Data-driven identification of Lax-integrable partial differential equations
Using the nonlinear Fourier transform and conserved quantities

de Koster, P.B.J.

DOI
10.4233/uuid:4f7650a0-2204-4493-a9cd-586e240b37a9

Publication date
2025

Document Version
Final published version

Citation (APA)
de Koster, P. B. J. (2025). *Data-driven identification of Lax-integrable partial differential equations: Using the nonlinear Fourier transform and conserved quantities*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:4f7650a0-2204-4493-a9cd-586e240b37a9

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Data-Driven Identification of Lax-Integrable Partial Differential Equations

Using the Nonlinear Fourier Transform and Conserved Quantities

Data-Driven Identification of Lax-Integrable Partial Differential Equations

Using the Nonlinear Fourier Transform and Conserved Quantities

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology
by the authority of the Rector Magnificus, prof. dr. ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates
to be defended publicly on
Monday 2 June, 2025 at 12:30

by

Pascal Bastiaan Johannes DE KOSTER

Master of Science in Applied Mathematics, Delft University of Technology, the Netherlands
born in Eindhoven, the Netherlands

This dissertation has been approved by the promotors.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof. Dr.-Ing. S. Wahls,	Karlsruhe Institute of Technology, Germany / Delft University of Technology, *promotor*
Dr. M. Kok,	Delft University of Technology, *promotor*
Independent members:	
Prof. dr. H.M. Schuttelaars	Delft University of Technology
Prof. dr. P.G. Kevrekidis	University of Massachusetts Amherst
Prof. Dr.-Ing. N. Hanik	Technical University of Munich, Germany
Dr. F. Maucher	Delft University of Technology
Prof. dr. ir. B. De Schutter	Delft University of Technology, reserve member

Keywords: Lax pair, integrable partial differential equations, identification, conserved quantities, solitons, nonlinear Fourier transformation, nonlinear Schrödinger equation, Korteweg-de Vries equation

Printed by: Drukkerij Haveka

Front & Back Allegory on the particle-like behavior of soliton-supporting water wave equations, such as the Korteweg-de Vries equations. Like pool balls, solitonic waves move through a medium and 'collide' elastically with other solitonic waves. The solitons emerge from any head-on collision unscathed, except for a small jump in their location. Created by Marc Estalilla, Pascal de Koster and Lars Frietman, using ChatGPT.

ISBN 978-94-6518-060-1

An electronic copy of this dissertation is available at
https://repository.tudelft.nl/.

CONTENTS

SUMMARY

Nonlinear partial differential equations (PDEs) are generally hard to solve, but over the past 65 years the notions of Lax integrability and *nonlinear Fourier transforms* (NFTs) have been developed to solve a large class of so called *Lax-integrable* partial differential equations. If we are able to find a suitable pair of linear operators, consisting of a spectral operator and a propagation operator, a so-called *Lax pair*, we are able to analytically solve the nonlinear PDE. If this pair of linear operators is a correct Lax pair for the PDE, then the spectral operator can be used to define an NFT, which transforms the signal from the physical domain to a PDE-specific spectral domain, in which all spectral components are independent, similar to the regular Fourier transform for linear problems. In the spectral domain, the transformed signal can be propagated using the linear propagation operator, after which the propagated signal can be transformed back into the physical domain, hence solving the nonlinear PDE.

The presence of a Lax pair can thus be used to solve a nonlinear PDE, but it also yields insight into the underlying dynamics of the system. Therefore, the identification of a Lax pair for a system is of great value. However, there is no general method to find a Lax pair given an evolution equation. Systems may also be non-integrable, and therefore a corresponding Lax pair may not even exist. To the best of our knowledge, no practical general methods are known in the literature to determine whether or not a system is Lax-integrable. Since the general problem is very complex, one may focus on more specific cases for practical purposes, in which a Lax-integrable PDE is sought using measurement data.

In this thesis, we therefore focus on data-driven identification of Lax-integrable partial differential equations (PDEs), specifically those of the AKNS-type. The primary objective is to find a Lax-integrable PDE that best explains given experimental measurement data, enabling a comprehensive analysis using the nonlinear Fourier transform (NFT). While real-world systems may not be exactly Lax-integrable due to imperfections, many are well approximated by such equations, including scenarios such as fiber optical wave propagation, surface wave propagation in shallow water canals, and mechanical wave propagation in coupled pendulums.

A significant portion of the thesis is devoted to identifying parameters for several widely applied Lax-integrable nonlinear PDEs, such as the Korteweg-de Vries equation (KdV) or the nonlinear Schrödinger equation (NLSE). The approach involves comparing the NFT of the waveform at various stages of evolution, and finding the parameters in the Lax pair at which the nonlinear spectra remains constant. This approach does not require any numerical propagation. Furthermore, no derivatives in the direction of evolution are required. A comparative analysis with propagation-based methods highlights the advantages and limitations of the proposed techniques.

Further contributions include the investigation of speeding up the discrete spectrum

part of NLSE-NFT, crucial for the application of the method, for example in optical fiber transmissions. The computational complexity of the discrete part of the NLSE-NFT typically scales super-linear with the time-bandwidth product of the considered signal. We propose a method to window the full signal in time and space, resulting in multiple signals with much smaller time-bandwidth products. The low eigenvalues are affected due to this windowing, but these can be efficiently detected and removed in our proposed method. These low eigenvalues are usually of less interest as they only represent a fraction of the total energy. The higher eigenvalues represent the largest part of the energy, and these are preserved very well despite the windowing. Furthermore, the combined computational cost of applying the NFT to all windowed signals is much lower than applying a single NFT to the full signal.

Finally, this thesis also explores direct parameter identification of AKNS-type Lax pairs by comparing conserved quantities. Instead of comparing the spectra resulting from an NFT, this method considers the global conserved quantities, related to the spectral operator of the Lax pair. Given that the spectral operator indeed fits the underlying evolution equation well, the global quantities are also conserved well. We demonstrate that this principle can be used to correctly identify which AKNS-type PDE fits a given measurement data best. In particular, the method only requires two snapshots of data, which allows it to be applied even when no derivatives in the direction of evolution are known.

1

INTRODUCTION

1.1. BRIEF HISTORY OF LAX-INTEGRABLE PARTIAL DIFFERENTIAL EQUATIONS

In this thesis, we focus on identifying Lax-integrable nonlinear partial differential equations (PDEs) that describe given measurement data best. Let a PDE be given by

$$u_t = K(u), \tag{1.1}$$

where $u(x,t)$ is the (typically scalar) function value, also referred to as the waveform $u(x)$ for a snapshot at a fixed t, or the wave field for the full solution $u(x,t)$. When dealing with simulated or measurement data, a waveform $u(x)$ may also be referred to as the signal. Next, x is the (possibly multidimensional) 'spatial' variable, t the variable of evolution, and $K(u)$ a transformation of $u(x,t)$ and spatial derivatives of u (possibly of higher order). Here and later, subscripts denote derivatives.

Nonlinear PDEs (i.e., with a nonlinear operator $K(u)$) are often notoriously hard to solve, and so far no general method has been developed to solve nonlinear PDEs. This forms a strong contrast with linear PDEs, such as the heat equation $u_t = u_{xx}$, which can be solved analytically using the Fourier transform, given suitable boundary conditions. This method transforms a function from the physical domain to a spatial frequency (i.e., wavenumber) domain, in which each wavenumber is associated with a sinusoidal component. Each sinusoidal components evolves independently, and usually follows a simple propagation relation. The solution can finally be obtained by applying the inverse Fourier transform to the evolved sinusoidal components. This process is also illustrated in Fig. 1.1. Unfortunately, the same method cannot be applied to nonlinear PDEs, because the sinusoidal components do not propagate independently due to the nonlinearity. It thus came as a huge surprise when a nonlinear PDE was solved with a novel method that conceptually resembled Fourier's method. This discovery would lay the foundation for the theory of Lax-integrability.

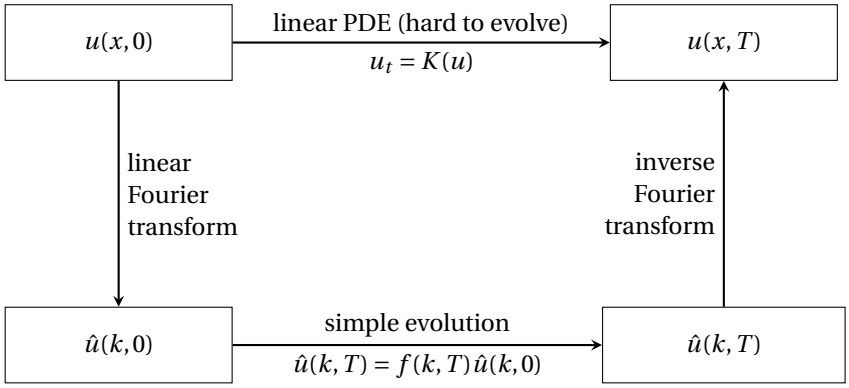


Figure 1.1: Solving a linear PDE with the classic Fourier transform, by first transforming the function from the physical domain to a wavenumber domain, where each wavenumber is associated with a sinusoidal function. The evolution of the linear PDE becomes a simple multiplication in the wavenumber domain. Finally, the inverse Fourier transform allows the function to be reconstructed in the physical domain.

1.1.1. THE FIRST DISCOVERED LAX-INTEGRABLE PDE: THE KORTEWEG-DE VRIES EQUATION

In 1967 Gardner, Greene, Kruskal and Miura presented a method [1] to analytically solve a weakly nonlinear PDE known as the Korteweg-de Vries equation (KdV), which is known to govern free-surface water waveforms in shallow canals [2, 3]. In particular, the KdV is known to support so-called *solitons*, localized particle-like wave packets that propagate without changing shape. Solitons are the result of dispersive and nonlinear effects exactly balancing out. Furthermore, solitons can overtake each other and emerge from the interaction unchanged, except for a phase shift, which is illustrated in Fig. 1.2 for a double-soliton solution of the KdV. The large, fast soliton overtakes the slower soliton, and both emerge unscathed from the collision except that the faster soliton is pushed further ahead while the slower soliton is pushed back.

Gardner et al. showed that the KdV could be solved by transforming the waveform to a spectral domain, in which all spectral components evolved independently as if it were a linear PDE. This spectral domain consisted of two parts: a continuous spectrum, which corresponded to dispersive wave components; and a discrete spectrum, which consisted of isolated points representing the solitons inside the solution.

This was a major breakthrough in the field of soliton theory and the field of completely integrable systems, which studies the existence of explicit solutions of PDEs. Solitary waves in shallow canals had been reported as early as 1845 by John Scot Russell [4], analytic soliton solutions of the KdV were shown to exist already in 1895 [2], and several numerical experiments halfway the 20th century also showed that KdV-governed waveforms could contain multiple solitonic components [5, 6]. However, thus far no systematic approach to calculate the exact soliton content had been found, also because low-energy solitonic components can easily be hidden within the waveform.

Gardner et al. showed that the KdV could be solved by association with an auxiliary lin-

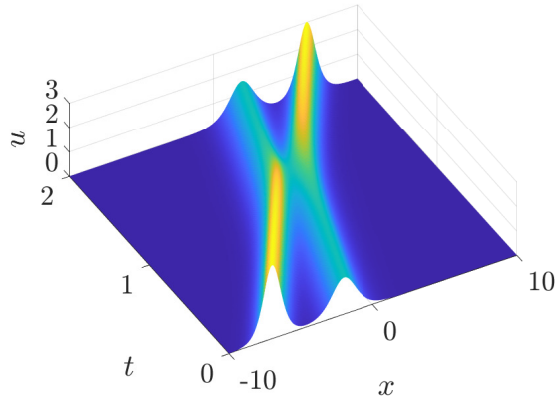


Figure 1.2: A double-soliton solution of the KdV, in which a fast and high soliton overtakes a lower and slower soliton.

ear equation, namely the Schrödinger eigenvalue problem, in which the KdV-governed waveform plays the role as potential function [1]. Let the KdV with vanishing boundary conditions by given by

$$u_t = -u_{xxx} - 6uu_x, \qquad \text{with} \quad u(x,t) \xrightarrow{x\to\pm\infty} 0 \text{ sufficiently fast} \quad \text{(KdV, vanishing BCs)}. \tag{1.2}$$

The KdV may then be associated with the Schrödinger eigenvalue problem, in which the potential $u(x,t)$ is governed by the KdV:

$$\begin{aligned} &\phi_{xx} - u\phi = -\lambda^2\phi, \quad \text{(Schrödinger eigenvalue problem)}, \\ &\text{with} \quad \alpha(\lambda,t)e^{-i\lambda x} \xleftarrow{x\to-\infty} \phi(x,t) \xrightarrow{x\to+\infty} e^{-i\lambda x} + \beta(\lambda,t)e^{+i\lambda x}. \end{aligned} \tag{1.3}$$

where $u(x,t)$ denotes the KdV-governed wave field, $\phi(x,t;\lambda)$ the eigenfunction of the Schrödinger equation, λ the spectral parameter, while $-\lambda^2$ plays the role of the energy level in the Schrödinger equation. Finally, $\alpha(\lambda,t)$ and $\beta(\lambda,t)$ denote the scattering coefficients, which are called as such due to their connection with a classical physical scattering problem, in which a wave comes from the right ($e^{-i\lambda x}$), which is partially transmitted to the left (with coefficient $\alpha(\lambda,t)$), and partially reflected back to the right ($\beta(\lambda,t)e^{+i\lambda x}$).

The Schrödinger eigenvalue problem in Eq. 1.3 allows a finite number of finite-energy solutions ϕ_k, $k = 1,\dots,K$, where each corresponds to an eigenvalue λ_k on the positive imaginary axis. Most importantly, Gardner et al. showed that these eigenvalues are time-invariant, if the potential $u(x)$ evolves according to the KdV equation, as well as that each level $-\lambda_k^2 = E_k > 0, E_k \in \mathbb{R}$ corresponds to a soliton amplitude when all solitons had separated after propagating for a sufficiently long time.

Furthermore, the authors showed that for every $\lambda \in \mathbb{R}, \lambda > 0$, a bounded solution (finite power, but not finite energy) exists for Eq. 1.3. The left side of the Schrödinger eigenvalue problem may be considered in functional analysis as the unbounded linear operator $L = \partial_x^2 - u$, where ∂_x is the spatial derivative operator. The eigenvalues λ_k correspond-

ing to finite-energy solutions ϕ may then be interpreted as the discrete spectrum, while the continuous spectrum then consisted of all λ on the positive real axis. It was later shown by AKNS that in the continuous spectrum, λ played a similar role as the linear wavenumber k in the linear Fourier spectrum [7].

Finally, given that $u(x,t)$ is governed by the KdV equation, the scattering coefficients $\alpha(\lambda,t)$ and $\beta(\lambda,t)$ (for both the discrete and the continuous spectrum) evolve over time according to a simple relation. When the discrete and continuous spectra are supplemented with the scattering coefficients, the waveform $u(x)$ can be reconstructed using a process of inverse scattering.

The KdV equation can thus be solved as illustrated in Fig. 1.3, similarly to the method of Fourier. First the waveform is transformed to the nonlinear spectral domain, in which the spectral components propagate independently, according to a simple dispersion relation $f(\lambda)$. Finally, the propagated spectral components are transformed back into the propagated waveform $u(x,T)$.

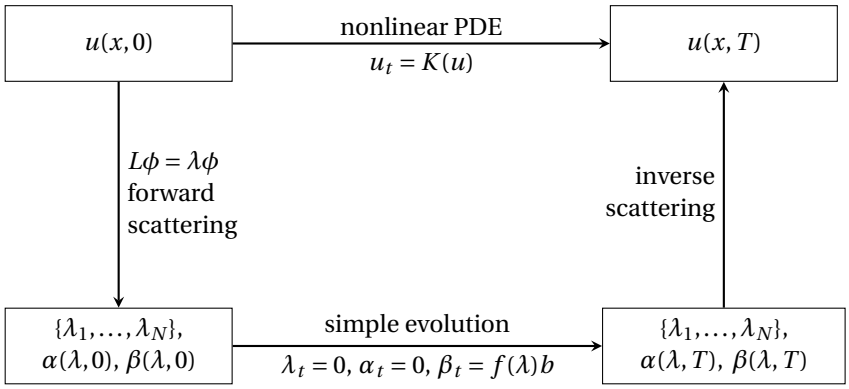


Figure 1.3: Solving a nonlinear PDE using a similar technique as the Fourier transform in Fig. 1.1. First, forward scattering is applied to the initial waveform, then the spectrum and scattering coefficients are evolved linearly, and finally inverse scattering is applied to reconstruct the propagated waveform in the physical domain.

Due to the transformations to and from scattering coefficients, this method would initially be coined the method of inverse scattering, or the forward scattering transform and the inverse scattering transform (IST) [7]. It was also proven that this spectral transformation not only resembled the linear Fourier transform, but even coincided with it for low-amplitude waveforms (i.e., when the KdV equation effectively linearises) [7, Eq. 4.19]. Therefore, the forward scattering transform is often referred to as the Nonlinear Fourier Transform (NFT).

1.1.2. LAX PAIRS

A year after the discovery of the integrability of the KdV equation, Peter Lax formalized the approach of solving the KdV equation as well as higher-order versions of the KdV equation, by introducing the so called *Lax-pair* of linear operators [8]. Given a PDE

$u_t = K(u)$, Lax assumed there existed some u-dependent operator $L(u)$ mapping functions from vector space X to vector space X_1, of which the eigenvalues λ and the rest of its spectrum would remain invariant as u evolves: $L\phi = \lambda\phi$ with $\lambda_t = 0$. As $u(x,t)$ evolves over time, so too will the operator L, as well as the eigenfunctions ϕ, and thus a propagation operator M for the eigenfunction could be defined, such that $\phi_t = M\phi$. Finally, Lax posed that such a pair of operators (L, M) should then be compatible when taking the time derivative:

$$\begin{aligned} &L\phi = \lambda\phi, \qquad \phi_t = M\phi, \\ &\lambda_t = 0 \iff (L\phi)_t = \lambda\phi_t \iff L_t\phi + L\phi_t = \lambda M\phi \\ &\iff \quad L_t\phi + \underbrace{(LM - ML)}_{=[L,M]}\phi = 0, \end{aligned} \tag{1.4}$$

where the compatibility equation in Eq. 1.4 is known as the *Lax equation*. The pair (L, M) is then called a Lax pair for the PDE $u_t = K(u)$.

A simple example of a PDE and a corresponding Lax pair is the advection equation $u_t = u_x$, with Lax pair $L = u$ (multiplication with u operator), $M = \partial_x$ (the spatial differentiation operator):

$$\begin{aligned} &L_t\phi = (L\phi)_t - L\phi_t = (u\phi)_t - u\phi_t = u_t\phi \\ &ML\phi - LM\phi = \partial_x(u\phi) - u\partial_x\phi = u_x\phi \\ &\Rightarrow \quad (L_t\phi = ML\phi - LM\phi) \iff (u_t = u_x). \end{aligned}$$

Lax formulated the following Lax-pair for the KdV equation, for which the Lax equation indeed is satisfied if u propagates according to the KdV equation:

$$L = -\partial_x^2 - u, \qquad M = 4\partial_x^3 + 6u\partial_x + 3u_x, \tag{1.5a}$$

$$\begin{cases} L_t\phi = \frac{\partial L\phi}{\partial t} - L\frac{\partial\phi}{\partial t} = \left(-\phi_{xxt} - (u\phi)_t\right) - \left(-\phi_{txx} - u\phi_t\right) = u_t\phi \\ [M,L]\phi = ML\phi - LM\phi = \cdots = (-u_{xxx} - 6uu_x)\phi \end{cases} \tag{1.5b}$$

$$\Rightarrow L_t = [M,L] \quad \iff \quad u_t = -u_{xxx} - 6uu_x \tag{1.5c}$$

This Lax pair indeed corresponded to the findings of Garner, Greene, Kruskal and Miura [1]. However, Lax furthermore showed that an entire hierarchy of higher-order KdV equations were integrable, which all shared the same spectral L-operator, but differed in their propagation operator M.

1.1.3. AKNS-TYPE LAX PAIRS

Shortly after the discovery of the Lax-pair formalism, many more Lax pairs were found. In 1972, Zakharov and Shabat [9] found a matrix Lax pair for the focusing nonlinear Schrödinger equation, $u_t = iu_{xx} + 2i|u|^2u$, and therewith showed that the Lax pair formalism was not a mere fluke.

Shortly after the discovery of Zakharov and Shabat, a general 2×2 matrix structure for Lax pairs was proposed by Ablowitz, Kaup, Newell and Segur (AKNS) [7], which could capture a large range of ubiquitous PDEs, including the KdV equation, modified KdV

equation (MKdV), the focusing and defocusing nonlinear Schrödinger equation (NLSE), the sine-Gordon equation, and sinh-Gordon equation. The AKNS-type Lax pairs thus covered a large variety of PDEs, while only differing in certain coefficients in their matrix entries. Furthermore, the authors showed that all these PDEs could be solved with the same method, which was a generalization of the earlier discovered methods to solve the KdV equation and the NLSE. The AKNS-type Lax pair is given by

$$L = \begin{bmatrix} i\partial_x & -iq(u) \\ ir(u) & -i\partial_x \end{bmatrix}, \qquad M = \begin{bmatrix} A(u) & B(u) \\ C(u) & -A(u) \end{bmatrix}, \tag{1.6}$$

where $q(u)$ and $r(u)$ are PDE-dependent transformations of u. The coefficients of the M-operator were assumed to be low-order power series in the spectral coefficient λ (for KdV, MKdV and NLSE), or negative power series in λ (for sin(h)-Gordon). Under these assumptions, several combinations of $q(u)$, $r(u)$, and power series satisfied the Lax equation, corresponding to the mentioned PDEs. The NFT then follows from spectral analysis of the L-operator.

After a Lax integrable PDE and its AKNS-type Lax pair were identified, the spectral L-operator could be used to calculate the NFT of the waveform. To solve $L\phi = \lambda\phi$ in the AKNS case, this eigenproblem is typically rewritten to the generalized Zakharov-Shabat eigenvalue problem form:

$$\frac{\partial}{\partial x}\begin{bmatrix}\phi_1(x,\lambda)\\ \phi_2(x,\lambda)\end{bmatrix} = \begin{bmatrix}-i\lambda & q(u)\\ r(u) & i\lambda\end{bmatrix}\begin{bmatrix}\phi_1(x,\lambda)\\ \phi_2(x,\lambda)\end{bmatrix}, \tag{1.7a}$$

$$\text{s.t.} \begin{bmatrix}e^{-i\lambda x}\\ 0\end{bmatrix} \overset{t\to-\infty}{\longleftarrow} \begin{bmatrix}\phi_1(x,\lambda)\\ \phi_2(x,\lambda)\end{bmatrix} \overset{t\to+\infty}{\longrightarrow} \begin{bmatrix}a(\lambda)e^{-i\lambda x}\\ b(\lambda)e^{+i\lambda x}\end{bmatrix}, \tag{1.7b}$$

in which $\phi(x,\lambda)$ is the vector eigenfunction corresponding to the complex spectral parameter $\lambda = \xi + i\eta$, and $a(\lambda)$ and $b(\lambda)$ are the scattering coefficients. Note that the $a(\lambda)$ and $b(\lambda)$ here play similar roles as $\alpha(\lambda)$ and $\beta(\lambda)$ in Eq. 1.3 for the earlier defined scattering coefficients for the KdV equation, and can in fact be transformed to each other through $b(\lambda)$.

1.1.4. MORE LAX-INTEGRABLE SYSTEMS

After the discovery of AKNS-type Lax pairs, other nonlinear PDEs were also found to be Lax-integrable, such as the Boussinesq equation [10], the Harry Dym equation [11], the Hirota-Satsuma equation [12], Kadomtsev-Petviashvili equation [13], the Camassa-Holm equation [14], and many others [15].

In the study of Lax-integrable partial differential equations, also many alternative definitions of Lax-integrability were proposed that would allow more general classes of PDEs to be solved. An alternative formulation was proposed where the spectral operator was replaced with a scattering matrix, leading to $\phi_x = X\phi, \quad \phi_t = T\phi$, supplemented with the zero-curvature condition $X_t - T_x + [X,T] = 0$, which is derived from the condition $\phi_{xt} = \phi_{tx}$. The zero-curvature condition plays the role of compatibility condition similar to the Lax equation in Eq. 1.4 for a classic lax pair (L,M). For the AKNS-type Lax pair in Eq. 1.6, the spectral problem $L\phi = \lambda\phi$ can indeed be reformulated as $\phi_x = X\phi$, which was in fact done in the Zakharov-Shabat eigenvalue problem in Eq. 1.7a. While

the AKNS-type Lax pairs can be formulated similarly well in both form of Lax pairs, the zero-curvature formulation is more general. Furthermore, the formulation $\phi_x = X\phi$ can typically be solved more efficiently than an eigenvalue problem $L\phi = \lambda\phi$ with numerical methods.

Lax pairs do not only exist for scalar partial differential equations (1+1 dimensional, 1 spatial, 1 temporal), but also for multi-dimensional equations, (e.g., the Kadomtsev–Petviashvili equation [16], 2+1 dimensional), for vector partial differential equations (e.g., the coupled nonlinear Schrödinger equation [17], 1+1-dimensional, 2-dimensional vector field), as well as for a system of ordinary differential equation (e.g., the Toda-lattice [18]).

Over the years, many PDEs have been identified as Lax-integrable, although in many different shapes and satisfying other integrability conditions. Simultaneously, many different test methods have been developed to identify integrable PDEs, or to test whether a given PDE is integrable. For example, Painlevé analysis [19] can be used to test if a given PDE is integrable, although it is not guaranteed to be always conclusive, as well as that it relies on an unproven conjecture.

Similarly, the Hamiltonian-based CLL method [20] attempts to explicitly construct a Lax pair for a given PDE, although this method too is not always definitive. The symmetry approach [21] has succeeded in finding all integrable PDEs $u_t = K(u)$ given specific structures of K. For a more complete overview of the techniques to identify integrable PDEs or test if a given PDE is integrable, we refer to [22]. These techniques are mostly useful when a PDE is already known and a corresponding Lax pair is required, but the question remains if a suitable Lax integrable PDE can be found when only measurement data is available.

In particular in engineering, the presence of loss and noise in PDEs prevent exact Lax pairs to exist, and therefore approximations are commonly made. Such approximations have many times allowed NFTs derived from Lax-integrable approximations to be applied in engineering, for example in fiber optics [23] and in water waveform analysis [24].

Within this thesis, we will focus on developing identification techniques for AKNS-type Lax pairs, as this class already contains a wide range of PDEs used in engineering. The identification methods that we develop can thus also be tested using experimental data, as opposed to only simulated data. To make this work as self-contained as possible, the next section goes into more detail on AKNS-type Lax pairs, its associated scattering transform, and its conserved quantities. All of these will be widely used throughout this thesis for the purpose of Lax pair identification.

1.2. AKNS-TYPE LAX PAIRS

In this section we focus on the structure of AKNS-type Lax pairs, and show their associated Lax-integrable PDEs. We then elaborate on how their nonlinear Fourier transforms can be computed, and illustrate their NFT spectra, and explain the connection to solitons. This section is largely based on the 1974 paper by Ablowitz, Kaup, Newell and Segur [7]

To derive PDEs fitting the AKNS-type Lax pair from Eq. 1.6, we rewrite the spectral

operator as a spatial derivative operator. The Lax equation is thus rewritten as an equivalent pair of ordinary differential equations with a spatial derivative operator X and the temporal derivative operator M:

$$L\phi = \begin{bmatrix} i\partial_x & -iq \\ ir & -i\partial_x \end{bmatrix}\phi = \lambda\phi \quad \Rightarrow \quad \begin{bmatrix}\phi_{1x}\\ \phi_{2x}\end{bmatrix} = \underbrace{\begin{bmatrix} -i\lambda & q \\ r & i\lambda \end{bmatrix}}_{X}\begin{bmatrix}\phi_1\\ \phi_2\end{bmatrix} \quad \Rightarrow \boxed{\partial_x\phi = X\phi},$$

$$\phi_t = M\phi \quad \Rightarrow \quad \begin{bmatrix}\phi_{1t}\\ \phi_{2t}\end{bmatrix} = \begin{bmatrix} A & B \\ C & -A \end{bmatrix}\begin{bmatrix}\phi_1\\ \phi_2\end{bmatrix} \quad \Rightarrow \boxed{\partial_t\phi = M\phi}.$$

where $q = q(x,t)$,$r = r(x,t)$, $A = A(x,t,\lambda)$, $B = A(x,t,\lambda)$ and $C = A(x,t,\lambda)$. This is only a Lax pair if the equations are compatible upon cross differentiation, $\partial_{xt}\phi = \partial_{tx}\phi \iff X_t\phi + XM\phi = \partial_t X\phi = \partial_x M\phi = M_x\phi + MX\phi$. Working out this equality thus leads to the following compatibility conditions:

$$M_x - X_t = XM - MX \tag{1.8a}$$

$$\Rightarrow \begin{bmatrix} A_x & B_x \\ C_x & -A_x \end{bmatrix} - \begin{bmatrix} -i\lambda_t & q_t \\ r_t & i\lambda_t \end{bmatrix} = \begin{bmatrix} -i\lambda & q \\ r & i\lambda \end{bmatrix}\begin{bmatrix} A & B \\ C & -A \end{bmatrix} - \begin{bmatrix} A & B \\ C & -A \end{bmatrix}\begin{bmatrix} -i\lambda & q \\ r & i\lambda \end{bmatrix} \tag{1.8b}$$

$$\Rightarrow \begin{bmatrix} A_x & B_x \\ C_x & -A_x \end{bmatrix} - \begin{bmatrix} 0 & q_t \\ r_t & 0 \end{bmatrix} = \begin{bmatrix} -i\lambda A + Cq & -i\lambda B - qA \\ rA + i\lambda C & -i\lambda A + rB \end{bmatrix} - \begin{bmatrix} -i\lambda A + Br & i\lambda B + qA \\ -i\lambda C - rA & qC - i\lambda A \end{bmatrix} \tag{1.8c}$$

$$\Rightarrow \begin{bmatrix} A_x & B_x - q_t \\ C_x + r_t & -A_x \end{bmatrix} = \begin{bmatrix} qC - rB & -2i\lambda B - 2qA \\ 2rA + 2i\lambda C & rB - qC \end{bmatrix}. \tag{1.8d}$$

We thus get the compatibility conditions

$$A_x = qC - rB, \tag{1.9a}$$

$$B_x + 2i\lambda B = q_t - 2qA, \tag{1.9b}$$

$$C_x - 2i\lambda C = r_t + 2rA. \tag{1.9c}$$

These compatibility conditions will prove essential for fitting PDEs to ANKS-type Lax pairs, as well as for deriving properties.

Next, the AKNS authors assumed that A, B and C are low-order polynomials in λ:

$$A = \sum_0^N A^{(n)}(x,t)\lambda^n, \quad B = \sum_0^N B^{(n)}(x,t)\lambda^n, \quad C = \sum_0^N C^{(n)}(x,t)\lambda^n. \tag{1.10}$$

We note here that the order N is directly related to the dispersion order of the AKNS-type PDEs. A low-order of λ thus corresponds to low-order dispersion relations, which are often of greater interest for engineering purposes.

To solve the $A^{(n)}$, $B^{(n)}$, and $C^{(n)}$, we first assume that $A^{(N)} = a_n(t)$, and $B^{(N)} = C^{(N)} = 0$. Next, by considering only the terms of order λ^n, we can find $B^{(n-1)}$ and $C^{(n-1)}$ from $A^{(n)}$, $B^{(n)}$ and $C^{(n)}$. Then $A^{(n-1)}$ may be found from $B^{(n-1)}$ and $C^{(n-1)}$. This recursion thus

becomes:

$$
\begin{aligned}
B^{(n-1)} &= \frac{i}{2}\left(2A^{(n)}q + B_x^{(n)}\right),\\
C^{(n-1)} &= \frac{i}{2}\left(2A^{(n)}r - C_x^{(n)}\right),\\
A^{(n-1)} &= \int qC^{(n-1)} - rB^{(n-1)}\,\mathrm{d}x + a_{n-1}(t),
\end{aligned}
$$

in which $a_{n-1}(t)$ is an integration constant. While the integral over x for $A^{(n-1)}$ may seem problematic, it turns out that it can be explicitly solved in every step. When we arrive at order λ^0, we obtain the compatibility condition for q_t and r_t, namely

$$
\begin{aligned}
q_t &= B_x^{(0)} + 2A^{(0)}q,\\
r_t &= C_x^{(0)} - 2A^{(0)}r.
\end{aligned}
$$

To illustrate this process, we will work out the results for $N = 3$:

$$
\begin{aligned}
&A^{(3)} = a_3(t), \quad B^{(3)} = C^{(3)} = 0\\
&\Rightarrow \begin{cases}
B^{(2)} = \frac{i}{2}\left(2a_3q\right) = ia_3q\\
C^{(2)} = \frac{i}{2}\left(2a_3r\right) = ia_3r\\
A^{(2)} = a_2 + ia_3\int \cancel{qr} - \cancel{rq}\,\mathrm{d}x = a_2
\end{cases}\\
&\Rightarrow \begin{cases}
B^{(1)} = \frac{i}{2}\left(2a_2q + ia_3q_x\right) = ia_2q - \frac{1}{2}a_3q_x,\\
C^{(1)} = \frac{i}{2}\left(2a_2r - ia_3r_x\right) = ia_2r + \frac{1}{2}a_3r_x,\\
A^{(n-1)} = a_1 + \int q(i\cancel{a_2r} + \frac{1}{2}a_3r_x) - r(i\cancel{a_2q} - \frac{1}{2}a_3q_x)\,\mathrm{d}x = a_1 + \frac{1}{2}a_3qr
\end{cases}\\
&\Rightarrow \begin{cases}
B^{(0)} = \frac{i}{2}\left(2(a_1 + \frac{1}{2}a_3qr)q + ia_2q_x - \frac{1}{2}a_3q_{xx}\right) = ia_1q + \frac{i}{4}a_3rq^2 - \frac{1}{2}a_2q_x - \frac{i}{4}a_3q_{xx},\\
C^{(0)} = \frac{i}{2}\left(2(a_1 + \frac{1}{2}a_3qr)r - ia_2r_x - \frac{1}{2}a_3r_{xx}\right) = ia_1r + \frac{i}{4}a_3qr^2 + \frac{1}{2}a_2r_x - \frac{i}{4}a_3r_{xx},\\
A^{(0)} = a_0 + \int q(i\cancel{a_1r} + \cancel{\frac{i}{4}a_3qr^2} + \frac{1}{2}a_2r_x - \frac{i}{4}a_3r_{xx})\\
\qquad\qquad -r(i\cancel{a_1q} + \cancel{\frac{i}{4}a_3rq^2} - \frac{1}{2}a_2q_x - \frac{i}{4}a_3q_{xx})\,\mathrm{d}x = a_0 + \frac{1}{2}a_2qr - \frac{i}{4}a_3(qr_x - rq_x).
\end{cases}
\end{aligned}
$$

We thus obtain the following for A, B and C:

$$A = a_3\lambda^3 + \left(\frac{1}{2}a_3qr + a_1\right)\lambda + \frac{1}{2}a_2qr - \frac{1}{4}a_1(qr_x - q_xr) + a_0, \tag{1.11a}$$

$$B = ia_3q\lambda^2 + \left(ia_2q - \frac{1}{2}a_3q_x\right)\lambda + \left(ia_1q + \frac{i}{2}a_3q^2r - \frac{1}{2}a_2q_x - \frac{i}{4}a_3q_{xx}\right), \tag{1.11b}$$

$$C = ia_3r\lambda^2 + \left(ia_2r + \frac{1}{2}a_3r_x\right)\lambda + \left(ia_1r + \frac{i}{2}a_3r^2q + \frac{1}{2}a_2r_x - \frac{i}{4}a_3r_{xx}\right), \tag{1.11c}$$

which lead to compatibility conditions for q_t and r_t:

$$\begin{aligned} q_t &= ia_1 q_x + \frac{i}{2}a_3(q^2 r)_x - \frac{1}{2}a_2 q_{xx} - \frac{i}{4}a_3 q_{xxx} + 2q\left(a_0 + \frac{1}{2}a_2 qr - \frac{i}{4}a_3(qr_x - rq_x)\right) \\ &= \frac{i}{4}a_3\left(-q_{xxx} + 6rqq_x\right) + \frac{1}{2}a_2\left(-q_{xx} + 2q^2 r\right) + a_1 iq_x + 2a_0 q \qquad (1.12a) \\ r_t &= ia_1 r_x + \frac{i}{2}a_3(r^2 q)_x + \frac{1}{2}a_2 r_{xx} - \frac{i}{4}a_3 r_{xxx} - 2r\left(a_0 + \frac{1}{2}a_2 qr - \frac{i}{4}a_3(qr_x - rq_x)\right) \\ &= \frac{i}{4}a_3\left(-r_{xxx} + 6qrr_x\right) + \frac{1}{2}a_2\left(r_{xx} - 2r^2 q\right) + a_1 ir_x - 2a_0 r. \qquad (1.12b) \end{aligned}$$

To make the final connection between this AKNS-type Lax pair and a PDE $u_t = K(u)$, the relations $q(u)$ and $r(u)$ must be established in such away that the equations for q_t and r_t are compatible. The AKNS authors found the following special cases:

1. $a_0 = a_1 = a_2 = 0$, $a_3 = -4i$

$$r = -1 \quad \Rightarrow \quad q_t = -q_{xxx} - 6qq_x \quad \text{(KdV)} \qquad (1.13)$$

$$r = \mp q \quad \Rightarrow \quad q_t = -q_{xxx} \pm 6qq_x \quad \text{(focussing}(-q)\text{/defocussing}(+q)\text{ MKdV)} \qquad (1.14)$$

2. $a_0 = a_1 = a_3 = 0$, $a_2 = -2i$

$$r = \mp q^* \quad \Rightarrow \quad q_t = iq_{xx} \pm 2iq^2 q^* \quad \text{(focussing}(-q)\text{/defocussing}(+q)\text{ NLSE)} \qquad (1.15)$$

In a similar fashion, the AKNS authors also considered an expansion of A, B and C in negative powers. Taking the lowest order expansion,

$$A(x,t,\lambda) = \frac{a(x,t)}{\lambda}, \quad B(x,t,\lambda) = \frac{b(x,t)}{\lambda}, \quad C(x,t,\lambda) = \frac{c(x,t)}{\lambda}, \qquad (1.16)$$

this resulted in the following equations:

$$\begin{aligned} &a_x = \frac{i}{2}(qr)_t, \quad b = -\frac{i}{2}q_t, \quad c = \frac{i}{2}r_t, \\ &q_{xt} = -4iaq, \quad r_{xt} = -4iar. \end{aligned}$$

Two important cases are

1. $r = -q = \frac{1}{2}u_x$, $\quad a = \frac{i}{4}\cos u$, $\quad b = c = \frac{i}{4}\sin u$,

$$\Rightarrow u_{xt} = \sin u \quad \text{(sine-Gordon equation)}, \qquad (1.17)$$

2. $r = q = \frac{1}{2}u_x$, $\quad a = \frac{i}{4}\cosh u$, $\quad b = c = \frac{i}{4}\sinh u$

$$\Rightarrow u_{xt} = \sinh u \quad \text{(sinh-Gordon equation)}. \qquad (1.18)$$

Many important PDEs can thus be captured by AKNS-type Lax pairs, and this method includes many more PDEs (in fact infinitely many). The shown ones are the most common examples, and we will therefore focus in particular on the shown equations.

When considering Eq. 1.13 through Eq. 1.18, we first note that transformations (such as scalings) of the presented PDEs are also Lax-integrable, simply by changing the relation $q(u)$ and $r(u)$. We will often apply this throughout this thesis, and this will be one of our main instruments of identifying scaling parameters in a PDE.

Next, we note that the relation for $q(u)$ and $r(u)$ do not allow any arbitrary functions $q(u)$ and $r(u)$. For the two shown power-series assumptions for A, B and C, we list the following as special cases:

1. $r = -1$ (KdV),
2. $r = -q$ (focussing MKdV, sine-Gordon),
3. $r = +q$ (defocussing MKdV, sinh-Gordon),
4. $r = -q^*$ (fNLSE),
5. $r = +q^*$ (dNLSE).

In particular linear combination of relations, such as $r = q - 1$, do not generally lead to feasible compatibility conditions, and thus the shown AKNS-type PDEs are not connected in a continuous way.

Interestingly, the focussing MKdV equation and sine-Gordon equation share the same structure $r = +q$, and so do the defocussing MKdV equation and the sinh-Gordon equation with $r = -q$. This implies that within each relation $r(q)$, the resulting spectral operators L are the same as well up to the transformation of the waveform $q(u)$. In particular, an initial waveform u would yield the same (de)focussing MKdV spectrum as the waveform $(-)u_x$ for the sin(h)-Gordon equation, and these spectra remain constant during propagation. The only difference between them thus lies in the propagation relation due to the M-operator.

A more common example of two Lax-integrable PDEs that share the same spectral L-operator is the hierarchy of (higher-order) KdV equations. While the shown KdV equation (of third order: u_{xxx}) is the most common one, there also exist the (trivial) "first-order KdV equation" (i.e., $u_t = u_x$), the fifth-order KdV equation ($u_t = u_{xxxxx} + \ldots$), the seventh-order KdV equation, etc., that all share the same spectral operator [8], and are all of AKNS-type. The AKNS-type Lax pairs for higher-order KdV equations follow from taking a polynomial assumption of the same order in Eq. 1.10.

1.2.1. NONLINEAR FOURIER TRANSFORM FOR AKNS-TYPE LAX PAIRS

After finding an AKNS-type Lax pair consisting of a spectral operator L (or equivalently X) and propagation operator M, the waveform u may be transformed into the spectral domain using the spectral operator. In the spectral domain, the propagation of the transformed waveform becomes linear, and is fully determined by the M operator.

1

Before stating the NFT, we first note here that the boundary conditions of a waveform are of great influence on its NFT. The most common two boundary condition are vanishing boundary condition on $q(x) \xrightarrow{|x|\to\infty} 0$ (sufficiently fast) [7], and periodic boundary condition $q(x) = q(x+L)$ (see e.g., [25]). While the NFT may be defined and computed for both of these types (as well as for other types such as non-vanishing boundary conditions, see e.g., [26]), in many practical problems in engineering, it suffices to consider vanishing boundary conditions. Vanishing boundary conditions have been studied thoroughly in the literature, and are in several ways easier to explain and interpret than periodic or non-vanishing boundary conditions. Furthermore, many practical waveforms can be well approximated by means of truncation, so that vanishing boundary conditions are still appropriate. We thus will only focus on waveforms u, that lead to a $q(u)$ with vanishing boundary conditions in the spatial direction. We also assume that $q(u)$ is bounded. For this thesis the mentioned conditions are sufficient, but the exact mathematical conditions for which the NFT exists require more nuance. For more formal conditions, we refer to [17].

THE NONLINEAR FOURIER TRANSFORM

Next, we define the nonlinear Fourier transform, by means of the AKNS forward scattering problem. Let $q(x)$ and $r(x)$ be the appropriate transformations of the waveform $u(x)$ at $t = t_0$ (or $t_0 = 0$ without loss of generality). We then first solve the scattering functions $\phi(x,\lambda)$, with $\lambda \in \mathbb{C}$ in the upper half plane ($\Im\lambda \geq 0$), from the spectral problem $L\phi = \lambda\phi$, or equivalently $\phi_x = X\phi$:

$$\frac{\mathrm{d}}{\mathrm{d}x}\begin{bmatrix}\phi_1(x,\lambda)\\ \phi_2(x,\lambda)\end{bmatrix} = \begin{bmatrix}-i\lambda & q(x)\\ r(x) & i\lambda\end{bmatrix}\begin{bmatrix}\phi_1(x,\lambda)\\ \phi_2(x,\lambda)\end{bmatrix}, \tag{1.19a}$$

$$\text{s.t.}\begin{bmatrix}e^{-i\lambda x}\\ 0\end{bmatrix} \xleftarrow{x\to-\infty} \begin{bmatrix}\phi_1(x,\lambda)\\ \phi_2(x,\lambda)\end{bmatrix} \xrightarrow{x\to+\infty} \begin{bmatrix}a(\lambda)e^{-i\lambda x}\\ b(\lambda)e^{+i\lambda x}\end{bmatrix}. \tag{1.19b}$$

Note that Eq. 1.19b denotes the boundary condition, which are required for the uniqueness of the scattering functions. Intuitively, this scattering equation may be interpreted as a complex exponential $e^{-i\lambda x}$ coming in from the left, that keeps on undisturbed so long as $q(x) \approx 0$. Indeed, if both $q(x) = 0$ and $r(x) = 0$, then we find the solutions $\phi_1(x,\lambda) = e^{-i\lambda x}$ and $\phi_2(x,\lambda) = e^{+i\lambda x}$. As $\phi(x,\lambda)$ evolves over x, it encounters the potentials q and r that "scatter" ϕ over two components, which stabilize again as $ae^{-i\lambda x}$ and $be^{+i\lambda x}$ after the interaction with $q(x)$ and $r(x)$ is over. This process is also illustrated in Fig. 1.4. When $\Im\lambda > 0$, the two components of the scattering field would also have an increasing/decreasing component.

After $\phi(x,\lambda)$ has been determined, $a(\lambda)$ and $b(\lambda)$ can be extracted from the limit of ϕ on the right, $x \to +\infty$. The coefficients a and b are called the scattering coefficients, and all information included in $\phi(x,\lambda)$ is fully represented by $a(\lambda)$ and $b(\lambda)$. It can be shown that $a(\lambda)$ and $b(\lambda)$ can be represented just by their values on the real axis, and by their values in the upper half plane where $a(\lambda) = 0$ [17]. The real line corresponds to the continuous spectrum of the L-operator, and the simple zeros of $a(\lambda)$ of the L-operator (in this thesis we ignore the cases with multiple roots) are the discrete spectrum (eigenvalues).

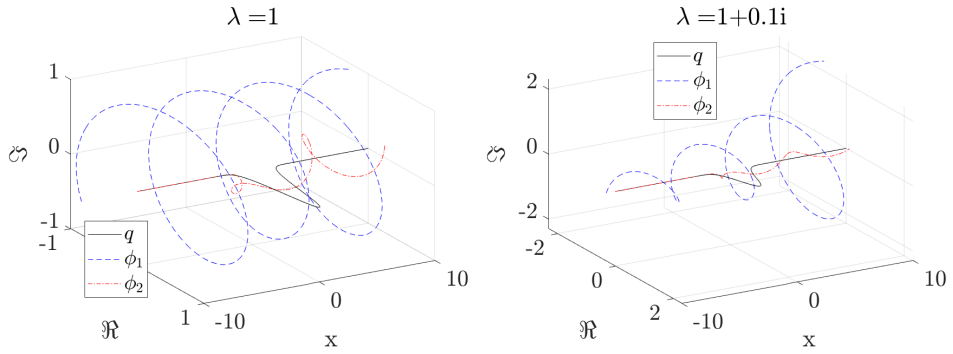


Figure 1.4: The scattering function $\phi(x,\lambda)$ of the fNLSE ($-r^* = q = u$) for the waveform $u(x) = e^{-x^2}$ at $\lambda = 1$ (left) and at $\lambda = 1+0.1i$ (right).

