

INSTITUTE OF APPLIED PHYSICS
TECHNISCHE UNIVERSITÄT DARMSTADT



Adaptive Spectrum Analysis for Nanooptical Systems

A Dissertation accepted by
the Department of Physics of
Technische Universität Darmstadt

for the degree of
Doktor der Naturwissenschaften
(Dr. rer. nat.)

by

Karl Koch
from Aschaffenburg

Darmstadt 2010
D17

Date of submission: 25.11.2009

Date of defense: 27.01.2010

Referee: Prof. Dr. Theo Tschudi

Coreferee: Prof. Dr. Thomas Walther

*Always do right, this will gratify
some people and astonish the rest.*

Mark Twain

Contents

1	Introduction	1
2	Nanooptical Model System	5
2.1	Micromechanical System Components	7
2.2	Nanooptics	13
3	Adaptive Methods for Spectrum Analysis	17
3.1	Adaptive Filtering Techniques	19
3.1.1	The Key Role of Crosscorrelation in Identification Problems	20
3.1.2	Least Square Adaptive Filtering Reviewed	23
3.1.3	The Essence of Kalman Filtering	28
3.2	Online Implementation of Spectrum Analysis Filters	33
3.2.1	Adaptation Based on Oscillator Ensembles	34
3.2.2	Realtime Aspects of Digital Signal Processing	57
4	System Characterization Experiments	67
4.1	Complex Line Tracks	69
4.1.1	Reference Synchronization	70
4.2.2	Surface Modes Response Rectification	73
4.2.3	Resonance Curve Recordings	79
4.2	Broadband Spectral Adaptation.....	85
5	Conclusions and Outlook	95

Bibliography	99
Appendix	107
A Code Examples	111
B Mathematical Derivations	131
C Mechanical Drawings	147
D Electronics Design	163
Acknowledgement	193
Curriculum Vitae	195

Summary

The scientist's signal processing toolbox for realtime frequency domain characterization of given systems, in particular ones made by state of the art nanooptical engineering, is incomplete. Nowadays the standard method for the spectral decomposition of time varying signals still uses a computational technique, that is older than 40 years. There exists a vast amount of literature on the challenge of workaround modifications and on refinement proposals to overcome the experimental bottlenecks of the Fast Fourier Transformation in realtime applications. This is a true motivation to rethink the analysis strategy. The thesis demonstrates unequivocally that a powerful alternative to the Fast Fourier Transformation can be provided by adaptive filtering, which is much more flexible, less computational demanding and also faster in realtime spectrum analysis than the prior standard. The train of thoughts on the new method and the experimental results on adaptation dynamics are published here for the first time. We denote the tool as the *Adaptive Spectrum Analysis Filter* (ASA).

The digital signal processing algorithm is evaluated in an atomic force microscope. The custom made instrument is designed to characterize nano resonances of piezomechanical actuators. The modes are sensed by means of a cantilever probe utilizing the well known beam deflection principle that turns the oscillations into a measurable analog voltage. The innovative realtime application of adaptive spectrum analysis decomposes the system response *time resolved* into the Fourier components and thus determines the complex mechanical transfer function of the actuators. The thesis reports on single line analysis as well as broadband spectral characterization experiments. The degree of complexity – defined here as the logarithmic ratio of system to structure size – is a most relevant and rapidly growing technology parameter. Typical values are on the order of 5 to 6. The study of such systems is a big challenge with respect to the instrument design. Said atomic force microscope is part of a nanocartographer that integrates micro and nano mechanical components by an innovative design to measure the system properties *spatially resolved*. The resolution limit of 10 nm on a navigation range of 10 cm equals a complexity degree of 7 and is demonstrated by coordinate based imaging of artificial opal films and large scale optical gratings. The nanocartographer plays the role of a nanooptical model system that is featured to qualify the new method of realtime spectrum analysis and to quantify the adaptation properties of the proposed algorithm. The nonlinear analysis techniques involved are rooted in adaptive Kalman filtering. The presented experiments and simulations give proof that a realtime implementation of the strategy on a digital signal processor (DSP) is clearly advantageous as compared to the customary Fast Fourier Transformation. Having up to a 10-fold less numerical effort the adaptive approach allows for a local refinement of the spectral resolution in the frequency domain, where the numerics can be tailored to the observation task in order to obtain a further reduced effort. In case of single frequency analysis the response time constant decreases by a factor of 10.

In the course of this project new possibilities of interdisciplinary applications showed up. To give an example : the method has been successfully ported from nanooptics to electrical engineering. Grid harmonics in a public power network have been spectrally decomposed. Moreover compensation mechanisms utilizing solar grid inverters were realized to achieve an effective damping of the harmonics. The application is patented and published under WO 2009/056158 A1.

Zusammenfassung

Der Werkzeugkasten des Wissenschaftlers für die Echtzeitspektralanalyse realer Systeme, insbesondere solcher aus dem Bereich der Nanooptik, ist unvollständig. Der Standard für die spektrale Zerlegung zeitabhängiger Signale verwendet heute immer noch maßgebend eine Methode der digitalen Signalverarbeitung, die vor über 40 Jahren erfunden wurde. Die Vielzahl an Publikationen zur Herausforderung, die schnelle Fouriertransformation durch Ergänzungen und Verfeinerungen mit Echtzeitfähigkeit auszustatten, regt dazu an über die grundsätzliche Strategie des Verfahrens nachzudenken. Im Rahmen der Doktorarbeit wird eindeutig gezeigt, daß es eine leistungsfähige Alternative aus dem Bereich der adaptiven Filter gibt, die deutlich flexibler, weniger rechenaufwendig und im Analyseergebnis schneller ist, als der bisherige Standard. Die gedanklichen Ansätze des neuen Verfahrens und die experimentellen Untersuchungen zur Adaptionsdynamik sind hier erstmalig publiziert. Das Werkzeug wird von uns als *Adaptiver Spektralanalyse Filter* bezeichnet (ASA).

Der signalverarbeitende Algorithmus wurde an einem Rasterkraftmikroskop getestet. Der Eigenbau dient dazu, Nanoresonanzen an piezomechanischen Baugruppen zu charakterisieren. Die Resonanzerscheinungen werden mit Hilfe einer Cantileversonde unter Anwendung des Lichtzeigerprinzips in meßbare Analogsignale umgewandelt. Das neue Verfahren der adaptiven Spektralanalyse zerlegt die Systemantwort *zeit aufgelöst* in ihre Fourier Komponenten und bestimmt auf diese Weise die komplexe mechanische Übertragungsfunktion der Baugruppen. Es werden in der Arbeit sowohl Experimente zur Einzelfrequenzanalyse als auch zur breitbandigen Systemcharakterisierung vorgestellt. Der Komplexitätsgrad Nanooptischer Systeme – definiert als das logarithmische Verhältnis von Systemgröße zu minimaler Strukturbreite – ist ein relevanter und rasant anwachsender Technologieparameter. Typische Werte liegen heute bei 5 bis 6 Größenordnungen. Die Untersuchung solcher Systeme stellt außergewöhnliche Anforderungen an den Instrumentenentwurf. Das genannte Rasterkraftmikroskop ist Teil eines Nanokartografen, in welchem sich mikro- und nanomechanische Module innovativ ergänzen, um die Systemeigenschaften *orts aufgelöst* erfassen zu können. Die Auflösungsgrenze von 10 nm bei einer Stellweite von 10 cm entspricht einem Komplexitätsgrad von 7 und wird an künstlichen Opalfilmen bzw. ausgedehnten optischen Gitterstrukturen demonstriert. Der Nanokartograf übernimmt die Rolle eines Nanooptischen Modellsystems, an dem die neue Methode zur Echtzeitspektralanalyse qualifiziert und die Analyseigenschaften des Verfahrens quantifiziert werden. Die angewandten nichtlinearen Analysetechniken basieren auf adaptiver Kalman Filterung. Die vorgestellten Experimente und Simulationen beweisen, dass der entwickelte Algorithmus bei einer Echtzeitimplementierung auf einem Digitalen Signal Prozessor (DSP) klare Vorteile gegenüber der üblichen schnellen Fouriertransformation besitzt. Bei einem bis zu 10-fach niedrigeren Rechenaufwand erlaubt der adaptive Ansatz eine lokale Verfeinerung der spektralen Auflösung im Frequenzraum, wodurch die Numerik an das zu untersuchende System optimal angepasst und der Rechenaufwand noch weiter minimiert werden kann. Im Falle der Einzelspektralanalyse reduzieren sich dann die Ansprechzeiten um einen Faktor 10.

Im Rahmen dieser Arbeit wurden neue interdisziplinäre Anwendungsmöglichkeiten des Verfahrens erschlossen. Beispielweise ließ sich die Methode erfolgreich von der Nanooptik in den Bereich der Elektrotechnik portieren. Dort wurden Spannungsoberwellen im öffentlichen Netz spektral aufgelöst und erfolgreich Kompensationsmechanismen mit Hilfe von Solarwechselrichtern umgesetzt. Die Anwendung ist unter der Veröffentlichung WO 2009/056158 A1 patentrechtlich geschützt.

1 Introduction

The experience of discovering something new is amazing. Mankind has always been pushed by the emotion, that accompanies this event. It makes people ask “ why ? “, on and on. But curiosity is merely the ignition spark. It takes serious questions, to form the nucleus of a goal-directed and sustainable work. The road map of technology rules the evolution of societies today. Thus it is an import and relevant source of questions, to be answered by the scientific community. This work is technology driven, but the train of thought starts from physics. It *queries the completeness* of the scientist's signal processing toolbox for the realtime frequency domain characterization of systems, in particular ones made by state of the art nanooptical engineering. Technology creates structures of steadily growing inner complexity. Faster and smaller, according to Moore's law. Nowadays standard spectrum analysis however, uses a computational technique [CT65], that is older than 40 years. There exists a vast amount of literature on the topic of workaround modifications to overcome the experimental bottlenecks of the Fast Fourier Transformation in realtime applications [BRI88]. This is a true motivation, to rethink the analysis strategy. The motivation of inventing new methods and instruments for the characterization of nature has a long tradition in physics. Educated capabilities in observation and conclusion are helpful, as the focus is set on an extension of the analysis toolbox. Getting to the point : the behavior of systems is determined by the response to a stimulus. To be measured, sensors are applied and signals are digitized. The time series are processed by spectral decomposition, in order to reveal the dependencies of system input and output quantities. The thesis answers the initial query in a clear statement : a powerful alternative to the Fast Fourier Transformation can be provided, which is much more flexible, less computational demanding and also faster in *realtime spectrum analysis* than the prior standard.

The spectrum of a signal can be found by *deterministic* methods that measure the spectral content. The approach is to be seen in contrast to *stochastic* analysis techniques that estimate magnitudes and phasing of the spectral lines based on statistical assumptions. As the decomposition in frequency domain is gradually approximated by a discrete time signal processing algorithm, the analysis is said to follow an *adaptive strategy*. This topic was explored in the past and results were reported for both the deterministic and the stochastic statement. But the dimension of the observation domain was strictly limited in the experiments, due to the computational demand raising to the second power. Mutschler *et al.* showed the potential of a deterministic signal processing algorithm in very fast single harmonic adaptation [MUT02]. The method computes a complex weight that maps a unity oscillator twiddling at constant frequency to a given sine wave, whereas magnitude and phase of the sine step randomly. Least squares error of the predicted signal acts as a merit of match. Heydt and Beides published on a stochastic identification of voltage harmonics in public grids based on Kalman filtering, which is capable of tracking a time variable spectral distribution [HEY91]. Low orders up to 13 are considered in the contribution. The present thesis however examines in particular,

how a Kalman filter operates, that maps a high dimensional ensemble of unity oscillators to a time varying reference signal by means of a Fourier synthesis. The oscillators twiddle at preselected frequencies. As the adaptive filter tunes the Fourier coefficients to gain best signal match in time domain, the spectrum of the reference is naturally estimated. This strategy is new and we refer to it as the *adaptive spectrum analysis filter*. It turns out, that the very special condition of the filter input variables being harmonic can be exploited to obtain a most significant reduction of the computational demand. Said relation is exclusively found in the stochastic statement. The effort collapses to a linear dependency on the spectral order. Notice the proposal supports a realtime spectrum analysis featuring several thousand unity oscillators as well as tailored observation scenarios on low orders and in frequency bands. Fast Fourier Transformation is not competitive in this field. Both the innovative idea and the conclusions drawn are reported for the first time.

To give an experimental proof of concept an application example is stated in the research field of nanooptics, more specific in scanning probe microscopy. The high resolution imaging technique usually scans a sharp tip that is attached to a cantilever probe across a sample [BIN86]. Interaction forces between tip and sample are measured to control the vertical distance in a closed loop, which allows for the reconstruction of three-dimensional surface maps. But the instrument can also be used for vibration analysis. In that case the tip touches the sample open loop and the vibration modes are sensed locally at the contact point in style of a record player. Said configuration is used here to measure the response of a commercial piezo actuator to different mechanical stimuli on the nanometer scale. The adaptive spectrum analysis filter estimates the complex transfer function of the device by realtime observation of the vibrational modes. The mode analysis is intended to support a design review of the actuator geometry with respect to an optimization of the motion dynamics. Since the invention and introduction of atomic force microscopy in 1986 by Binnig *et al.* considerable progress has been gained in the major topics resolution [GIS05] and scanning speed [SAL02] [SCH02] [SEB04]. Former is based on frequency modulated cantilever probe imaging. A continuous improvement of the technique has been achieved in the last years. The lever is periodically driven in normal eigenoscillation at a close distance to the sample surface ¹. Detuning of the resonance frequency, rooted in the overlap of the tip and sample wave functions, is measured and allows force imaging on an atomic scale of length. Experiments reveal, that driving amplitude and q -factor of the cantilever play an intimate role in spatial resolution. The influence is clarified in the latest publications ². Scanning at higher speed yet retaining image quality is a question of robust force control strategy. Modern digital signal processing algorithms have been implemented to achieve an enhanced control performance with respect to the equivalent time constant of the closed loop. Image acquisition has been demonstrated at scan speeds 20 times faster than supported by the prior control standards ³. Amongst other robust design options \mathcal{H} -infinity control [SKO01] has been applied to scanning probe microscopy in the cited works. The adaptive spectrum

1 Analogon : imagine a surface atom magnified to the size of a tennis ball, which is about 65 mm. The range of the bonding forces is roughly half the diameter. The front atom of the tip approaches from a distance of 15m and the atomic forces count in the last view centimeters of the approximation.

2 See ref. [GIS05] for a broad review.

3 Refer to the citation of Schetter *et al.* [SCH02].

analysis filter could have been also tested in said research fields. But the less complex resonance experiment was deliberately chosen, to alleviate a clear interpretation of the experimental results and yield an unequivocal validation of the new method.

This thesis is organized as follows :

Chapter 2 is dedicated to the definition and explanation of a model system that is appropriate to act as a test platform for the digital signal processing algorithm. As aforementioned the key experiment is recording vibrational surface modes of a piezo actuator. To explain : the piezo ceramic stack is built in a metal housing. As a sinusoidal stimulus is applied standing waves form on the metal surfaces. The modes show a lateral dimension of several centimeters and an oscillation magnitude in the sub micron range. To image the modes the atomic force measurement head has to be positioned at different recording points of interest. Thus a linear translation subsystem is required to carry the nanooptics and support a coordinate based triple axis navigation. The micromechanical construction is explained in the first section. Flatness, linearity and backlash of the linear motion are measured by scanning different calibration standards. The experimental results of the instrument characterization qualify the unit for resonance recording. In section two the working principle of the custom made atomic force microscope is introduced. The image of an artificial opal film demonstrates the capabilities of the instrument design. Since the integrated system architecture of micromechanics and nanooptics supports coordinate based high resolution imaging the instrument is referred to as *nanocartographer*. It represents the nanooptical model system.

Theoretical understanding of the mathematical concept behind adaptive spectrum analysis is promoted by the topics of chapter 3. In the first section different adaptive filtering techniques are contrasted. A special focus is laid in the discussion on the key role of crosscorrelation in system identification experiments. Least square adaptive filtering is reviewed as an example of a deterministic system identification approach. Kalman filtering represents a stochastic statement of the task. One discovers eye catching similarities of the numerics in both algorithms. But whereas a recursive formulation of the least square filter suffers from numerical instability the computation of the Kalman filter is stable. Evidence for this will be given in that chapter. In section two we elaborate in depth and breadth the time course of the Kalman filter variables, as the input data vector is given by a set of harmonic quadrature oscillations. The insight paves the way for a tremendous simplification of the algorithm, which allows to utilize it in realtime spectral decomposition. A smart reformulation of the Kalman filter for adaptive spectrum analysis and the evaluation of the numerical effort, as implemented on a modern digital signal processor, rounds off the chapter.

Resonance can be driven by single frequency or broadband stimulus. The first option allows to scan the complex transfer function of the nanooptical model system by means of a frequency wobbling experiment in a predefined window. Model system response on an arbitrary stimulus may be synthesized later based on the scanned spectrum. The adaptation dynamics of the adaptive spectrum analysis filter can be judged in a microcomputer simulation and the convergence can be analyzed by comparison of the filter kernel with the reference spectrum. Chapter 4 gives a summary of the related experiments and simulations respectively. It also reports on the performance gain of

the new algorithm as compared to the prior signal processing standards. In section one tracking of single harmonics is illustrated. The major issues of synchronizing the adaptive filter to a sinusoidal stimulus and rectifying the system response are addressed. A common synchronization technique is the digital phase locked loop algorithm whereas digital lock-in amplification is popular in magnitudes and phase rectification. Performance analysis is based on tracking sudden jumps in magnitudes, phase and frequency, as the standards are computed in competition to the adaptive filter. The section is concluded with a documentation of exemplary resonance curves that are recorded by single line adaptive spectrum analysis at different cartographic coordinates on the piezo actuator. Section two presents the results of broadband spectral adaptation. The response of the nanooptical model system is tracked in parallel on a frequency comb with 2048 lines. Both the evolution of estimated magnitudes and phasing are studied in form of various time sequences. The chapter is closed by a determination of the experimental conditions, that allow to measure the transfer function of the nanooptical model system by means of a radix-2 Fast Fourier Transformation within a comparable fidelity.

Finally a summary of the theoretical and experimental results is given. New questions that arise from this work are explicated. Moreover promising interdisciplinary applications are addressed. One that features the new method in the field of electrical engineering [KKO09] has been patented 2009 under WO 2009/056158 A1.

2 Nanooptical Model System

The title of the thesis contains the keyword nanooptics. What do we think of as referring to a nanooptical system ? A concentrated review of the literature on applied physics reveals the topic to be discussed rather diverse. Is it scanning probe microscopy [SEB04] ? Or near field optics [OTH02] ? What about photonic crystals [JOA95] or microlasers [KKO03] ? Surface plasmons [POH07] ? We try a definition: A nanooptical system is one, that embeds a process measured or controlled by the interaction of photon and matter on a scale below the wavelength of visible light. The process may address any physical quantity. For example the magnitude of a mechanical vibration mode, the velocity of an energy transport, the timing of a thermal relaxation and more. Though we are talking about optics, our interest may be directed towards a different quantity. Of course a microlaser is not a nanooptical system. But as we pump or probe the resonator by means of a nearfield sensor the configuration matches the definition.

It becomes self evident that before writing on said systems one has to specify the process and the addressed physical quantities. Moreover the nature of the photon-matter interaction that underlies the nanoscale characterization or control task. To make it short: in this work mechanical properties of nano structures are studied utilizing a scanning probe microscope to stimulate and visualize surface oscillations on the solid-ambient interfaces. So the process is the mechanical coupling of the cantilever probe to the structure modes and the addressed physical quantities are oscillation magnitudes and phasing. The stimulus is tailored to drive sub wavelength response harmonics. We apply the well known beam deflection principle [BIN86] to measure these quantities ¹, thus the photon-matter interaction lies in the reflection of the laser beam on the cantilever's top side. In gist of the matter the 4 system units are: scanner, solid structure, probe and the microscope's laser beam deflection unit. We will refer to this setup as the *nanooptical model system* throughout the thesis. The degree of complexity – defined here as the logarithmic ratio of structure size to minimal signature width – is a most relevant and rapidly growing technological parameter. Nowadays values typically raise up to magnitude 6. Commercial scanning probe microscopes are designed to image surfaces in a limited area ² but with very high spatial resolution ³. They support the characterization of systems that have a degree of complexity between 4 and 5 [VOR01]. Such an instrument is not apt to act as a model system, because the complexity is one magnitude less than the technology limit. This chapter is dedicated to the extensions drafted for a new instrument generation that is competitive.

The first section reports on how customized components of the optomechanical toolbox are recombined to allow for a coordinate-reading nanoscale motion. The translation subsystem carries an atomic force microscope (AFM) sensor to convert the time

1 annotation : the concept converts deviations of the probe into an analog sensor voltage

2 common less than 100 micron length of edge.

3 in air down to 1 nm

dependent surface deformations into a measurable analog signal. Consider the method of adaptive spectrum analysis mentioned in the title to process latter output. The system architecture allows to measure spectral variates in nanostructures with a degree of complexity up to 7 and significantly extends the capabilities of a turnkey scanning probe equipment. It is explained, how creep and thermal drift in the linear translation stages are minimized. Moreover the coordinate measurement is discussed. Since the setup allows for 3-axis linear traveling of the AFM in a virtual cartographic grid the instrument is labeled *nanocartographer*, whereas said coordinate encoding is realized with commercial position sensing technology at sub micron resolution. The drives are user controlled from a laboratory computer. To verify linearity in horizontal translation a flatness calibration standard is scanned, moving the probe across in closed loop contact. As the procedure is repeated with an optical grating⁴ the meshes are calibrated by the recorded line distance. Results of the instrument characterization are added here for sake of completeness.

In section two the working principle of the AFM is illustrated. The microscope has been constructed and setup in this work at the Institute of Applied Physics, TU Darmstadt. While stimulating mechanical modes in the nanooptical model system the AFM has to sense in constant height mode⁵. Control of the probe-surface interaction is deactivated in that state, so unwanted relative motion in the linear translation sliders is crucial. To qualify the nanocartographer for idem operation, a constant height image of an artificial opal film [SPA08] is taken. The specimen consists of face-centered cubic lattices build from SiO_2 monospheres with a mean diameter of 200 nm and is known as photonic crystal layer [VDI02]. The designation artificial opal goes back to the similarities with the natural gemstone opal. Customary cantilever probes are employed both in scanning probe imaging and resonance recording. The image data shows, that the nano cartographer will support spatially resolved vibration analysis. As expected the optomechanical design is insensitive to nonlinear laser beam deflection, mechanical creep and thermal drift. Since the beam deflection unit is not part of the optical toolbox the electronics and optics had to be custom designed. Signal quality severely counts. Thus the analog conditioning circuits have been embedded in the microscope body, to minimize the electronic net length and gain an optimal signal to noise ratio. The deflection of the laser beam is measured by a 4-quadrant detector. The electronic design and layout of the printed circuit board can be found in appendix D. Mechanical drawings of the nano cartographer are included in appendix B.

An extension of the nanocartographer, constructed to measure evanescent fields on optical waveguides, has been developed in a co works at the institute [HAH08]. Data acquisition in nearfield mode also confirms the quality of the instrument design.

4 Specification : chrome coating on glass substrate, layer thickness 150 nm, line spacing 1 micron, fabricated by means of a laser lithography.

5 The z -coordinates of the translation mimic and the scanner are frozen. Orthonormal stimuli are applied to the scanner open loop. A schematic draw of the experimental setup is illustrated on page 69.

2.1 Micromechanical System Components

The optomechanical toolbox offers a turnkey solution for linear positioning tasks. Most instrumentations are based on translation stages with built in linear tracks, whereas the seamless motion is achieved through roll or ball bearing units. The stages are equipped with stepping or servo drives and the relative position of the slider to the base mount is measured by rotary or linear encoders. Common encoding techniques use optical beats and Hall sensors for offset calibration. Three dimensional coordinate based motion is proposed by orthogonal cross mounting of the stages [VOR01]. Applications in micro machining, wafer handling, machine tool inspection and the wide range of optical research [FLU96] show, that this is a reliable and precise solution as long as the length scale of interest is not too much below one micron. But what if we tend to move the sliders on a nanometer scale? Will accuracy be sufficient? Is mechanical play, backlash, creep and thermal drift acceptable? This section will give an answer by construction, setup and experimental validation.

Fig. 2.1.1 depicts a prototype of the nanocartographer. The linear stages of the micromechanical translation unit are mounted on a block of solid aluminum. Sliders are driven by servo motors fixed to the stage sockets. The long range traveling x–slider employs recirculating ball screws for zero–backlash motion. Short range navigation on the z– and y–axis utilizes creep optimized roll bearings but is affected from a minor mechanical play. Backlash is measured in the instrument calibration and stored for the compensation of coordinate actuation errors. The drives support a rapid change

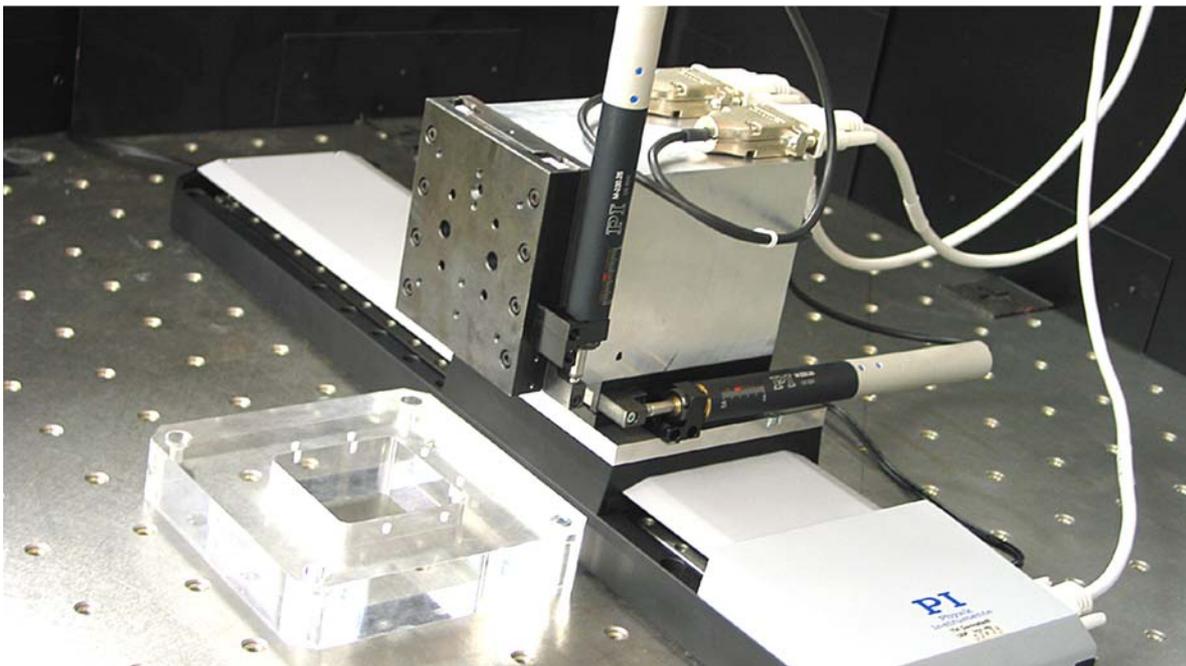


Figure 2.1.1 : *Nanocartographer 3–axis navigation based on cross mounted linear translation stages. The AFM is to be screwed to the vertical slider and the acrylic body marks the designated position of the scanner.*

between the depicted measurement and other service positions typically faster than 10 seconds. The nominal travel range in x–direction is 300 mm, whereas z– respectively y–direction is limited to 25 mm. Notice gravity to act as a position independent counteracting force to shift the z–slider. All *drives* guarantee a design resolution of 100 nm. Thermal drift in the solid components of such a system is an issue, as the following calculation advises. The coefficients for linear thermal expansion of stainless steel and aluminum are [KUC96]:

$$\text{Fe} \quad \alpha = 12.0 \cdot 10^{-6} / K \quad \text{Al} \quad \alpha = 23.2 \cdot 10^{-6} / K \quad (2.1)$$

Length expansion follows the well known formula:

$$l = l_0 \cdot (1 + \alpha \cdot \Delta T) \quad (2.2)$$

The design length of the sliders is $l_0 = 104.00 \pm 0.01 \text{ mm}$. A temperature drift of $\Delta T = 1K$ leads to an expansion:

$$\text{Fe} \quad \Delta l = 1.248 \mu\text{m} \pm 0.5 \text{ nm} \quad \text{Al} \quad \Delta l = 2.413 \mu\text{m} \pm 1.0 \text{ nm} \quad (2.3)$$

So if we combine both materials in one construct we might expect an elongation difference in the range of $1.165 \mu\text{m} \pm 1.1 \text{ nm}$. As it is highly intended to minimize thermal drift the selection of one single bulk material advises. The z– and y–sliders shown in fig. 2.1.1 are steel made in the prototype status, whereas the final version of the nanocartographer employed in the spectral analysis experiments is manufactured from stress relieved *Al Zn₄ Mg₂ Mn* alloy.

