vec{x_i})\Delta\theta_k \right)^2. \tag{4.63}$$

Thus, the data that the linear expression f_{lin} is fitted to, is the measured or simulated data y_i minus the non-linear model function. The model function is computed by inserting the first estimation for each parameter, $f(\vec{\theta_0})$. The result visualises the region $\Delta\vec{\theta}$ for which Equation 4.62 holds:

Plotting r_{lin} against $\Delta\theta$ for a set of different estimations, $\vec{\theta_0}$, around the correct value illustrates within which region the linearisation is a good approximation of the model function. Figure 4.18(b) on page 105 shows this plot for Experiment 1 where the noise in the interferometer output, $\vec{o}$, is white. This investigation is done for all

three experiments on data with white noise. However, results must be the same for the simulated data with whitened noise since the purpose of whitening is to be able to treat the whitened noise exactly the same as white noise. The described plots of r_{lin} against $\Delta\theta$ are done for every experiment. Figure 4.21(b) shows the curve for Experiment 3 and 4.24(b) shows the same for Experiment 2. These Figures provide information on how well the model function can be linearised in each parameter for a certain region. In Section 4.7 these plots will help understanding the obtained results.

Equation 4.62 gives the expression that builds the input of the linear fitting routine: a linear combination of unknowns to be estimated on the right hand side, and on the left hand side the so-called 'target function' including the measurements. For the model under investigation, described by Equation 4.56, this linearisation according to Equation 4.62 yields the following expression:

$$
\begin{aligned}
(o_1 + o_{12}) \quad - \quad & [(T_{11}(\vec{\theta}_0) + T_{12}(\vec{\theta}_0))o_{i1} + T_{22}(\vec{\theta}_0)o_{i2}] \approx \\
& \sum_k \left[\left(\left.\frac{\partial T_{11}}{\partial \theta_k}\right|_{\theta_{k0}} + \left.\frac{\partial T_{12}}{\partial \theta_k}\right|_{\theta_{k0}} \right) o_{i1} + \left.\frac{\partial T_{22}}{\partial \theta_k}\right|_{\theta_{k0}} o_{i2} \right] \Delta\theta_k \quad (4.64) \\
= \quad & \left[\left(\left.\frac{\partial T_{11}}{\partial G_{\text{df}}}\right|_{G_{\text{df}0}} + \left.\frac{\partial T_{12}}{\partial G_{\text{df}}}\right|_{G_{\text{df}0}} \right) o_{i1} \right] (G_{\text{df}} - G_{\text{df}_0}) \\
+ \quad & \left[\left(\left.\frac{\partial T_{12}}{\partial G_{\text{sus}}}\right|_{G_{\text{sus}0}} \right) o_{i1} + \left.\frac{\partial T_{22}}{\partial G_{\text{sus}}}\right|_{G_{\text{sus}0}} o_{i2} \right] (G_{\text{sus}} - G_{\text{sus}_0}) \\
+ \quad & \left[\left(\left.\frac{\partial T_{11}}{\partial \omega_1^2}\right|_{\omega_{1_0}^2} + \left.\frac{\partial T_{12}}{\partial \omega_1^2}\right|_{\omega_{1_0}^2} \right) o_{i1} \right] (\omega_1^2 - \omega_{1_0}^2) \\
+ \quad & \left[\left(\left.\frac{\partial T_{12}}{\partial \omega_2^2}\right|_{\omega_{2_0}^2} \right) o_{i1} + \left.\frac{\partial T_{22}}{\partial \omega_2^2}\right|_{\omega_{2_0}^2} o_{i2} \right] (\omega_2^2 - \omega_{2_0}^2) \\
+ \quad & \left[\left(\left.\frac{\partial T_{12}}{\partial \delta}\right|_{\delta_0} \right) o_{i1} \right] (\delta - \delta_0). \quad (4.65)
\end{aligned}
$$

o_1 and o_{12} were summed in order to fit both channels simultaneously and thus make use of all information gained from these measurements. By adding the two interferometer channels together, the resulting expression is again a linear combination of model functions and unknown parameters as required for the linear parameter estimation method and represented in Equation 4.62.

From Equation 4.56 it can be seen that some derivatives are independent from certain parameters and hence the derivative with respect to these parameters are zero. The zero derivatives are not included in the above expression. The derivatives of the three transfer functions (T_{11} T_{12} and T_{22}) with respect to all parameters are displayed in a matrix like notation in Table 4.2.

A first approximation of the parameters is given by the data generation team (Table 4.3). During the mission it is expected that these parameters can also be initially estimated. This first approximation is $\vec{\theta}_0$. The linear fitting routine will

Table 4.2: Complete set of derivatives of all transfer functions relevant for MDC2 with respect to all parameters. Top: Derivatives of T_{11} and $T22$ with respect to all parameters. Bottom: Derivatives of T_{12} with respect to all parameters

	T_{11}	T_{22}
∂G_{df}	$\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}^2}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)^2} - \dfrac{H_{\mathrm{df}}}{s^2 + G_{\mathrm{df}}H_{\mathrm{df}}^2 - \omega_1^2}$	0
$\partial G_{\mathrm{sus}}$	0	$\dfrac{G_{\mathrm{sus}}H_{\mathrm{sus}}^2}{(s^2+G_{\mathrm{sus}}H_{\mathrm{sus}}+\omega_2^2)^2} - \dfrac{H_{\mathrm{sus}}}{s^2+G_{\mathrm{sus}}H_{\mathrm{sus}}^2-\omega_2^2}$
$\partial \omega_1^2$	$\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)^2}$	0
$\partial \omega_2^2$	0	$\dfrac{G_{\mathrm{sus}}H_{\mathrm{sus}}}{(s^2 + G_{\mathrm{sus}}H_{\mathrm{sus}} + \omega_2^2)^2}$
$\partial \delta$	0	0

	T_{12}
∂G_{df}	$\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}^2(-\Delta\omega + \delta(s^2 + \omega_2^2))}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)^2(s^2 + \omega_2^2)} - \dfrac{H_{\mathrm{df}}(-\Delta\omega + \delta(s^2 + \omega_2^2))}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)(s^2 + G_{\mathrm{sus}}H_{\mathrm{sus}} + \omega_2^2)}$
$\partial G_{\mathrm{sus}}$	$\dfrac{G_{\mathrm{df}}H_{\mathrm{sus}}H_{\mathrm{df}}(-\Delta\omega + \delta(s^2 + \omega_2^2))}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)(s^2 + G_{\mathrm{sus}}H_{\mathrm{sus}} + \omega_2^2)^2}$
$\partial \omega_1^2$	$-\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}(-\Delta\omega + \delta(s^2 + \omega_2^2))}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)^2(s^2 + G_{\mathrm{sus}}H_{\mathrm{sus}} + \omega_2^2)}$
$\partial \omega_2^2$	$-\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}((-1+\delta)G_{\mathrm{df}}H_{\mathrm{df}}}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)(s^2 + G_{\mathrm{sus}}H_{\mathrm{sus}} + \omega_2^2)} + \dfrac{G_{\mathrm{df}}H_{\mathrm{df}}(-\Delta\omega + \delta(s^2 + \omega_2^2))^2}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)(s^2 + G_{\mathrm{sus}}H_{\mathrm{sus}} + \omega_2^2)^2}$
$\partial \delta$	$\dfrac{G_{\mathrm{df}}H_{\mathrm{df}}(s^2 + \omega_2^2)}{(s^2 + G_{\mathrm{df}}H_{\mathrm{df}} + \omega_1^2)(s^2 + G_{\mathrm{sus}}H_{\mathrm{sus}} + \omega_2^2)}$

return the parameter $\Delta\theta_k = \theta_k - \theta_0$, where θ_k stands for the value of each parameter to be estimated. Thus in fact the parameter is estimated by finding the difference between the a priori estimation and the correct value. $\theta_k = \Delta\theta_k + \theta_0$ is then inserted into the model equations for the experiment and the result gets checked against the simulated data o_1 and o_{12}. If the difference between model and data gets smaller, the old parameter estimation θ_0 is overwritten by the new θ_k and the linear fit is repeated. This iteration procedure is terminated if the difference between model and data has not decreased significantly in comparison to the result of the previous iteration. The flow chart describing the iterative linear fitting process is shown in Figure 4.15.

The iterations are necessary because the model is in fact non-linear and only approximated by the first term of the Taylor expansion. The number of iterations depends on how much the first approximation differs from the correct value. There is no guarantee for convergence, but the expectation for convergence is greater when the first term of the Taylor series (see Equation 4.61) is a good approximation in the region over which θ_k is varied during the procedure. This implies that particularly in highly non-linear problems like the one presented here, it is necessary to start from a good first approximation [47].

Table 4.3: Information about unknown parameters as provided by the data generation team.

Parameter	Nominal value	Allowed region
G_{df}	1	$[0.5,\ 1.5]$
G_{sus}	1	$[0.5,\ 1.5]$
ω_1^2	-1.3×10^{-6}	$[-3 \times 10^{-6},\ -1 \times 10^{-6}]$
ω_2^2	-2×10^{-6}	$[-3 \times 10^{-6},\ -1 \times 10^{-6}]$
δ	1×10^{-4}	$[-1.75 \times 10^{-4},\ 1.75 \times 10^{-4}]$

4.6.4 The method of parameter estimation: Singular Value Decomposition

There exists several methods for solving a set of linear equations. For the work presented here the method of *Singular Value Decomposition* (SVD) has been chosen since it is numerically very stable. SVD is described in detail in standard textbooks dealing with Linear Algebra. For completeness the main concept of the method will be given here. After that the application of the method to the model of MDC2 will be demonstrated to give an insight to the accomplished parameter estimation. The method of SVD is based on a theorem of Linear Algebra that will be given in the following without proof. It states that every $N \times M$ matrix $\mathbf{A}$, with $N > M$ can be decomposed such that:

$$\mathbf{A} = \mathbf{V} \cdot \mathrm{diag}(\mathrm{w_j}) \cdot \mathbf{U}^{\mathrm{T}}, \tag{4.66}$$

with:
$\mathbf{U}:\quad N \times M -$ matrix (column-orthogonal)
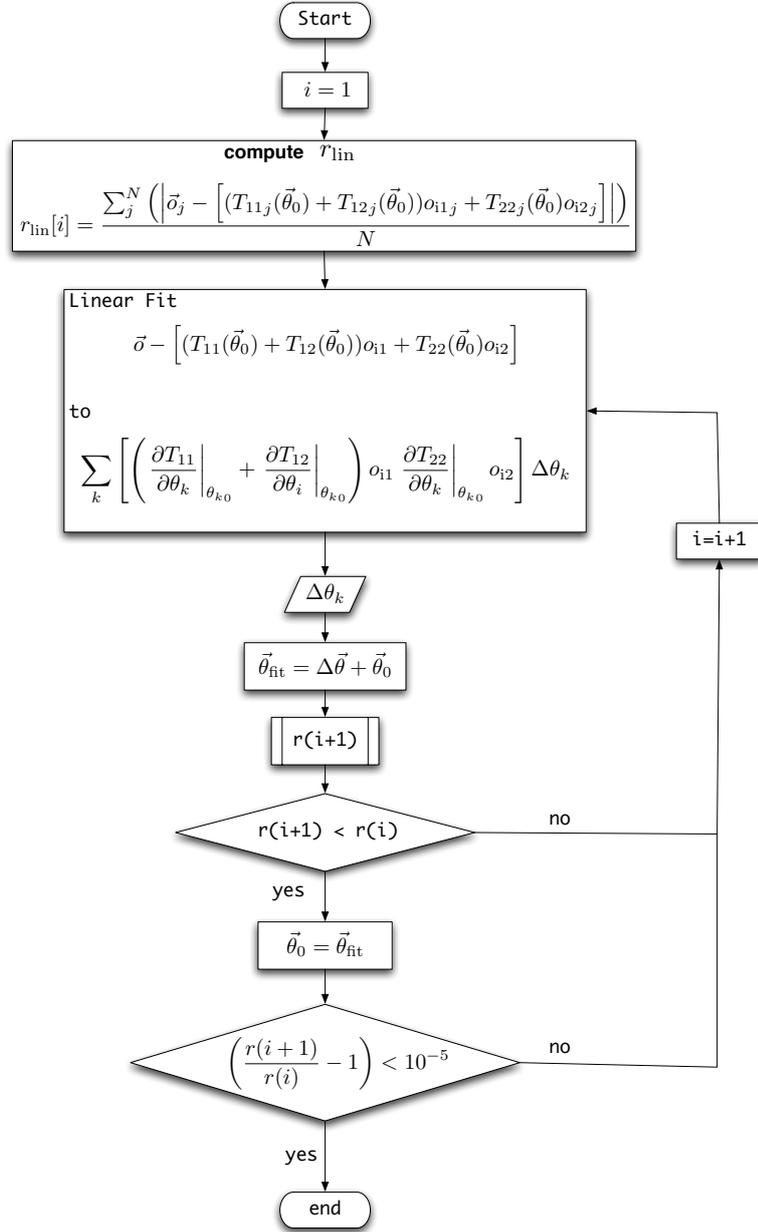
Figure 4.15: Flow chart of the iterative linear fitting routine. The end condition of the
for-loop is that the change in the residual, r, is less than 10^{-5}. The number
is arbitrary but has proven to indicate a good estimation of the parameter
values.

$\mathbf{V}:$ $\quad N \times N -$ matrix (orthogonal)

w_j: singular values.

The procedure of how this decomposition is established numerically can be found for example in [28]. For solving a set of linear equations it is necessary to compute the inverse of $\mathbf{A}$. From the SVD of $\mathbf{A}$ in Equation 4.66 the inverse can be computed as

$$\mathbf{A}^{-1} = \mathbf{V} \cdot diag(1/w_j) \cdot \mathbf{U}^T. \tag{4.67}$$

Such that the above equations lead to the solution of a set of linear equations by solving:

$$\mathbf{A}\vec{x} = \vec{b} \rightarrow \vec{x} = \mathbf{A}^{-1}\vec{b} = \mathbf{V} \cdot diag(1/w_j) \cdot \mathbf{U}^T \cdot \vec{b}. \tag{4.68}$$

The task to be solved within this chapter of parameter estimation, however, represents an overdetermined system, where there exists more equations than variables. Still SVD gives the optimal solution for x using Equation 4.68. The resulting parameter set $\vec{x}$ minimises the Equation

$$r_{\text{lin}}^2 = |\mathbf{A}\vec{x} - \vec{b}|^2. \tag{4.69}$$

For an intuitive description of the application of this model to the system of MDC2, the formalism shall be moved to matrix notation.