The NFT spectrum can be defined through these two sets by adding certain values of the scattering coefficients to the spectra, resulting in a complete NFT spectrum consisting of a continuous NFT spectrum and a discrete NFT spectrum. More specifically, the continuous NFT spectrum can be defined as the value of $b(\lambda)$ (or alternatively b/a) on the real line, $\lambda \in \mathbb{R}$, and the discrete NFT spectrum as $\{\lambda_k, b_k\}_{k=1}^{K}$ (or alternatively by $b/a_\lambda = \frac{b}{\partial a/\partial\lambda}$), in which the eigenvalues λ_k are the zeros of $a(\lambda)$ in the complex upper half plane, and $b_k = b(\lambda_k)$.

We note here that many numerical methods have been developed to calculate the NFT [27, 28]. Much work has been to speed up the NFT, in particular for the NLSE (see e.g., [29–31]) and the KdV equation (see e.g., [32–34]). In particular, the software library *FNFT* (Fast Nonlinear Fourier Transform) [35] is widely used for the computation of the KdV-NFT and the NLSE-NFT in this thesis.

INTERPRETATION OF THE NFT SPECTRUM

An example of the NFT of KdV type ($r = -1; q = u$) is shown in Fig. 1.5. The spectrum consists of the continuous line, and several points in the discrete spectrum in the upper half plane. For the KdV equation, all eigenvalues lie on the imaginary axis, which is also observed here. For the continuous spectrum, we also showed $b/a = b(\lambda)/a(\lambda)$ (the variations are better visible than just $b(\lambda)$ for the KdV), and for the discrete spectrum we also show $\ln(|b|)$. For the KdV, each eigenvalue in the discrete spectrum corresponds to a soliton, a localized waveform that retains it shape, even after interaction with other solitons or wave components. The eigenvalue is directly related to the energy of the soliton. The continuous spectrum of the KdV corresponds to dispersive wave components, and may be roughly compared to the linear Fourier spectrum. In the low-energy limit, the continuous spectrum of the NFT even coincides with the linear Fourier spectrum under appropriate scaling [7, p. 270]

While the NFT for other AKNS-type PDEs may appear different, they all share the properties that they consist of a continuous spectrum covering the real axis, and a discrete spectrum of eigenvalues that correspond to solitons. For example, the discrete spec-

trum of the fNLSE can contain eigenvalues anywhere on the upper half-plane (not just on the imaginary axis). The dNLSE on the other hand always has an empty discrete spectrum for waveforms with vanishing boundary conditions, and indeed this equation cannot support solitons in this case (eigenvalues and so-called "dark" solitons are however possible for the dNLSE for non-vanishing boundary conditions) [36]. The NFT for the sine-Gordon equation can contain eigenvalues, which also correspond to solitons. For all these equation, the continuous spectrum is also associated with dispersive wave components.

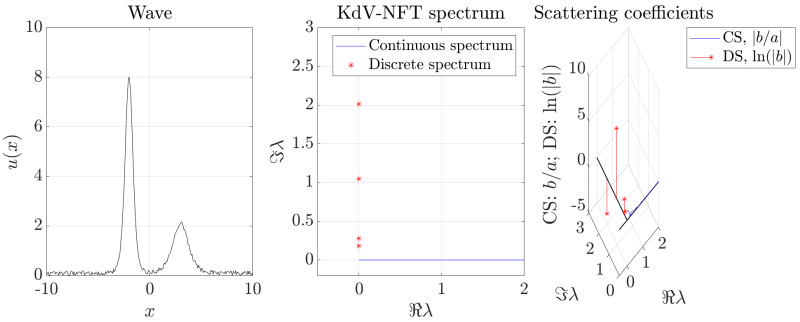


Figure 1.5: An exemplary waveform and its KdV-NFT. **Left:** the waveform was created by taking a two-soliton solution of the KdV equation (one soliton of height 8 with $\lambda = 2i$, and one of height 2 with $\lambda = 1$), and adding some noise. Outside the shown range, the waveform amplitude is 0. **Middle:** the NFT spectrum consists of the real positive line, and eigenvalues (red stars) in the upper-half plane, where the highest two corresponds to the two solitons, and the lower two with low-energy solitons that result from the addition of the noise. **Right**: the NFT spectrum, now complemented with information from the scattering coefficients. Note that different quantities are shown for the discrete spectrum (DS) and for the continuous spectrum (CS).

PROPAGATION IN THE NFT DOMAIN

After determining the spectrum of a signal, we now consider the propagation. First we note that the discrete spectrum remains constant due to the assumption that $\lambda_t = 0$. However, the scattering coefficients $a(\lambda)$ and $b(\lambda)$ will evolve over time. Recall that $a(\lambda)$ and $b(\lambda)$ resulted from the behavior in the right far-field of ϕ, which propagates according to $\phi_t = M\phi$. In the right far-field, we assumed that $q(x) \to 0$, and therefore the M-operator greatly simplifies. In particular, all solutions in Eq. 1.13-Eq. 1.18 satisfy $B(x,t,\lambda) \xrightarrow{x\to\infty} 0$, $C(x,t,\lambda) \xrightarrow{x\to\infty} 0$, and $A(x,t,\lambda) \xrightarrow{x\to\infty} A_0(\lambda)$ (we note here that KdV forms an exception due to $r = -1$, but this can be overcome by a simple transformation [37, p. 122917]). The propagation operator M thus becomes diagonal, thus leading to linear and independent evolution equations for $a(\lambda)$ and $b(\lambda)$. The NFT thus converts a the nonlinear propagation in the physical domain to be transformed to a linear transformation in the spectral domain.

We evaluate the propagation relations for the $a(\lambda)$ and $b(\lambda)$ for a few of the earlier considered PDEs, by evaluation $A(\lambda)$ with the appropriate assumptions for $r(u)$, $q(u)$, and the coefficients in the power expansion of A from Eq. 1.11 and Eq. 1.16. We obtain:

1. KdV, MKdV ($a_3 = -4i, a_0 = a_1 = a_2 = 0$) $\Rightarrow A_0(\lambda) = -4i\lambda^3$
2. f/dNLSE, ($a_2 = -2i, a_0 = a_1 = a_3 = 0$) $\Rightarrow A_0(\lambda) = -2i\lambda^2$
3. sine/sinh-Gordon, $\left(a \xrightarrow{u \to 0 \,(\mathrm{mod}\, 2\pi \text{ for sine-Gordon})} \frac{i}{4}\right)$ $\Rightarrow A_0(\lambda) = \frac{i}{4\lambda}$.

We thus get that $\phi_{1t} = A_0(\lambda)\phi_{1t}$ and $\phi_{2t} = -A_0(\lambda)\phi_{2t}$, at both the left and right end of ϕ, resulting in $\phi_1(t) = \phi_1(0)e^{A_0(\lambda)t}$ and $\phi_2(t) = \phi_2(0)e^{-A_0(\lambda)t}$. However, due to the imposed boundary conditions in Eq. 1.19b, the left hand side of ϕ_1 must remain constant over time, which can be done by a simple multiplication of ϕ by $e^{-A_0(\lambda)t}$. This re-scaling also affects the right hand side, which thus results in the following propagation equations for the scattering coefficients:

$$a(\lambda) = 0, \quad b(\lambda) = e^{-2A_0(\lambda)t}. \tag{1.20}$$

In this thesis, we are interested in the identification and parameterization of Lax-integrable PDEs. One of the methods we will apply widely is comparing the NFT spectra of a waveform at different states of its evolution. If we found a suitable Lax-pair (or at least suitable spectral operator L), then their discrete spectra should be equivalent. We can then also consider the change in scattering coefficients to determine the M-operator.

INVERSE NFT

After determining the NFT of a waveform, and propagating its scattering coefficient in the NFT domain, the propagated waveform in the physical domain can be reconstructed using the inverse nonlinear Fourier transform, or inverse scattering transform. This was demonstrated early on [7, 38], and numerical methods are now widely available to determine the inverse NFT (see e.g., [35] for KdV and NLSE). The focus of this thesis however lies not on the inverse transformation, but rather on the comparison of various waveforms in the NFT domain, or the comparison of the conserved quantities of the waveforms. The inverse Fourier transform therefore lies outside the scope of this thesis.

1.2.2. CONSERVED QUANTITIES OF AKNS-TYPE PDES

In this section, we focus on conserved quantities of AKNS-type PDEs. So far, we have explained that the NFT-spectrum of waveforms governed by an AKNS-type PDEs remain constant, which may be used for identification of a suitable Lax pair. However, Lax-integrable PDEs are also associated with an infinite number of conserved quantities, which may also be used for identification. For example, to check if a PDE is of KdV type, we may consider the conserved quantities of the KdV, and check if these indeed remain constant during evolution.

Conserved quantities are in general of the following form:

$$C = \int_D f(u)\,\mathrm{d}x, \quad \text{with } \frac{\partial C}{\partial t} = 0. \tag{1.21}$$

Already before the discovery of the method of inverse scattering, it was known that several equations possessed an infinite number of conserved quantities. For example, the

KdV $u_t = -u_{xxx} - 6uu_x$ was known to possess an infinite number of conserved quantities, of which the first three are given by [39]

$$C_1 = \int_D u \,\mathrm{d}x, \qquad C_2 = \int_D u^2 \,\mathrm{d}x, \qquad C_3 = \int_D -2u^3 + q_x^2 \,\mathrm{d}x, \tag{1.22}$$

where D is the full domain for x. While we assume D to be $(-\infty,\infty)$, similar results also hold when e.g., periodic boundary conditions are used on a finite domain.

Indeed, when evaluating the time derivative of the first two, we find that both conserved quantities are time invariant. Note that we use integration by parts, and apply the vanishing boundary conditions to cancel out $\int_{-\infty}^{\infty}(\cdot)_x \,\mathrm{d}x$:

$$C_{1t} = \int_{-\infty}^{\infty} u_t \,\mathrm{d}x = \int_{-\infty}^{\infty} \left(-u_{xx} - 3u^2\right)_x \,\mathrm{d}x = 0,$$
$$C_{2t} = \int_{-\infty}^{\infty} 2uu_t \,\mathrm{d}x = \int_{-\infty}^{\infty} -2uu_{xxx} - 4(u^3)_x \,\mathrm{d}x = [-2uu_{xx}]_{-\infty}^{\infty} + \int_{-\infty}^{\infty} (u_x^2)_x \,\mathrm{d}x = 0.$$

For an L-operator of AKNS-type, the existence of an infinite number of conserved quantities may also directly be shown. To do so, we again consider Eqs. 1.9c. Starting from Eq. 1.9a and substituting B by (1.9b) and C by (1.9c), and applying partial integration on the terms $qC_x + rB_x$, we get

$$\begin{aligned} & A_x = qC - rB, \quad B = \frac{1}{i\lambda}(q_t - B_x - 2qA), \quad C = \frac{1}{i\lambda}(-r_t + C_x - 2rA) \\ \Rightarrow\ & A_x = \frac{1}{i\lambda}\left[q(-r_t + C_x - 2rA) - r(q_t - B_x - 2qA)\right] \\ & = \frac{1}{i\lambda}\left[\underbrace{-qr_t - rq_t}_{=-(qr)_t} \underbrace{-2Arq + 2Aqr + (qC)_x + (rB)_x}_{\text{drops out (when integrating)}} - \underbrace{q_xC + r_xB}_{\mathcal{O}(1/\lambda^2)}\right]. \end{aligned} \tag{1.23}$$

Note that in the final expression, B and C arise again, so these can again be substituted, which adds terms of one order higher in $1/i\lambda$. This process can be repeated indefinitely [17], and iteration n will yield another term in the form $\left(c_n(q,r)\right)_t$, associated with $1/(i\lambda)^n$.

Upon integrating over x over the full domain, we find that the left hand side in Eq. 1.23 drops out as $A(\infty) = A(-\infty)$, and thus the right hand side of Eq. 1.23 must also be equal to 0. The right hand side can thus be expanded as a polynomial in $1/i\lambda$:

$$0 = \frac{\partial}{\partial t}\int_{-\infty}^{\infty} \sum_{n=1}^{\infty} c_n(q,r)\frac{1}{(i\lambda)^n} \,\mathrm{d}x. \tag{1.24}$$

This polynomial in $1/i\lambda$ can only equal 0 if and only if every term $\frac{\partial}{\partial t}\int_{-\infty}^{\infty} c_n \,\mathrm{d}x = \frac{\partial}{\partial t} C_n$ is zero, and thus this gives rise to an infinite number of conserved quantities C_n of AKNS type.

I would like to note here that the iterative substitution of B and C and the subsequent partial integrations may be performed manually, but the process is tedious. For my research, I worked out the first five conserved quantities by hand, using repeated integration by parts, which already proved a challenge upon its own. I wrote a symbolic mathematics script that could automatically generate the conserved quantities, which can be made available to the reader upon reasonable request.

The first five conserved quantities of AKNS-type are given by

$$C_1 = \int_D qr \,\mathrm{d}x, \qquad C_2 = \int_D rq_x - r_x q \,\mathrm{d}x, \tag{1.25a}$$

$$C_3 = \int_D q^2 r^2 + q_x r_x \,\mathrm{d}x, \qquad C_4 = \int_D -r_{xx} q_x + q_{xx} r_x + \tfrac{3}{2}\left(r^2 (q^2)_x - q^2 (r^2)_x\right) \mathrm{d}x, \tag{1.25b}$$

$$C_5 = \int_D 2q^3 r^3 + q_{xx} r_{xx} + \left(q_x^2 r^2 + r_x^2 q^2\right) + 8 q q_x r r_x \,\mathrm{d}x. \tag{1.25c}$$

From these first five terms, we can already note certain patterns. Higher conserved quantities become increasingly complex, involving higher orders of q, r and higher spatial derivatives. When considering noisy data, the higher-order terms are typically increasingly sensitive to noise, so for the purpose in this paper, we will only consider a few lower-order terms. We also note here that not all conserved quantities are non-trivial for certain choices of $r(q)$. For example, for $r = -1$, we observe that $C_2 = \int_{-\infty}^{\infty} -q_x \,\mathrm{d}x = 0$. However, the odd quantities are nontrivial for the discussed choices $r(q) \in \{-1, \pm q, \pm q^*\}$.

We thus observe that only a small number of the conserved quantities will be useful when analyzing experimental data, but we will show in this thesis that this can already lead to fast and reliable parameter identification, and distinguishing of integrable PDEs.

1.3. RESEARCH GOAL

Several fields in engineering have been able to harness the powerful analysis that comes along with the existence of a Lax-integrable description of a system. The question remains what other practical systems can be described by a Lax-integrable PDE. In particular the question arises whether a Lax-integrable PDE can be found given a set of (noisy) measurement data.

This problem could in theory be addressed by first attempting to identify a PDE using conventional data-driven methods (e.g., [40, 41]), and then finding a corresponding Lax-pair. However, since the data set is often noisy, the identified PDE will rarely be exactly Lax-integrable, which leaves the question which – if any – Lax integrable PDE is 'closest' to the identified (non-integrable) PDE. Even when the data set is noiseless and a perfectly fitting PDE is found, it is still likely that this PDE is not exactly integrable, while it could be approximated by a Lax-integrable variant and NFT analysis could be applied.

An alternative method would be to attempt to directly find a Lax pair from measurement data. However, to the best of our knowledge only a single data-driven method has been proposed to directly identify Lax pairs [42]. However, also this methods requires derivatives with respect to the evolution variable and is therefore also very sensitive to noise, as well as that the measurement data may also be sparse in the direction of evolution, and that derivatives may therefore not be determined. There thus remains a large challenge in determining whether given measurement data can be well described by a Lax-integrable PDE, and if so, what its Lax pair is.

In this thesis, we propose several data-driven approaches to directly identify a suitable ANKS-type Lax pair and a corresponding Lax-integrable PDE, and demonstrate the feasibility of various approaches. The approaches proposed in this thesis are application

driven. The main focus in this thesis is on AKNS-type Lax-pairs, as this class of problems already captures a large class of ubiquitous PDEs. The restriction to AKNS-type PDEs allows us to exploit their overarching structure in the identification. As the set of Lax-integrable PDEs is very diverse, we envision that the task of finding a suitable integrable PDE to a dataset is split up into multiple classes, where for each class a specialized identification method is performed. Due to the structure of the AKNS-type PDEs, we chose to first dedicate our attention to this class.

AKNS-type Lax pairs are often recognized in engineering problems for which the method of inverse scattering is of practical use, which is not necessarily true for all Lax integrable systems. AKNS-type Lax pairs also cover a wide range of Lax pairs, using a single structure, and can therefore be parameterized up to a certain extend. Finally, the PDEs associated with AKNS-type Lax pairs contain many PDEs without very high-order derivatives (in particular up to third derivatives). PDEs with very high-order derivatives can become very sensitive to noise, and these are thus very hard to identify from measurement data, as well as that the underlying model can become too complex to describe an experimental system accurately. For these engineering reasons, we are thus mainly interested in (relatively) low-order PDEs, many of which are already provided by AKNS-type PDEs.

1.3.1. MAIN RESEARCH QUESTION

The goal of this thesis is to develop novel data-driven methods that identify a Lax-integrable system that explain given measured data as well as possible, even when the true system is not Lax-integrable. As the space of all Lax-pairs is still vast, we only focus on AKNS-type Lax pairs in this thesis. The main research question may thus be formulated as follows:

What AKNS-type Lax pair fits best to given data?

Within this thesis, we consider two main approaches for Lax pair identification. Our first approach assumes a known Lax integrable PDE and corresponding Lax pair, except for one or two parameters. The difficulty lies in that waveform data is only sparsely available throughout the evolution, and thus temporal derivatives cannot be used. We compare the NFT-spectrum at multiple stages of evolution, for different parameter values. For the correct parameter values, the NFT spectra are identical up to the linear evolution of the scattering coefficients. We demonstrate this approach for both the KdV equation, and the NLSE equation, using simulated data as well as experimental data.

Our second approach relies on comparison of conserved quantities. We make the assumption that the underlying system is governed by an AKNS-type PDE, and parameterize the corresponding Lax pairs. This parametrization generalizes the parametrizations from the first method, and relies on the fact that many AKNS-type PDEs share the same structure of conserved quantities. For the correct parameter values, the conserved quantities indeed remain constant during propagation. This method is demonstrated for a various types of simulated data with noise.

1.4. OUTLINE OF THE DISSERTATION

In Chapter 2, we focus on identifying coefficients of the Korteweg-de Vries equation and its Lax pair, that best explain experimentally measured shallow-water waveforms. We propose and validate two novel identification methods. First, we compare eigenvalues corresponding to solitonic components, using the nonlinear Fourier transform. Second, we compare the conserved quantities of the KdV at different snapshots of the developing waveforms, and identify the system parameters such that the conserved quantities are indeed conserved.

In Chapter 3, we demonstrate similar methods as in Chapter 2, but now for simulated waveforms in an optical single-mode fiber, modeled by the focussing nonlinear Schrödinger equation. We demonstrate that both the discrete spectrum (solitonic components) and the continuous spectrum (dispersive components) may be used to identify a suitable Lax pair, as well as extract relevant coefficients of the PDE, such as the dispersion and nonlinearity coefficient of the single-mode fibers, as well as the fiber length.

In Chapter 4, we apply the method from Chapter 3 on experimental data. It is shown that in particular the discrete spectrum is useful for the identification of the parameters of the NLSE. Furthermore, we also compare the nonlinear spectrum-based method to a conventional propagation-based method in terms of accuracy and speed, and demonstrate that the speed of the NFT-based identification methods is mostly independent of the fiber length.

In Chapter 5, we propose a method to efficiently calculate the soliton content of an NLSE-governed waveform, by means of windowing the considered waveform in both the frequency and time domain. In Chapter 4 we observed that Lax pair identification by comparing solitonic components is a powerful tool, but that the calculation of the solitonic components for very rich waveform becomes computationally challenging. The proposed method overcomes this limitation for the NLSE-based NFTs.

Finally in Chapter 6, we develop a general conserved quantities-based method to quickly identify AKNS-type Lax pairs and identify their parameters. We already developed PDE-specific conserved quantity-based identification methods for KdV in Chapter 2 and for NLSE in Chapter 3, but in this chapter we do so for a wide class of AKNS-type systems.

2

WATER-DEPTH IDENTIFICATION FROM FREE-SURFACE DATA USING THE KDV-BASED NONLINEAR FOURIER TRANSFORM

We propose a novel method to determine the average water depth from shallow, weakly nonlinear water waves that are approximated by the Korteweg-de Vries equation. Our identification method only requires free-surface measurements from two wave gauges aligned in the direction of wave propagation. The method we propose is based on comparing solitonic components in wave packets, which are computed using the nonlinear Fourier transform (NFT) (typical time-series data often contains at least some solitonic components, even when these components are not directly visible). When the correct water depth is used for the normalization of the wave, the solitonic components found by the NFT remain constant as the wave packet propagates, whereas any other water depth will result in solitonic components that do not remain constant. The basic idea is thus to iteratively determine the water depth that leads to a best fit between the solitonic components of time series measurements at two different gauge positions. We present a proof-of-concept on experimental bore data generated in a wave flume, where the identified water depth is within 5% of the measured value.

2.1. INTRODUCTION

The Korteweg-de Vries (KdV) equation closely models the propagation of progressive free-surface waves in shallow water with depth-to-wavelength ratio $h/L < 0.22$ [44, 45], and finds many applications in coastal engineering [46]. The water depth is the essential governing parameter in the KdV equation. However, the water depth may not always be known, as it may be hard to measure, or is slowly changing over time. This is for example the case in wave flumes with moving-bed experiments, where the average water depth

may change due to sediment transportation [47, p.108-111], or water losses due to overtopping [48, p.4]. Another example occurs for arrays of buoys in front of the coast, where the average water depth can slowly vary due to drifting of the buoys over an uneven bottom, sediment transportation, or the changing tidal elevation [49, 50].

Existing methods for water depth measurement at the coast, in waterways, and in flumes usually rely on direct measurement of water depths, using sound or light [51]. Alternatively, statistical properties such as wavelength and period may be derived from aerial photographs, which may be related to water depth as well [52]. However, the mentioned examples use very specialized devices to determine the depth, whereas these are not always present or accessible. In contrast, buoys at the coast and in waterways measuring the free surface amplitude are often already installed for more general purposes, and wave flumes are often already equipped with wave gauges, providing plentiful free surface measurements at certain fixed locations in both cases. We investigate the possibility to identify the average water depth from this type of data. It has already been shown that the bathymetry can be roughly estimated from space-time series free-surface data, given that the average water depth is approximately known [53]. However, this method used space-time data (two-dimensional data), as opposed to just time series at two locations, and would therefore not be applicable on buoy or gauge data. A final approach for identifying the water depth is through the constants of motion of the KdV equation, analogous to a method discussed in [54]. The water depth can directly be extracted as the ratio of certain global quantities of the wave at two different positions, as shown in Appendix D. However, this method strongly depends on the pre-processing method, and is sensitive to noise and distortion.

In this study, we propose a simple method to use the free-surface elevation data at just two locations aligned in the direction of wave propagation to determine the water depth, under the assumption that the wave propagates in constant depth and is approximately governed by the Korteweg-de Vries equation. Neither the exact distance between the gauges (e.g., in the case of buoy data) nor precisely time-synchronized measurements are required in our proposed method, as long as the measurements are of the same progressive free-surface waves. The method we propose is based on the nonlinear Fourier transform (NFT) for the KdV equation (KdV-NFT) [24]. In this study, we will demonstrate a proof-of-concept by focussing on wave packets, so that the NFT with vanishing boundary conditions can be applied.

The NFT decomposes a wave packet into a discrete spectrum and a continuous spectrum, representing two types of waves. The discrete spectrum represents translatory stable waves (solitons), while the continuous spectrum represents dispersive oscillatory wave components (radiation). If a free-surface wave in constant depth evolves perfectly according to the KdV equation, both the solitonic components and the amplitudes of the continuous spectrum remain constant during propagation. As the water depth governs the propagation of a wave packet, the water depth is also the governing parameter in the NFT and its resulting spectra. Therefore, the basic idea is to iteratively determine the water depth that leads to a best fit between the nonlinear Fourier spectra of time series measurements at two different gauge positions. For our measurements, most of the energy was contained in the solitonic components, and we will therefore focus on matching the discrete spectra in this study. The value at which the NFT spectra fit best is

then identified as the optimal water depth, under the assumption that the water depth between the gauges is constant.

Although NFT-based identification has not been applied for water depth identification as of yet to the best of our knowledge, we already showed in earlier work that NFT-based parameter identification is possible for optical fiber systems [54, 55]. Light propagation through optical fibers is governed by the nonlinear Schrödinger equation (NLSE), which also allows for an (NLSE-based) nonlinear Fourier transform. Similar to the KdV-NFT, the NLSE-NFT decomposes a signal into solitonic components and dispersive components, allowing us to use similar approaches from [54] in this research.

The structure of this chapter is as follows. First, we present the KdV model and its nonlinear Fourier transform. Second, we propose our novel water-depth identification algorithm, based on comparing solitonic components. Third, we validate the identification method on simulated and experimental data. Finally, we conclude the chapter.

2.2. THE KDV EQUATION AND THE NONLINEAR FOURIER TRANSFORM

The development of long unidirectional progressive free-surface waves in shallow water is modeled by the time-like Korteweg-de Vries equation [2],

$$\eta_l + c_0'\eta_\tau + \alpha'\eta\eta_\tau + \beta'\eta_{\tau\tau\tau} = 0, \tag{2.1}$$

where $\eta(t,s)$ [m] denotes the free-surface elevation compared to the still-water level, l [m] the position, τ [s] the time, c_0' [s/m] the inverse wave celerity, α' [s/m^2] the nonlinearity coefficient and β' [s^3/m] the dispersion coefficient. Subscripts denote partial derivatives. For progressive free-surface waves, the governing coefficients depend only on the water depth [56]:

$$\begin{aligned} c_0 &= \sqrt{gh}, & \alpha' &= -\frac{3}{2hc_0} = -\frac{3}{2h\sqrt{gh}}, \\ c_0' &= \frac{1}{c_0} = \frac{1}{\sqrt{gh}}, & \beta' &= -\frac{h^2}{6c_0^3} = -\frac{h^2}{6(\sqrt{gh})^3}, \end{aligned} \tag{2.2}$$

where c_0 [m/s] is the shallow water wave celerity, c_0' [s/m] the inverse wave celerity, g [m/s^2] the gravitational acceleration, and h [m] the still-water depth.

The idea of this study is to identify the still-water depth at which the KdV equation best relates free surface measurements at consecutive wave gauges. One option would be to numerically propagate the signal at the first gauge to the second gauge using Eq. (2.1) for different water depths, and keep the water depth at which the simulated and measured signal at the second gauge fit best. However, we instead consider the nonlinear spectrum of the signals, which has the advantage that it does not depend on the distance traveled by the wave or the time synchronization of the signals. This property is especially useful when the distance between measurement devices is not known. Finally, this method has the advantage that the equation does not have to be solved with time-stepping methods, which increase in computation time as the distance increases.

2.2.1. NORMALIZATION

The nonlinear Fourier transform is often determined from the normalized and dimensionless KdV equation:

$$q_x + 6qq_t + q_{ttt} = 0, \tag{2.3}$$

in which $q(t,x)$ is the normalized free-surface elevation, t the normalized time and x the normalized location. To obtain the normalized KdV in Eq. (2.3) from the dimensional one in Eq. (2.1), we apply the following change of variables:

$$t = \frac{1}{T_0}(\tau - c_0' l), \quad x = \frac{1}{T_0^3} c_x l, \quad q(t,x) = T_0^2 c_q \eta(\tau, l), \tag{2.4a}$$

$$\text{with} \quad c_x = \beta' = -\frac{\sqrt{h}}{6g^{3/2}}, \qquad c_q = \frac{\alpha'}{6\beta'} = \frac{3g}{2h^2}. \tag{2.4b}$$

The amplitude-normalization coefficient c_q and space-normalization coefficient c_x both only depend on the water depth. Note that the amplitude normalization coefficient c_q represents the ratio of the nonlinearity α' versus the dispersion β'. The time normalization T_0 [s] is entirely free to choose, and will only linearly scale the signal, its linear Fourier spectrum and its NFT spectrum. For simplicity, we will use $T_0 = 1$. We also note here that the value of c_0' is not strictly necessary for the normalization: using a wrong c_0' only translates the signal without influencing its shape, and therefore does not influence the solitonic components or amplitudes of the continuous spectrum of the NFT (see Property 3 in Appendix B). Throughout this study, we will therefore use $c_0' = 0$ for the normalization step. In this study, we will first identify the amplitude normalization coefficient c_q (i.e., the ratio between nonlinear and linear effects), and then extract h from c_q:

$$h = \sqrt{\frac{3g}{2c_q}}. \tag{2.5}$$

2.2.2. SOLITONS AND THE NONLINEAR FOURIER TRANSFORM

It is well known that the KdV equation supports both dispersive waves and waves of translation, so called solitons. The energy of dispersive wave components will spread out more and more over time, but each soliton remains localized, and will be visible indefinitely. More specifically, any normalized wave packet $q(t)$ will evolve into a train of $N \geq 0$ solitons after sufficiently long time [57, p. 83]:

$$q(t,x) \approx \sum_{n=1}^{N} 2k_n^2 \operatorname{sech}^2\left(k_n(t - 4k_n^2 x - t_n^0)\right), \tag{2.6}$$

in which t_n^0 is a time shift depending on the initial conditions. Most importantly, the height $(2k_n^2)$, the width $(1/k_n)$ and the celerity $(4k_n^2)$ of a soliton are all determined by a generalized wave number k_n.

Although the detection of solitons and their k_n is straightforward when all solitons have separated, it is not clear which solitons will come out of some arbitrary wave packet. However, the NFT is able to precisely determine which solitons are present in any wave

packet. If we normalized the KdV equation correctly, the same solitonic components will remain present in a wave packet during propagation. All solitons present in a signal are represented by the discrete spectrum of the NFT, which is a set of pairs of purely imaginary eigenvalues $\lambda_n = ik_n$ (directly related to the generalized wave number of each soliton in Eq. 2.6) and residues r_n (indirectly related to the location of each soliton):

$$\Lambda^{\mathrm{ds}} = \{(\lambda_n, r_n), n = 1, \ldots, N\}. \tag{2.7}$$

Assuming propagation according to Eq. (2.3), the eigenvalues remain constant, and the residues grow exponentially with the traveled distance

$$\lambda_n(x = X) = \lambda_n(x = 0), \quad r_n(x = X) = r_n(x = 0)e^{8i\lambda_n^3 X}. \tag{2.8}$$

We used the *FNFT*-software library [35] to determine the eigenvalues and residues from the normalized signal. Further details on the exact definition and calculations of the NFT may be found in Appendix A. Some relevant properties of the NFT are discussed in Appendix B.

The NFT can also extract the oscillatory wave components from a wave packet, represented by a continuous spectrum. However, we will only focus on the discrete spectrum, as we find that the experimental signals considered in this study are strongly soliton-dominated.

2.3. SPECTRAL MATCHING ALGORITHM

As stated in the previous section, the eigenvalues in the nonlinear Fourier spectrum of a KdV-governed signal remain constant, given that the correct amplitude normalization constant c_q was applied. As c_q relates directly to the water depth h, our strategy is to consider a wave packet at two consecutive wave gauges (an 'input' and an 'output' gauge), normalize both signals with a certain $c_q(h)$ and compare the eigenvalues of their NFT spectrum. The value of c_q is iteratively adapted until one is found at which the NFT eigenvalues at the two wave gauges match optimally. This identified normalization constant c_q^{ID} is then converted to the identified water depth h^{ID} using Eq. (2.4b).

To quantify the error between the NFT eigenvalues of the two signals, we first sort the eigenvalues in each spectrum from largest to smallest imaginary part, $k_1 > k_2 > \cdots > k_N > 0$. We then match the highest input eigenvalue λ_1^{in} to the highest output eigenvalue λ_1^{out}, the second highest to the second highest, and so on. If either spectrum contains more eigenvalues than the other ($N^{\mathrm{in}} \neq N^{\mathrm{out}}$), the remaining (lowest) eigenvalues are matched to artificial '0-eigenvalues' at the origin. We finally considered two possibilities for the error-norm: $p = 1$, corresponding to the absolute difference between input and output eigenvalues; or $p = 3$, proportional to the absolute difference between the *energy* $\mathscr{E}_n$ of input and output eigenvalues ($\mathscr{E}_n \propto k_n^3$, see Eq. (2.23b) in Appendix B). Precisely, we define the error as

$$E^p = \frac{\sum_n \left| \left(k_n^{\mathrm{in}}\right)^p - \left(k_n^{\mathrm{out}}\right)^p \right|}{\sum_n \left(k_n^{\mathrm{in}}\right)^p + \sum_m \left(k_m^{\mathrm{out}}\right)^p}, \quad p \in \{1, 3\}. \tag{2.9}$$

The error is normalized, such that the maximum possible error is 1. Due to the observed continuity of the eigenvalues as c_q is varied, and the fact that eigenvalues can only (dis)appear at the origin (see Appendix B), the error is also continuous in c_q.

We finally identify the optimal c_q^{ID} and corresponding water depth h^{ID} by applying local minimization over c_q using Eq. (2.9) as cost function. To obtain an initial starting position, we perform a rough grid search over realistic values of h, and take the one with lowest error.

This concludes the spectral matching algorithm that we will apply in this study. We note that the spectral matching algorithm is somewhat similar to the identification method from global conserved quantities in Appendix D, as the eigenvalues can also be considered as conserved quantities. As shown in Appendix D, the method of conserved quantities can give good results, while being easy to implement. However, it should be kept in mind that the estimate of the method is not complemented by an error, so it cannot be deduced how reliable the estimate is. The method may also be sensitive to noise and pre-processing, and can give biased results in some cases [54]. The NFT-based algorithm is therefore often the more reliable choice.

Finally, we mention here that other parts of the NFT spectrum can also be taken into account for identifying the water depth, namely the continuous spectrum and the residues of the discrete spectrum. The moduli of the continuous spectrum of the NFT (representing oscillatory wave components) also remain constant during propagation, and could therefore be compared as well. As mentioned before however, the continuous spectrum of our experimental data contained too little energy to test continuous-spectrum-based identification. On the other hand, the residues of the discrete spectrum can be of use, as discussed below.

2.3.1. EXTENSION USING RESIDUES

Although the proposed spectral-matching algorithm only uses the eigenvalues, each eigenvalue is complemented by a residue containing information relating to its position. As each soliton moves with its own speeds, so too do the residues grow with different speeds as the wave packet propagates, as shown in Eq. (2.8). We may thus check how well the solitons match by checking if their residues indeed grew proportionally to the distance between the wave gauges (if known). Alternatively, after the water depth was identified using only the eigenvalues, we may estimate the distance between the input and output gauges using the residues, and compare how well this matches the measured distance.

Instead of considering the residues directly, we will convert each residue to a 'soliton (time-)location', t_n, which we define as the position of the soliton peak in the case the wave packet was a pure single-soliton (the presence of multiple solitons and continuous spectrum cause shifts, as well as that no clear peak may be distinguishable when multiple solitons are close [37, 58]). The propagation speed of this soliton location is equal to

the soliton celerity $c_n = 4k_n^2$:

$$t_n(x) = \frac{1}{2k_n} \ln\left(\frac{r_n(x)}{2ik_n}\right), \tag{2.10}$$

$$t_n(x+X) = t_n(x) + \underbrace{4k_n^2}_{=c_n} X. \quad \text{(propagation relation)} \tag{2.11}$$

Recall that the normalized position is given by $x = \frac{1}{T_0^3} c_x l$, thus the normalized traveled distance is given by $X = \frac{1}{T_0^3} c_x L$, with L the physical distance between the wave gauges.

Finally, the normalized frames of the wave packets at input and output will experience some (unknown) time shift t_0 due to the wave celerity c_0' and due to a possible synchronization mismatch. The influence of this t_0 is the same for all soliton locations, and may thus be taken into account with one t_0 for all pairs (k_n, t_n):

$$t_n^{\text{out}} - t_n^{\text{in}} = 4(k_n^{\text{in}})^2 \frac{c_x}{T_0^3} L + t_0. \tag{2.12}$$

The propagated distance L may thus be identified by fitting both L and t_0 simultaneously in a least-squares approach to the shifts in soliton locations of all solitons, when there are at least two solitons present.

2.4. RESULTS

Within this section, we demonstrate the proposed water-depth identification algorithm using free-surface measurements from a wave flume at two different wave gauges. Furthermore, we also test the identification algorithm on simulation data, where the wave packet at one wave gauge was taken as input, and then numerically propagated with the KdV equation to the position of a consecutive wave gauge.

2.4.1. EXPERIMENTAL SETUP

The experimental data used in this study was measured at the Hydraulic Engineering Laboratory at the National University of Singapore (NUS). The data was originally generated for a different study [58], and shared with us for the analysis in this study. The experimental setup is shown in Fig. 2.1. The flume is 0.9 m wide, 0.9 m deep, and 36 m long. At the left-end side is a movable piston-type wave maker. After 28.7 m, the flume bottom slopes upward. Four wave gauges marked CG1 to CG4 (Capacity Gauge) are present, with respective positions $x_1 = 5.193$ m, $x_2 = 9.886$ m, $x_3 = 14.882$ m, and $x_4 = 18.869$ m. The wave gauges consist of two thin metal wires, which measure the water-height dependent voltage difference. The voltage is then linearly converted to the wave height. Measurements were taken every 0.05 s, for a duration of 90 s. Initially, the water was at rest with measured depth $h = 9.80$ cm. Then, the wave maker was moved from $x = 0$ m to $x = 4.0$ m in approximately 3 s to generate a trapezoidal-shaped wave with a relative height of 2.0 cm. Note that the wave maker remained at $x = 4.0$ m, so the filled part of the flume became shorter, and the still water level would increase after all waves have died out. This takes a very long time, and for the course of the experiment, we took the initial still water level $h^{\text{ref}} = 9.80$ cm as reference level.

The measurements of the free-surface elevation are shown in Fig. 2.1. Initially, we observe the initial trapezoidal-shaped wave at CG1, right behind the wave maker. As the wave travels, its fastest solitons start to separate from the bore, and reach the next gauges first. Traveling at approximately $c_0 = 1\,\mathrm{m/s}$, the front of the wave reaches the right end of the flume around $t = 28\,\mathrm{s}$, reflects imperfectly from the sloped end, and propagates to the left, reaching the gauges in opposite order. Around $t = 51\,\mathrm{s}$, the (now left-going) wave reflects nearly perfectly at the piston at the left side of the flume, and starts moving to the right again. The second reflection and first reflection interfere at most of the wave gauges, contradicting the assumption of unidirectional waves in the KdV approximation.

We will compare only the incoming waves at the wave gauges to identify the water depth, as these measurements are not affected by interference. We compare several combinations to investigate the influence of the traveled distance on the identified water depth: if two gauges are very close, the waves are very similar, and thus the influence of the water depth will be small, making it harder to identify; for longer distances, the accumulated effect of the water depth will be greater, and thus the water depth should be easier to identify.

2.4.2. PRE-PROCESSING

For the pre-processing of the data, we extracted a time window as large as possible without being distorted by the reflected wave. We truncated the signals measured at CG1 to CG4 to the time interval $t \in [0, 34]$. We note here that the NFT assumes zero-boundary conditions at the left and right of the signals. Although the signal smoothly goes to zero at $t = 0$, this is not the case at $t = 34$, where the new steady water level was measured to be $-0.40\,\mathrm{cm}$. The block signal is followed by a trough, which does not seem to go back to the initial still-water level within the time frame of the experiment (in fact the water level seems to decrease further down to $-0.72\,\mathrm{cm}$ after the reflected wave has passed over). Fortunately, it was observed in [58] that the presence or absence of a trough mainly influences the continuous spectrum, while leaving the discrete spectrum nearly unchanged. We confirmed that this is indeed the case, by comparing the discrete spectra of the signal at CG1 with and without trough (see Appendix C).

As the trough influences the discrete spectrum only marginally, we cut off the right part of the signal at the time that the tail of the wave packet reaches zero for the first time, and replace the trough with zeros, following the method of [58]. This ensures that the vanishing boundary conditions are satisfied without a jump in water level at the right hand side.

Finally, we perform a small rescaling of the data, similar to the pre-processing in [58]. We rescale the data by comparing the first moment $\int_{-\infty}^{\infty} \eta \mathrm{d}t$ and the second moment $\int_{-\infty}^{\infty} \eta^2 \mathrm{d}t$ of the wave packets. These two integrals are conserved quantities of the KdV equation [39], and thus should be equal at all four wave gauges. Cutting off the trough is important for this recalibration, as both moments would otherwise have depended on the considered length of the trough. We rescaled the data such that the second moment (proportional to the wave energy) of all signals are equal to the one at the first wave gauge, resulting in the following rescaling factors for the free-surface elevation for CG1 to CG4: [1, 0.990, 0.989, 1.013]. Although these changes are small, we found that they significantly improve the matching of the discrete spectra and the estimation of h.

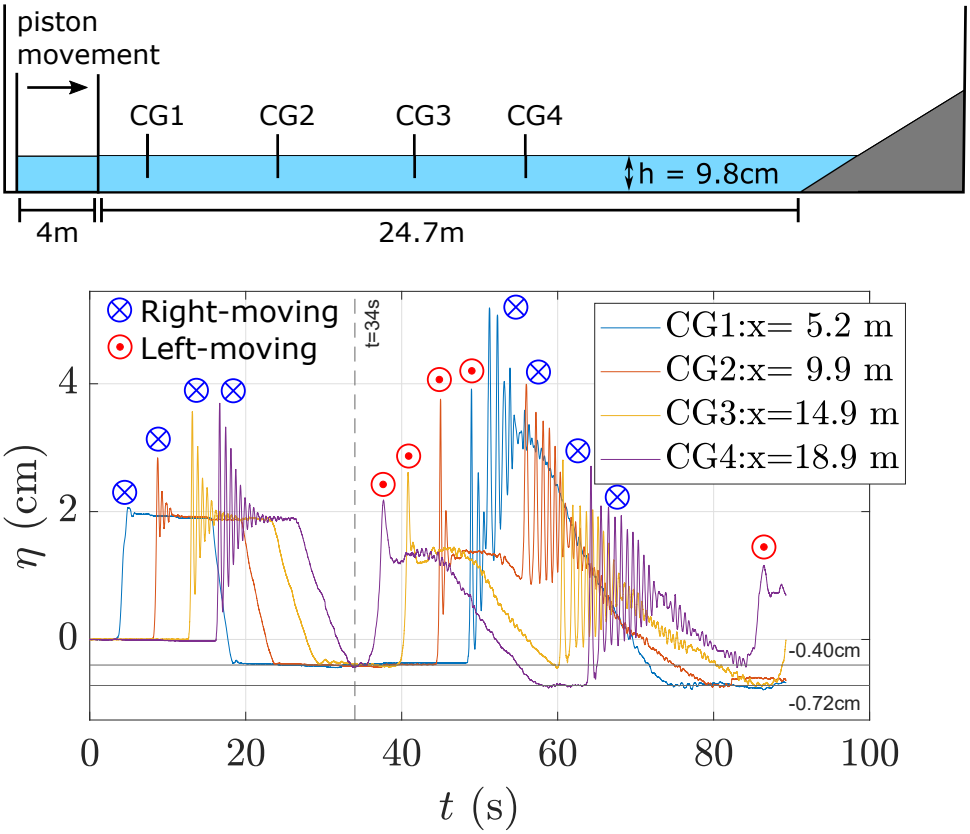


Figure 2.1: Top: a side view of the experimental setup, with the positions of the four wave gauges. Bottom: the measured wave height with respect to initial still-water level at the four wave gauges. The vertical dashed line marks the time up to which the wave gauges are not affected by the reflected wave. Re-made after [58].

After the pre-processing of the data, we calculated the energy in the discrete and the continuous NFT spectrum for the signal at CG1, to check our assumption that the signal is soliton dominated. Using $h^{\text{ref}} = 9.80\,\text{cm}$ for the normalization in the NFT, we found from Eq. (2.23b) in Appendix B that the discrete spectrum contained 99.9% of the signal energy, satisfying our assumption.

2.4.3. NUMERICAL VALIDATION OF THE ALGORITHM

Before considering the experimental data, we first validate the spectral matching algorithm on simulated data, generated using the KdV equation. We took the pre-processed data at wave gauges CG1, CG2 and CG3 as input data, and numerically propagated each input signal to the position CG4, using the dimensional KdV equation in Eq. (2.1) with h^{ref} for all coefficients. For this numerical case, the eigenvalues should remain constant

when normalizing the signal with h^{ref}, and the spectral matching error should be zero up to numerical errors. Fig. 2.2 shows from top to bottom the input-output data (the blue and black lines), the corresponding eigenvalues as function of h, and the spectral matching error as function of h. From the normalized eigenvalues we observe that as h decreases (i.e., higher nonlinearity), the number of detected solitons increases, while the eigenvalues all drift upwards. We observe consistently that eigenvalues of the input signal (blue) and the eigenvalues of the simulated output (black) exactly coincide at $h = h^{\text{ref}}$, while drifting apart for other values of h. As a result, the spectral matching errors E^1 and E^3 show a clear minimum at h^{ref} for the simulated data set, while the errors increase as h is further from the reference value. This validates that the spectral-matching algorithm performs well when the considered data is exactly governed by the KdV equation.

However, we observe from the pre-processed signals in Fig. 2.2 that the numerical output signals (black) differ somewhat from the experimental signals (red). The KdV only models the wave propagation approximately, and the model mismatch becomes apparent here. In particular, the KdV seems to develops the shape of the input wave too fast, which can also be visually deduced from the fact that the highest numerical soliton arrives earlier than the experimental soliton, and the distance between the solitons is larger than in the measured wave. Also the right tail of the numerical wave packet flattens faster than the tail the measured output.

Although the KdV does not perfectly describe the wave propagation of our experimental data, it was shown in [58] that the solitonic components predicted by the KdV-NFT were indeed present. Therefore, our proposed method of comparing solitonic components at different gauges may still perform well despite the observed model mismatch.