Knowledge on position sensing technology in the servo drives is proprietary to the manufacturer¹. Except from the specification in the component data sheet little public information is available. The relative coordinate of the slider to the base mount is measured on x–axis with help of an exposed optical linear encoder. It consists of a glass scale and a free traveling scanning head that resolves the relative motion counting beats. A similar technique is used on other axis but the linear glass scale is replaced by a rotary one. Turns are proportional to the linear slider translation. Zero coordinate calibration is supported by means of the integrated Hall effect origin and limit switches.

The translation stage assembly is illustrated in a mechanical drawing² (see fig. 2.1.2). Bearing creep in the guide ways is suppressed through plastics gear wheels. A photograph of the linear rails shows the cage assist in detail³. The pinion design is primarily made to stabilize high accelerated slider motion which does not count here. But it also enhances the coordinate based travel of the nanocartographer, due to the additional guide forces. The rails are manufactured from stainless steel, however the small body size renders thermal length expansion negligible.

¹ PI Physikinstrumente : M-531, M-230

² annotation : the draft goes back to a design of Dr. F. Laeri at the Institute of Applied Physics from 1983, which has been modified to fit the nanocartographer's geometry.

³ Schneeberger formula–S rail system

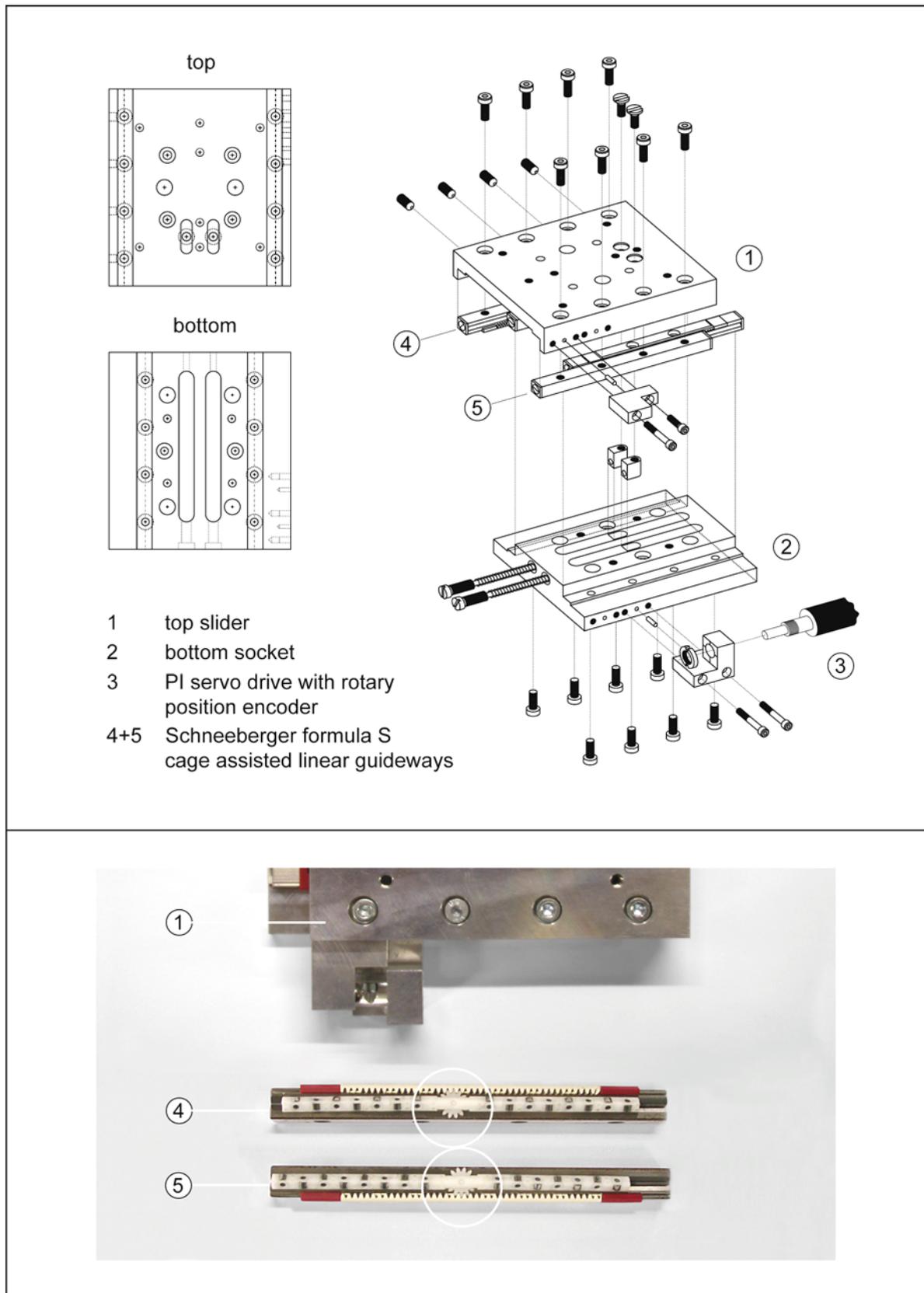


Figure 2.1.2 : Translation stage assembly. The photograph shows two cage assisted guide ways. The extension of the standard roll bearings minimizes creep forced dislocation of the sliders.

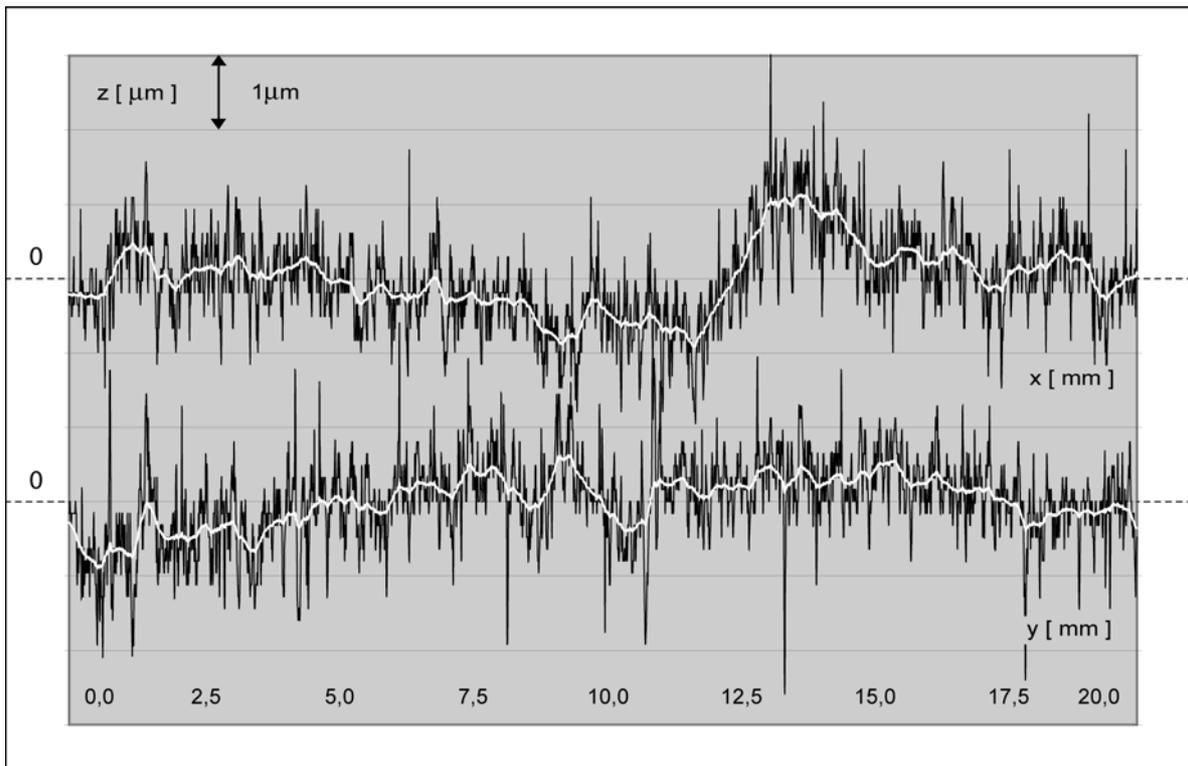


Figure 2.1.3 : Flatness curves of horizontal nanocartographer translation measured on a silicon wafer specimen. The white lines yield from exponential smoothing of the data set with a decay of 0.95 . The out of plane error is more developed in the long range translation stage on x-axis than in the short range one on y-axis due to different linear guiding technologies. Peak to valley height is smaller than 2 micron in the total area of 20 x 20 mm.

Assume the AFM microscope and the scanner installed ⁴. As we mount a flatness calibration standard [WIL05] on the scanner at rest ⁵, touch down with the microscope in closed loop contact and move the AFM across by means of the drives, flatness of the horizontal motion can be quantified through the topographical data. Fig. 2.1.3 shows the result for both x- and y- direction. The microscope head has been moved with a constant drive velocity of 20.0 $\mu\text{m}/\text{s}$ over a distance of 20.0 mm . The flatness on x-axis is determined as $1.5 \pm 0.4 \mu\text{m}$ and on y-axis as $2.0 \pm 0.4 \mu\text{m}$.

In order to quantify, how accurate the AFM moves to selected target coordinates, an optical grating is also scanned as described above. Traveling speed of line scans is controlled by the drive electronics. The topography is charted on an oscilloscope. Since the speed is constant, time is proportional to travel distance. Scans are performed at 1.0 $\mu\text{m}/\text{s}$ with the grating orientated parallel as well as orthogonal to the x-axis. Different line spacings ⁶ are used to omit imaging artifacts. The calibration data is gathered on a length of 10 grating periods and compared with the nominal coordinates as set by the computer. The mean distance of the lines between rising and falling edges respectively is evaluated. Moreover the root mean square error of the absolute

⁴ to get an idea look at fig. 2.2.2 in the next section

⁵ plasma cleaned 4 inch silicon wafer

⁶ 300 nm chrome layer on a glass substrate, nominal line spacing 1 μm and 2 μm

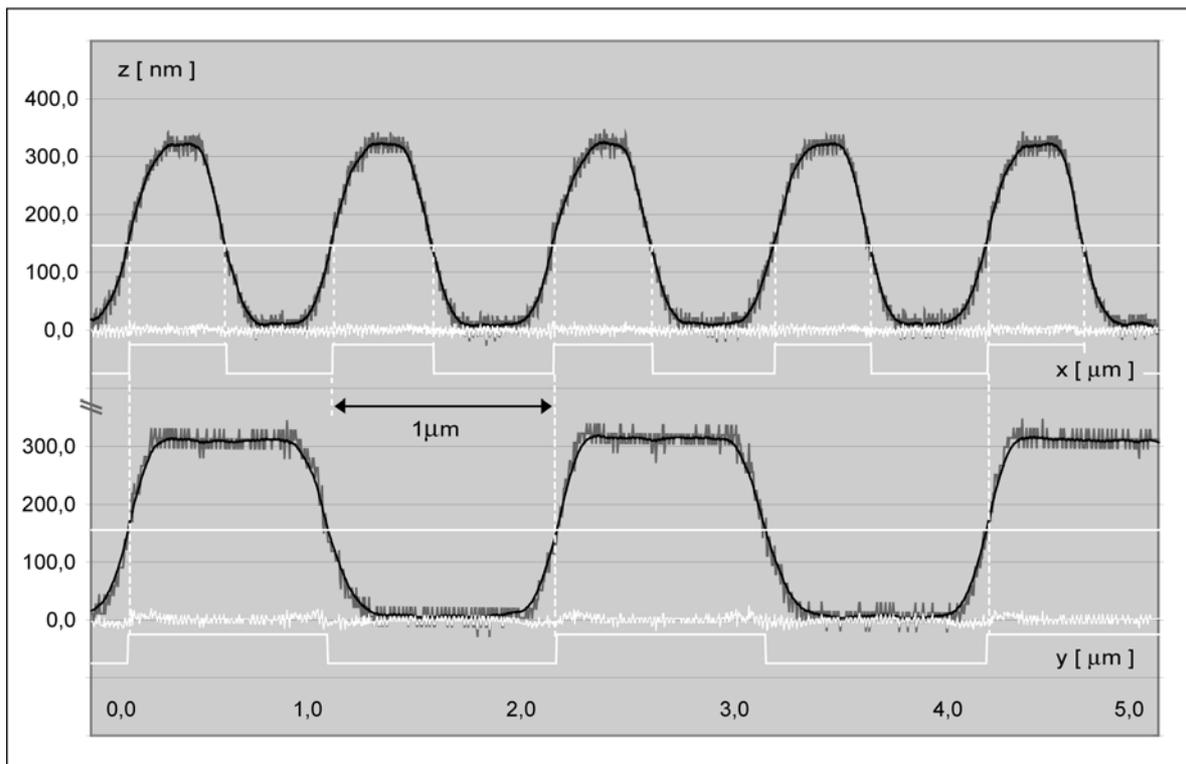


Figure 2.1.4 : *Nanocartographer virtual mesh characterization. The figure summarizes the line scans performed on different calibration standards. Optical gratings with 1 and 2 micron line spacing are scanned by means of the nanocartographer's linear translation mechanics. To demonstrate the perfect match the topography profiles have been aligned in the diagram with respect to the first rising edge.*

edge coordinates. Exemplary scans are plotted in fig. 2.1.4. As the raw data is exponentially smoothed by a weight of 0.95, profile crossings of the mean value indicate the location of the line edges. The recovery is indicated in the graphs by the added frames and the instrument characterization is based on said ones. Edges allow to state the mesh width. Moreover they act as an absolute position trigger, to check for travel linearity. The regular spacing measured in the computer generated virtual grid is $1.01 \pm 0.03 \mu\text{m rms}$ both in x- and y-direction. The position error of the line edge transitions is found at $\pm 0.05 \mu\text{m rms}$. This is more accurate than one can expect from the drive specifications. Detailed and comprehensive knowledge on the instrument performance is a prerequisite to feature the nanocartographer as a model system. Parameters are missing for backlash. To close the gap an optical interference pattern, exposed on a glass substrate in photo resist ⁷, is utilized for the characterization. The nominal line width is also 1 micron. This time the scanner moves the specimen. While AFM imaging, the microscope is shifted by the drives back and forth at an arbitrary instance in time for a nominal value of 200 nm. As illustrated in fig. 2.1.5 the image shows a significant parallel offset at the incident. A series of backlash parameters, from 1.0 to 3.0 μm with an increment of 100 nm, is tried to find the best realignment match. The experiments yield an optimum of $1.8 \pm 0.1 \mu\text{m}$.

⁷ 300 nm layer thickness

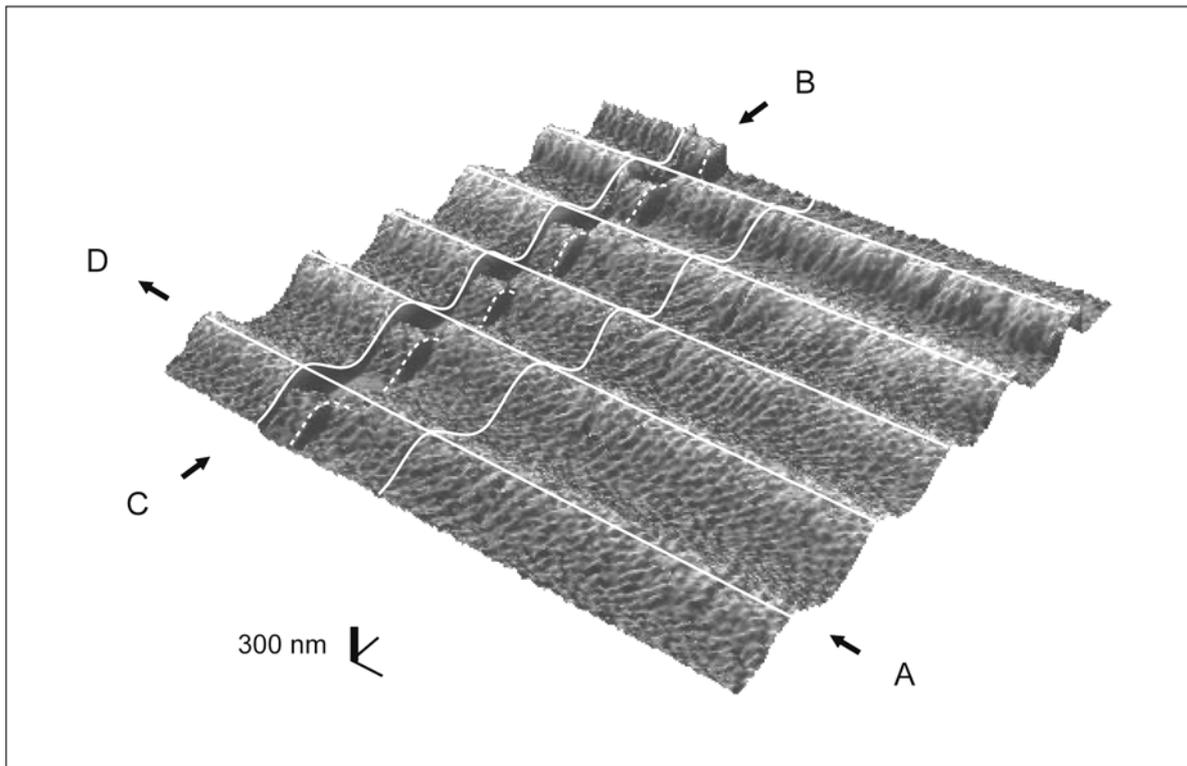


Figure 2.1.5 : Translation stage backlash characterization. White auxiliary lines are added to the AFM scanned image for an enhanced visualization of the manual realignment procedure. The image scan starts in A. At B a lateral shift of the AFM microscope head is applied. In C the shift is reverted testing several backlash parameters. The realignment match is finally judged at D, whereas the crosscorrelation of profile intersections before the shift and after realignment is evaluated.

The instrument analysis has proven that the tailored design makes the nano cartographer serve our needs well. But the present status of the project is to be seen as an intermediate step towards a true coordinate based metrology. The *Physikalisch Technische Bundesanstalt*⁸ started a European initiative for the development of guidelines and international standards for scanning probe calibration in 2006. The goal is to develop methods for the scientific and industrial community, to establish a traceability of the coordinate measurement to the SI unit meter. Latter shall incorporate system built in laser interferometry to guard the motion of the linear stages [WIL05]. The guideline has been published under VDI/VDE 2656 Part 1: 2008–05⁹.

In the next section the working principle of the AFM microscope head will be discussed. It rounds off the survey on the core components of the nanooptical model system.

⁸ German National Standards and Metrology Laboratory

⁹ VDI Verein Deutscher Ingenieure (Association of German Engineers), VDE Verband der Elektrotechnik, Elektronik und Informationstechnik e.V. (Association of Electrical Engineering, Electronics and Information Technology)

2.2 Nanooptics

Atomic force microscopy (AFM) is a common tool in nanoscale surface analysis. It employs silicon etched cantilever probes, with an attached sharp tip, to scan object surfaces [BIN86]. Common imaging techniques generate 3-dimensional maps of physical, chemical or electrical parameters in ambient conditions with a lateral resolution down to the nanometer and in ultrahigh vacuum in the sub nanometer range. A normal resolution near to picometers has been reported [GIS05].

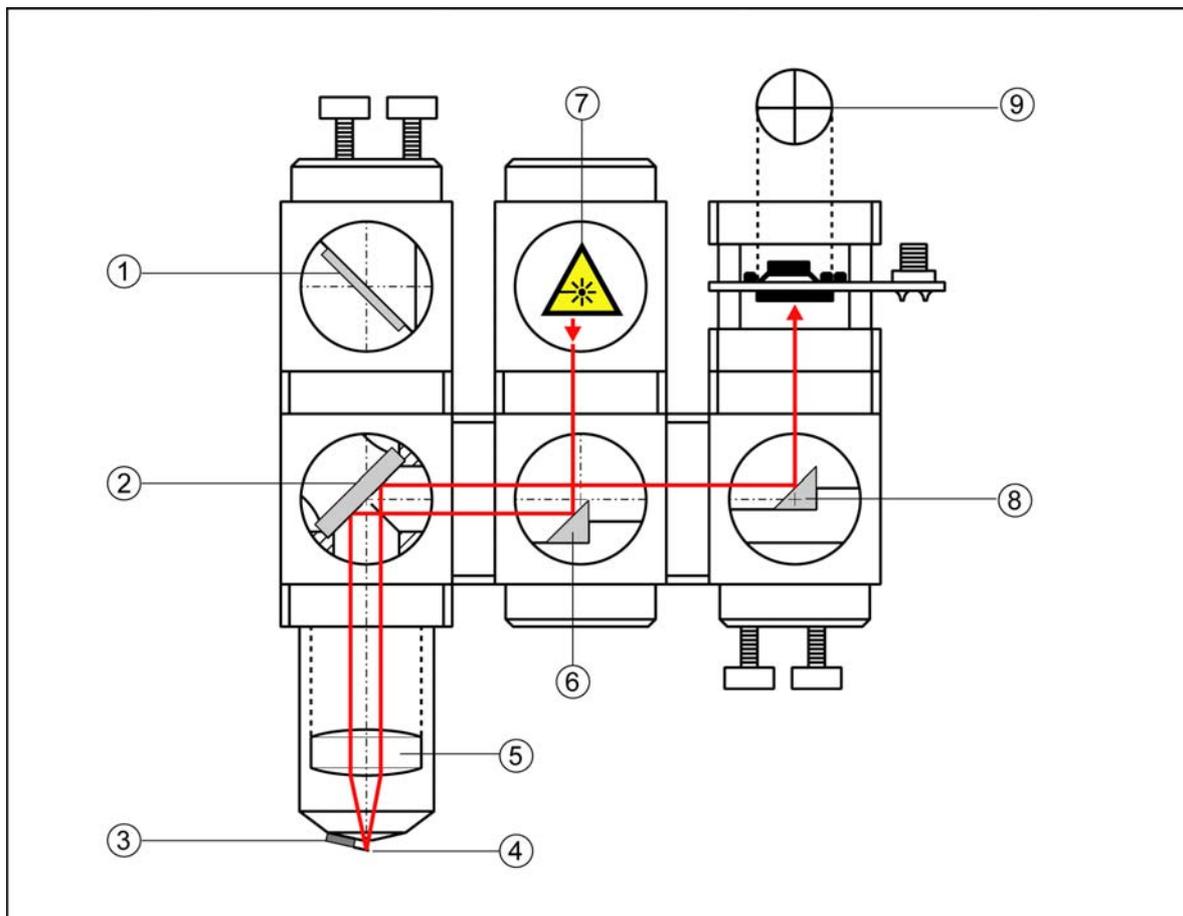


Figure 2.2.1: Optomechanical design of the nanocartographer's integrated AFM unit.

As illustrated in fig. 2.2.1 the tip-sample interaction forces are resolved by means of an infrared laser beam ⑦, that is focused on the cantilever ④. As the lever bends, the beam deflection changes, which can be detected with a 4-quadrant photo diode ¹ ①. High resolution techniques feature a harmonic stimulus of the cantilever, whereby the frequency detuning of the resonance, due to the interaction forces, is measured and evaluated for control purpose. The force sensor is fixed to the bottom of the microscope objective ² by a neodymium magnet ③. The beam is focused on the device traveling

¹ Hamamatsu S5980 , 5×5 mm active area, 30 micron gap.

² commercial optics from *WITEC*, Germany.

through an integrated collimation lens ⑤. The optical axis is folded by the infrared mirror³ ② and the gold plated prisms ⑥, ⑧, which yield an effective pointer length of $22.0 \pm 0.5 \text{ cm}$. The cantilever contact zone may be inspected through the aperture ① for probe and beam adjustment. Since the hot mirror ② is transparent in the visible band the optical interface allows a clear view through the microscope column. The mechanical construction of an AFM in particular requires a stable framework for the probe mount and the optics of the beam deflection unit. This is well supported by the microbench system from LINOS, that allows a very flexible setup of the optical path [KKO05]. The aluminum cubes are interconnected with buried steel rods to establish a high stiffness in the frame.

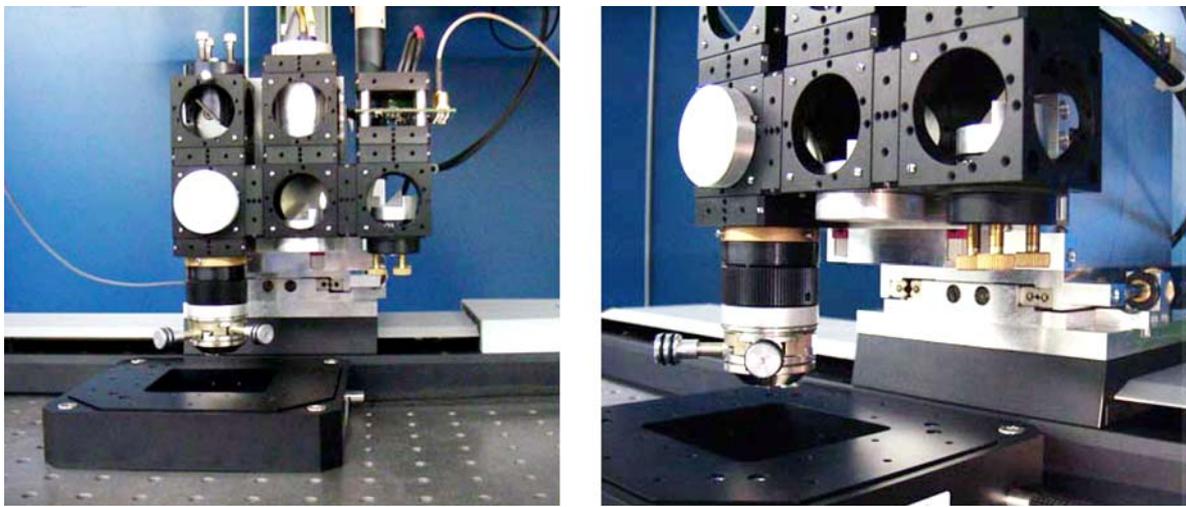


Figure 2.2.2 : Nanocartographer instrumentation. The AFM body, built from microbench optocubes, is mounted to the computer controlled 3-axis translation mimic. The microscope objective in the foreground carries a contact mode cantilever probe at the bottom end. The instrument is shown in cartography position centered to the piezo scanner. A vibration isolating optical table prevents acoustic coupling of the laboratory environment to the experimental setup. The optics design provides a tilt mirror inspection interface in the left top cube, which is aligned to the optical axis of an optional video camera.