The Equation to be solved by the fitting routine is:

$$\Delta\vec{\theta}_{\text{opt}} = \underset{\Delta\vec{\theta}}{\operatorname{argmin}}(r_{\text{lin}}(\vec{x}, \vec{y}, \vec{\theta})), \tag{4.70}$$

where r_{lin} was defined in Equation 4.63. Defining the *Design matrix*, $\mathbf{A}$, the measurement vector, $\vec{b}$, and the parameter vector, $\Delta\vec{\theta}$

$$\mathbf{A} = \{A_{ik}\}, \text{ where } A_{ik} = \frac{\partial f}{\partial \theta_k}(\vec{\theta}_0, \vec{x}_i) \quad (N \times M) - \text{Matrix},$$

$$\vec{b} = \begin{bmatrix} y_1 - f_1(\vec{\theta}_0) \\ \vdots \\ y_N - f_N(\vec{\theta}_0) \end{bmatrix}, \quad \Delta\vec{\theta} = \begin{bmatrix} \Delta\theta_1 \\ \vdots \\ \Delta\theta_M \end{bmatrix} \tag{4.71}$$

r_{lin} can be rewritten as:

$$r_{\text{lin}}(\vec{x}, \vec{y}, \Delta\vec{\theta}) = |\mathbf{A} \cdot \Delta\vec{\theta} - \vec{b}|^2. \tag{4.72}$$

And hence the optimal parameter vector reads

$$\Delta\vec{\theta}_{\text{opt}} = \underset{\Delta\vec{\theta}}{\operatorname{argmin}} \left(|\mathbf{A} \cdot \Delta\vec{\theta} - \vec{b}|^2 \right). \tag{4.73}$$

Finally the above system of equations can be solved using the SVD of $\mathbf{A}$:

$$\Delta\vec{\theta}_{\text{opt}} = \mathbf{V} \cdot diag(1/w_j) \cdot \mathbf{U}^T \cdot \vec{b}. \tag{4.74}$$

Additionally the errors of the above found solutions need to be computed. The formalism is as follows:

$$f = \sum_k^M \Delta\vec{\theta}_{\text{opt}_k} \cdot \mathbf{A}_{ik} \tag{4.75}$$

$$S = (\mathbf{A}^T\mathbf{A})^{-1} \cdot \sigma \tag{4.76}$$

$$\text{std} = \sqrt{\text{diag(S)}}, \tag{4.77}$$

where σ is known as the variance, S is named covariance and std is the standard deviation. This error analysis is obligatory for every parameter estimation presented in this thesis. The standard deviations are given for every estimated parameter in the tables of Section 4.7. The covariance matrices are used in Section 4.8 to combine the obtained results.

There are a variety of methods with which the the linear system of equations can be solved. SVD has been chosen since it is numerically very robust and efficient.

4.6.5 Noise whitening

The interferometer data, which will be measured during the LISA Pathfinder mission, as well as the here investigated simulated data, contain a so-called 'coloured' noise. The formalism of the Linear-least squares method as it is presented here, only applies for white noise. One possible method to estimate parameters from the simulated (or measured) data is to whiten the noise in the data before applying the algorithm (another one would be to weight the data). Figure 4.16 illustrates the necessity of whitening the noise on the data: the residual function according to Equation 4.59 from Section 4.6.3 is computed for a set of approximations of $\vec{\theta}$ as described in the previous sections. The reason for the residual function not having its minimum at the correct parameter value is the coloured noise on the data. A fitting routine applied to this data with the parameter dependent noise term would return these false parameter values for which the residual function is minimised. The plot in Figure 4.16(a) shows the residual function of the data from Experiment 1 as it was generated by the simulation. Figure 4.16(b) comprises the same plots for the same experiment but with whitened noise. Those plots show that the residual functions reach their minimum at exactly the true parameter, where $\Delta\vec{\theta} = \vec{\theta} - \vec{\theta}_0 = 0$.

Since the two measurement channels o_1 and o_{12} are strongly correlated, the noise included in the two channels is correlated as well. As mentioned above, the LTPDA function `whiten2D` is used for for whitening two correlated signals.

The data whose noise has been whitened will in the following be denoted by $\vec{o}_w$. The so-called *whitening filters* applied to the data, $\vec{o}$, will be represented by $\mathcal{F}$. The following equations describe how the formalism of the simulated data $\vec{o}$ given in Equation 4.57 changes for $\vec{o}_w$.
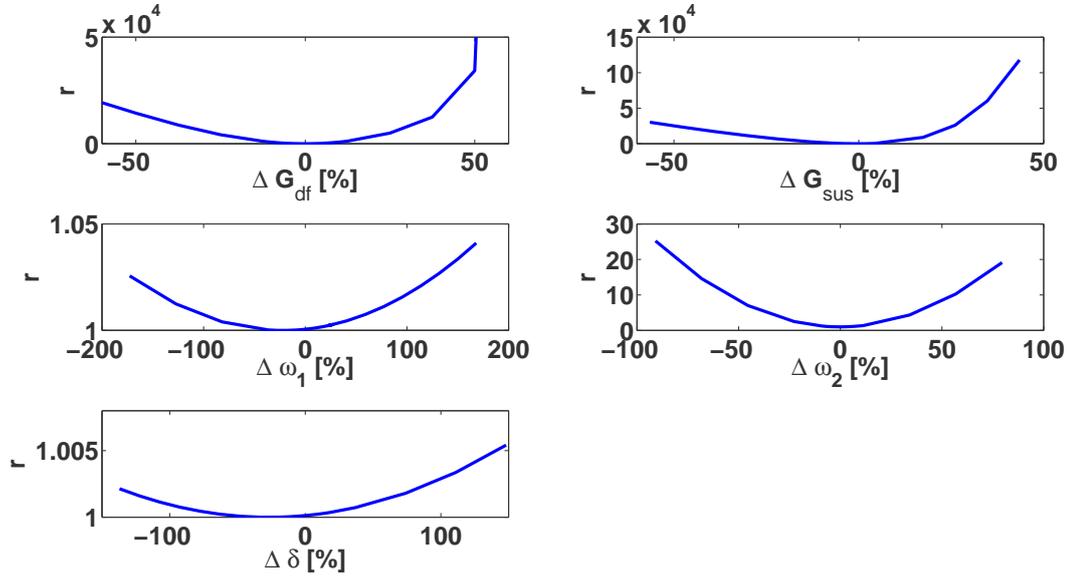
(a) Residual functions for the data with coloured noise. The graphs for $\Delta\omega_1$ and $\Delta\delta$ do not have their minimum at zero, which would represent the true parameter value. The result of a least squares fit would be the value of $\Delta\vec{\theta}$ for which the function r reaches its minimum and hence would be wrong.
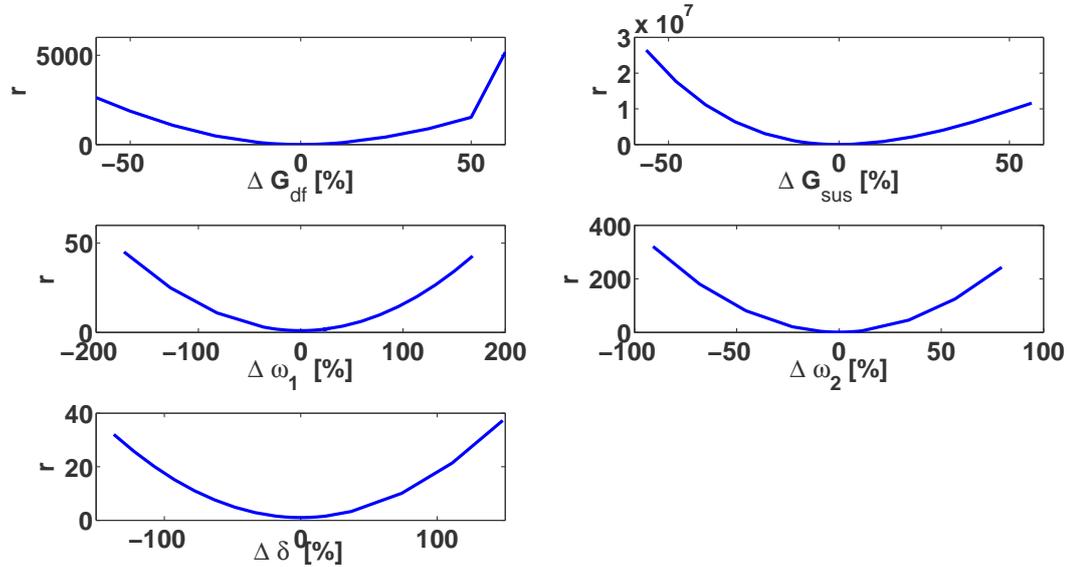
(b) Residual functions for the data with whitened noise. All minima lie at the correct value, which is zero, the difference between the correct value and the estimation. Hence with whitened data it is possible to obtain the correct result from a least squares fit.

Figure 4.16: Residual functions for all parameters in Experiment 1. All functions are normalised at their minimum. The x-axes give the discrepancy between the estimated parameter value and the true value in percent. The minimum does not lie at the correct value $\vec{\theta} - \vec{\theta}_0 = 0$ for all parameters if the noise in the data is not white (a), but they do if the noise is whitened (b).

$$
\begin{bmatrix} o_{w_1} \\ o_{w_{12}} \end{bmatrix} = \begin{bmatrix} \mathcal{F}_{11} & \mathcal{F}_{12} \\ \mathcal{F}_{21} & \mathcal{F}_{22} \end{bmatrix} \begin{bmatrix} o_1 \\ o_{12} \end{bmatrix} \tag{4.78}
$$

$$
= \begin{bmatrix} \mathcal{F}_{11} & \mathcal{F}_{12} \\ \mathcal{F}_{21} & \mathcal{F}_{22} \end{bmatrix} \begin{bmatrix} T_{11} & 0 \\ T_{12} & T_{22} \end{bmatrix} \begin{bmatrix} o_{i1} \\ o_{i2} \end{bmatrix} \tag{4.79}
$$

$$
= \begin{bmatrix} \mathcal{F}_{11}T_{11} + \mathcal{F}_{12}T_{12} & \mathcal{F}_{12}T_{22} \\ \mathcal{F}_{21}T_{11} + \mathcal{F}_{22}T_{12} & \mathcal{F}_{22}T_{22} \end{bmatrix} \begin{bmatrix} o_{i1} \\ o_{i2} \end{bmatrix} \tag{4.80}
$$

$$
= \begin{bmatrix} \mathcal{W}_{11} & \mathcal{W}_{12} \\ \mathcal{W}_{21} & \mathcal{W}_{22} \end{bmatrix} \begin{bmatrix} o_{i1} \\ o_{i2} \end{bmatrix} \tag{4.81}
$$

The input to the function is an approximated spectrum of noise. The output are the whitened data and the four whitening filters, $\mathcal{F}$, that have been applied to the data. These filters then need to be applied to the whole model in order to remove artefacts introduced by applying the filters to the simulated data.

In order to estimate the unknown parameters with the linear fitting routine from the whitened data $\vec{o}_w$, the formalism derived in Equation 4.65 needs to be adapted to the relationship described above. For the linearised model Equation 4.65 becomes:

$$
(o_{w_1} + o_{w_{12}}) - [\mathcal{W}_{11}(\vec{\theta}_0) + \mathcal{W}_{21}(\vec{\theta}_0)]o_{i1} + [\mathcal{W}_{12}(\vec{\theta}_0) + \mathcal{W}_{22}(\vec{\theta}_0)]o_{i2}
$$
$$
\approx \sum_k \left[\left(\left.\frac{\partial \mathcal{W}_{11}}{\partial \theta_k}\right|_{\theta_{k0}} + \left.\frac{\partial \mathcal{W}_{21}}{\partial \theta_k}\right|_{\theta_{k0}} \right) o_{i1} + \left(\left.\frac{\partial \mathcal{W}_{12}}{\partial \theta_k}\right|_{\theta_{k0}} \left.\frac{\partial \mathcal{W}_{22}}{\partial \theta_k}\right|_{\theta_{k0}} \right) o_{i2} \right] \Delta\theta_k. \tag{4.82}
$$

The above equation illustrates the fact that the linear model gets more complex for the case of whitened data.

4.7 Analysis of simulated signals

In this section results from the data analysis for the three experiments are presented. As mentioned previously, investigations are first made on data with white noise, before attempting to whiten the noise in the simulated data with the method described above. Investigations comprise the computation of the linearised model as well as the original, non-linear model function derived in Section 4.6.3 around the correct parameter value. The linear residual function, r_{lin}, and the non-linear function r are plotted for each experiment and the original simulated data which includes coloured noise. The information gained from both functions are useful during the process of the parameter estimation. In the following a description of the information gained from each function is given.

Non-linear residual function: By investigating the plot of these functions it can be assured that it is possible to extract the correct parameter value from any fitting routine. If the noise in the data is perfectly whitened, the minimum of the function lies at the correct parameter value. Else common fitting routines will return a false value, namely the one of the minimum found.

Linearised residual function: This function depicts how well the non-linear model function is approximated by the first term of its Taylor expansion. These

plots are particularly useful for approximating the maximal amount of the difference between the first estimation and the correct value for the linear fit that still converges to an unbiased result.

The results of the parameter estimations are then presented only for the original simulated data, whose noise has been whitened. The parameter values obtained are given for each experiment separately and a combination of those results is given in the last part of this section.

As it can be seen from the following descriptions of each experiment, it makes sense to start with the investigation of Experiment 1, followed by Experiment 3, which does not add information to the first experiment but can be regarded as *redundant*. The discussion will close with Experiment 2, being a modification of Experiment 3.

It shall be noted that all data analysed in the following section have been *zero padded*. This means that before the first stimulus signal and after the last one, zeros were added for about 200 seconds of data points. This procedure helps to suppress artefacts when computing the FFT.

4.7.1 Experiment 1

A signal, o_{i1}, is injected into the drag-free loop and a signal, o_{i2}, is injected into the electrostatic suspension loop simultaneously. Thus, a longitudinal jitter is applied to the micro-Newton thruster and to the second test mass via the servo. Only in this experiment is a signal injected into the electro-static suspension loop. A schematic of the setup has already been shown in Figure 4.12. The model for the simulated interferometer data resulting from this experiment can be derived from Equation 4.57. As mentioned above, this experiment represents a special case where two signals are injected at the same time. Hence, the output, o_{12}, is a combination of the two input signals, o_{i1} and o_{i2}. The two input signals are described by the parameters given in Table 4.4.

<div align="center">

Table 4.4: Input signals for Experiment 1

Amplitude [μm]	Frequency[Hz]	Duration [s]
1	0.01	1000
0.6	0.02	1000
0.8	0.06	1000
1	0.1	1000

Amplitude [μm]	Frequency[Hz]	Duration [s]
1	2×10^{-3}	2500
0.5	5×10^{-3}	1000
1	1×10^{-2}	1000

</div>

As described earlier in this section, the frequencies of the input signal were chosen

to stimulate the system at the frequencies of interest when measuring the transfer functions. These frequencies were depicted as well in Figure 4.14.