2.4.4. EXPERIMENTAL RESULTS OF NFT-BASED WATER-DEPTH IDENTIFICATION

We now consider the spectral matching algorithm for the pre-processed experimental data. First, we determined the E^1-error and E^3-error over a grid, as shown in Fig. 2.2. We observe that the E^3-error seems to be much smoother than the E^1-error. This is due to the appearance of new eigenvalues, that can quickly grow in amplitude and contribute significantly to the error. The energy-based error E^3 overcomes this drawback, as the energy of these new eigenvalues is only very small. Due to the smoothness of E^3, we will use this error norm for determining our final estimation of the water depth for each of the data sets. After the grid search, we apply local minimization of E^3 to identify the water depth h^{ID} at which E^3 is minimized.

For the experimental data, the optimal matching with the input eigenvalue is close to the reference value for all considered cases. In particular, the minimum error for the CG2-CG4 case was extremely low, and corresponds very well to the reference value. The largest error observed (for the CG1-CG4 case, $h^{\text{ID}} = 9.34\,\text{cm}$) was less than 5% off the reference value. When using the reference water depth, we found that for all signals the highest eigenvalues matched well, except for CG1-CG4. Our hypothesis is that the wave at CG1 is still influenced by the deceleration of the piston, as it is only 119 cm behind the piston. This deceleration may have caused a larger mismatch with the KdV equation during the first few meters of propagation, causing the spectra at the first gauge to be

different from the other gauges.

Next, we also observe that the identified water depth for the CG3-CG4 case ($h^{\mathrm{ID}} = 10.10\,\mathrm{cm}$) is also relatively far off (3%). This is probably due to the fact that these gauges are very close, so the total influence of the water depth on the propagation becomes harder to measure. This may also be observed from the fact that the error well for the CG3-CG4 case is less steep than in the other cases.

The CG2-CG4 case provides the best setup: it features a sufficiently long distance, while not being influenced (too much) by the start-up interference due to the piston. As a result, we observe that the minimum observed E^3 is very close to 0, indicating an excellent match between input and output eigenvalues. The identified water depth ($h^{\mathrm{ID}} = 9.70\,\mathrm{cm}$) is only 1% off the reference depth. This validates that the water depth can be identified very accurately when using suitable data. However, even when our data contained more distortion or was measured at close gauges, we still managed to get estimates with less than 5% error.

2.4.5. VALIDATION OF NFT-BASED IDENTIFICATION USING NUMERICAL PROPAGATION

To further validate the NFT-based water depth identification method, we compare it to an approach based on numerically forward propagating the input wave to the position of the output gauge, using the dimensional tKdV from Eq. (2.1) for various values of h. Although this method lacks the benefits from NFT-based identification (solution depends on numerical space-step size, propagation distance must be known or has to be identified as well, time synchronization is often required), it is one of the most straightforward approaches to find the governing water depth in the KdV equation.

This method propagates the input signal for a distance L (space step size $\Delta l = 0.02\,\mathrm{m}$) to the position of the output wave gauge using the dimensional KdV in Eq. (2.1), for different values of h. We then identify the water depth as the h at which the forward-propagated input signal $\eta^{\mathrm{in,\,prop}}(t;h,L)$ matches $\eta^{\mathrm{out}}(t)$ as well as possible. As it may often occur that the time measurements at wave gauges are not properly synchronized, we will only consider the shapes of the wave packets, and allow for a horizontal time-translation t_0. Our error-norm for this propagation-based matching is

$$E^{\mathrm{prop}}(h,L) = \min_{t_0} \frac{\int_{-\infty}^{\infty} \left|\eta^{\mathrm{in,\,prop}}(t;h,L) - \eta^{\mathrm{out}}(t-t_0)\right| \mathrm{d}t}{\int_{-\infty}^{\infty} \left|\eta^{\mathrm{out}}(t)\right| \mathrm{d}t}. \tag{2.13}$$

From Fig. 2.2, we have already observed that the simulated output using h^{ref} is already quite different form the measured output. We observed that this was caused by a model mismatch, where the KdV equation would develop the shape of the wave faster than the measured wave. Therefore, we will optimize over both the water depth and the traveled distance L. We consider the case with CG2-CG4, as this gave the best fit for the eigenvalue-matching, and we thus expect that this dataset fits KdV-propagation best, although probably for a different propagation distance.

The result is shown in Fig. 2.3. As observed before, the measured signal and the signal propagated with $h_{\mathrm{ref}} = 9.8\,\mathrm{cm}$ and $L = 9\,\mathrm{m}$ do not fit well, as the solitons have already separated too much, indicating that the simulated wave has indeed developed too

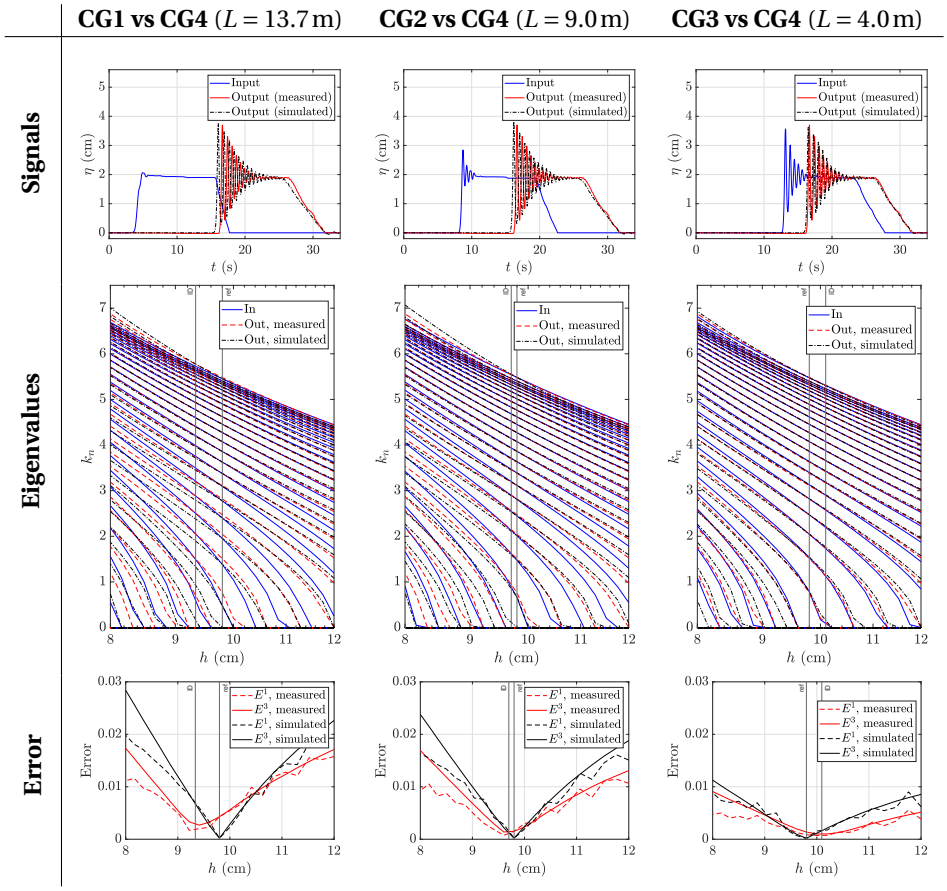


Figure 2.2: The used data for the identification algorithm. Top: the pre-processed data at various wave gauges, and the theoretical output when the input signal was propagated according to the KdV at the reference water depth. Middle: the normalized eigenvalue energies ($\sim k_n^3$) for various water depths. Bottom: the error between the input and output eigenvalues, and the identified water depth using local minimization of E^3.

much. The third line was optimized only for the traveled distance, and shows a good fit, but an effective propagation distance of only $L = 5$ m. Optimizing over both the depth and the propagation distance results in the fourth line, and we identify $h = 9.57$ cm and $L = 5.68$ m. This identified water depth is only 2% off the measured value, and in close correspondence with the identified value of NFT-based matching ($h = 9.70$ cm). For CG1-CG4 we identified $h = 9.33$ cm, which is also in close correspondence with the value from NFT-based identification ($h = 9.34$ cm), which shows that NFT-based matching indeed gives similar results as the propagation-based method. Only for the CG3-CG4 case, the identified values were further apart, $h = 9.20$ cm for the propagation-based method, and $h = 10.10$ cm for the NFT-based method. This difference may be explained due to

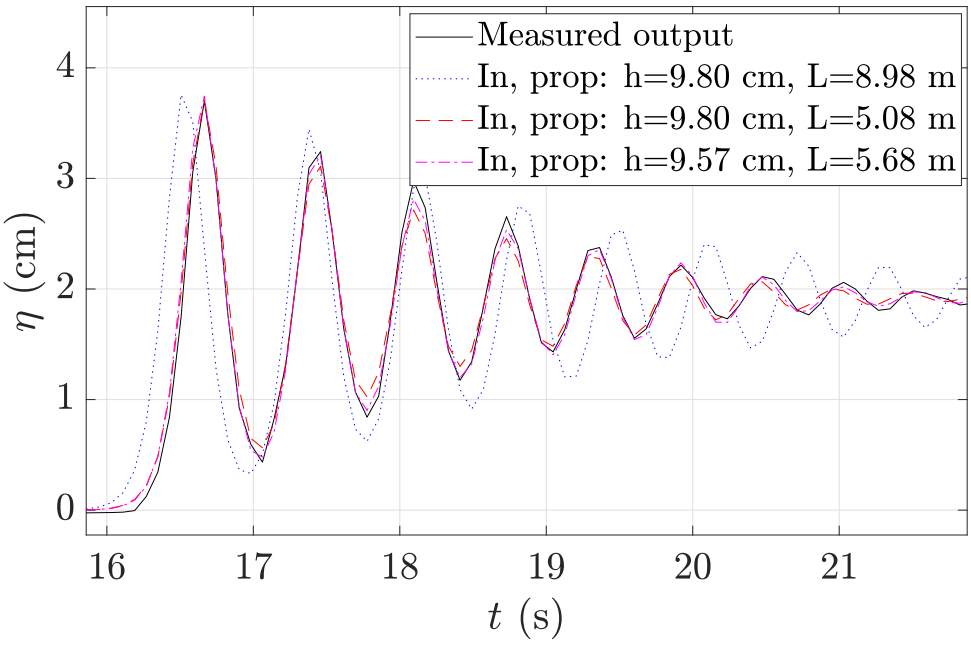


Figure 2.3: The bore front of the measured output at CG4, compared to the KdV-propagated signal from CG2 for various combinations of distance l and water depth h, while time-synchronizing as well as possible. The second curve uses the reference values for h and l, the third curve is the result of local optimization for the traveled distance, and the fourth curve for simultaneously optimizing for both the distance and the water depth.

the short distance between the wave gauges, so the effects of the underlying mechanics are hard to measure.

All together, KdV-propagation-based water depth identification indeed gives similar results as NFT-based water-depth identification, when also allowing the distance and the time-offset as parameters in the propagation-based matching. However, we did observe that the NFT-based estimates were all slightly closer to the measured water depth, as well that the propagation-based method required careful considerations regarding the numerical instabilities and the space-step size. Although we did not pay special attention to computation time, we do mention here that the NFT-based method was somewhat faster than the propagation-based method using the KdV-solver from the software *Chebfun* [59].

2.4.6. VALIDATION OF NFT-BASED IDENTIFICATION USING DISCRETE SPECTRUM RESIDUES

The NFT-based method managed to identify the water depth with at most 5% error by only using the eigenvalues. However, it did not take the soliton location into account. In this section, we show that the soliton locations can be used for validation purposes as well, for example to identify the traveled distance with the propagation relation in

Eq. (2.12). During the propagation-based identification, we observed that the identified propagation distance was about 30% lower than expected when only considering the wave shape. We show here that the soliton locations indicate a similar result.

We consider here only the case CG2-CG4, as this data showed the closest agreement with the KdV model. Fig. 2.4 shows the calculated soliton locations at the input (CG2) and at the output (CG4). As expected, the solitons arrived later at the output gauge than at the input gauge, and thus their t_n are higher. The right figure shows the difference in soliton locations between the input and output gauge. We observe that the solitons with higher k_n moved faster in general, as their soliton locations required less time to cover the distance between CG2 and CG4 (the fastest required only 7.8 s, the slowest 9.1 s). Against expectation however, we observe that the theoretically fastest (with the largest k_n) has not moved faster than some other solitons. This could be due to random effects or a slight model mismatch. However, the general trend looks in correspondence with Eq. (2.12): each soliton location should have moved ahead of the wave packet quadratically in its generalized wave number k_n.

We fitted the difference in soliton location for L and t_0 according to Eq. (2.12), using all soliton locations except for those of the lowest three solitons, as these contain little energy and are prone to noise. We identified a propagation distance of $L = 6.4$ m, which is significantly lower than the actual wave gauge distance of 9.0 m for the CG2-CG4 case. We emphasize that this distance takes only takes into account how much the shape of the packet has developed, which is unrelated to the wave speed c_0 or inverse wave speed c_0'. We note that the identified value of $L = 6.4$ m is close to value identified by the propagation based method ($L = 5.68$ m). Also, upon performing a similar analysis for CG1-CG4, and for CG3-CG4, we find that the identified propagation distance is systematically about 30% lower than the measured gauge distance. Apparently, the KdV equation does provide the correct solitons, but overestimates the wave development speed.

2.5. CONCLUSION

We proposed a method to identify the water depth from wave packets approximately governed by the KdV equation, based on the nonlinear Fourier transform of free-surface measurements. By comparing the solitonic wave components in a wave packet in a wave flume at two consecutive locations, we were able to determine the water depth with at most 5% error, and only a 1% error under the most suitable circumstances. Furthermore, when using simulated data, the identified water depth matched the reference value almost exactly for all considered cases, and outperformed a method based on numerical propagation of the KdV equation. We finally demonstrated that the residues of the NFT spectrum could also be used to extend the identification method. Further research could focus on validation and application of the method outside a lab environment, such as water-depth identification from coastal buoy data.

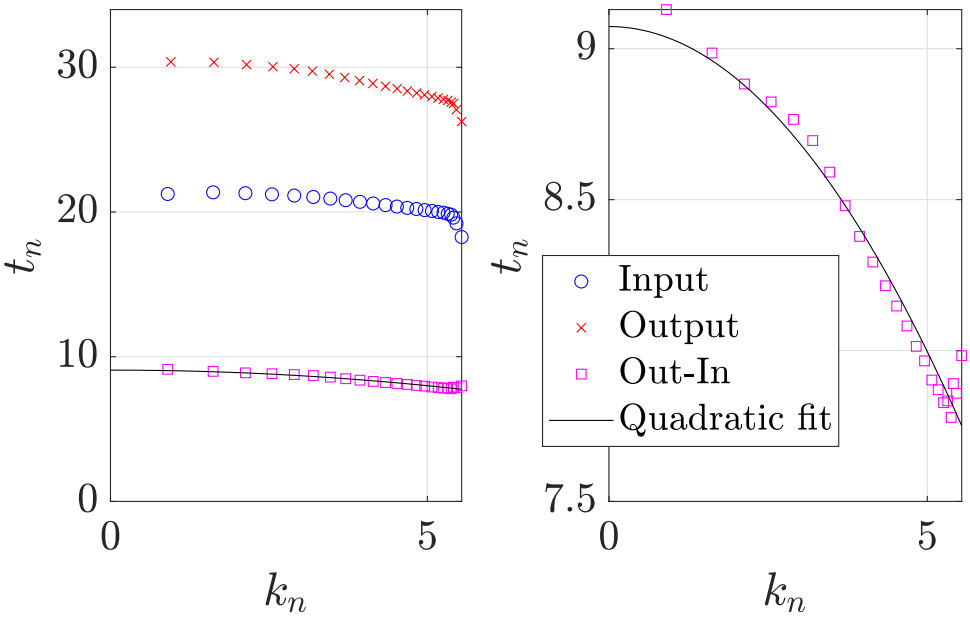


Figure 2.4: The soliton locations t_n according to Eq. 2.10, and the quadratic fit between the difference (zoomed in at the right).

APPENDIX A: DEFINITION AND CALCULATION OF THE NFT SPECTRUM

Given a system governed by the normalized KdV equation from Eq. (2.3), we may determine the NFT while assuming vanishing boundary conditions for the wave packet, $q(t) \to 0$ as $t \to \pm\infty$ fast enough. The NFT is defined through the Schrödinger eigenvalue problem, which uses the wave packet as potential [1, 7, 8]:

$$\phi_{tt} + q(t)\phi = (i\lambda)^2\phi, \tag{2.14}$$

where ϕ is the eigenfunction and $\lambda \in \mathbb{C}$ the spectral parameter. For a real signal $q(t)$, the Schrödinger eigenfunction problem only allows two types of solutions such that the eigenfunctions ϕ do not blow up. The first type of solutions ϕ correspond to $\lambda \in \mathbb{R} \setminus \{0\}$, which result in oscillatory eigenfunctions of infinite energy, but finite power. These solutions relate to the so-called continuous spectrum, and represent dispersive wave components in the signal $q(t)$. The second type of solutions consist of a finite number of discrete, purely-imaginary eigenvalues in the upper-half plane $\lambda_n = ik_n$, $0 < k \in \mathbb{R}$, $n = 1, \ldots, N$. The corresponding eigenfunctions ϕ_n decay exponentially in both tails and are finite-energy. It is well known that these discrete eigenvalues correspond to the solitonic wave components present in the signal $q(t)$, which remain stable during propagation of $q(t)$ [1, 7], and eventually separate into the train of solitons from Eq. (2.6).

One of the most convenient ways of solving the Schrödinger eigenvalue problem is by rewriting Eq. (2.14) as a system of two first order differential equations. We do so by

switching to the basis from [37], in which the Schrödinger scattering problem is given as follows:

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \frac{1}{2i\lambda} \begin{bmatrix} i\lambda - \frac{\partial}{\partial x} \\ i\lambda + \frac{\partial}{\partial x} \end{bmatrix} \phi, \qquad \text{(change of basis)} \tag{2.15a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} v_1(t,\lambda) \\ v_2(t,\lambda) \end{bmatrix} = \begin{bmatrix} -i\lambda + \frac{q(t)}{2i\lambda} & \frac{q(t)}{2i\lambda} \\ -\frac{q(t)}{2i\lambda} & i\lambda - \frac{q(t)}{2i\lambda} \end{bmatrix} \begin{bmatrix} v_1(t,\lambda) \\ v_2(t,\lambda) \end{bmatrix}, \tag{2.15b}$$

$$\begin{bmatrix} e^{-i\lambda t} \\ 0 \end{bmatrix} \xleftarrow[q\to 0]{t\to-\infty} \begin{bmatrix} v_1(t,\lambda) \\ v_2(t,\lambda) \end{bmatrix} \xrightarrow[q\to 0]{t\to+\infty} \begin{bmatrix} a(\lambda)e^{-i\lambda t} \\ b(\lambda)e^{+i\lambda t} \end{bmatrix}. \quad \text{(BCs)} \tag{2.15c}$$

Here, Eq. (2.15b) corresponds with the Schrödinger eigenvalue problem, and Eq. (2.15c) are the boundary conditions (BCs) that we impose to obtain the so-called scattering coefficients $a(\lambda)$ and $b(\lambda)$.

The full nonlinear Fourier transform spectrum consists of a discrete spectrum and a continuous spectrum. The discrete spectrum (ds) Λ^{ds}, contains all purely imaginary eigenvalues $\lambda_n = ik_n$ in the upper-half plane such that the eigenfunction $[v_1, v_2]^T$ is finite energy. This can only occur if $a(\lambda_n) = 0$. To complete the discrete spectrum, the eigenvalues are supplemented by their residues r_n, which relate to the locations of the solitons:

$$\Lambda^{\mathrm{ds}} := \left\{ \left(\lambda_n = ik_n, r_n = \frac{b}{a_\lambda}(\lambda_n) \right) : a(ik_n) = 0, \text{ and } 0 < k_n \in \mathbb{R} \right\}, \text{ with } a_\lambda(\lambda_n) = \left. \frac{\partial a(\lambda)}{\partial \lambda} \right|_{\lambda=\lambda_n}. \tag{2.16}$$

The continuous spectrum (cs), Λ^{cs}, consists of the so-called reflection coefficient b/a on the real line:

$$\Lambda^{\mathrm{cs}} := \left\{ \frac{b}{a}(\xi), \xi \in \mathbb{R} \setminus 0 \right\}. \tag{2.17}$$

As the wave packet evolves over x according to the normalized KdV equation, the eigenvalues remain constant, and the scattering coefficients a and b evolve trivially [7, 56]:

$$\lambda_n(x) = \lambda_n(0), \tag{2.18a}$$

$$a(\lambda, x) = a(\lambda, 0), \tag{2.18b}$$

$$b(\lambda, x) = b(\lambda, 0)e^{8i\lambda^3 x}. \tag{2.18c}$$

As the λ_n remain constant, so do the k_n.

APPENDIX B: PROPERTIES OF THE NFT

Property 1 (Scattering coefficients codomain) *The scattering coefficients $a(\lambda)$ and $b(\lambda)$ satisfy [7]:*

$$|a(\lambda)| \in [1, \infty), \quad |b(\lambda)| \in [0, \infty), \quad \left| \frac{b}{a}(\lambda) \right| \in [0, 1), \tag{2.19a}$$

$$|a(\lambda)|^2 - |b(\lambda)|^2 = 1. \tag{2.19b}$$

Property 2 (Linear Fourier transform as $q \to 0$) *In the small amplitude limit, $q(t) \to 0$, the Fourier transform degenerates to the linear Fourier transform. No eigenvalues will be present, and the continuous spectrum and linear Fourier spectrum relate as follows [56, Eq. (4.7)]:*

$$\lim_{q(t)\to 0, |b/a|\to 0} 2i\xi \frac{b}{a}(\xi) = -\mathscr{F}\{q\}(\omega = 2\xi), \tag{2.20}$$
$$\text{with } \mathscr{F}\{q\}(\omega) = \int_{-\infty}^{\infty} q(t) e^{-i\omega t}\,\mathrm{d}t \quad \text{the Fourier transform,}$$

in which ω denotes the linear angular frequency.

Property 3 (Time translation) *Let $\{a(\lambda), b(\lambda)\} = NFT\{q(t)\}$ be the scattering coefficients of the NFT corresponding to the signal $q(t)$. Then a time translation $t \to t - t_0$ in the signal results only in an exponential term in the b-coefficient [7]:*

$$\left\{a(\lambda), b(\lambda) e^{-2i\lambda t_0}\right\} = NFT\{q(t - t_0)\}. \tag{2.21}$$

Note that, the eigenvalues λ_n (i.e., the zeros of $a(\lambda)$) remain unchanged, as $a(\lambda)$ remains unchanged.

Property 4 (Energy in the KdV-NFT) *Let the energy of a normalized KdV-governed signal be defined as*

$$\mathscr{E}^q = \int_{-\infty}^{\infty} q(t)^2\,\mathrm{d}t = \frac{1}{2\pi}\int_{-\infty}^{\infty} \left|\mathscr{F}\{q\}(\omega)\right|^2 \mathrm{d}\omega, \tag{2.22}$$

then the energy of the discrete spectrum $\mathscr{E}^{ds}$ and of the continuous spectrum $\mathscr{E}^{cs}$ are given as follows [60, p.286], [61, 1.6.21b]:

$$\mathscr{E}^q := \mathscr{E}^{cs} + \mathscr{E}^{ds}, \tag{2.23a}$$
$$\mathscr{E}^{ds} = \sum_{n=1}^{N} \underbrace{\frac{16}{3} k_n^3}_{\mathscr{E}_n}, \tag{2.23b}$$
$$\mathscr{E}^{cs} = \frac{1}{\pi}\int_{-\infty}^{\infty} -4\xi^2 \ln\left(1 - \left|\tfrac{b}{a}(\xi)\right|^2\right)\mathrm{d}\xi, \tag{2.23c}$$

where $\mathscr{E}_n$ is the energy of solitonic component $\lambda_n = ik_n$. We note here that for the low signal-amplitude case, the continuous-spectrum energy reduces to the energy in the linear Fourier spectrum: $\mathscr{E}^{cs} \xrightarrow{|b/a|\to 0} \frac{1}{\pi}\int_{-\infty}^{\infty} 4\xi^2 \left|\frac{b}{a}(\xi)\right|^2 \mathrm{d}\xi = \frac{1}{2\pi}\int_{-\infty}^{\infty} \left|\mathscr{F}\{q\}(\omega = 2\xi)\right|^2 \mathrm{d}\omega$, aligning with Property 2

Conjecture 5 (Continuity of NFT in c_q (unproven)) *Let $q(t; c_q) := c_q \eta(t)$, and $\{\lambda_n(c_q),\ n = 1 \ldots N(c_q)\} = NFT\{c_q \eta(t)\}$ denote the eigenvalues of a signal $\eta(t)$ as a function of the normalization coefficient c_q. Then the position of each eigenvalue $\lambda_n(c_q)$ is a continuous function of c_q over the imaginary axis. Moreover, eigenvalues can only (dis)appear at the real axis. Increasing c_q increases the signal energy, and most often causes existing eigenvalues to drift upwards, while new eigenvalues appear from the real axis.*

APPENDIX C: THE INFLUENCE OF A TROUGH ON THE DISCRETE SPECTRUM

We validate that the presence or absence of a trough after a signal has little influence on the discrete spectrum, which was also mentioned (although not shown) in [58]. We show here that indeed the trough has little influence on the eigenvalues for the incoming signal at wave gauge 1. We consider three different cut-off points: 1) at 18 s, when the bore reaches the still-water level for the first time again; 2) at 34 s, the time frame considered in all experiments; 3) at 48 s, the largest time possible before the reflected wave reaches wave gauge 1 again. The three signals with different cut-off point are shown in Fig. 2.5. The eigenvalues were determined for the reference water depth $h^{\text{ref}} = 9.80\,\text{cm}$. All eigenvalues are visually identical, except for the lowest eigenvalue (which has very little signal energy). This confirms that the trough indeed has very little influence on which solitons are present in the discrete spectrum.

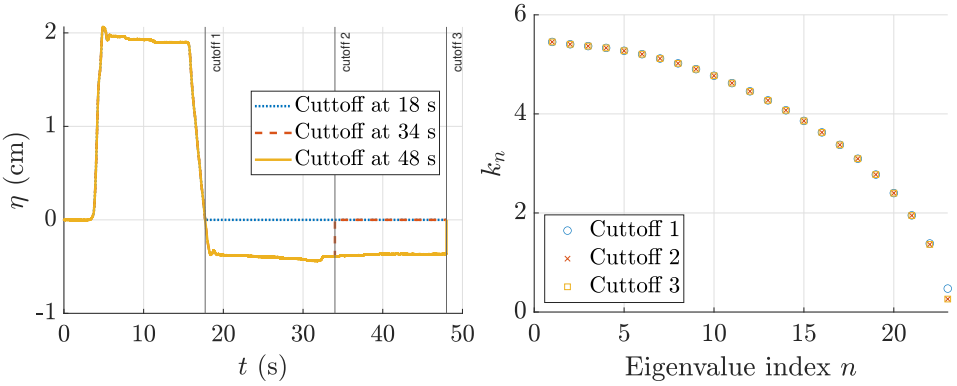


Figure 2.5: Signals with different cut-off points for their troughs, and their corresponding eigenvalues for the reference water depth. the effect of the trough is only visible for the last (lowest-energy) soliton.

APPENDIX D: IDENTIFICATION USING CONSERVED QUANTITIES

It is well known that the normalized KdV equation in Eq. (2.3) has an infinite number of conserved quantities or constants of motion. These quantities are conserved during propagation for tKdV-governed experimental data, given that the correct amplitude normalization coefficient c_q was used. It is thus possible to identify c_q as the value for which these quantities are indeed equal at input signal and output. A similar method has also been demonstrated for NLSE-governed signals in [54].

The first two constants of motion of the KdV are related to conservation of mass and conservation of energy. The consecutive constants are less intuitive, and will be referred to as the third moment, fourth moment, etc. The first conservation laws for KdV in

Eq. 2.3 are given as follows [25, 39]:

$$\begin{aligned} C_1 &= \int q \,\mathrm{d}t, & C_3 &= \int \left(-2q^3 + q_t^2\right) \mathrm{d}t, \\ C_2 &= \int q^2 \,\mathrm{d}t, & C_4 &= \int \left(9q^4 - 18qq_t^2 + \tfrac{9}{5}q_{tt}^2\right) \mathrm{d}t. \end{aligned} \tag{2.24}$$

When we equate $C_n^{\text{in}} = C_n^{\text{out}}$, and substitute $q^{\text{in}}(t) = T_0^2 c_q \eta^{\text{in}}((\tau - c_0 l)/T_0)$, and $q^{\text{out}}(t) = T_0^2 c_q \eta^{\text{out}}((\tau - c_0 l)/T_0)$, we will find that T_0 always drops out. However, c_q also drops out if only a single power of q is present, as is the case for C^1 and C^2. Therefore, C^3 is the first quantity for which c_q does not drop out upon equating $C_3^{\text{in}} = C_3^{\text{out}}$ (we set $T_0 = 1$ to simplify the demonstration):

$$\begin{aligned} &\int_{-\infty}^{\infty} -2\left(c_q \eta^{\text{in}}(\tau)\right)^3 + \left(c_q \eta_\tau^{\text{in}}(\tau)\right)^2 \mathrm{d}\tau \\ &\qquad = \int_{-\infty}^{\infty} -2\left(c_q \eta^{\text{out}}(\tau)\right)^3 + \left(c_q \eta_\tau^{\text{out}}(\tau)\right)^2 \mathrm{d}\tau, \\ \Rightarrow\ & c_q^2 \int_{-\infty}^{\infty} \left(\eta_\tau^{\text{in}}(\tau)\right)^2 - \left(\eta_\tau^{\text{out}}(\tau)\right)^2 \mathrm{d}\tau \\ &\qquad = 2c_q^3 \int_{-\infty}^{\infty} \left(\eta^{\text{in}}(\tau)\right)^3 - \left(\eta^{\text{out}}(\tau)\right)^3 \mathrm{d}\tau, \\ \Rightarrow\ & c_q = \frac{\int_{-\infty}^{\infty} \left(\eta_\tau^{\text{in}}(\tau)\right)^2 - \left(\eta_\tau^{\text{out}}(\tau)\right)^2 \mathrm{d}\tau}{2\int_{-\infty}^{\infty} \left(\eta^{\text{in}}(\tau)\right)^3 - \left(\eta^{\text{out}}(\tau)\right)^3 \mathrm{d}\tau}. \end{aligned} \tag{2.25}$$

This relation provides a fast and easy method to obtain estimates for c_q. Although all conserved quantities except for C_1 and C_2 can be used to find c_q in a similar fashion, the higher conserved quantities contain higher derivatives and powers, which are increasingly sensitive to noise. Furthermore, we can see that in C_4 three different orders of c_q will pop up (order 4, 3 and 2), which will lead to an underdetermined system if we only have measurements at two wave gauges. Even when three gauges are considered, a system of equations will have to be solved, which is probably more prone to noise. We will therefore only use C_3.

Using this method on the simulated data from CG2-CG4, and converting c_q to h, we successfully recovered the correct water of $h = 9.8\,\text{cm}$, validating the method for ideal KdV-governed data. Also when applying this method to the pre-processed experimental data, we identified depths close to the measured water depth, as shown in Table. 2.1. Although this method is very fast and can yield good results, it can be sensitive to noise and offsets due to its dependence on derivatives and high powers in q. Also due to the fact that only a single number is given as output, it is unknown how reliable this number is. By itself, this method is therefore often not suitable for application in practice, but may in many cases provide initial estimates or additional validation.

Table 2.1: The identified water depths from the pre-processed wave gauge data.

Data set	Measured	CG1-CG4	CG2-CG4	CG3-CG4
h	9.80 cm	9.40 cm	9.59 cm	9.90 cm

3

DISPERSION AND NONLINEARITY IDENTIFICATION FOR SINGLE-MODE FIBERS USING THE NONLINEAR FOURIER TRANSFORM

Efficient fiber-optic communication requires precise knowledge of the fiber coefficients, but these often change over time due to factors such as aging or bending. We propose a novel algorithm that identifies the average second-order dispersion and Kerr nonlinearity coefficient of a fiber, without employing any special training signals. Instead, ordinary input and output data recorded during normal operation is used. To the best of our knowledge, this is the first such algorithm. The algorithm is based on the nonlinear Fourier spectrum of the signal, which is known to evolve trivially as the signal propagates through an idealized model of the fiber. The algorithm varies the values of the fiber coefficients until the corresponding nonlinear Fourier spectrum at transmitter and receiver match optimally. We test the algorithm on simulated transmission data over a 1600 km link, and accurately identify the fiber coefficients. The identification algorithm is in particular well suited for providing a fiber model for nonlinear Fourier transform-based communication. This chapter is an adaptation of [54]. ©2020 IEEE. Reprinted, with permission.

3.1. INTRODUCTION

Digital signal processing has become increasingly more important in fiber-optic communication systems with high data rates, as it allows to compensate for transmission impairments such as chromatic dispersion and the Kerr nonlinear effect [62–64]. In order to digitally compensate such effects, the fiber is often modelled with a lossy and noisy nonlinear Schödinger equation, in which the second-order dispersion coefficient β_2 and the Kerr nonlinearity coefficient γ are assumed to be known. The values of these two coefficients are usually supplied by the manufacturer of the fiber link, but it often occurs that the supplied values do not exactly fit the fiber anymore after installation of

the fiber due to bending, aging, and splicing [62]. It may also occur that the fiber coefficients are not readily available, or lost, as may be the case in small scale experimental setups. For such scenarios, we present a novel nonlinear Fourier transform (NFT)-based fiber identification algorithm to determine the values of β_2 and γ, using input-output transmission data only. This study improves and extends our earlier NFT-based fiber identification algorithm [65] in terms of robustness, versatility and accuracy.

Several fiber identification algorithms have already been developed, but these either identify only β_2 (e.g., [66, 67]), or identify γ using digital back-propagation (e.g., [68–71]). However, these algorithms using digital back-propagation suffer from at least one of the following shortcomings: β_2 has to be known; a specific modulation format has to be used; the quality of the estimates depends on the chosen spatial numerical step. Other methods determine γ by measuring self-phase modulation, cross-phase modulation, or four-wave mixing using training signals [72–74], but these require that normal operation of the fiber is interrupted, which can be undesirable in scenarios where the system is in constant operation, or when the training signals are not straight-forward to generate or analyze. Furthermore, several of these identification algorithms were demonstrated for short fibers, and may therefore be less suitable for identifying long links. Finally, we remark that applying standard black-box machine learning techniques for fiber parameter identification is not straight-forward. Due to their black-box nature, known physical models are not exploited. Instead, one has to provide large representative data sets that cover a wide range of real-world scenarios, which is a challenging problem in itself.

To overcome these drawbacks of current identification methods, we present a novel algorithm that identifies average values for both β_2 and γ, by comparing the nonlinear Fourier spectrum of transmitted and received signals. Theory predicts that the NFT spectrum at the transmitter and the NFT spectrum at the receiver are linearly related for a noiseless lossless link [9]. Since additional loss and noise effects occurring in realistic links can be taken into account using proper transformations, it is possible to use the NFT spectrum for identification.

The NFT is typically computed with respect to a normalized nonlinear Schrödinger equation (NLSE), which requires the provided input-output data to be normalized. Our algorithm uses the fact that the NFT spectrum at input and output only match if the input and output signal are normalized with the correct amplitude normalization constant c_q and normalized fiber length Z, which both depend on β_2 and γ. First, an initial estimate for c_q is determined by comparing the third conserved quantities of the NLSE of the input and output signals [7]. Starting from this initial guess, c_q is varied first and then Z until we find the normalization at which the NFT spectrum at input and output match optimally. Assuming that the attenuation, fiber length, and amplifier spacing are known, we can derive β_2 and γ from the identified normalization c_q and Z.

Another application of our proposed algorithm is the identification of a suitable model for NFT-based communication systems [23], without any prior knowledge of the fiber. NFT-based transmission systems typically only require the correct normalization constant c_q and normalized length Z, and can therefore be calibrated, even if no information of the fiber link is available at all.

This chapter is structured as follows. Sec. 3.2 gives an overview of the fiber model and the nonlinear Fourier transform. Sec. 3.3 provides two identification algorithms, based

on the continuous NFT spectrum and the discrete NFT spectrum, respectively. Sec. 3.4 combines both algorithms into one final robust algorithm. Sec. 3.5 evaluates the final algorithm with simulated test cases and Sec. 3.6 concludes the chapter.

3.2. FIBER MODEL AND NONLINEAR FOURIER TRANSFORM

The propagation of light through an optical single-mode fiber under the influence of anomalous dispersion, self-focusing, attenuation, lumped amplification, and noise can be modeled by the focusing nonlinear Schrödinger equation [62, Ch. 9.1.1]:

$$A_l = -i\frac{\beta_2}{2}A_{\tau\tau} + i\gamma|A|^2A - \frac{\alpha}{2}A + \sum_{n=1}^{N}(rA + G(r,n))\delta(l - nL_{\text{span}}), \quad (3.1)$$

in which τ denotes retarded time, l the position in the fiber, $A(\tau, l)$ the complex field envelope, $\beta_2 < 0$ the dispersion coefficient, γ the Kerr nonlinearity coefficient, α the attenuation coefficient, i the unit imaginary number, and δ the Dirac delta function. Subscripts indicate partial derivatives. The link of length L consists of N equidistant fiber spans, each with length $L_{\text{span}} = L/N$. At the end of each span, $l = nL_{\text{span}}$, an Erbium Doped Fiber Amplifier (EDFA) is used to amplify the signal with a factor $r = e^{\alpha L_{\text{span}}/2}$ (lumped amplification) to compensate the loss. Additive white Gaussian noise $G(r, n)$ enters the system through Amplified Spontaneous Emission (ASE) during amplification, with noise power dependent on the noise figure of the EDFA and the amplification factor r.

Within each fiber span, the amplifiers can be ignored, and we may change to the variable $Q(\tau, l) = e^{\alpha l/2}A(\tau, l)$, which yields a lossless NLSE with varying Kerr effect:

$$Q_l = -i\frac{\beta_2}{2}Q_{\tau\tau} + i\gamma e^{-\alpha l}|Q|^2Q. \quad (3.2)$$

Assuming that the wave envelope does not change much within each fiber span, we may approximate Eq. 3.2 with a lossless path-averaged (LPA) NLSE [75], in which the varying nonlinearity coefficient is approximated with its path average, γ_1. This leads to the LPA-NLSE:

$$\gamma_1 = \frac{1}{L_{\text{span}}}\int_0^{L_{\text{span}}}\gamma e^{-\alpha l}\,\mathrm{d}l = \gamma\frac{1 - e^{-\alpha L_{\text{span}}}}{\alpha L_{\text{span}}}, \quad (3.3)$$

$$Q_l \approx -i\frac{\beta_2}{2}Q_{\tau\tau} + i\gamma_1|Q|^2Q. \quad (3.4)$$

When distributed Raman amplification is applied instead of lumped amplification, path averaging can also be applied using an integral with l-dependent attenuation. A link with distributed Raman amplification is typically approximated better by the LPA-NLSE than a similar link with lumped amplification [76]. In this study, we therefore focus on the more challenging case of lumped amplification.

The proposed fiber identification algorithm in this study is based on comparing the NFT of input and output signals, for which the NLSE is required in normalized form. Let T_0 be a free time scaling parameter. We will only consider $T_0 = 1\,\mathrm{s}$ for the identification algorithm, as increasing T_0 only linearly scales the nonlinear frequency λ in the NFT. We then switch to the normalized variables [28]

$$t = \frac{1}{T_0}\tau, \quad q = T_0 \underbrace{\sqrt{\frac{\gamma_1}{-\beta_2}}}_{c_q} Q, \quad z = \frac{1}{T_0^2} \underbrace{\frac{-\beta_2}{2}}_{c_z} l, \tag{3.5}$$

This results in the normalized NLSE:

$$q_z = i q_{tt} + 2i|q|^2 q. \tag{3.6}$$

Note that β_2 appears in the denominator of c_q. Small mismatches in β_2 might therefore result in large changes in the normalized signal q. Hence, the proposed NFT-based identification algorithm may be less suited for dispersion managed links with near-zero average dispersion.

The normalized NLSE may be solved exactly in the nonlinear Fourier domain, in which the NFT spectrum of the signal evolves trivially. The NFT of a signal $q(t)$ can be determined by solving the Zhakarov-Shabat scattering problem [7, 9]:

$$\begin{aligned} \frac{d}{dt}\begin{bmatrix}\phi_1(t,\lambda)\\ \phi_2(t,\lambda)\end{bmatrix} &= \begin{bmatrix}-i\lambda & q(t)\\ -q^*(t) & i\lambda\end{bmatrix}\begin{bmatrix}\phi_1(t,\lambda)\\ \phi_2(t,\lambda)\end{bmatrix}, \\ \begin{bmatrix}\phi_1(t,\lambda)\\ \phi_2(t,\lambda)\end{bmatrix} &\overset{t\to-\infty}{\longrightarrow} \begin{bmatrix}e^{-i\lambda t}\\ 0\end{bmatrix}, \end{aligned} \tag{3.7}$$

where $(\cdot)^*$ denotes the complex conjugate. We then define the scattering coefficients $a(\lambda)$ and $b(\lambda)$ as the normalized limits of ϕ_1 and ϕ_2 for $t \to +\infty$:

$$a(\lambda) := \lim_{t\to\infty} e^{i\lambda t}\phi_1(t,\lambda), \quad b(\lambda) := \lim_{t\to\infty} e^{-i\lambda t}\phi_2(t,\lambda). \tag{3.8}$$

The NFT of $q(t)$ consists of a continuous and a discrete spectrum. We define the continuous spectrum as $b(\lambda)$, $\lambda \in \mathbb{R}$, and the discrete spectrum as $\{\lambda_m, b_m\}_{m=1}^{M}$, in which the eigenvalues λ_m are the zeros of $a(\lambda)$ in the complex upper half plane, and $b_m = b(\lambda_m)$. Each eigenvalue λ_m in the discrete spectrum corresponds to a *solitonic component* in the signal, a shape-retaining, localized wave. The continuous spectrum represents dispersive components. As the signal $q(t,z)$ propagates in the z-direction according to the normalized NLSE (3.6), the eigenvalues remain invariant, while the scattering coefficients evolve trivially [9]:

$$a(\lambda,z) = a(\lambda,0), \quad b(\lambda,z) = b(\lambda,0)e^{4i\lambda^2 z}. \tag{3.9}$$

3.3. IDENTIFICATION USING ONLY CONTINUOUS OR DISCRETE SPECTRUM

In this section, we present two separately executable algorithms to estimate the fiber coefficients, which are combined into one robust, final algorithm in Section 3.4. Both algorithms are based on the simple evolution of the scattering coefficients in Eq. 3.9. The first

algorithm takes only the continuous spectrum into account; the second algorithm considers only the discrete spectrum. Both algorithms can be split into two separate single-parameter identification problems: first, the amplitude normalization $c_q = \sqrt{\gamma_1/(-\beta_2)}$ is identified, and second, the normalized link length $Z := c_z L = -\frac{\beta_2}{2} L$. The amplitude normalization will be determined using a local optimization technique, starting from an intitial guess. The third integral of motion of the normalized NLSE can be used to obtain an initial estimate, c_q^0, as described in the Appendix. If other prior knowledge is available of the fiber, this may also be taken into account for the initialization.

For the proposed algorithm, we have chosen to define the NFT spectrum as $b(\lambda)$, although it is also common to define the continuous spectrum as $\hat{q}(\lambda) := b/a$, $\lambda \in \mathbb{R}$, and the discrete spectrum as $\{\lambda_m, \tilde{q}_m\}_{m=1}^M$, with $\tilde{q}_m := b/a_\lambda(\lambda_m)$ (i.e., the residue of $\hat{q}$ in λ_m). We have chosen to use only $b(\lambda)$, as it is usually less noisy than $b(\lambda)/a(\lambda)$, (see e.g., [77]).

3.3.1. IDENTIFICATION FROM THE CONTINUOUS NFT SPECTRUM

The first identification algorithm considers only the continuous part of the NFT, $b(\lambda)$, $\lambda \in \mathbb{R}$. Note that the nonlinear Fourier transform of a signal depends on the applied amplitude normalization, c_q, in Eq. 3.5. We denote the b-coefficient in (3.8) that corresponds to the signal $q(t) = T_0 c_q Q(T_0 t)$ in (3.7) by $b(\lambda, z; c_q)$, where z denotes the normalized position of Q in the fiber.

We observe from Eq. 3.9 that the correct value $c_q = c_q^\star$ leads to a constant absolute value of $|b|$ throughout an ideal fiber, and in particular, the absolute value at input ($z = 0$) and at output ($z = Z/T_0^2$) are equal:

$$|b(\lambda, Z/T_0^2; c_q^\star)| = |b(\lambda, 0; c_q^\star)|, \quad \lambda \in \mathbb{R}. \tag{3.10}$$

We may substitute $b^{\text{in}}(\lambda; c_q) = b(\lambda, 0; c_q)$ and $b^{\text{out}}(\lambda; c_q) = b(\lambda, Z/T_0^2; c_q)$, which are determined from the NFT of the normalized transmitted signal $q^{\text{in}}(t)$ and received signal $q^{\text{out}}(t)$ respectively. Our strategy will be to vary c_q, and identify the normalization for which the $|b|$ of the transmitted and received signal match as well as possible. As shown in Fig. 3.1, the absolute continuous spectrum at input and output are indeed nearly identical for the optimal amplitude normalization, $c_q^\star$, whereas a sub-optimal c_q may cause a significant mismatch.

To quantify the mismatch, we measure the normalized absolute error over a range of nonlinear frequencies $[\lambda_{\min}, \lambda_{\max}]$:

$$E_{|\text{cs}|}(c_q) = \frac{\int_{\lambda_{\min}}^{\lambda_{\max}} \big| |b^{\text{out}}(\lambda; c_q)| - |b^{\text{in}}(\lambda; c_q)| \big| \, \mathrm{d}\lambda}{\int_{\lambda_{\min}}^{\lambda_{\max}} |b^{\text{in}}(\lambda; c_q)| \, \mathrm{d}\lambda}, \tag{3.11}$$

in which $E_{|\text{cs}|}$ denotes the relative error in the absolute value of the continuous spectrum. The error is normalized with the b-coefficient of the transmitted signal, as it does not contain any ASE-noise, in contrast to the received signal. The range $[\lambda_{\min}, \lambda_{\max}]$ is determined by considering the occupied bandwidth, i.e., the smallest *linear* frequency

range containing at least (e.g.) 90% of the signal energy:

$$[\lambda_{\min}, \lambda_{\max}] = [-\omega_{\max}/2, -\omega_{\min}/2], \quad \text{such that}$$
$$\int_{\omega_{\min}}^{\omega_{\max}} \left|\mathscr{F}(q)(\omega)\right|^2 \mathrm{d}\omega \geq 0.9 \int_{-\infty}^{+\infty} \left|\mathscr{F}(q)(\omega)\right|^2 \mathrm{d}\omega, \tag{3.12}$$

with $\mathscr{F}(q)(\omega)$ the linear Fourier transform of $q(t)$ at angular frequency ω. The reason for this choice is that the NFT of an infinitesimal-energy signal $q(t)$ and the linear Fourier transform of $q(t)$ relate through [28]

$$b(\lambda) = -\left(\mathscr{F}(q)(-2\lambda)\right)^*. \tag{3.13}$$

Also for higher-energy signals, (i.e., when the linear and nonlinear Fourier transform differ significantly), we observed that the frequency range $[\lambda_{\min}, \lambda_{\max}]$ still contains a significant amount of the energy, leading to a good signal-to-noise ratio for the identification algorithm.