For routine replacement the cantilevers are glued on metal washers. The sensors are clicked manually to the AFM measurement head as illustrated in fig. 2.2.3. To locate the lever precisely into the focus of the laser beam, the bottom end of the microscope objective is designed in form of an x-y-shift. Latter can be adjusted orthogonally to the optical axis with help of two built in micrometer drives. The width of the standard probes⁴ used in this work is $50.0 \mu\text{m}$ (a high resolution scanning electron-microscope picture of the tip is shown in fig. 4.1.14). The infrared laser diode⁵ operates at 820 nm wavelength and the beam is stopped down to a diameter of 2 mm . The focal length of the collimation optics is specified as 25 mm . Thus in accordance to the theory of Gaussian beams the focus spot has a minimal diameter of [EIC98] :

³ Edmund Industry Optics, IR pass band 700 nm, incident angle 45° , diameter 25 mm.

⁴ Nanosensors PointProbe, contact mode AFM imaging.

⁵ Laser Components, Flexpoint module, 8 mW

$$d_f = \frac{4}{\pi} \cdot \frac{\lambda \cdot f}{d_0} = \frac{4}{\pi} \cdot \frac{820nm \cdot 25mm}{2mm} = 13.1\mu m \quad (2.4)$$

Real beam diffraction broadens the spot. The visual inspection interface mentioned above allows to image the beam waist conveniently. A typical value of $20.0 \pm 5.0 \mu m$ is estimated from the video monitor. Since the sensitivity of the micrometer drives is nominal $1.0 \mu m$, the cantilever alignment with respect to the focus is no problem. The change of a cantilever probe can be done in less than 30 seconds, after the triple axis translation mimic has driven the AFM into a service position. Standard tweezers are used to set the sensor into the correct position.

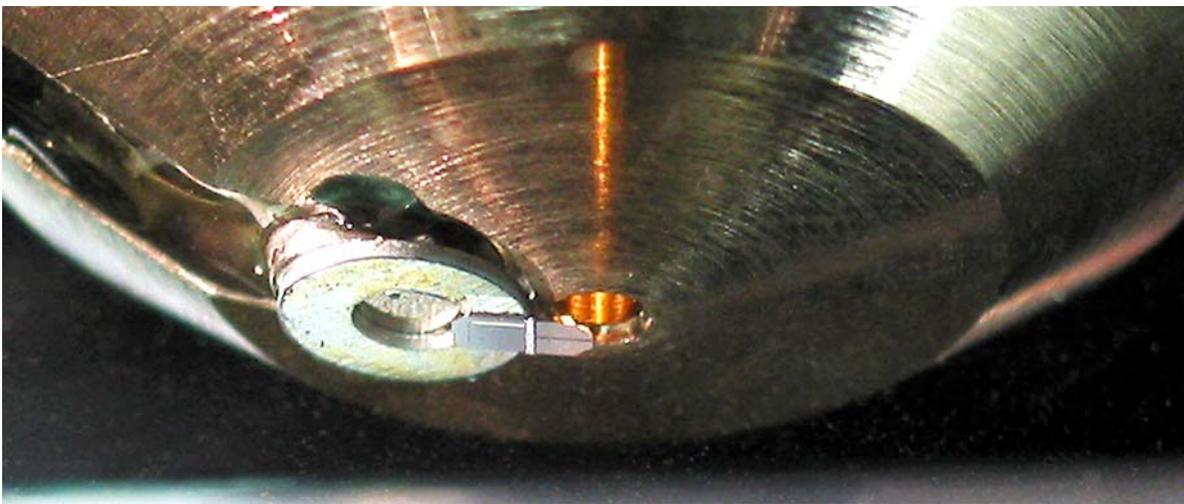


Figure 2.2.3 : *Cantilever Mount.* The silicon chip sensor is attached to a metal washer, that is magnetically fixed to the front end of the AFM microscope objective. The cantilever tip is aligned to the optical axis of the instrument, which is set off by an illumination of the microscope column.

To conclude the section the nanocartographer's AFM unit is qualified for constant height imaging, which is the preferred mode of operation for the resonance experiments reported later in the thesis. In this mode the cantilever tip is brought in open loop surface contact. The object is moved horizontally or vertically by the piezo scanner and the ups and downs of the cantilever are recorded through the beam deflection unit. We memorize, that the characterization of the micromechanical translation stages in the previous section gave evidence, that the built in drives do navigate the probe relative to the object with an accuracy of about 100nm. But it has to be proven that the position of the tip is kept stable over a minimum of 100 seconds as the stages are locked. This is the specification for the spectrum analysis experiments ahead. We image an artificial opal film on a silicon wafer substrate in constant height mode within 120s to substantiate, that the instrument is competitive. The film is fabricated by means of spin coating the wafer with a dispersion of SiO_2 monospheres. A scanning electron microscope (SEM) image of the nanostructure is shown in fig. 2.2.4 and the AFM image is presented in fig. 2.2.5. Sphere diameter is nominal 200nm and mean film thickness $2.5 \mu m$.

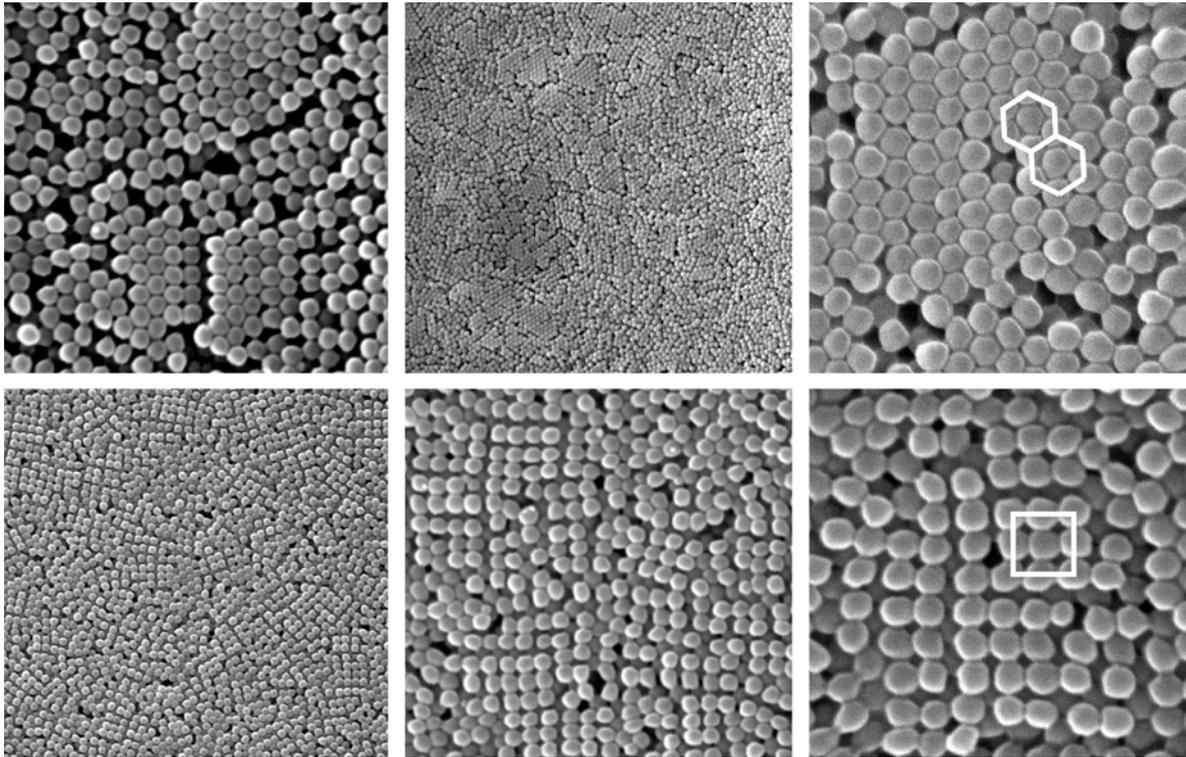


Figure 2.2.4 : SEM image of an artificial opal film taken at the Institute of Applied Physics, TU Darmstadt. The standard mu-bar is omitted in the multi zoom images, since the diameter of the spheres acts as a precise scale of length (specified by the manufacturer as $200 \pm 10\text{nm}$). The spheres arrange in different characteristic domain patterns.

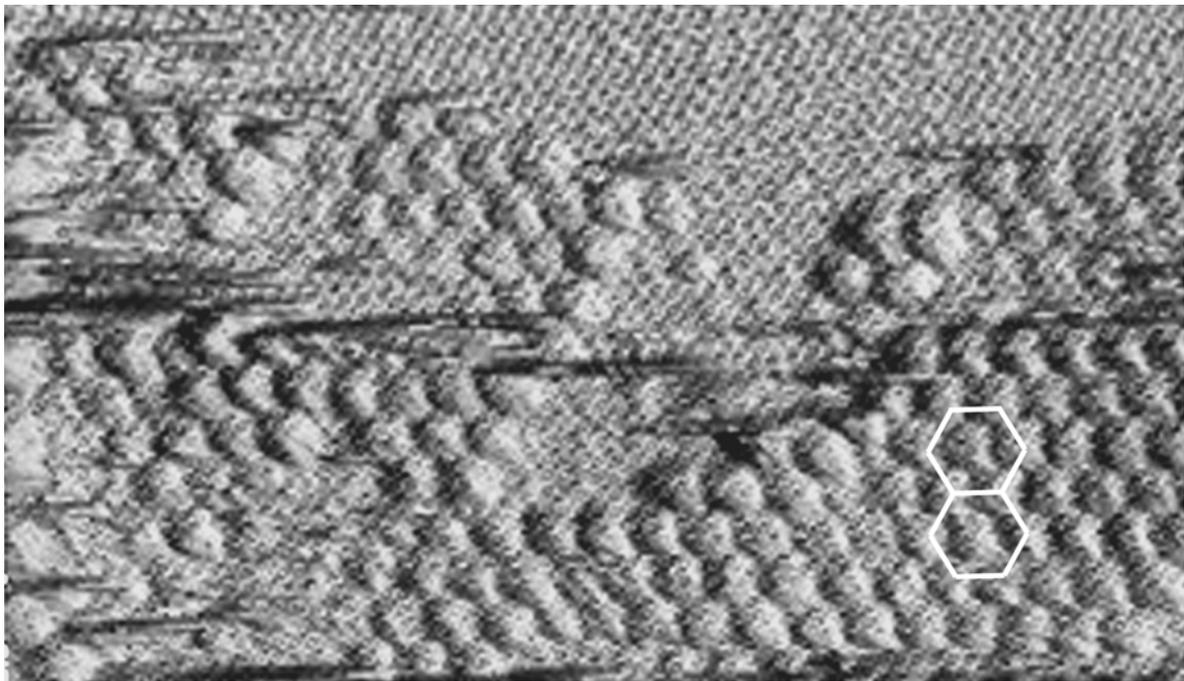


Figure 2.2.5: AFM image of the same film, scanned in constant height mode. The tip loses the surface contact at the domain border. The adhesive forces due to triboelectricity are strong enough to prevent the spheres' structure from being scratched.

3 Adaptive Methods for Spectrum Analysis

The knowledge of the *complex dynamics* of a nanomechanical device [SCH02] [SAL02] allows optimized motion control. It needs modern digital signal processing techniques to measure the key parameters of a dynamic system model, but control utilizing these parameters offers high performance and flexibility [SKO01, ch.9]. Thus an implementation of the observer, the controller design and the system control loop on a single embedded platform is desired. In that case the computational tasks run in parallel. Such a device characterization is named *online system identification*. This chapter is dedicated to a review of available adaptive identification methods that allow to measure spectra online. One of the methods is significantly improved by a detailed mathematical examination with respect to the computational burden such that it supports the spectrum analysis of high order systems.

The complex dynamics can be described by a *transfer function*. It allows to predict the actuation in case of a known driving signal. If a device is one-dimensional¹, it is called a Single-Input-Single-Output (SISO) system [SKO01, p.160]. Moreover if it obeys the superposition principle² it is named linear [SKO01, p.113]. In terms of linear system theory the mechanics performs a filter operation [KAI80]. It follows for all actuators that are a linear SISO system an appropriate model is a digital filter, whose transfer function approximates the physical one of the device. *Nanomechanical actuators* embedded in a scanning probe microscope are traditionally based on piezoelectric motion [PAE96, p.103]. An external electric field induces a stress inside the solid that leads to a reversible elongation of the ceramics on a nanometer scale. The piezoelectric devices are either used standalone or encapsulated in a metal housing. The actuators match the criterion to be a SISO system. Although it is known that piezo-driven actuation is highly nonlinear it has been shown by Daniele et al. [DAN97] that the nonlinear behavior of piezoelectric materials such as hysteresis, creep and coupling between various motion axes can be neglected in a first step for small amplitude variations. So the actuators may be approximated by a *linear system model* and therefore by a digital filter. This assumption is clearly supported by the experimental data gathered here (see chapter 4). The use of finite impulse response (FIR) filter kernels [SMI99] implies linearity so they are an educated model choice.

Adaptive Spectrum Analysis (ASA) means tuning the kernel of a digital filter online, such that it reproduces the transfer function of the linear system under observation³. The adaptation rule is selected to optimize the degree of spectral correspondence. It follows from a quantitative merit criterion which is continuously evaluated in the characterization experiment. Signal processing literature on adaptive filtering covers two main streams [HAY02] : evaluation based on measurement or otherwise founded in statistical expectations. The first approach is called *deterministic* whereas the

1 i.e. it has a single input and output channel

2 homogeneity and additivity

3 pertaining linear systems adaptation is always convergent [HOF00], however convergence rate and error are sensitive to a proper kernel dimension

second one is known as *stochastic* merit. As we will see in chapter 3.1 both lead to a set of almost identical computations though the mathematical background is completely different. The ideas of the deterministic statement are intuitive and go back to *C.F. Gauss*. They are very popular in physics. We discover the common bullets in the discussion line to gain a better understanding of the stochastic method. It turns out that in case of high dimensional spectral analysis both adaptive filter algorithms are too comprehensive to be computed on commercially available digital signal processors. Studies at low orders however reveal, that the stochastic statement exclusively leads to quasi-stationary filter conditions which allow for a tremendous reduction of the numerical effort. The coordinates of the adaptation gain show a periodical course with characteristic magnitudes and phasing. Instead of computing the gain according to the filter equations, it can be emulated. This insight is the major contribution of the present work, since it paves the way for analyzing spectra containing several thousand lines in an online experiment. Section 3.2 is dedicated to the close mathematical study of the stationary state and the development of an adaptive filter variant that provides said online capability.

Two recording techniques that feature the new developed algorithms will be compared, based on experiments and simulations, in the follow up chapter 4. Both measure the mechanical transfer function of the nanooptical model system introduced in chapter 2, which is equivalent to an identification of the digital filter kernel that fits best [SMI99]. One technique is the acquisition of resonance curves by means of frequency wobbling. The mechanical resonance is irradiated on a single complex line. While scanning the system *sequentially* in frequency space the response is tracked by an adaptive filter. In that case the length of the kernel is 2 since we treat real and imaginary part in the complex plane. The other technique drives the system on a broadband frequency comb. The spectral response is recorded *in parallel* on all lines of the comb. Now the kernel size is 4096 which means 2048 lines are involved. Since all dimensions of the stimulus superpose to a single analog system input the SISO criterion still holds. The test structure will be the mechanical housing of a commercial piezo actuator⁴. The results of the wobble experiment and the broadband adaptation equally approve concept of ASA filtering. The multiple line application in particular acts as a thesis milestone.

The source code for the new ASA application is supplied in appendix A. The appendix lists the assembly for some basic mathematical routines. Further the initialization of the signal processing board is included. The reader will find a complete implementation of the stochastic statement and its minimal stationary form the ASA filter.

4 *piezosystem jena*

3.1 Adaptive Filtering Techniques

It has been mentioned before that the transfer function of an unknown linear system can be modeled as a digital filter. The characterization technique is known as adaptive filtering [HOF00, ch.1]. The *deterministic* form of system identification discussed here is the least square (LS) method. A recursive formulation of LS is known in modern signal processing literature as recursive least square (RLS) filtering [GOE02, ch.3 and ch.8]. The concept of RLS offers a fast convergence rate [HOF00, ch.6], so it seems to be an attractive candidate for system identification. But theoretical insight reveals it suffers from numerical instability while being executed online¹ which is related to the ill-posed nature of the underlying mathematical model [PAR92] whose singularities have to be regularized in some way. Though a lot of competitive regularization methods are known today that gain stability [BER99], all these sophisticated workaround scenarios share the minus of an added computational burden². This is a show stopper if we target online system identification. Ill-posedness however is off the topic in a *stochastic* statement of the identification problem. It avoids additional regularization effort through a tailored model choice. The system is characterized by a parameter vector in state space that performs a random walk with known statistics [HAY02, ch.10]. Instead of measuring the state as in the deterministic approach it is estimated by means of a try and error calculus. The ideas go back to *R.E. Kalman* who published the technique in a seminal paper 1960 [KAL60] which gave the algorithm its denomination Kalman filter. It solves the stability issues and takes benefit from a maximal convergence rate [MAY79].

To give an outline of the discussions: We will start first with a generalized view on the identification problem in section 3.1.1. The task to find an optimal fit to the system's transfer function will be formulated in terms of correlations. The linear transformation that fits best shall have the same input-output crosscorrelation as the system under observation. If the crosscorrelation is not directly measurable the method shall hold for the expectations³. This interpretation is innovative and natural in light of physics. We designate it the method of *conserving crosscorrelation*. Second we review the RLS scheme in that context. Evidence will be given in section 3.1.2 on the roots of instability. Third we apply the conservation rule to understand how the Kalman filter works and how it eliminates said drawback in section 3.1.3. Both RLS and Kalman filter theory is extensive. For ease of understanding only the mathematics relevant for the discussion of the conservation rule is included in the sections. The complete treatment is printed in appendix **B**.

Many descriptions of the RLS algorithm as well as the Kalman filter can be found in literature [BER99] [HOF00] [HAY02]. The notation is quite apparent neither uniform

1 An example code for the original version of the RLS algorithm has been published in [ANA94, ch.6]. The code is dated in the year 1991.

2 Annotation : It is well known that major extensions have to be added to the RLS code (e.g. exponential forgetting and resetting algorithm, EFRA [SAL88]) to guarantee numerical stability. The focus of this work is laid on the characterization of piezomechanical systems which typically show resonance profiles up to 2.5 kHz bandwidth [VOR01]. According to the Nyquist theorem [GRUE01] a DSP data throughput of 5k samples is just compliant. DSPs do not meet the specifications to run the EFRA algorithm online at high filter orders.

3 i.e. in the mean sense provided statistics known

nor didactically edited. The thesis edition will strictly follow the common $\langle bra|ket \rangle$ formalism. Notation is easy to read and allows to reveal a parallelism to familiar concepts in physics which is an excellent start for a generalization of the method. Motivation arises from a close relation of adaptive filtering to mathematical techniques used in modern quantum mechanics for modeling complex microscopic systems (quantum state estimation and control [BEL05] [GER03]). As far as the author is aware this has not been made explicit for both RLS and discrete time Kalman filter yet.

3.1.1 The Key Role of Crosscorrelation in Identification Problems

Fig. 3.1.1 depicts a block diagram of the system identification signal flow. Both adaptive filter and unknown SISO system are connected to a driving signal. The system response ⁴ and the filter output are compared. The adaptation attempts to minimize the identification error. In case of convergence the transfer function of the unknown system is approximated by the filter kernel. Notice the arrows in the block diagram indicate causality. The experimental setup is partitioned into subsystems with causal dependence. We stimulate and assume the response to be caused by the stimulus only. External distortions are not part of the standard model and assumed to be blocked by the experimental setup, for instance by a vibration isolating optical table ⁵.

If we implement this scenario on a digital signal processor, we have to digitize two analog signals:

- the stimulus (this is the adaptive filter input data stream, if we generate the stimulus by the DSP itself latter is optional)
- the system response (we want to compare that signal to the adaptive filter output data stream)

Stimulus has to be synthesized such that the adaptation is driven at all relevant SISO resonance frequencies. As mentioned in footnote ² the bandwidth of the digitizer must comply with the Nyquist theorem. Theoretically the minimal sampling frequency of the DSP system must be twice the highest resonance frequency. A printed circuit board *design proposal* is given in appendix D that supports a perfect match ⁶. The DSP platform ships with two discrete analog input and output channels at 16 bit resolution and the sampling frequency can be set from 1Hz up to 500kHz. In case of a broadband stimulus the adaptation is naturally driven on all dominating frequencies thus we gain minimal convergence time and parameter misadjustment. Consider the strong symbiosis between the system identification experiment and the DSP computations.

⁴ piezo actuators typically provide the response information by position sensor voltages [VOR01]

⁵ We will see later in the discussion this is not mandatory. Uncorrelated distortions do not count.

⁶ The pcb design study is based on the ADDS-21160 EZKIT LITE evaluation board rev. 2.3 which has been used for application development and experimental proof of concept in this work. The evaluation board is optimized for audio signal processing and does not support variable sample rates (the sampling frequency of the analog front end is fixed at 48kHz, specification and technical data see AD1881A data sheet). The analog input and output subsystem has been redesigned embedding two discrete AD7671 analog-digital and AD5541 digital-analog converters.

The microcomputer may operate as an embedded part of the identification loop as well as a system model in a simulation. This implies, that the adaptive filter in principle must feature a degree of freedom that is comparable to that one of the observed system. It has to provide adequate dynamical bandwidth and resolution. Otherwise the tracking process will fail due to missing correlation. To give an illustrating example: A piezo actuator typically oscillates over several thousand sampling periods after stimulation by a delta-pulse. We remember that an FIR filter kernel equals its digitized impulse response within a limited time window [GRUE01]. Thus the decay time of the actuator motion roughly determines the necessary kernel size. It is possible to identify such a system based on a high order FIR filter kernel.⁷ But if the computations are too demanding and we clip the kernel, the long term oscillations are stripped and the adaptive filter tries to compensate for that continuously modifying the higher modes. Due to missing correlation the compensation fails and the identification will fall rather poor⁸.

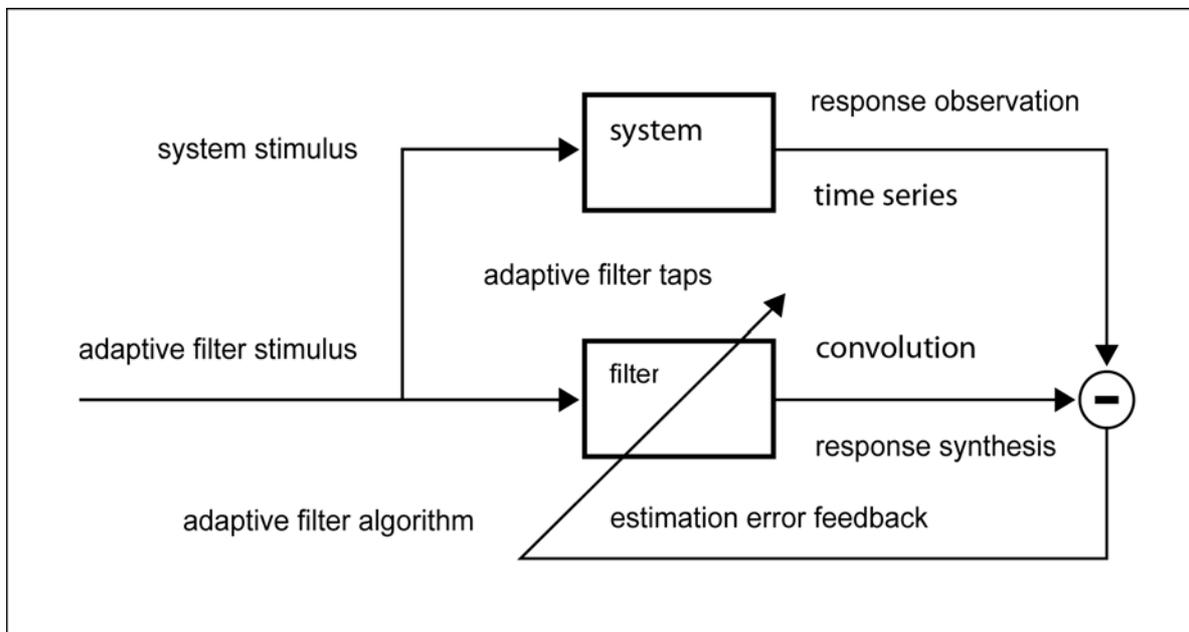


Figure 3.1.1 : Principle of system identification by adaptive filtering.

Now that we talked about the technical issues we turn to the fundamental ones. The notation of the mathematical equations ahead follows the scheme depicted in fig. 3.1.2. Assume a descriptive linear regression model in form of

$$\mathbf{C}^* = \mathbf{B}^* \cdot \mathbf{A} \quad (3.1)$$

whereby \mathbf{C}^* is the prediction of a given endogenous operator \mathbf{C} . \mathbf{B}^* is a linear transfer and \mathbf{A} is a known exogenous operator. Then

$$\mathbf{D}^* = \mathbf{C}^* \mathbf{A}^+ = \mathbf{B}^* \cdot \mathbf{A} \mathbf{A}^+ \quad (3.2)$$

⁷ assumptions: the dominant resonance frequency lies at 1kHz, the actuator oscillation lasts 100 periods and the sampling frequency is set to 50 kHz → the order of an adequate FIR filter kernel is 5000

⁸ infinite impulse IIR as well as autoregressive moving average ARMA kernels are not very helpful because of their known sensitivity to self oscillations

is the crosscorrelation between the predictand C^* and the predictor A .

We assume the system under observation to fulfill

$$C = B \cdot A \tag{3.3}$$

Then

$$D = C A^+ \tag{3.4}$$

is the crosscorrelation between the reference operator C and the system input operator A . The context is illustrated in fig. 3.1.2. To gain best model fit we we choose *conserving crosscorrelation* as our favorite method and yield

$$D = D^* \tag{3.5}$$

which is equivalent to $C A^+ = B \cdot A A^+$ (3.6)

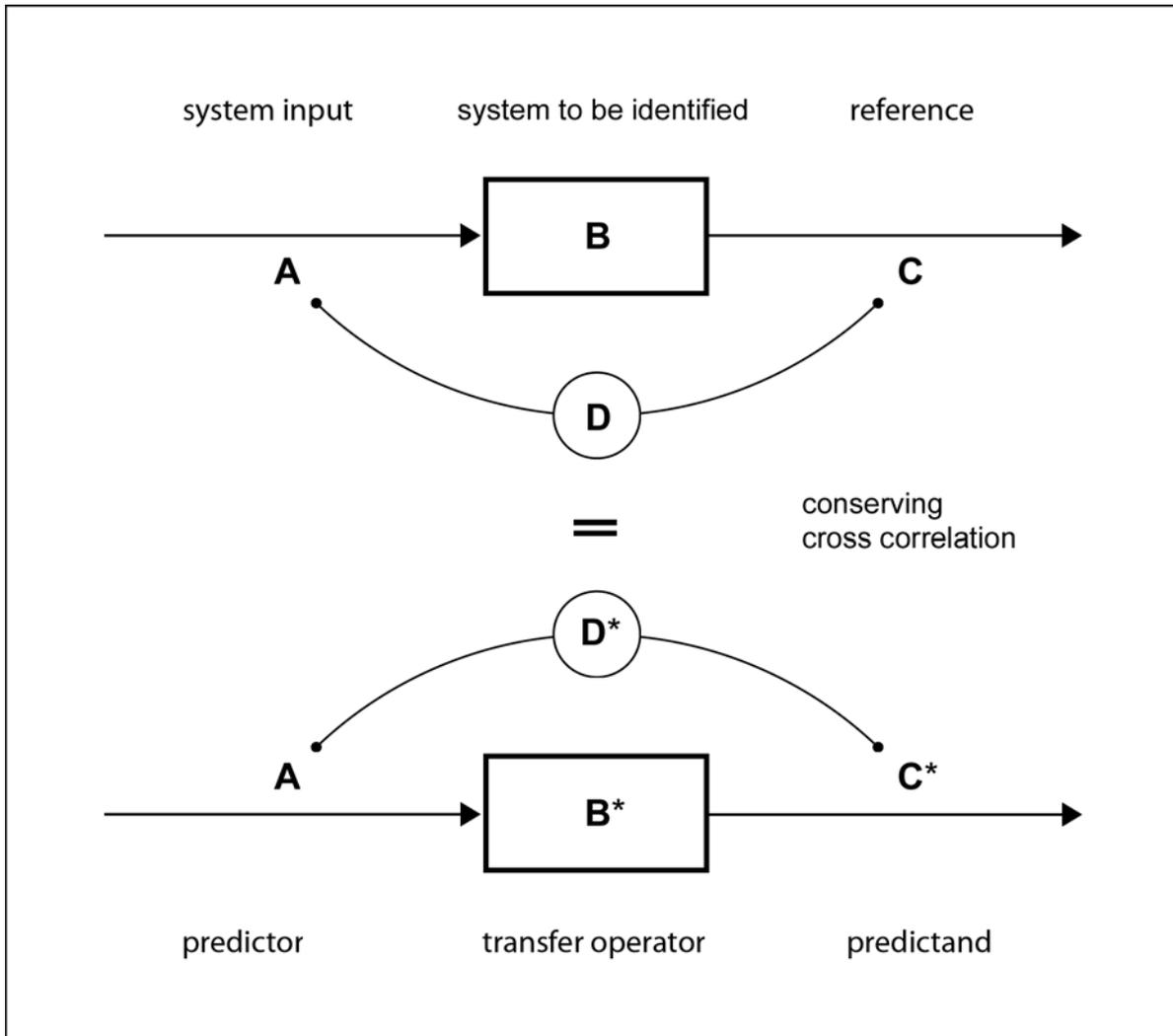


Figure 3.1.2 : System identification conserving input–output crosscorrelation.