Like already mentioned, the analysis is shown first on a data set with white noise added instead of the more realistic coloured noise term, $\vec{n}$, from Equation 4.55. Figure 4.17 shows this simulated interferometer data for Experiment 1 together with the injected signals, $\vec{o}_i$, defined in Table 4.4.
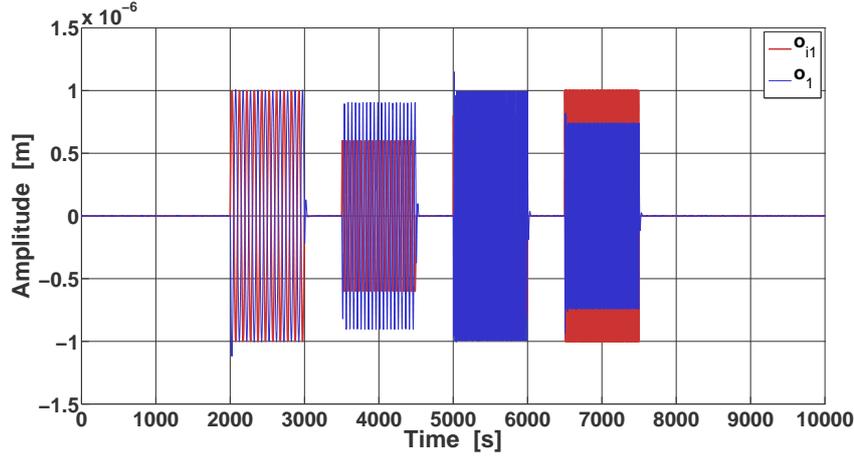

(a) o_1 together with the input signal o_{i1}
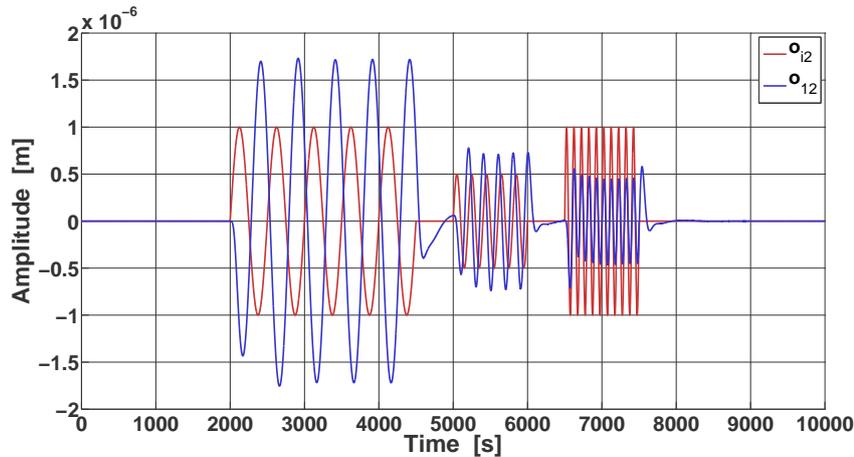
(b) o_{12} together with the input signal o_{i2}

Figure 4.17: Input and output signals of the two interferometer channels for Experiment 1. The data includes different modulation frequencies and an underlying coloured noise.

As explained in Section 4.6.3 the analysis of the data comprises not only the linear parameter estimation but also the investigation of the non-linear residual function which should have a minimum at the correct parameter value to be found by the fitting routine applied. Only if the minimum lies at the correct parameter value is it possible to estimate it precisely. The residual functions will be plotted for all parameters in $\vec{\theta}$ for data with white noise (Figure 4.18(a)). Furthermore the linear function defined in Equation 4.63 is plotted for a set of estimations of parameters

$\vec{\theta}_0$ in Figure 4.18(b). These plots depict in which region of $\Delta\vec{\theta}$ the non-linear model can be approximated by the first term of the Taylor expansion.

For Experiment 1 the expression for the data sets, o_1 and o_{12}, can be read from Equation 4.56 and the linearisation of the model representing the function to which the data is fitted, is given by Equation 4.65. Figure 4.18(a) shows that the minima of the residual functions of each parameter lie at the correct value. This means that it is possible to find these values by an appropriate fitting routine, if the noise is sufficiently white. Furthermore, investigating Figure 4.18(b), the linear residual functions suggest that the model function defined by Equation 4.65 is linear in the parameter, δ, for a large range of at least 1.5 times the correct parameter value. The constant function in the plot showing the dependency of the linear residual function on δ, illustrates that the linearised model is applicable for this parameter range. The same is true for the rest of the parameter set, but the region for the discrepancy of the respective parameter is considerably smaller.

In Table 4.5 the results of the linear parameter estimation done in the time-domain of Experiment 1 with whitened noise are listed. The first approximations were taken from Table 4.3 which was provided together with the simulated data from the data generation team. The presented results were obtained after about ten iterations and are completely reproducible. The number of iterations necessary for the fit to converge depends on how far the starting value is away from the correct one. The larger the difference between the starting value and the correct result, the more iterations are necessary.

The results from the linear fitting routine agree very well with the correct values that were used when the data was generated. The discrepancy between the correct value and the fit result are merely of the order of a few percent. The standard deviations, derived from the covariance matrix, which was defined in Equation 4.77 in Section 4.6.4 enclose these percentages very well. Only for δ the standard deviation is slightly smaller than the error computed for the result.

It should be emphasised here that the starting values for some parameters deviate by about 200 % from the ones to be found by the fitting routine (cp Table 4.5) and nevertheless the linear fit converges to almost the exact values for all five parameters. The chosen starting values are not at all presenting the limits of the range where the routine converges to the correct result. Finding the exact limits would be a very complex procedure, since the merit function depends differently on each of the five parameters. Thus, the cost-value ratio of finding the exact limits is rather low and instead a trial and error method was applied. Thus the presented starting values are not necessarily the maximum allowable.

4.7.2 Experiment 3

The setup for Experiment 3 differs from the previous one only by the absence of the second input signal. Only a signal, o_{i1}, is injected into the drag-free loop, producing a longitudinal jitter on the micro-Newton thruster. From this experiment a good measurement of the differential stiffness, $\Delta\omega$, is expected. In principle this experiment is redundant with respect to the first one, but the results obtained can be combined and hence improve accuracy. The combination of results is done in
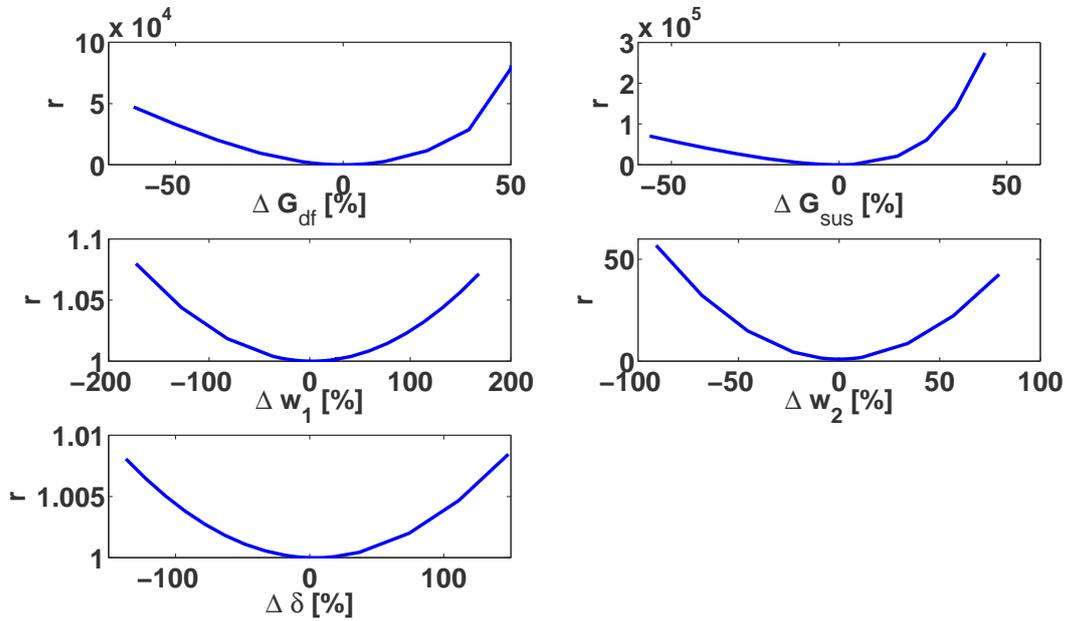
(a) Non-linear residual function
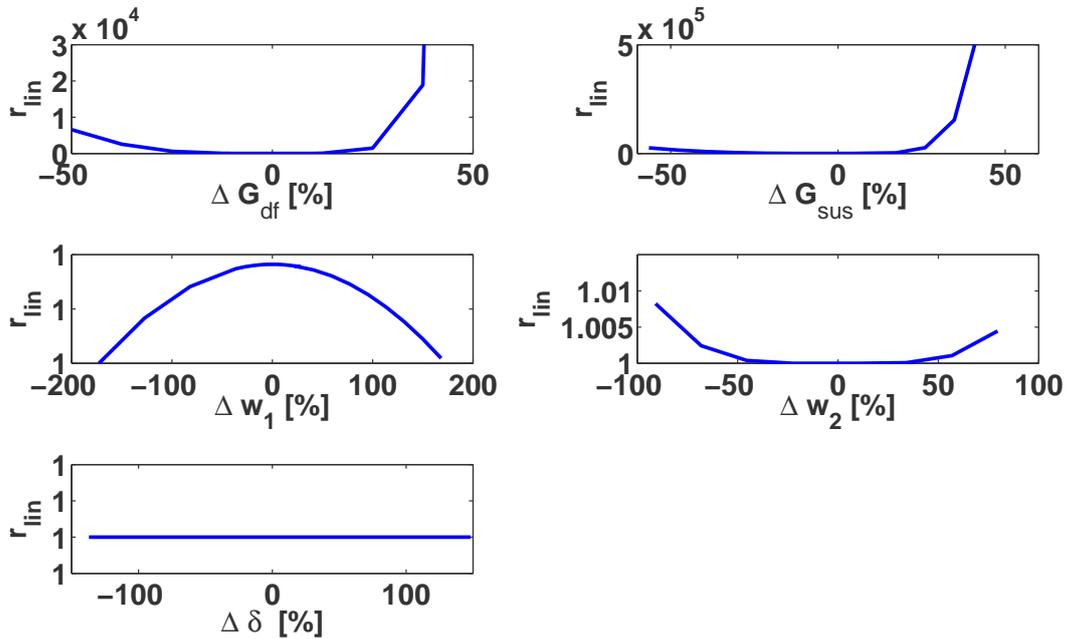
(b) Linear residual function

Figure 4.18: Experiment 1: The non-linear residual functions in (a) are minimal for the true parameter value and the linear residual functions in (b) depicts the region of parameter values for which the linearisation is a good approximation. All functions are normalised at their minimum. The x-axes give the discrepancy between the estimated parameter value and the true value in percent.

Table 4.5: Linear fitting results for Experiment 1 with whitened noise.

| Parameter	correct value	1st estimation	fit result	standard error
G_{df}	0.8	1.5	0.80014	4×10^{-5}
G_{sus}	1.15	0.5	1.1499991	6×10^{-7}
ω_1^2	-1.1×10^{-6}	-3×10^{-6}	-1.098×10^{-6}	1×10^{-9}
ω_2^2	-2.2×10^{-6}	-1×10^{-6}	-2.2002×10^{-6}	4×10^{-10}
δ	1.35×10^{-4}	-1.75×10^{-4}	1.354×10^{-4}	1×10^{-7}

Section 4.8.
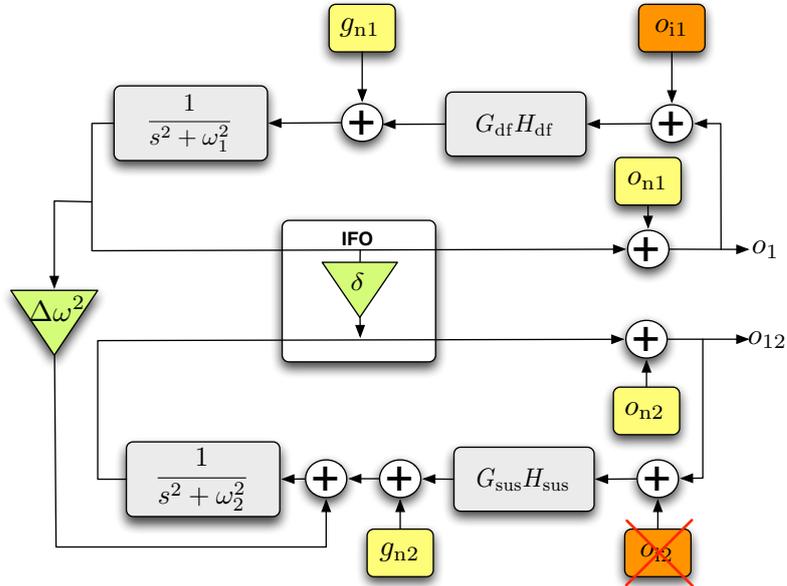
Figure 4.19: Schematic of Experiment 3, which is identical to the one of Experiment 1 shown in Figure 4.12 but without the second input signal o_{i2}. The two loops are again coupled via $\Delta\omega$ and δ

.

The model for the interferometer output data has been derived from Equation 4.57 as usual. The input signal designed for this experiment is described by the parameters given in Table 4.6.

The data with white noise is shown in Figure 4.20. The plot on the left hand side also includes the input signal o_{i1}.

For Experiment 3 the linearised model functions are derived from Equation 4.65, by setting o_{i2} to zero.