We minimize (3.11) with respect to c_q by first performing a coarse line search around the initial guess, c_q^0. The grid point with minimal error is then used as initial guess for the local minimization scheme from [78] (i.e., the standard *fminsearch* function in *Matlab*) to obtain an estimate $\hat{c}_q$.

COMPARING PHASE SHIFTS TO FIND Z

After identifying c_q from $|b(\lambda)|$, we may extract the normalized length Z by comparing the phase shift between the continuous spectrum at receiver and transmitter. From this point on, we consider only $b(\lambda, z) = b(\lambda, z; \hat{c}_q)$ with $\hat{c}_q$ identified in the previous step. Assuming propagation through an ideal fiber with $\hat{c}_q = c_q^\star$ and the correct normalized length, $Z^\star$, the phase shift in $b(\lambda, z)$ according to (3.9) is given by

$$\angle b(\lambda, Z^\star/T_0^2) = \angle b(\lambda, 0) + 4\lambda^2 Z^\star/T_0^2 \mod 2\pi, \quad \lambda \in \mathbb{R}. \tag{3.14}$$

By replacing $b(\lambda, 0)$ with the determined $b^{\text{in}}(\lambda)$, and $b(\lambda, 0)$ with $b^{\text{out}}(\lambda)$, Eq. 3.14 will hold approximately if $\hat{c}_q \approx c_q^\star$ and $Z \approx Z^\star$. Note that Eq. 3.14 corresponds to fitting a parabola λ^2 to the phase shift in b, in a 2π-periodic space. When identifying Z, we will also allow an offset ψ_0 in the parabola.

We first estimate Z and ψ_0 by fitting $4\lambda^2 Z/T_0^2 + \psi_0$ with linear least squares to the unwrapped phase difference, $\Delta\psi(\lambda) := \text{unwrap}(\angle b^{\text{out}} - \angle b^{\text{in}})$, with $\Delta\psi(0) \in [-\pi, \pi)$, and $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. Although the resulting estimates are usually accurate, it may occur that the phase difference is unwrapped to the wrong side, causing 2π jumps in the unwrapped phase difference (see Fig. 3.2, left). Therefore, we define an error norm that is not affected by these 2π jumps:

$$E_{\angle\text{cs}}(Z, \psi_0) = \frac{1}{\lambda_{\max} - \lambda_{\min}} \int_{\lambda_{\min}}^{\lambda_{\max}} \Big| \big[\angle b^{\text{out}}(\lambda) - \angle b^{\text{in}}(\lambda) - \left(4\lambda^2 Z/T_0^2 + \psi_0\right) + \pi\big] \mod 2\pi - \pi \Big| \mathrm{d}\lambda, \tag{3.15}$$

in which $E_{\angle\text{cs}}$ denotes the average error in the phase of the continuous spectrum. Note that we add π before the modulo operation, and subtract π afterwards to ensure the

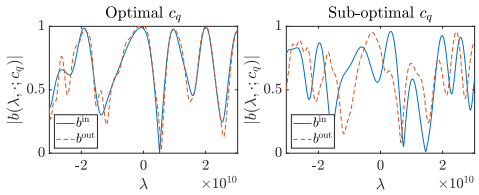


Figure 3.1: The absolute continuous spectrum at input and output at optimal c_q (left), and at sub-optimal c_q (right).

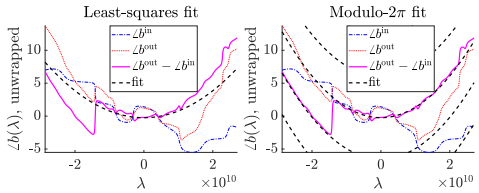


Figure 3.2: Left: the initial least squares quadratic fit to the unwrapped phase difference. Right: the quadratic fit to the unwrapped phase difference resulting from (3.15), which allows for 2π phase shifts.

phase mismatch at every λ is in the region $[-\pi,\pi)$ instead of $[0,2\pi)$. Similar to the optimization of $E_{|\text{cs}|}$ (3.11), we first perform a coarse line search around the linear least squares estimate for Z, and then the iterative minimization scheme from [78] to obtain final estimates $\hat{Z}$ and $\hat{\phi}_0$.

3.3.2. IDENTIFICATION FROM THE DISCRETE NFT SPECTRUM

In this subsection, we propose a method to identify the normalization c_q and normalized length Z by comparing the discrete spectra of the transmitted and received signal. Similar to the continuous spectrum algorithm, we will first identify c_q, and Z afterwards.

We start by writing the discrete spectrum with an explicit dependence on c_q. We denote the discrete spectrum that corresponds to the signal $q(t) = T_0 c_q Q(T_0 t)$ by $\{\lambda_m(c_q), b_m(z;c_q)\}_{m=1}^{M(c_q)}$, with z the normalized position of Q in the fiber. We note here that the energy $\mathscr{E}_m$ of the solitonic component corresponding to the eigenvalue λ_m increases with its imaginary part, $\mathscr{E}_m = 4\Im(\lambda_m)$ [28]. Increasing c_q increases the normalized signal energy, which usually increases energy in the discrete spectrum as well. As a result, on average the eigenvalues meander upwards in the complex plane in continuous trajectories, and new eigenvalues may appear from the real axis [79],[80, p. 12]. For small values of c_q however, the normalized energy will be too low, and the discrete spectrum will contain no eigenvalues at all [81].

If the correct normalization $c_q^\star$ is applied, the eigenvalues λ_m remain constant, and each $b(\lambda_m)$ evolves according to (3.9), assuming a lossless, noiseless fiber. Also for a non-ideal fiber, the solitonic components of the eigenvalues are still preserved relatively well, as long as both the span length is much shorter than the corresponding soliton period [82] and the signal-to-noise ratio is sufficiently high. We will use this property,

and identify the normalization c_q for which the discrete spectrum at transmitter and at receiver correspond at well as possible. To illustrate the effect of the normalization c_q on the discrete eigenvalues, Fig. 3.3 shows the spectrum of a 7-soliton signal [83] at both transmitter and receiver. The left eigenvalues were determined with the optimal $c_q^\star$. The right spectra were determined with a 22% larger c_q, which caused a larger difference between the spectra compared to the optimal c_q. We also note that the increase in c_q resulted in more normalized energy, and accordingly, the eigenvalues have shifted upwards, and a new, unmatched output eigenvalue has spawned from the real axis.

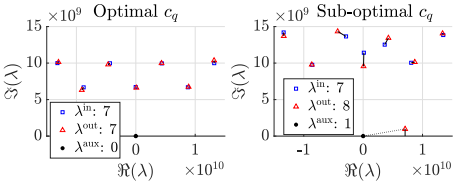


Figure 3.3: The eigenvalues of a transmitted 7-soliton signal, evaluated with the optimal normalization c_q (left), and with a sub-optimal c_q (right). Connections indicate the least-cost perfect matching.

To quantify the error between the input and output spectrum for a given c_q, we propose an error norm based on creating pairs of eigenvalues at input and output, and sum the error in each pair. Note that the number of input eigenvalues, M^{in}, and the number of output eigenvalues, M^{out}, may be unequal, $M^{\text{in}} \neq M^{\text{out}}$. To allow for a perfect matching, we keep adding 'auxiliary eigenvalues' at zero, $\lambda^{\text{aux}} = 0$, to the smaller set until the sets are equally large (see also Fig. 3.3, right). Next, we create a complete bipartite graph as shown in Fig. 3.4. Each input eigenvalue λ_k^{in} connects to each output eigenvalue λ_l^{out} with edges $(\lambda_k^{\text{in}}, \lambda_l^{\text{out}})$, with associated cost E_{kl}. The cost of the edges represents the mismatch between eigenvalue λ_k^{in} and λ_l^{out}, and will be specified later in this section, but we already mention that each edge cost E_{kl} is always at most $\Im\left(\lambda_k^{\text{in}}\right) + \Im\left(\lambda_l^{\text{out}}\right)$, i.e., proportional to their combined energy.

We define the total error by finding a least-cost perfect matching of the bipartite graph with edge costs E_{kl}. Let $l(k)$ denote a perfect matching, which assigns each input eigenvalue λ_k^{in} to the output eigenvalue $\lambda_{l(k)}^{\text{out}}$ (see Fig. 3.3 and Fig. 3.4). The least-cost matching may be found in $\mathcal{O}(M^3)$ time, $M = \max(M^{\text{in}}, M^{\text{out}})$, for example with fast versions of the Hungarian algorithm [84]. The final error is the cost of the least-cost perfect matching, normalized by the sum of imaginary parts of all eigenvalues at input and output:

$$E_{\text{ds}}(c_q) = \frac{\min_{l(k)} \sum_{k=1}^{M} E_{kl(k)}}{\sum_{k=1}^{M} \Im\left(\lambda_k^{\text{in}}\right) + \sum_{l=1}^{M} \Im\left(\lambda_l^{\text{out}}\right)}, \tag{3.16}$$

in which M, E_{kl}, and λ_m depend implicitly on c_q. In case no eigenvalues were found for both the input and output, we set E_{ds} to its maximum value, 1. We find an estimate $\tilde{c}_q$ as the value minimizing $E_{\text{ds}}(c_q)$.

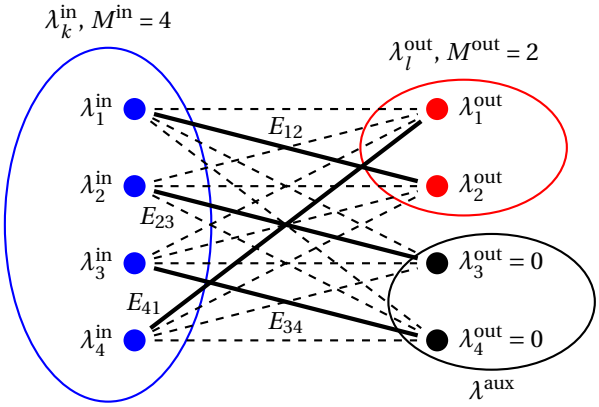


Figure 3.4: The complete bipartite graph corresponding to 4 input eigenvalues, and 2 output eigenvalues. 2 auxiliary eigenvalues have been added to the set of output eigenvalues to ensure that the output and input eigenvalue sets are equally large. A possible perfect matching $l(k)$ is marked: $\{l(1) = 2, l(2) = 3, l(3) = 4, l(4) = 1\}$. The cost of this matching is $E_{12} + E_{23} + E_{34} + E_{41}$.

EDGE COST E_{kl}

Next, we consider the edge cost E_{kl} to represent the mismatch E_{kl} between λ_k^{in} and λ_l^{out}. A straightforward, but effective edge cost E_{kl} is the Euclidean distance between input and output eigenvalue. However, with this norm, eigenvalue pairs with with small imaginary part, but large difference in real part may dominate the error, whereas their energy ($\propto$ imaginary part) can be arbitrarily small. To ensure that eigenvalue pairs with low energy cannot dominate the total error, we put an upper bound on E_{kl}, equal to the sum of the imaginary parts of the eigenvalue pair:

$$E_{kl} = \min\left(\left|\lambda_k^{\text{in}} - \lambda_l^{\text{out}}\right|, \Im\left(\lambda_k^{\text{in}}\right) + \Im\left(\lambda_l^{\text{out}}\right)\right). \tag{3.17}$$

This way, if λ_k^{in} and λ_l^{out} contain little energy, they can also contribute little to the total error. A mathematical motivation for this upper bound for E_{kl} is that new eigenvalues may appear at the real axis as c_q is varied. Therefore, we can argue that instead of connecting an input and output eigenvalue to each other, it may also be the case that each connects to a hypothetical eigenvalue which is about to appear on the real axis right beneath it, resulting in an error of $\Im\left(\lambda_k^{\text{in}}\right) + \Im\left(\lambda_l^{\text{out}}\right)$. Note that the connection between an eigenvalue λ_m and an auxiliary eigenvalues $\lambda^{\text{aux}} = 0$ always assumes its maximum bound, $\Im(\lambda_m)$.

Although the edge cost in Eq. 3.17 suffices in most circumstances, an alternative definition for E_{kl} may be used when we already have a reliable estimate for Z available. Z may be available when β_2 and L are already known, and only γ_1 is to be identified, or when Z was reliably estimated by the continuous spectrum algorithm. If Z is available, we may define the error E_{kl} using both a mismatch in eigenvalue, E_{kl}^{λ}, as well as a mismatch in spectral function, E_{kl}^{b}, (opposed to only using the mismatch in λ as in (3.17)). Let the mismatch in eigenvalue be given by the Euclidean distance between the paired eigenvalues,

$$E_{kl}^{\lambda} := |\lambda_k^{\text{in}} - \lambda_l^{\text{out}}|. \tag{3.18}$$

Second, we consider the mismatch in the spectral function b. According to Eq. 3.9, $b(\lambda, Z/T_0^2) = b(\lambda,0)e^{4i\lambda^2 Z/T_0^2}$. Let $\lambda_{kl} := \frac{\lambda_k^{\text{in}}+\lambda_l^{\text{out}}}{2}$ be the average of the input and output eigenvalue. We may express the difference between b_k^{in} and $b_l^{\text{out}}e^{-4i\lambda_{kl}^2 Z/T_0^2}$ as the result of a difference $\Delta\lambda$ in λ_{kl}:

$$\begin{aligned} b_k^{\text{in}} &= b_l^{\text{out}} e^{-4i(\lambda_{kl}+\Delta\lambda)^2 Z/T_0^2}, \\ \Rightarrow b_k^{\text{in}} &= b_l^{\text{out}} e^{-4i\lambda_{kl}Z/T_0^2 - 8i\lambda_{kl}\Delta\lambda Z/T_0^2 + \mathcal{O}(\Delta\lambda^2)}. \end{aligned}$$

By dropping the quadratic term $\mathcal{O}(\Delta\lambda^2)$, we get a simple expression for the mismatch $\Delta\lambda$ in the spectral data:

$$|\Delta\lambda| \approx E_{kl}^b := \left| \frac{\log\left(b_l^{\text{out}} e^{-4i\lambda_{kl}Z/T_0^2}/b_k^{\text{in}}\right)}{8i\lambda_{kl}Z/T_0^2} \right|, \tag{3.19}$$

and note that this relationship requires knowledge of Z. We finally define the total mismatch as the average of both errors, and bound it by the imaginary value of the eigenvalue pair:

$$E_{kl} := \min\left(\frac{E_{kl}^\lambda + E_{kl}^b}{2}, \Im\left(\lambda_k^{\text{in}}\right) + \Im\left(\lambda_l^{\text{out}}\right) \right). \tag{3.20}$$

When a reliable estimate for Z is available, using edge cost (3.20) generally results in better estimates $\tilde{c}_q$ compared to using (3.17).

DETERMINE Z FROM $|b(\lambda_m)|$

We can (re-)estimate Z by comparing the $|b^{\text{in}}(\lambda_k)|$ with $|b^{\text{out}}(\lambda_{l(k)})|$ using the identified $\tilde{c}_q$ and assignment $l(k)$. The spectral function at input and output for λ_m are related through Eq. 3.9, from which estimates $\tilde{Z}_k$ may be obtained as

$$\begin{aligned} |b(\lambda_m, Z/T_0^2)| &= |b(\lambda_m,0)|e^{\Re(4i\lambda_m^2)Z/T_0^2} \\ \Rightarrow \tilde{Z}_k &= T_0^2 \frac{\log\left|b^{\text{out}}\left(\lambda_{l(k)}\right)\right| - \log\left|b^{\text{in}}\left(\lambda_k\right)\right|}{\Re\left(4i\lambda_{kl}^2\right)}, \end{aligned} \tag{3.21}$$

where $\tilde{Z}_k$ is an estimate for Z from the k^{th} eigenvalue pair, and $\lambda_{kl} := \frac{\lambda_k^{\text{in}}+\lambda_{l(k)}^{\text{out}}}{2}$ is the average of the paired input and output eigenvalue. To use the available data as well as possible, we consider all $\tilde{Z}_k$ for a final estimate $\tilde{Z}$. First, all pairs with maximal distance, $E_{kl} = \Im\left(\lambda_k^{\text{in}}\right) + \Im\left(\lambda_l^{\text{out}}\right)$, are discarded, as these eigenvalues are very distant from each other and are unlikely to be related. Second, we discard outliers, i.e., estimations more than two standard deviations away from the mean. Outliers are common, as almost purely imaginary eigenvalues lead to a small denominator in (3.21), yielding unstable estimations for Z. Third, as eigenvalues with large imaginary part contain more energy, we assign their associated estimates a larger weight, $w_k = \Im(\lambda_{kl})$. The final $\tilde{Z}$ is the weighted average of the remaining estimates, $\tilde{Z} = \sum_k w_k \tilde{Z}_k / \sum_k w_k$.

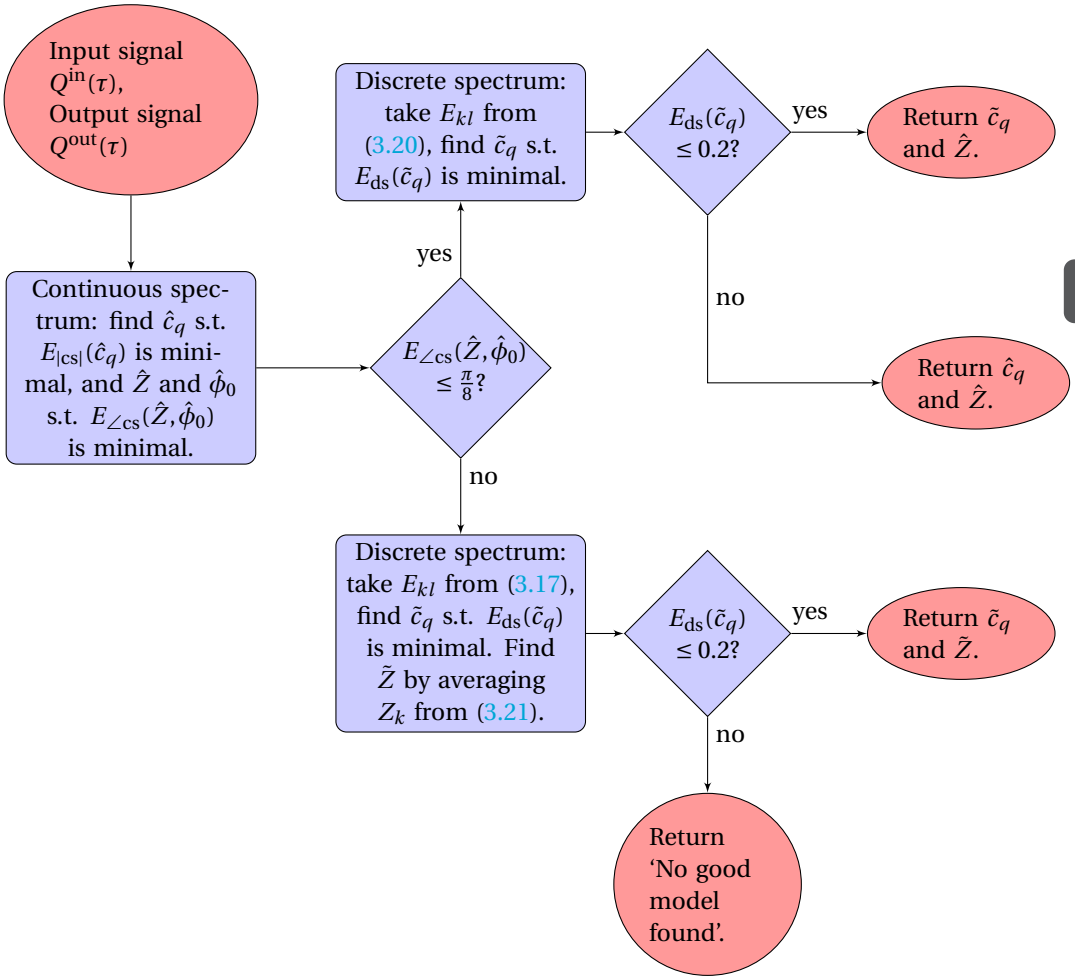


Figure 3.5: Overview of the final algorithm.

3.4. FINAL IDENTIFICATION ALGORITHM

In this section, we combine all previously described algorithms to create one final robust algorithm. From our experience with the continuous and discrete spectrum, we have found that the continuous spectrum algorithm usually yields better results for Z, while the discrete spectrum algorithm yields more reliable results for c_q. Therefore, we will first attempt to use the continuous spectrum to identify Z, and then the discrete spectrum to identify c_q. If Z is identified reliably from the continuous spectrum, this Z may also be used for the discrete spectrum identification through Eq. 3.18-3.20.

In general, it holds that if the continuous or discrete spectrum contains too little energy, or the signal-to-noise ratio is too low, no good resemblance between input and output spectrum exists, and thus the estimates of the corresponding algorithm should

be discarded. We therefore reject estimates for the continuous spectrum when $E_{\angle \mathrm{cs}} > \frac{\pi}{8}$ (average absolute phase mismatch) for the continuous spectrum algorithm, and $E_{\mathrm{ds}} > 0.2$ (relative error) for the discrete spectrum.

We propose a final algorithm as summarized in Fig. 3.5. First we find estimates $\hat{c}_q$ and $\hat{Z}$ using the continuous spectrum algorithm. If these estimates are accepted, we apply the discrete spectrum algorithm using the identified $\hat{Z}$ to re-estimate c_q, and accept this estimate $\tilde{c}_q$ if $E_{\mathrm{ds}} < 0.2$. Otherwise, it is discarded. If the continuous spectrum algorithm estimates were rejected, we fully rely on the discrete spectrum algorithm. If the discrete spectrum error is sufficiently small, $E_{\mathrm{ds}} < 0.2$, we accept the resulting estimates $\tilde{c}_q$ and $\tilde{Z}$. In case both the continuous and discrete algorithm did not provide reliable results, we can enlarge our search range for the initial guess c_q^0. If this does not yield any reliable results either, the algorithm cannot identify a fitting normalization, most likely because the LPA-NLSE does not model the fiber well enough, or because the considered signal is too noisy. A more reliable estimate can be obtained by running the identification algorithm for multiple signals, discarding outlier estimates, and averaging over the remaining ones.

3.5. RESULTS: IDENTIFICATION ON SIMULATED DATA

In this section, we demonstrate the capabilities of the identification algorithm on noisy transmission data simulated with Eq. 3.1. We have considered two applications: first, we identified the second-order dispersion β_2, and the Kerr nonlinearity coefficient γ using a conventional transceiver; second, without any prior knowledge, the algorithm was used to calibrate a nonlinear Fourier transform-based transmission system.

We applied the algorithm on input-output data from a fiber-optic transmission system, simulated with the software *NFDMlab* [85]. For both applications, we considered the same NZ-DSF fiber link, with 20 spans of $L_{\mathrm{span}} = 80\,\mathrm{km}$ each, resulting in a total transmission length of 1600 km. The second-order dispersion of the fiber was $\beta_2 = -5.00 \cdot 10^{-27}\,\mathrm{s^2 m^{-1}}$, the Kerr nonlinearity coefficient was $\gamma = 1.20 \cdot 10^{-3}\,(\mathrm{Wm})^{-1}$, and the attenuation coefficient was $\alpha = 0.2 \cdot 10^{-3}\,\mathrm{dB/m}$. After each span, an EDFA compensated the accumulated loss through lumped amplification, and added white Gaussian noise with a noise figure of $f_n = 6$ dB. We used the LPA-NLSE (3.4) as reference solution, which predicted a path-averaged (PA) model with path-averaged Kerr nonlinearity coefficient $\gamma_1 = 0.318 \cdot 10^{-3}\,(\mathrm{Wm})^{-1}$, normalization coefficient $c_q = 2.52 \cdot 10^{11}\,\mathrm{W^{-1/2} s^{-1}}$, and normalized length $Z = 4.00 \cdot 10^{-23}\,\mathrm{s^2}$. At the link input and output, ideal low-pass filters were applied to account for transceiver bandwidth limitations [85]. We assume that all other real-world effects not included in the model are either zero-mean and can be included in the noise term, or that they have been removed by appropriate post-processing. For example, a carrier frequency offset can be detected and compensated with a simple nonlinear-frequency shift [28]. We finally assume a coherent receiver since both phase and amplitude information are required. To compute the NFT in the identification algorithm, we used the *Fast Nonlinear Fourier Transform* (FNFT) software library [30].

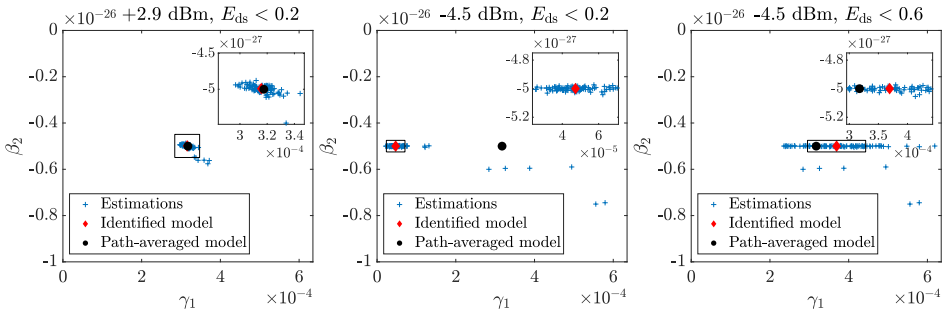


Figure 3.6: The identified path-averaged Kerr nonlinearity and second-order dispersion from conventional transceiver data. Left: high launch power (2.9 dBm) and unadapted algorithm. Middle: medium launch power (-4.5 dBm) and the unadapted algorithm (middle). Right: medium launch power (-4.5 dBm), increased acceptance rate from $E_{ds} < 0.2$ to $E_{ds} < 0.2$ for the discrete algorithm part.

3.5.1. IDENTIFYING FIBER COEFFICIENTS FROM A CONVENTIONAL TRANSCEIVER

In this benchmark, we simulated data transmission using a conventional time domain transceiver, and show that the fiber coefficients β_2 and γ can be accurately recovered. We assume that the fiber length, amplifier spacing, and attenuation coefficient are known. The transmitted signal consists of a linear sum of time-shifted raised cosine pulses, modulated with Quadrature Phase Shift Keying (QPSK). The roll-off factor was 0.5 and symbol duration $T = 0.04$ ns. The time shift between two subsequent pulses is also $T = 0.04$ ns, to ensure zero intersymbol interference (ISI) in the transmitted signal. The transmitted signal consisted of burst of 128 pulses. In between two bursts, a guard interval of $16T$ was added. We considered two cases: one with optimal launch power, -4.5 dBm, and one with high launch power, 2.9 dBm. The optimal launch power was found by optimizing detection performance after applying digital (linear) dispersion compensation and average rotation compensation of the symbols. Note that the NFT spectrum of this signal was not specially tailored, we only spaced the bursts sufficiently far apart with guard intervals such that the bursts did not significantly interfere at output, and could be analysed separately.

For both cases, we transmitted a signal with 100 bursts, and applied our algorithm to each burst at transmitter and receiver. The identified coefficients were determined by first discarding outliers (more than two standard deviations away from the mean), and then averaging over the remaining estimations. The path averaged coefficients were used as reference solution. First, we consider the result for the high power signal, which are shown on the left in Fig. 3.6. The identified coefficients were $\beta_2 = -4.99 \cdot 10^{-27}\,\mathrm{s^2 m^{-1}}$ (PA: $-5.00 \cdot 10^{-27}$) and $\gamma_1 = 3.16 \cdot 10^{-4}\,(\mathrm{Wm})^{-1}$ (PA: $3.18 \cdot 10^{-4}$). The corresponding Kerr nonlinearity coefficient was $\gamma = 1.19 \cdot 10^{-3}\,(\mathrm{Wm})^{-1}$ (true: $1.20 \cdot 10^{-3}$). Both coefficients were less than 1% off the true values, showing that the fiber coefficients can be accurately identified using our algorithm if the launch power is sufficiently high.

Next, we consider the case with optimal launch power. The algorithm still reliably

identifies β_2, but the estimate of γ is poor. See the middle of Fig. 3.6. The poor performance of the algorithm with respect to γ is due to the fact that at optimal launch power (for a linear transceiver), nonlinear effects are insignificant compared to the dispersive effect over the considered fiber length. During further analysis, we observed that while the signals were actually dominated by solitonic components (about 80% of the total signal energy), the discrete spectrum was consistently rejected due to significant mismatch in $b(\lambda_j)$ in Eq. 3.19. (Similar observations in which conventional OFDM signals were found to be soliton-dominated have been reported in [86]. Note that the solitons do not disentangle since the fiber is too short.) The current estimates were thus solely based on the continuous spectrum. In order to get a good estimate of the nonlinearity parameter γ at optimal transmit power (for a linear transceiver), we should not discard the discrete spectrum. Hence, we increased the acceptance rates for the discrete estimates from 0.2 to 0.6 (this corresponds to accepting the discrete estimates in almost all cases). The result is shown on the right in Fig. 3.6. The new estimate is much better than before, although is it still biased and and the individual data points show a large variance. Thus, we were eventually able to obtain a reasonable estimate of the nonlinearity coefficient. However, the benchmark also shows that for highly accurate estimates of the nonlinearity parameter, the launch power has to be high enough such that nonlinear effects are not negligible.

3.5.2. CALIBRATING AN NFT-BASED TRANSCEIVER

In our second benchmark, we considered a scenario in which we modulated data in the discrete NFT-spectrum. If the correct normalization was used for the modulation, the transmitted signal would consist of bursts of multi-solitons with 7 eigenvalues λ_m and zero continuous spectrum. Bits were modulated into the phase of the residues, $\tilde{q}_m = b/a_\lambda(\lambda_m)$, of each eigenvalue using QPSK, identically following the modulation format of Bülow et al. [83]: $\angle\left(\tilde{q}(\lambda_m^{\text{in}})\right) \in \{-\frac{3\pi}{4}, -\frac{\pi}{4}, \frac{\pi}{4}, \frac{3\pi}{4}\}$. The time normalization constant was $T_0 = 2\,\text{ns}/(14\pi)$, which transformed the bursts with normalized duration 14π to bursts of physical duration 2 ns. We emphasize that it is no problem if the identification algorithm uses a different value for T_0 when normalizing the input and output data. (When data is modulated in the normalized nonlinear Fourier domain and then transformed into the physical time-domain, the choice of the normalization parameter influences the physical duration and amplitude of the fiber input and thus has to be chosen correctly. In contrast, for the identification algorithm, physical data is the starting point and the time normalization constant only influences in which normalized domain the nonlinear spectra are compared. The only important point is that the input and output data have to be normalized using the same constant.)

Only the correct normalization c_q and the normalized length Z are required for successful transmission, but these were guessed poorly as $c_q = 3.00 \cdot 10^{11}\,\text{W}^{-1/2}\text{s}^{-1}$ (PA: $2.52 \cdot 10^{11}$) and $Z = 8.00 \cdot 10^{-23}\,\text{s}^2$ (PA: $4.00 \cdot 10^{-23}$). As a result, the error vector magnitude (EVM) in the received symbols was 35 dB, and communication was unsuccessful. Although the symbols could not be recovered, we could use the transmitted and received signals to identify a new model. We applied the combined algorithm to each of 100 bursts of presumed 7-soliton signals, discarded outlier estimates (more that 2 standard deviations away from the mean), and averaged the remaining estimates to obtain $c_q =$

$2.54 \cdot 10^{11}\,\mathrm{W}^{-1/2}\mathrm{s}^{-1}$ and $Z = 3.85 \cdot 10^{-21}\,\mathrm{s}^2$, as shown in Fig. 3.7. Using this new normalization, we generated a signal modulated with new symbols, and the symbols were successfully received. The transmission with the identified model resulted in an EVM of −8.2 dB in the symbols, whereas the path-averaged model resulted in −8.6 dB. Thus the identified model performed comparable to the path-averaged model, only showing a minor increase in EVM.

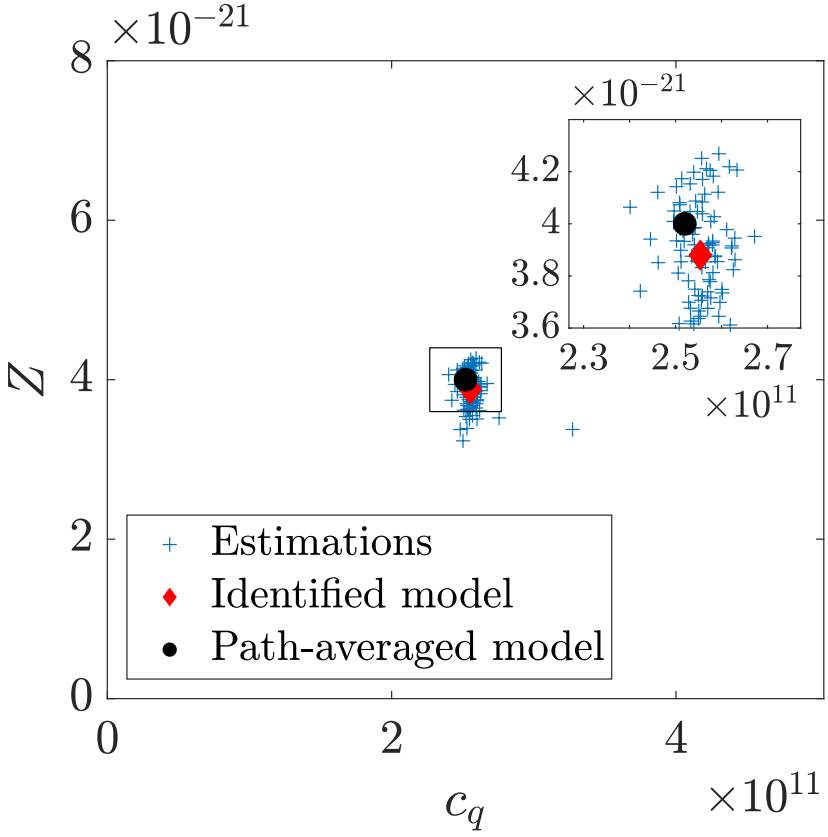


Figure 3.7: The identified normalization from multi-soliton transceiver data. No prior knowledge of the fiber was used.

3.6. CONCLUSION

We have proposed an algorithm to identify the second-order dispersion β_2 and Kerr nonlinearity coefficient γ of an optical fiber, based on the nonlinear Fourier transform (NFT) of transmitted and received signals. The algorithm models the fiber with a lossless, path-averaged nonlinear Schrödinger equation, and identifies a normalization for the NFT such that the corresponding NFT spectrum of the transmitted and received signal match optimally to each other. β_2 and γ are finally derived from the identified normalization,

using the known fiber length, amplifier spacing, and attenuation coefficient. Although β_2 and γ cannot be found without prior knowledge of the fiber, the normalization can nonetheless be identified, which already suffices to calibrate an NFT-based transceiver. The entire algorithm can be applied on any sufficiently high energy signal, and does not require special training signals.

We have demonstrated the capabilities of the algorithm with two benchmarks, in which the fiber coefficients were accurately identified. Due to its versatility and accuracy, the proposed identification algorithm may prove an attractive alternative to currently existing fiber identification methods.

APPENDIX: IDENTIFICATION FROM CONSERVED QUANTITIES

The normalized focusing NLSE (3.6) has an infinite number of conserved quantities [7], the first three of which are

$$C_1 = -\int_{-\infty}^{\infty} |q|^2 \,\mathrm{d}t, \tag{3.22}$$

$$C_2 = \frac{1}{2}\int_{-\infty}^{\infty} q q_t^* - q^* q_t \,\mathrm{d}t, \tag{3.23}$$

$$C_3 = \int_{-\infty}^{\infty} |q|^4 - |q_t|^2 \,\mathrm{d}t. \tag{3.24}$$

The value of these constants does not depend on the location in the fiber as long as the signal propagates according to the NLSE. Given an arbitrary input signal $q^{\mathrm{in}}(t) = T_0 c_q Q^{\mathrm{in}}(T_0 t)$ and its corresponding output signal $q^{\mathrm{out}}(t) = T_0 c_q Q^{\mathrm{out}}(T_0 t)$, we may determine c_q by comparing C_3 at input and output:

$$C_3^{\mathrm{in}}(c_q) = C_3^{\mathrm{out}}(c_q), \quad \Rightarrow$$

$$\int_{-\infty}^{\infty} |c_q Q^{\mathrm{in}}|^4 - |c_q Q_\tau^{\mathrm{in}}|^2 \,\mathrm{d}\tau = \int_{-\infty}^{\infty} |c_q Q^{\mathrm{out}}|^4 - |c_q Q_\tau^{\mathrm{out}}|^2 \,\mathrm{d}\tau,$$

$$\Rightarrow c_q^2 = \frac{\int_{-\infty}^{\infty} |Q_\tau^{\mathrm{in}}(\tau)|^2 - |Q_\tau^{\mathrm{out}}(\tau)|^2 \,\mathrm{d}\tau}{\int_{-\infty}^{\infty} |Q^{\mathrm{in}}(\tau)|^4 - |Q^{\mathrm{out}}(\tau)|^4 \,\mathrm{d}\tau}. \tag{3.25}$$

This relation provides a fast and easy method to obtain estimates for $c_q > 0$. We note that the first and second conserved quantities cannot be used in a similar fashion because c_q drops out when equating them. On the other hand, higher conserved quantities can be used, but contain higher derivatives and powers, which are increasingly sensitive to noise. Using C_3 through Eq. 3.25 is thus the most suitable to estimate c_q. Using optimal linear launch power (-4.5 dBm) and high-power (2.9 dBm) signals from the conventional transceiver, described in Sec. 3.6, we have determined c_q^2 from 100 bursts. The results are shown in Fig. 3.8. At −4.5 dBm launch power, the estimates are highly biased, and the resulting c_q is a factor 2.5 too high (i.e., c_q^2 about 6 times too high). At higher launch power, the estimates for c_q improve, but still contain a bias of about 5%. In general, we observed that changing the modulation format or the fiber itself can drastically influence the bias. Furthermore, the estimates are very sensitive to noise, considered bandwidth and interference with neighboring bursts. The estimates are unfortunately too

biased and unstable to immediately use as final estimates for c_q, but they still provide an order-of-magnitude indication, and can be used as initial estimates for our NFT-based identification algorithm.

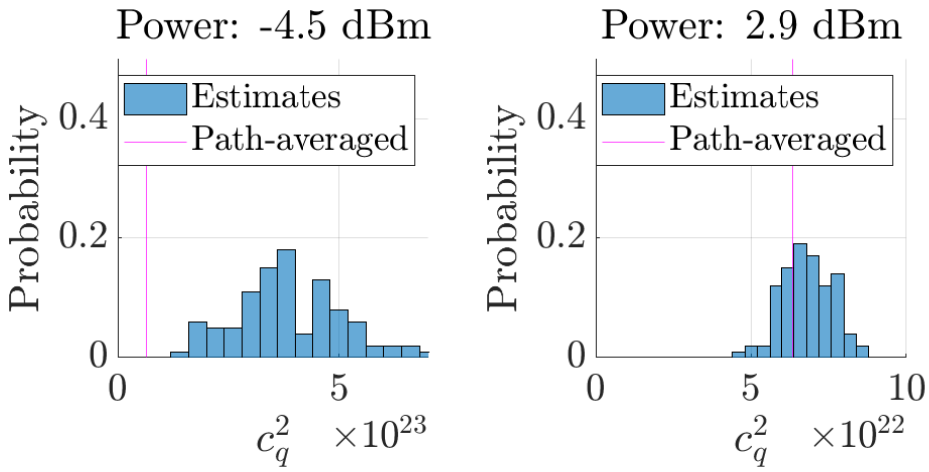


Figure 3.8: The distribution of c_q^2, estimated from the conserved quantity C_3 of a transmitted raised cosines signal at launch powers of −4.5 dBm and 2.9 dBm. Note the difference in scale between the plots.

4

FAST SINGLE-MODE FIBER NONLINEARITY MONITORING: AN EXPERIMENTAL COMPARISON BETWEEN SPLIT-STEP AND NONLINEAR FOURIER TRANSFORM-BASED METHODS

We experimentally investigate the problem of monitoring the Kerr-nonlinearity coefficient γ from transmitted and received data for a single-mode fiber link of 1600 km length. We compare the accuracy and speed of three different approaches. First, a standard split-step Fourier method is used to predict the output at various γ values, which are then compared to the measured output. Second, a recently proposed nonlinear Fourier transform (NFT)-based method, which matches solitonic eigenvalues in the transmitted and received signals for various γ values. Third, a novel fast version of the NFT-based method, which only matches the highest few eigenvalues. Although the NFT-based methods do not scale with link length, we demonstrate that the SSFM-based method is significantly faster than the basic NFT-based method for the considered link of 1600 km, and outperforms even the faster version. However, for a simulated link of 8000 km, the fast NFT-based method is shown to be faster than the SSMF-based method, although at the cost of a small loss in accuracy. This chapter is an adaptation of [87].

4.1. INTRODUCTION

The characterization of the Kerr nonlinearity coefficient of an installed optical fiber link is fundamental for fiber optical communications when employing higher launch powers,

for example for digital back-propagation (DBP) [62, 88–90], nonlinear frequency division multiplexing (NFDM) [23], or for performance estimation [91]. However, the nonlinearity coefficient is often not well known by the operator, so it is critical to measure the nonlinearity coefficient to optimally choose the transmission system parameters afterwards. Even when estimates of the nonlinearity coefficients are provided by the manufacturer, the coefficients may vary slightly between fibers, or change after installation due to aging, bending or splicing [62]. The re-characterization or even continuous monitoring of installed fibers is therefore often beneficial or even necessary. Monitoring of the nonlinearity coefficient can furthermore reveal changes of the link.

Practical communication systems today often avoid high powers at which the Kerr-nonlinear effect starts to distort the signal significantly. Therefore, the exact value of the Kerr-nonlinearity coefficient received little attention so far. However, as optical networks are being pushed towards higher bit-rates - often employing higher launch powers - estimating the Kerr-nonlinearity coefficient becomes increasingly important.

We are thus interested in methods to accurately and quickly determine the value of the Kerr-nonlinearity coefficient. Many links are in constant use. Furthermore, the transmission format is often difficult to change. Hence, it would be desirable to use already available regular transmission data to identify the Kerr-nonlinearity coefficient. This implies that the usual identification methods relying on specific training signals (e.g., using four-wave mixing [74], cross-phase modulation [92] or self-phase modulation [73]) are not well-suited for characterization of operational links. Instead, the Kerr-nonlinearity coefficient is typically identified by numerically simulating the propagation through the link with a split-step Fourier method (SSFM), for various nonlinearity coefficients. The nonlinearity coefficient at which the numerically forward-propagated input matches the measured output best is kept [68, 69, 71, 93]. However, the time required by such methods increases with the fiber length. They are therefore more time consuming for long links.

In earlier work, we proposed an alternative identification method which does not directly depend on the link length. This method compares solitonic components found using the nonlinear Fourier transform (NFT) of the transmitted and received signals [54, 55]. Solitons are stable waves that arise from a balance between dispersive and nonlinear effects. In an idealized lossless and noise-free link, the eigenvalues associated with the solitons are conserved exactly during propagation. Nevertheless, even in the presence of loss and noise the solitonic eigenvalues are still very stable [62]. As hidden solitons are present in many kinds of typical fiber-optical transmission data [55, 86], the NFT-based identification method is in principle widely applicable. The Kerr-nonlinearity coefficient is required to normalize the data before computing the NFT, and thus affects the detected solitons. The NFT-based identification method compares the solitonic components at the transmitter and receiver for different values of the Kerr-nonlinearity coefficient, and keeps the one at which the solitonic components match best. No numerical propagation is necessary. Thus the method does not depend on the link length, in contrast to propagation based-methods. We thus investigate if a speed-up can be achieved with respect to SSFM-based methods by instead comparing the signals in the NFT domain.

The NFT-based method has been investigated in simulations [54] and experiments [55,

94], where it has been found to provide similar estimates to split-step methods. However, it has not yet been tuned and assessed in terms of computational complexity, which is the goal of this study. We propose and investigate a faster version of the NFT-based nonlinearity characterization from [55], which only considers a few high-energy solitons. (While all nonlinear Fourier components are required to reconstruct a signal, one component in general is already sufficient to estimate the fiber parameters under ideal circumstances.) The solitonic eigenvalues are computed faster by using the old eigenvalues at an earlier γ estimate as initial guesses for the eigenvalues at the new γ estimate. This drastically speeds up the NFT-based matching, at the cost of some accuracy.

Finally, we quantitatively compare the SSFM-based identification method, the NFT-based method from [55] and the faster NFT-based method in terms of speed and accuracy. We first compare these three methods on well-separated pulses, such that neighboring pulses do not distort each other due to channel memory. Next, we compare the three methods on signal segments, that have been cut out from a full burst. As the channel memory distorts the received signal segment, we adapt the SSFM-based method slightly to take this into account. The NFT-based methods were less affected by the distortion due to channel memory and were therefore not adjusted.

The main contributions in this study are thus as follows:

- **The proposal of a novel fast version of the NFT-based nonlinearity identification algorithm.**
- **An experimental investigation of the accuracy of the SSFM-based method, the original NFT-based method and the novel NFT-based method.**
- **A run-time comparison of these three algorithms.**

This chapter is organized as follows. Section 4.2 describes the nonlinearity identification algorithm through the split-step Fourier method. Section 4.3 recapitulates how the NFT can be used to identify solitonic components in a signal, and to identify the nonlinearity coefficient. Section 4.4 describes the proposed faster NFT-based method which only compares the highest few eigenvalues. Section 4.5 compares the three identification methods on experimental measurement data for a 1600 km link. Section 4.6 compares the three identification methods on simulated data for an 8000 km link. Finally, Sec. 4.7 concludes the chapter.