Thus if the inverse exists the solution to the identification problem is given by

$$\mathbf{B} = \mathbf{C} \mathbf{A}^+ \cdot (\mathbf{A} \mathbf{A}^+)^{-1} \quad (3.7)$$

If a *deterministic* approach is not applicable (maybe we cannot measure the reference but we have knowledge about the statistics of the system) the *stochastic* variant of equations 3.2 ... 3.7 is

$$(3.2) \rightsquigarrow E(\mathbf{D}^*) = \mathbf{B}^* \cdot E(\mathbf{A} \mathbf{A}^+)$$

$$(3.4) \rightsquigarrow E(\mathbf{D}) = E(\mathbf{C} \mathbf{A}^+)$$

$$(3.5) \rightsquigarrow E(\mathbf{D}) = E(\mathbf{D}^*)$$

provided that the inverse exists

$$(3.6) \rightsquigarrow \mathbf{B} = E(\mathbf{C} \mathbf{A}^+) \cdot E(\mathbf{A} \mathbf{A}^+)^{-1} \quad (3.8)$$

In the following subsections we utilize equations 3.7 and 3.8 respectively to formulate algorithms that solve the identification problem for the deterministic and the stochastic statement.

3.1.2 Least Square Adaptive Filtering Reviewed

The impulse response of any digital filter is measurable by a computer experiment ⁹. If the system response is also measurable we can record it and derive the kernel in a straight forward manner [SMI99]. But not all systems must be probed by a delta pulse. Moreover if we tend to examine the small signal regime the response might be buried in noise. Thus a system identification technique that deploys adaptive filtering is commonly based on broadband spectral irradiation. As a matter of course real life laboratory experiments demand for the choice of a meaningful merit criterion that quantifies the degree of spectral correspondence between the system and the digital filter. In case of the *Least Square* method (LS) the merit is given by the sum of squared prediction errors, which the adaptation tries to minimize. The next section introduces briefly the mathematics behind the LS method. It is demonstrated that an adaptation *conserving crosscorrelation* indeed fulfills the LS criterion.

The most simple digital filter we might employ is an FIR filter. ¹⁰ Using vector notation

⁹ Assume a δ -pulse of magnitude 1 to propagate through an FIR filter structure. The pulse sequentially selects the weighted filter taps $w(0), w(1), \dots, w(N-1)$ to generate the time series at the output of the filter stage (see eq. 3.11). The weighted tap vector equals the impulse response function.

¹⁰ for more information on FIR filters see [ANA94 , ch.6].

¹¹ the symbol $[\]^T$ means this is a transposed vector, i.e. $|x\rangle$ as well as $|w\rangle$ are matrix columns

$$|x(n)\rangle = [x(n), x(n-1), x(n-2) \dots x(n-N+1)]^T \tag{3.9}$$

$$|w(n)\rangle = [w_1(n), w_2(n), w_3(n) \dots w_N(n)]^T \tag{3.10}$$

it performs the following mathematical operation:

$$y(n) = \langle x(n) | w(n) \rangle \tag{3.11}$$

whereas w_k are called the filter taps. In case of FIR filtering the output is the scalar product of the input data vector $|x\rangle$ and the tap vector $|w\rangle$. The taps equal the digitized *filter kernel* within a limited window of size N [GRUE01]. N is called the filter length. With respect to fig. 3.1.1 the samples x represent the stimulus and y the prediction of the system response.

If we denote the system response as r the prediction $r^* = y$ equals

$$r^*(n) = \langle x(n) | w(n) \rangle \tag{3.12}$$

With $x(n) \equiv 0 \ \forall \ n \leq 0$ the evolution of the response prediction is given in short form

$$|r^*(n)\rangle = \mathbf{X}(n) \cdot |w(n)\rangle \tag{3.13}$$

and in explicit form

$$\begin{array}{c}
 \begin{bmatrix} r^*(n) \\ \vdots \\ r^*(N) \\ \vdots \\ r^*(3) \\ r^*(2) \\ r^*(1) \end{bmatrix} = \begin{bmatrix} x(n) & x(n-1) & x(n-2) & \dots & x(n-N+1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x(N) & x(N-1) & x(N-2) & \dots & x(1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x(3) & x(2) & x(1) & \dots & 0 \\ x(2) & x(1) & 0 & \dots & 0 \\ x(1) & 0 & 0 & \dots & 0 \end{bmatrix} \cdot \begin{bmatrix} w_1(n) \\ w_2(n) \\ \vdots \\ w_N(n) \end{bmatrix} \\
 |r^*(n)\rangle \qquad \qquad \qquad \mathbf{X}(n) \qquad \qquad \qquad |w(n)\rangle
 \end{array}$$

where $|r^*(n)\rangle$ is an n -dimensional vector
 $\mathbf{X}(n)$ is an $n \times N$ matrix
 $|w(n)\rangle$ is an N -dimensional vector (3.14)

The assignment (compare fig. 3.1.2)

$$\begin{array}{lclclcl}
 \mathbf{X}^+(n) & \rightsquigarrow & \mathbf{A} & & \langle w(n)| & \rightsquigarrow & \mathbf{B} \\
 \langle r(n)| & \rightsquigarrow & \mathbf{C} & & \langle r^*(n)| & \rightsquigarrow & \mathbf{C}^*
 \end{array} \tag{3.15}$$

allows to find the filter kernel according to equation 3.7

$$|w(n)\rangle = \left(\mathbf{X}^+(n) \mathbf{X}(n) \right)^{-1} \cdot \mathbf{X}^+(n) |r(n)\rangle \quad (3.16)$$

Using the substitutes

$$\mathbf{Z}(n) = \mathbf{X}^+(n) \mathbf{X}(n) \quad \text{and} \quad |b(n)\rangle = \mathbf{X}^+(n) |r(n)\rangle \quad (3.17)$$

yields
$$|w(n)\rangle = \mathbf{Z}^{-1}(n) \cdot |b(n)\rangle \quad (3.18)$$

We will proof now that the solution given in (3.18) minimizes the sum of squared prediction errors. The error vector is calculated with help of equation 3.13 :

$$|e\rangle = |r\rangle - |r^*\rangle = |r\rangle - \mathbf{X} \cdot |w\rangle \quad (3.19)$$

The sum of squared errors equals the norm

$$\begin{aligned} \|e\| &= \langle e|e\rangle \\ &= \langle r - w \cdot \mathbf{X}^+ | r - \mathbf{X} \cdot w \rangle \\ &= \langle r|r\rangle - \langle r|\mathbf{X} \cdot w\rangle - \langle w \cdot \mathbf{X}^+ | r\rangle + \langle w \cdot \mathbf{X}^+ | \mathbf{X} \cdot w \rangle \\ &= \langle r|r\rangle - 2 \cdot \langle w | \mathbf{X}^+ \cdot r \rangle + \langle w | \mathbf{X}^+ \mathbf{X} \cdot w \rangle \end{aligned}$$

which leads to

$$(3.17) \rightsquigarrow \|e\| = \langle r|r\rangle - 2 \cdot \langle w|b\rangle + \langle w|\mathbf{Z} \cdot w\rangle \quad (3.20)$$

Assume we compare the error norm of two different estimates $|w_0\rangle$ and $|w_1\rangle$.

$$\begin{aligned} \|e_0\| &= \langle r|r\rangle - 2 \cdot \langle w_0|b\rangle + \langle w_0|\mathbf{Z} \cdot w_0\rangle \\ \|e_1\| &= \langle r|r\rangle - 2 \cdot \langle w_1|b\rangle + \langle w_1|\mathbf{Z} \cdot w_1\rangle \end{aligned}$$

$$\begin{aligned} \|e_1\| - \|e_0\| &= -2 \cdot \langle w_1|b\rangle + \langle w_1|\mathbf{Z} \cdot w_1\rangle \\ &\quad + 2 \cdot \langle w_0|b\rangle - \langle w_0|\mathbf{Z} \cdot w_0\rangle \end{aligned} \quad (3.21)$$

If we choose $|w_0\rangle$ according to (3.18) then

$$|b\rangle = \mathbf{Z} \cdot |w_0\rangle \tag{3.22}$$

which simplifies (3.21)

$$\begin{aligned} \|e_1\| - \|e_0\| &= \langle w_0 | \mathbf{Z} \cdot w_0 \rangle - 2 \cdot \langle w_1 | \mathbf{Z} \cdot w_0 \rangle + \langle w_1 | \mathbf{Z} \cdot w_1 \rangle \\ &= \langle (w_0 - w_1) | \mathbf{Z} \cdot (w_0 - w_1) \rangle \\ &= \langle (w_0 - w_1) \cdot \mathbf{X}^+ | \mathbf{X} \cdot (w_0 - w_1) \rangle \geq 0 \end{aligned} \tag{3.23}$$

Thus we conclude $\|e_1\| \geq \|e_0\|$ (3.24)

and prove that $|w_0\rangle$ is the LS solution of the identification problem.

Equation 3.22 is called *deterministic normal equation*. The kernel $|w_0\rangle$ is named *least square optimal filter*. The operator \mathbf{Z} is the *deterministic autocorrelation* of the system input \mathbf{X} . The vector $|b\rangle$ represents the *deterministic crosscorrelation* between the system input and the output $|r\rangle$. From a computational point of view the identification problem reflects on the calculation of the inverse autocorrelation matrix. Straight forward inversion is not applicable for online signal processing due to the high computational expense [HOF00, ch.4]. A recursion algorithm for that purpose is derived in appendix B1. Table 3.1.1 gives a summary whereas the meaning of the quantities in use is :

$ w\rangle$	adaptive filter kernel	$ x\rangle$	input data vector
r	system response	$ f\rangle$	filtered data vector
$ g\rangle$	adaptation gain vector	σ	initialization parameter
e^*	a priori ¹² system identification error	\mathbf{Z}^{-1}	inverse autocorrelation matrix

With respect to fig. 3.1.1 the computations perform :

12 The terminology *a priori* is used to emphasize that the error e^* in cycle n results from a *prediction* based on the kernel as given in cycle $n - 1$. The *a posteriori* error e is given by the *evaluation* of the estimate in cycle n namely $e(n) = r(n) - \langle x(n) | w(n) \rangle$. Though the RLS recursion is based on the *a priori* error it should be noted that the method optimizes the *a posteriori* error.

Recursive Least Square Adaptive Filter

Initialize the algorithm by setting

$$|w(0)\rangle = |o\rangle, \quad \mathbf{Z}^{-1}(0) = \sigma^2 \cdot \mathbf{I}$$

and $\sigma = \begin{cases} \text{large constant for high system noise} \\ \text{small constant for low system noise} \end{cases}$

For each instance of time compute

$$e^*(n) = r(n) - \langle x(n) | w(n-1) \rangle \quad \textcircled{1}$$

$$|f(n)\rangle = \mathbf{Z}^{-1}(n-1) \cdot |x(n)\rangle \quad \textcircled{2}$$

$$|g(n)\rangle = \frac{|f(n)\rangle}{1 + \langle x(n) | f(n) \rangle} \quad \textcircled{3}$$

$$|w(n)\rangle = |w(n-1)\rangle + e^*(n) \cdot |g(n)\rangle \quad \textcircled{4}$$

and $\mathbf{Z}^{-1}(n) = \mathbf{Z}^{-1}(n-1) - |g(n)\rangle \langle f(n)| \quad \textcircled{5}$

Table 3.1.1: Reformulation of the RLS adaptive filter algorithm in $\langle \text{bra} | \text{ket} \rangle$ notation.

- ① digital filtering, whereby a finite impulse response filter $|w\rangle$ is excited by a known stimulus $|x\rangle$. The *a priori* estimation error e^* is calculated comparing the response prediction $|r^*\rangle$ with the system response $|r\rangle$
- ② time variation of the estimation error feedback gain, the filtered data vector
- ③ $|f\rangle$ acts as an intermediate of the gain $|g\rangle$
- ④ adaptation feedback loop, The kernel $|w\rangle$ is updated by an increment proportional to the *a priori* estimation error e^* and the time-varying feedback $|g\rangle$ (designating it as the gain vector)
- ⑤ recursive update of the inverse autocorrelation matrix \mathbf{Z}^{-1} , notice the analysis of the stimulus ② ③ and ⑤ to be independent of the kernel tuning.

We recognize from equation 3.17 that due to the special form of the recursion the autocorrelation matrix \mathbf{Z} grows infinitely, compelling the inverse \mathbf{Z}^{-1} to fade out. As a matter of course we find the value of $|f\rangle$ to decrease continuously¹³ so the gain vector $|g\rangle$ diminishes corollary¹⁴. This implies two major issues concerning numerical stability. First the feedback loop loses alertness in long term observation since the

¹³ see ②

¹⁴ see ③

tuning reaches a deadlock rooted in missing gain. Second the unavoidable presence of roundoff-errors leads to an accumulation of gain uncertainty. The adaptation may be severely affected by a prediction error blow up [HAY02, ch.9]. Regularization tools to protect the system identification from uncertainty windup are known [HAN01] but as we will see in the next subsection they simply can be omitted using a mathematically adequate strategy.

3.1.3 The Essence of Kalman Filtering

If we apply Kalman filter theory to treat least mean square estimation we can benefit from two aspects that are not contained in the RLS scheme. To be mentioned first *process noise* is explicitly modeled, which adds *native regularization terms* to the filter update ¹⁵. Second the *stochastic merit criterion* equips the algorithm with the power to track non stationary system parameters. We have seen that the RLS adaptive filter minimizes the system response prediction error which can be directly measured, therefore the diction deterministic. The Kalman filter does the same for the model prediction error. Latter is not observable, thus the statement holds for the expectations and we deal with statistics. The system behavior unfolds in two equations [HAY02, ch.10] :

the measurement equation

$$|r(n)\rangle = \underbrace{\mathbf{X}(n) \cdot |s(n)\rangle}_{\text{deterministic}} + \underbrace{|\rho(n)\rangle}_{\text{stochastic}} \tag{3.25}$$

and the process equation

$$|s(n)\rangle = \underbrace{\Phi(n) \cdot |s(n-1)\rangle}_{\text{deterministic}} + \underbrace{|\sigma(n)\rangle}_{\text{stochastic}} \tag{3.26}$$

Above notation is called state space modeling. The approach is familiar with respect to fig. 3.1.1. The state $|s\rangle$ is approximated by the filter kernel. The task is to find a state approximation that is optimal at a merit to be defined. Both response $|r\rangle$ and stimulus \mathbf{X} are measured. The state transition is supposed to constitute from a deterministic part modeled by the transition operator Φ and a zero-mean Gaussian random vector $|\sigma\rangle$. Moreover the measurement of the system response is supposed to be noisy which is covered by a random vector $|\rho\rangle$. The stochastic processes $|\sigma\rangle$ and $|\rho\rangle$ are assumed to be fully uncorrelated. In analogy to the RLS filter the update of the state estimation $|w\rangle$ is embedded in a response prediction error feedback loop. The prediction error yields from

$$|r^*(n)\rangle = \mathbf{X}(n) \cdot |w(n)\rangle \tag{3.27}$$

$$|e(n)\rangle = |r(n)\rangle - |r^*(n)\rangle \tag{3.28}$$

¹⁵ Numerical regularization provides stabilization methods that add proper side constraints to a computed solution. Theory advices [TIK77] how to find appropriate regularization parameters that give a certain weight to the constraints such that the approximated result fits well to the exact solution.

We are searching for an operator $\mathbf{\Delta}$ that transforms the response prediction error $|e\rangle$ back to a model correction $|\varepsilon^*\rangle$ compensating for the *a priori* model error $|\varepsilon\rangle$ with

$$|\varepsilon^*(n)\rangle = \mathbf{\Delta}(n) \cdot |e(n)\rangle \quad (3.29)$$

whereas

$$|\varepsilon(n)\rangle = |s(n)\rangle - |w(n)\rangle \quad (3.30)$$

Due to the correction the *a posteriori* model error $|\mathcal{G}\rangle$ will take

$$|\mathcal{G}(n)\rangle = |\varepsilon(n)\rangle - |\varepsilon^*(n)\rangle \quad (3.31)$$

Notice neither $|\varepsilon\rangle$ nor $|\mathcal{G}\rangle$ are observable. Thus the strategy to find an optimal transform operator $\mathbf{\Delta}$ falls to minimize the expected error norm

$$\|\mathcal{G}(n)\| = E \langle \mathcal{G}(n) | \mathcal{G}(n) \rangle \quad (3.32)$$

In light of fig. 3.1.2 the response prediction error $|e\rangle$ represents the system input. The perfect correction operator equals the system to be identified and the reference resides in the hidden model error $|\varepsilon\rangle$. The prediction error $|e\rangle$ also plays the role of the predictor and the model correction $|\varepsilon^*\rangle$ is the predictand. $\mathbf{\Delta}$ shall match such that cross correlations are conserved.

$$E |\varepsilon^*(n)\rangle \langle e(n)| = E |\varepsilon(n)\rangle \langle e(n)| \quad (3.33)$$

The assignment

$$\begin{array}{ccc} |e(n)\rangle & \dashrightarrow & \mathbf{A} & \mathbf{\Delta}(n) & \dashrightarrow & \mathbf{B} \\ |\varepsilon(n)\rangle & \dashrightarrow & \mathbf{C} & |\varepsilon^*(n)\rangle & \dashrightarrow & \mathbf{C}^* \end{array} \quad (3.34)$$

allows to state the transform operator according to equation 3.7

$$\mathbf{\Delta}(n) = E |\varepsilon(n)\rangle \langle e(n)| \cdot E |e(n)\rangle \langle e(n)|^{-1} \quad (3.35)$$

The expectations follow (see appendix **B** subsection 2.8)

$$\begin{aligned} E |\varepsilon(n)\rangle \langle e(n)| &= \mathbf{O}_1(n) \cdot \mathbf{X}^+(n) = \mathbf{F}(n) \\ E |e(n)\rangle \langle e(n)| &= \mathbf{X}(n) \cdot \mathbf{F}(n) + \mathbf{O}_2(n) \end{aligned} \quad (3.36)$$

with

$$\mathbf{O}_1(n) = E |\varepsilon(n)\rangle \langle \varepsilon(n)|$$

and

$$\mathbf{O}_2(n) = E |\rho(n)\rangle \langle \rho(n)|$$

The operator \mathbf{O}_1 is called estimation error covariance. It can be computed recursively

as shown in appendix **B** subsection 2.12. We call the substitute **F** filtered data matrix and \mathbf{O}_2 is noted as observation noise covariance. Latter is a model parameter and has to be selected carefully to consider for the signal quality of the measurement equipment. Finally the transfer operator is given by the equation

$$\mathbf{\Delta}(n) = \mathbf{F}(n) \cdot (\mathbf{X}(n)\mathbf{F}(n) + \mathbf{O}_2(n))^{-1} \quad (3.37)$$

Our primary interest in the discussion is to find an adaptation rule for the state estimation. Provided the transform operator $\mathbf{\Delta}$ in accordance with equation 3.26 the tuning of the state estimation is written

$$|w(n+1)\rangle = \mathbf{\Phi}(n) \cdot (|w(n)\rangle + |\varepsilon^*(n)\rangle) = \mathbf{\Phi}(n) \cdot (|w(n)\rangle + \mathbf{\Delta}(n)|e(n)\rangle) \quad (3.38)$$

particular defining the Kalman gain matrix $\mathbf{G}(n) = \mathbf{\Phi}(n) \cdot \mathbf{\Delta}(n)$ (3.39)

$$|w(n+1)\rangle = \mathbf{\Phi}(n)|w(n)\rangle + \mathbf{G}(n)|e(n)\rangle \quad (3.40)$$

We close the review by approval that indeed the gain \mathbf{G} as stated in equations 3.37 and 3.39 tunes the adaptive filter at minimal model error norm $\|\mathcal{G}\|$.

With help of equation 3.31 we describe the norm.

$$\begin{aligned} \|\mathcal{G}\| &= E \langle \mathcal{G} | \mathcal{G} \rangle \\ &= E \langle \varepsilon - \varepsilon^* | \varepsilon - \varepsilon^* \rangle \\ &= E \langle \varepsilon | \varepsilon \rangle - 2 \cdot E \langle \varepsilon^* | \varepsilon \rangle + E \langle \varepsilon^* | \varepsilon^* \rangle \end{aligned} \quad (3.41)$$

We compare two different model corrections $|\varepsilon_0^*\rangle$ and $|\varepsilon_1^*\rangle$

$$\begin{aligned} \|\mathcal{G}_0\| &= E \langle \varepsilon | \varepsilon \rangle - 2 \cdot E \langle \varepsilon_0^* | \varepsilon \rangle + E \langle \varepsilon_0^* | \varepsilon_0^* \rangle \\ \|\mathcal{G}_1\| &= E \langle \varepsilon | \varepsilon \rangle - 2 \cdot E \langle \varepsilon_1^* | \varepsilon \rangle + E \langle \varepsilon_1^* | \varepsilon_1^* \rangle \end{aligned}$$

$$\|\mathcal{G}_1\| - \|\mathcal{G}_0\| = -2 \cdot E \langle \varepsilon_1^* | \varepsilon \rangle + E \langle \varepsilon_1^* | \varepsilon_1^* \rangle + 2 \cdot E \langle \varepsilon_0^* | \varepsilon \rangle - E \langle \varepsilon_0^* | \varepsilon_0^* \rangle \quad (3.42)$$

The correction $|\varepsilon_0^*\rangle$ is chosen conserving cross correlations

$$E |e\rangle \langle \varepsilon_0^*| = E |e\rangle \langle \varepsilon|$$

$$\text{thus} \quad E | \varepsilon_0^* \rangle \langle \varepsilon | = \mathbf{\Delta}_0 E | e \rangle \langle \varepsilon | = \mathbf{\Delta}_0 E | e \rangle \langle \varepsilon_0^* | = E | \varepsilon_0^* \rangle \langle \varepsilon_0^* |$$

$$\text{further} \quad E | \varepsilon_1^* \rangle \langle \varepsilon | = \mathbf{\Delta}_1 E | e \rangle \langle \varepsilon | = \mathbf{\Delta}_1 E | e \rangle \langle \varepsilon_0^* | = E | \varepsilon_1^* \rangle \langle \varepsilon_0^* | \quad (3.43)$$

The traces of the covariances equal the corresponding scalar products insofar :

$$E \langle \varepsilon_0^* | \varepsilon \rangle = E \langle \varepsilon_0^* | \varepsilon_0^* \rangle \quad \text{and} \quad E \langle \varepsilon_1^* | \varepsilon \rangle = E \langle \varepsilon_1^* | \varepsilon_0^* \rangle \quad (3.44)$$

Insert to equation 3.39

$$\begin{aligned} \| \mathcal{G}_1 \| - \| \mathcal{G}_0 \| &= E \langle \varepsilon_0^* | \varepsilon_0^* \rangle - 2 \cdot E \langle \varepsilon_1^* | \varepsilon_0^* \rangle + E \langle \varepsilon_1^* | \varepsilon_1^* \rangle \\ &= E \langle \varepsilon_0^* - \varepsilon_1^* | \varepsilon_0^* - \varepsilon_1^* \rangle = \| \varepsilon_0^* - \varepsilon_1^* \|^2 \geq 0 \end{aligned}$$

$$\text{thus} \quad \| \mathcal{G}_1 \| \geq \| \mathcal{G}_0 \| \quad (3.45)$$

which gives a rigid proof that $|\varepsilon_0^*\rangle$ equals the least mean squares correction. Table 3.1.2 summarizes the Kalman filter based on one step prediction. In that case the dimension of the response vector $|r\rangle$ is set to 1. As a matter of course some operators introduced in the review collapse to vector quantities and vectors become scalar values. The full version is printed in appendix B 2.14. The variables and parameters used to formulate the solution to the least mean squares optimal filter problem are :

$ w\rangle$	adaptive filter kernel	$\mathbf{X}, x\rangle$	input data matrix / vector
$ r\rangle, r$	system response vector / scalar	$\mathbf{F}, f\rangle$	filtered data matrix / vector
$\mathbf{G}, g\rangle$	Kalman gain operator / vector	$ e\rangle, e$	a priori system identification error vector / scalar
\mathbf{O}_1	state estimation error covariance matrix	\mathbf{O}_2, ρ^2	observation noise covariance matrix / variance
\mathbf{O}_3, σ^2	process noise covariance matrix / variance		

The great resemblance of the computations in the RLS and the Kalman filter is quite apparent despite the significant difference of the mathematical background. The one detail that makes the Kalman filter well-posed is the additional refresh term in the state estimation error covariance update¹⁶ which is rooted in process noise. Obviously turning to a random view of the system identification experiment paves the way for numerical stabilization.

¹⁶ see iteration step ⑤ in the filter algorithm

Kalman Filter

Initialize the algorithm by setting

$$|w(0)\rangle = |o\rangle, \quad \mathbf{O}_1(0) = \sigma^2 \cdot \mathbf{I}$$

and

$$\sigma = \begin{cases} \text{large constant for high process noise} \\ \text{small constant for low process noise} \end{cases}$$

$$\rho = \begin{cases} \text{large constant for high signal to noise ratio} \\ \text{small constant for low signal to noise ratio} \end{cases}$$

For each instance of time compute

$$e(n) = r(n) - \langle x(n) | w(n-1) \rangle \quad \textcircled{1}$$

$$|f(n)\rangle = \mathbf{O}_1(n-1) \cdot |x(n)\rangle \quad \textcircled{2}$$

$$|g(n)\rangle = \frac{|f(n)\rangle}{\rho^2 + \langle x(n) | f(n) \rangle} \quad \textcircled{3}$$

$$|w(n)\rangle = |w(n-1)\rangle + e(n) \cdot |g(n)\rangle \quad \textcircled{4}$$

and

$$\mathbf{O}_1(n) = \mathbf{O}_1(n-1) - |g(n)\rangle \langle f(n)| + \sigma^2 \cdot \mathbf{I} \quad \textcircled{5}$$

Table 3.1.2: The Kalman filter algorithm in $\langle \text{bra} | \text{ket} \rangle$ notation. The simplified form features a unity state transition operator $\Phi = \mathbf{I}$.