The the non-linear residuals for this experiment are shown in Figure 4.21(a) and the linear residuals in Figure 4.21(b). The relations between the different residuals of this experiment are not the same as in Experiment 1 but also no significant difference is visible. This suggests that the choice of starting values that resulted in good parameter estimates for Experiment 1 should as well be appropriate for

Table 4.6: Input signals for Experiment 3

Amplitude [μm]	Frequency [Hz]	Duration [s]
1	0.002	2500
1	0.006	1700
1	0.01	1000
0.5	0.03	330
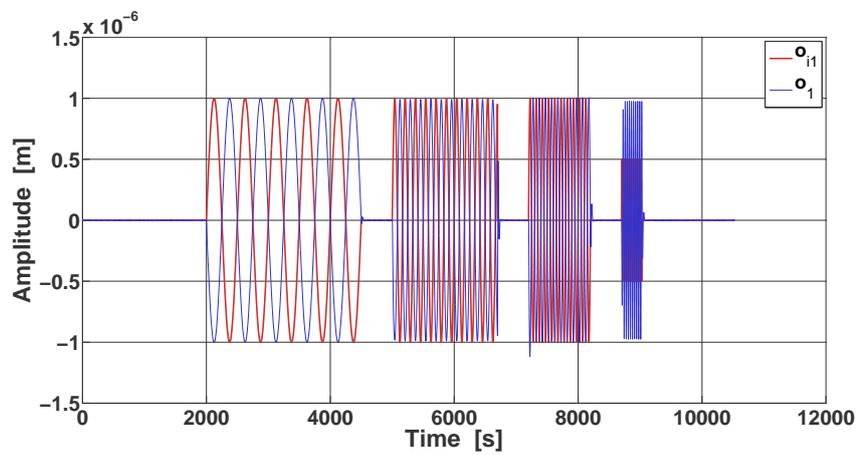
(a) o_1 together with the input signal, o_{i1}
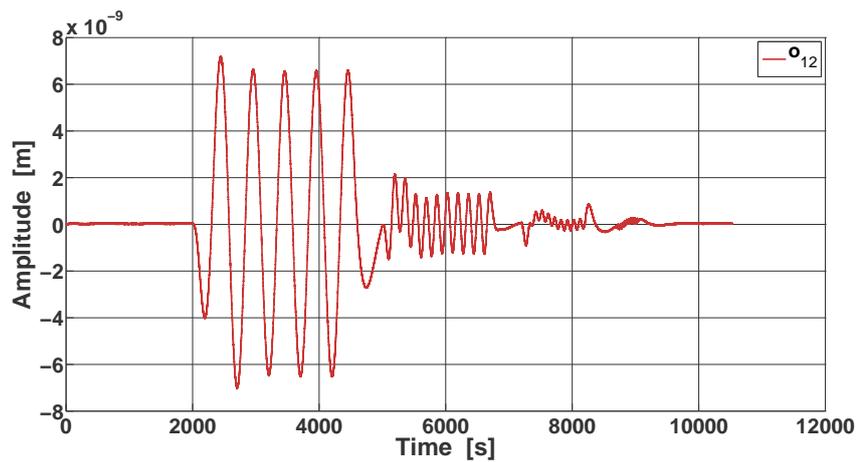
(b) o_{12}

Figure 4.20: Original data with underlying coloured noise measured for Experiment 3 when injecting the signal shown in (a).
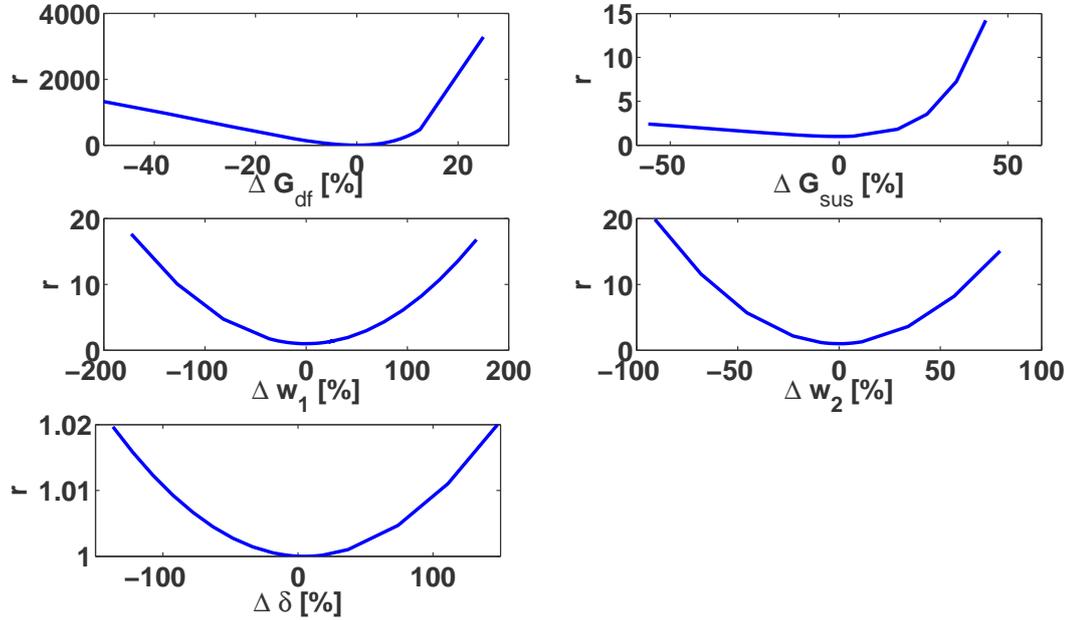
(a) Non-linear residual function
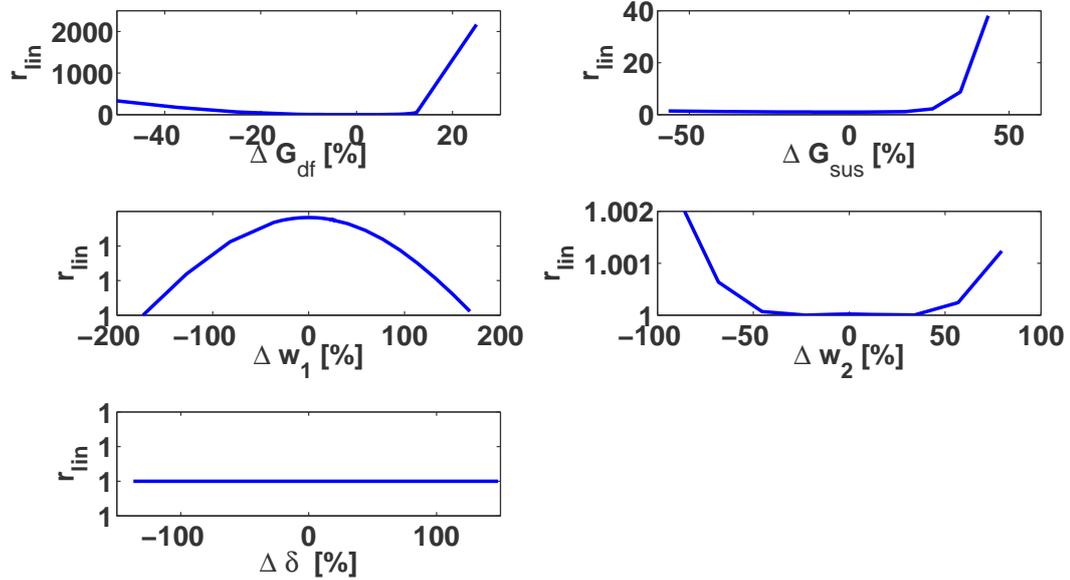
(b) Linear residual function

Figure 4.21: Residual functions for Experiment 3 plotted for all parameters separately. The non-linear residual functions in (a) are minimal for the true parameter value and the linear residual functions in (b) depict the region of parameter values for which the linearisation is a good approximation. All functions are normalised at their minimum. The x-axes give the discrepancy between the estimated parameter value and the true value in percent.

Experiment 3. As such the first estimations are chosen to be the same as in Experiment 1. And as can be seen from the results for the whitened data presented in Table 4.7 the method converged to the correct values for all parameters.

Returning to the statement made above, that this experiment is expected to be very suitable for measuring $\Delta\omega = \omega_2^2 - \omega_1^2$: the results for the two stiffness parameters are indeed not exact (but they lie within their computed standard deviations), whereas the result for their difference is exact. This is in agreement with other results not shown here and is expected from analysis presented in [48].

All deviations of the obtained results lie within the region of the computed standard deviation.

Table 4.7: Linear fitting results for Experiment 3 with whitened noise.

Parameter	correct value	1st estimation	fit result	standard error
G_{df}	0.8	1.5	0.8001	3×10^{-4}
G_{sus}	1.15	0.5	1.1502	5×10^{-4}
ω_1^2	-1.1×10^{-6}	-3×10^{-6}	-1.3×10^{-6}	2×10^{-7}
ω_2^2	-2.2×10^{-6}	-1×10^{-6}	-2.4×10^{-6}	2×10^{-7}
δ	1.35×10^{-4}	-1.75×10^{-4}	1.348×10^{-4}	2×10^{-7}

4.7.3 Experiment 2

A signal, o_{i1}, is injected into the drag-free loop, where the difference of the two stiffness terms $\Delta\omega = \omega_2^2 - \omega_1^2$ is set to zero. It follows that the two control loops are coupled only by the interferometer cross-talk, δ, (see Figure 4.22). Thus, it is expected that this experiment is very suitable for measuring δ.

This experiment, where the stiffness of both test masses is equal, are of special importance for LTP. Without the interferometer cross-talk, δ, the response of the second interferometer channel, o_{12}, would be suppressed by matching the stiffness. In reality matching the stiffness will be realised by applying a signal to the electrostatic suspension of TM 1, which increases ω_1 and can be adjusted such that $\omega_1 = \omega_2 = \omega_{\mathrm{match}}$ corresponding to $\Delta\omega = 0$. For the simulation discussed here, however, such a signal is not considered. Instead $\Delta\omega$ is simply set to zero in the data generation algorithms. Figure 4.22 illustrates the signal flow for Experiment 2.

Deriving the model for the interferometer data resulting from this experiment and inserting ω_{match} gives, according to Equation 4.57, the model for the two interferometer channels. Deriving the Equations is again straightforward and shall not be repeated here.

The frequencies are chosen to probe the transfer functions at its most characteristic attributes (see Figure 4.14). Figure 4.23 shows the data to be analysed. The plots show o_1 and o_{12} with the noise term $\vec{n}$ being the original data with underlying coloured noise.

Also for Experiment 2 the target and model function for the linear fitting routine can be read from Equation 4.65. In this case only one signal, o_{i1}, is injected into the system such that $o_{\mathrm{i2}} = 0$. Considering furthermore $\omega_{\mathrm{match}} = \omega_1 = \omega_2$, Equation
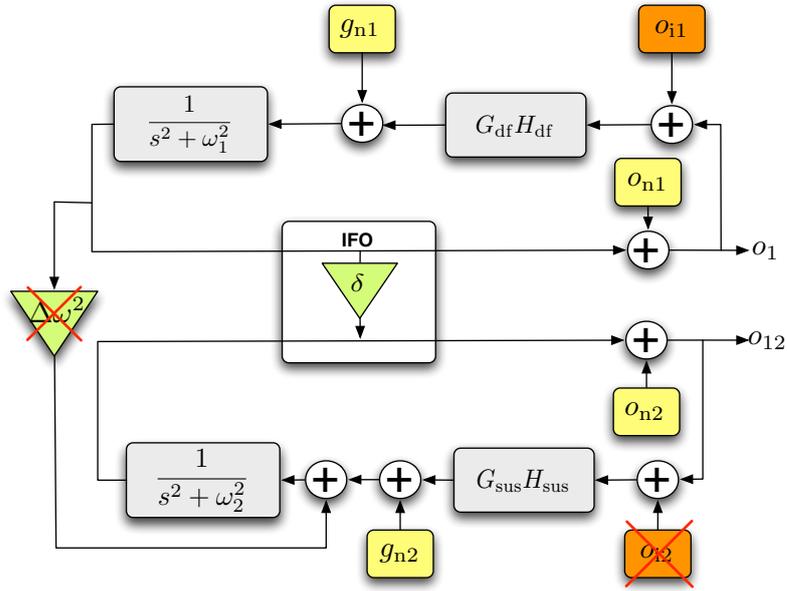
Figure 4.22: Schematic of Experiment 2. A signal, o_{i1}, is injected at the drag-free controller, into the drag-free loop. $\Delta\omega$ is set to zero such that the two loops are only coupled via the interferometer cross coupling δ.

Table 4.8: Input signals for Experiment 2

| Amplitude [μm]	Frequency [Hz]	Duration [s]
1	0.002	2500
1	0.01	1500
1	0.03	1700
0.5	0.1	330

4.65, yields the expression that enters the linear fit. Giving these Equations would only be a repetition, but the derivatives involved can be found in Figure 4.2.

For Experiment 2 several different sets of starting values have been tried out as input to the linear fitting routine. Not all results will be given here in detail but a short summary of the investigations made shall be listed:

- The fit does not converge to any sensible answer if the starting guesses are taken to be the same as for Experiment 1.

- The linear fit converges to the true answer if the first estimations are not more than a few percent away.

- The fit converges to the correct results, if ω_{match} is kept fixed and excluded from the process of fitting. With this method the same starting values as in the other two experiments can be chosen. The results from this method are the ones chosen to be presented in this thesis. They are given in Table 4.9.

- An obvious experiment to be performed is to simply ignore the particularity
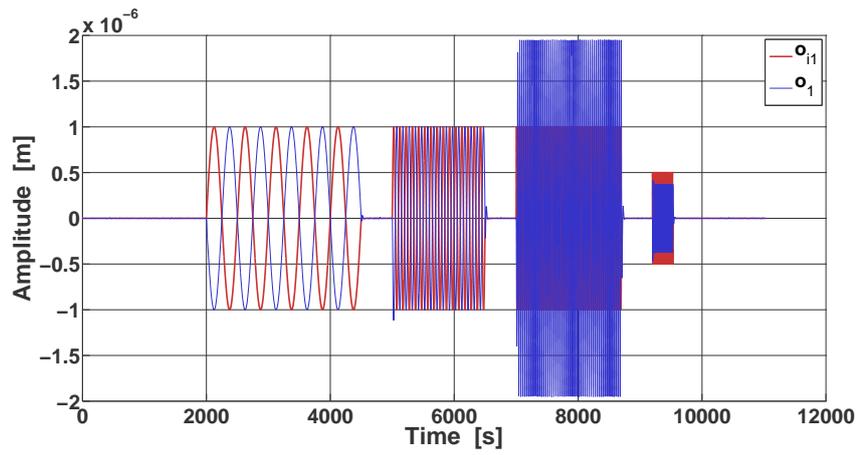
(a) o_1 together with input signal o_{i1}.
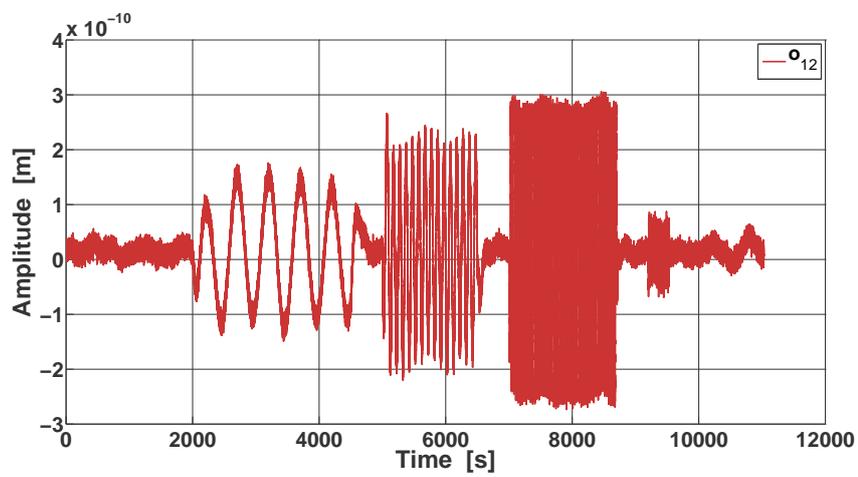
(b) o_{12}