4.2. SPLIT-STEP FOURIER METHOD FOR FIBER NONLINEARITY IDENTIFICATION

The propagation of an optical signal through a span of single-mode fiber (SMF) can be modeled by the lossy nonlinear Schrödinger equation (NLSE),

$$A_l = -i\frac{\beta_2}{2}A_{\tau\tau} + i\gamma|A|^2A - \frac{\alpha}{2}A, \tag{4.1}$$

in which A (in $\sqrt{\text{W}}$) is the complex signal amplitude, l (in m) the distance, τ (in s) the time, β_2 (in $\frac{\text{s}^2}{\text{m}}$) the second-order dispersion coefficient, α (in $\frac{1}{\text{m}}$) the loss coefficient,

and finally γ (in $\frac{1}{\mathrm{Wm}}$) the Kerr-nonlinearity coefficient to be identified in this study. Subscripts denote partial derivatives. At the end of each span an Erbium-doped fiber amplifier (EDFA) boosts the signal to a fixed launch power, resulting in a sawtooth-like power profile over the entire link.

In the first method considered in this study, the nonlinear coefficient γ is determined by numerically propagating the transmitted signal for various γ values, and keeping the γ value at which the forward propagated input and measured output match best. Here, we use the split-step Fourier method (SSFM) for the digital propagation. This method simulates the evolution of an optical signal by propagating the signal numerically in spatial steps of size Δl [95]. Each step is split up in a linear part and a nonlinear part. The linear part solves the effects of the linear terms in the NLSE analytically in the Fourier domain. Let $\tilde{A}(\omega, l) = \mathscr{F}\{A(\tau, l)\}$ be the Fourier transform of A, then the linear part of the NLSE is solved in the Fourier domain through

$$\tilde{A}^{\mathrm{lin}}(\omega, l + \Delta l) = \tilde{A}(\omega, l)\, e^{\left(i\frac{\beta_2}{2}\omega^2 - \frac{\alpha}{2}\right)\Delta l}. \tag{4.2}$$

Next, the nonlinear contribution is added by assuming that $|A(\tau, l)|^2$ is approximately constant during the spatial step, and the contribution is again added in the Fourier domain:

$$\tilde{A}(\omega, l + \Delta l) = \tilde{A}^{\mathrm{lin}}(\omega, l + \Delta l)\, e^{i\gamma\mathscr{F}\{|A^{\mathrm{lin}}(\omega, l+\Delta l)|^2\}\Delta l}. \tag{4.3}$$

At the end of each fiber span, the signal is numerically amplified to model the EDFA loss compensation. However, no amplified spontaneous emission (ASE) noise is added in the simulation, since we consider an idealized propagation.

In order to determine the best value of γ, we consider the relative L_1-error between the measured output and the numerically propagated input:

$$E(\gamma) = \frac{\int_T (|A^{\mathrm{out}}(\tau) - A^{\mathrm{out,\,SSFM}}(\tau;\gamma)|)\,\mathrm{d}\tau}{\int_T |A^{\mathrm{out}}(\tau)|\,\mathrm{d}\tau}, \tag{4.4}$$

in which $A^{\mathrm{out}}(\tau)$ is the measured received signal, $A^{\mathrm{out,\,SSFM}}(\tau;\gamma)$ the transmitted signal after numerical propagation with γ, and $T = [T_0, T_1]$ the time window of the considered signal. This error is evaluated for every γ in a grid, and the γ with the smallest error is kept. For the numerical propagation, the transmitted signal is first re-sampled using Fourier interpolation at sampling frequency equal to four times the 99%-energy bandwidth of the transmitted signal. We found that this sampling rate was required to obtain accurate results.

We note here that the identified γ implicitly depends on the assumed values of β_2 and α using the SSFM-method (this will also hold for the NFT-based methods). Using wrong values here implies that the value of γ does not correspond to the actual value of the fiber. However, the identified value will still correspond to the γ describing the system best given the assumed α and β, and would therefore still be 'optimal' for the purpose of signal processing.

The spatial step Δl is chosen as large as possible, but small enough that the identified value for γ has converged. The propagation part of the SSFM-based method is performed using a fast *C* version of the software *ssprop* [96], while the error is calculated

in *MATLAB*. This SSFM-based method requires $\mathcal{O}(N\log(N)L)$ floating point operations (FLOPs), where L is the total link length, and $N\log(N)$ is the computational cost of the fast Fourier transforms.

4.2.1. SSFM-BASED IDENTIFICATION FROM WINDOWED SIGNALS

The SSFM-based algorithm processes a burst of a finite duration. The boundaries of the burst should contain sufficient guard intervals in order to account for signal broadening and the periodicity of the discrete Fourier transform. When only a cut-out part of the burst is processed, the guard intervals are missing and these effects have to be taken into account.

We formally define a 'windowed' signal as a segment (i.e., window) $q(\tau)$, $\tau \in [T_0, T_1]$, cut-out from a full signal. The windowed signal is identical to the full signal for the duration of the window. The numerical SSFM propagation implicitly assumes that outside the window, the signal repeats periodically, and thus the numerical propagation will differ from the physical propagation. However, this difference is only limited to a boundary region at the edges of the window. We approximate the size of the affected boundary region using the formula for pulse broadening (pb) [62, Eq. 2.3.3]:

$$\Delta T^{\mathrm{pb}} = |L\beta_2\Delta\Omega|, \tag{4.5}$$

where ΔT^{pb} is the amount that a Gaussian-shaped signal with radial bandwidth $\Delta\Omega$ gets broader as it travels through a fiber of length L with dispersion coefficient β_2. For the purpose in this study, it sufficed to estimate the pulse broadening using only the dispersion, as we noticed that the contribution of nonlinearity to pulse broadening was insignificant in comparison to the dispersion for the considered links.

We found that we only need to cut away a quarter of the pulse broadening at both window boundaries to sufficiently prevent the influence of the pulse broadening. We can thus define an interior region of the window, that is barely affected by the windowing of the signal. We define the interior of the window as

$$T^{\mathrm{int}} = [T^{\mathrm{int}}_{\mathrm{left}}, T^{\mathrm{int}}_{\mathrm{right}}] = [T_0 + \tfrac{1}{4}\Delta T^{\mathrm{pb}},\ T_1 - \tfrac{1}{4}\Delta T^{\mathrm{pb}}]. \tag{4.6}$$

When the SSFM-based method is applied to windowed signals, we will simulate the propagation of the windowed signal in the whole window T, but only consider the error on the interior T^{int}:

$$E(\gamma) = \frac{\int_{T^{\mathrm{int}}} (|A^{\mathrm{out}}(\tau) - A^{\mathrm{in,\,SSFM}}(\tau;\gamma)|)\,\mathrm{d}\tau}{\int_{T^{\mathrm{int}}} |A^{\mathrm{out}}(\tau)|\,\mathrm{d}\tau}. \tag{4.7}$$

4.3. NONLINEAR FOURIER TRANSFORM-BASED FIBER NONLINEARITY IDENTIFICATION USING SOLITONS

An alternative method for fiber-nonlinearity identification considers the solitonic content in transmitted signals. Solitons are stable, particle-like waves resulting from a balance between the dispersive and the nonlinear effects. Due to their stability, solitons existing in the transmitted signal are normally also present at the receiver, while only

moderately affected by noise. Although solitons are often not directly recognizable from the signal shape, earlier research has already shown that many conventional fiber-optic transmission signals contain significant amounts of solitons [54, 86, 97]. The nonlinear Fourier transform (NFT) can be used to determine the solitonic spectrum of a signal, in which the nonlinearity coefficient γ plays the role of a parameter that is used to normalize the signal. When the correct γ value is assumed in the normalization, the solitons at the transmitter are identical to the solitons at the receiver for a noiseless and lossless fiber, while a wrong γ value will result in a mismatch between transmitted and received solitons. We may thus identify γ by determining the value at which the transmitted solitons and received solitons match best. The greatest advantage of NFT-based identification is that no signal propagation is required. The required time of the method therefore does not directly scale with the link length, as opposed to the SSFM-based method. In the following, we recapitulate the most important aspects of NFT-based nonlinearity identification.

4.3.1. NORMALIZED NONLINEAR SCHRÖDINGER EQUATION

The soliton content of a signal may be determined using the nonlinear Fourier transform. The NFT is typically computed from the lossless normalized and dimensionless NLSE. To obtain a lossless NLSE, we apply path-averaging to the lossy NLSE from Eq. 4.1:

$$Q = Ae^{\alpha l/2}, \quad \gamma_1 = \frac{1}{L_\mathrm{s}}\int_0^{L_\mathrm{s}} \gamma e^{-\alpha l}\,\mathrm{d}l = \gamma\frac{1-e^{-\alpha L_\mathrm{s}}}{\alpha L_\mathrm{s}}, \tag{4.8}$$

$$\Rightarrow \quad Q_l \approx -i\frac{\beta_2}{2}Q_{\tau\tau} + i\gamma_1|Q|^2Q, \tag{4.9}$$

where γ_1 is the path-averaged nonlinearity coefficient, and L_s is the uniform span-length. Finally, the equation is normalized [28]:

$$t = \frac{1}{T_0}\tau, \quad q = T_0\underbrace{\sqrt{|\gamma_1/\beta_2|}}_{c_q}Q, \quad z = \frac{1}{T_0^2}\underbrace{(-\beta_2/2)}_{c_z}l, \tag{4.10}$$

$$\Rightarrow \quad q_z = iq_{tt} + 2i|q|^2q, \tag{4.11}$$

where c_q is the amplitude normalization constant (c_q^2 is the effective nonlinearity-dispersion ratio of the fiber link), and c_z the space normalization constant. The time normalization $T_0 \neq 0$ is a free parameter. Similar to the linear Fourier transform, it simply rescales the NFT spectrum. Throughout this study, we use the receiver sampling time $T_0 = \Delta t = 0.0125\,\mathrm{ns} = 1/(80\,\mathrm{GHz})$.

4.3.2. NONLINEAR FOURIER TRANSFORM

After normalizing the signal according to the fiber parameters β_2, γ, α, and L_s, the solitons can be extracted by computing the NFT of $q(t)$. The NFT is found by solving the

Zakharov-Shabat scattering problem [9],

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\phi_1(t,\lambda)\\ \phi_2(t,\lambda)\end{bmatrix} = \begin{bmatrix}-i\lambda & q(t)\\ -q^*(t) & i\lambda\end{bmatrix}\begin{bmatrix}\phi_1(t,\lambda)\\ \phi_2(t,\lambda)\end{bmatrix}, \tag{4.12a}$$

$$\text{s.t.}\begin{bmatrix}e^{-i\lambda t}\\ 0\end{bmatrix} \overset{t\to-\infty}{\longleftarrow} \begin{bmatrix}\phi_1(t,\lambda)\\ \phi_2(t,\lambda)\end{bmatrix} \overset{t\to+\infty}{\longrightarrow} \begin{bmatrix}a(\lambda)e^{-i\lambda t}\\ b(\lambda)e^{+i\lambda t}\end{bmatrix}, \tag{4.12b}$$

in which $\phi(t,\lambda)$ is the eigenfunction corresponding to the complex spectral parameter $\lambda = \xi + i\eta$, and $a(\lambda)$ and $b(\lambda)$ are the scattering coefficients indicated by the right boundary conditions in Eq. 4.12b. The full NFT spectrum finally consists of a discrete (solitonic) spectrum, and a continuous spectrum. We define the continuous spectrum as the value of $b(\lambda)$ over the real axis, $\Lambda^{\mathrm{c}} := \{b(\xi) : \xi \in \mathbb{R}\}$, and the discrete spectrum using the zeros of $a(\lambda)$ in the upper half plane, $\Lambda^{\mathrm{d}} := \{(\lambda_k, b(\lambda_k)) : \Im(\lambda_k) > 0, a(\lambda_k) = 0\}$. Each eigenvalue $\lambda_k = \xi_k + i\eta_k$ corresponds to a soliton. The eigenvalue λ_k defines the shape and speed of the soliton, while the b-coefficient $b_k = b(\lambda_k)$ provides information about the soliton location and phase [7]. Throughout the rest of this study, we are only interested in the eigenvalues.

4.3.3. NONLINEARITY IDENTIFICATION FROM EIGENVALUE MATCHING

The Kerr-nonlinearity coefficient may be determined by comparing the NFT spectra of a transmitted signal and its corresponding received signal for various values of γ, which influence the NFT spectra through the normalization in Eqs. 4.8–4.11. As the value of γ is varied, there is an optimal value at which the NFT spectra at the transmitter and the receiver match best. While both the continuous spectrum and discrete spectrum may be used for spectral matching, the continuous spectrum often contains little power. We observed that the power in the continuous spectrum of the considered signals was only 2% at 2 dBm, 6% at -1 dBm, 10% at -4 dBm and 15% at -7 dBm signal power. The power in the continuous spectrum thus seems to be low, which, for some NFT-based transmitters, can even be proven mathematically [98]. In general many signals close to the linear regime still have a significant portion of their energy in the discrete spectrum, and can contain many solitons [54, 86]. There are also specifically designed nonlinear frequency division multiplexed (NFDM) signals that only use the continuous spectrum [99]. In such situations, the discrete spectrum version of the algorithm from [54] cannot be utilized, but instead the version for the continuous spectrum given there should be used. Finally, it was also observed in [54] that using the continuous spectrum of low-energy signals to determine γ also causes significant bias. We therefore chose not to use the continuous spectrum for identification.

We thus only consider the discrete spectrum, and follow the discrete spectrum matching method used in [54] and [55], which is summarized below:

1. A γ value is selected from a grid, and its corresponding c_q value is used for the normalization.

2. The eigenvalues of the transmitted and received signals are determined from the normalized signals.

3. The matching error E is determined as follows:

$$E = \min_{m(k)} \frac{\sum_k E_{km(k)}}{\sum_k \Im\left(\lambda_k^{\text{in}}\right) + \sum_m \Im\left(\lambda_m^{\text{out}}\right)}, \quad \text{with} \tag{4.13a}$$

$$E_{km} = \min\left(|\lambda_m^{\text{out}} - \lambda_k^{\text{in}}|, \Im(\lambda_k^{\text{in}} + \lambda_m^{\text{out}})\right), \tag{4.13b}$$

where $m(k)$ denotes the perfect matching which connects the input eigenvalue λ_k^{in} to the output eigenvalue λ_m^{out}, and E_{km} is the cost of connecting these eigenvalues. In case the input and output spectra have different numbers of eigenvalues, unmatched eigenvalues of the larger spectrum are assigned a maximum cost: $E_{k-} = \Im\left(\lambda_k^{\text{in}}\right)$, $E_{-m} = \Im\left(\lambda_m^{\text{out}}\right)$. By assigning a maximum cost to eigenvalue pairs and to unmatched eigenvalues, each eigenvalue cannot contribute more to the error than its associated energy ($\propto \Im\lambda_k$). This ensures that random low eigenvalues with very little energy do not dominate the error, but the high (energetic) eigenvalues are most important. After assigning each matching a cost, the minimum-cost matching may be efficiently determined (e.g., using the Hungarian algorithm [84]).

The steps 1)-3) are repeated for every γ value in the grid. The γ value with the lowest error is kept. This procedure is performed for each signal block, and all estimates for γ are averaged for a final estimate.

For this method, we found accurate results when using a sampling rate of two times the bandwidth of the transmitted signal, so only half the number of samples compared to the SSFM-based method. The full discrete spectrum was determined using the software library *FNFT* (branch *add_bsloc_methods2*, commit 9756b3) [35]. We used the default sub-sample and refine method with the *4split4B* discretization. The desired number of samples after sub-sampling was chosen using the Nyquist rate. In this configuration finding the eigenvalues requires $\mathcal{O}(N^2)$ FLOPs, where N is the number of samples.

4.3.4. NFT-BASED IDENTIFICATION FROM WINDOWED SIGNALS

When considering a windowed signal, the NFT implicitly assumes it to be zero outside the window while in reality it is not. The unaccounted interactions with the outside lead to distortions in the soliton spectra. We investigated the option of deriving the location of each soliton based on its b-coefficient, and only accept the eigenvalues that were inside the interior for the matching. However, we found that doing so did not significantly improve the NFT-based identification. We therefore decided to omit it.

We investigated why accepting and discarding solitons based on their location did not significantly improve the identification algorithm. We found that windowing mostly influences the low eigenvalues, but their contribution to the full error is very limited. Furthermore, we observed that most of the medium and high eigenvalues appear inside the interior by default, and also stay there during propagation. We believe there are two reasons for this.

First, any potentially high eigenvalue close to the edge has part of its soliton cut off, and its eigenvalue is thus lowered in the process. Most of the detected high eigenvalues are thus already inside the interior. Second, all large solitons (corresponding to high eigenvalues) have a short duration, and thus large bandwidth. This large bandwidth should fit entirely within the signal bandwidth, and thus their center frequency

lies closer to the center (i.e., zero). The drift speed of each soliton is proportional its center frequency [28], which is therefore also relatively low. The highest solitons therefore have relatively low drift speeds, and thus usually stay within the window during the transmission.

As accepting and rejecting eigenvalues due to the windowing does not seem to influence the highest eigenvalues too much, we will use the same NFT-based algorithm for both full signals and windowed signals.

4.4. FAST NFT-BASED NONLINEARITY IDENTIFICATION THROUGH LOCAL REFINEMENT OF HIGH EIGENVALUES

In this section, we adapt the NFT-based algorithm from the previous section to speed it up, and create a fast NFT-based algorithm. To do so, we only consider the highest few eigenvalues of the signal. The highest eigenvalues contain most of the signal energy, and are also more robust to noise and model mismatch compared to lower eigenvalues. As we consider less information from the signal, the accuracy of the identified γ could slightly worsen. However, this loss in accuracy should be limited. It may even be beneficial to disregard the lower eigenvalues when they are known to be unreliable, as we will observe in some of the results.

The eigenvalues are a continuous function of γ. Therefore, as we sweep the value for γ in small steps, the eigenvalues at the new γ value will be close to the previous eigenvalues. We may thus refine the eigenvalues at the previous γ value to find the eigenvalues at the current γ value. Local refinement of the K highest eigenvalues only takes $\mathcal{O}(KN)$ FLOPs, as opposed to the $\mathcal{O}(N^2)$ FLOPs when the full spectrum has to be determined from scratch.

Our new strategy will be thus to compute a full spectrum only once, and then use the highest K eigenvalues as initialization for local refinement as the γ value is being varied. After the eigenvalues have been determined for the initial γ value, sweeping γ takes significantly less time than computing a full NFT at every γ.

The fast NFT-based identification algorithm proposed here is similar to the original NFT-based algorithm from Sec. 4.3.3 in the main lines. In both cases, we sweep over a grid of values for γ, and measure the error by how well the input- and output-eigenvalues match. The main difference is that we will now only match the highest few eigenvalues, and use a fast refinement method to update the eigenvalues when proceeding from one γ value to the next.

The full algorithm is summarized in Alg. 1, and has the following steps: 1)-2) Re-scale the time, the received signal and the transmitted signal using T_0. Normalize the transmitted and received signals using the $c_q(\gamma)$ (Eq. 4.10) corresponding to the highest value of γ in the grid, resulting in normalized signals $q^{0,\text{in}}$ and $q^{0,\text{out}}$. 3)-4) Calculate the full discrete spectrum of $q^{0,\text{in}}$ and $q^{0,\text{out}}$ using the NFT in Sec. 4.3.2. 5) Store the highest K^{high} input eigenvalues, and the highest $3K^{\text{high}}$ output eigenvalues. These will be used as initial guesses. More output eigenvalues than input eigenvalues are kept, as we will only focus on matching all the highest input eigenvalues. When we took the same number of output eigenvalues as input eigenvalues, we found that the highest K input eigenvalues and highest K output eigenvalues found at the initial guess $c_q(1)$ would often not

Algorithm 1: Identifying c_q by matching high eigenvalues

Input:

- Transmitted signal, $A^{\text{in}}(\tau)$,
- Received signal, $A^{\text{out}}(\tau)$,
- Grid of decreasing normalization constants $c_q(1) > c_q(2) > \cdots > c_q(J)$,
- Number of high eigenvalues to compare, K^{high},
- Time normalization, T_0.

Output:

- c_q^{ID}.

Algorithm:

- $t = \tau / T_0, \quad Q^{\text{in}}(t) = T_0 A^{\text{in}}(\tau), \quad Q^{\text{out}}(t) = T_0 A^{\text{out}}(\tau)$;
- $q^{0,\text{in}}(t) = c_q(1) Q^{\text{in}}(t), \quad q^{0,\text{out}}(t) = c_q(1) Q^{\text{out}}(t)$;
- $\{(\lambda_k^{0,\text{in}}, b_k^{0,\text{in}}), k = 1, \ldots, K^{\text{all}}\} \leftarrow \text{NFT}(q^{0,\text{in}}(t))$;
- $\{(\lambda_m^{0,\text{out}}, b_m^{0,\text{out}}), m = 1, \ldots, M^{\text{all}}\} \leftarrow \text{NFT}(q^{0,\text{out}}(t))$;
- Keep the highest K^{high} input eigenvalues, and highest $3K^{\text{high}}$ output eigenvalues;

for $j = 1, \ldots, J$ **do**

- $q^{j,\text{in}}(t) = c_q(j) Q^{\text{out}}(t), \quad q^{j,\text{out}}(t) = c_q(j) Q^{\text{out}}(t)$;
- $\{(\lambda_k^{j,\text{in}}, b_k^{j,\text{in}}), k \le K^{\text{high}}\} \leftarrow \text{NFT}(q^{j,\text{in}}) \,|\, \lambda_k^{j-1,\text{in}}$;
- $\{(\lambda_m^{j,\text{out}}, b_m^{j,\text{out}}), m \le 3K^{\text{high}}\} \leftarrow \text{NFT}(q^{j,\text{out}}) \,|\, \lambda_m^{j-1,\text{out}}$;
- $E(j) \leftarrow \left(\{\lambda_k^{j,\text{in}}\}, \{\lambda_m^{j,\text{out}}\}\right)$ from Eq. 4.14;

end

-Return $c_q(j)$, at which $E(j)$ is minimal.

correspond to the highest K eigenvalues at the correct c_q as initially high eigenvalues often drift downwards as c_q is varied. To ensure that at the correct c_q all input eigenvalues have their correct counterpart present in the output eigenvalues, we thus need to consider more initial output eigenvalues than input eigenvalues. For the purpose in this study we observed that considering three times as many output eigenvalues usually sufficed to include the correct counterparts of all the considered input eigenvalues, while still keeping the computation time low by considering only a small set of eigenvalues. This step concludes the initialization of the input and output eigenvalues. 6) Loop over all values of $c_q(\gamma)$, starting with the highest value. Perform steps 7)-10) in every loop. 7) Normalize the transmitted and received signals corresponding to the current value $c_q(\gamma)$. 8)-9) Determine the K^{high} highest input eigenvalues, and the $3K^{\text{high}}$ highest output eigenvalues, by locally refining (using Newton's method, *bsloc_newton* in *FNFT*) the eigenvalues found with previous normalization $c_q^{\text{j-1}}$. 10) From the current set of K^{high} input and $3K^{\text{high}}$ output eigenvalues, calculate the matching error E as follows:

$$E = \frac{\sum_{k=1}^{K^{\text{high}}} E_{km(k)}}{\sum_{k=1}^{K^{\text{high}}} \Im\left(\lambda_k^{\text{in}}\right)}, \quad \text{with} \tag{4.14a}$$

$$E_{km} = \min\left(|\lambda_m^{\text{out}} - \lambda_k^{\text{in}}|, \Im(\lambda_k^{\text{in}})\right), \tag{4.14b}$$

where $m(k)$ is the minimum-cost matching [84], which assigns each input eigenvalue k to output eigenvalue m at the cost E_{km}. 11) After the error was determined for every c_q, return the value c_q at which the error is minimized. This concludes the algorithm.

We finally note that the same algorithm is used for full signals and for windowed signals, for the same reasons as in Sec. 4.3.4.

4.5. EXPERIMENTAL RESULTS

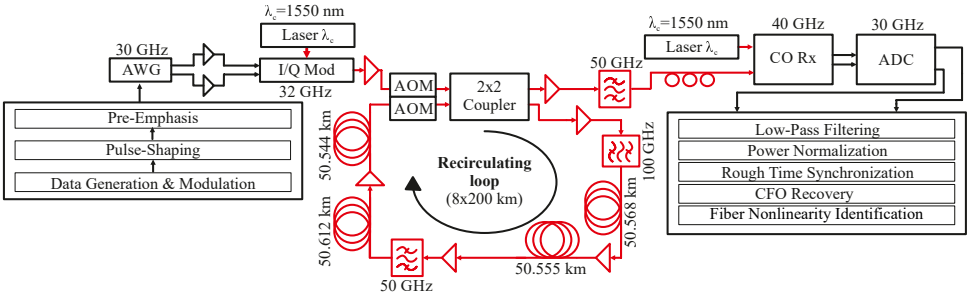


Figure 4.1: The experimental setup with exact fiber lengths, positions of optical filters, and applied signal-processing. Acronyms: AWG = arbitrary waveform generator; AOM = acousto-optic modulator; CFO = carrier frequency offset.

In this section, we compare the SSFM-based algorithm, the original NFT-based algorithm and the fast NFT-based algorithm with only high eigenvalues on experimental data. First, we compare the speed and accuracy of these three methods on full bursts separated by guard intervals. This ensures the signal is not influenced by neighboring pulses, which is implicitly assumed in both the SSFM-, and in the NFT-based methods. Second, we compare the three methods on windowed signals that are cut out from the same bursts. Windowing signals provides control over the signal length, and thus the computation time. However, this does imply that the edges of the windowed signals are affected by their surrounding, which is not the case for the complete bursts with guard intervals.

4.5.1. EXPERIMENTAL SETUP

The experimental setup is shown in Fig. 4.1. All digital signals were pre-compensated for the measured frequency response of the back-to-back transceiver setup. Digital-to-analogue conversion was done using an 88 GSa/s arbitrary waveform generator (AWG). The analogue signal was converted into the optical domain at 1550 nm carrier wavelength using an I/Q modulator and a laser with <100 kHz linewidth. The optical signal was amplified to 2 dBm launch power before every fiber span, and circulated 8 times through a loop of 4 spans of 50 km OFS AllWave SMF for a total of 1600 km. Although higher launch powers are desirable for both nonlinear communication and identifying the nonlinearity coefficient, 2 dBm was the highest launch power we could use due to limitations in the experimental setup. The used launch power was well above the levels typically used in linear transmission, so the nonlinearity should still be sufficiently large to demonstrate and qualitatively compare all identification methods.

The reference fiber coefficients for α and β_2 were taken from the data sheet: $\beta_2^{\text{ref}} = -21.2\frac{\text{fs}^2}{\text{km}}(D = 16.6\frac{\text{ps}}{\text{nm}\cdot\text{km}}), \alpha^{\text{ref}} = 0.19\frac{\text{dB}}{\text{km}}$. The value for γ was not provided by the data sheet, so we used a typical value from the literature [100, p.157] as initial guess: $\gamma^{\text{ref}} = 1.26\frac{1}{\text{Wkm}}$. After polarization de-rotation, the optical signal was received using a coherent receiver using a local laser with <10 kHz linewidth, and an oscilloscope with 80 GSa/s.

The signal was post-processed as indicated in Fig. 4.1. Finally, we used the SSFM-based method with a space-step size of $\Delta l = 1000\,\mathrm{m}$, and the NFT-based methods to estimate the nonlinearity coefficient γ. Both the SSFM-based method and NFT-based methods were run on the same computer to ensure that the computation times shown in Tab. 4.1 can be fairly compared.

4.5.2. IDENTIFICATION FROM COMPLETE BURSTS

We first compare the identification algorithms on bursts with sufficient guard intervals, such that neighboring bursts do not interfere during the propagation. Therefore, the SSFM- and NFT-based methods can both be applied individually to each burst, without paying additional attention to interactions with neighboring bursts. Each burst consists of a summation of time-shifted raised cosine shaped carriers (roll-off factor 0.5), each multiplied with a symbol. More precisely, the transmitted signal $q^{\mathrm{in}}(\tau)$ was given by

$$A^{\mathrm{in}}(\tau) = \sum_{s}^{N_s} a_s A^{\mathrm{c}}(\tau - sT^{\mathrm{c}}), \quad \text{with} \tag{4.15a}$$

$$A^{\mathrm{c}}(\tau) = \begin{cases} \frac{\pi}{4T^{\mathrm{c}}}\operatorname{sinc}\left(\frac{\tau}{T^{\mathrm{c}}}\right), & \tau = \pm T^{\mathrm{c}}, \\ \frac{1}{T}\operatorname{sinc}\left(\frac{\tau}{T^{\mathrm{c}}}\right)\frac{\cos(\pi\tau/2T^{\mathrm{c}})}{1-(\tau/T^{\mathrm{c}})^2)}, & \text{otherwise}, \end{cases} \tag{4.15b}$$

in which a_s is the QPSK-symbol with index s, $N_s = 128$ the number of symbols in a single burst, $T^{\mathrm{c}} = 0.1\,\mathrm{ns}$ the carrier spacing, and A^{c} the raised cosine carrier. The full burst duration with guard intervals was 16 ns. This signal-type was transmitted 100 times with random QPSK symbols over 1600 km at 2 dBm launch power. We observed that this type of signal contains about 70 solitons when normalized using $\gamma = 1.261/(\mathrm{W{\cdot}km})$ (small variations occur depending due to the randomness of the symbols). The transmitted signals and corresponding received signals were used for the identification of γ. For a fair comparison, we consider the same grid of γ values for each of the three methods, and determine the error for every γ in the grid. The considered grid for the γ values ranged from $0.8\gamma^{\mathrm{ref}}$ to $1.2\gamma^{\mathrm{ref}}$, in 21 equidistant steps. We chose this grid as typical values of γ in data sheets include about 2% uncertainty [101]. A few percent difference in the used γ only marginally affect the received signal (and present solitons), and is typically acceptable for signal processing. The range of the grid was chosen large enough such that the distribution of estimated γ could be observed, as well as that we expect the correct γ to be within 20% of the reference value for the considered fiber.

For each of the methods, γ was identified for each of the 100 bursts, and a final estimate was calculated by averaging all estimates that were not at the edge of the grid. Estimates at the edge were discarded, as this implies that the error failed to converge within reasonable values for γ, and were thus considered outliers. The resulting γ distributions are shown in Fig. 4.2, and the final estimated γ values are shown in Tab. 4.1.

We note that the distribution of the full NFT and SSFM-based methods are similar, and peak at approximately the same position of $\gamma = 1.201/\mathrm{W{\cdot}km}$. We will consider this identified coefficient as the 'correct' coefficient to compare later results with. Although the full NFT-based method and SSFM-based method yield the same final estimate, the standard deviation of the distribution for the NFT-based method is larger than the variance for the SSFM-based method, which was also observed in an earlier study on simulation

data [54]. We believe this is mostly due to the mismatch in the path-average approximation. In [102], it was shown that the stability of path-average second-order solitons in specific signals start to degrade significantly for signals around 2 dBm launch power, potentially explaining the large variance in the estimates for γ.

Next, we observe that the identified γ of the fast NFT-based method with 10 eigenvalues contains a 7% bias towards higher values of γ. We already observed that the input and output eigenvalues do not perfectly match due to the path-average approximation and noise, and that this causes a larger standard deviation for the identified γ. We suspect that the bias occurs because higher γ values generally create denser clouds of high eigenvalues, as more high eigenvalues are present due to more nonlinearity. The high input eigenvalues may then randomly find a closer output eigenvalue than at the correct γ, causing a bias towards slightly higher γ.

The computation times for the three identification algorithms are shown in Tab. 4.1. We observe a speed-up of a factor 5 when considering only 10 eigenvalues in the fast NFT-based method compared to considering the full spectrum for every γ value. The SSFM-based method was still slightly faster than the fast NFT-based method.

We thus conclude that the SSFM-based method resulted in the most accurately identified γ values, while also being the fastest method for this signal. However, with $N = 358$ samples, this signal was rather long. The original NFT-based method requires two full NFTs per γ value in the grid, so 42 NFTs in total for this grid. The fast NFT-based method also requires two full NFTs for initialization before the eigenvalue refinement. As both NFT-based methods calculate full NFTs, both require $\mathcal{O}(N^2)$ FLOPs, while the SSFM-based method only requires $\mathcal{O}(N \log N)$ FLOPs given that the number of spatial steps

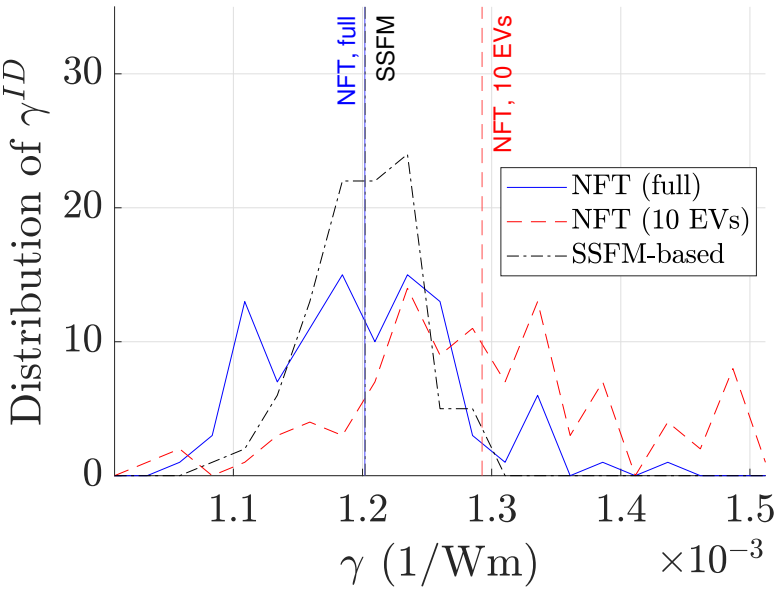
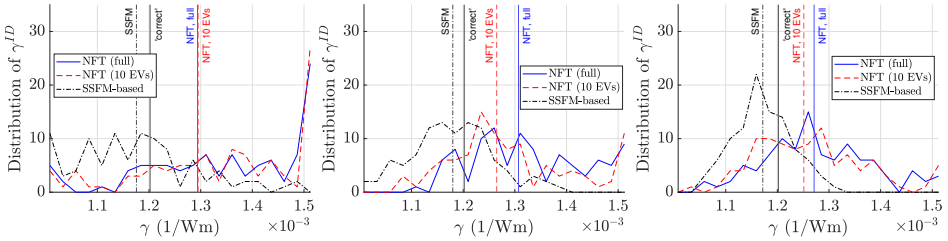


Figure 4.2: The distribution of the identified γ from raised cosine bursts with vanishing tails.

is kept constant. Considering shorter signals could thus allow NFT-based methods to perform faster. In the next section we therefore window the bursts to control the signal length, and compare the NFT- and SSFM-based methods on these signals.

4.5.3. IDENTIFICATION FROM WINDOWED SIGNALS



(a) 3.2 ns window (b) 5.6 ns window (c) 8.0 ns window

Figure 4.3: Distributions of identified γ from a windowed raised-cosine signal for various window durations.

In this section, we consider the identification of the nonlinearity coefficient from windowed signals (see Sec. 4.2.1), short segments cut out from the bursts. In practical communication systems, bursts are often made as long as possible to minimize the impact of the guard interval on the spectral efficiency. Therefore, it will often occur that only long signals are available for the identification, so that windowing might significantly reduce the complexity. As opposed to the complete bursts from the previous subsection, the edges of the windowed signals will be affected by the signal outside the window, and thus this has an impact on the identification. On the other hand, windowing the signal has the benefit that the length of the considered signal can be controlled, and thus the identification may be sped up by considering shorter windows.

The SSFM-based method was slightly adapted to deal with the cut-off part of the bursts as described in Sec. 4.2.1. For both NFT-based methods, we used the same algorithms as before, as explained in Sec. 4.3.4.

For the identification in this section, we consider the raised cosine signals as in the previous section, but now consider only a small window centered around the middle of each raised cosine burst. A single block of the raised cosine signals had a duration of 16 ns, of which 3.2 ns was guard interval and the block of symbols was 12.8 ns. We are interested in the case where the considered window is a piece from a continuous stream of symbols, and thus we consider a center piece from the block which is sufficiently far away from the guard intervals. The pulse broadening (Eq. 4.5) was approximately 1.2 ns. The smallest window we could theoretically consider would be 1.2 ns, but this entire window would then be influenced by the outside of the window. To ensure that we have a sufficiently large interior of the window, we chose the smallest window size as 3.2 ns.

The largest possible window size was 11.6 ns. Considering larger windows also becomes less efficient due to the superlinear scaling of the computation time as a function of signal duration. Instead of considering a long signal, it would be much faster to consider two signals with half the duration at some point. We thus chose the largest window

to be 8.0 ns.

We also considered an intermediate window size of 5.6 ns. For these three different window sizes, we compared the algorithms in terms of accuracy and speed.

The distributions of the identified γ value from 100 windowed signals are shown in Fig. 4.3 for three different windows, and the identified γ can also be found in Tab. 4.1. For the shortest window of 3.2 ns, we observe a large standard deviation in the distribution of the γ values for the NFT-based methods, while many of the identified γ are on the edge of the grid. The SSFM-based method also shows a large standard deviation, but close to the expected value of $\gamma = 1.20$ 1/(W·km) identified from the full bursts. This window is barely large enough to effectively identify the value of γ using the SSFM-based method, while both NFT-based methods contain too many outliers to be reliable. Upon inspection, it indeed turned out that there were only about 15 eigenvalues within this windowed signal type, of which only a few were high enough to be unaffected by the windowing.

The window of 5.6 ns duration shows clearly distinguishable peaks around the expected value for γ. For this window size, the SSFM-based method becomes reliable, and the fast NFT-based method with 10 eigenvalues is closer to the expected γ value than the full NFT. This is likely because the full NFT contains many low eigenvalues that become unreliable when the signal is windowed. Although the low eigenvalues have a limited influence on the error in Eq. 4.13a, they can still bias the estimate. The bias towards higher values of γ is again attributed to random good matchings that are more likely due to the denser cloud of eigenvalues at higher nonlinearities. Next, the 8 ns window shows clear peaks in all distributions, and the full NFT and refined NFT show similar distributions. However, both NFT-based estimates show a clear deviation from the SSFM-based method, and are about 5% higher. The windowing thus seems to structurally bias both NFT-based methods. We also investigated other intermediate window durations, but these results were very similar to the shown results (or simply interpolations) and were therefore omitted from this study.

Finally, we also investigated the influence of the considered number of high eigenvalues K^{high} for the fast NFT-based method on the identified γ. We compared $k^{\text{high}} \in \{5, 8, 10, 12, 15\}$, with the result shown in Fig. 4.4. Overall, the distributions of the identified γ depend marginally on the chosen K^{high}, although for $K^{\text{high}} = 15$ we found that the bias towards higher γ increases, probably due to considering several low eigenvalues that are influenced by the windowing of the signal. While considering only 5 eigenvalues may put too much weight on single good or bad matchings of eigenvalues, we find that considering between 5-10 eigenvalues leads to results closest to the correct γ. Nevertheless, the influence of K^{high} is still rather limited.

The computation times for the windowed signals are indicated in Tab. 4.1. We observe that the computation time for the SSFM-based method scales approximately linearly with the number of signal samples. As expected, both NFT-based methods scale worse with the number of signal samples than the SSFM based method. Reducing the window to the minimum required length therefore in particular speeds up the NFT-based methods. However, we observed that this can cause a structural deviation in the identified γ value, or causes the methods to fail entirely due to a lack of high eigenvalues. Furthermore, the NFT-based methods required 180 samples before the distribution for the identified γ value stabilized, and at this point the SSFM-based method is already faster.

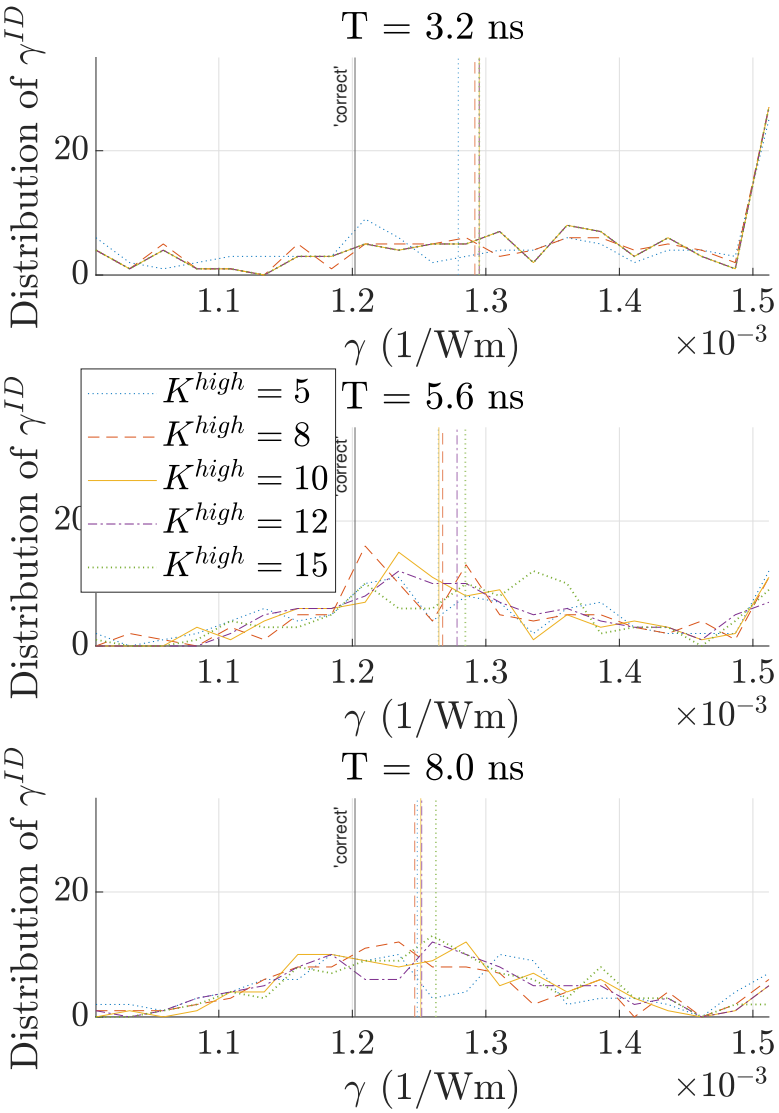


Figure 4.4: The identified γ from windowed signals using a varying number of high eigenvalues K^{high}.

We thus conclude that the SSFM-based method yields more accurate and faster results than both NFT-based methods for the considered 1600 km SSFM link.

4.5.4. DISCUSSION

In this section, we discuss several possible improvements to SSFM- and NFT-based identification methods, and share our view on the feasibility for each option.

Accuracy and consistency. In the shown examples, the SSFM-based method consistently identified $\gamma \approx 1.201/(\text{W·km})$, whereas the NFT-based estimates were often biased, as well as that the estimates had a larger standard deviation. In particular the bias of up to 7% for the NFT-based methods can be too large to be acceptable in some cases. To investigate the reason for the larger standard deviation, the error-curves of the full bursts from Sec. 4.5.2 are shown in Fig. 4.5. When considering the average error in Fig. 4.5b, we observe that the NFT- and SSFM-based errors show a clear optimum. However, we observe that for a single burst the SSFM-based method has a much smoother error-curve than the full NFT-based method. This is caused by sudden changes in the eigenvalue matching as γ is varied. The NFT-based method with only 10 eigenvalues is much smoother, as the high input- and output-eigenvalues usually stay matched in the same pairs, while the error only changes because the eigenvalues drift away/closer as γ is varied. However, as discussed before, the NFT-based matching using only high eigenvalues seems to be biased towards high γ values. We note here that the NFT-based method could be biased as it is optimizing a different criterion: it attempts to only match the eigenvalues as well as possible, instead of the full signals. It may therefore be possible that the NFT-based γ value is optimal in this sense. However, our observations suggest that this is not the case. We believe the bias is introduced due to the denser clouds of eigenvalues at higher γ values that allow for random good matchings. While it might be possible to account for this effect, it seems challenging to do so.

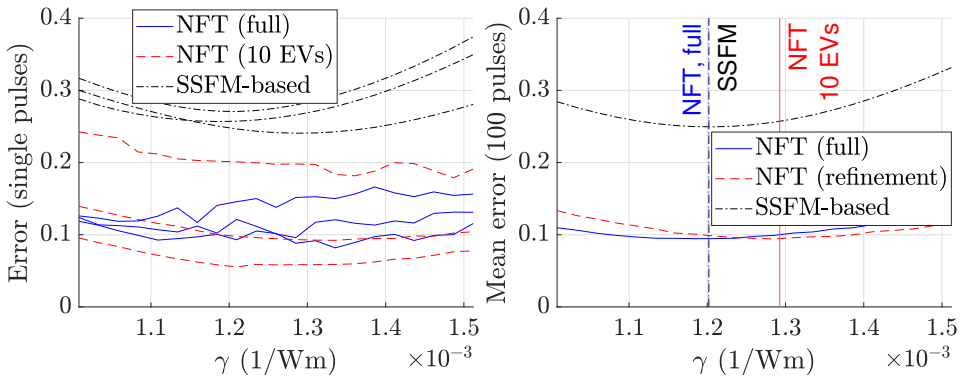
Computation time. We observed that the NFT-based methods can be faster than the SSFM-based method when the fiber length is long and the signal is short, but for the 1600 km link the SSFM-based method is usually faster. However, both the NFT-based methods and SSFM-based methods can be further sped up in various ways. For the NFT-based methods, the most important constraint is the calculation of the discrete spectrum, which takes $\mathcal{O}(N^2)$ FLOPs for the method used in this study, which may also be observed in Table 4.1. When comparing the computation time per signal as a function of the number of signal samples, we indeed observe this relation for the full NFT-based method. The fast NFT-based method uses only two full NFTs for the first grid point, but the refinement-based NFTs for all other grid points require only $\mathcal{O}(N)$ time, so the $\mathcal{O}(N^2)$ scaling is much less visible there. Especially signals with large bandwidths will require more samples to capture, and thus take longer computation times for NFT-based methods (see e.g. the 42 GB scenario). This drawback would pose a problem as the current trend in optical communication is towards high data rates, which are associated with large bandwidths.

Fortunately, recent research has shown that the higher eigenvalues of a signal may be determined quickly using both time windowing *and* frequency windowing [103]. While we were already using some form of time-windowing in this study, the paper [103] showed that frequency windowing (i.e., partitioning the full signal into band-limited signals, calculating the eigenvalues of each frequency-windowed partition, and then recombining the discrete spectra) could drastically reduce the computation time for calculating the higher eigenvalues from signals with large bandwidth. This would largely overcome the drawback of computational scaling in the signal bandwidth associated with NFT-based identification methods. In scenarios with higher bandwidths than in our experiments, an additional frequency windowing step as in [103] is thus recommended to reduce com-

putation times.