It is a most remarkable property of the Kalman filter, that the recursion of the error covariance \mathbf{O}_1 follows a course in time domain, that is strictly determined by the system stimulus without notice of the system state. This is self explaining if we cancel steps ① and ④ yet processing ②, ③ and ⑤ in an infinite loop. Assume the spectral distribution of the stimulus to be stationary. One might expect the covariance elements to form characteristic patterns? We will examine that in the next subsection since it might be an opportunity to omit expensive computations and replace quantity updates by initial constants. Maybe the stimulus for spectral characterization can be chosen properly to scale down the numerical effort of the ASA application.

3.2 Online Implementation of Spectral Analysis Filters

Adaptive Spectrum Analysis features the stochastic identification technique discussed right before to estimate the harmonics of an observed signal. Before dealing with details we start with a survey of terminology. A typical setup in *spectral characterization* designates analog signals like sensor voltages to be sampled by analog-to-digital converters (ADC). This provides an input data stream for digital signal processing (DSP) ¹. Real time *spectral stimulus* is possible by generating an output data stream that modulates digital-to-analog converters (DAC). To give an example the DAC applies a boost voltage to a mechanical actuator or an optical modulator. All converters are triggered synchronously at constant frequency [GRUE01, ch.1]. The trigger period determines the maximum number of DSP computation cycles ² available between two samples on a microcomputer. ³ If the computational expense of both stimulus and spectral characterization does not exceed that limit acquiring and processing data may run in parallel. We already designated that scenario *online implementation*. As the condition does not match tasks have to be sequenced which gives the approach its label *offline implementation*. The contribution of this work is the development of new algorithms that do have online capability.

The Kalman filter as formulated in section 3.1. is of high numerical effort [HAY02, ch.10]. Though it is known to offer optimal tracking performance [MAY79, ch.1] it is regarded to be far too consuming for an online identification of higher order systems [HOF00, ch.4]. However in case of adaptive spectrum analysis the signals dealt with have the unique property to be harmonic. We will learn in this chapter that the latter opens the door to a collapse of the computational demand. A new algorithmic concept follows immediately. It is explained in the first paragraph. For a deeper insight a discussion of the filter operation in the complex plane reveals the smart analogies to the discrete Fourier transformation (DFT) [BRI88]. A significant part of the embedded software had to be written in assembly to compete with the demand. ⁴ Embedded hardware like a DSP is conveniently programmed in a high level language - for example C. But the approach cannot exploit the computational power of the device due to missing or unspecified low level optimization. This can only be achieved in assembler code. The advanced algorithms examined here fully exhaust the technology limit to master the numerical load (compare footnote ³). Tailored scheduling on the microcomputer's internal buses is mandatory to program a truly optimized application. As a matter of course we need to understand how the processor operates on system level. ⁵

1 It is also common to name the microcomputer that executes the digital signal processing operation a DSP.

2 Annotation : A computation cycle is a single computing step like a multiplication or an accumulation of data values. The DSP computations run at core clock frequency. Modern DSPs are clocked up to 450MHz. The duration of an instruction cycle is related to the core clock.

3 Example : A DSP core frequency of 200 MHz and a sampling frequency of 20 kHz yields a maximum number of 10000 computation cycles per sampling period.

4 Information technology calls lowest hardware level programming language assembly. [SMI99]

5 Annotation : Programming a DSP in C is very popular, since it demands only basic knowledge on the target hardware. All modern DSP compilers support that industry standard. The advantage is clearly

The implementation of the adaptive spectrum analysis filter on the embedded hardware will be elaborated in the second paragraph.

3.2.1 Adaptation Based on Oscillator Ensembles

We addressed common concepts for system identification. The question remains open how well known adaptive techniques have to be modified to be apt for online spectral analysis of a given time series. What is the key of the new approach and how does the innovative method work? For now imagine we have solved the identification problem, namely we have found the complex amplitudes of the spectrum which equal the Fourier coefficients. In that case the signal profile may be *mathematically reconstructed* in time domain. Synthesis is well known to be given by the inverse Fourier transform [GRUE01, ch.4]. In other words the reconstruction yields from a convolution. Namely one of a numerically controlled oscillator ensemble (NCOs twiddling at predefined observation frequencies and phase offsets) and the complex vector of Fourier coefficients (being the result of a decomposition analysis). Now we turn the point of view. We assume that the observed time series is *physically generated* from the said oscillator ensemble stimulating a system with unknown transfer function as shown in fig. 3.2.1. The time series equals the system response. The convolution box may be seen as a container of spectral information. In terms of system theory the box is a linear system with a complex transfer function defined on the Fourier coefficients. Notice it does not count for the considerations whether the box is virtual or embodied. We call this model an *harmonic process*. If the size of the ensemble grows to infinity the process is said to be white [HAY83, ch.3]. The process is understood as a digital filter. So we let an adaptive filter find the optimal and most parsimonious representation of the harmonic process and we name the overall procedure *adaptive spectrum analysis*. The result of the adaptation will be an optimal estimate of the complex vector of Fourier coefficients. The method is illustrated in fig. 3.2.2.

Of course not all processes can be described like that with equal success. To give a counter example: if the spectrum under observation varies too fast within one period of the lowest harmonic a well defined process identification will fail. But this chapter is dedicated to the question how to overcome the computational burden not the physical one. The dynamical limits of the method will be discussed later in the experimental chapter 4. Now we start with an examination of some mathematical properties inherent to the Kalman filter that are essential for an online implementation of the scheme.⁶ First, we note:

- ① The input data vector $|x\rangle$ is defined on a set of complex NCOs with constant twiddling frequencies and phase offsets. All parameters are *ab initio* known.
- ② If the estimation error autocovariance matrix \mathbf{O}_1 is known, the filtered data vector $|f\rangle$ is computed based on the knowledge of the input data vector $|x\rangle$.

that once a software project is realized it may run on different embedded targets.

⁶ Refer to the summary of the Kalman filter equations given in chapter 3.1 for details of notation.

- ③ With *ab initio* defined measurement noise correlation matrix \mathbf{O}_2 the Kalman gain vector $|k\rangle$ follows from the filtered data vector $|f\rangle$ also given $|x\rangle$.
- ④ Provided *ab initio* defined process noise correlation matrix \mathbf{O}_3 and the latest coordinates of $|f\rangle$ and $|k\rangle$ we might update the estimation error autocovariance matrix \mathbf{O}_1 .
- ⑤ The Kalman filter processes the steps ②, ③ and ④ in an endless loop.
- ⑥ Choosing a meaningful initialization⁷ for the estimation error autocovariance matrix the time dependency of \mathbf{O}_1 , $|f\rangle$ and $|k\rangle$ is strictly determined by the trajectory of the input data vector $|x\rangle$ in time space.

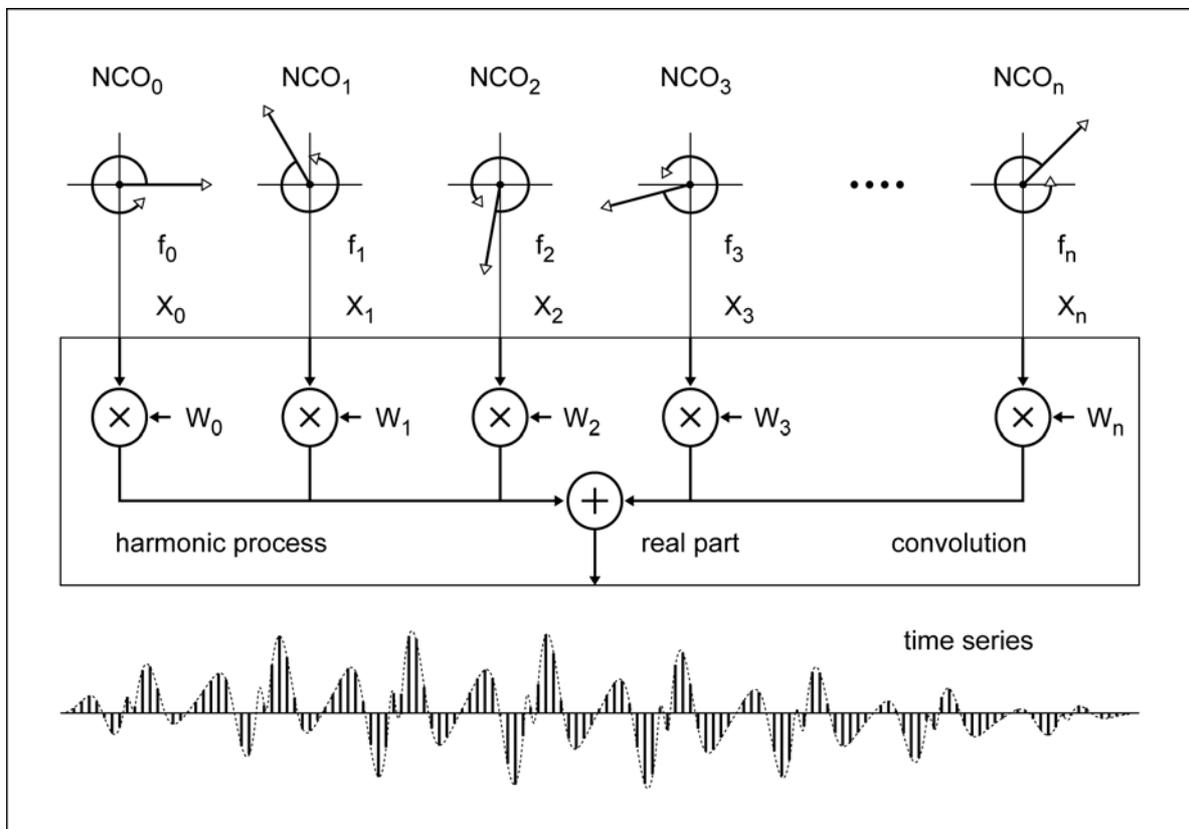


Figure 3.2.1: Synthesis of a real time series by convolution of an oscillator ensemble $NCO_0 \dots NCO_n$ and a set of complex weights $w_0 \dots w_n$. The oscillators are twiddling counter clockwise at frequencies $f_0 \dots f_n$ with arbitrary phase offset. The convolution box is the model core of the adaptive spectrum analysis filter.

Notice that in case of applying the Kalman filter to adaptive spectrum analysis the trajectory is *a priori* given through a selection of twiddling frequencies and phase offsets for the oscillator ensemble.

⁷ It is common to start with a scaled unity matrix [HOF00]. The diagonal elements might be set to the square of the expected maximal spectral peak.

Second, we state:

- ① One of the most costly computations within the Kalman filter is laid in the update of the estimation error autocovariance matrix \mathbf{O}_1 . Given NCO order N the update equation requires the execution of N^2 multiplication and accumulation cycles on the DSP. Thus we must investigate whether there exists a steady state of \mathbf{O}_1 case the input data vector $|x\rangle$ is driven by the oscillator ensemble. If such a state exists it will eventually allow us to avoid the recalculation of already known constant values or characteristic timings.
- ② The evaluation of the filtered data vector $|f\rangle$ is of equal expense. So we might also benefit hereupon with the estimation error autocovariance matrix gaining convergence. We have to clarify whether a stationary matrix causes the coordinates of the filtered data vector to take on a characteristic trajectory in time space, maybe even a periodical course.
- ③ Provided items ① and ② are answered in a positive sense we must exploit the results to minimize the runtime of the Kalman filter.

To solve these issues we observe the filter operation *offline* under normative conditions. With given initial values we compute the estimation error autocovariance, the filtered data vector and the Kalman gain. We omit the error feedback path since

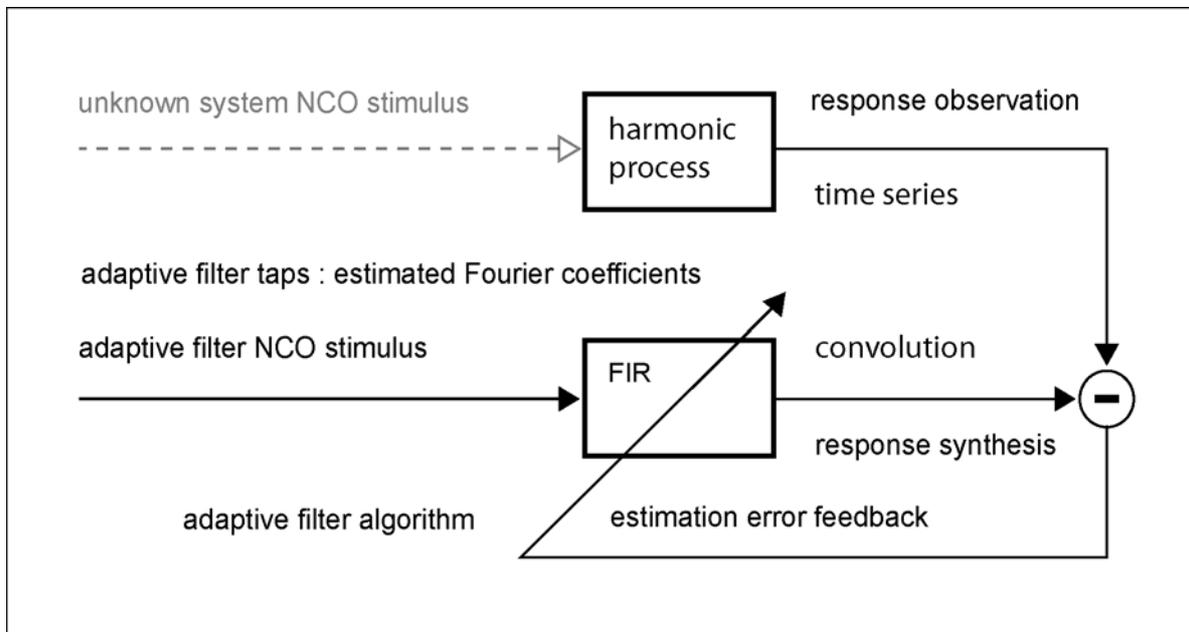


Figure 3.2.2: Mapping of spectral analysis into the context of system identification by adaptive filtering. The filter operation is modeled by a finite impulse response filter kernel (FIR). The filter input vector is given by real and imaginary parts of the numerically controlled oscillators (NCO). If the order of the oscillator ensemble is N the length of the input vector is 2 times N . The same applies to the vector of adaptive filter taps which play the role of Fourier coefficients to be estimated.

we do not want to perform an adaptation. So the loop of the Kalman filter is not paced by the sampling period T_s but the computational runtime T_{loop} . For example if we feature an NCO ensemble of order 128 the filter roughly spoken runs in slow motion. The twiddling of the oscillators however is generated on the base of the sampling period as shown

$$\Delta\varphi_{NCO} = 2\pi \cdot f_{NCO} \cdot \Delta t \quad \text{with} \quad \Delta t = T_s \quad (3.46)$$

Hence the sampling rate is set constant to 20 kHz which is suited in practice to analyze spectra up to 2 kHz bandwidth⁸. Exemplary autocovariance matrix elements are recorded in parallel with the input data vector for comparison. The input data vector is built with the real and imaginary parts of the considered harmonic oscillators. To handle complex numbers we use interleaved notation. Real part and imaginary part of a single complex NCO supply an even–odd real coordinate pair. Consequently, the input data vector is treated as a real quantity. The same applies to the autocovariance. Recordings are repeated in analogy for the filtered data and the Kalman gain vector. To distinguish transients from long term behavior two sets of data are acquired for each quantity. One covers the startup interval which is defined from zero cycle to 20000. The other set replays a time window of 1024 samples starting at cycle 200000. The normative conditions of the filter operation are as follows : the measurement noise is canceled without loss of generality. The harmonic process noise is arbitrarily chosen as 1.0 . In a first approach the oscillator frequencies range from 1 Hz to 2048 Hz and are located equidistant in frequency space. We describe an harmonic process of order 128. Thus, the frequency resolution is 16.12 Hz . The order conditions the input data vector, the filtered data vector and the Kalman gain vector to be of length 256. The autocovariance matrix is quadratic and of size 256×256 entries. Indexing starts with 0 and finishes on 255.

First, we capture the startup interval of the autocovariance to obtain information on typical convergence classes. Fig. 3.2.3 depicts a track of the following diagonal matrix elements. Row index 1 is related to the bandwidth minimum 1 Hz whereas 255 marks the maximum 2048 Hz⁹. Index 51 stands for a medium range frequency 403.9 Hz. The process noise determines the level of mean values and modulation amplitudes¹⁰. Both parameters settle quickly after a view oscillations with regard to the corresponding NCO order. Fig. 3.2.4 reveals non–diagonal matrix elements are mean free. The property of constant modulation amplitude at steady state still holds and is demonstrated here for row index 10 and column index 150 which represent the frequencies 81.6 Hz and 1209.9 Hz. The observed profiles, especially the characteristic beats in fig. 3.2.4 , give rise to the hypothesis that all alternating magnitudes can be seen as crosscorrelation terms of two harmonic oscillators with tailored frequencies and phase offsets. Evidence for that is supported by the mathematical structure of the Kalman filter. We remember the filter equations that we explored in chapter 3.1.

⁸ Though the Nyquist frequency [GRUE01] for a 2.5 kHz bandwidth limited system lies at 5 kHz modern digital spectrum analysis equipment features at least 5 times oversampling to guarantee frequency data fidelity .

⁹ The NCO order is given by index value modulo 2

¹⁰ Variations of the process noise cause a linear change

The estimation error autocovariance matrix is initialized as :

$$\mathbf{O}_1 = \mathbf{I} \tag{3.47}$$

At startup we obtain an approximation of the filtered data vector :

$$|f\rangle = \mathbf{O}_1 \cdot |x\rangle \approx |x\rangle \tag{3.48}$$

Moreover taking zero measurement noise for the Kalman gain vector :

$$|k\rangle = \frac{|f\rangle}{\langle f|x\rangle} \approx \frac{1}{\langle x|x\rangle} \cdot |x\rangle \tag{3.49}$$

Since the input data vector sets from a harmonic oscillator ensemble of order N the norm simplifies

$$\langle x|x\rangle = \sum_{i=1}^N \cos^2(\varpi_i t + \varphi_i) + \sin^2(\varpi_i t + \varphi_i) = N \tag{3.50}$$

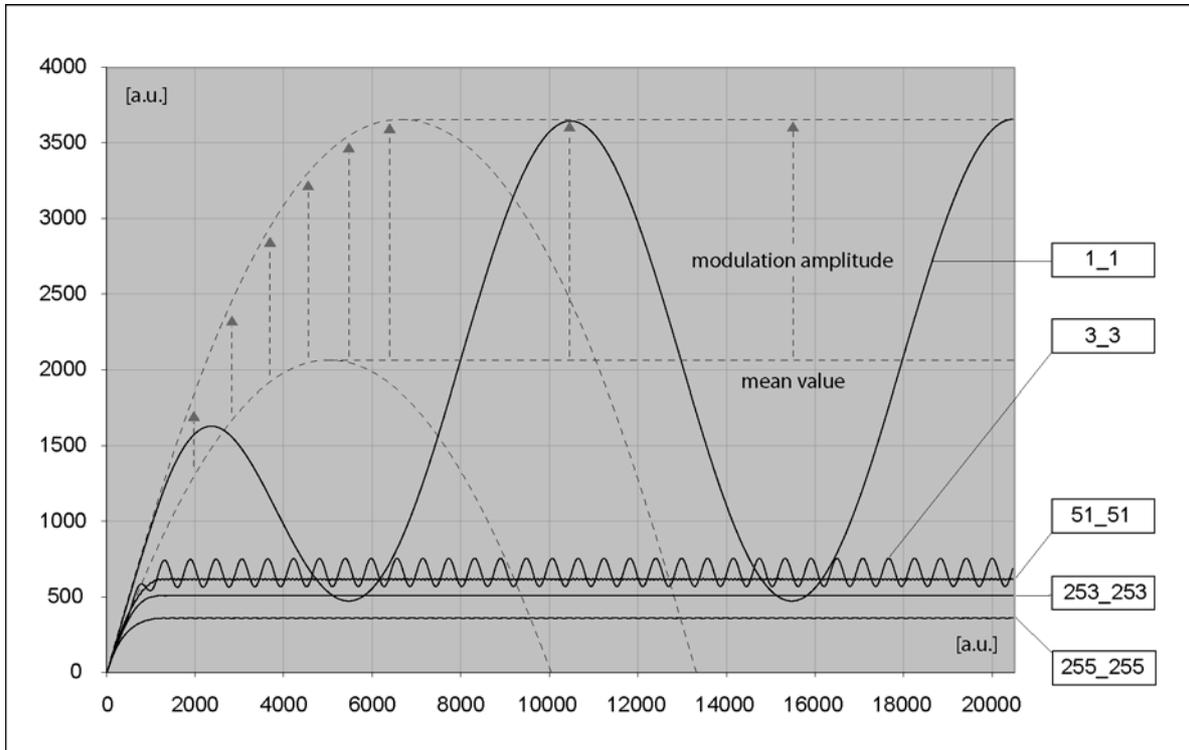


Figure 3.2.3: Parabolic transients of selected estimation error autocovariance matrix diagonal elements. The numerical experiment features 128 oscillators. The x-axis is scaled in ADC conversion cycles. The plot shows a time window of 1 second after the adaptive filter computation is started.

and also the Kalman gain :

$$|k\rangle \approx \frac{1}{N} \cdot |x\rangle \quad (3.51)$$

The autocovariance update obeys an integral equation :

$$\mathbf{O}_1 = \sum (1 - |k\rangle\langle f|) \approx \sum \left(1 - \frac{1}{N} \cdot |x\rangle\langle x| \right) \quad (3.52)$$

The outer product $\sum |x\rangle\langle x|$ clearly finds the observed autocorrelation terms as expected. It is a remarkable property of adaptive spectrum analysis that this obvious signature of the autocovariance is conserved throughout the filter evolution. To conclude the considerations on autocovariance startup we empirically classify the function types, that the transients are assigned to. We notice in fig. 3.2.3 that the mean values of the autocovariance diagonal elements show a parabolic ascent. The data plotted in fig. 3.2.4 suggests that non-diagonal elements are mean free. Modulation amplitude of an arbitrary autocovariance matrix element grows linearly. This is consistent to the integration of corresponding entries in the autocovariance

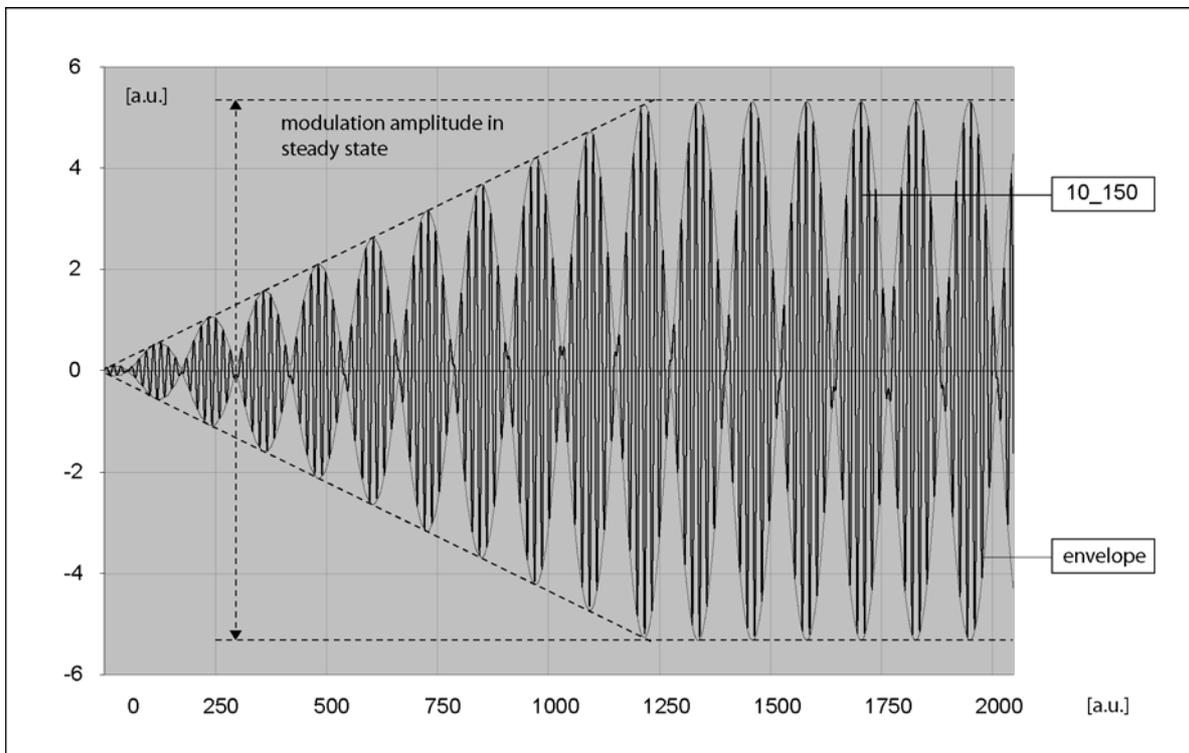


Figure 3.2.4: *Linear convergence course of an arbitrary estimation error autocovariance matrix non-diagonal element within the first 2,000 cycles of adaptive filter iteration. The index notation i_j refers to the matrix row i and column j . Steady state is given at stationary modulation amplitude. The course is mean free.*

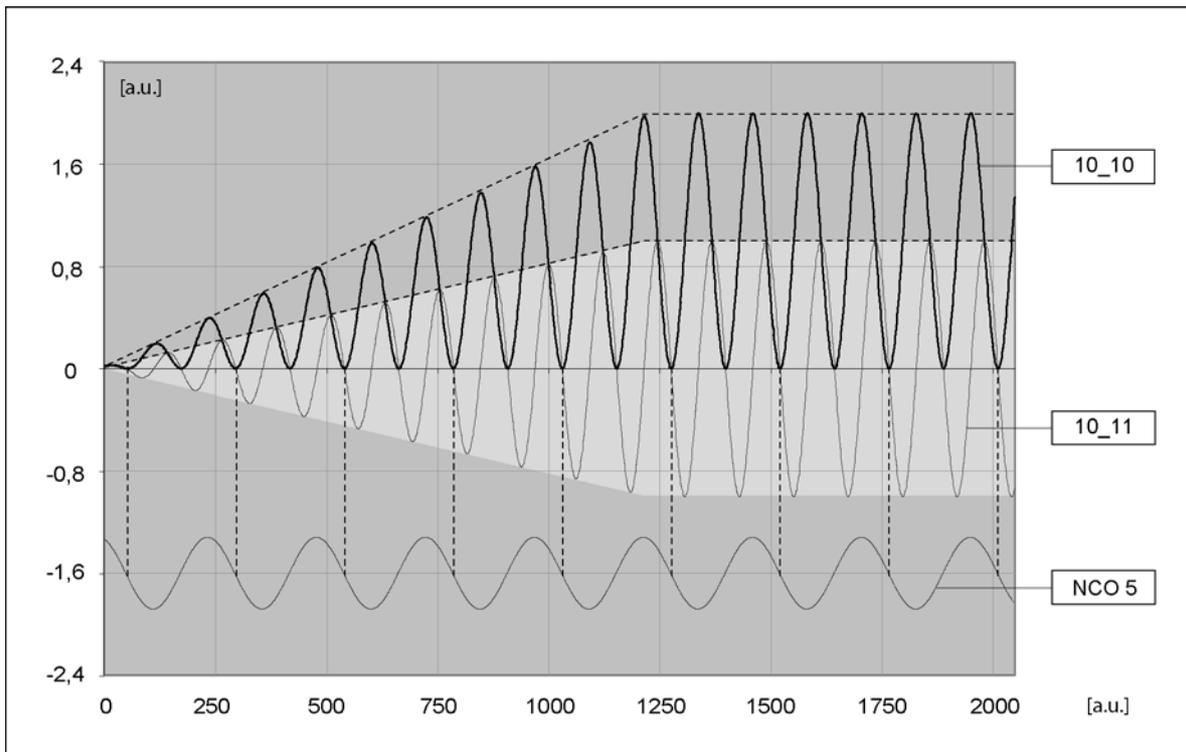


Figure 3.2.5: Linear growth of an autocovariance decay rate diagonal element. For comparison the neighbor-in-row element is also graphed. The mean of the diagonal element equals half modulation amplitude, the non-diagonal element is mean free. The depicted pair of matrix elements is entangled at 90 degrees phase shift and oscillates at twice the frequency of the corresponding NCO which is added in the lower half of the diagram.

decay rate matrix $|k \rangle \langle f|$ as shown in fig. 3.2.5¹¹. Reaching mean apex for all diagonal elements distinguishes the convergence state. In that state decay and harmonic process noise form an equilibrium.