Figure 4.23: Original data with underlying coloured noise for Experiment 2
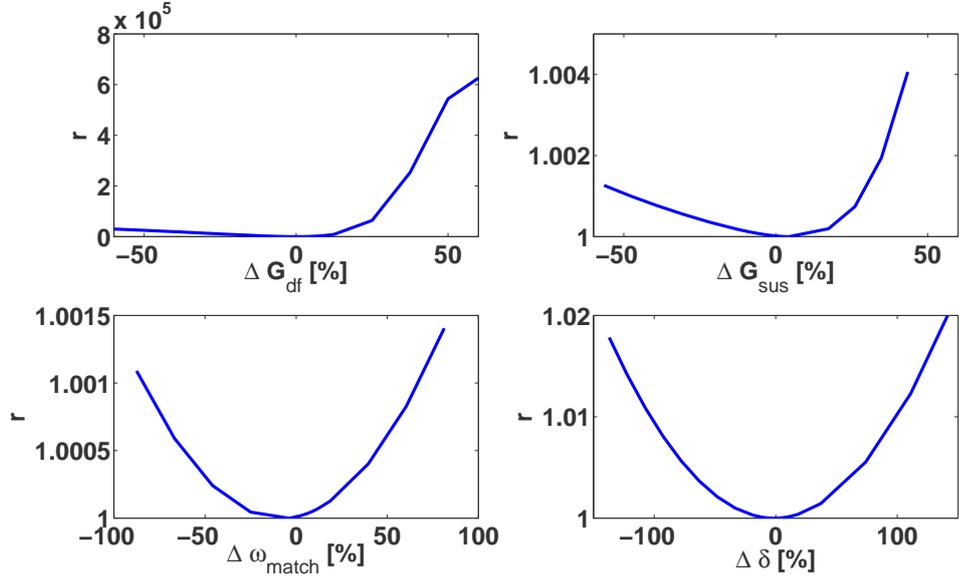
(a) Non-linear residual function
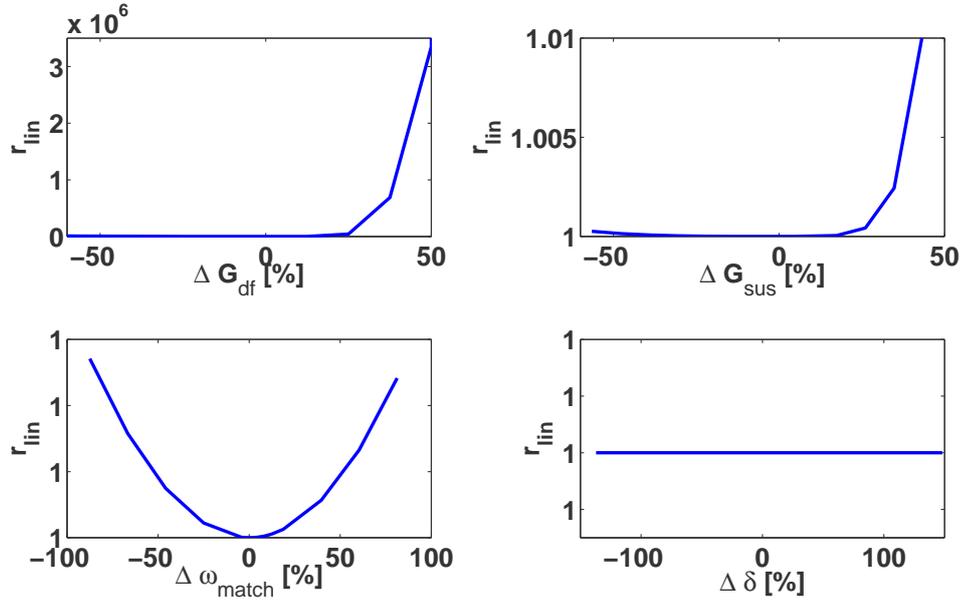
(b) Linear residual function

Figure 4.24: Residual functions for Experiment 2 plotted for all parameters separately. The non-linear residual functions in (a) are minimal for the true parameter value and the linear residual functions in (b) depict the region of parameter values for which the linearisation is a good approximation. All functions are normalised at their minimum. The x-axes give the discrepancy between the estimated parameter value and the true value in percent.

of this experiment and naively perform the same fit as in Experiment 3: fitting the two stiffness terms, ω_1 and ω_2 separately. Indeed in this case the fit converges to the correct parameter values for the controller gains, G_{df}, G_{sus}, and the IFO cross-coupling, δ with same starting values as for the other experiments. For the two stiffness terms the resulting estimation will indeed differ more than 100% from the correct value, but the difference between them, Δ_ω, which was set zero is estimated correctly. The values will not be given here explicitly but this result is in agreement with observations made in Experiment 3 (see Section 4.7.2) and with other fitting routines (see for example [48]).

In investigating Figure 4.24(b), no crucial differences from the residual functions from other experiments can be found. However, the new parameter ω_{match} depicts the most obvious difference and its linear residual function also differs from the ones of the other two experiments for ω_1 and ω_2. This is the reason for probing the method leading to the results given in Table 4.9:

Since the parameter that makes the experiment differ from the others is ω_{match}, it is the one being suspected to spoil the approximation done by the linearisation and as such is kept fixed. Not fitting ω_{match}, of course means that its value can not be measured, if it is not included in any other experiment. After all for the results presented here, the value of ω_{match} does not need to be known exactly. It has been fixed to a value of more than 100 % away from the correct one. In Table 4.9 the fixed value is stated as the first estimation and the fit result of this parameter does of course not exist.

Table 4.9: Linear fitting results for Experiment 2 with whitened noise.

Parameter	correct value	1st estimation	fit result	standard error
G_{df}	0.8	1.5	0.80005	6×10^{-5}
G_{sus}	1.15	0.5	1.146	3×10^{-3}
$\omega^2_{\mathrm{match}}$	-2.4×10^{-6}	-1×10^{-6}	fixed	X
δ	1.35×10^{-4}	-1.75×10^{-4}	1.3498×10^{-4}	7×10^{-8}

4.8 Summary and combination of results by means of the covariance matrices

To conclude this section about the linear parameter estimation for the second mock data challenge, the experiments carried out are listed together with the parameters measured by it in Table 4.10. This Table provides a better overview when summarising the results obtained. G_{df}, G_{sus} and δ are parameters of the transfer functions of all experiments and it was possible to estimate them with every experiment. The correct value for ω_{match} however could not be estimated. The only Experiment including this parameter is Experiment 2 and there the estimation method did not

converge when including it. Instead the value for ω_{match} has been fixed to a first guess (see Section 4.7 for details) to measure the remaining parameter values.

Table 4.10: Overview of parameters measured in each experiment. 'X' indicates that the parameter is measured by the allocated experiment and '-' indicates that it is not measured by an experiment. G_{df}, G_{sus} and δ can be measured by all experiments. ω_{match} appears only in Experiment 2 and was not included in the parameter estimation routine. For the reasons see text.

	Experiment 1	Experiment 2	Experiment 3
G_{df}	X	X	X
G_{sus}	X	X	X
ω_1	X	-	X
ω_2	X	-	X
ω_{match}	-	-	-
δ	X	X	X

Finally it makes sense to combine the results of the three experiments performed, since several parameters have been measured multiple times. This is done by combining the obtained covariance matrices, $\mathbf{S}$. The definition of the covariance matrix is given by Equation 4.77 in Section 4.6.3). The combination is done as follows:

$$\mathbf{S}_{\text{c}}^{-1} = \sum_{i=1}^{N} \mathbf{S}_{i}^{-1}. \tag{4.83}$$

From the combined covariance matrix, $\mathbf{S}_{\text{c}}$, a new mean value for the parameters θ is obtained:

$$\vec{\theta}_{\text{c}} = \mathbf{S}_{\text{c}} \sum_{i=1}^{N} \mathbf{S}_{i}^{-1} \theta_{i}. \tag{4.84}$$

Since not every parameter is measured in every experiment, the covariance matrices $\mathbf{S}$ of the experiments do not all have the same size. While S_1 and S_3 are of size 5×5 (all 5 parameters are measured), S_2 is of size 3×3 (ω_1 and ω_3 are not measured). For applying Equation 4.84, however, all matrices S need to be of the same size. For this S_2 is filled in the following way: the variances of the parameter values, that are not measured are set to infinity and their covariances with the remaining parameter are set to zero. Hence, the 5×5 covariance matrix for Experiment 2 reads:

$$
\begin{matrix}
\text{std}_{\text{G}_{\text{df}}}^2 & X & 0 & 0 & X \\
X & \text{std}_{\text{G}_{\text{sus}}}^2 & 0 & 0 & X \\
X & X & \text{inf} & 0 & X \\
X & X & 0 & \text{inf} & X \\
X & X & 0 & 0 & \text{std}_{\delta}^2,
\end{matrix}
\tag{4.85}
$$

where std_x is the standard deviation of the corresponding parameter x. With this recipe for the covariance matrix of Experiment 2, Equation 4.84 can be applied to

combine the results of all parameter values measured in all three experiments. The final result is given in Table 4.11.

Table 4.11: Combined results from all experiments. Parameter value and standard deviation

Parameter	correct value	fit result	standard error
G_{df}	0.8	0.80006	3×10^{-5}
G_{sus}	1.15	1.1499991	6×10^{-7}
ω_1^2	-1.1×10^{-6}	-1.1005×10^{-6}	6×10^{-10}
ω_2^2	-2.2×10^{-6}	-2.2002×10^{-6}	5×10^{-10}
δ	1.35×10^{-4}	1.3499×10^{-4}	6×10^{-8}

It has been shown in Section 4.6.5 by Figure 4.16 that ignoring the coloured and correlated noise terms in the simulated interferometer data could be fatal for measuring parameters. Whitening the noise in the data was chosen as appropriate method to deal with this issue within the scope of this thesis. Whitening the noise in the two correlated data channels with the newly developed tool *whiten2D* and then including the applied filters correctly in the model turned out to be rather complex. This is why the analysis has been done first on data with white noise to confirm that the applied model and all derivatives are correct.

In the second step the linear data analysis including all parameter estimations are done on the more realistic data simulated by the data generation team, where coloured and parameter dependent noise terms were included. The application of the linear parameter estimation required that the noise was whitened, which was done using advanced methods implemented in LTPDA. After the model has been adapted accordingly as described in Section 4.6.5, parameters were estimated successfully by the method of Singular Value Decomposition (SVD). The results obtained are finally combined in Table 4.11. From these results we conclude that the model underlying the second Mock Data Challenge can be very well approximated by a linearisation if appropriate experiments are applied.

With Experiment 1, where two signals were injected into the two control loops independently, all parameter values are estimated accurately by a linear fitting routine. The results exceed the expectations in the sense that the first estimations are allowed to differ significantly from the true values. The same holds for Experiment 3 where the same starting values are used as used in Experiment 1. The linearisation of the transfer functions resulting from this experiment is as well a sufficient approximation.

For Experiment 2 the linear fit converges again from the same starting values as in the other experiments, if the new parameter $\omega_1 = \omega_2 = \omega_{match}$ is excluded from the fitting routine. Although it has been shown that this parameter does not need to be known very well for the fit of the rest of the parameter set to converge, this means that ω_{match} cannot be measured by the linear approach within this setup.

The investigations on the linear analysis of the rather naively developed three experiments for LTP has turned out to be very enlightening. The complex non-linear model for the dynamics of LTP can be approximated very well by its Taylor

expansion if appropriate stimulus signals are applied. The comprehensive approach of linear parameter estimation constitutes its power. In the region, where the linear approximation holds, the computation of the covariance matrices and with it, the individual standard deviations are straightforward and well understood. Hence, we have shown here that it makes sense to choose a linear fitting routine over a complex, non-intuitive, non-linear fitting routine whenever a good enough first guess of the parameter values are at hand.

4.9 Conclusion

As pointed out in the introduction to this chapter, the development of MDCs, and analysis tools in general, is closely connected to the design of the experiments.

During the evolution of this thesis the two MDCs presented in this chapter have been carried out successfully. During the process of MDC1 one of the key components of the data analysis for LISA Pathfinder, the conversion of interferometer readout to test mass acceleration has been implemented (for the ideal, one-dimensional model). The key point of MDC2 was to investigated a variety of parameter estimation methods and in this thesis the linear approach was presented. As a result the work on MDC2 drove the development and investigation of all fitting routines under investigation.

The complexity of the simulated experiments is increased stepwise, such that the MDCs will become increasingly realistic and all mission experiments can be tested using realistic MDC models. These MDCs have been proven to be a good instrument for driving the development of the analysis tools as well as forcing a good understanding of the experiment on all scientists involved. With every MDC, the model used shall become more advanced and the analysis outputs will be used as reference data during the mission.

5 Injecting non-Gaussian noise source into LISA simulations

The injection of non-Gaussian noise into LISACode is a first attempt at injecting a real noise source observed in LISA Pathfinder into a LISA Data simulator. This project is meant to serve as a catalyst for a more intense collaboration between the data analysis efforts done in LISA Pathfinder and those of LISA. The focus of the two is very different since LISA Pathfinder will be insensitive to gravitational wave signals. From the technical point of view, however, it is important to transfer the experience made in LISA Pathfinder to LISA. This includes the sideband induced noise (SIN) presented in this thesis (see also Section 3.4), which has been be observed in the LISA Pathfinder interferometer readout and which has been well studied in the course of LISA Pathfinder data analysis. SIN serves as model since expected noise from ghost beams and stray light may have similar properties.

In this chapter the LISA data generator LISACode will be introduced together with the main components which it aims to simulate. The noise sources that are already part of the simulations performed will be presented. It follows a discussion on the noise source to be injected into the data generator. After this the procedure of the computation of the LISA sensitivity curve will be explained and finally the impact of the injected noise source on the LISA sensitivity curve is presented.

5.1 The LISA data generator: LISACode

The LISA data generator, 'LISACode', is presented in [49]. The software has been written by Antoine Petiteau from APC, Paris in the programming language C++ and an executable file produced. This file can be executed with an input xml-file providing the desired parameters of the simulation to be performed. The output of LISACode is a time series of the simulated phasemeter data. The schematic in Figure 5.1 shows the individual components of the software. The box encloses the core part of the software. The gravitational waves calculations and the application of time delay interferometry (TDI) are part of the code but are not part of LISA and as such are indicated as external boxes in the schematic.

The individual items are discussed in [49]. The most important items will be presented in more detail in the following.

LISA Orbits In LISACode the computation of the orbits of the three spacecraft are based on the formulae given in [50]. The orbits include the rotation of the LISA triangle as well as the flexing of the distance between the spacecraft [49]. The light propagation time can be computed by simply using the distance between the
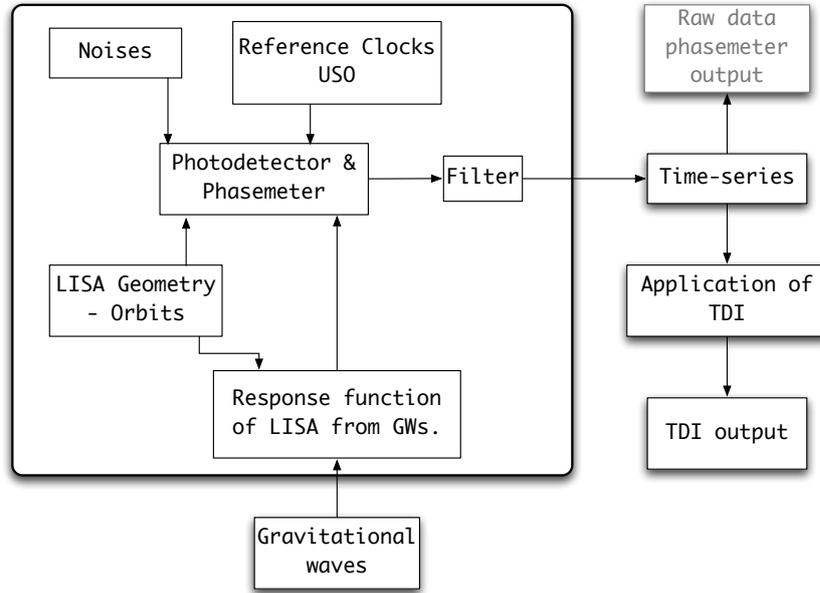
Figure 5.1: Schematic of LISACode. Shown are the individual components included in the data generation process. The core part of the software is indicated by the enclosing box. Two types of output time-series are created: one without applying TDI and one on which TDI has been applied to. In this thesis only the time-series with TDI are investigated. Source: [49].

spacecraft but can be arbitrarily complex and more precise by taking into account relativistic effects [51].