The SSFM-based method could be sped up further, for example by using Volterra series for the digital propagation (see e.g., [104]), which require less steps per span for similar accuracy. Furthermore, due to the observed smoothness in the error-curves for the SSFM-based method, the grid search could be replaced by a local-descend method, allowing for even faster identification. We shortly investigated this possibility, and found that the SSFM-based error already converged in approximately four steps to the optimal γ value using Newton's method due to its smooth error-curves.

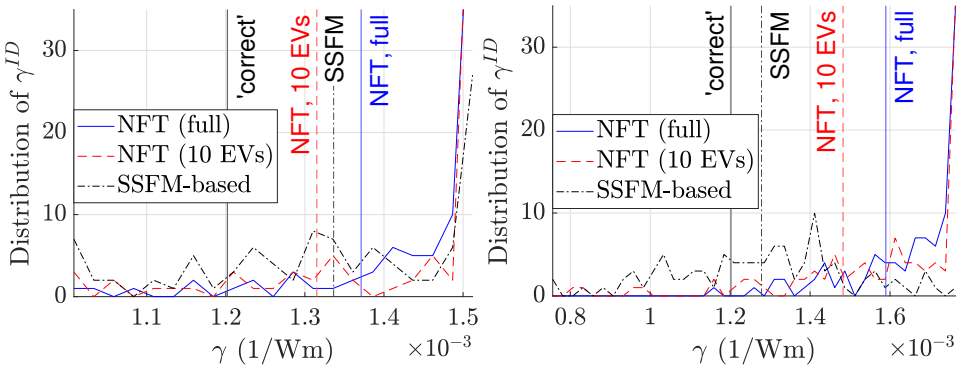
Dependence on power spectral density. In order to identify the nonlinearity coefficient, the signal needs to have a sufficiently high power spectral density such that the nonlinear effects become observable. We shortly consider two cases here with lower power spectral density, and compare how the identification methods perform. The results are shown in Fig. 4.6. The left case considers the same windowed raised cosine signal as from Fig. 4.3c, but with only -4 dBm launch power (four times lower). The right case considers a raised cosine signal at 2 dBm launch power, but with symbol time T four times shorter. This also increases the bandwidth by a factor four, and hence the power spectral density is also four times lower. We observe that the SSFM-based method can still yield reliable estimates, whereas the NFT-based method fails entirely in most of the cases. For smaller power spectral densities, there are much fewer high eigenvalues. As a result the NFT-based methods become very biased towards higher γ values, for which there are more eigenvalues to be matched. We thus observe that the SSFM-based method is also more robust for identification on signals with low power spectral densities. We note here that for low-power signals, the continuous spectrum contains a larger portion of the signal energy and might be used to identify γ.



(a) Error from three different vanishing pulses.

(b) The error, averaged over 100 vanishing pulses.

Figure 4.5: The error-curves of the SSFM-based method and the NFT-based methods for the raised cosine pulse with vanishing tails, from Sec. 4.5.2 and Fig. 4.2.



(a) 8 ns, 13 GHz, -4 dBm

(b) 11 ns, 42 GHz, 2 dBm

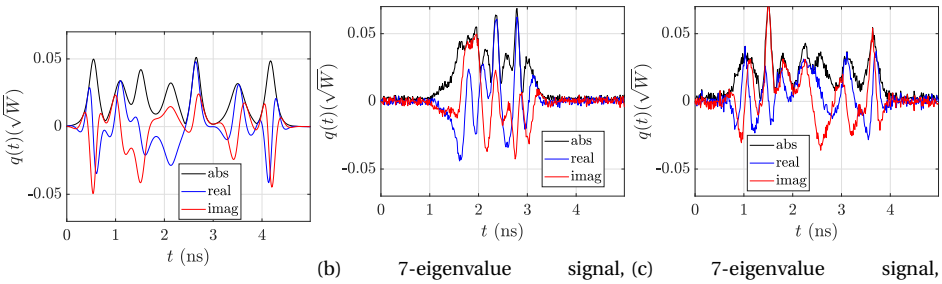
Figure 4.6: The identified γ values for two signal types (duration, bandwidth and power as indicated) where the NFT-based method fails. The left signal type was a raised cosine with too little power for high eigenvalues. The right signal had a broader spectrum, with the same power, and therefore less power per Hz, also resulting in too low eigenvalues. As a result, most of the NFT-based estimates do not converge within the grid. However, the SSFM-based method still shows a peak near the expected value for the right case.

4.6. SIMULATION RESULTS FOR LINKS UP TO 8000 KM, WITH IDEAL RAMAN AMPLIFICATION

In this section, we investigate a scenario in which NFT-based identification should outperform the SSFM-based identification. We will consider simulation data of an 8000 km fiber link, with ideal Raman amplification to ensure a uniform signal energy profile throughout the propagation. This scenario with long link lengths and ideal Raman amplification should be especially suitable for NFT-based identification, as the path-average approximation (Eq. 4.8) is no longer required due to the ideal Raman amplification. Furthermore, the computation time of the SSFM-based method will be about 5 times as long as compared to the 1600 km link due to the increased link length, whereas the NFT-based methods are not directly influenced. Due to practical limitations, we were not able to demonstrate this example experimentally, and we present here results from simulated data instead. Finally, we estimate at which distance NFT-based methods are faster than the SSFM-based method, by also considering simulated links of 1600 km and 4800 km.

4.6.1. SIMULATION CHANNEL

The considered link in the simulations had a total length of 8000 km, while the propagation was simulated with the NLSE from Eq. 4.1, with $\beta_2^{\text{ref}} = -21.2\,\frac{\text{fs}^2}{\text{km}}$ ($D = 16.6\,\frac{\text{ps}}{\text{nm·km}}$), $\gamma^{\text{ref}} = 1.201/(\text{W·km})$, such that the simulated fiber corresponded to the experimental fiber. The loss parameter was set to $\alpha^{\text{ref}} = 0\,\frac{\text{dB}}{\text{km}}$ in the simulation due to the ideal Raman amplification. The propagation was performed by a split-step method, with steps of 2 km. At the end of every step, realistic Raman noise was added to the signal, with spontaneous emission factor of $n_{sp} = 3.5$, corresponding to a pump wavelength at ap-



(a) 7-eigenvalue signal, $z = 0$ km. (b) 7-eigenvalue signal, $z = 4000$ km. (c) 7-eigenvalue signal, $z = 8000$ km.

Figure 4.7: Example of a realization of a 7-eigenvalue signal traveling through an ideally Raman amplified single mode fiber of 8000 km length, at transmitter ($z = 0$), halfway ($z = 4000$ km), and at the receiver ($z = 8000$ km).

4

proximately 1450 nm and a carrier wavelength of 1550 nm [105, Eq. 7.3.9].

4.6.2. SIGNAL TYPES

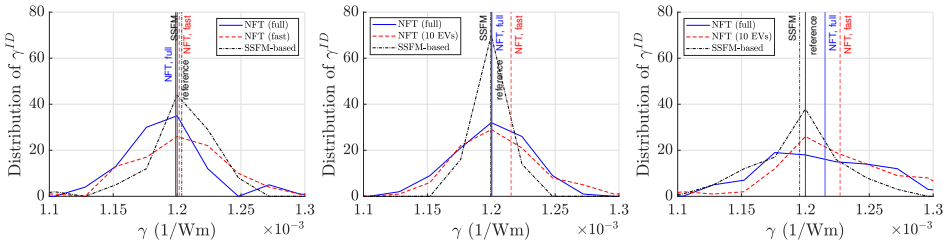
For the simulation, we considered two types of signals: a raised cosine-signal, similar to the one in the previous section, and a 7-eigenvalue signal, tailored to the fiber and designed to mitigate signal broadening, inspired by [83].

First, we considered a 7-eigenvalue signal. The signal was generated by initializing the NFT spectrum with 7 eigenvalues in the discrete spectrum, and $b(\lambda) = 0, \forall \lambda \in \mathbb{R}$ for the continuous spectrum. In a normalized domain (using $c_q = 2.38 \cdot 10^{11}$ $1/(\mathrm{s}\sqrt{\mathrm{W}})$, $T_0 = 1$ s and Eq. 4.10), we set the eigenvalues as $\lambda = 10^9[-7.5 + 5.7i, -5.0 + 3.8i, -2.5 + 5.7i, 0 + 3.8i, 2.5 + 5.7i, 5.0 + 3.8i, 7.5 + 5.6i]$, respectively with absolute b-coefficients $\log|b| = [-18.7, -9.12, -10.38, -2.62, 1.14, 6.24, 15.86]$. The phases of the b-coefficients were randomly assigned to $\pm\pi/4$ or $\pm 3\pi/4$. The bandwidth of this type of signal was approximately 9 GHz. This type of signal corresponds to eigenvalue communication, in which the bits are modulated with QPSK on the phase of the b-coefficient of each eigenvalue. The signal was designed such that the left-most soliton travels to the right and vice versa during the propagation over 8000 km. The effective duration of the signal thus first gets shorter, before getting wider again, and reaching the receiver at the approximate same duration as the transmitted signal. This type of eigenvalue communication signal is illustrated in Fig. 4.7. The launch power of this signal was -3.7 dBm.

Second, we considered the same raised-cosine signal as in the previous section with $N_c = 128$ symbols per block, and $T^c = 0.1$ ns, but adding an 11.2 ns guard interval (instead of 3.2 ns), resulting in blocks of 24 ns. The additional guard interval is required to prevent interference between two blocks, as the additional link length causes each block to spread out much further. We set the average launch power to -4 dBm. This launch power with Raman amplification roughly leads to the same path-average power as +2 dBm launch power for the previous link with lumped amplification. The exact same transmitted 7-eigenvalue signals and raised cosine signals were also used for propagation over 1600 km and 4800 km. We note here that the long guard intervals required for the 8000 km transmission were excessive for the 1600 km and 4800 km transmission, but

we chose to still include the long guard intervals to make the comparison fair.

4.6.3. IDENTIFICATION FROM SIMULATED SIGNALS



(a) 7-eigenvalue type signal, full block (b) Raised-cosine type signal, full block (c) Raised-cosine type signal, half block

Figure 4.8: Identified γ for the SSFM with ideal Raman amplification from 8000 km transmission data. The distributions for 4800 km and 1600 km were similar, and are therefore omitted.

We identified γ from both mentioned types of signals, using both NFT-based methods, and the SSFM-based method. First we considered the full blocks of the 7-eigenvalue type signals, second the full blocks of the raised-cosine type signals, and third the windowed raised-cosine type signals, where each window contained only the middle 50% of each block.

The identified γ value for each signal type, each method and each fiber length are shown in Tab. 4.1. As the γ distributions were very similar for the three considered link lengths, we only show the result for 8000 km in Fig. 4.8. We observe that for the 7-eigenvalue type signal, all three methods yield very accurate results, although the fast NFT-based method shows a slightly larger variance in the identified γ than the other two methods. For both the full and the 50% windows of the raised cosine-type signals, the full NFT-based method as well as the SSFM-based methods again yield highly accurate results with low variance, but the fast NFT-based method again shows a small bias towards higher γ values, and has larger variance. In all scenarios the SSFM-based method yields accurate results with the lowest variance, and is therefore the most accurate method.

However, when comparing the computation times in Tab 4.1 for the 8000 km link, we observe that the full NFT-based method is significantly faster than the SSFM-based method except for the longest signal, and the fast NFT-based method is consistently 4 times as fast as the SSFM-based method for all types of signals. We thus conclude that the NFT-based methods can indeed provide a significant speed up for very long links, at the cost of a small loss in accuracy and increase in uncertainty. Figure 4.9 also shows that the full NFT-based method is faster than the SSFM-based method for the 7-eigenvalue (7-EV) signal already at 2000 km, and for the other two signals comparably fast at 8000 km. The fast NFT-based method using only the 10 highest eigenvalues is already faster for all considered signals for links longer than 2000 km, and much faster at 8000 km.

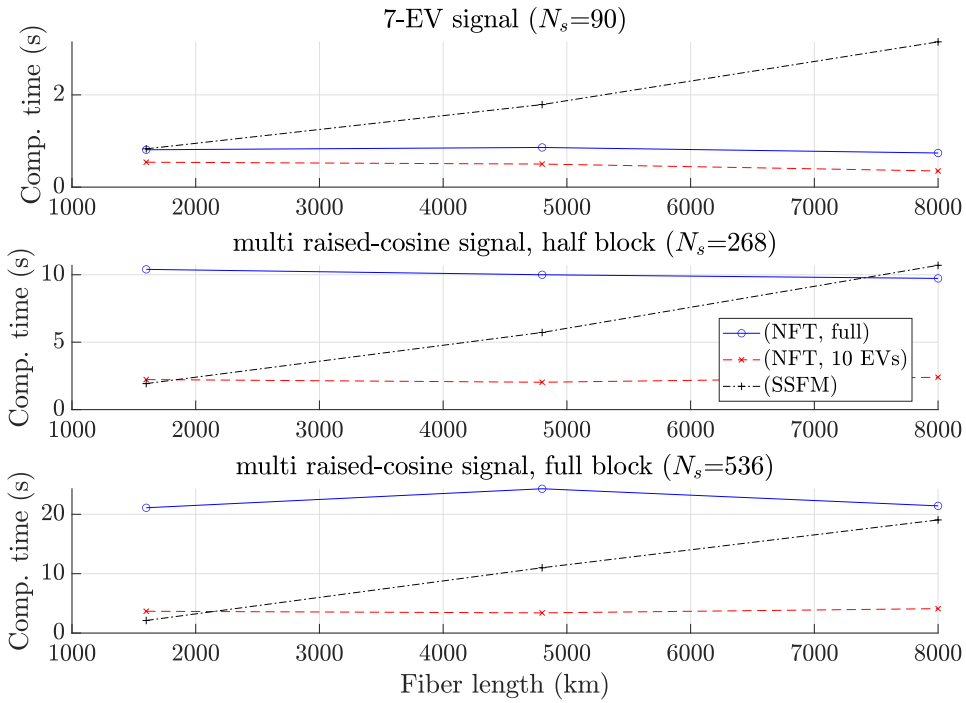


Figure 4.9: The mean computation time per signal for each simulated signal as a function of fiber length, also shown in Tab. 4.1). The computation times for the NFT-based methods remain approximately constant, whereas the SSFM-based method increases linearly with distance.

4.7. CONCLUSION

We compared three different methods for identifying the Kerr-nonlinearity coefficient for an installed optical single-mode fiber link from available transmission data. The first method was based on numerical split-step propagation of the transmitted signal and comparing input and output. The second method compared all solitonic eigenvalues at input and output using the nonlinear Fourier transform (NFT). The third method only compared the highest few eigenvalues, and calculated these faster using local refinement of previous eigenvalues. The third method was first proposed in this study. We compared these three methods on complete bursts, and on cut-out parts of bursts, for which we adapted the SSFM-based method to deal with interference due to the channel memory.

We showed that the SSFM-based method was faster and more accurate than both NFT-based identification methods for an experimental 1600 km link. The NFT-based method with few eigenvalues was significantly faster than the full NFT-based method, but showed some bias when compared using complete bursts with vanishing tails. When applied on a windowed signal without vanishing tails, both NFT-based methods showed some bias, but the method with few eigenvalues was more accurate in this case. Nevertheless, the SSFM-based method was more accurate in all aspects for both the vanish-

ing pulse, and the windowed signal. We thus consider the SSFM-based method to be the method of choice for the identification of the nonlinearity coefficient in operational single-mode fibers at equivalent transmission distances.

Finally, we investigated whether NFT-based identification could outperform SSFM-based identification for a very long link of 8000 km. Indeed the NFT-based methods did not scale with the link length, and showed similar or lower computation times than the SSFM-based method when applied to signals for a simulated ideal-Raman amplified link, although the SSFM-based method was still slightly more accurate. In case that a small loss in accuracy is acceptable, the NFT-based method could therefore be the method of choice for long links.

Table 4.1: The mean computation time to compute the γ value from a single burst, for the 6 different signal types (details specified per row), using the full NFT-based method, the fast NFT-based method using only 10 eigenvalues, and the SSFM-based method. For each signal type, the computation time for the identification was averaged over 100 bursts. The correct γ was 1.20 1/(W·km). The standard deviation in the distribution of the identified γ are also mentioned after the mean identified value.

Figure	# signal samples (×2 for SSFM)	# grid points for γ	bandwidth (99%)	launch power	amplification type	fiber length	comp. time per signal (NFT, full)	comp. time per signal (NFT, 10 EVs)	comp. time per signal (SSFM)	γ^{ID} [1/(W·km)] (NFT, full)	γ^{ID} [1/(W·km)] (NFT, 10 EVs)	γ^{ID} [1/(W·km)] (SSFM)
4.2	358	21	11 GHz	2 dBm	EDFA (50 km apart)	1600 km	13.0 s	2.53 s	2.40 s	1.20 ± 0.07	1.29 ± 0.10	1.20 ± 0.04
4.3a	72	21	11 GHz	2 dBm	EDFA (50 km apart)	1600 km	0.96 s	0.50 s	0.35 s	1.29 ± 0.13	1.30 ± 0.12	1.18 ± 0.12
4.3b	126	21	11 GHz	2 dBm	EDFA (50 km apart)	1600 km	2.32 s	1.15 s	0.48 s	1.31 ± 0.10	1.27 ± 0.09	1.18 ± 0.08
4.3c	180	21	11 GHz	2 dBm	EDFA (50 km apart)	1600 km	4.80 s	1.53 s	0.57 s	1.27 ± 0.10	1.25 ± 0.09	1.20 ± 0.06
4.6a	180	21	11 GHz	-4 dBm	EDFA (50 km apart)	1600 km	3.46 s	1.51 s	0.68 s	1.37 ± 0.20	1.31 ± 0.16	1.33 ± 0.18
4.6b	996	41	42 GHz	2 dBm	EDFA (50 km apart)	1600 km	72.5 s	8.45 s	6.30 s	1.59 ± 0.14	1.48 ± 0.22	1.28 ± 0.22
-	90	21	9 GHz	-3 dBm	ideal Raman	1600 km	0.81 s	0.54 s	0.56 s	1.20 ± 0.01	1.20 ± 0.02	1.20 ± 0.01
-	90	21	9 GHz	-3 dBm	ideal Raman	4800 km	0.69 s	0.31 s	1.79 s	1.20 ± 0.04	1.21 ± 0.04	1.20 ± 0.02
4.8a	90	21	9 GHz	-3 dBm	ideal Raman	8000 km	0.74 s	0.35 s	3.15 s	1.20 ± 0.03	1.20 ± 0.05	1.20 ± 0.03
-	536	21	11 GHz	-4 dBm	ideal Raman	1600 km	21.2 s	3.67 s	2.12 s	1.19 ± 0.03	1.21 ± 0.03	1.20 ± 0.01
-	536	21	11 GHz	-4 dBm	ideal Raman	4800 km	24.3 s	4.32 s	11.0 s	1.19 ± 0.03	1.22 ± 0.05	1.20 ± 0.01
4.8b	536	21	11 GHz	-4 dBm	ideal Raman	8000 km	21.42 s	4.11 s	19.05 s	1.20 ± 0.03	1.22 ± 0.05	1.20 ± 0.01
-	268	21	11 GHz	-4 dBm	ideal Raman	1600 km	10.4 s	2.23 s	1.93 s	1.20 ± 0.04	1.21 ± 0.04	1.20 ± 0.01
-	268	21	11 GHz	-4 dBm	ideal Raman	4800 km	10.0 s	2.03 s	5.72 s	1.20 ± 0.05	1.22 ± 0.06	1.20 ± 0.02
4.8c	268	21	11 GHz	-4 dBm	ideal Raman	8000 km	9.73 s	2.40 s	10.47 s	1.22 ± 0.06	1.23 ± 0.06	1.20 ± 0.03

5

FAST AND RELIABLE DETECTION OF SIGNIFICANT SOLITONS IN SIGNALS WITH LARGE TIME-BANDWIDTH PRODUCTS

We present a fast method to calculate the significantly large solitonic components of signals with large time-bandwidth products governed by the nonlinear Schrödinger equation, for which the computation typically becomes prohibitively expensive and/or numerically unstable. We partition the full signal in both frequency and time to obtain short signals with a constant number of samples, independent of the size of the full signal. The solitons within each short signal are computed using a conventional nonlinear Fourier transform (NFT) algorithm. The partitioning in general leads to spurious solitons not present in the full signal. We therefore design an acceptance scheme that removes spurious solitons. The remaining solitons are attributed to the full signal. Solitons that are too wide to fit into the short signals cannot be detected by this approach, but since wide solitons must be of low amplitude, the significant solitons will be found. This approach only requires $\mathcal{O}(N)$ floating point operations, with N the number of signal samples. It can furthermore be applied to signals with large time-bandwidth products for which conventional NFT algorithms become unreliable or even fail. When applying our proposed method to a signal of 15,000 samples, the significant solitonic components were computed 14 times faster than when considering the whole signal, for which the conventional algorithm furthermore provided wrong results. We found that time-partitioning yields accurate results, while frequency-partitioning causes a small loss in accuracy. Combined frequency-time partitioning leads to the fastest computation, but also suffers from the same loss in accuracy as with frequency-partitioning. As time-partitioning yields a significant speed-up at nearly no loss in accuracy, we regard this as the method of choice in most practical scenarios. This chapter is an adaptation of [103].

5.1. INTRODUCTION

The nonlinear Schrödinger equation (NLSE) describes wave propagation in optical fibers [62]. The lossless NLSE is known for the existence of so-called solitons, which are localized particle-like waves [106, 107]. Even though individual solitons have a distinctive hyperbolic secant shape for the NLSE, they often cannot be determined by visual inspection of a wave packet because nonlinear interactions with other signal components temporarily change their form. In such cases, the nonlinear Fourier transform (NFT) is nevertheless able to detect them [7, 108]. Due to their stability, solitons have found many applications in fiber optics, such as fiber-optical communication [23, 28, 76, 83, 109–115], fiber parameter estimation [54, 55], the analysis of laser radiation [116–119], optical resonators [120, 121] and optical combs [122]. It is well known that the number of hidden solitons in a rectangular pulse grows with both its amplitude and duration, see e.g., [28]. This suggests that especially long and/or high power signals are typically rich in solitons. Accordingly, conventional orthogonal frequency division multiplexing (OFDM) and Nyquist-shaped communication signals with long durations and/or powers have surprisingly been observed to contain large numbers of hidden solitons [54, 86].

The numerical computation of the NFT is however a nontrivial problem due its nonlinear nature. Even though many different algorithms to compute the discrete part of the NFT that corresponds to solitons have been proposed in the literature (e.g., search methods [27, 29, 31, 123, 124], matrix methods [29, 123, 125], subdivision methods [31, 126], machine learning methods [127], phase tracking [128] and all-pass filter synthesis [129]), a wave packet with 32 solitons is still considered a challenging example [128]. For more complicated signals, the numerical algorithms often become computationally expensive and/or unreliable when using standard double precision floating point arithmetic. The efficient detection of the potentially hidden solitons in signals with large time-bandwidth products would be directly relevant for example in fiber parameter identification [54, 55], soliton-based communication systems [76, 109, 110, 115, 130–132], analysis of optical soliton gases [133, 134], optical rogue wave analysis [135], as well as in areas outside fiber optics (e.g., ocean waves [136–138]).

Since solitons are localized in both the physical and the frequency domain, it should however be possible to accurately calculate the significantly large solitons using only a limited part of the full signal. This would result in a faster and more reliable calculation. Several previous studies have indeed already observed that solitons are not influenced by other signal components that are sufficiently separated in the time or frequency domain [137, 139, 140]. If we thus take a sufficiently large frequency-time window, the significant solitons in the center of the window are expected not to be influenced by the solitons outside the window.

In this study, we propose a novel method based on these findings to quickly calculate the significant solitons within a signal. Our idea is to partition the full signal into shorter signals, and/or band-pass filtered signals. The center frequencies of the band-pass filtered signals are temporarily shifted to zero such that they can be captured with less samples. The signals with less samples are then evaluated separately. We define rectangular frequency-time windows of fixed bandwidth $\Delta\Omega^{\text{window}}$ and duration ΔT^{window}, and cover the full rectangular frequency-time domain of the original signal with such windows. The windows are allowed to overlap. The signal content in each window can

then be captured with only $M \leq N$ samples, where N is the number of samples of the full signal (proportional to the time-bandwidth product of the full signal), and M the number of samples of the windowed signal (proportional to the smaller time-bandwidth product of the frequency-time windows). The computation time for a single windowed signal will thus only depend on M and the chosen NFT method, but does not scale with the full signal length N. As we require $\mathcal{O}(\frac{N}{M})$ windows to cover the entire frequency-time domain, the computation of the significant eigenvalues in the discrete spectrum can thus be performed with only $\mathcal{O}(N)$ floating point operations (FLOPs). The proposed method does not suffer from numerical reliability issues that normally arise for signals with large time-bandwidth products due to the nonlinear nature of the NFT. Spurious solitons, which can occur as a result of the partitioning, are avoided by rejecting solitons whose support is not sufficiently contained within the window. The impact of the partitioning on the estimated soliton parameters is controlled by taking sufficiently large sub-domains of the signal. This allows us to quickly and reliably find the significant solitons of the full signal. Here, solitons are considered significant if their amplitudes are above a threshold that only depends on the chosen window sizes.

Our main contributions are as follows. We formalize the above ideas of partitioning the signal and calculating the eigenvalues associated to the significant solitons from small frequency-time windows. We derive a heuristic for the minimal required window size to be able to capture all significant solitons. By letting the windows overlap, we can ensure that all significant eigenvalues are captured, although some eigenvalues may be doubly captured. We then introduce an acceptance scheme, to keep only the most reliable version of eigenvalues that were captured in multiple windows due to their overlap, and to reject unreliable or spurious eigenvalues. Finally, we demonstrate the accuracy and speed of this frequency- and time-windowing method on several signals with large time-bandwidth products.

This chapter is organized as follows. Sec. 5.2 will recapitulate the nonlinear Fourier transform to define the eigenvalues associated with the solitonic components. Sec. 5.3 will define the window size based on the required support to capture the expected soliton content and describes an algorithm to reject spurious or inaccurate eigenvalues. Sec. 5.4 summarizes the final algorithm. Sec. 5.5 tests the described frequency- and time-windowing algorithm on signals with large time-bandwidth products. Finally, Sec. 5.6 concludes the chapter.

5.2. SOLITONS, THE NONLINEAR FOURIER TRANSFORM, AND SOLITON LOCATION

We consider the focusing nonlinear Schrödinger equation (NLSE) for signals with vanishing tails [62],

$$q_z = i q_{tt} + 2i|q|^2 q, \qquad q \xrightarrow{t\to\pm\infty} 0 \text{ sufficiently fast,} \tag{5.1}$$

with $q(t,z)$ the complex signal amplitude and i the imaginary unit. Subscripts denote partial derivatives. We consider the normalized and unitless NLSE, but for the sake of clarity we will refer to t as time and z as position, which is often the case in fiber optics

[62]. The normalization procedure itself depends on the application, but can be found in the corresponding literature. For fiber optics, see e.g. [28, Eq. 3] or [54, Eq. 5].

The focusing NLSE has soliton solutions, which are particle-like waves that retain their shape even after interacting with other solitons or dispersive waves. The defocusing NLSE, which differs from Eq. 5.1 by a sign in front of the nonlinear term, in contrast has no soliton solutions. A pure 1-soliton solution of the NLSE may be associated with a complex eigenvalue $\lambda_k = \xi_k + i\eta_k$ ($\xi_k \in \mathbb{R}$, $\eta_k > 0$), where k is an index that is used later for signals containing multiple solitons. We refer to λ_k as an eigenvalue because it arises as such from a spectral problem, as explained in the next section. The 1-soliton solution is given by [7]

$$q_k(t,z;\lambda_k,t_k^0,\psi_k^0) = \overbrace{2\eta_k \operatorname{sech}\big(2\eta_k(t-t_k^0-\underbrace{(-4\xi_k)}_{c_k}z)\big)}^{\text{envelope}} \underbrace{\times \exp\left(-2i\xi_k(t-t_k^0)-\psi_k^0-4i(\xi_k^2-\eta_k^2)z\right)}_{\text{carrier}}, \tag{5.2}$$

in which t_k^0 and ψ_k^0 are, respectively, the time and phase offset of the soliton at $z = 0$. The envelope wave speed is $c_k = -4\xi_k$. For the rest of this study we will assume $z = 0$, as we are not interested in the propagation of solitons. We note some important properties of the 1-soliton.

- **The envelope shape** is only determined by the eigenvalue height $\eta_k = \Im(\lambda_k)$: the soliton amplitude scales linearly with η_k, but the soliton also becomes narrower, as illustrated in Fig. 5.1.
- **The carrier** is only determined by the real part $\xi_k = \Re(\lambda_k)$. The linear center frequency of this soliton is given by $\omega_k = -2\xi_k$. Shifting the real part of the eigenvalue by $\Delta\xi$ thus causes a linear frequency shift of $\Delta\omega = -\Delta\xi/2$ and vice versa.
- **The energy** of the 1-soliton is given by

$$E_k = \int_{-\infty}^{\infty} \left|q_k(t,z;\lambda_k)\right|^2 \mathrm{d}t = 4\eta_k, \tag{5.3}$$

  so the highest eigenvalues (i.e., with the largest imaginary part η_k) contain the most energy.

If we consider an arbitrary signal with K solitonic components with different speeds (i.e., $\xi_k \neq \xi_j$ if $k \neq j$), the solitons will eventually separate, and evolve into a train of K 1-solitons [141, 142]:

$$q(t,z) \approx \sum_{k=1}^{K} q_k(t,z;\lambda_k,t_k^{\pm},\psi_k^{\pm}), \text{ as } z \to \pm\infty. \tag{5.4}$$

When considering an arbitrary signal, this separation will probably not have happened for practically relevant values of z. Instead, the solitons may be packed close together, and interact with one another as well as with the dispersive part of the signal. In this case, the solitons typically cannot be distinguished visually. To identify which solitons are present, the nonlinear Fourier transform is often employed.

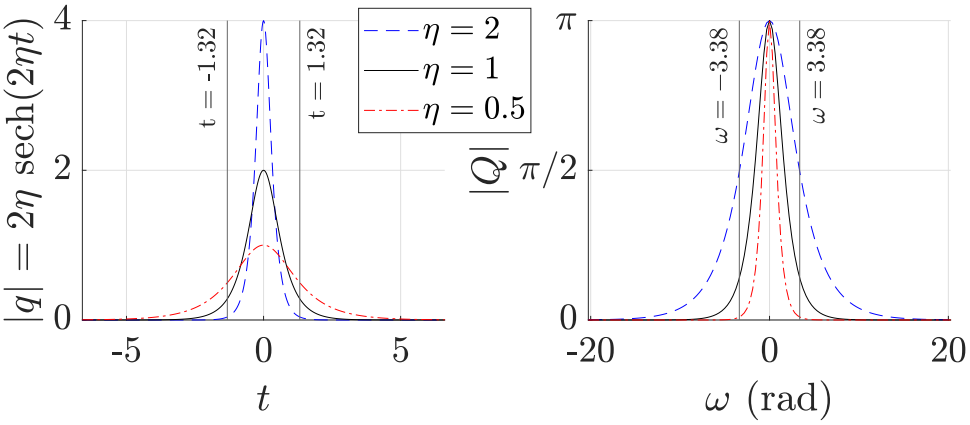


Figure 5.1: The envelope of three solitons with different η in the time- and frequency-domain, and the 99%-energy support of the unit soliton with $\eta = 1$ (vertical lines).

5

5.2.1. THE NONLINEAR FOURIER TRANSFORM

The soliton content of a signal $q(t)$ governed by the NLSE can be determined by the NFT. The NFT is defined through the Zakharov-Shabat scattering problem [108],

$$\frac{\partial}{\partial t}\begin{bmatrix}\phi^{(1)}(t,\lambda)\\ \phi^{(2)}(t,\lambda)\end{bmatrix} = \begin{bmatrix}-i\lambda & q(t)\\ -q^*(t) & i\lambda\end{bmatrix}\begin{bmatrix}\phi^{(1)}(t,\lambda)\\ \phi^{(2)}(t,\lambda)\end{bmatrix}, \tag{5.5a}$$

$$\text{s.t.}\begin{bmatrix}e^{-i\lambda t}\\ 0\end{bmatrix} \overset{t\to-\infty}{\longleftarrow} \begin{bmatrix}\phi^{(1)}(t,\lambda)\\ \phi^{(2)}(t,\lambda)\end{bmatrix} \overset{t\to+\infty}{\longrightarrow} \begin{bmatrix}a(\lambda)e^{-i\lambda t}\\ b(\lambda)e^{+i\lambda t}\end{bmatrix}, \tag{5.5b}$$

in which $\phi(t,\lambda)$ is the vector eigenfunction corresponding to the complex spectral parameter $\lambda = \xi + i\eta$, and $a(\lambda)$ and $b(\lambda)$ are the scattering coefficients that characterize the behavior at the right boundary conditions in Eq. 5.5b. The naming convention of 'eigenfunction', 'spectral parameter' and 'eigenvalue' are because the Zakharov-Shabat scattering problem may be rewritten as $L\phi = \lambda\phi$, with L a linear operator. Each λ_k related to a solitonic component in Eq. 5.4 turns out to be an eigenvalue of L, with eigenfunction ϕ_k.

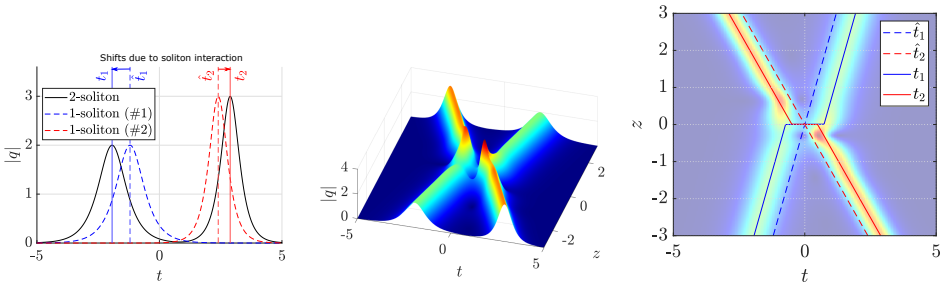
The full NFT of $q(t)$ consists of a continuous spectrum and a discrete spectrum. We define the continuous spectrum over the real axis, $\Lambda^c := \{b(\xi) : \xi \in \mathbb{R}\}$. This spectrum corresponds to the dispersive wave content of the signal. The continuous spectrum is mentioned here for completeness, but we are not further interested in it throughout this study.

We define the discrete (solitonic) spectrum using the zeros of $a(\lambda)$ in the upper half plane, $\Lambda^d := \{(\lambda_k, b(\lambda_k)) : a(\lambda_k) = 0, \Im(\lambda_k) > 0\}$, where $b_k = b(\lambda_k)$ denotes the solution of Eq. 5.5 at λ_k. Methods for the computation of $a(\lambda)$ and $b(\lambda)$ can e.g. be found in [27, 29, 123, 143]. Such methods can be utilized to find the λ_k in various ways, as was pointed out in the introduction. The computation of b_k is known to yield numerical issues without special precautions, but we overcome this by computing b_k with the NLSE-version of the adapted bidirectional algorithm from [37]. The eigenvalues $\lambda_k = \xi_k + i\eta_k$ correspond to those in the previously mentioned 1-soliton solutions. The eigenvalue λ_k defines the

shape and speed of the soliton, while the b-coefficient $b_k = b(\lambda_k)$ provides information about the soliton location and phase [7]. As a signal propagates according to the normalized NLSE in Eq. 5.1, all eigenvalues remain constant, and the b-coefficients evolve in a simple manner:

$$b(\lambda; z) = b(\lambda; 0)e^{4i\lambda^2 z} \quad \Rightarrow |b_k(z)| = |b_k(0)|e^{-8\xi\eta z}. \tag{5.6}$$

5.2.2. SOLITON LOCATION



(a) 1-soliton vs. refined soliton location.

(b) Propagation of an exact 2-soliton.

(c) Soliton location, overlaid on the 2-soliton.

Figure 5.2: Illustration of the soliton location, by comparing a 2-soliton with two 1-solitons with the same spectrum. The spectral data of the two solitons are $(\lambda_1 = -0.1 + 1i, b_1 = \exp(2i))$ and $(\lambda_2 = 0.2 + 1.5i, b_2 = \exp(0))$ at $z = 0$. **Left**: the 2-soliton at $z = -3$ compared to the two 1-solitons with the same spectral data (λ_k, b_k). The corresponding 1-soliton locations $\hat{t}_k$ and soliton locations t_k are indicated, as well as the shift due to the soliton interaction. **Middle**: the propagation of the 2-soliton. Soliton 1 overtakes solitons 2 at $z = 0$. **Right**: the 1-soliton locations and final soliton locations, overlaid on the propagating 2-soliton. Note the jump in t_1 and t_2 as result of the overtaking at $z = 0$.

At the start of this section, we noted that every 1-soliton is localized at t_k^0 (at $z = 0$), where the peak of its envelope is located. For the further analysis in this study, we assume that hidden solitons in a general signal are also localized around some soliton location t_k, similar to a 1-soliton. If we take a sufficiently large section of the full signal around that soliton location, it should be possible to detect the associated soliton accurately by computing the NFT of that section. The soliton location can usually only be observed visually in the far field, where all solitons have separated and the dispersive part becomes negligible. Therefore, we define a novel soliton location t_k for the near field when the solitons have not separated yet, suitable for the purpose in this study. The idea for t_k is to find an initial soliton location from b_k, assuming that the soliton is isolated. Then we refine that initial soliton location by taking all soliton interactions into account.

First, we assign each soliton a '1-soliton location' $\hat{t}_k$, which corresponds with the location of the soliton derived from b_k if the signal had been a 1-soliton with eigenvalue λ_k as given in Eq. 5.2 [141], [142, Eq. 1.7],

$$\hat{t}_k = \frac{\ln|b_k|}{2\eta_k}. \tag{5.7}$$

Next, we refine the 1-soliton locations by taking all the pairwise soliton interactions into account. Let two solitons be represented by (λ_1, b_1) and (λ_2, b_2). Fig. 5.2a compares the corresponding individual 1-solitons, and the 2-soliton with this discrete spectrum (i.e., a signal with exactly two eigenvalues, and zero continuous spectrum). Clearly, the locations of the peaks have shifted away from each other in the 2-soliton, despite the fact that the same b-coefficients were used. The result of the soliton interaction is that the solitons are pushed apart: the left one further to the left, and the right one further to the right.

To approximate the size of this shift, we consider the total shift between the soliton peaks as one overtakes the other, as is illustrated in Fig. 5.2b and Fig. 5.2c. When the interaction is fully completed, the total time shift for soliton k as result of the interaction with soliton j is given by [142, Eq. 1.18]:

$$\Delta t_{k,j} = \frac{1}{\eta_k} \left| \ln \left| \frac{\lambda_k - \lambda_j}{\lambda_k - \lambda_j^*} \right| \right|. \tag{5.8}$$

For the soliton location, we assume that the interaction is instantaneous, and add (resp. subtract) half the total time shift to (from) the 1-soliton location if the soliton k is right (left) of soliton j. We add half the time shift, as the interaction is symmetric, and when the right- and left-most solitons switch place (as if one overtook the other), the result would indeed be the full time shift. Fig. 5.2c shows the result of the soliton location for a 2-soliton before, during and after the interaction. Indeed the soliton locations corresponds very well to the peak locations, even close to the interaction.

For a multi-soliton signal, it is well known that the total time shift is simply the summation of all pair-wise time shifts [141]. We thus define the refined soliton location as the 1-soliton location, compensated for every soliton-pair interaction:

$$t_k = \hat{t}_k + \sum_{j=1, j\neq k}^{K} \operatorname{sign}\left(\hat{t}_k - \hat{t}_j\right) \frac{1}{2\eta_k} \left| \ln \left| \frac{\lambda_k - \lambda_j}{\lambda_k - \lambda_j^*} \right| \right|. \tag{5.9}$$

Note that we disregard the interaction with the continuous spectrum. This choice is deliberate, as it is hard to determine which dispersive components are 'left' or 'right' of the soliton, while it is only significant when much of the energy is in the continuous spectrum. We obtained good results without it, although we only considered signals with most of the energy in the discrete spectrum.

It has been observed empirically that various pulse shaping methods commonly employed for fiber-optical communications can be dominated by solitons at practically relevant transmit powers [54, 86]. For a specific class of NFT-based fiber-optic transmitters, it has even been proven that without using solitons, the signal power must approach zero as the signal duration increases [98]. Ignoring the continuous spectrum in the soliton location is therefore often a reasonable approximation.

5.3. PARTITIONING THE FREQUENCY-TIME DOMAIN INTO WINDOWS, AND ACCEPTING EIGENVALUES

In this section, we define heuristics to choose the frequency-time window size to capture the significant solitons accurately. We will find, consistently with Eq. 5.2, that the signif-

icant (i.e., higher amplitude and η_k, respectively) solitons occupy a broad bandwidth but short duration, while solitons with lower eigenvalues occupy longer durations, but shorter bandwidths. To ensure that the windows can capture all of the significant solitons, we first estimate the mean eigenvalue height using the mean energy density, and adjust the window size accordingly. Next, we allow the windows to overlap to ensure that all higher solitons are captured. Finally, we filter out unreliable and/or doubly-detected solitons with an eigenvalue acceptance scheme.

5.3.1. ESTIMATION OF THE MEAN EIGENVALUE HEIGHT

We wish to choose a window size such that all significant solitons can be captured. To do so, we start by estimating a 'mean' eigenvalue height η^{mean} of the solitons in the signal, as the eigenvalue height will be the decisive factor for the required bandwidth and duration of the windows. To estimate the mean eigenvalue height, we define a mean energy density ρ^{mean} over the considered frequency-time domain, and then estimate the mean eigenvalue height of the signal by comparing the mean energy density of the signal to the mean energy density of a 1-soliton.

First, we define the mean energy density ρ^{mean} of a 1-soliton as the total energy, divided by the time-bandwidth product that captures most of the energy. Let the 99%-energy ($p = 0.99$) bandwidth (resp. duration) be defined as the smallest bandwidth (resp. duration) required to capture a fraction p of the total signal energy. For a 1-soliton q_k with $\xi_k = 0$ and $t_k = 0$ as shown in Fig. 5.1, the support bandwidth and duration are

$$\int_{-\frac{\Delta T_k}{2}}^{\frac{\Delta T_k}{2}} \left|q_k(t)\right|^2 \mathrm{d}t = \underbrace{pE_k}_{0.99(4\eta_k)} \qquad \Rightarrow \Delta T_k = \frac{2.64}{\eta_k}, \tag{5.10a}$$

$$\frac{1}{2\pi}\int_{-\frac{\Delta\Omega_k}{2}}^{\frac{\Delta\Omega_k}{2}} |Q_k(\omega)|^2 \,\mathrm{d}\omega = pE_k \qquad \Rightarrow \Delta\Omega_k = 6.75\eta_k, \tag{5.10b}$$

where $Q_k(\omega) = \int_{-\infty}^{\infty} q_k(t)e^{-i\omega t}\,\mathrm{d}t$, i.e., the Fourier transform of $q_k(t)$. The time-bandwidth product $\Delta\Omega_k\Delta T_k = 17.82$ rad of a 1-soliton does not depend on η_k because they are related to each other through time- and amplitude scalings, which leave the time-bandwidth product invariant. We then define the mean energy density of a 1-soliton with eigenvalue height η_k as its total energy ($E_k = 4\eta_k$), divided by its time-bandwidth product:

$$\rho_k^{\text{mean}} = \frac{E_k}{\Delta T_k \Delta\Omega_k} = \frac{4\eta_k}{17.82} = 0.224\eta_k. \tag{5.11}$$

Next, we calculate the mean energy density of the signal too as the total energy divided by its 99%-energy bandwidth $\Delta\Omega$ and 99%-energy duration ΔT. Finally we estimate that the 'mean' soliton has the same mean energy density as the signal ρ_q^{mean},

$$\rho_q^{\text{mean}} = 0.224\eta^{\text{mean}} \Rightarrow \eta^{\text{mean}} = \frac{\rho_q^{\text{mean}}}{0.224} = \frac{E_q}{0.244\Delta\Omega\,\Delta T}, \tag{5.12}$$

with η^{mean} the 'mean' eigenvalue height. Note that the mean eigenvalue height only roughly indicates a center for the final eigenvalue cloud. It does not correspond with the actual mean height, as low eigenvalues usually appear more frequently than high ones.

5.3.2. CHOOSING THE WINDOW FREQUENCY-TIME SIZE

After establishing the estimated mean eigenvalue height, we will choose the frequency-time window sufficiently large to capture all eigenvalues which have η_k around the estimated eigenvalue height. We will first define a significant frequency-time support of a soliton with eigenvalue λ, and then design the frequency-time window large enough such that it can contain the significant support of all significant solitons of interest. We finally create the window size based on the expected mean eigenvalue height defined in Eq. 5.12, and the height range of eigenvalues that we want to capture.

We assume that a soliton with the mean eigenvalue height η^{mean} occupies a rectangular frequency-time support of $\Delta\Omega \times \Delta T$, with $\Delta\Omega = 6.74\eta^{\text{mean}}$ and $\Delta T = 2.64/\eta^{\text{mean}}$, centered around its soliton location t_k and center frequency ($\omega_k = -2\xi_k$). To capture higher eigenvalues we require a larger bandwidth $\Delta\Omega$, and for lower eigenvalues a longer duration ΔT. To capture a wide range of solitons, we chose the window bandwidth as $c_\Omega = 16$ times the mean-eigenvalue bandwidth, and the window duration as $c_T = 4$ times the mean-eigenvalue duration:

$$\Delta\Omega^{\text{window}} \times \Delta T^{\text{window}} = \underbrace{c_\Omega}_{=16}\left(6.74\eta^{\text{mean}}\right) \times \underbrace{c_T}_{=4}\left(2.64/\eta^{\text{mean}}\right). \tag{5.13}$$

We made the bandwidth 16 times larger to ensure that all high eigenvalues are captured. We made the duration only four times larger because this mainly allowed us to capture more low eigenvalues, in which we are less interested. This window size gave us accurate results for signals with approximate uniform energy distribution of the occupied frequency-time domain, but of course the window size may be tailored to specific applications. When only time-windowing is desired, simply set the frequency-window size equal to the full occupied bandwidth $\Delta\Omega^{\text{window}} = \infty$, and vice versa for only frequency-windowing.

5.3.3. DIVIDING THE DOMAIN AND WINDOW OVERLAP

After deciding on the window size, we will cover the entire occupied frequency-time domain with windows of this size. The most straightforward method would be to use as few windows as possible while still covering the entire domain. However, we could then miss several high eigenvalues with soliton location t_k or center frequency ω_k close to the window edges. Part of the soliton support would then fall outside the window, and the detected eigenvalue could be significantly distorted as a result.