Equation 3.48 tells us that the filtered data vector results from a multidimensional convolution of the input data vector and the autocovariance. It has been verified that all matrix elements bear a characteristic beat signature. Thus one can expect the phase relations within the beats will play a key role for the phase trimming of the filtered data vector. Now that we elaborated the characteristics of stationarity we take a closer look on the phasing of the matrix elements as entered steady state. Modulation and carrier frequencies are related to the addressing indexes: index modulo 2 yields one NCO member's order¹². Row and column indexes select two characteristic frequencies from the numerically controlled oscillator ensemble. Spoken in terms of modulation the higher frequency defines a carrier while the lower frequency shapes a modulator. For top triangular matrix elements the modulator is given by the row index and the carrier by the column index. This is rendered in fig. 3.2.6. For the bottom triangular matrix the roles are reversed. The autocovariance is symmetric because of the mathematical structure of the update process (see equation 3.52). Concerning diagonal elements modulation and carrier frequencies are equal, thus entities are oscillating at twice the corresponding NCO frequency. This is

¹¹ Proof can be lead by trivial analysis [HEU94].

¹² Modulo 2 arithmetic is due to interleaved notation of complex vectors.

consistent to the transient recordings. All matrix elements gathered in fig. 3.2.6 share the modulator frequency 81.6 Hz and have the carrier frequencies 403.9 Hz , 1209.9 Hz and 2015.8 Hz. Notice the constant offset in the diagonal element 10_10 while the non-diagonal elements 10_50 , 10_150 and 10_250 are mean free. For advanced illustration reference oscillations at the assumed carrier and modulation frequencies are added to the plots. The match to the autocovariance data is excellent.

Concerning phase entanglement we zoom into the plots and study half and quarter modulation period data. Fig. 3.2.7 reveals even-odd index neighbors are oscillating with a constant phase lag of 90 degrees which is noted as Δ . Correlated NCOs are drawn in thin lines for comparison. The row index is common for the depicted courses. Consequently all graphs share equal modulation. A tailored offset is added to the diagonal element 10_10 for educated presentation. The autocovariance is symmetric which leads to identical plots for the transposed matrix elements. It will turn out soon that we may root a significant reduction of the adaptive filter's computational complexity to locked phases of even-odd proximately indexed matrix elements.

We next derive simplified equations for the filtered data vector $|f\rangle$ and the Kalman gain vector $|k\rangle$. Before proceeding we memorize how to compute the coordinates of the filtered data vector

$$f_i = \langle o_i^1 | x \rangle \quad (3.53)$$

with $\langle o_i^1 |$ as the row i of the autocovariance \mathbf{O}_1 .

As we found the stationary matrix elements in cycle n carry the most general form of a beat signature [FEY89] :

$$\begin{aligned} o_{ij}^1(n) = & a_{ij}^1 \cdot \sin\left(n \cdot (\varpi_i + \varpi_j) \cdot T_s + (\varphi_i + \varphi_j)\right) + \\ & b_{ij}^1 \cdot \sin\left(n \cdot (\varpi_i - \varpi_j) \cdot T_s + (\varphi_i - \varphi_j)\right) \end{aligned} \quad (3.54)$$

with the sampling period T_s and the modulation amplitudes a_{ij}^1 and b_{ij}^1 . Even-odd proximately indexed elements fulfill :

$$\begin{aligned} o_{i2k}^1(n) = & a_{i2k}^1 \cdot \sin\left(n \cdot (\varpi_i + \varpi_{2k}) \cdot T_s + (\varphi_i + \varphi_{2k})\right) + \\ & b_{i2k}^1 \cdot \sin\left(n \cdot (\varpi_i - \varpi_{2k}) \cdot T_s + (\varphi_i - \varphi_{2k})\right) \\ o_{i2k+1}^1(n) = & -a_{i2k}^1 \cdot \cos\left(n \cdot (\varpi_i + \varpi_{2k}) \cdot T_s + (\varphi_i + \varphi_{2k})\right) + \\ & b_{i2k}^1 \cdot \cos\left(n \cdot (\varpi_i - \varpi_{2k}) \cdot T_s + (\varphi_i - \varphi_{2k})\right) \end{aligned} \quad (3.55)$$

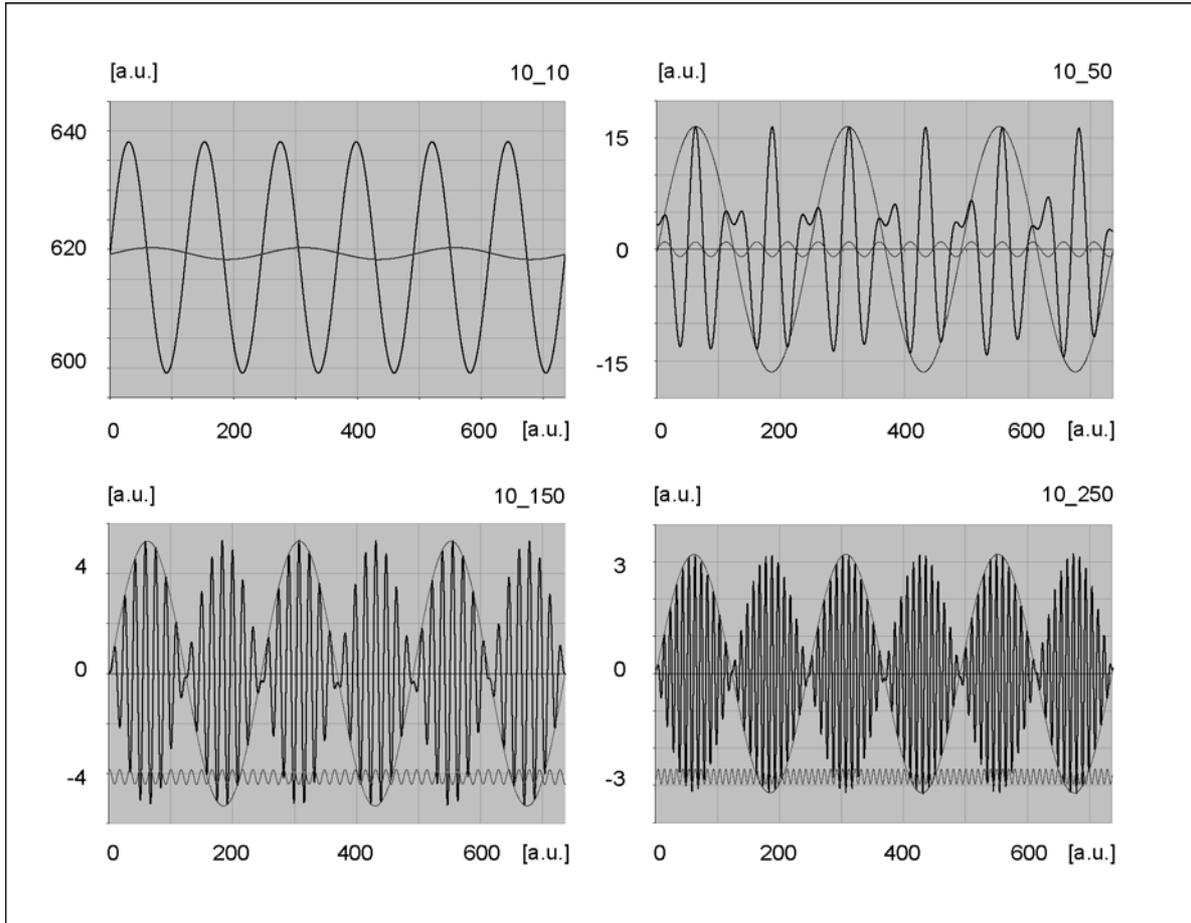


Figure 3.2.6 : A series of autocovariance elements mapped to the top triangular matrix. All entries are located in the same row. Corresponding coordinates of the input data vector are complex weighted by means of a least square fit to find smooth envelopes to the captured autocovariance entities. The plots show a window of three complete modulator periods at a time. Sets are arbitrarily cut from the time line as the Kalman filter has entered steady state.

where we have utilized

$$\begin{aligned} a_{i2k+1}^1 &= a_{i2k}^1 & \varpi_{2k+1} &= \varpi_{2k} & \varphi_{2k+1} &= \varphi_{2k} - \pi/2 \\ b_{i2k+1}^1 &= b_{i2k}^1 \end{aligned} \quad (3.56)$$

The coordinates of the input data vector $|x\rangle$ are :

$$x_j(n) = \begin{cases} \cos(n \cdot \varpi_j T_s) & \text{for } j = 2k \\ \sin(n \cdot \varpi_j T_s) & \text{for } j = 2k + 1 \end{cases} \quad k \in N \quad (3.57)$$

To evaluate the scalar product in equation 3.53 we divide the sum into even and odd indexed summands. The NCO order is given as N .

$$\langle o_i^1 | x \rangle = \sum_{j=0}^{2N-1} o_{ij}^1 \cdot x_j = \sum_{k=0}^{N-1} o_{i2k}^1 \cdot x_{2k} + \sum_{k=0}^{N-1} o_{i2k+1}^1 \cdot x_{2k+1} \quad (3.58)$$

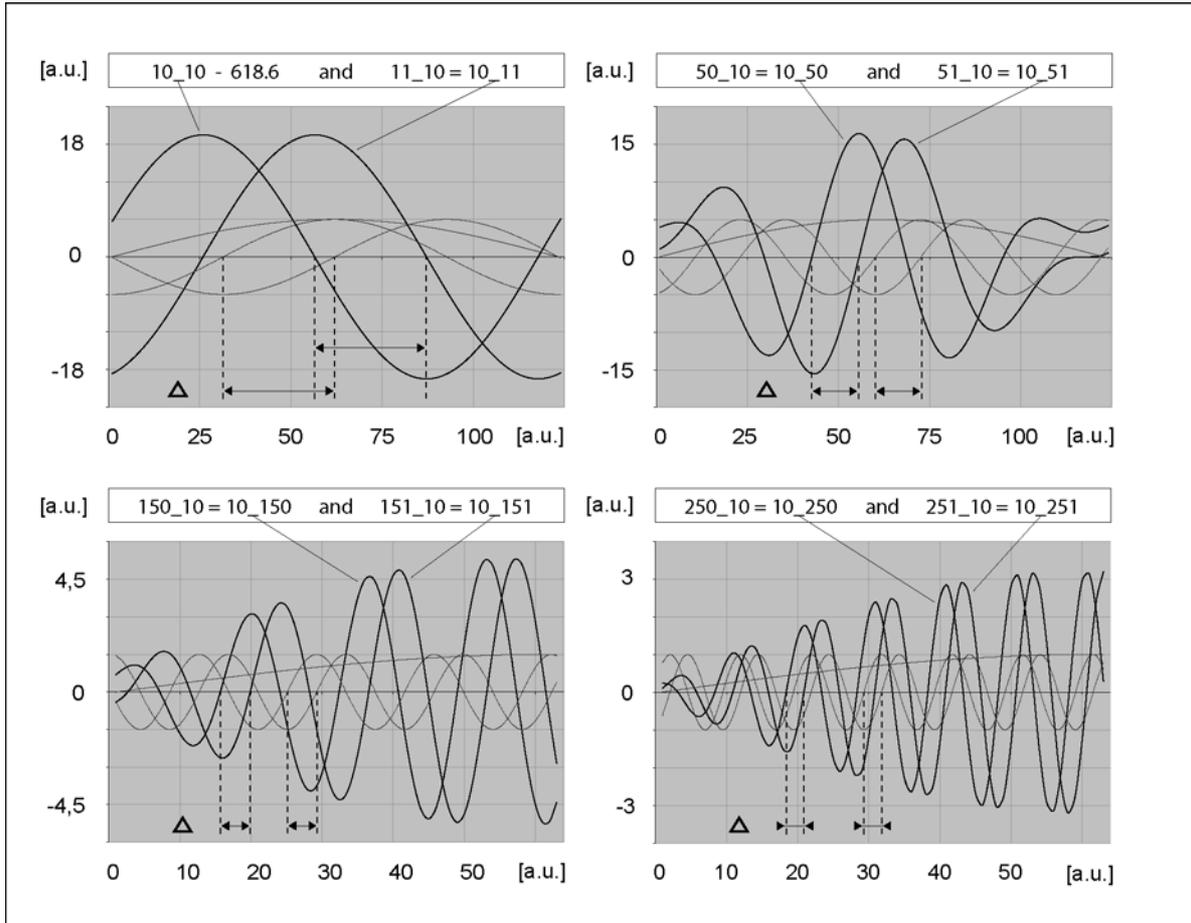


Figure 3.2.7: Time resolved acquisitions of proximate autocovariance matrix elements. Even-odd index neighbors oscillate at constant phase lag. Added reference harmonics are taken from the input data vector. Reference amplitude is scaled for ease of comparison while phase is original. The modulation period corresponding to index 10 equals 12.3 ms. The top diagrams show half and the bottom ones quarter modulation period data.

We find through trigonometric addition theorems [BSMM93]

$$\begin{aligned}
 o_{i2k}^1(n) \cdot x_{2k}(n) = & \\
 a_{i2k}^1 \cdot \sin(n \cdot \varpi_i T_s + \varphi_i) \cdot \cos^2(n \cdot \varpi_{2k} T_s + \varphi_{2k}) &+ 0.5 \cdot a_{i2k}^1 \cdot \cos(n \cdot \varpi_i T_s + \varphi_i) \cdot \sin(2n \cdot \varpi_{2k} T_s + \varphi_{2k}) + \\
 b_{i2k}^1 \cdot \sin(n \cdot \varpi_i T_s + \varphi_i) \cdot \cos^2(n \cdot \varpi_{2k} T_s + \varphi_{2k}) &- 0.5 \cdot b_{i2k}^1 \cdot \cos(n \cdot \varpi_i T_s + \varphi_i) \cdot \sin(2n \cdot \varpi_{2k} T_s + \varphi_{2k})
 \end{aligned}$$

and

$$\begin{aligned}
 o_{i2k+1}^1(n) \cdot x_{2k+1}(n) = & \\
 -0.5 \cdot a_{i2k}^1 \cdot \cos(n \cdot \varpi_i T_s + \varphi_i) \cdot \sin(2n \cdot \varpi_{2k} T_s + \varphi_{2k}) &+ a_{i2k}^1 \cdot \sin(n \cdot \varpi_i T_s + \varphi_i) \cdot \sin^2(n \cdot \varpi_{2k} T_s + \varphi_{2k}) + \\
 0.5 \cdot b_{i2k}^1 \cdot \cos(n \cdot \varpi_i T_s + \varphi_i) \cdot \sin(2n \cdot \varpi_{2k} T_s + \varphi_{2k}) &+ b_{i2k}^1 \cdot \sin(n \cdot \varpi_i T_s + \varphi_i) \cdot \sin^2(n \cdot \varpi_{2k} T_s + \varphi_{2k})
 \end{aligned}$$

which leads us to :

$$o_{i2k}^1(n) \cdot x_{2k}(n) + o_{i2k+1}^1(n) \cdot x_{2k+1}(n) = (a_{i2k}^1 + b_{i2k}^1) \cdot \sin(n \cdot \varpi_i T_s + \varphi_i) \quad (3.59)$$

So the scalar product becomes :

$$\langle o_i^1 | x \rangle = \sum_{k=0}^{N-1} (a_{i2k}^1 + b_{i2k}^1) \cdot \sin(n \cdot \varpi_i T_s + \varphi_i) \quad (3.60)$$

Introducing the substitutes $f_i = \sum_{k=0}^{N-1} a_{i2k}^1 + b_{i2k}^1$ and $\varphi_i = \theta_i + \pi/2$ we summarize :

$$|f\rangle = [f_0, f_1 \dots f_{2N-1}]^T$$

with $f_i(n) = f_i \cdot \cos(n \cdot \varpi_i T_s + \theta_i)$

whereas $f_{2k+1} = f_{2k} \quad \varpi_{2k+1} = \varpi_{2k} \quad \theta_{2k+1} = \theta_{2k} - \pi/2 \quad (3.61)$

The variable θ denotes the phase shift between input data vector and filtered data vector. Because even-odd proximately indexed coordinates share the same mode the filtered data vector $|f\rangle$ shows the amazing property to be phase locked to the input data vector $|x\rangle$. This is confirmed in fig. 3.2.8 which compares some of the already discussed orders to the real and imaginary parts of addressed NCOs. If we decimate $|f\rangle$ to a complex vector we may take down a magnitude and phase plot for all harmonic orders¹³. For the readers convenience both plots are combined in one diagram (bottom right). The dependent values are plotted versus the NCO order. As we can see complex $|f\rangle$ steadily evolves into the limit functions. Keep in mind we aim to exploit stationarity to minimize the computational effort. The results are very promising up to now. If magnitude and phasing of the filtered data vector is to be treated as constant, moreover if it is *ab initio* known, we can derive $|f\rangle$ straight forward from the input data vector $|x\rangle$ without knowledge of the autocovariance \mathbf{O}_1 . This fact advices us to measure both quantities *offline* in steady state and apply the captured encoding *online*. While stationarity was explained more or less empirically with respect to the autocovariance we have to clarify now how to distinguish it quantitatively for the filtered data vector. After all we need a decision criterion to detect whether the Kalman filter has settled. Notice the magnitudes to converge on a constant level for all harmonic orders. We focus on the magnitude's empirical standard deviation σ [LEH92] to define an intuitive cost function (argumentation see footnote¹⁴).

¹³ Annotation : One even-odd proximately indexed coordinate pair yields a complex number

¹⁴ Fig. 3.2.9 explicates we are offered several degrees of freedom to choose an observation parameter and build a merit criterion. Apparently phasing converges faster than magnitude. We may compare the measured functions to the steady state limit ones which are assumed to have formed after 200000 cycles. After 200 cycles the root mean squared relative error is 25.1 percent in phase and 65.7 percent in magnitude. After 1000 cycles it decreases to 9.2 percent respectively 11.9 percent. Thus to detect stationarity – which means none of the parameters changes significantly – magnitudes are an educated choice.

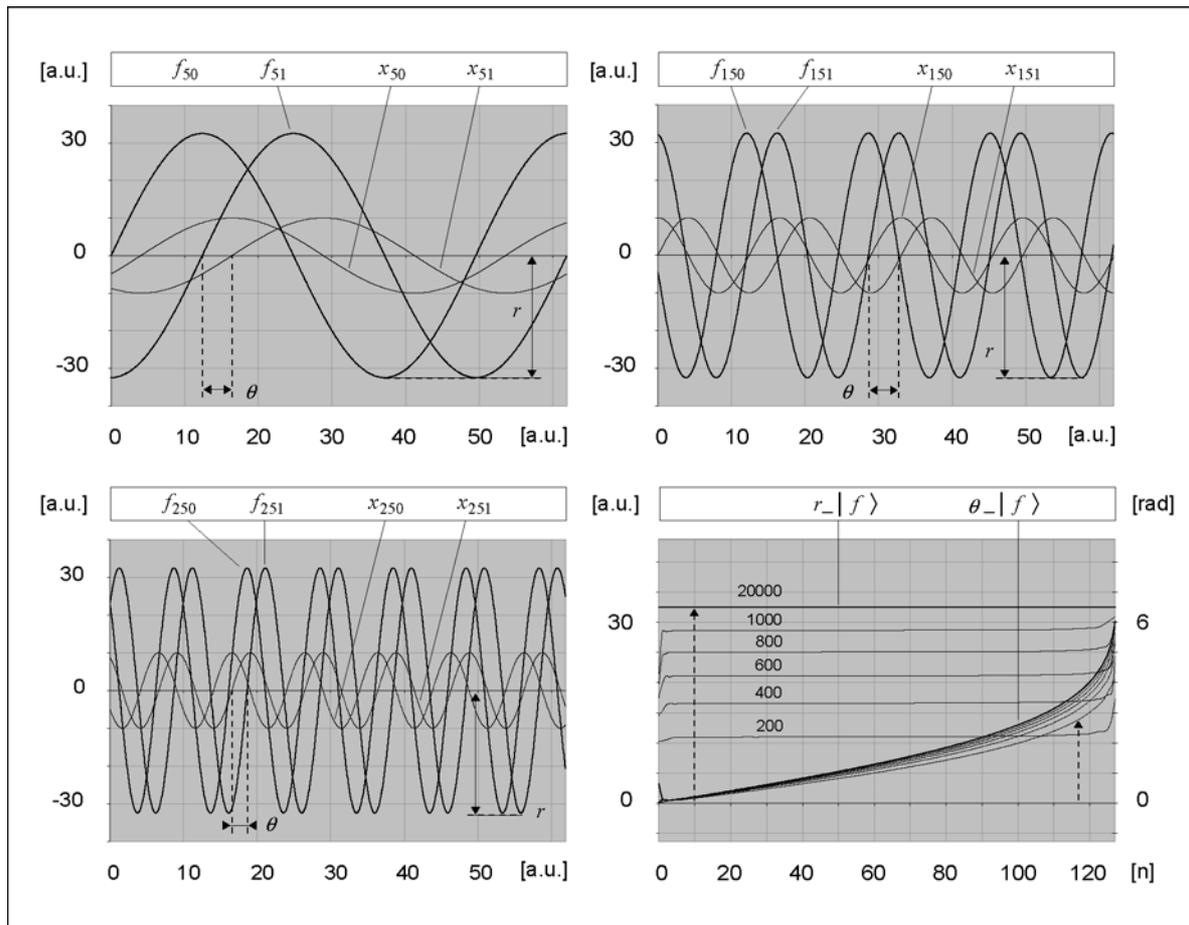


Figure 3.2.8: Data recordings of proximate filtered data vector coordinates. Even-odd index neighbors oscillate phase locked to the real and imaginary part of the corresponding NCO. The bottom right diagram gives an overview of the phase coding θ within the first 20,000 adaptive filter loops. Snapshots are taken at the listed cycles. The direction of the timeline is indicated by the dashed arrows. For completeness the magnitude r is also embedded here.

The left hand diagram in fig. 3.2.9 shows σ of magnitudes r versus the adaptive filter loop cycles in bold line while σ of phasing θ is printed in thin line. While the ascent in θ disqualifies the quantity for a decision the descent in r is apt for steady state detection. Additionally the \mathcal{H}_∞ norms [SKO01] of both gradient vectors Δr and $\Delta \theta$ are added to the plot¹⁵. The \mathcal{H}_∞ norm Δr_{max} could be used for a decision with equal success. The right hand diagram shows the metric norm of the filtered data vector $|f\rangle$ and its orthogonal projection on the input data vector $|x\rangle$ [ART91]. The projection is scaled upon 100 : 1 for better visibility. The favorite merit criterion is consistent with the evolution of the metric. We state the Kalman filter to operate stationary if the standard deviation of magnitudes falls short of a given limes.

¹⁵ Annotation : The \mathcal{H}_∞ norm of a vector equals the coordinate with the highest absolute value.

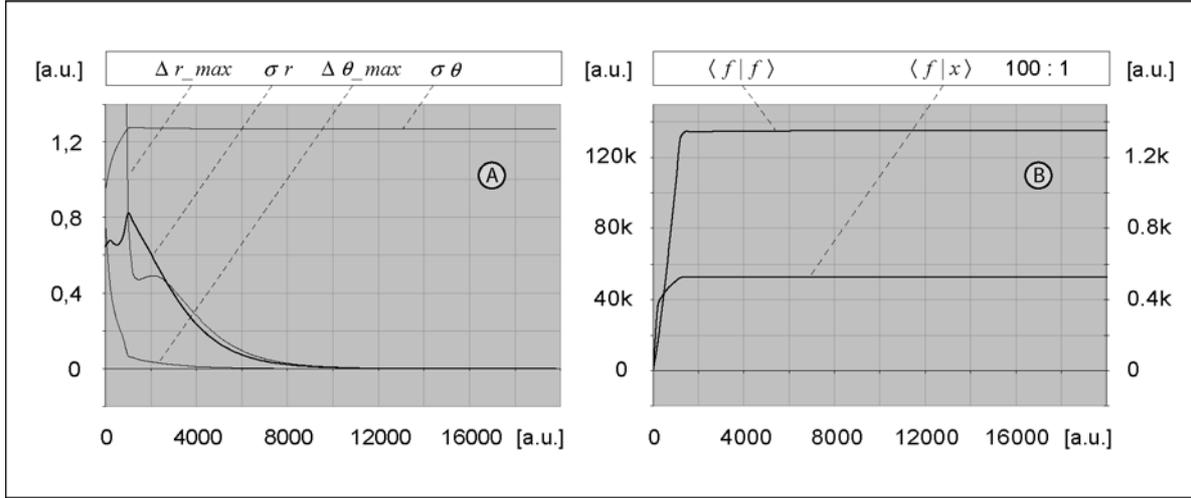


Figure 3.2.9: A – Comparison of different cost functions to feature steady state detection in adaptive spectrum analysis. The plot clusters the empirical standard deviation of the filtered data vector and the \mathcal{H}_∞ norm of the gradient for both magnitudes Δr_{max} and phases $\Delta \theta_{max}$ within the first second of adaptive filter runtime. B – Capture of the filtered data vector's metric norm and orthogonal projection on the input data in the same time window. The left ordinate refers to the norm, the right ordinate to the projection.

Now that we unfolded all signatures of the filtered data vector we finally simplify the computation of the Kalman gain and conquer. We recall from equation 3.49 that the orthogonal projection $\langle f|x \rangle$ of the filtered data on the input data vector is mandatory to derive the gain. It already indicated in fig. 3.2.9 that the projection either stabilizes.