The phasemeter   LISACode simulates the phasemeter output, which is a combination of the measured gravitational wave strain and the effect of the various noises. As mentioned in Section 1.2, each satellite consists of two optical benches and two phasemeters. One phasemeter measures the phase between the incident laser beam from another LISA spacecraft and the local laser beam (far interferometry), and the other measures the relative phase of the two local laser beams on the same satellite (local interferometry). A schematic of one LISA spacecraft is shown in Figure 1.4 in Section 1.2. In LISACode the phasemeter signals are given in the unit of relative frequency fluctuation $(\delta f / f)$.

Ultra stable oscillator (USO)   This noise source has an impact on the phasemeter output as well as on the application of TDI. As reference clock the USO will provide the sampling clock and hence influence the phasemeter output. Moreover, the USO will time-stamp the time series and hence any deviation will introduce an error into the application of TDI.

Time delay interferometry (TDI)   It has been mentioned before that the LISA interferometry is divided in a local interferometry, measuring the relative phase between the two optical benches on one spacecraft, and the far interferometry mea-

suring the relative phase between two Spacecraft. This implies the application of two independent lasers, which introduces a noise of the order of 10^{-13} (in $\delta f/f$ units). The gravitational wave signal is near 10^{-21} which implies that the noise has to be suppressed by 8 orders of magnitude. This is achieved by the method of TDI. The main idea of this method is performing linear combinations of phasemeter outputs at different times and thereby cancelling the laser noise. The subject of TDI is not covered in detail in this thesis but it shall be stated here that it is fully implemented in LISACode and an extensive description is given in [7, 52, 53].

5.2 Noise sources

Noise can be divided into two groups: external noise sources and internal noise sources. External noises are transmitted from one satellite to the another therefore can be suppressed to a certain level by applying TDI [49]. Internal noises have a local impact on the phasemeter output and cannot be suppressed by TDI. Laser noises for example are external noises. A list of noise sources is given in Table 5.1. The measurement noises included in LISACode are taken from a list given in the PrePhaseAReport [6]. A set of noise sources including, for example, noise from the Ultra Stable Oscillator (USO) and the laser beam pointing instability, is summed up in optical path noise in LISACode (see Table 5.1).

Table 5.1: Noise sources included in LISACode. The PSDs of the noise sources given in the second column have been taken from Table 4.1 of [6]. The third column gives the respective values used in LISACode in the same unit as in the first column. For the 2nd to the 6th row the noises were added quadratically and averaged to be combined for the use in LISACode. Source: [49].

Type of noise source	Value of error	LISACode
Measurement noise		
Detector shot noise	11×10^{-12} [m/$\sqrt{\text{Hz}}$]	11
USO	5×10^{-12} [m/$\sqrt{\text{Hz}}$]	
Laser beam pointing stability	10×10^{-12} [m/$\sqrt{\text{Hz}}$]	15.7
Laser phase measurement and offset lock	5×10^{-12} [m/$\sqrt{\text{Hz}}$]	
Scattered light effects	5×10^{-12} [m/$\sqrt{\text{Hz}}$]	
Other substantial effects	8.5×10^{-12} [m/$\sqrt{\text{Hz}}$]	
Residual laser phase noise	5×10^{-12} [m/$\sqrt{\text{Hz}}$]	not included
Acceleration noise		
Test mass noise	3×10^{-15} [m/(s^2 $\sqrt{\text{Hz}}$)]	3

The table shows the default errors included in the simulation but they can be

changed by the user. The software takes user defined xml-files as input files. In these files the amplitude of the individual internal noise sources can be given for each of the six phasemeters individually. For the curves presented here the default values have been chosen. The laser noise, however is not included in LISACode, since it represents the residual noise term after the application of TDI and this is treated explicitly in LISACode. The test mass noise is the only acceleration noise in the list. The LISA Pathfinder mission is expected to provide a measurement on this.

5.3 Sideband induced noise (SIN)

The sideband induced noise was observed in the LISA Pathfinder experiment. The noise source has been investigated intensively during the past years. Analysis tools for estimating the noise contribution in LTP have been developed and implemented within the software tool for the data analysis of the mission, LTPDA. Once the noise contribution is estimated it is possible to subtract it from the measurement data. All of this has already been discussed in Section 3.4, where the sideband induced noise was introduced.

This section deals with the same noise source in a more general way. Since unlike in LISA Pathfinder for LISA no experimental investigations on this type of noise source exist. Also the model with which the noise is described and which the noise generation is based on is expressed in a more general, to start with. For the Engineering Model of the LTP interferometer the model for the sideband induced noise was given in Equation 3.30. For the injection into LISACode it is simplified to

$$\varepsilon = A \cdot \cos(\phi(t)), \tag{5.1}$$

where A is the amplitude of the noise source and corresponds to the combined amplitude of the optical sidebands and ϕ is the phase error measured by the interferometer. $\phi(t)$ represents a random process driven by the pathlength changes in optical fibres or in parasitic interferometers.

Even though the model is simplified compared to Equation 3.30, the important periodic non-linearity is kept by the cosine term. As discussed in Section 3.4, in the LTP interferometer the origin of the optical sidebands causing this noise contribution are electrical sidebands originating from the acousto optic modulators (AOMs) in the Laser modulator. In LISA, however, no AOMs are used but the same effect has been observed in the presents of stray light, which as well can cause optical sidebands [30].

The spectrum of $\phi(t)$ is not easy to predict. A spectrum of fibre pathlength fluctuations under laboratory conditions serves as template for the generation of noise shown in this section. The noise in orbit is expected to be lower but it is not known by how much. Following the template, the amplitude spectral density (ASD) of $\phi(t)$ is taken to follow a $1/f$ model. Again this model might not be the one measured in LISA but since not much is known about the behaviour of the measurement phase due to this error source in LISA, this model has been chosen to begin with. Also a model described by $1/f$ is true for many common noise sources. The parameters that have been changed in the scope of the work presented here

are the amplitude, A, and the gain of the transfer function describing $\phi(t)$, while the slope of the transfer function is always kept to $1/f$. Since the convention in LISACode is to use the unit of fractional frequency fluctuations, $\nu/\Delta\nu$, and so far the noise has been generated in radians, the following transformation formula needs to be applied to the noise term, ε:

$$h_\varepsilon = \varepsilon \cdot \frac{\lambda}{4\pi^2 L}, \tag{5.2}$$

where the λ is the laser wavelength, which reads 1.064×10^{-6} m and L is the interferometer armlength and is 5×10^9 m.

5.4 Generation of sideband induced noise

The generation of the noise is based on the model given in Equation 5.2. The noise generation is done in LTPDA (see Chapter 2). In the following the individual steps of the procedure are explained and for each step the intermediate results are plotted. Because of the non-linearity of the cosine-function the best way for simulating this noise is in the time-domain. The complete procedure of the noise generation is carried out in the time-domain but the intermediate results are plotted as respective amplitude spectral densities.

In principle the procedure of noise generation comprises generating a random time-series with a prescribed spectrum of $1/f$ and inserting it into the formula given in 5.1. In detail the procedure of the noise generation using LTPDA is as follows:

1. Generation of time series with the prescribed $1/f$ spectrum using Franklin's noise generator (see Section 2.5). The time-series represents $\phi(t)$ in Equation 5.1.

2. Computation of the noise term, ε according to Equation 5.1.

3. Computation of the time derivative of ε to obtain a frequency noise, which is the convention for noises in LISACode.

4. Transformation from radians to the unit of $\nu/\Delta\nu$, the convention used in LISACode. The equation used is given in 5.2.

As mentioned earlier, the parameters to be varied for the noise injection into LISACode are the amplitude, A, and the gain of the pole/zero model representing the prescribed spectrum of the generated time-series. The pole/zero model used as template for $\phi(t)$ consists of a single pole at f = 1×10^{-4} Hz. The gains, g, are set to 500, 100 and 10 respectively. Figure 5.2 shows the three curves with the different gains. Shown are the amplitude spectral densities of the generated time-series $\phi(t)$ together with the analytic response function of the respective pole/zero model.

In the second step, the computation of ε, the cosine of $\phi(t)$ is computed and it is multiplied with the amplitude, A. Figure 5.3 shows the result for the three noise curves (ASD) from Figure 5.2. For the curves shown A has been set to 1 milliradian.
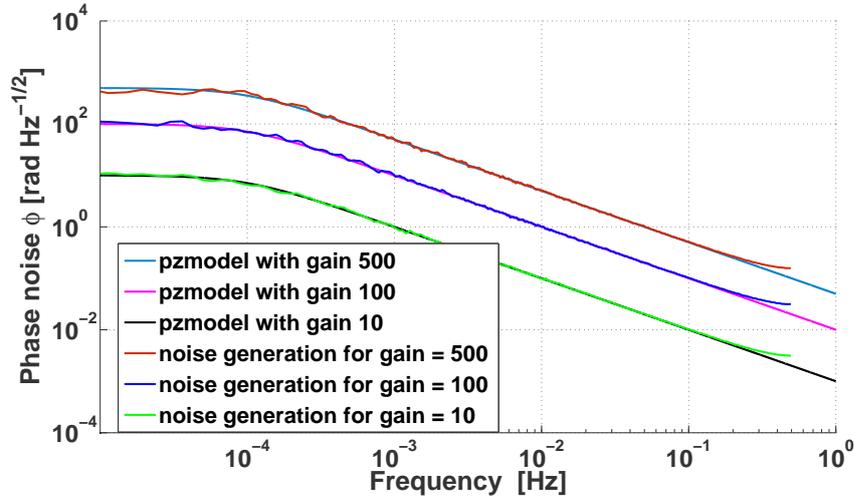
Figure 5.2: Amplitude spectral densities of phase noises $\phi(t)$. The underlying time-series have been generated using Franklin's method (2.5). For comparison the responses of the pole/zero models serving as template are also shown. The pole/zero model has a single pole at f $= 1 \times 10^{-4}$ Hz. The gains are set to 500, 100 and 10 respectively.

The plot illustrates the periodic cosine behaviour of ε: The ASD of the function with the highest gain, g $= 500$, has a lower amplitude at low frequencies than the ASD of the function with g $= 100$. But since also the corner frequency shifts to higher frequencies with increasing gain, the overall amplitude stays the highest for the function with the highest gain.

The next step is the computation of the time derivative of the noise term, ε. This is done numerically using the function `deriv` in LTPDA. This is the preparation for matching the unit of the generated noise with the convention used in LISACode. In fact the input into LISACode is simply this time derivative but internally it is then transformed into the unit of fractional frequency fluctuations, $\nu/\Delta\nu$. For easier comparison with the sensitivity curve plotted in Figure 5.7 later in Section 5.5.2 we plot here the final result used in LISACode after the transformation to $\nu/\Delta\nu$. In Figure 5.4(a) the three ASDs of the final results for the generated noise terms with $A = 1$ mrad are plotted. Figure 5.4(b) shows the result for the same procedure with $A = 5$ mrad.

The plots in Figure 5.4 show that the ASD of the noise term increases with increasing amplitude, A. They also show that with increasing amplitude the behaviour of the curves does not change significantly whereas from a high enough gain of the transfer function the behaviour of the curve changes such that the periodic behaviour of the noise term can be sensed.

5.5 The LISA sensitivity curve

The LISA sensitivity curve was already shown in the introduction to this thesis in Section 1.3. In this section we will have a closer look at this curve. The sensitivity
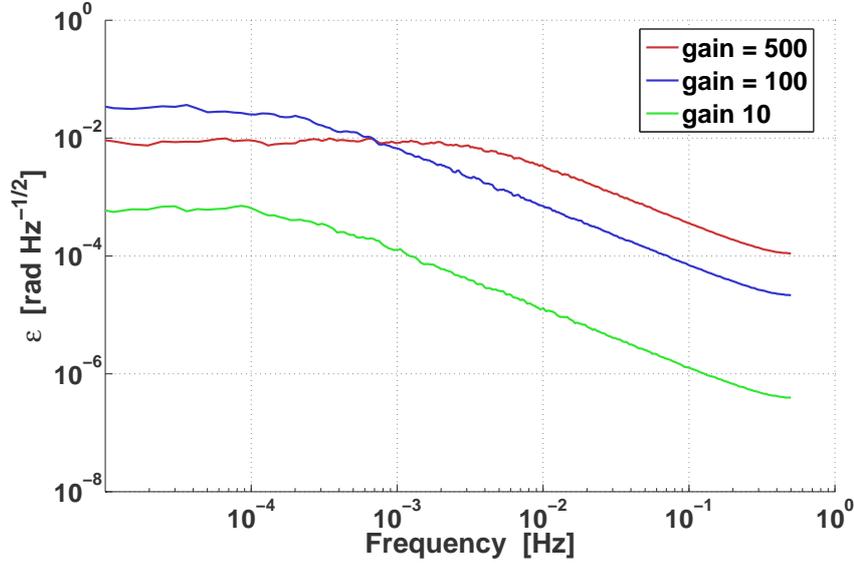
Figure 5.3: Error term of sideband induced noise. $\varepsilon = \cos{(\phi(t))}$, where $\phi(t)$ is plotted in Figure 5.2.

curve provides a measure for LISA's sensitivity as a function of frequency. It combines influences like noises, duration of the measurement and the signal to noise ratio (SNR) with the response of LISA to gravitational noise sources. This response is traditionally computed for an isotropic distribution of sources with random polarisation. The equation that leads to this curve is the following

$$h = \text{SNR}\sqrt{\frac{\text{PSD}_{\text{noise}}}{\text{T}}} \times \frac{1}{\text{RESP}_{\text{GW}}^{\text{rms}}}, \tag{5.3}$$

where SNR is the signal-to-noise-ratio, t is the time over which the measurement is averaged. $RESP_{GW}^{rms}$ represents the gravitational wave response of an isotropic distribution of sources with random polarisation. The traditional values for the parameters in the equation above are:

SNR = 5,

T = 1 year.