To ensure that all higher eigenvalues are captured well by at least one window, we let all neighboring windows overlap for a fraction of at least $0 \leq R < 1$ for both frequency- and time-windowing. After creating the equispaced windows with this minimum overlap fraction, the actual time overlap fraction $R_T \geq R$ and frequency overlap fraction $R_\Omega \geq R$ are often slightly larger because the number of windows is an integer number. In the worst-case scenario for time-windowing (resp. frequency-windowing), a soliton has its t_k (resp. ω_k) exactly in the middle of the overlapping region, such that the largest soliton support fully contained in either window is $R_T\Delta T^{\text{window}}$ (resp. $R_\Omega\Delta\Omega^{\text{window}}$) units. Following Eq. 5.10 and Eq. 5.13, the lowest η_k thus captured within this time support and

the highest η_k captured within this frequency support are

$$\eta_k \in [\eta^{\min}, \eta^{\max}] = [\tfrac{1}{Rc_T}\eta^{\text{mean}}, Rc_\Omega \eta^{\text{mean}}]. \tag{5.14}$$

Note that when using only time-windowing (resp. frequency-windowing), only the lower (resp. upper) limit is relevant. Lower values of R will narrow down the range of solitons in Eq. 5.14 we can capture reliably. Choosing higher values of R causes windows to overlap more, requiring more windows and thus more computation time. A well-balanced overlap fraction is thus around $R = 0.5$. Instead of choosing even higher $R > 0.5$ and thus increasing the number of required windows, c_Ω or c_T (i.e., the window size) can also be increased to capture a wider range, which is often computationally faster.

The full domain is thus covered as follows:

1. Choose a window size $\Delta\Omega^{\text{window}} \times \Delta T^{\text{window}}$ and overlap fraction $0 \leq R < 1$.
2. Cover the full frequency domain Ω with equispaced frequency sections of size $\Delta\Omega^{\text{window}}$ such that consecutive sections overlap for a fraction of at least R. This results in N_Ω frequency sections $\Omega^{(n)}$, $1 \leq n \leq N_\Omega$. If $\Delta\Omega^{\text{window}} > \Delta\Omega$, set $\Delta\Omega^{\text{window}} = \infty$, and $N_\Omega = 1$.
3. Do the same for the time domain T with equispaced sections of size ΔT^{window}, resulting in M_T time sections $T^{(m)}$, $1 \leq m \leq M_T$.
4. Define $N_\Omega \times M_T$ frequency-time windows $\Omega^{(n)} \times T^{(m)}$ as the cross product of the frequency section and time sections.

This process is illustrated on the left in Fig. 5.4 with slight offsets to improve visibility.

We note here that we only cover the 99%-energy bandwidth and 99%-energy duration of the full signal $\Delta\Omega \times \Delta T$, and thus cut away parts of the signal. We do so to be able to assign any signal a finite bandwidth and duration to avoid wasting computation time analyzing any slowly decaying tails of the signal (either in frequency or time domain), which generally do not contain high solitons. However, Ω and T may also be be set manually if clear bounds for the signal are known.

5.3.4. CREATING WINDOWED SIGNALS

After defining the frequency-time windows, we extract a short signal for each window. First, we apply an ideal band-pass filter to the full signal to remove all frequency content outside of $\Omega^{(n)}$ by calculating the Fourier transform of the full signal, and setting all frequencies components outside $\Omega^{(n)}$ to zero. Next, we temporarily shift the center frequency ω_n (middle of $\Omega^{(n)}$) to zero for the filtered signal, $q(t) \to q(t)e^{-i\omega_n t}$ (i.e., $Q(\omega) \to Q(\omega - \omega_n)$), such that the filtered frequency content can be captured without aliasing with much fewer samples than before the center shift, namely using sampling time $\Delta t < \frac{2\pi}{\Delta\Omega^{\text{window}}}$. The center frequency is added back later to the detected eigenvalues, where we will use that linear frequency shifting a signal results in a linear shift in the real part ξ_k of each eigenvalue: $q(t) \to q(t)e^{-i\omega_n t} \Rightarrow \xi_k \to \xi_k + \omega_n/2$ [28, p. 4319, D6, frequency shift property].

After frequency-windowing, the filtered signal is time-windowed by only keeping those samples within the time-window $T^{(m)}$, resulting in a short signal containing the signal

content within the frequency-time window $\Omega^{(n)} \times T^{(m)}$. We used rectangular filters/time-windows in our implementation as the impact of general linear filters and time-windows on the nonlinear Fourier spectrum is not well-understood.

5.3.5. EIGENVALUE ACCEPTANCE

After all the frequency- and time-windowed signals have been created, the eigenvalues of each windowed signal are computed using the NFT and the center frequency is added back to the real part of the eigenvalues again, $\lambda_k \to \lambda_k - \omega_n/2$. The eigenvalues from the windowed signal are estimations for the eigenvalues of the full signal. However, many of the eigenvalues from the full signal will have been affected by the windowing process, and we wish to only keep the eigenvalues that are also present in the full signal. We therefore introduce an acceptance criterion for the eigenvalues of the windowed signal.

The idea for the eigenvalue acceptance is that the soliton support (see Eq. 5.10) should lie entirely within the window. The required time-support of an eigenvalue grows with $1/\eta_k$, so the lowest eigenvalues will require a broader time-support than the window duration. We should therefore discard the too-low eigenvalues by default. Similarly, the highest eigenvalues require a large frequency-support. We set the acceptance height range for the detected eigenvalue height equal to the earlier defined eigenvalue capture region from Eq. 5.14, and reject all eigenvalues outside this height range. The frequency-windows should be chosen spaciously, so that it is nearly guaranteed that the highest eigenvalues are captured. An additional rough measure for the highest eigenvalue present (based on the 1-soliton in Eq. 5.2) is the maximum absolute wave height $\max_k \eta_k \propto |q|/2$. Choosing the frequency-window size such that $\eta^{\max} \gg \max_t |q(t)|/2$ ensured that the highest solitons were captured for our data.

After rejecting all eigenvalues below the acceptance height range, it may still occur that several of the remaining eigenvalues are too close to the window edge, such that their significant support still lies partially outside the considered window. These eigenvalues should thus be rejected. To do so, we first determine the soliton location t_k from Eq. 5.9 using the (λ_k, b_k) of every soliton above the acceptance height of the considered window. Next, we assign each window an associated acceptance region as defined below, which corresponds to the window minus a narrow strip near the window edges. The idea is that we only keep those eigenvalues from the windowed signal that also have their frequency-time location $(\omega_k = -\xi_k/2, t_k)$ within the acceptance region of that window.

We initially define the acceptance region $\Omega_+^{(n)} \times T_+^{(m)}$ of a window as the part of the full frequency-time domain closer to the center of the window than to any other window center, as shown in Fig. 5.4. The result is that all overlapping regions are exactly split into half and divided equally over the two overlapping windows. For the overlap fraction R_Ω (resp. R_T), the initial frequency (resp. time) acceptance region $\Omega_+^{(n)} = [\Omega_{+,\mathrm{l}}^{(n)}, \Omega_{+,\mathrm{r}}^{(n)}]$ (resp. $T_+^{(m)} = [T_{+,\mathrm{l}}^{(m)}, T_{+,\mathrm{r}}^{(m)}]$) is thus a fraction $R_\Omega/2$ (resp. $R_T/2$) smaller on both sides than the frequency-window $[\Omega_\mathrm{l}^{(n)}, \Omega_\mathrm{r}^{(n)}]$ (resp. time-window $[T^{(m)})_\mathrm{l}, T_\mathrm{r}^{(m)}]$). Here, the subscript l is short for 'left' and r is short for 'right'. We thus get $[\Omega_{+,\mathrm{l}}^{(n)}, \Omega_{+,\mathrm{r}}^{(n)}] = [\Omega_\mathrm{l}^{(n)} + \frac{R_\Omega \Delta\Omega^{\text{window}}}{2}, \Omega_\mathrm{r}^{(n)} - \frac{R_\Omega \Delta\Omega^{\text{window}}}{2}]$ and $[T_{+,\mathrm{l}}^{(m)}, T_{+,\mathrm{r}}^{(m)}] = [T_\mathrm{l}^{(m)} + \frac{R_T \Delta T^{\text{window}}}{2}, T_\mathrm{r}^{(m)} - \frac{R_T \Delta T^{\text{window}}}{2}]$. The left edge of the leftmost frequency-windows and time-windows and right edge of the rightmost win-

dows are not shortened, as those regions are not captured better by any other window. We thus partition the full domain into disjoint acceptance regions, which usually prevents eigenvalues from being detected twice by two neighboring overlapping windows.

However, we found that it may sometimes occur that eigenvalues are entirely missed this way. If a soliton has true soliton frequency-time location $(\omega_k = -\xi_k/2, t_k)$, its observed time or frequency location can slightly vary depending on the used window. If may therefore occur that t_k is on the boundary between two acceptance regions, and that the detected soliton location from the left window is $t_k + \epsilon$, and for the right window $t_k - \epsilon$. It would then fall outside both windows, and be rejected by both acceptance regions.

To avoid doubly-rejecting eigenvalues that are on the boundaries of the initial acceptance regions, we slightly extend the acceptance regions in both the time and frequency domain by a small fraction ε of the window size, as illustrated in Fig. 5.3. The final acceptance regions $\Omega_+^{(n)} \times T_+^{(m)}$ thus become

$$[\Omega_{+,\mathrm{l}}^{(n)}, \Omega_{+,\mathrm{r}}^{(n)}] = [\Omega_{\mathrm{l}}^{(n)} + (\tfrac{R_\Omega}{2} - \varepsilon)\Delta\Omega^{\text{window}}, \Omega_{\mathrm{r}}^{(n)} - (\tfrac{R_\Omega}{2} - \varepsilon)\Delta\Omega^{\text{window}}], \tag{5.15a}$$

$$[T_{+,\mathrm{l}}^{(m)}, T_{+,\mathrm{r}}^{(m)}] = [T_{\mathrm{l}}^{(m)} + (\tfrac{R_T}{2} - \varepsilon)\Delta T^{\text{window}}, T_{\mathrm{r}}^{(m)} - (\tfrac{R_T}{2} - \varepsilon)\Delta T^{\text{window}}], \tag{5.15b}$$

$$\text{using } \Omega_{\mathrm{l}}^{(1)} = T_{\mathrm{l}}^{(1)} = -\infty, \quad \Omega_{\mathrm{r}}^{(N)} = T_{\mathrm{r}}^{(M)} = +\infty.$$

A typical value for the acceptance region extension fraction is $\varepsilon = 0.05$, but should be much smaller than $R/2$, else the acceptance region will become larger than the window itself. While the overlap fraction R itself does not play an active role in the acceptance, we emphasize that it should be chosen sufficiently large to facilitate the capturing of eigenvalues close to window boundaries.

Higher extension fractions ε increase the chance that all eigenvalues are captured, but also increase the chance that some eigenvalues are captured twice. In choosing the extension fraction, we prioritized ensuring that all high eigenvalues were detected, which caused some eigenvalues to be detected doubly. Higher eigenvalues that are detected twice can often be easily distinguished from the case with two different solitons, as for twice-detected solitons both their λ_k and their t_k are near-identical. Had those λ_k belonged to two different solitons, then they would have had very different t_k as it is very hard for the peaks of two different solitons with near-identical λ_k to get close (see Eq. 5.9 and Fig. 5.2). However, solitons with lower eigenvalues that were captured twice are captured less accurately, and thus it is harder to filter low twice-captured solitons with certainty. The focus in this study is on capturing all solitons, so we simply accept it when some solitons are captured twice.

5.4. SUMMARY OF THE FREQUENCY- AND TIME-WINDOWING NFT ALGORITHM

For completeness, we summarize the steps of the full algorithm. The partitioning of the frequency-time domain into windows is described in Alg. 2, while the windowing of

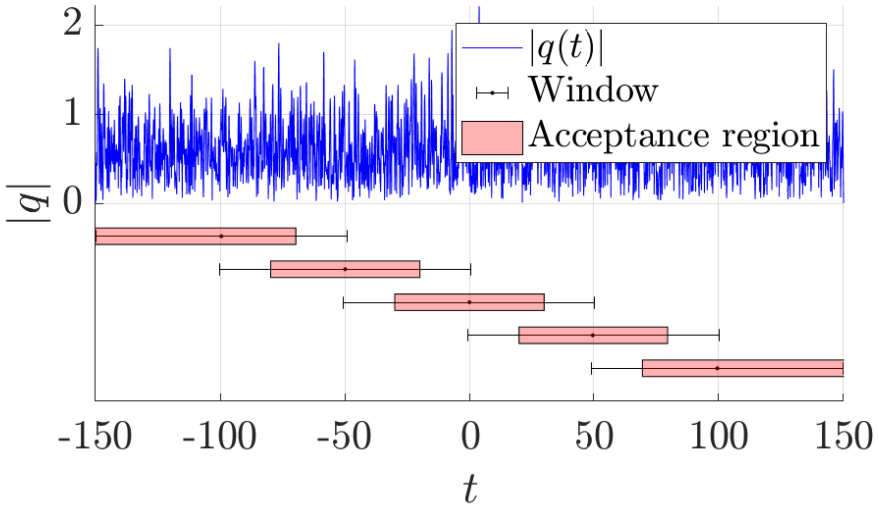


Figure 5.3: An exemplary signal of duration $\Delta T = 300$, with its associated time windows (shown below the signal) using $R = 0.5$ (neighboring windows overlap at least 50%), and the corresponding acceptance regions using $\varepsilon = 0.05$. For $\varepsilon = 0$, the acceptance regions would not overlap, and would exactly partition the time domain.

the full signal and the calculation of the higher eigenvalues from the windowed signals is described in Alg. 3. For the full estimation of the higher eigenvalues, we first apply Alg. 2, followed by Alg. 3. Alg. 2 first estimates a representative 'mean' soliton height, and chooses the window size accordingly. Next, the occupied domain of the signal is covered with overlapping windows of this size. After defining the windows, Alg. 3 calculates the higher eigenvalue content in each window. For every window, the full signal is band-pass filtered to the window frequency domain, the center frequency is shifted to zero, and the resulting signal is finally truncated to the window time domain. The eigenvalues of the resulting windowed signal are calculated with the NFT. Eigenvalues are accepted if the eigenvalue height is within the reliable height range, and if the soliton frequency-time location (ω_k, t_k) is within the acceptance region. Finally, all accepted eigenvalues are combined for an approximation of the higher eigenvalues of the full signal.

5.5. RESULTS

In this section, we investigate the accuracy and speed for the proposed combined frequency- and time-windowing NFT algorithm, as well as for only time-windowing and only frequency-windowing. First, we test the performance of the algorithm on a bandwidth-limited random signal, sampled at oversampling rate $s = 3$, resulting in 3000 samples. This is the signal with the largest time-bandwidth product that we could produce such that the full NFT could still be reliably calculated with the software library *FNFT* [35] (commit 9756b3, default settings with *4split4B*, with sub-sampling disabled as the considered

signals are already sampled close to Nyquist frequency). This method requires $\mathcal{O}(N^2)$ FLOPs to calculate the discrete spectrum of a signal with N samples. Second, we investigate the accuracy for a very long signal. For this long signal, the NFT of the full signal can

Algorithm 2: Creating frequency-time windows

Input:

- Signal $q(t)$ with finite time domain D.
- Window time-broadening factor c_T and window frequency-broadening factor c_Ω (suggested: $c_T = 4$, $c_\Omega = 16$). Set $c_T = \infty$ if not time-windowing, and $c_\Omega = \infty$ if not frequency-windowing.
- Window overlap fraction R (suggested: $R = 0.5$).
- Acceptance region extension fraction ε (suggested: $\varepsilon = 0.05$).

Output:

- A set of frequency-time windows $\Omega^{(n)} \times T^{(m)}$, that cover the significant frequency-time domain of $q(t)$, and their associated acceptance regions $\Omega_+^{(n)} \times T_+^{(m)}$.

- Set $\Omega_q = [\Omega_\mathrm{l}, \Omega_\mathrm{r}]$, $T_q = [T_\mathrm{l}, T_\mathrm{r}]$ as the smallest frequency and time domain containing 99% of the signal energy $E_q = \int_D |q(t)|^2 \,\mathrm{d}t = \frac{1}{2\pi}\int_{-\infty}^{\infty} |Q(\omega)|^2 \,\mathrm{d}\omega$;
- Set $\Delta\Omega = \Omega_\mathrm{r} - \Omega_\mathrm{l}$, $\Delta T = T_\mathrm{r} - T_\mathrm{l}$;
- Set $\eta^{\mathrm{mean}} = \frac{E_q}{0.224\Delta\Omega\Delta T}$, (see Eq. 5.12);
- Set $\Delta\Omega^{\mathrm{window}} = \min(c_\Omega 2.64/\eta^{\mathrm{mean}}, \Delta\Omega)$, and $\Delta T^{\mathrm{window}} = \min(c_T 6.74\eta^{\mathrm{mean}}, \Delta T)$, (See Eq. 5.13);
- Set $N_\Omega = 1 + \lceil \frac{\Delta T - \Delta T^{\mathrm{window}}}{(1-R)\Delta T^{\mathrm{window}}} \rceil$, $M_T = 1 + \lceil \frac{\Delta\Omega - \Delta\Omega^{\mathrm{window}}}{(1-R)\Delta\Omega^{\mathrm{window}}} \rceil$, (the number of required frequency- and time-windows respectively);
- Create $N_\Omega \times M_T$ frequency-time windows $\Omega^{(n)} \times T^{(m)}$,
$n = 1, \ldots, N_\Omega$, $m = 1, \ldots, M_T$, where
$\Omega^{(n)} = [\Omega_n, \Omega_n + \Delta\Omega^{\mathrm{window}}]$ (or $\Omega^{(1)} = \mathbb{R}$, if $N_\Omega = 1$),
with $\Omega_n = \Omega_\mathrm{l} + (n-1)\frac{\Delta\Omega - \Delta\Omega^{\mathrm{window}}}{\max(N_\Omega - 1, 1)}$, and
$T^{(m)} = [T_m, T_m + \Delta T^{\mathrm{window}}]$ (or $T^{(1)} = \mathbb{R}$ if $M_T = 1$),
with $T_m = T_\mathrm{l} + (m-1)\frac{\Delta T - \Delta T^{\mathrm{window}}}{\max(M_T - 1, 1)}$;
- Set $R_T = \frac{T_\mathrm{r}^{(1)} - T_\mathrm{l}^{(2)}}{\Delta T^{\mathrm{window}}}$, $R_\Omega = \frac{\Omega_\mathrm{r}^{(1)} - \Omega_\mathrm{l}^{(2)}}{\Delta\Omega^{\mathrm{window}}}$ (actual overlap)
- Create $N_\Omega \times M_T$ acceptance regions $\Omega_+^{(m)} \times T_+^{(n)} :=$
$[\Omega_\mathrm{l}^{(n)} + (\frac{R_\Omega}{2} - \varepsilon)\Delta\Omega^{\mathrm{window}}, \Omega_\mathrm{r}^{(n)} - (\frac{R_\Omega}{2} - \varepsilon)\Delta\Omega^{\mathrm{window}}] \times$
$[T_\mathrm{l}^{(m)} + (\frac{R_T}{2} - \varepsilon)\Delta T^{\mathrm{window}}, T_\mathrm{r}^{(m)} - (\frac{R_T}{2} - \varepsilon)\Delta T^{\mathrm{window}}]$, but with $\Omega_\mathrm{l}^{(1)} = T_\mathrm{l}^{(1)} = -\infty$,
$\Omega_\mathrm{r}^{(N)} = T_\mathrm{r}^{(M)} = +\infty$;
- Return the windows $\Omega^{(n)} \times T^{(m)}$ and associated acceptance regions $\Omega_+^{(n)} \times T_+^{(m)}$.

no longer be computed with the mentioned settings of *FNFT*. To validate the accuracy of the windowing NFT, we instead calculate the reference eigenvalues from the full signal by Newton-refining (using the method in [27]) for a dense grid of initial guesses with high precision arithmetic. Third, we quantitatively measure the computation times for the frequency- and/or time-windowing NFT. As combined frequency- and time-windowing uses windows with the smallest time-bandwidth product, this method should yield the largest speed-up.

Algorithm 3: NFT on windowed signals

Input:

- Signal $q(t)$.
- Desired oversampling rate s (suggested: $s = 3$).
- $N_\Omega \times M_T$ frequency-time windows $\Omega^{(n)} \times T^{(m)}$ and associated acceptance regions $\Omega_+^{(n)} \times T_+^{(m)}$.

Output:

- An approximate set of higher eigenvalues $\Lambda^{\mathrm{d,high}}$

- Set $Q(\omega) = \mathscr{F}\{q(t)\}$;
- Initialize the set of high eigenvalues $\Lambda^{\mathrm{d,high}} = \emptyset$;
for $n = 1, \ldots, N_\Omega$ **do**
  - Set $Q^{(n)}(\omega) = Q(\omega)$ for $\omega \in \Omega^{(n)}$, and $Q^{(n)}(\omega) = 0$ otherwise (ideal band-pass filtering);
  - Set $\omega_n = \frac{\Omega_\mathrm{r}^{(n)} + \Omega_\mathrm{l}^{(n)}}{2}$ (center frequency), and $\xi^n = -\omega_n/2$ (nonlinear center frequency);
  - Set $q^{(n)}(t) = \mathscr{F}^{-1}\{Q^{(n)}(\omega - \omega_n)\}$ (center shift, inverse Fourier transform);
  for $m = 1, \ldots, M_T$ **do**
    - Set $q^{n,m}(t) = q^{(n)}(t)$, $q^{n,m}(t \notin T^{(m)}) = 0$;
    - Set $\{\lambda_k, b_k\}_{k=1}^{K^{n,m}} \xleftarrow{\text{Sec. 5.2}} \mathrm{NFT}\{q^{n,m}(t)\}$ (sample $q^{n,m}(t)$ at s times the Nyquist frequency, sample time $\Delta t = \frac{1}{s} \frac{2\pi}{\min(\Delta\Omega, \Delta\Omega^{\mathrm{window}})}$);
    - Set $\{t_k\}_{k=1}^{K^{n,m}} \xleftarrow{\text{Eq. 5.9}} \{b_k\}_{k=1}^{K^{n,m}}$;
    - Accept only the eigenvalues with both $\eta_k \in [\frac{1}{R_T c_T}\eta^{\mathrm{mean}}, R_\Omega c_\Omega \eta^{\mathrm{mean}}]$ (Eq. 5.14), and $(\omega_k = -2\xi_k, t_k)$ within the acceptance region $\Omega_+^{(n)} \times T_+^{(m)}$;
    - Add the accepted eigenvalues λ_k to $\Lambda^{\mathrm{d,high}}$;
  end
end
- Return $\Lambda^{\mathrm{d,high}}$.

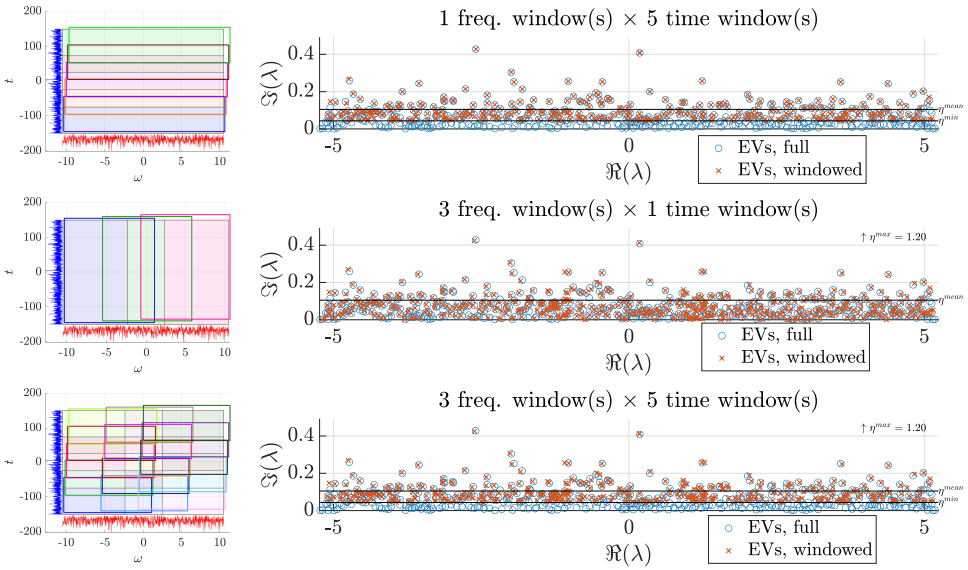


Figure 5.4: A comparison between the eigenvalues found by the NFT of a full 3000 sample signal and the corresponding windowed signals using Alg. 2 and Alg. 3. Three cases were considered for windowing: only time-windowing (top), only frequency-windowing (middle), and combined frequency- and time-windowing (bottom). The left column shows the used windows (windows are slightly offset for visibility). The bandwidth limited noise signal $|q(t)|$ is shown in blue at the left of the domain and $|Q(\omega)|$ below in red for illustration purposes (not to scale). Both $q(t)$ and $Q(\omega)$ quickly decay outside the shown domain $\Omega \times T = [-10.47, 10.47]\,\text{rad} \times [-150, 150]$. The right column shows the full eigenvalue spectrum and the accepted eigenvalues from the windowed signals.

5.5.1. RESULTS FOR A RANDOM 3000-SAMPLE SIGNAL

We first validate the frequency- and time-windowing NFT on a signal with 3000 samples, generated from ideally low-pass filtered, complex, circularly symmetric, zero-mean, white Gaussian noise. This signal has the following properties: time domain $T = [-150, 150]$, sample time $\Delta t = 0.1$, a maximum bandwidth of $[-10\pi, +10\pi]$ rad, and occupied bandwidth = $[-10\pi/3, +10\pi/3] = [-10.5, 10.5]$ rad (ideal low-pass filtered, oversampling rate $s = 3$). While *FNFT* in general works well for at this oversampling rate, we found that it occasionally still missed some eigenvalues in both the full NFT and in the windowing NFT. To ensure that we only study the effect of the windowing, we also redid the full and windowing NFTs after upsampling the signal by a factor two using band-limited interpolation, and used those results for Fig. 5.4. The amplitude of the signal was scaled such that the mean energy density was 0.0224, which corresponds to an expected mean eigenvalue height of $\eta^{\text{mean}} = 0.1$ according to Eq. 5.12. The corresponding 1-soliton has a frequency-time support of 0.68 rad × 26.

To be able to capture the largest solitons with broad bandwidths, we choose the window bandwidth 16 times the mean eigenvalue bandwidth ($c_\Omega = 16$), and the duration as four times the mean eigenvalue duration ($c_T = 4$). The resulting window size was

11.3 rad × 101. Imposing an overlap fraction of $R = 0.5$, we require three frequency-windows $\Omega^{(n)}$ and five time-windows $T^{(m)}$. We thus obtain 15 windows $\Omega^{(n)} \times T^{(m)}$, with $\Omega^{(n)} \in \{[-10.5, 0.8], [-5.4, 5.4], [-0.8, 10.5]\}$, and $T^{(m)} \in \{[-150, -49], [-100, 1], [-50, 50], [-1, 100], [49, 150]\}$. The acceptance region extension fraction was chosen as $\varepsilon = 0.05$. The time windows and acceptance regions are also shown in Fig. 5.3.

We first consider the cases with only time-windowing and only frequency-windowing to observe the individual effects on the detected eigenvalues. Then, we apply both frequency- and time-windowing, which will yield the largest speed-up. The resulting time-windowing, frequency-windowing and combined frequency- and time-windowing are shown on the left in Fig. 5.4 (windows are slightly offset for visibility).

RESULTS USING ONLY TIME-WINDOWING

In the top row of Fig. 5.4, the eigenvalues resulting from only time-windowing are shown. We observe that the higher eigenvalues above η^{mean} correspond very well to the eigenvalues from the full signal. Even the lower eigenvalues between η^{mean} and η^{min} seem to be rather accurate.

Upon closer inspection, it turns out that some eigenvalues have been captured doubly by neighboring windows. However, both eigenvalues are so close that they are not visibly distinguishable. As explained, this is due to the acceptance region extension fraction $\varepsilon = 0.05$. We also investigated $\varepsilon = 0$, and $\varepsilon = 0.02$, which respectively resulted in eight and three missing high eigenvalues above η^{mean}.

We thus find that time-windowing yields very accurate results for the high eigenvalues above η^{mean}, and fairly good results for the eigenvalues between η^{min} and η^{mean}. The acceptance region extension fraction allows a trade-off between increasing the chance to capture all eigenvalues versus capturing some eigenvalues doubly.

RESULTS USING ONLY FREQUENCY-WINDOWING

The windows and detected eigenvalues using only frequency-windowing are shown in the second row of Fig. 5.4. We observe that the eigenvalues are detected rather accurately, but not as accurately as with time-windowing. The high eigenvalues above η^{mean} are within 2% of their correct eigenvalue from the full signal. Although frequency-windowing may cut away a part of the frequency support of in particular the highest eigenvalues, this effect seems to be limited. This is likely due to the large frequency-broadening factor $c_\Omega = 16$. This hypothesis is supported by the fact that $\eta^{\text{max}} = 1.20$ is several times larger than the highest eigenvalue. We also observe that a few eigenvalues are also captured twice due to the overlapping acceptance regions. The doubly captured eigenvalues can in several cases be distinguished visually, indicating that the eigenvalues are less accurately captured than for only time-windowing.

Below η^{mean}, many of the lower eigenvalues seem to be captured with similar accuracy as the higher ones. However, we also observe that many of the lower eigenvalues are missed, but we found that many of these were missed because the oversampling factor s was too low for the used *FNFT* method, and not due to the frequency windowing itself: upon upsampling the signal using band-limited Fourier interpolation to twice as many samples, many of the missing low eigenvalues were found again. Despite the fact that

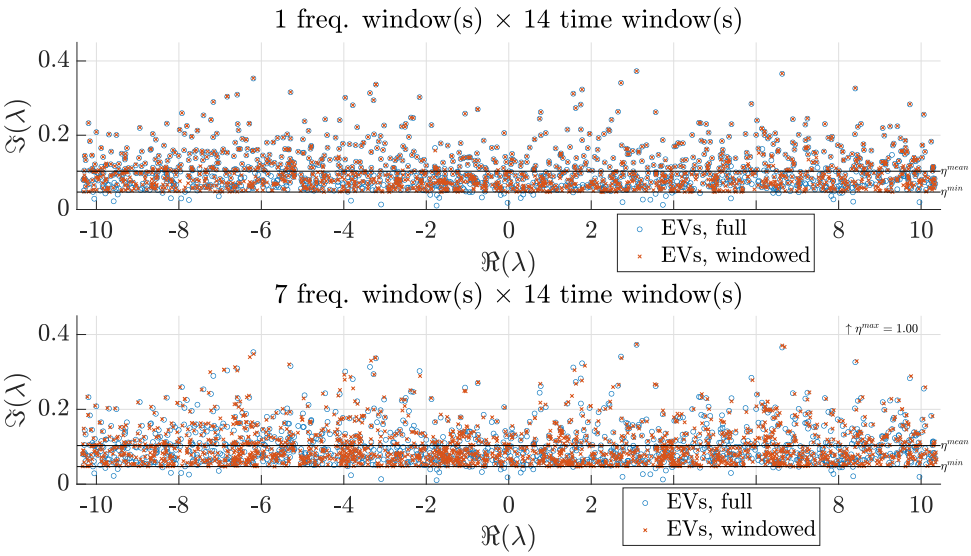


Figure 5.5: The eigenvalues of a signal with 15,000 samples (at oversampling $s = 3$), detected from windowed signals, and the higher eigenvalues detected from the full signal by Newton refining a dense grid of initial guesses. For both the top and bottom case, we used five time-windows. However, the top case used only one frequency-window (i.e., the full bandwidth), whereas the lower figure used seven frequency-windows.

there is no lower bound $\eta^{\min}$, it seems that frequency-windowing thus also affects the lower eigenvalues.

RESULTS USING COMBINED FREQUENCY- AND TIME-WINDOWING

Finally, we combined frequency- and time-windowing, as described in Alg. 3. The results are shown in the bottom of Fig. 5.4. We observe that the eigenvalues from combined frequency- and time-windowed signals are very similar to those from the only frequency-windowed signals. It thus seems that the time-windowing does not incur a significant additional error on top of the inaccuracies that the frequency-windowing is incurring.

In conclusion, the combined frequency- and time-windowing method yields quite accurate estimates with less than 2% error for the higher eigenvalues for the chosen example. Most of the error was caused by the frequency-windowing, while the time-windowing did not cause a visible error at all in the higher detected eigenvalues above η^{mean}.

In conclusion, we observe that time-windowing hardly infers any loss in accuracy, and may thus be used to speed up the calculation of the higher solitons. Furthermore, if a slight loss of accuracy in the eigenvalues is acceptable, the combined frequency- and time-windowing NFT can be used to obtain an even greater speed-up.

5.5.2. RESULTS FOR A RANDOM 15,000 SAMPLE SIGNAL

After considering the 3000 sample signal, we are interested how the windowing algorithm performs for signals with very large time-bandwidth products. We therefore consider a signal with an occupied frequency-time domain five times as large as the 3000 sample signal, resulting in a signal of 15,000 samples. This long signal again consists of ideally low-pass filtered, complex, circularly symmetric, white Gaussian noise with zero mean, sampled at oversampling rate $s = 3$ with respect to the Nyquist frequency corresponding to the occupied signal bandwidth. The time domain of this signal was $T = [-375, 375]$, sample time $\Delta t = 0.05$, and occupied bandwidth $\Omega = [-20\pi/3, +20\pi/3] = [-20.9, 20.9]$. The signal amplitude was scaled such that the mean energy density corresponded to $\eta^{\text{mean}} = 0.1$, equal to the 3000 sample signal. The mean eigenvalue height of the 15,000 sample signal should therefore be equal to the 3000 sample signal, only approximately five times as many solitons should be present.

The 15,000 sample signal was too long to directly compute the eigenvalues with the used version of *FNFT*, as only spurious eigenvalues were detected that were far too high or far too low. We believe this is due to the rational approximation of the function $a(\lambda)$, which requires high precision arithmetic at some point for signals with very long durations. To obtain a ground truth, we instead used conventional Newton refinement to refine a dense grid of initial guesses. To verify that the found eigenvalues were indeed correct, the refinement was repeated using high-precision arithmetic.

We performed the combined frequency- and time-windowing NFT algorithm by running Alg. 2 and then Alg. 3 on the 15,000 sample signal, with the suggested parameters $c_T = 4$, $c_\Omega = 16$, $R = 0.5$ and $\varepsilon = 0.05$. This resulted in $N_\Omega = 7$ frequency-windows and $M_T = 14$ time-windows.

The eigenvalues from the windowed signals are compared to the correct eigenvalues in Fig. 5.5. The top figure shows the result of only time-windowing, and the bottom of combined frequency- and time-windowing. When using only frequency-windowing, all detected eigenvalues were spurious. Even when we used 28 narrow frequency-windows (instead of 7) all eigenvalues were still spurious. This is likely again due to the rational approximation of $a(\lambda)$, that yields numerical issues for signals with long durations.

We observe that time-windowing yields very accurate results for the majority of the eigenvalues above η^{mean}, while only a few of the high eigenvalues are visually different from the correct eigenvalues. A few eigenvalues are still missing (one example is around $\lambda = 9.2 + 0.16i$), despite the extension of the acceptance region with $\varepsilon = 0.05$. We attempted to further enlarge the acceptance regions, but this resulted in several spurious eigenvalues being accepted, while still not capturing all missing eigenvalues. Apparently, a small number of high eigenvalues are still missed altogether. However, this issue could be overcome by redoing the calculation with an upsampled version of the signal (e.g., using band-limited interpolation). Upon upsampling the signal to $s = 4.5$ (from $s = 3$, i.e. 50% more samples) and increasing the acceptance region to $\varepsilon = 0.10$, all missing eigenvalues above η^{mean} were recovered, although several poorer copies of eigenvalues with errors up to 5% were also accepted (figure omitted).

The results of the combined frequency- and time-windowing NFT are shown in the bottom of Fig. 5.5. We observe clear visual differences between the eigenvalues from the frequency- and time-windowed signals and the eigenvalues from the full signal. As for

the 3000 sample signal, this is mostly the results of the frequency-windowing. Although some eigenvalues show an error of up to 10%, most eigenvalues are still within 1% of the correct value. The resulting eigenvalues are therefore still useful as rough estimates of the eigenvalues. They may also be used as initial points for local refinement using the full signal to find the correct eigenvalues.

In summary, we observe from the random 15,000 sample signal that time-windowing provides highly accurate results for the higher eigenvalues. Only frequency-windowing did not suffice to analyze the signal due to failures of the used NFT method. Finally, combined frequency- and time-windowing causes significant errors due to the frequency-windowing. We thus suggest to apply time-windowing to reduce the computation time for finding the higher eigenvalues, and only additionally use frequency-windowing when time-windowing alone does not provide sufficient speed up, or when a rough estimate of the discrete spectrum is sufficient.

5.5.3. SPEED UP DUE TO WINDOWING

In this section, we consider the speed-up achieved through frequency- and/or time-windowing. Note that the time-bandwidth product of the window in Eq. 5.13 only depends on the choice for the frequency- and time-broadening factors c_Ω and c_T, but not on the signal itself. Independent of the occupied time-bandwidth product of the full signal, we may thus window the full signal into pieces with small and constant time-bandwidth product. The complexity of computing the NFT for each window is therefore approximately constant. As the time-bandwidth product per window is constant, the number of required windows increases linearly with the time-bandwidth product of the full signal. Therefore, a windowing NFT only requires $\mathcal{O}(N)$ FLOPs, as opposed to a full NFT, which typically requires $\mathcal{O}(N^2)$ or $\mathcal{O}(KN)$ (with K the total number of solitons) for the considered configuration.

To demonstrate the speed-up, we considered a variety of low-pass filtered Gaussian white noise signals, with various bandwidths and durations. All signals were generated to have a mean energy density of approximately 0.026 1/rad, corresponding to $\eta^{\text{mean}} = 0.1$, and had oversampling rate of $s = 3$. For all signals we used *FNFT* to calculate the high eigenvalues from the full signal directly, and using the windowing NFT method with the suggested parameters. The correctness of the detected eigenvalues was not validated, only the computation times were recorded for the speed analysis.

The computation times are shown in Tab. 5.1. The first three rows show the computation times of the 3000-sample signal when applying only frequency-windowing, only time-windowing, and combined frequency- and time-windowing NFT. Between rows 3-8, we gradually increased the number of samples in the signal, mostly by increasing the duration of the signal. We doubled the bandwidth of the signal for the signals of 6000 samples and longer, but we observe that this only affects the configuration of the windowing, while the computation time required for the windowing NFT hardly changes. We observe that the windowing NFT determines the higher eigenvalues faster than the full NFT in all considered cases, although the speed-up is most apparent for the longest signal of 15,000 samples, where a speed-up of $\frac{623\,\text{s}}{44.7\,\text{s}} \approx 14$ times was observed for combined frequency-and time-windowing, and a speed-up of $\frac{623\,\text{s}}{111\,\text{s}} \approx 6$ for only time-windowing. We observe that the full NFT indeed requires $\mathcal{O}(N^2)$ computation time (e.g.,

$\frac{15000}{3000} = 5$ times more samples results in about $\frac{623\,\mathrm{s}}{26.1\,\mathrm{s}} \approx 25$ times the computation time), while the windowing NFT only requires $\mathcal{O}(N)$ time (e.g.: $\frac{15000}{3000} = 5$ times more samples results in about $\frac{44.7}{9.61} \approx 5$ times the computation time).

We thus conclude that the windowing NFT is much faster than the full NFT for signals with large time-bandwidth products. While the demonstrated window size works well in most applications, the windows size can be further shrunk to allow faster computations, or increased for higher accuracy of the eigenvalues.

5.6. CONCLUSION

We have proposed and validated a method to compute the significantly large solitonic components in the discrete part of the nonlinear Fourier transform for the nonlinear Schrödinger equation quickly and accurately, by partitioning the signal in the frequency and time domain. We divide the full occupied frequency-time domain into smaller overlapping windows, determine the higher eigenvalues within each individual window, reject the unreliable eigenvalues, and finally combine all accepted higher eigenvalues to obtain the full spectrum of higher eigenvalues associated with the larger solitons. The computation time of the NFT of a signal scales superlinearly in the occupied time-bandwidth product, so it is thus computationally faster to divide the full signal in many small frequency-time domains than to compute the NFT of the full signal. Our proposed frequency- and time-partitioning NFT uses a pre-defined time-bandwidth product for the windows, and thus only scales linearly in the number of used windows. It can therefore also be applied to very complicated signals for which other methods either fail or require very long computation times.

We tested the accuracy of the method on several signals and configurations, first by partitioning only the time domain, then by partitioning only the frequency domain, and finally by partitioning both the frequency and time domain. When only partitioning the time domain, the results are very good, and the higher solitons are captured with high accuracy. Only partitioning the frequency domain resulted in some loss of accuracy of the higher eigenvalues. Combined frequency- and time-partitioning resulted in the fastest result, but with a small loss in accuracy, mostly due to the frequency-partitioning.

To the best of our knowledge, we have thus presented the first method that can calculate the higher eigenvalues of signals with arbitrarily large time-bandwidth product, while the computation time depends only linearly on the occupied time-bandwidth product.

Table 5.1: The computation times for the full NFT and windowing NFT of various signals (the resulting eigenvalues were not validated). All signals were generated as bandwidth-limited white noise signals, with approximate mean energy density corresponding to $\eta^{\text{mean}} = 0.1$. All signals had the same oversampling rate of $s = 3$.

Full domain size ($\Omega \times T$)	Window size ($\Omega^{\text{window}} \times T^{\text{window}}$)	Number of samples, total / per window	Number of windows ($N_\Omega \times M_T$)	Full NFT computation time	windowing NFT computation time, all / single window(s)
20.9 rad × 300	20.9 rad × 101	3000 / 1010	5 (1 × 5)	26.1 s	9.61 s / 3.20 s
20.9 rad × 300	11.3 rad × 300	3000 / 1626	3 (3 × 1)	26.1 s	22.0 s / 7.32 s
20.9 rad × 300	11.3 rad × 101	3000 / 547	15 (3 × 5)	26.1 s	7.84 s / 0.53 s
20.9 rad × 600	11.4 rad × 101	6000 / 547	33 (3 × 11)	122 s	17.7 s / 0.54 s
41.9 rad × 300	11.4 rad × 101	6000 / 547	35 (7 × 5)	85 s	19.1 s / 0.54 s
41.9 rad × 450	11.3 rad × 101	9000 / 546	56 (7 × 8)	224 s	26.6 s / 0.48 s
41.9 rad × 600	11.3 rad × 101	12000 / 546	77 (7 × 11)	442 s	37.9 s / 0.49 s
41.9 rad × 750	41.9 rad × 103	15000 / 2052	14 (1 × 14)	623 s	111 s / 7.92 s
41.9 rad × 750	11.1 rad × 103	15000 / 546	98 (7 × 14)	623 s	44.7 s / 0.46 s

6

DATA-DRIVEN IDENTIFICATION OF THE SPECTRAL OPERATOR IN AKNS LAX PAIRS USING CONSERVED QUANTITIES

Lax-integrable partial differential equations (PDEs) can by definition be described through a compatibility condition between two linear operators. These operators are said to form a Lax pair for the PDE, which itself is usually nonlinear. Lax pairs are a very useful tool, but unfortunately finding them is a difficult problem in practice. In this study, we propose a method that determines the spectral operator of an AKNS-type Lax pair such that the corresponding PDE fits given measurement data as well as possible. The spectral operator then enables practitioners to solve or analyze the underlying PDE using the induced nonlinear Fourier transform. The underlying PDE only has to be approximately Lax-integrable; the method will find the spectral operator that explains the data best. Together with the dispersion relation, the spectral operator of AKNS type completely determines an integrable PDE that approximates the true underlying PDE. We identify the most suitable spectral operator by matching PDE-dependent quantities that should be conserved during evolution. The method is automatic and only requires recordings of solutions at two different values of the evolution variable, which do not have to be close.
This chapter is an adaptation of [144].

6.1. INTRODUCTION

Many nonlinear partial differential equations (PDEs) can be solved with the help of Lax pairs, which consist of a linear spectral operator L and a linear propagation operator A, and the inverse scattering method (ISM) [7, 8]. Well-known examples include the Korteweg-de Vries equation (KdV) [1], the modified Korteweg-de Vries equation (MKdV) [145] and the nonlinear Schrödinger equation (NLSE) [9]. The ISM for the solution of

such so-called Lax-integrable systems proceeds as follows. First, the initial condition is used as a potential function for the linear spectral operator L. The spectrum of this operator together with the corresponding (generalized) eigenfunctions leads to an equivalent spectral representation of the initial condition. The evolution of this spectral representation can then be performed in closed form with the help of the linear propagation operator A. Finally, a suitable spectral theorem is used to recover the solution in the original domain from the evolved spectral representation. The spectral representation can be interpreted as a nonlinear Fourier transform (NFT; a.k.a. forward scattering transform) that moves a signal from its original domain to a PDE-specific spectral domain in which the propagation becomes trivial [7]. The spectral representation furthermore has physical interpretations. Most importantly, it is able to reveal hidden solitons, which are localized particle-like waveforms [61]. NFTs have therefore found practical applications in areas such as ocean wave data analysis (e.g. [135, 146–150]) and fiber-optical communication (e.g. [23, 28, 83, 99, 111, 112, 151–154]). In these areas, many dynamics are known to be well-approximated by the most common Lax-integrable PDEs.

However, the situation is different in areas where the notion of Lax-integrability is not yet well-known. When a system is suspected to be Lax-integrable, e.g. because it admits soliton solutions, the question arises if the system can be described by a Lax-integrable PDE, and if so, which PDE. In practice, it is often very difficult to determine which Lax-integrable PDE approximates a given system best. From a practical point of view, it is often preferable to approach this question in a data-driven way as the analytical derivation of Lax-integrable approximations requires very specific human expertise, time and sometimes also some luck. Data-driven approaches instead try to identify the system directly from measurements of the system. Of course, they come with their own challenges. For the identification of PDEs, one of the biggest practical problems is that the data can typically only be measured for a few sparse values of the evolution variable, and thus the derivative in the direction of propagation cannot be determined. The sparseness of measurement points occurs naturally e.g. in wave flume experiments [43, 150], where wave gauges record time series at a limited number of locations, or in optical fiber experiments [54, 87], where time series can only be recorded at the ends of the fiber link. In contrast, many popular generic PDE identification methods such as [155] assume that the data contains derivatives w.r.t. the evolution variable, which is not possible with sparse measurement points. To the best of our knowledge, so far only a single method for the data-driven identification of Lax pairs has been proposed in the literature [42]. However, also this methods requires derivatives w.r.t. the evolution variable. The lack of a practical system identification method for Lax-integrable systems hinders the application of the strong mathematical theory behind Lax-integrable systems in practice.