With $f_i = f \ \forall i$ the projection depends on the phasing of the filtered data vector as :

$$\langle f|x \rangle = f \cdot \sum_{k=0}^{N-1} \cos(n \cdot \varpi_{2k} T_s + \theta_{2k}) \cdot \cos(n \cdot \varpi_{2k} T_s) + \sin(n \cdot \varpi_{2k} T_s + \theta_{2k}) \cdot \sin(n \cdot \varpi_{2k} T_s) \quad (3.62)$$

And similar to equation 3.59 utilizing trigonometric addition theorems :

$$\begin{aligned} \langle f|x \rangle &= f \cdot \sum_{k=0}^{N-1} \cos^2(n \cdot \varpi_{2k} T_s) \cdot \cos(\theta_{2k}) + \sin^2(n \cdot \varpi_{2k} T_s) \cdot \cos(\theta_{2k}) \\ &= f \cdot \sum_{k=0}^{N-1} \cos(\theta_{2k}) \end{aligned} \quad (3.63)$$

The Kalman gain coordinates follow from equation 3.49.

$$|k \rangle = \frac{|f \rangle}{\langle f|x \rangle} = [k_0, k_1 \dots k_{2N-1}]^T \quad (3.64)$$

with $k_i(n) = g \cdot \cos(n \cdot \varpi_i T_s + \theta_i)$

whereas
$$g^{-1} = \sum_{k=0}^{N-1} \cos(\theta_{2k})$$

moreover
$$\varpi_{2k+1} = \varpi_{2k} \quad \theta_{2k+1} = \theta_{2k} - \pi/2$$

So we can conclude the above examinations with convincing results:

- ① The matrix elements of the estimation error autocovariance \mathbf{O}_1 constitute from single harmonics cross correlation. Frequencies and phase offsets are strictly determined by the numerically controlled oscillator ensemble that drives the adaptive spectrum analysis. The autocovariance is assumed to be stationary as all of its mean values and modulation amplitudes have settled on constant levels.
- ② The phase entanglement within the autocovariance entries rules the filtered data vector coordinates to oscillate phase locked to the numerically controlled oscillators. This conditions even–odd coordinate pairs to share a common frequency. In steady state $|f\rangle$ has a characteristic phase encoding profile $|\theta\rangle$ and – provided that the process noise is homogeneously distributed – equal oscillation amplitudes. Since the Kalman gain vector $|k\rangle$ is proportional to $|f\rangle$ said properties also apply to the gain. Stationarity is detected from a threshold of magnitude's standard deviation.
- ③ It was demonstrated that a steady state capture of the phase encoding profile allows to fully reconstruct the Kalman gain from the input data. We can skip the expensive updates of the autocovariance and the filtered data vector from the common gain computation if we know the profile. The reconstruction effort is on the order of N which is a highly desirable reduction compared to the order N^2 of the standard filter. The gain coordinates follow from the input data coordinates by phase shifting and amplitude scaling. Shift is immediately given by the profile while amplitude yields from the inverse sum of phase cosines. Thus the proposal how to equip adaptive spectrum analysis with online capability is to run the Kalman filter *offline* and check for stationarity. As detected the phase encoding profile is to be measured and stored. Now based on the profile the filter can be executed at minimized loop time which allows for an *online* application.

The results are verified by extensive computations. Phase encoding profiles of the Kalman gain vector are captured in the scenarios listed in tab. 3.2.1 .

s	nob	n	d	f_0 [Hz]	f_1 [Hz]	f_2 [Hz]	f_3 [Hz]
1	1	128	equal	1.0	–	–	128.0
2	1	128	equal	1.0	–	–	2048.0
3	2		equal				
4	2	21	random	275.0	484.5	1048.7	1757.9
5	1	512	equal	1.0	–	–	512.0
6	1	512	equal	1.0	–	–	2048.0
7	2		equal				
8	2	72	random	269.4	505.7	1070.6	1787.6
9	1	2048	equal	1.0	–	–	2048.0
10	2	1022	equal	258.0	512.0	1025.0	1791.0
11	2	304	random	259.0	511.0	1029.0	1791.0

Table 3.2.1 : *Experimental boundaries of different Kalman gain generation scenarios*

s	:	<i>scenario index</i>
nob	:	<i>number of frequency bands used in the experiment</i>
n	:	<i>identification order, featured number of NCOs</i>
d	:	<i>distribution type, NCO frequencies are either equally or randomly distributed</i>
f_0	:	<i>lowest order NCO frequency</i>
f_1	:	<i>optional boundary for band separation I , used only if nob > 1</i>
f_2	:	<i>optional boundary for band separation II , used only if nob > 1</i>
f_3	:	<i>highest order NCO frequency</i>

Standard deviation of magnitudes is referenced as the criterion of stationarity. Captures are triggered on a threshold of $1.0e^{-9}$. Fig. 3.2.10 depicts the measured data pool. We recognize a decreasing Kalman gain magnitude with increasing harmonic order in all plots. This is reasonable from a physical point of view. The less gain the slower the adaptation will turn out. Indeed higher order spectra need longer analysis time than lower order ones. Each diagram in the left graphics column compares two gain vectors with identical order but different frequency spacing. Spacing is equidistant and can be queried from above table. As we find the phase encoding profiles differ only slightly whereas the magnitudes vary on a larger scale. On the one hand this provides an indicator that the phase encoding of the Kalman gain plays an intimate role within the approach. On the other hand the high sensitivity on variations of the encoding is a very good point to qualify the theoretical model behind adaptive spectrum analysis. We just learned magnitudes can be reconstructed from phase data. Therefore the expected dependency in equation 3.64 is evaluated and the recomputed values are added to the plots. In accordance to the standard deviation of magnitudes the agreement is 9 digits of precision which can only be hint at the graphics. The right column diagrams oppose two gain vectors with different orders whereas all oscillators are placed in common frequency bands. The higher order one

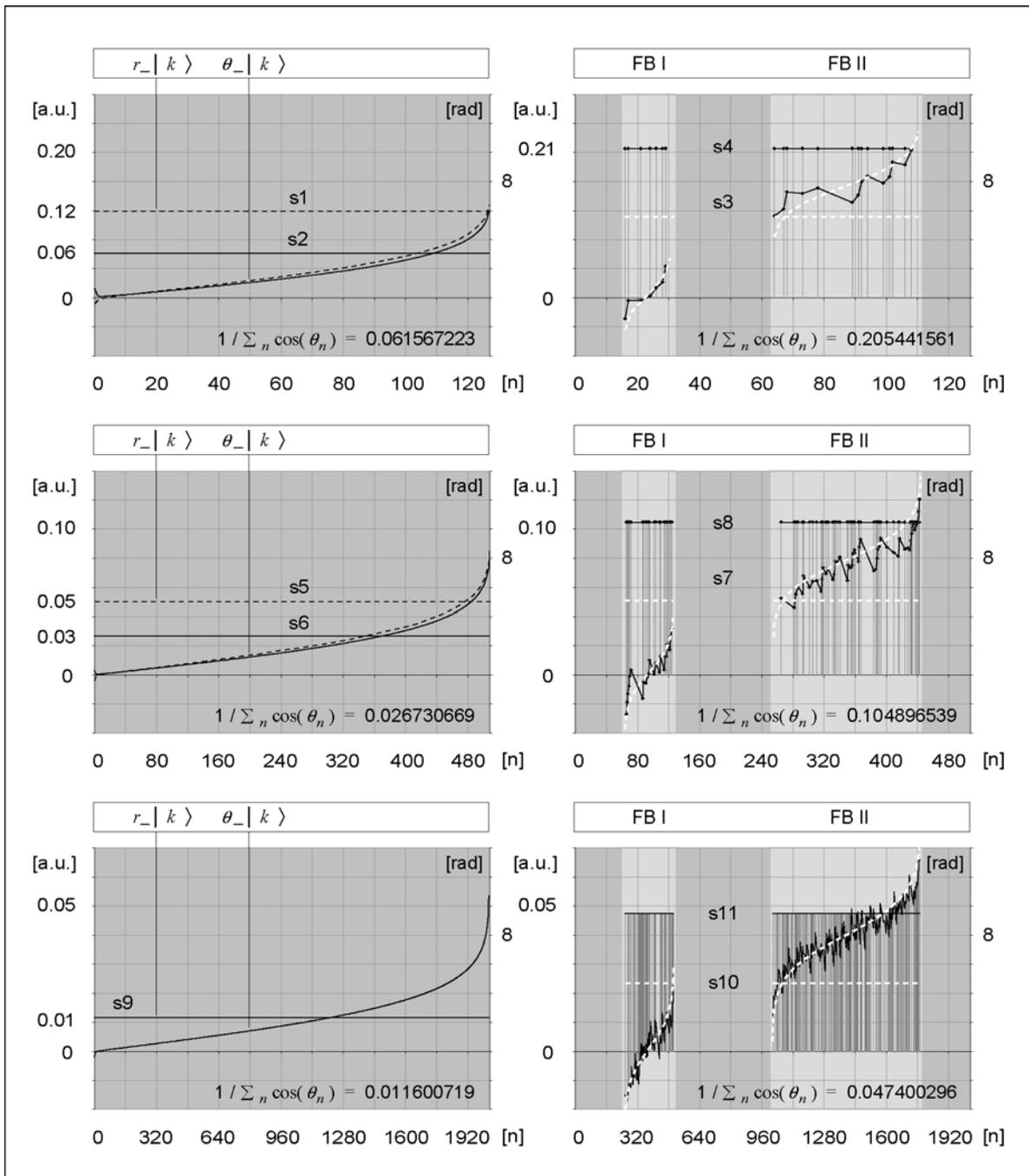


Figure 3.2.10: Magnitude and phase plots of the Kalman gain vector in different spectrum analysis scenarios. The scenario indexes refer to above table where the experimental boundary conditions are listed, amongst others the assignment of frequencies to harmonic orders. The top row shows the experiments at order 128, the middle one at 512 and the bottom one at 2048. The left hand plots originate from equally distributed NCO frequencies. The right hand plots depict the gain when the frequency band is segmented and the considered NCO frequencies are randomized. Dashed lines are references gathered from 1Hz frequency combs at the given orders. Theory advises the inverse sum of phase cosines to equal the gain amplitude which is in perfect agreement with the experimental data. Though phasing seems to be noisy in case of randomized frequency distributions, we substantiated that the ripple is a deterministic signature of the gain.

results from a segmented frequency comb that scales the bands. The lower order one is randomly taken from the comb structure. Vertical lines indicate the stochastic frequency selection in the highlighted bands. Once more the relation between magnitudes and phasing is checked. Even in case of segmentation and randomization we find excellent fidelity. Though phase encoding seems to be noisy we know it is deterministic and *a priori* given through the preselection of a numerically controlled oscillator ensemble. The high tracking performance of the Kalman filter [MAY79] results from the unique phasing of the gain as shown in the above profiles.

Finally we compare fig. 3.2.10 right and left hand graphs. It is clearly visible that the gain can be increased by reducing the frequency to narrow bands. Notice the span of phasing and the sum of cosines expand in parallel which is related to the very special profile. The increased gain speeds up the adaptation. So we find, that observing in frequency bands accelerates the spectral estimation. Once more this is in accordance with physical understanding, because local frequency refinement reduces the analytical order. It sets the stage to an advanced spectrum analysis technique, that can be matched to observed system. Remember refinement is not available in the Fast Fourier Transformation (FFT) which is the most popular spectrum analysis algorithm [GRUE01]. This is a unique property of the proposed method. Table 3.2.2 presents a summary of the variables and parameters used to formulate the adaptive spectrum analysis algorithm.

A brief review of the adaptive spectrum analysis filter assimilates the basics of the discrete Fourier transformation (DFT). The spectrum of a real signal is symmetric in frequency space. The magnitude shows axial- and the phasing centro-symmetry. To demonstrate the analogies evidence will be given that the new algorithm also yields conjugate pairs of spectral estimates. First we will expand an NCO ensemble in a symmetrical form to deal with positive and negative angular frequencies. Second we will give a proof that the gain remains phase-locked to the ensemble. Provided that we will finally derive from the error tuning mechanism that the adaptation has said symmetry and leads to conjugate complex Fourier coefficients. We arbitrarily choose the boundary conditions of above scenario 2 for the experiment. Every oscillator gets a counter-twiddling twin, positive as well as negative angular frequencies are generated. To compute the complex Kalman filter we must take into account that converting a $|ket\rangle$ into a $\langle bra|$ vector and vice versa demands for complex conjugation. As before we capture the phase encoding profile that transforms the input data coordinate wise into the Kalman gain. Moreover we grab the matrix of complex weights o_{ij} that transform the deterministic input data autocorrelation¹⁶ element wise into the autocovariance¹⁷. This approach is natural with respect to the correlations found in real domain and unfolds the inner complex structures. The weights are embedded in the autocovariance like (see equation 3.65):

16 The deterministic autocorrelation of the input data is given by the outer product $|x\rangle\langle x|$.

17 Motivation : see also equation 3.52

Adaptive Spectrum Analysis Filter ASA

initialize the computation by setting :

$$|\varpi\rangle = [\varpi_0, \varpi_1 \dots \varpi_{N-1}], |w\rangle = [w_0, w_1 \dots w_{2N-1}] = |0\rangle, \mathbf{O}_1 = \mathbf{I},$$

$T_s = \text{sampling period e.g. } 50\mu\text{s}$

do compute in a loop :

$$|x(n)\rangle = [x_0(n), x_1(n) \dots x_{2N-1}(n)]$$

with $x_{2i}(n) = \cos(n \cdot \varpi_i T_s)$ and $x_{2i+1}(n) = \sin(n \cdot \varpi_i T_s)$ $i = 0 \dots N-1$

$$|f(n)\rangle = \mathbf{O}_1(n-1) \cdot |x(n)\rangle$$

$$|k(n)\rangle = \frac{|f(n)\rangle}{\langle f(n)|x(n)\rangle}$$

and $\mathbf{O}_1(n) = \mathbf{O}_1(n-1) - |k(n)\rangle\langle f(n)| + \mathbf{I}$

evaluate $|g(n)\rangle = [g_0(n), g_1(n) \dots g_{N-1}(n)]$

with $g_i(n) = \sqrt{k_{2i}^2(n) + k_{2i+1}^2(n)}$ $i = 0 \dots N-1$

while $\sigma(g_i(n)) > \text{steady state limit e.g. } 1.0 \cdot 10^{-9}$

capture once :

$$|\theta\rangle = [\theta_0, \theta_1 \dots \theta_{N-1}]$$

where $\theta_i = \theta_i^k - \theta_i^x$ $i = 0 \dots N-1$

with $\theta_i^k = \begin{cases} \arctan(k_{2i+1}/k_{2i}) & k_{2i} > 0 \\ \text{sign}(k_{2i+1}) \cdot \pi/2 & \text{if } k_{2i} = 0 \\ \arctan(k_{2i+1}/k_{2i}) + \pi & k_{2i} < 0 \end{cases}$

$\theta_i^x = \begin{cases} \arctan(x_{2i+1}/x_{2i}) & x_{2i} > 0 \\ \text{sign}(x_{2i+1}) \cdot \pi/2 & \text{if } x_{2i} = 0 \\ \arctan(x_{2i+1}/x_{2i}) + \pi & x_{2i} < 0 \end{cases}$

which yields $g = 1 / \sum \cos \theta_i$		
for each forthcoming instance in time compute		
	$ x(n)\rangle = [x_0(n), x_1(n) \dots x_{2N-1}(n)]$	as shown above.
	$e^-(n) = r(n) - \langle x(n) w(n-1) \rangle$	
	$ k(n)\rangle = [k_0(n), k_1(n) \dots k_{2N-1}(n)]$	
with	$k_{2i}(n) = g \cdot (x_{2i}(n) \cdot \cos \theta_i - x_{2i+1}(n) \cdot \sin \theta_i)$ and	
	$k_{2i+1}(n) = g \cdot (x_{2i}(n) \cdot \sin \theta_i + x_{2i+1}(n) \cdot \cos \theta_i)$	
update	$ w(n)\rangle = w(n-1)\rangle + e^-(n) \cdot k(n)\rangle$	

par. / var.	definition	dimension
N	dimension of NCO ensemble	1
$ \varpi\rangle$	vector of spectrum analysis frequencies	N
$ x\rangle$	vector of NCO quadrature oscillations	$2N$
$ f\rangle$	filtered data vector	$2N$
$ k\rangle$	Kalman gain vector	$2N$
$ g\rangle$	vector of gain magnitudes	N
$ \theta\rangle$	phase encoding profile	N
\mathbf{O}_1	estimation error autocovariance matrix	$2N \times 2N$
g	learning rate	1
e^-	<i>a priori</i> adaptation error feedback weight	1
$ w\rangle$	vector of estimated Fourier coefficients	$2N$

Table 3.2.2: Summary of the adaptive spectrum analysis filter based on NCO ensembles. The first part shows the equations necessary to compute magnitudes and phasing of the Kalman gain with respect to the vector of quadrature oscillations. The second part guides how to reconstruct the gain based on the captured encoding profile and exploit it for spectral estimates. Part I is very time consuming due to an effort scaling up with $2N \times 2N$. Thus offline processing is standard while part II can be realized with an effort proportional to N which is apt for online adaptation.

$$\mathbf{O}_1 = \begin{bmatrix} o_{00}^1 & o_{01}^1 \cdot e^{i(\varpi_0 - \varpi_1)t} & o_{02}^1 \cdot e^{i(\varpi_0 - \varpi_2)t} & \dots & o_{0, N-1}^1 \cdot e^{i(\varpi_0 - \varpi_{N-1})t} \\ o_{10}^1 \cdot e^{i(\varpi_1 - \varpi_0)t} & o_{11}^1 & o_{12}^1 \cdot e^{i(\varpi_1 - \varpi_2)t} & & \\ o_{20}^1 \cdot e^{i(\varpi_2 - \varpi_0)t} & o_{21}^1 \cdot e^{i(\varpi_2 - \varpi_1)t} & o_{22}^1 & & \vdots \\ \vdots & & & \ddots & \\ o_{N-1,0}^1 \cdot e^{i(\varpi_{N-1} - \varpi_0)t} & \dots & & & o_{N-1, N-1}^1 \end{bmatrix} \quad (3.65)$$

The autocorrelation matrix as written in equation 3.66 is spanned by complex oscillators rotating at characteristic frequencies, that are determined by a certain combination of row and column index. Due to conjugation the angular frequency of an entity is given by the difference between two NCO frequencies. The latter are picked from the input data vector by former indexes. Since angular frequencies ascend by order the difference is negative for the top triangular matrix elements, positive for the bottom ones and zero for the diagonal. So the sense of rotation is clockwise for top and counterclockwise for bottom triangle while diagonal elements do not rotate.

$$|x\rangle\langle x| = \begin{bmatrix} 1 & e^{i(\varpi_0 - \varpi_1)t} & e^{i(\varpi_0 - \varpi_2)t} & \dots & e^{i(\varpi_0 - \varpi_{N-1})t} \\ e^{i(\varpi_1 - \varpi_0)t} & 1 & e^{i(\varpi_1 - \varpi_2)t} & & \\ e^{i(\varpi_2 - \varpi_0)t} & e^{i(\varpi_2 - \varpi_1)t} & 1 & & \vdots \\ \vdots & & & \ddots & \\ e^{i(\varpi_{N-1} - \varpi_0)t} & \dots & & & 1 \end{bmatrix} \quad (3.66)$$

Captured data sets are summarized in fig. 3.2.11. Editing with respect to the input data autocorrelation allows a differentiating view. Symmetry in the top row plots is eye catching for both magnitude and phase. Smooth gradients tell us the form of visualization is educatedly chosen. If we compute the filtered data and the Kalman gain vector we can expect oscillatory terms to interfere which forces both vectors to be phase locked to the NCO ensemble.

Explicated in formulas:

$$|f\rangle = \mathbf{O}_1 \cdot |x\rangle$$

and with $\langle o_i^1 |$ as the row i of the autocovariance \mathbf{O}_1

$$f_i = \langle o_i^1 | x \rangle = \sum_{j=0}^{N-1} o_{ij}^1 \cdot e^{i(\varpi_i - \varpi_j)t} \cdot e^{i\varpi_j t} = \sum_{j=0}^{N-1} \underline{o_{ij}^1} \cdot \underline{e^{i\varpi_i t}} = \underline{f} \cdot \underline{e^{i\theta_i}} \cdot \underline{e^{i\varpi_i t}} \quad (3.67)$$

Moreover $|k\rangle = \frac{|f\rangle}{\langle f|x\rangle}$

While we derive

$$\langle f | x \rangle = \sum_{j=0}^{N-1} f \cdot e^{-i\theta_j} \cdot e^{-i\varpi_j t} \cdot e^{i\varpi_j t} = \underline{f \cdot \sum_{j=0}^{N-1} e^{-i\theta_j}} = \underline{f \cdot g^{-1}}$$

this yields the gain coordinates

$$k_i = g \cdot e^{i\theta_i} \cdot e^{i\varpi_i t} \quad (3.68)$$

It follows in return that the integral update of the autocovariance impresses a magnitude profile on the matrix approximately inverse proportional to the angular frequency of an addressed element. This can be understood from

$$\mathbf{O}_1 = \sum (\mathbf{I} - |k\rangle\langle f|) \quad (3.69)$$

since all non-diagonal elements obey :

$$o_{ij} \cdot e^{i(\varpi_i - \varpi_j)t} = -g \cdot f \cdot e^{i(\theta_i - \theta_j)} \cdot \sum e^{i(\varpi_i - \varpi_j)t} \quad \text{with } i \neq j \quad (3.70)$$

So in light of
$$\sum e^{i(\varpi_i - \varpi_j)nT} \approx \frac{1}{T_s} \cdot \int e^{i(\varpi_i - \varpi_j)t} dt = -\frac{i}{(\varpi_i - \varpi_j) \cdot T_s} \cdot e^{i(\varpi_i - \varpi_j)t} \quad (3.71)$$

given the sampling period T_s

we find
$$o_{ij} = \frac{g \cdot f}{(\varpi_i - \varpi_j) \cdot T} \cdot e^{i(\theta_i - \theta_j + \pi/2)} \quad \text{with } i \neq j \quad (3.72)$$

In steady state the diagonal elements are constant which forces the filtered data and the Kalman gain amplitudes to fulfill (according to equation 3.66)

$$1 - g \cdot f = 0 \quad (3.73)$$

Thus equation 3.72 simplifies

$$o_{ij} = \frac{1}{(\varpi_i - \varpi_j) \cdot T_s} \cdot e^{i(\theta_i - \theta_j + \pi/2)} \quad \text{with } i \neq j \quad (3.74)$$

Said circular dependencies are clearly observable in the extracted secondary diagonal intersections shown below. As we compare the measured phase encoding profiles of the Kalman gain in complex and real domain we can state a perfect match on the positive frequency axis. Notice the delicate distortion in the phase plots on low order harmonics, rooted in the missing zero order NCO. The special quality is highlighted by an arrow. Idem signature is also contained in the real domain profile **s2** plotted in fig. 3.2.10 .

Obviously the Kalman gain coordinates are centro-symmetrical in phase :

$$\varpi_i = -\varpi_{N-i-1} \quad \text{respectively} \quad \theta_i = -\theta_{N-i-1}$$

From equation 3.68 we gather $k_i = k_{N-i-1}^*$ (3.75)

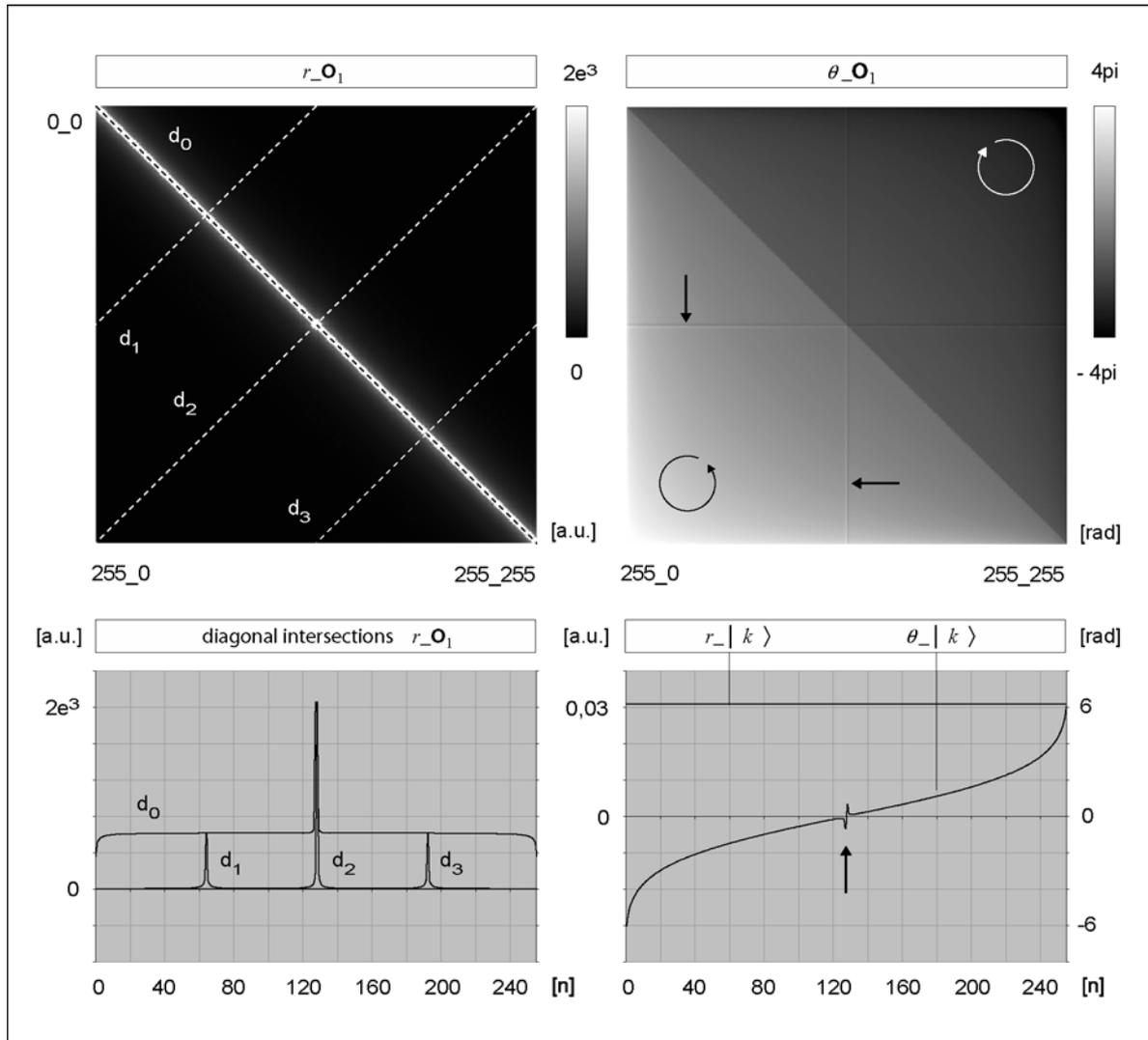


Figure 3.2.11: Magnitudes and phasing of the autocovariance matrix \mathbf{O}_1 and the Kalman gain vector $|k\rangle$ in case of complex Kalman filtering. The experiment is driven by a symmetric oscillator ensemble with clockwise and counterclockwise twiddling NCOs at conjugated frequencies. Magnitude and phase weights r and θ of the autocovariance are illustrated with respect to the input data autocorrelation matrix elements. The embedded complex oscillators in the top and bottom triangular matrices feature a characteristic sense of rotation which is indicated in the phase plot. The z -axis of the matrix diagrams is gray scale encoded as shown in the linear blend bars. Bottom left vector plot extracts diagonal intersections from the magnitudes data. Bottom right one depicts the centro-symmetric phase encoding profile of the complex Kalman gain vector that corresponds to the given symmetry of the oscillator ensemble.

Thus the tuning mechanism of the adaptive spectrum analysis filter (tab. 3.2.2)

$$|w(n)\rangle = |w(n-1)\rangle + e^-(n) \cdot |k(n)\rangle \quad (3.76)$$

with the complex spectral estimate $|w\rangle$ and the real ¹⁸ prediction error e^- adapts the Fourier coefficients with indexes i and $N-i-1$ in complex conjugated directions.

$$\Delta \varpi_i(n) = e^-(n) \cdot k_i(n)$$

$$\Delta \varpi_{N-i-1}(n) = e^-(n) \cdot k_{N-i-1}(n)$$

Since equation 3.75 rules

$$e^-(n) \cdot k_{N-i-1}(n) = \left(e^-(n) \cdot k_i(n) \right)^*$$

the adaptation steps are related to each other as shown by

$$\Delta \varpi_i(n) = \Delta \varpi_{N-i-1}^*(n)$$

and the spectral estimates also hold

$$\varpi_i(n) = \sum_n \Delta \varpi_i(n) = \sum_n \Delta \varpi_{N-i-1}^*(n) = \varpi_{N-i-1}^*(n) \quad (3.77)$$

The result helps to understand, why the complex Kalman gain amplitude falls to half the real one (compare figures 3.2.10 and 3.2.11). Now this is trivial to prove. We doubled the number of NCOs to create a symmetric oscillator ensemble but equation 3.77 tells us we kept the degrees of freedom since orders i and $N-i-1$ are dependent. Notice said two complex identification orders build one real spectral component in the time series under observation. Thus complex magnitudes must be half real ones to be consistent. As the relation also holds for complex and real gain magnitudes it will take equal time to adapt the Fourier coefficients no matter which domain is chosen for the mathematical description. This is very important to know because the computational expense of real adaptive spectrum analysis is half the complex one yet offering identical analysis performance.