These values are also the ones chosen for the computation of the sensitivity curves in this thesis. The variable $\text{PSD}_{\text{noise}}$ used in the computation of the sensitivity curve shown in Figure 5.5 includes the power spectrum for the noises listed in Table 5.1. The values for the noise sources are the default values given in Table 5.1. As indicated in the table, noise contributions from line 3 to 8 were combined to a single term, called the optical path noises, by summing them quadratically. For this, there frequency dependence was taken to be the same. In the plot in Figure 5.5 the shot noise is as well included in the combination, such that we are left with three curves. The test mass noise, the optical path noise including shot noise and the resulting sensitivity curve. The sensitivity curve will in the following be plotted
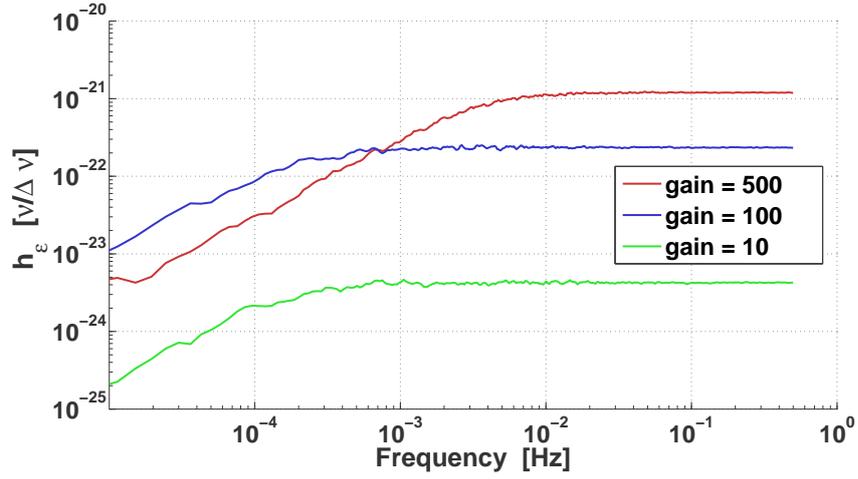
(a) Noise term with amplitude of A = 1 mrad.
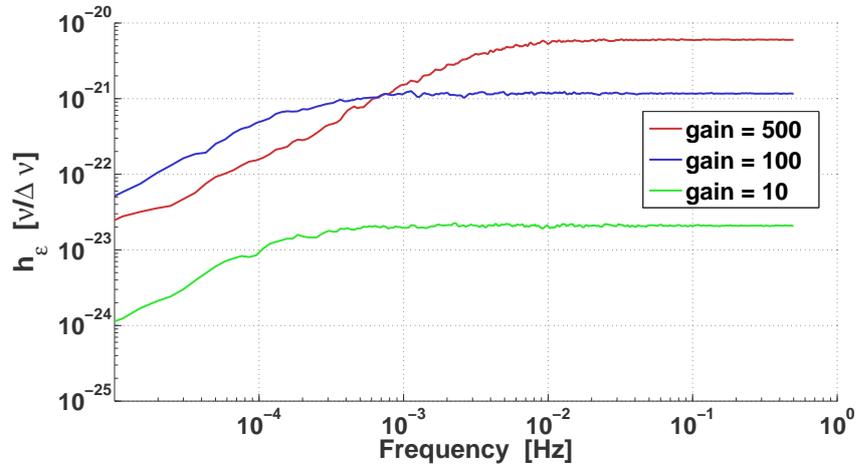
(b) Noise term with amplitude of A = 5 mrad.

Figure 5.4: Error term of sideband induced noise for two different values of the amplitude, A. h_ε in the units of fractional frequency fluctuations, $\nu/\Delta\nu$. The ASD of the error term ε corresponding to the amplitude of 1 mrad (a), was plotted in rad/$\sqrt{\text{Hz}}$ in Figure 5.3. For every amplitude three different gains of transfer functions have been used.

as reference curve for the visualisation of the impact of the newly injected sideband induced noise (SIN) on the sensitivity of LISA.
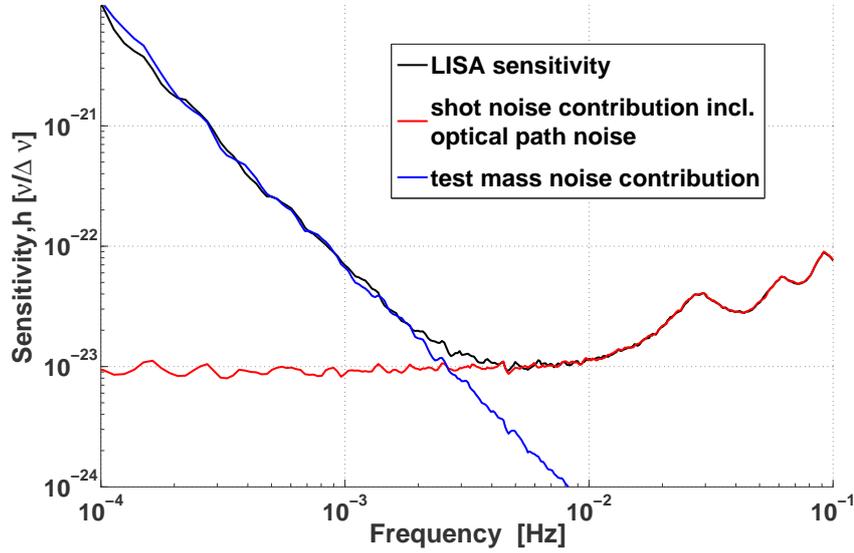

Figure 5.5: LISA noise budget. Plotted are test mass noise, and shot noise including optical path noise.

5.5.1 Data generation

As described above, the executable file of LISACode is called together with an input xml-file by which the desired parameter values used are set. Traditionally this is done separate for the input noises and the gravitational wave response of LISA introduced above.

For the data generation two input xml-files are generated: one for the definition of the noise sources to be injected and one for setting the parameters used in the computation of the response to gravitational waves. For each input file LISACode is called separately and hence two individual time-series are generated accordingly.

For the computation of the sensitivity curve these two data sets need to be combined according to Equation 5.3. This is done using LTPDA and the steps are as follows:

1. The output time-series of LISACode according to the noise sources set and LISAs response to gravitational wave sources are stored in Analysis Objects (AO).

2. The amplitude spectral densities of the time-series are computed using `lpsd` ([24]).

3. The sensitivity, h, is calculated according to Equation 5.3.

For the generation of the sensitivity curve shown in Figure 5.5 the input xml-file for the noises contains the default values as given in Table 5.1. The input file for the computation of LISAs gravitational wave response also contained the

traditional settings. As such the response was computed based on a finite set of coherent gravitational wave sources with random polarisation.

For the case of the sideband induced noise time-series the behaviour of LISACode with inputs needed to be changed slightly. The input noise file usually consists of the frequency dependency of the individual noise sources. But the sideband induced noise was simulated in the time-domain in order to keep the periodic non-linear behaviour of the error term.

For this project the scientists programming LISACode have changed it such that it is now possible to read in time-series data as noise sources. The command for reading the generated data sets is included in the xml-file for injecting the noise sources. Since the noise is set separately for each of the six phasemeters in LISA, six data streams are read into LISACode. This is the first time that a noise source has been injected into LISACode in the form of a time domain.

The complete process from the generation of noise to the resulting sensitivity curve is shown schematically in Figure 5.6. The resulting data of the described procedure are presented in the following section.

5.5.2 The impact of the sideband induced noise on the LISA sensitivity curve

Here the final sensitivity curves with the newly injected sideband induced noise are presented. The plots in Figure 5.7 show a sensitivity curve for each noise term generated following the above described process and injected into LISACode. The reference curve (black) is the same as in Figure 5.5 it was generated using the default noise terms without any additional noise.

Figure 5.7(a) demonstrates the impact on the sensitivity for the SIN with an amplitude of, A = 1 mrad. It can be seen that with this amplitude and a transfer function gain of 10 the non-linear noise source does not change the sensitivity curve. For higher gains, g = 100 and g = 500, the sensitivity decreases significantly.

In Figure 5.7(b) the same is shown for an amplitude of A = 5 mrad. In this case all three generated noise terms have an impact on the LISA sensitivity curve.

Hence reducing the amplitude of the sideband induced noise lowers the impact on the sensitivity as expected. Reducing the amplitude corresponds to reducing the combined amplitude of the sidebands due to electronic cross-talk (see Chapter 3.4 and [15]) and straylight.

Note that the model of the noise generation originates from the interferometric measurements accomplished in LISA Pathfinder laboratory experiments. These parameters are most likely too pessimistic estimates for LISA. Hence, the magnitude of the resulting impact of the noise terms on the LISA sensitivity curves does not claim to be realistic.

5.6 Conclusion

The aim of the project presented in this chapter was to introduce the first non-linear and non-Gaussian noise source into a data generator for LISA. It has been a start of transferring investigations and experiences made in LISA Pathfinder to
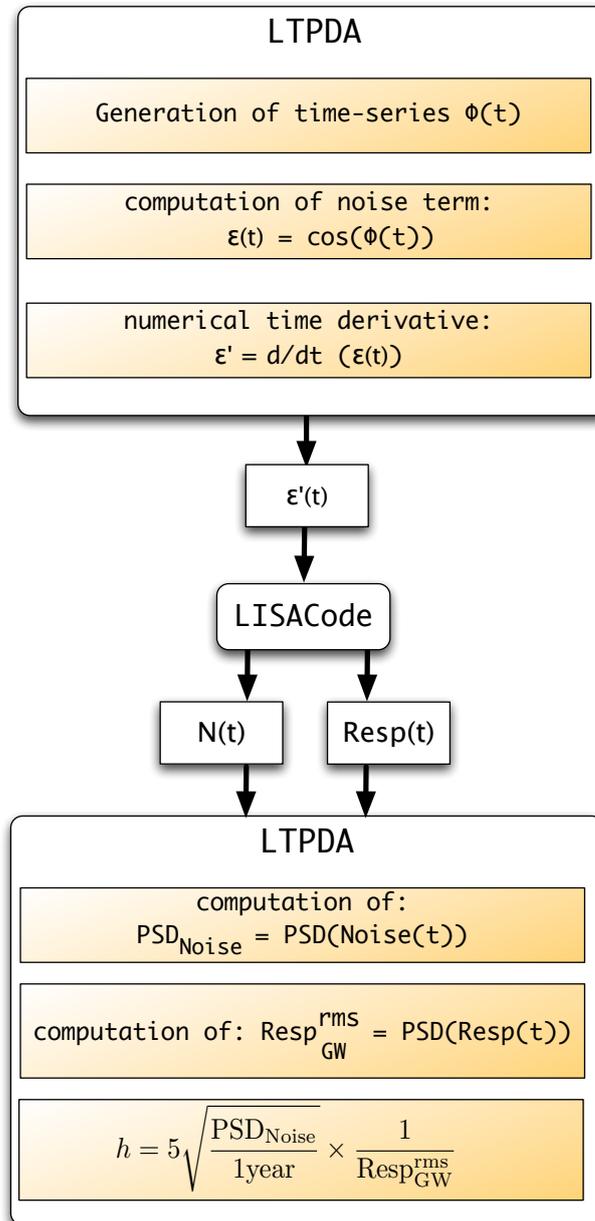
Figure 5.6: Schematic of process of noise injection into LISACode and computation of the resulting sensitivity curve. The software tool LTPDA was used for every computation outside LISACode.
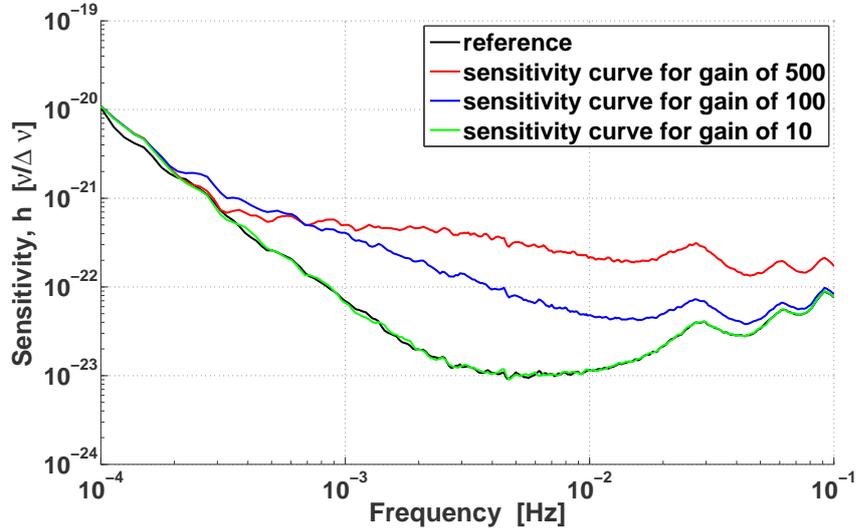
(a) Sensitivity curve with noise term ε where A = 1 mrad.
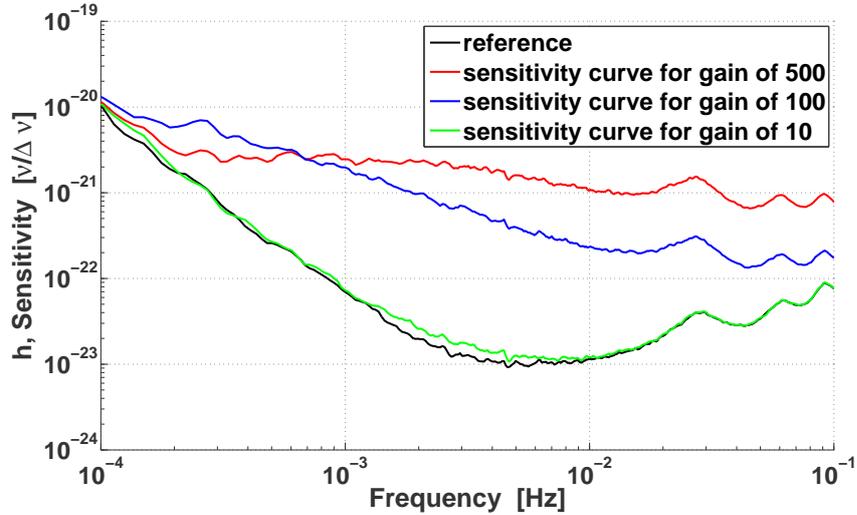
(b) Sensitivity curve with noise term ε where A = 5 mrad.

Figure 5.7: Impact of the sideband induced noise on the LISA sensitivity. The curves are shown for generated noises, ε, with amplitudes A = 1 mrad (a) and A = 5 mrad (b). For each amplitude noise according to a transfer function with three different gains were produced and the influence on the sensitivity curve was probed. For the reference curve no additional noise was injected. It is the same as shown in Figure 5.5.

the ongoing preparation of the data analysis for LISA. The order of magnitude of the noise might not be transferrable from LISA Pathfinder to LISA but at present we see no reason why this type of noise will not be observed in LISA. Further investigations on such realistic noise sources will improve the quality of the mock LISA data challenges (MLDCs) accomplished in preparation of the mission.

Bibliography

[1] S. A. Hughes É. É. Flanagan. The basics of gravitational wave theory. *New J.Phys.*, 7(204), 2005.

[2] A. Einstein. On the electrodynamics of moving bodies. *Ann. Phys.*, 17(891), 1905.

[3] A. Einstein. On the general theory of relativity. *Sitzungsberichte Preußische Akademie der Wissenschaften, Math. Phys.*, (778), 1915.

[4] A. Einstein. On gravitational waves. *Sitzungsberichte Preußische Akademie der Wissenschaften, (Math. Phys.)*, 154, 1918.

[5] R. A. Hulse and J.H. Taylor. Discovery of a pulsar in a binary system. *Astrophys. J.*, 195:L51–L53, 1975.

[6] LISA study team. LISA. Laser Interferometer Space Antenna for the detection and observation of gravitational waves. Technical report, 1998.

[7] M. Tinto, F. B. Estabrook, and J. W. Armstrong. Time delay interferometry and LISA's Sensitivity to Sinusoidal Gravitational Waves. *http://list.caltech.edu/doku.php?id=mission_documents*, 2002.