To conclude : It was examined in the first subsection how the spectral components of a digitized time series may be estimated by means of an adaptive filter with minimal computational expense but maximal tracking performance. The new approach suggested here features Kalman filtering for that purpose and we call the method *adaptive spectrum analysis*. Thereby an arbitrary time series to be analyzed is assumed to result from a numerically controlled oscillator ensemble that stimulates a black box. The adaptive filter tries to estimate the complex weights in the box that yield the observed series at the box output resulting in an approximation of the discrete Fourier transform at the ensemble frequencies. It was explored in detail that

¹⁸ Remember from fig. 3.2.1 that the ASA filter yields complex estimates through the observation of a real time series. It follows, that the prediction error used as a tuning feedback for the adaptation, is a real number too.

the very special experimental boundaries allow to skip the most expensive computations from the Kalman filter and minimize the effort. Reduction goes from order N^2 to N which makes the algorithm highly attractive for online signal processing. The author showed the reduction is mathematically rooted in the filter variables reaching a phase-locked steady state with respect to the stimulus. Different type of criteria for steady state detection have been studied, amongst others \mathcal{H}_∞ norm and standard deviation of the Kalman gain vector magnitudes. Complex and real domain variants of adaptive spectrum analysis have been computed and compared with respect to data processing analogies and adaptation dynamics. It was found that both variants match on the positive frequency axis thus the adaptation dynamics are domain independent. It is remarkable that the complex filter just like the discrete Fourier transform yields a symmetric spectrum. Though from a mathematical point of view it is the more elegant candidate. But due to the fact that symmetric weights are dependent physical redundancies can be canceled. We implement the algorithm based on a real oscillator model to minimize the signal processing runtime. The optimal tracking performance is demonstrated later on in the experimental chapter 4, where the method is utilized in single line and broadband resonance characterization of a piezomechanical nano actuator.

3.2.2 Realtime Aspects of Digital Signal Processing

All spectral analysis experiments were realized on an Analog Devices ADSP-21160 SHARC digital signal processor. The SHARC¹⁹ offers the following hardware features to acquire and process input or generate output data streams (refer to fig. 3.2.12 for enumeration):

- ① an interface to an ADC²⁰
- ② internal memory space for data cache and storage
- ③ computational units for data processing
- ④ an interface to a DAC²¹

This paragraph explains the handling of data streams on that DSP. It starts with a description of the I/O interface²² to external hardware. The reader will be given a graphic illustration how data is bidirectionally streamed between the interface and the processor memory where it is cached for digital signal processing. Further we become acquainted with loading data into the computational units, executing the adaptive spectrum analysis filter in the DSP core and storing back results. The introduction includes a short description of the unique qualification of the SHARC architecture that speeds up the application to nowadays technology limit.

¹⁹ Super Harvard Architecture [ANA02]

²⁰ Analog to Digital Converter

²¹ Digital to Analog Converter

²² This is the digital input and output unit of the signal processor. It provides external data buses to converters or DSP clusters and external memory interface options.

I/O Processor

The interface to the external converters applied in this work is a time-divided-multiplexing (TDM) serial port. Serial data transfer between a connected converter and the I/O processor (IOP) features bitwise transmit and receive actions on a pair of single wire transmit and receive lines. Thus the transfer execution time is used to define start and end of a data word. Dividing the time line into slots with adequate word transfer length moreover allows to implement multichannel data streaming. Addressing a time slot means multiplexing input and output data channels which gives the technology its name (refer to fig. 3.2.13). Once configured in the IOP through software one bidirectional multichannel device can be handled per port. The SHARC offers two serial ports labeled SPORT in fig. 3.2.12 whereas only one is featured in the ASA filter application. The serial bus connection follows the industry standard described in [TOM99]. For information on how to configure the IOP to run in TDM mode read [ANA02 , ch.9]. If we aim to implement the scheme proposed in fig. 3.2.2 we need one receive channel to capture the output of the harmonic process and one transmit channel to generate the process stimulus. Latter is optional and only required, as the harmonic process represents a physically embodied system, that we want to stimulate and probe.

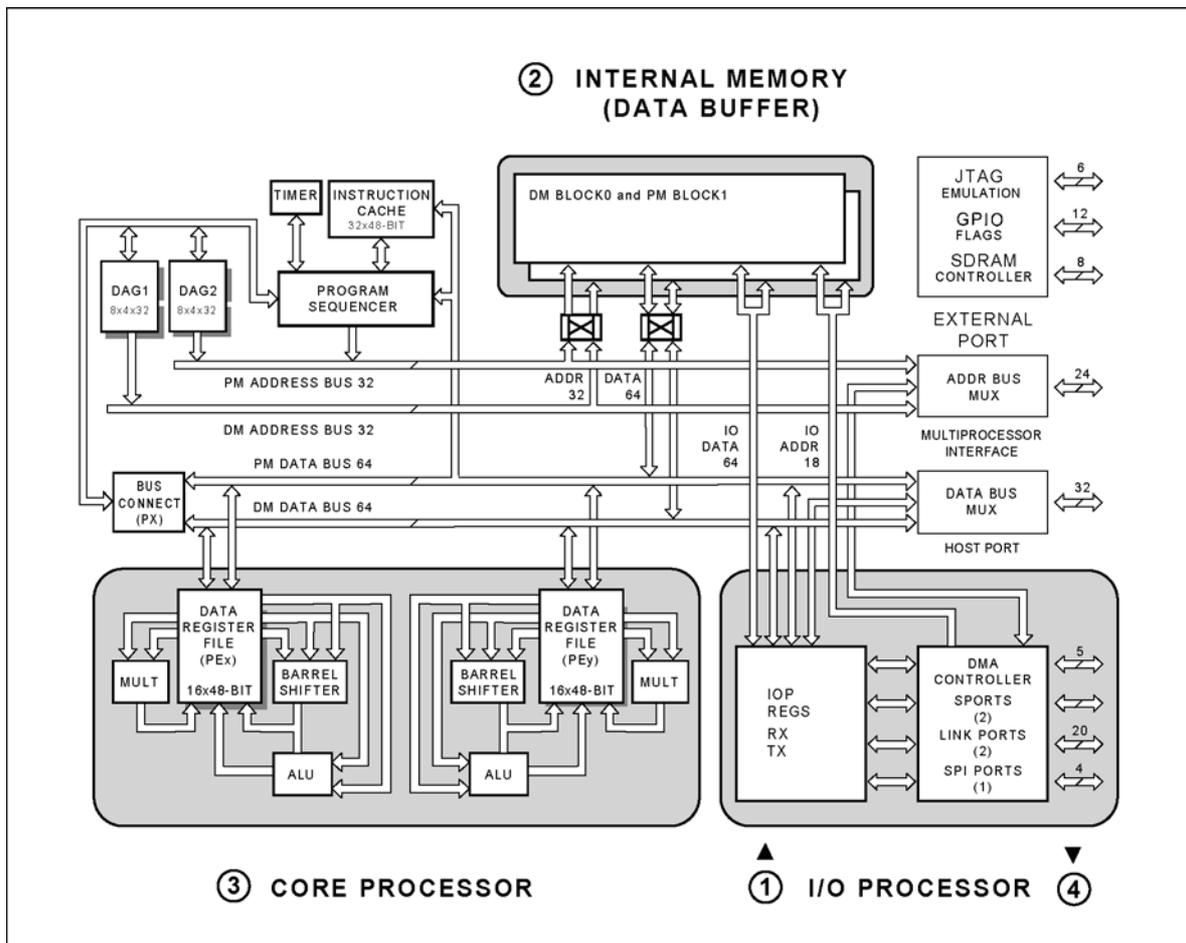


Figure 3.2.12 : ADSP21160 SHARC DSP block diagram [ANA02]

DSP Internal Memory

Each SPORT is equipped with one transmit and receive shift register, both buffered by a two-level first in first out (FIFO) stack. Input data is automatically latched from the receive shift register to the reception stack. Output data is loaded vice versa from the transmission stack to the transmit shift register as shown in fig. 3.2.13. Limited memory depth in IOP for serial bus data requires seamless service to avoid FIFO overflow. The architecture of the internal memory supports a sophisticated schedule. We can install serial bus input and output data buffers in software to continuously fetch multichannel data from the reception stack respectively push it to the transmission one as illustrated. Transfer is automatically done from SPORT to said buffers and vice versa by so called direct memory access (DMA). That is a common feature available in all modern DSPs [HEI99].²³ A multichannel ADC may digitize up to 16 analog input signals in parallel. Each sampling period a frame of 16×16 bit data is acquired and transmitted from the converter to the DSP via serial bus as depicted. There is a one-to-one correspondence between the address of a sample in the buffer and the related time slot on the serial bus. Analog signal generation features the same protocol which is described more detailed in [TOM99]. Transmission and reception frames are synchronized. Thus 16×16 bit data may be streamed to a multichannel DAC based on identical timing. To match specifications input and output buffer are defined to have length 16. Buffers are updated at the converters sampling rate. For a discussion of read and write timing issues read the cited reference. The number of signal channels multiplexed in adaptive spectrum analysis depends on the given experimental scenario. If stimulation is done by the DSP, one transmit channel has to be addressed. Probing takes one receive channel. Case stimulus is generated externally it can be captured and analyzed by means of a second one. Time slots reserved for the application are slot 3 and 4. All other slots are omitted.

Notice buffers are continuously rewritten. So in general to keep old data available for DSP computations it must be cached. In most applications new input samples have to be copied as shown in the flowchart from the serial bus input buffer to a signal processing input cache as to prevent data loss. The same applies for the output direction if data history counts. The minimal size of a cache depends on the application software demand.²⁴ In case of ASA filtering the size is one. Thus cache is optional. But the strategy to implement software caches is well documented [SMI99] and it also very apt to address the vector quantities in the filter algorithm, like the input data vector $|x\rangle$ and the Kalman gain vector $|k\rangle$. Thus we will have a closer look on it. For efficient memory access it is common to implement caches in form of rings, that are addressed in circular mode. To explain : The SHARC memory contains 4 Megabit of SRAM, organized in two blocks of equal size, which are specified as block 0 and block 1 by the manufacturer (see fig. 3.2.12). Each may store combinations of

²³ The configuration parameters of a single DMA data transfer, like source and target address, are stored in a Transfer Control Block (TCB). A TCB has to be configured by the embedded software engineer. The I/O processor reads the settings from the TCB and executes the transfer. Iteration is possible by reusing the parameters stored in the TCB. This procedure is called DMA chaining. More information on that topic can be found in [ANA02 , p.6-12, p.6-25, p.6-28].

²⁴ annotation : if the latest 100 values are in process the cache size must be at least 100.

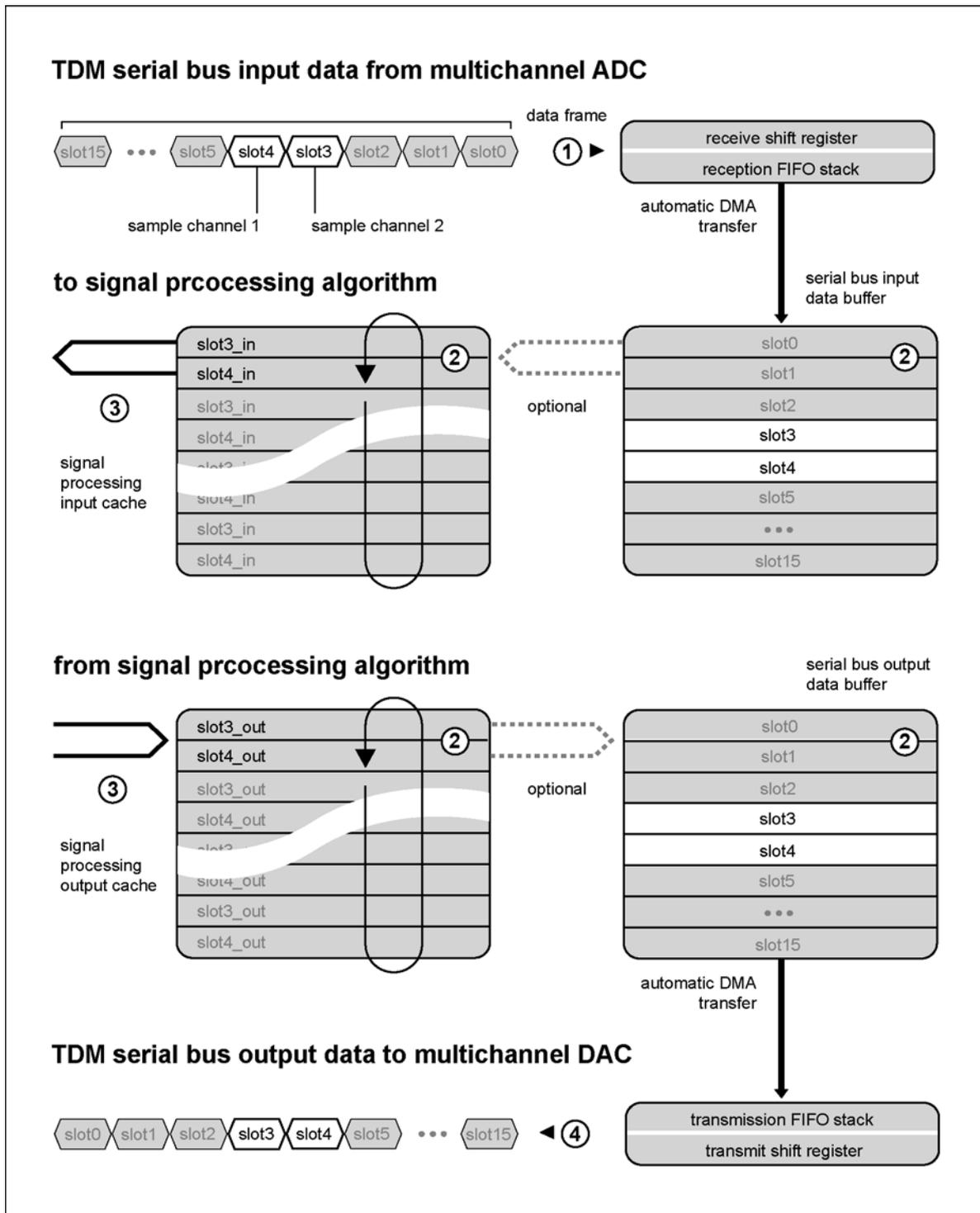


Figure 3.2.13: Common DSP serial data I/O flowchart

data and software code instructions. Memory access switches an electronic port between a register and one of the data buses which allows to transfer data. The path is encoded in the address lines that build the corresponding address bus. The ring elements are allocated in a connected memory address range defined by the ring base and length. Circular mode stands for an incrementation of access addresses by a wrap around modifier. The range is guarded from a modulo logic. It follows ring access

requires to compute these target addresses given base, length, actual index and modifier. The SHARC provides that by hardware in form of two data address generators (DAG , see fig. 3.2.12 top left corner) which compute the addresses in parallel to the core application. Circular buffering is said to run with zero overhead in DSP jargon. DAG1 is related to block 0 and DAG2 to block 1 addressing whereas each can handle 8 circular buffer indexes. Concerning the ASA filter the address generators manage coordinate access to the different vector quantities. Programming stall-free data streams on the microcomputer is discussed in depth at [ANA02 , ch.5].

Core Processor

As we have seen in chapter 3.1 the digital signal processor has to execute very comprehensive algorithms. This is common in many DSP applications [SMI99]. Thus commercial processors are optimized with respect to the standard algorithms on custom demand. Amongst others the fast Fourier transformation, digital filtering and signal convolution. All the variety of algorithms share the need to perform a series of multiplications and accumulations (MAC) as fast as possible. The adaptive spectrum analysis filter does not make an exemption here. To give an example we take the tuning of the complex spectral estimates (tab. 3.2.2) coordinate wise

$$w_i(n) = w_i(n-1) + e^-(n) \cdot k_i(n) \quad \text{for} \quad i = 0 \dots 2N-1 \quad (3.78)$$

whereas N represents the model order. The hardware instance that executes these MAC operations is the core processor. The core of the SHARC contains two identical data processing elements labeled PEx and PEy (notice fig. 3.2.12). Each one consists of a data register file to hold input and output values plus three independent computation units that cover total and difference formation, multiplication and bit manipulation.²⁵ Distributing tasks minimizes the computation time in a very smart way. The processing element PEy is restricted to run in lock-step with PEx. Both execute identical code instructions, but process different kind of input data. This feature is called single instruction multiple data (SIMD) processing mode. The programmer explicitly specifies the operation for core x, enables the SIMD mode on the DSP by software and all computations will be executed in parallel on core y. The same applies for data moves. Memory is addressed explicitly for PEx and implicitly for PEy input data. For more information read [ANA99] . Obviously the SIMD feature is ideal for the adaptive spectrum analysis filter, since odd- and even- indexed vector coordinates may be updated in parallel. For instance real and imaginary part of complex spectral estimates may be processed simultaneously, which halves the computational expense.

Assume the *a priori* adaptation error feedback weight e^- already being evaluated and intermediately stored in one of the data registers. Moreover the spectral estimate vector $|w\rangle$ and the Kalman gain vector $|k\rangle$ being allocated in the internal memory as described before. Then the DSP :

²⁵ Addition and subtraction is done by the arithmetic logic unit (ALU), multiplication by the multiplier (MULT) and bit manipulation by the barrel shifter (SHIFT). These abbreviations are used in all architecture maps shown here.

- loads equal coordinates of $|w\rangle$ and $|k\rangle$ from memory into the core data register file ; the vectors are addressed in circular mode as described above ; even coordinates are processed in PEx and odd ones in PEy.
- executes the multiplication of e^- and k_{2i} respectively k_{2i+1} from data register files x and y ; results are kept in the core.
- accumulates the updates of w_{2i} and w_{2i+1} in parallel.
- stores the dual core update of the spectral estimate vector $|w\rangle$ back to memory.

Instructions have to be passed in a loop till all coordinates of $|w\rangle$ are processed. Tasks can be parallelized. Data moves and core computations may complete in a single core cycle. The resulting schedule – known as software pipelining – is depicted in tab. 3.2.3. Pipelined code design needs educated programming concepts for efficient use of the DSP hardware. Tutorial information can be found in [ANA94] [ANA99] [ANA02, ch.2].

As we know dual core computation automatically loads neighbored coordinates of vector quantities to the processor elements PEx respectively PEy. The complex coordinates of $|w\rangle$ and $|k\rangle$ are written in an real–imaginary interleaved form. That means above pipeline handles a *single complex line* of the spectral comb per loop cycle. Notice the complex gain vector $|k\rangle$ of length $2N$ to be represented by real phasing $|\theta\rangle$ and magnitudes $|r\rangle$ both of length N . SIMD addressing will pick two different frequencies. This inherently leads to parallel processing of *two complex lines* per loop which has to be considered in the reconstruction of the Kalman gain $|k\rangle$ from the NCO ensemble $|x\rangle$.²⁶ A more elaborated software pipeline is required for that task. It significantly differs from tab. 3.2.3 and is summarized in tab. 3.2.4. Free slots are filled to gather the system response prediction y with zero overhead, thus after $|k\rangle$ being reconstructed the *a priori* adaptation error feedback weight e^- may be evaluated and the adaptation of $|w\rangle$ be started immediately.

Application code must be stored in memory. The program sequencer fetches the instructions to decode them and control the core computations (memorize fig. 3.2.12). Notice fetches occur on PM data bus. This constraint is given by the Harvard architecture. So latter is normally locked because the sequencer continuously fetches instructions. To allow for data moves on the PM data bus one expects the occurrence of penalty cycles for arbitration that degrade the computational performance. However if instructions are encapsulated in a software loop as suggested in tab. 2.2.3 and tab. 2.2.4 the SHARC only fetches them once in the first and automatically caches them for the remaining loop cycles. So the PM data bus is unlocked for moves from memory to core and vice versa until the loop expires. The instruction cache significantly enhances the data throughput in number crunching algorithms like ASA.

²⁶ annotation : one line processed in PEx , other in PEy .

cycle	multiplication	accumulation	DM bus data transfer	PM bus data transfer
	e^- stored in core		$ k\rangle$ allocated in block 0	$ w\rangle$ allocated in block 1
0	—	—	load k_0 from block 0 into core	load w_0 from block 1 into core
1	$\Delta w_0 = e^- \cdot k_0$	—	—	—
2	—	$w_0 += \Delta \varpi_0$	load k_2 from block 0 into core	load w_2 from block 1 into core
3	$\Delta w_2 = e^- \cdot k_2$	—	—	store w_0 back to block 1
4	—	$w_2 += \Delta \varpi_2$	load k_4 from block 0 into core	load w_4 from block 1 into core
5	$\Delta w_4 = e^- \cdot k_4$	—	—	store w_2 back to block 1
6	—	$w_4 += \Delta \varpi_4$	load k_6 from block 0 into core	load w_6 from block 1 into core
7	$\Delta w_6 = e^- \cdot k_6$	—	—	store w_4 back to block 1
8	—	$w_6 += \Delta \varpi_6$	load k_8 from block 0 into core	load w_8 from block 1 into core
\vdots	\vdots	\vdots	\vdots	\vdots
2N-2	—	$w_{2N-4} += \Delta \varpi_{2N-4}$	load k_{2N-2} from block 0 into core	load w_{2N-2} from block 1 into core
2N-1	$\Delta w_{2N-2} = e^- \cdot k_{2N-2}$	—	—	store w_{2N-4} back to block 1
2N	—	$w_{2N-2} += \Delta \varpi_{2N-2}$	—	—
2N+1	—	—	—	store w_{2N-2} back to block 1

Table 3.2.3: Update schedule of spectral estimates in the adaptive spectrum analysis filter in PEX. The programming technique of executing data moves in parallel to the core computations is called software pipelining. The history of the estimate coordinate w_0 is highlighted. Pipeline is filled in the first two and emptied in the last two core cycles. Brackets mark repeating instructions dispatched to the core that are encapsulated in a software loop.

cycle	multiplication	accumulation	DM bus data transfer	PM bus data transfer
	none	model response $y = 0$	sine table stored in block 0	$ w\rangle$ allocated in block 1
		optional	$ x\rangle$ represented by phase vector $ \varphi\rangle$ allocated in block 0	phase vector gradient $ \Delta\varphi\rangle$ allocated in block 1
		$ k\rangle$ represented by phase vector $ \kappa\rangle$ temporarily stored in core	$ k\rangle$ phasing $ \theta\rangle$ and magnitudes $ r\rangle$ allocated in block 0	optional

Table 3.2.4 a : Kalman gain reconstruction - object allocation and initialization.

cycle	multiplication	accumulation	DM bus data transfer	PM bus data transfer
0	—	—	load φ_0 from block 0 into core	load $\Delta\varphi_0$ from block 1 into core
1	—	ramp phase $\varphi_n += \Delta\varphi_n$	load θ_0 from block 0 into core	load w_0 from block 1 into core
2	—	reconstruction of $\kappa_n = \varphi_n + \theta_n$	store φ_0 back to block 0	load w_2 from block 1 into core
3	—	get pointer 1 and pointer 2 to sine table from phase φ_0 and φ_1	broadcast r_0 block 0 to core	—
4	—		broadcast r_1 block 0 to core	—
5	—	get pointer 3 and pointer 4 to sine table from phase κ_0 and κ_1	set pointer 1 in DAG	—
6	—		set pointer 2 in DAG	—
7	—	—	set pointer 3 in DAG	—
8	—	—	set pointer 4 in DAG	—

Table 3.2.4 b : to be continued

9	—	—	load x_0 into core by pointer 1	—
10	$\Delta y = w_0 \cdot x_0$	—	load x_2 into core by pointer 2	—
11	$\Delta y = w_1 \cdot x_1$	$y += \Delta y$	load k_0 into core by pointer 3	—
12	weight $k_0 * = r_0$	$y += \Delta y$	load k_2 into core by pointer 4	—
13	weight $k_2 * = r_1$	—	store k_0 back to block 0	—
14	—	—	store k_2 back to block 0	—
⋮	⋮	⋮	⋮	⋮

Table 3.2.4 b : Update schedule of Kalman gain reconstruction – first loop cycle. The loop body is framed. A history of gain reconstruction is highlighted while non marked instructions refer to system response prediction.

cycle	multiplication	accumulation	DM bus data transfer	PM bus data transfer
... – 15	—	—	load φ_{N-2} from block 0 into core	load $\Delta\varphi_{N-2}$ from block 1 into core
... – 14	—	ramp phase $\varphi_{N-2} += \Delta\varphi_{N-2}$	load θ_{N-2} from block 0 into core	load w_{2N-4} from block 1 into core
... – 13	—	reconstruction of $\kappa_{N-2} = \varphi_{N-2} + \theta_{N-2}$	store φ_{N-2} back to block 0	load w_{2N-2} from block 1 into core
... – 12	—	get pointer 1 and pointer 2 to sine table from phase φ_{N-2} and φ_{N-1}	broadcast r_{N-2} block 0 to core	—
... – 11	—		broadcast r_{N-1} block 0 to core	—
... – 10	—	get pointer 3 and pointer 4 to sine table from phase κ_{N-2} and κ_{N-1}	set pointer 1 in DAG	—
... – 9	—		set pointer 2 in DAG	—

Table 3.2.4 c : to be continued

...-8	—	—	set pointer 3 in DAG	—
...-7	—	—	set pointer 4 in DAG	—
...-6	—	—	load x_{2N-4} into core by pointer 1	—
...-5	$\Delta y = w_{\gamma N-4} \cdot x_{\gamma N-4}$	—	load x_{2N-2} into core by pointer 2	—
...-4	$\Delta y = w_{\gamma N-3} \cdot x_{\gamma N-3}$	$y += \Delta y$	load k_{2N-4} into core by pointer 3	—
...-3	weight $k_{2N-4} * = r_{N-2}$	$y += \Delta y$	load k_{2N-2} into core by pointer 4	—
...-2	weight $k_{2N-2} * = r_{N-1}$	—	store k_{2N-4} back to block 0	—
...-1	—	—	store k_{2N-2} back to block 0	—
$\frac{15N}{2}$	complete system response prediction adding PEx and PEy accumulation and store back result to block 0 in 3 cycles			

Table 3.2.4 c : Update schedule of Kalman gain reconstruction – last loop cycle.

We can state, that the reconstruction of the Kalman gain $|k\rangle$ and the model prediction at given order N takes $7.5N+3$ core cycles plus 6 cycles overhead. The computation of the *a priori* estimation error e^- is done in 4 and the adaptation of the complex spectrum $|w\rangle$ in $2N+2$ cycles.

Thus the overall effort E of the adaptive spectrum analysis filter is given per loop (for code reference see **appendix A** - section 8) :

$$E = 9.5N + 15 \quad \text{for} \quad \text{model order } N \quad (3.79)$$

Now that we know, how the method technically works, we start to explore in the next chapter, how it can be used in the laboratory for single line and broadband spectral characterization experiments. We will quantify the extreme tracking performance of the adaptive spectrum analysis filter, that shows up in a lightning manner.