[8] J. Baker et al. LISA: Probing the Universe with gravitational waves, science case document, 2007.

[9] W. J. Weber, A. Cavalleri, R. Dolesi, G. Fontana, M. Hueller, and S. Vitale. Position sensors for LISA drag-free control. *Class. Quantum Grav.*, 19:1751.–1756, 2002.

[10] [online]Available from: `http://sci.esa.int/lisapf`.

[11] S. Anza et. al. The LTP experiment on the LISA Pathfinder mission. *Class. Quantum Grav.*, 22:125–138, 2005.

[12] F. Steier. The end-to-end test bed of the optical metrology system on-board LISA Pathfinder. *Class. Quantum Grav.*, 26(9), 2009.

[13] N. Brandt, M. Hirth, W. Fichter, R. Schubert, C. Warren, and D. Wealthy. S2-ASD-RP-3036 Experiment Performance Budget. Technical report, EADS Astrium Deutschland, 2007.

[14] M. Hewitson. Acceleration noise characterisation along the x-axis. Technical report, Albert-Einstein-Institut Hannover, In preparation.

[15] G. Heinzel, V. Wand, A. García Marín, F. Steier, F. Guzmán, C. Killow, D. Robertson, H. Ward, and C. Braxmaier. S2-AEI-TN-3028 Investigation of noise sources in the LTP interferometer. Technical report, Albert-Einstein-Institut Hannover, 2005.

[16] M. Hewitson, A. García Marín, and M. Nofrarias. S2-AEI-TN-3058 Laser frequency-noise characterisation for LTP. Technical report, Albert-Einstein-Institut Hannover, 2009.

[17] A. Monsky, F. Steier, and A. García Marín. S2-AEI-TN-3062 OPD noise investigations for LTP. Technical report, Albert-Einstein-Institut Hannover, 2009.

[18] M. Armano et al. LISA Pathfinder: the experiment and the route to LISA. *Class. Quantum Grav.*, 26(9), 2009.

[19] M. Hewitson et al. Data analysis for the LISA Technology Package. *Class. Quantum Grav.*, 26(9), 2009.

[20] S. Vitale. User requirements document - s2-utn_rs-3001. Technical report, University of Trento.

[21] M. Hewitson. S2-AEI-RS-3001 LTP Data Analysis Software Requirements Document. Technical report, Albert-Einstein-Institut Hannover, 2008.

[22] G. Heinzel, M. Hewitson, and A. Monsky. S2-AEI-TN-3037 Some thoughts on data analysis. Technical report, Albert-Einstein-Institut Hannover, 2006.

[23] [online]Available from: `http://www.lisa.aei-hannover.de/ltpda/usermanual/helptoc.html`.

[24] G. Heinzel M. Tröbs. Improved spectrum estimation from digitized time series on a logarithmic frequency axis improved spectrum estimation from digitized time series on a logarithmic frequency axis improved spectrum estimation from digitized time series on a logarithmic frequency axis. *MES*, 2005.

[25] Graphviz. graph visualisation software.

[26] J. N. Franklin. Numerical simulation of stationary and non-stationary Gaussian random processes. *SIAM*, 7(1):68–80, 1956.

[27] G. Heinzel. S2-AEI-TN-3034 Generation of random time series with prescribed spectra. Technical report, Albert-Einstein-Institut Hannover, 2006.

[28] Numerical recipes. Available from: `http://www.nrbook.com/`.

[29] A. García Marín. *Minimisation of optical pathlength noise for the detection of gravitational waves with the spaceborne laser interferometer LISA and LISA Pathfinder.* PhD thesis, Institut für Gravitationsphysik, Universität Hannover, 2007.

[30] F. Steier. *Interferometry techniques for spaceborne gravitational wave detectors.* PhD thesis, Institut für Gravitationsphysik Universität Hannover, 2008.

[31] F. Guzmán Cervantes. *Gravitational wave observation from space: optical measurement techniques for LISA and LISA Pathfinder.* PhD thesis, Institut für Gravitationsphysik, Universität Hannover, 2009.

[32] V. Wand. Interferometrie und Phasenauslesung für die wissenschaftliche Weltraummission SMART2. Master's thesis, Institut für Gravitationsphysik, Universität Hannover, 2003.

[33] G. Heinzel, V. Wand, A. García, O. Jennrich, C. Braxmaier, D. Robertson, K. Middleton, D. Hoyland, A. Rüdiger, R. Schilling, U. Johann, and K. Danzmann. The LTP interferometer and phasemeter. *Class. Quantum Grav.*, 21:581–587, 2004.

[34] Private communication with W. Fichter, 2006.

[35] F. Guzmán Cervantes, F. Steier, G. Wanner, G. Heinzel, and K.Danzmann. Subtraction of test mass angular noise in the LISA technology package interferometer. *Appl. Phys. B*, 90:395–400, 2008.

[36] Paolo Fornasini. *The Uncertainty in Physical Measurements.* Springer, 2008.

[37] V. Wand, J. Bogenstahl, C. Braxmaier, K. Danzmann, A. García, F. Guzmán, G. Heinzel, J. Hough, O. Jennrich, C. Killow, D. Robertson, Z. Sodnik, F. Steier, and H. Ward. Noise sources in the LTP heterodyne interferometer. *Class. Quantum Grav.*, 23:159–167, 2006.

[38] M. Hewitson and A. García Marín. Laser amplitude noise characterisation for LTP. Technical report, Albert-Einstein-Institut Hannover, 2010.

[39] A. Monsky et al. The first mock data challenge for LISA Pathfinder. *Class. Quantum Grav.*, 26(9), 2009.

[40] W. Fichter, P. Gath, S. Vitale, and D. Bortoluzzi. LISA Pathfinder drag-free control and system implications. *Class. Quantum Grav.*, 22:139–148, 2005.

[41] S. Vitale. S2-UTN-TN-3060. Technical report, University of Trento, 2007.

[42] S. Vitale. S2-UTN-TN-3045 Measurement of LTP dynamical coefficients by system identification. Technical report, University of Trento, 2008.

[43] S. Vitale. Generation of noise for the mock data challenge of the LTP analysis. Technical report, University of Trento, 2007.

[44] L. Ferraioli, M. Hueller, and S. Vitale. Discrete derivative estimation in LISA Pathfinder data reduction. *Class. Quantum Grav.*, 26(9), 2009.

[45] S. Vitale and K. Danzmann. LTPA-UTN-ScRD Science Requirements and Toplevel Architecture Definition for the LISA Technology Package (LTP) on Board LISA Pathfinder (SMART-2). Lpf project document. 9,11, 2005.

[46] M. Nofrarias. S2-AEI-TN-3066 Data generation for the third release of the LTP Mock Data Challenge II (MDC2r3). Technical report, Albert-Einstein-Institut Hannover, 2009.

[47] Siegmund Brandt. *Data Analysis*. Springer, 1989.

[48] M. Nofrarias and C. Röver. Bayesian parameter estimationin MDC2 release3 (MDC2r3). Technical report, Albert-Einstein-Institut Hannover, 2009.

[49] A. Petiteau, G. Auger, H. Halloin, O. Jeannin, E. Plagnol, S. Pireaux, T. Regimbau, and J. Vinet. Lisacode: A scientific simulator of lisa. *Phys. Rev. D*, 77, 2007.

[50] S. Dhurandhar, K. Rajesh Nayak, S. Koshti, and J.-Y. Vinet. Fundamentals of the LISA stable flight formation. *Class. Quantum Grav.*, 22:481–487, 2005.

[51] B. Chauvineau, T. Regimbau, J.-Y. Vinet, and S. Pireaux. Relativistic analysis of the LISA long range optical link. *Phys. Rev. D*, 72(122003), 2005.

[52] M. Vallisneri. Geometric time delay interferometry. *Phys. Rev. D*, 2005.

[53] S. V. Dhurandhar, K. Rajesh Nayak, and J.-Y. Vinet. Algebraic approach to time delay data analysis for LISA. *Phys. Rev. D.*, 2002.

Acknowledgements

I count myself most lucky for having given the possibility to do my PhD thesis on the LISA and LISA Pathfinder project at the AEI. It is a pleasure to work at the institute and my work as a PhD student was most valuable and exciting.

My gratitude to Prof. Dr. Karsten Danzmann for giving me this opportunity.

I want to thank Eric Plagnol for agreeing to be my second examiner. Also, I want to thank him for his kind support and long discussions on the LISA related work, and his hospitality at APC, Paris. Merci beaucoup.

Most of my knowledge on data analysis I owe to Dr. Martin Hewitson. He provided the best mentoring I could imagine. There was always time for my questions and discussions. Thank you for allowing me to be part of the exciting project of LTPDA. I also very much appreciate that all the daily work was kept away from me during the writing period.

Thank you to Dr. Gerhard Heinzel for the great scientific guidance during my time as a PhD student. A substantial part of the work presented here rests upon his ideas and I am aware of my luck having him as a group leader.

I'm grateful for the most wonderful working atmosphere among the LTPDA team. Especially I would like to thank Mauro Hueller, Nicola Alex Tateo, Luigi Ferraioli, Boutheina Rais, Michele Armano, Adrien Grynagier, Marc Diaz, Josep Sanjuan, Miquel Nofrarias and Ingo Diepholz for the amazing collaboration and team spirit, despite the remote working conditions. I also have to say thanks for the enjoyable leisure activities after our working meetings far away from home.

Thank you to the LISA group at AEI for the friendly atmosphere at work. I would particularly like to thank Felipe Guzmán, Frank Steier and Antonio García for acquainting me with the LISA and LISA Pathfinder project. They made sure that I didn't lose sight of the laboratory experiment when assembling the data analysis tools, I am sure my work benefited a lot from this.
The long discussions with Miquel Nofrarias have contributed very much to my understanding of the theoretical side of the data analysis and I am aware of his workload due to my thesis writing. Ingo Diepholz has always kindly assisted me with toolbox issues no matter how huge his own workload was.

Thank you for the great times also after work: Antonio, Felipe, Heather Audley, Miquel, Ingo, Juan José Esteban, Roland Fleddermann and Christian Diekmann.

For proof reading this thesis I'm in debt to Martin, Gerhard, Frank and Miquel. All remaining mistakes are of course my own responsibility.

Thank you to my family for their generous and unrestricted support at all times. To Frank: your contribution is priceless. Your comfort and encouragement simply brings out the best in me. Danke für alles.

Curriculum Vitae

Anneke Monsky

Schneiderberg 17
30167 Hannover

Born on December 13$^{\text{th}}$, 1981 in Celle.

11/2006 - date:	Doctoral studies in Physics and scientific assistant Gottfried Wilhelm Leibniz Universität Hannover.
11/2006 - date:	Member of the International Max Planck Research School (IMPRS).
05/2005 - 04/2006:	Diploma Thesis, Laserzentrum Hannover.
10/2004 - 08/2006:	Master of Science in Physics, Carl von Ossietzky Universität Oldenburg.
09/2002 - 02/2003:	Semester abroad: University of Szeged, Hungary.
10/2000 - 06/2004:	Bachelor of Engineering, Carl von Ossietzky Universität Oldenburg.
07/1993 - 07/2000:	Abitur, Robert Bosch Gesamtschule Hildesheim.
1987 - 1993:	Grundschule & Orientierungsstufe Bockenem.

Publications

- A. Monsky, M. Hewitson, L. Ferraioli, G. Wanner, M. Nofrarias, M. Hueller, I. Diepholz, A. Grynagier, M. Armano, M. Benedetti, J. Bogenstahl, D. Bortoluzzi, P. Bosetti, N. Brandt, A. Cavalleri, G. Ciani, I. Cristofolini, M. Cruise, K. Danzmann, R. Dolesi, J. Fauste, D. Fertin, W. Fichter, A. García, C. García, F. Guzmán, E. Fitzsimons, G. Heinzel, D. Hollington, J. Hough, D. Hoyland, O. Jennrich, B. Johlander, C. Killow, A. Lobo, D. Mance, I. Mateos, P. W. McNamara, D. Nicolini, D. Nicolodi, M. Perreur-Lloyd, E. Plagnol, G. D. Racca, J Ramos-Castro, D. Robertson, J. Sanjuan, M. O. Schulte, D. N. A. Shaul, M. Smit, L. Stagnaro, F. Steier, T. J. Sumner, N. Tateo, D. Tombolato, G. Vischer, S. Vitale, H. Ward, S. Waschke, P. Wass, W. J. Weber, T. Ziegler, P. Zweifel.
  The first mock data challenge for LISA Pathfinder Class. Quantum Grav., 26, 2009.

- M. Hewitson, M. Armano, M. Benedetti, J. Bogenstahl, D. Bortoluzzi, P. Bosetti, N. Brandt, A. Cavalleri, G. Ciani, I. Cristofolini, M. Cruise, K. Danzmann, I. Diepholz, R. Dolesi, J. Fauste, L. Ferraioli, D. Fertin, W. Fichter, A. García, C. García, A. Grynagier, F. Guzmán, E. Fitzsimons, G. Heinzel, M. Hewitson, D. Hollington, J. Hough, M. Hueller, D. Hoyland, O. Jennrich, B. Johlander, C. Killow, A. Lobo, D. Mance, I. Mateos, P. W. McNamara, A. Monsky, D. Nicolini, D. Nicolodi, M. Nofrarias, M. Perreur-Lloyd, E. Plagnol, G. D. Racca, J Ramos-Castro, D. Robertson, J. Sanjuan, M. O. Schulte, D. N. A. Shaul, M. Smit, L. Stagnaro, F. Steier, T. J. Sumner, N. Tateo, D. Tombolato, G. Vischer, S. Vitale, G. Wanner, H. Ward, S. Waschke, P. Wass, W. J. Weber, T. Ziegler, P. Zweifel.
  Data analysis for the LISA Technology Package Class. Quantum Grav., 26, 2009.

- M. Armano, M. Benedetti, J. Bogenstahl, D. Bortoluzzi, P. Bosetti, N. Brandt, A. Cavalleri, G. Ciani, I. Cristofolini, M. Cruise, K. Danzmann, I. Diepholz, R. Dolesi, J. Fauste, L. Ferraioli, D. Fertin, W. Fichter, A. García, C. García, A. Grynagier, F. Guzmán, E. Fitzsimons, G. Heinzel, D. Hollington, J. Hough, M. Hueller, D. Hoyland, O. Jennrich, B. Johlander, C. Killow, A. Lobo, D. Mance, I. Mateos, P. W. McNamara, A. Monsky, D. Nicolini, D. Nicolodi, M. Nofrarias, M. Perreur-Lloyd, E. Plagnol, G. D. Racca, J Ramos-Castro, D. Robertson, J. Sanjuan, M. O. Schulte, D. N. A. Shaul, M. Smit, L. Stagnaro, F. Steier, T. J. Sumner, N. Tateo, D. Tombolato, G. Vischer, S. Vitale, G. Wanner, H. Ward, S. Waschke, P. Wass, W. J. Weber, T. Ziegler, P. Zweifel.
  LISA Pathfinder and the route to LISA Class. Quantum Grav., 26, 2009.