In this study, we therefore propose a novel approach to identify a Lax pair from given measurement data that does not require derivatives in the direction of propagation. Instead, we exploit that the spectral operator of any Lax pair produces infinitely many conserved quantities that should not change during propagation. The parameters of the spectral operator are chosen such that the variation in the (ideally) conserved quantities in minimal. Our method is data-driven and identifies the PDE using only measurements space series (snapshots) of solutions, taken at different time points (assuming from here on, without loss of generality, that the time t is the evolution variable of the PDE, and the

location x the other variable).

In earlier work, we have already demonstrated for the NLSE [55, 87] and the KdV equation [43] that parameters of specific integrable PDEs can be identified based on conserved quantities. Furthermore, conserved quantities have also successfully been applied for other applications such as training neural networks for solving Lax-integrable PDEs [156]. However, so far conserved quantities have not been exploited for the data-driven identification of Lax pairs.

In the literature, there already is a large variety of non-data-driven techniques that determine Lax pairs for a given Lax-integrable PDE (e.g., [19–21, 155, 157–162]). One might thus be tempted to first identify a PDE using conventional data-driven methods (e.g. [40, 41]), and then find a Lax pair for it. However, this approach has several major problems. First of all, existing methods for finding Lax pairs assume that the given PDE is exactly Lax-integrable. However, in practice this will only be the case approximately. Even if the true underlying PDE is exactly Lax-integrable, the model identified by a conventional method still will not have this property due to measurement noise. Second, existing techniques to find Lax pairs from a given Lax-integrable PDE are not guaranteed to succeed, to the best of our knowledge. Third, as mentioned above, conventional PDE identification methods require measurements of the solution at closely spaced time points, which, as already pointed out earlier, is often unpractical.

In our approach, we will assume that the Lax-pair is of the AKNS-type, which is named after Ablowitz, Kaup, Newell and Segur [7]. The AKNS class captures a large class of ubiquitous Lax-integrable PDEs, such as, e.g., the (modified) Korteweg-de Vries equation (MKdV/KdV), the (de)focussing nonlinear Schrödinger equation (dNLSE/fNLSE), and the sine/sinh-Gordon equation [7], as well as transformations and higher-order variants of these equations. While the restriction to the AKNS class of course means a loss of generality,[1] we point out that on the other hand it allows us to design a more efficient method. Most importantly, we can exploit that the conservation laws of AKNS Lax pairs all have the same structure. We envision that in the future, more specialized algorithms for other classes of Lax pairs will be developed. The user can then choose the right class either using physical insight, or by simply checking which class provides the best fit. This would be similar to conventional nonlinear system identification, where specialized algorithms exist for various system classes such as Wiener-Hammerstein systems, finite Volterra series, or nonlinear ARMA models [164].

In our method, we focus on identifying the spectral L operator of the AKNS-type Lax-pair, as the Lax propagation operator A can be found by combining the spectral operator and the linearised dispersion relation [7, p. 253]. Given suitable measurement data, the dispersion relation can be found easily by comparing phase shifts in the linear Fourier domain. To identify the L operator that fits the given data best, we will exploit the explicitly known conserved global quantities associated with AKNS-type Lax pairs. Our strat-

[1] The "integrability ex machina" (IeM) method in [42] employs neural networks to represent the Lax pair and therefore, at first sight, appears to cover a much larger class of systems than our approach. However, this is not the case. The neural network in the IeM method implements a conventional polynomial approach [42, p. 4]. The advantage of the neural network formulation is that advanced algorithms to find the optimal weights can be employed. In [163, p. 65], it is clarified that the IeM method cannot use arbitrary neural networks (specifically, multi-layer perceptrons). The IeM method also requires physical intuition to select dictionaries [42, p. 6].

egy is to identify the L operator, parameterized by two potential functions, for which the global quantities vary as little as possible over time.

Our method has the advantage that it is fully automatic and data driven. Its main advantage is thus that it only requires measurements at two different points in time, which do not have to be close. (The use of more time points is possible.) We finally remark that the proposed method may of course also be used for data in the form of time series that are measured at different locations if the roles of space and time are switched in the PDE so that the location is the evolution variable. This is a common scenario in applications. See, e.g., [23, 43, 150].

This Chapter is organized as follows. Sec. 6.2 recapitulates the AKNS-type Lax pair and the associated conserved quantities. Sec. 6.3 describes the method to identify the most suitable AKNS-type PDE from measurement data by comparing global quantities. Sec. 6.4 demonstrates the method on various data sets. Finally, Sec. 6.5 concludes the study.

6.2. THEORY: AKNS LAX PAIRS AND CONSERVED QUANTITIES

Let a nonlinear partial differential equations (PDE) be given by

$$u_t(x,t) = F(u, u_x, u_{xx}, \ldots), \tag{6.1}$$

where u denotes the (possibly complex) signal amplitude, x the position, t the time (or more generally, the evolution variable), F the nonlinear evolution function, and subscripts denote partial derivatives. A nonlinear PDE has a Lax pair $(L(t), A(t))$, where $L(t)$ and $A(t)$ are linear operators that depend on $u(x,t)$, such that the following Lax equation is equivalent to the PDE [8, Eq. 1.4]:

$$L_t = AL - LA. \tag{6.2}$$

We will refer to L as the spectral operator, and to A as the propagation operator. If the Lax pair (L, A) fits the PDE of interest, then the spectrum of $L(t)$ is constant over time [8]. We say that the PDE is Lax-integrable if the Lax pair enables the solution of the PDE using the inverse scattering method [165]. A simple example of a PDE and a corresponding Lax pair is the advection equation $u_t = u_x$, where the Lax pair consists of the multiplication operator $L = u$ and the derivative operator $A = \partial_x$. For any suitable $\phi(x,t)$, we have

$$L_t\phi = (L\phi)_t - L\phi_t = (u\phi)_t - u\phi_t = u_t\phi \quad \text{and} \quad AL\phi - LA\phi = \partial_x(u\phi) - u\partial_x\phi = u_x\phi$$

$$\Rightarrow \quad L_t = AL - LA \iff u_t = u_x.$$

Many Lax-integrable PDEs exist, and the form of the associated Lax pairs can widely vary. However, the AKNS-type Lax pairs all possess the same structure, but still capture many ubiquitous PDEs. The AKNS-type Lax pairs are of the form [7]

$$L = \begin{bmatrix} i\partial_x & -iq(u) \\ ir(q) & -i\partial_x \end{bmatrix}, \qquad A = \begin{bmatrix} A_{11}(u) & A_{12}(u) \\ A_{21}(u) & -A_{11}(u) \end{bmatrix} \tag{6.3}$$

in which $q(u)$ and $r(q) = r\left(q(u)\right)$ are potential functions depending on the x.

Not every choice of $q(u)$ and $r(q)$ is allowed, as the Lax equation in Eq. 6.2 imposes a compatibility condition (see [7, Eq. 2.2-2.7]). Many of the known Lax-integrable system can be obtained using only a small number of $r(q)$ relations [7]:

$$r=-1 \text{ (KdV)},\ r=-q^* \text{ (focussing NLSE)},\ r=q^* \text{ (defocussing NLSE)},$$
$$r=-q \text{ (sine-Gordon, focusing MKdV)},\ r=q \text{ (sinh-Gordon, defocusing MKdV)}. \quad (6.4)$$

We will therefore consider only these relations for $r(q)$. Of course, more choices can be integrated into the method, although it should be checked beforehand if those relations $r(q)$ can lead to a compatible Lax pair. For most equations, such as KdV, MKdV and NLSE, $q(u)$ simply takes the form of $q=cu$ (i.e., a scaling of the signal amplitude, with c the scaling coefficient). The sine-Gordon and sinh-Gordon equations on the other hand require $q=cu_x$. Furthermore, we may also be dealing with the case that the measurement process does not directly provide u, but a transformed version of it. For example, $q(u)=u^3, r(q)=-1$ would still result in a feasible spectral operator of the KdV-type, but the associated PDE would correspond to a transformed version of the KdV (the exact PDE would depend on the linearised dispersion relation).

6.2.1. CONSERVED QUANTITIES OF AKNS-TYPE PDES

AKNS-type PDEs possess an infinite number of conserved quantities, that are fully determined by $q(u)$ and $r(q)$. The conserved quantities can be iteratively derived using a relation from AKNS [7, Eq. 7.33]. The first five conserved quantities are

$$C_1=\int_D qr\,\mathrm{d}x, \qquad C_2=\int_D rq_x-r_xq\,\mathrm{d}x, \quad (6.5a)$$
$$C_3=\int_D q^2r^2+q_xr_x\,\mathrm{d}x, \qquad C_4=\int_D -r_{xx}q_x+q_{xx}r_x+\tfrac{3}{2}\left(r^2(q^2)_x-q^2(r^2)_x\right)\mathrm{d}x, \quad (6.5b)$$
$$C_5=\int_D 2q^3r^3+q_{xx}r_{xx}+\left(q_x^2r^2+r_x^2q^2\right)+8qq_xrr_x\,\mathrm{d}x, \quad (6.5c)$$

where D denotes the spatial domain. Both periodic and vanishing boundary conditions are allowed.

The choices $r=-1$ and $r=\pm q$ lead to trivial conserved quantities $C_2=0$ and $C_4=0$, but the odd quantities are non-trivial for all considered choices of $r(q)$. Therefore, we only consider the odd global quantities C_1, C_3 and C_5 throughout the rest of this chapter. These three conserved quantities turn out to be sufficient in our numerical experiments. Higher ones could of course be integrated, and with large libraries for the expansion of q (see Sec. 6.3), this may even become necessary for a successful identification. Unfortunately, determining the higher conserved quantities analytically is a difficult task, even with the help of symbolic computer algebra systems. Higher order conserved quantities furthermore contain derivatives of high orders, which are very sensitive to measurement noise. We therefore believe that the exploitation of higher order conserved quantities for large libraries will require a numerical method for their computation from noisy data.

We finally note that it is necessary to consider C_5, as only C_1 and C_3 are not sufficient to distinguish most PDEs. Many PDEs conserve both $\int_D u\,\mathrm{d}x$ and $\int_D u^2\,\mathrm{d}x$, thus if $q=u$ then $r\in\{1,\pm u\}$ all lead to a conserved C_1. Furthermore, if only C_1 is used, and we found

some relation $q = f(u)$ such that $C_1(t)$ is constant, then any scaling $q = cf(u)$ will also lead to a constant $C_1(t)$. To extract the scaling constant, at least C_3 is required for $r \in \{\pm q, \pm q^*\}$, as C_3 then has two terms with different scaling in q. For $r = -1$, C_3 still only has a single term ($\int_D q^2 \, \mathrm{d}x$), and therefore C_5 is required.

6.3. METHOD: IDENTIFYING THE SPECTRAL OPERATOR USING CONSERVED QUANTITIES

In this section, we present our novel method to identify the spectral operator L of the AKNS Lax pair that fits a given data set best. The methods checks which $q(u)$ results in the least variation in the conserved quantities, for each of the choices of $r(u)$ in Eq. 6.4. The combination $\big(r(q), q(u)\big)$ that results in the least variation of the conserved quantities is then considered the best candidate for the spectral operator explaining the data.

6.3.1. EXPANDING $q(u)$ IN FUNCTIONS FROM A LIBRARY

Our approach starts with choosing a library of D operators of u, i.e., $g_1(u), g_2(u), \ldots, g_D(u)$. We will allow $q(u)$ to be any linear combination of the operators in a library G, which we chose for this study as the lowest-order polynomials and derivatives of u:

$$q(u) = \sum_{d=1}^{D} c_d g_d(u), \text{ with } D = 5,$$
$$G = \{g_1 = u,\, g_2 = u_x,\, g_3 = u_{xx},\, g_4 = u^2,\, g_5 = uu_x\}. \quad (6.6)$$

Here, c_d are the coefficients which are optimized during the identification. We chose the shown library to include at least u and u_x, such that standard versions of the KdV, MKdV and NLSE ($q = u$) and sine-Gordon and sinh-Gordon ($q = u_x$) are in the current library space. If more information of the PDE is available, the library can be expanded or shrunk accordingly. For example, if the underlying system is known to be either KdV or MKdV, the library can be shrunk to $G = \{g_1 = u\}$, and the problem then simplifies to a simple identification of the scaling constant c_1, and a choice between $r = -1$, $r = -q$ and $r = +q$.

6.3.2. ERROR FUNCTION BASED ON CONSERVED QUANTITIES

To identify the most suitable potential function $r(q)$ as in (6.4) and q as in (6.6) for the L operator in (6.3), we will minimize an error based on the conserved quantities (6.5). We assume that our available data set consists of N independent trajectories

$$u^{(n)} = \Big(u^{(n)}(\cdot, t_1), u^{(n)}(\cdot, t_2), \ldots, u^{(n)}(\cdot, t_M)\Big), \quad n = 1, \ldots, N,$$

where the first "snapshot" $u^{(n)}(x, t_1)$ of the nth trajectory is assumed to be a measurement of the initial condition of the PDE, and the later snapshots at one or multiple later times $t_2, \ldots, t_M$ are measurements of the correspondingly evolved initial conditions. Given a single trajectory $u^{(n)}$, each choice of $\big(q(u), r(q)\big)$ results in three time-dependent values $C_1(t)$, $C_3(t)$ and $C_5(t)$ for the conserved quantities in (6.5). If the relations $\big(q(u), r(q)\big)$ are chosen correctly and the data set was noiseless and obtained from

an exactly Lax-integrable system, the conserved quantities will indeed be conserved, i.e., constant. We thus create an error that penalizes fluctuations in the conserved quantities over time.

We formally define the error as the standard deviation in time in the conserved quantities C_1, C_3 and C_5, relative to the average absolute conserved quantities:

$$E(c,r;u) := \sum_{k\in\{1,3,5\}} \left(\sum_{n=1}^{N} \frac{\sigma[C_k(t;c,r,u^{(n)})]}{\mu\left[\left|C_k(t;c,r,u^{(n)})\right|\right]} \right), \quad \text{with}$$

$$\mu[C_k(t)] = \frac{1}{M}\sum_{m=1}^{M} C_k(t_m), \text{ and } \sigma[C_k(t)] = \sqrt{\frac{1}{M}\sum_{m=1}^{M}\left(C_k(t_m)-\mu[C_k(t)]\right)^2}, \quad (6.7)$$

where $C_k(t;c,r,u^{(n)})$ denotes the corresponding conserved quantity in Eq. 6.5 computed at time t, which furthermore depends on the choices for the coefficient vector c, r, and the currently considered trajectory through Eqs. 6.4- 6.6. The algorithm thus identifies r and $q = \sum_d c_d g_d$ by minimizing over the error in Eq. 6.7:

$$(r^{(\mathrm{ID})}, c^{(\mathrm{ID})}) = \mathrm{argmin}_{r\in\{-1,\pm q,\pm q^*\}} \mathrm{argmin}_{c\in\mathscr{C}}\, w(c)E(c,r;u), \quad \mathscr{C}\subseteq\mathbb{R}^d, \quad (6.8)$$

with $w(c)$ denoting the weight of coefficient c. For each choice of r, the error is minimized over c using a local minimization method. We used the *fminsearch* method from *Matlab* (a simplex search method), starting at multiple initial starting positions. The search space $\mathscr{C}$ for the coefficient vectors can in principle be chosen as the whole $\mathbb{R}^d$, but in our numerical examples we made a different choice. Similarly to conventional identification methods for PDEs (e.g. [41]), we prefer Lax pairs that are "simple" and therefore prioritize sparse coefficient vectors c. This will also prevent over-fitting, and can help to reduce the computation time. To find a sparse solution c we do not search the full space spanned by the library G at once, but rather search only its low-dimensional sub-spaces one-by-one as explained in the next section.

6.3.3. SPARSE SOLUTIONS AND CHOICE OF THE STARTING POINTS

For large libraries G, it can be very hard to find a globally optimal $q(u)$ in the corresponding function space. However, for many of the well-known AKNS-type PDEs, the corresponding $q(u) = \sum_d c_d f_d(u)$ has only a few nonzero c_d. We exploit this by only searching the low-dimensional sub-spaces of functions in G that have at most $D^{(\mathrm{sub})}$ non-zero coefficients. In our numerical examples, we investigate all 1D, 2D and 3D sub-spaces. While the 1D and 2D sub-spaces are also contained within the 3D sub-space, it often occurs that a one- or two-dimensional solution performs nearly as good as a three dimensional solution, and should thus be preferred. Furthermore, the optimization process in lower-dimensional sub-spaces is more likely to find the global optimum in that sub-space. We thus also take the lower sub-spaces into account.

Given D basis functions, there are $\binom{D^{(\mathrm{sub})}}{D}$ sub-spaces of dimension $D^{(\mathrm{sub})}$. In each of these sub-spaces, we minimize the error in Eq. 6.7, and keep the optimal 1D-, 2D- and 3D-solutions. Finally, we penalize higher-dimensional solutions by multiplying each error with a heuristically chosen factor. The search space and weight for (6.8) are thus

$$\mathscr{C} = \{c\in\mathbb{R}^D : \#c \le D^{(\mathrm{sub})}, D^{(\mathrm{sub})} \le 3\}, \quad w(c) = p^{\#c}, \quad (6.9)$$

with $\#c$ the number of non-zero coefficients in c. We found that $p = 1.2$ resulted in a good balance for finding a suitable but sparse coefficient vector.

We assume that the data set evokes both the linear and the nonlinear terms of the PDE, at comparable levels, so that all terms can be identified. If this is not the case, a richer data set must be considered, as many PDEs can otherwise not be distinguished. To define initial starting points, we determine a central estimate vector c^0, for which the terms $c_d^0 g_d(u)$ are all of the same order of magnitude, such that each g_d is relevant. We start with the first coefficient c_1^0, as $c_1^0 u$ is often part of the solution for q. We cannot use the conserved quantity C_1 for this purpose, as it contains only a single term; C_3 has two different terms, so we can find a c_1^0 such that $\int_D q^2 r^2 \,\mathrm{d}x$ and $\int_D q_x r_x \,\mathrm{d}x$ are of similar order. Only for $r = -1$ we have that $r_x = 0$, so in this case we consider C_5 instead. This leads to the following initial r-dependent guesses for c_1^0:

$$r = -1,\ q = c_1 u, \quad C_5 = \int_D 2c_1^3 u^3 + c_1^2 u_x^2 \,\mathrm{d}x \qquad \Rightarrow c_1^0 \approx \frac{\overline{|\int_D u_x^2(x,t)\,\mathrm{d}x|}}{\overline{|\int_D u^3(x,t)\,\mathrm{d}x|}} \tag{6.10a}$$

$$r = \mp q,\ q = c_1 u, \quad C_3 = \int_D c_1^4 u^4 + c_1^2 u_x^2 \,\mathrm{d}x \qquad \Rightarrow c_1^0 \approx \sqrt{\frac{\overline{|\int_D u_x^2(x,t)\,\mathrm{d}x|}}{\overline{|\int_D u^4(x,t)\,\mathrm{d}x|}}} \tag{6.10b}$$

$$r = \mp q^*,\ q = c_1 u, \quad C_3 = \int_D c_1^4 |u|^4 + c_1^2 |u_x|^2 \,\mathrm{d}x \qquad \Rightarrow c_1^0 \approx \sqrt{\frac{\overline{|\int_D |u_x|^2(x,t)\,\mathrm{d}x|}}{\overline{|\int_D |u|^4(x,t)\,\mathrm{d}x|}}}, \tag{6.10c}$$

where $\overline{(\cdot)}$ denotes the mean over all t and trajectories.

Next, we wish to ensure that all functions $c_d^0 g_d$ are of similar size for our initial guess c^0, as this will allow a change in every term to be significant. We thus choose

$$c_d^0 = \frac{c_1 \overline{|g_1|}}{\overline{|g_d|}}. \tag{6.11}$$

This defines our central initial guess.

Next, for each of the sub-spaces, we choose the first initial starting point as the projection of c^0 onto the corresponding subspace (i.e., we set $c_d^0 = 0$ if g_d is not in the subspace). Next, we create a hypercube around the projected c_d^0 by multiplying or dividing each term by a scale factor $s > 1$. For example, for the subspace containing the first two basis functions, the five resulting initial starting points are the projected estimate $(c_1^0, c_2^0, 0, 0, 0)$ and the four vertices $(s^{\pm 1} c_1^0, s^{\pm 1} c_2^0, 0, 0, 0)$. Throughout this study, we chose $s = 10$. This distribution of initial starting points sufficed for the examples considered in this study. More initial starting points can of course be added to increase the chance of finding the global optimum when necessary, at the expense of longer computation times. We also note that all coefficients of our initial starting points were positive, as this sufficed for the considered examples. However, starting points with negative coefficients can of also be taken into account.

6.4. RESULTS

In this section, we evaluate the proposed method in numerical examples for several PDEs. We first demonstrate it for noisy data obtained from the MKdV. Then, a lossy

NLSE, which is only approximately integrable, is considered with noise. Next, we consider the sine-Gordon equation with noise, which requires a non-standard choice for the first potential function, namely $q = u_x$. Finally, we consider a transformed KdV, where $q(u) = u + u^2$ is a linear combination of two of the library functions. In all examples, the library of functions for $q(u)$ from Eq. 6.6 is used, in combination with the described low-dimensional sub-space search and the described initial starting points for c from Sec. 6.3.3.

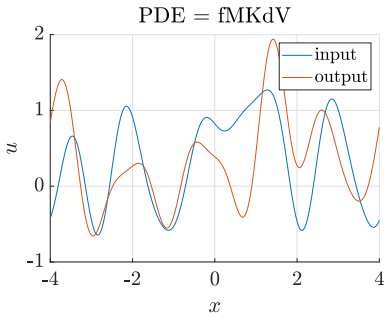
6.4.1. CASE 1: NOISY DATA OBTAINED FROM THE EXACTLY INTEGRABLE MKDV

In this subsection, we consider noisy data obtained from the focusing MKdV $u_t = -u_{xxx} - 6u^2 u_x$, consisting of multiple periodic signals measured at two different times. This example illustrates that the algorithm can distinguish between the $r(q)$ relations for the focusing MKdV ($r = -q$), the defocusing MKdV ($r = +q$) and the (standard) KdV ($r = -1$), and also find the correct relation $q(u) = u$ from the full library function space.

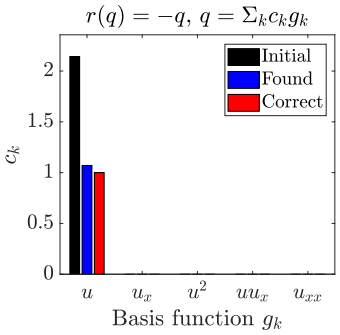
The considered data set consist of $N = 21$ trajectories, where each trajectory consists of space series at only two time points, which are too far apart to determine u_t. The measurement locations and times were respectively $x = -4, -3.98, -3.96, \ldots, 4$ (periodic boundary conditions), and $t \in \{0, 0.5\}$. The input signals ($t = 0$) were generated as zero-mean Gaussian distributed random numbers with variance one, ideally low-pass filtered to maximum wavenumber 1 cycle/spatial unit. Every input signal was scaled such that the mean squared amplitude of each trajectory varied between 0 and 2.5, to ensure that the dataset was rich enough. Next, all input signals were propagated for 0.1 temporal units. Finally, low-pass filtered white Gaussian noise (maximum wavenumber 1 cycle/spatial unit) with 1% of each considered signal's root-mean-squared (RMS) amplitude was added, i.e, the signal to noise ratio was $\mathrm{SNR} = \frac{\int_D |u_{(\mathrm{signal})}|^2 \,\mathrm{d}x}{\int_D |u_{(\mathrm{noise})}|^2 \,\mathrm{d}x} = 10^4$. We note here that the amount of noise was chosen such that the correct PDE could still be identified. This is often around 1% due to the use of higher derivatives and polynomials, which is typical for PDE identification methods (see e.g., [41]).

The results of the algorithm are shown in Fig. 6.1. One of the 21 input-output pairs is shown in the top left. For each choice of $r(q)$, we optimized c. The corresponding global quantities are shown at the bottom. The choice $r(q) = -q$ (indeed corresponding to focusing MKdV) shows the least variation between input and output, and thus the system is identified as $r(q) = -q$. Note that even for the correct choice of $r(q) = -q$, the global quantities are not entirely conserved due to the noise.

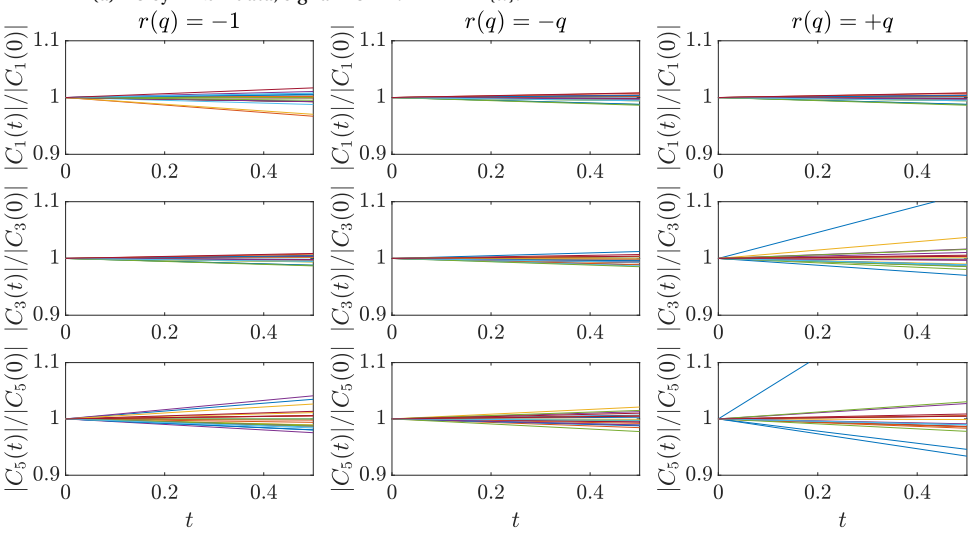
From Fig. 6.1b we find that the identified c corresponded to the correct c (up to a small error due to noise), which was identified from the 1D sub-space spanned by $\{u\}$. A near-correct c was also identified as $(1.06, 0, 0, 6 \cdot 10^{-3}, 0)$ in the $2D$ sub-spaces spanned by $\{u, u_{xx}\}$, but the corresponding error was only marginally smaller. Therefore, the 1D solution was accepted. The correct solution was not found at all in any of the 3D sub-spaces from the specified starting points, and the resulting error was therefore around 20 times larger. While it would in principle be possible to address this by just using more starting points, this example also shows that searching all lower-dimensional sub-spaces can be more effective if the reference solution can be assumed to be low-dimensional.



(a) Noisy MKdV data, signal 1 of 21.



(b) Identified coefficients, assuming $r(q) = -q$. This solution was found in the 1D-subspace spanned by $\{u\}$.



(c) The global quantities at the identified choice for c, under the assumption of the each $r(q)$.

Figure 6.1: Focusing MKdV generated dataset and the results of the identification algorithm. (a) One of the trajectories at first and last measuring time. (b) The initial, identified, and correct coefficients when assuming the relation $r = -q$, i.e., the correct relation for the focusing MKdV. (c) The conserved quantities at the identified coefficients for each of the three considered relations $r(q)$. The choice $r(q) = -q$ shows the least variation and is therefore accepted.

6.4.2. CASE 2: NOISY DATA FROM A LOSSY, AND THUS ONLY APPROXIMATELY INTEGRABLE, NLSE

In this section, we consider complex-valued noisy data set obtained from the lossy focusing NLSE (fNLSE) $q_t = iq_{xx} + 2iq|q|^2 - 0.5\alpha q$ with $\alpha = 4$. The data is instantaneously amplified just before each measurement point to compensate the loss, and is therefore energy conserving (up to noise) between measuring points. This type of system is widely used – with reversed roles for time and space – in fiber-optical communication systems

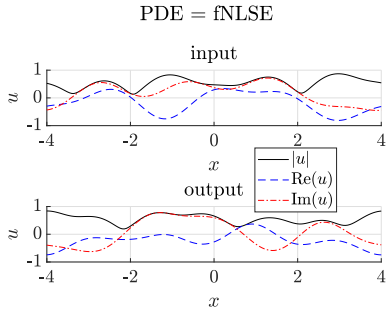
with lumped amplification [62, 166]. The path-average approximation of the lossy fNLSE with amplification is obtained by averaging the decaying energy during over t, and then considering a lossless NLSE with rescaled amplitude $\overline{q}$ [62, Ch. 9]:

$$\mathscr{E}^{\text{avg}} = \int_D [q(x,t=0)]^2 \,\mathrm{d}x \int_0^T e^{-\alpha t}\,\mathrm{d}t = \int_D [q(x,t=0)]^2 \,\mathrm{d}x \frac{1-e^{-\alpha T}}{\alpha T}$$
$$\Rightarrow \overline{q(x,t)} = \sqrt{\tfrac{1-e^{-\alpha T}}{\alpha T}}\, q(x,t=0), \quad (6.12)$$

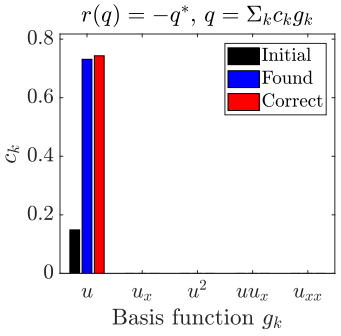
with $\mathscr{E}^{\text{avg}}$ the signal energy averaged over t, $\overline{q(x,t)}$ the path-average amplitude, and T the duration between two amplifications (and thus measurement times). The measurement time points were at $t \in \{0, \frac{1}{3}, \frac{2}{3}, 1\}$, so with $T = \frac{1}{3}$ and $\alpha = 4$, we expect an amplitude normalization constant of $c_1 = \sqrt{\frac{1-e^{-\alpha T}}{\alpha T}} = 0.74$, and thus the relation $u = c_1 q$, and $r = -q^*$. Note that while the path-average approximation (6.12) is Lax-integrable, the true underlying lossy NLSE with amplification is not. This example thus demonstrates that the method is able to identify an integrable approximation to a non-integrable system from data.

We generated the input signals under periodic boundary conditions from complex, circularly symmetric, zero-mean Gaussian distributed random samples, and ideally low-pass filtered these to maximum wave number 0.5 cycles/spatial unit. The measurement points were $x = -4, -3.95, -3.9, \ldots, 4$. Next, we propagated each signal numerically, boosting the signal just before every measurement. Finally, low-pass filtered radially symmetric white Gaussian noise (maximum wavenumber 0.5 cycles/spatial unit) with root mean squared amplitude equal to 1% of the signal amplitude was added to the input and output signals (SNR = 10^4). We generated $N = 21$ input signals in total, with mean squared amplitudes varying between 0.2 and 0.6, to ensure that the data was rich enough. The initial starting points were chosen as described in Sec. 6.3.3.

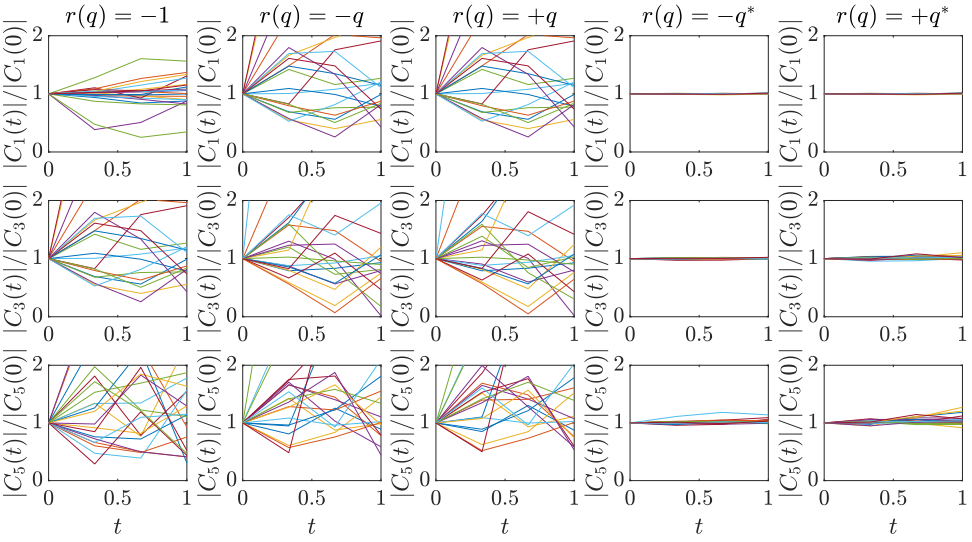
The results of the algorithm are shown in Fig. 6.2. From Fig. 6.2c we note that only the focusing and defocusing NLSE approximately conserve the first global quantity. We then observe that the fNLSE ($r = -q^*$) and dNLSE ($r = +q^*$) can be distinguished by the fact that C_3 and C_5 are not conserved very well for $r = +q^*$, whereas $r = -q^*$ conserves these much better. From Fig. 6.2b, we observe that the guess based on Eq. 6.10 is indeed reasonably close to the reference value for the scaling constant c_1, despite the path-average approximation and the noise. However, we found that the identified c_1 was structurally lower than the reference value by 3%, even when the data was noiseless. This may indicate that the global quantities become slightly biased with respect to the path-averaged approximation.



(a) Noisy fNLSE data at $t = 0$ (input) and at $t = 1$ (output), signal 1 of 21. The signals at $t = 1/3$ and at $t = 2/3$ are omitted here.



(b) Identified coefficients, assuming $r(q) = -q^*$. This solution was found in the 1D-subspace spanned by $\{u\}$.



(c) Global quantities at optimal choice for c_1 for various hierarchies $r(q)$.

Figure 6.2: Focusing NLSE generated dataset, $t \in \{0, 0.33, 0.67, 1\}$, $N = 21$, $x = -4, -3.95, -3.9, \ldots, 4$, signal bandwidth = 1 Hz. (a) One of the 21 signals from the dataset, shown at $t = 0$ and $t = 1$. (b) The initial, identified, and correct coefficients when assuming the relation $r = q^*$, i.e., the correct relation for the fNLSE. (c) The conserved quantities at the identified coefficients for each of the five considered relation $q(r)$. The choice $r(q) = -q^*$ shows the least variation and is therefore accepted.

6.4.3. CASE 3: NOISY DATA FROM THE INTEGRABLE SINE-GORDON DATA (NON-TRIVIAL $q(u)$)

In this subsection, we consider data obtained from the sine-Gordon equation $u_{tx} = \sin(u)$ in light-cone coordinates. This equation has correct choice $q = 0.5u_x$, which is different from the more common relation $q = c_1 u$ used for the KdV and the NLSE. The data set was generated by simulating a kink-antikink collision of the sine-Gordon equation in physical coordinates (α, β), i.e. $u_{\beta\beta} = u_{\alpha\alpha} - \sin(u)$, and then transforming the data through $t = \frac{\alpha+\beta}{2}$, $x = \frac{\alpha-\beta}{2}$. The initial data was taken as a linear summation of a kink- and an antikink solution [167] that were sufficiently far apart to closely approximate an exact double soliton:

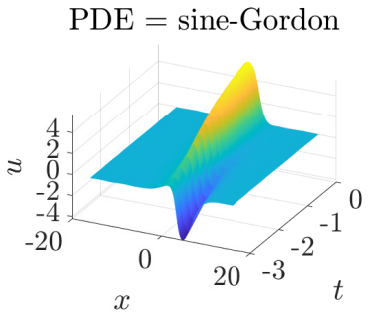
$$u(\alpha,0) = \underbrace{4\text{atan}\left(e^{\gamma_1(\alpha-\alpha_1)}\right)}_{\text{kink}} + \underbrace{-4\text{atan}\left(e^{\gamma_2(\alpha-\alpha_2)}\right)}_{\text{anti-kink}}, \tag{6.13a}$$

$$u_\beta(\alpha,0) = \underbrace{-2\gamma_1 v_1 \text{sech}\left(-\gamma_1(\alpha-\alpha_1)\right)}_{\text{kink}} + \underbrace{-2\gamma_2 v_2 \text{sech}\left(-\gamma_2(\alpha-\alpha_2)\right)}_{\text{anti-kink}}, \tag{6.13b}$$

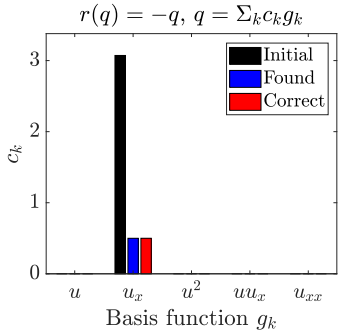
$$\text{with } \left(v_1 = 0.2, \gamma_1 = \sqrt{\tfrac{1}{1-v_1^2}}, \alpha_1 = -2\right) \text{and } \left(v_2 = -0.1, \gamma_2 = \sqrt{\tfrac{1}{1-v_2^2}}, \alpha_2 = 3\right), \tag{6.13c}$$

where α_1 (resp. α_2) is the initial positions of the (anti-)kink, and v_1 (resp. v_2) the velocity. The initial signal was then numerically propagated until after the collision, Measurements in physical coordinates were taken in the region $\alpha = -20, -19.96, \ldots, 20$, and $\beta = 0, 0.04, \ldots 20$, which were transformed to light-cone coordinates. The resulting light-cone domain becomes diamond-shaped, and we considered only a square domain from it, resulting in the final measurement locations $x = -16, -15.92, -15.84, \ldots, 16$, $t = -3, -2.8, -2.6, \ldots, 0$. Only a single $(N = 1)$ trajectory was used, which measured the developing wave nearly continuously, instead of at only a few moments in time. This example thus illustrates that it is also possible to use continuous fields. Finally, we measured the smallest bandwidth containing 99% of the power of the initial condition $u(x, t = -3)$, and added white Gaussian noise, ideally low-pass filtered to this bandwidth, to all space-series measurements leading to SNR = 10^4. The full signal is shown in Fig. 6.3a.

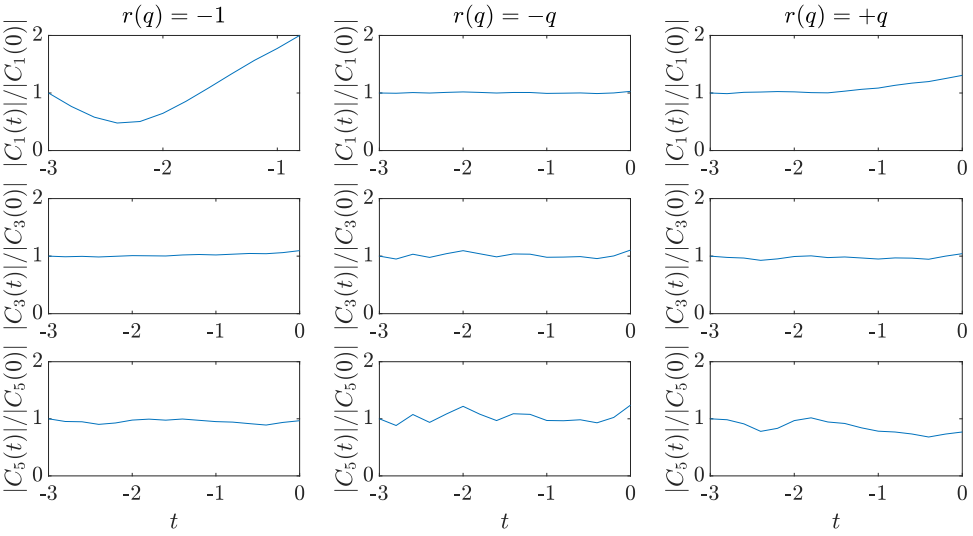
The results are shown in Fig. 6.3. From Fig. 6.3c we observe that the conserved quantities are best conserved by the choice $r(q) = -q$. The correct choice $q(u) = 0.5u_x$ was indeed identified up to noise as shown in Fig. 6.3b.



(a) Kink-anti-Kink collision in light-cone coordinates.



(b) Identified coefficients, assuming $r(q) = -q$



(c) The global quantities, at the optimal $q(u)$ given the shown relation for $r(q)$.

Figure 6.3: Noisy Sine-Gordon data. (a) The full dataset consisting of a kink-antikink soliton in light-cone coordinates, shown between $t = -3$ and $t = 0$. (b) The initial, identified, and correct coefficients when assuming the relation $r = q^*$, i.e., the correct relation for the fNLSE. (c) The conserved quantities at the identified coefficients for each of the considered relations $q(r)$.

6.4.4. CASE 4: NOISELESS AND NOISY TRANSFORMED DATA FROM THE INTEGRABLE KDV (MORE COMPLICATED $q(u)$)

In this subsection, we again consider only two measurement time points, but this time for a transformed KdV. The data was generated by initializing a random, Gaussian distributed random signal with periodic boundary conditions, and ideally low-pass filtering it a maximum wave number of 1 cycles/spatial unit, with $x = -4, -3.95, -3.9, \ldots, 4$. The generated input signal was propagated according to the KdV $v_t = -v_{xxx} - 6vv_x$ for $T = 1$ time unit. Finally, all signals were mapped to the measurement variable u, given by $u = -\frac{1}{2} + \sqrt{v + \frac{1}{4}} : [-\frac{1}{4}, +\infty) \to [-\frac{1}{2}, +\infty)$, such that the correct mapping back is $v = q(u) = u + u^2$. We carefully respected the domain for v to ensure that the mapping is a bijection on the given domains. We generated $N = 41$ input signals in total, with power $\overline{u^2}$ varying between 1 and 20 after the transformation, to ensure that the data was rich enough. Since we found that the method fails for the SNR used in the previous examples, we first consider the case of no noise. We then investigate other SNRs and discuss a potential explanation.

The results for the noiseless case are shown in Fig. 6.5. We observe from the conserved quantities in Fig. 6.5c that the conserved quantities do not vary much for all three choices of $r(q)$, but only the KdV conserved them perfectly (due to the lack of noise in the signal). From Fig. 6.5b we observe that a nearly perfect solution was identified: $q^{\text{ID}}(u) = 1.0000u - 0.0009u_x + 0.9999u^2$. Although the term u_x was negligible, it decreased the error by 30% with respect to the reference solution $q^{\text{ref}} = u + u^2$, and was thus accepted over the 2D solution.

Upon redoing the experiment with noise, we found that a near-correct solution was still identified at SNR=10^7. However, when increasing the noise to the level of previous examples, SNR=10^4, we found that the identified solution $c^{\text{ID}} = (9.62, 0, -1.29, 0, 0)$ was completely different from the reference solution $(1, 0, 0, 1, 0)$, although the conserved quantities were still varying very little. Upon inspection, we found that the error at this c was about twice as low as at the reference solution.

We have the hypothesis that the significantly lower error in the identified solution than in the reference solution is due to the different noise amplification properties of the library functions. For example, consider the influence of noise in the library functions $g_1(u) = u$ and $g_3(u) = u^2$. Let $u = u_0 + \varepsilon$ denote the noiseless signal plus noise ε, with $|\varepsilon| \ll |u|$. The relative noise in the library function $u = u_0 + \varepsilon$ is then given by $\frac{u-u_0}{u_0} = \frac{\varepsilon}{u_0}$. However, in the quadratic library function $u^2 = (u_0 + \varepsilon)^2$, the relative noise is given by $\frac{u^2-u_0^2}{u_0^2} = \frac{(\cancel{u_0^2} + 2u_0\varepsilon + \cancel{\varepsilon^2}^{\sim 0}) - \cancel{u_0^2}}{u_0^2} \approx 2\frac{\varepsilon}{u_0}$. We thus observe that the relative noise in library function $g_3(u) = u^2$ is twice as large as in library function $g_1(u) = u$. We thus expect the error to consist of a contribution due to the model mismatch, and a part due to the noise: $E = E_{\text{mismatch}} + E_{\text{noise}}$. For the reference solution, there is no contribution due to model mismatch, but the error due to noise will be approximately twice as large than for incorrect solutions of the form $q = c_1 u$. If the mismatch is small, a wrong q might thus be identified even for quite high SNRs. To test this hypothesis, we compared the errors at the reference solution $q = u + u^2$ and at $u = c_1 u$ (such that c_1 minimized the error) for various noise levels. As is shown in Fig. 6.4, it indeed turns out that the error contribu-

tion due to noise is approximately twice as large in $q = u + u^2$ as in $u = c_1 u$, and that the reference solution leads to a higher error already at SNR= 10^6.

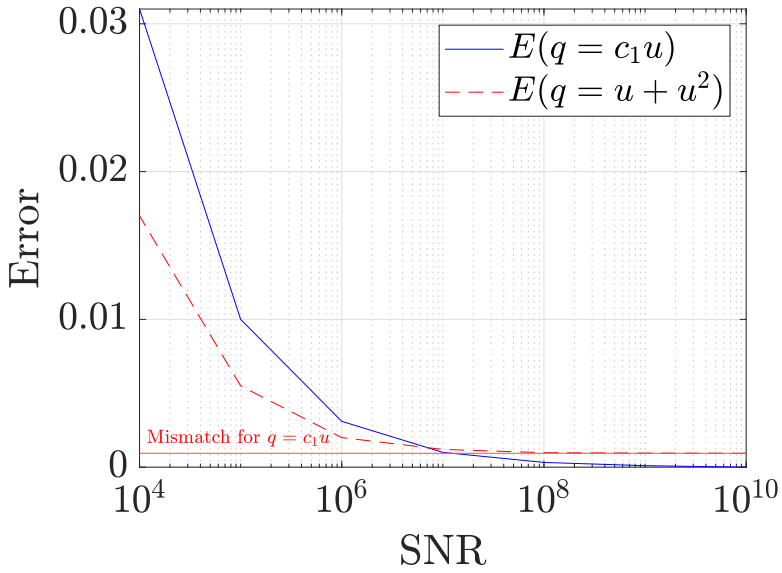
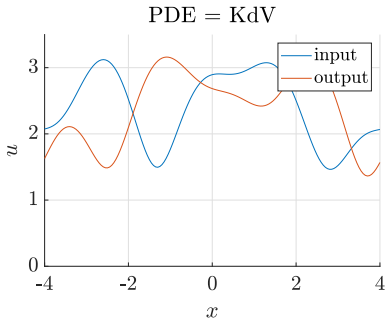
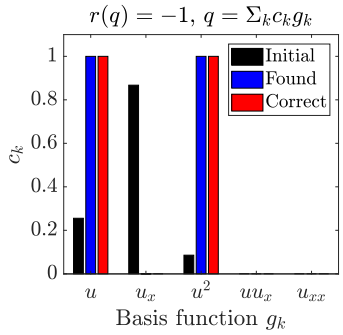


Figure 6.4: The error for the reference solution $q = u + u^2$ and for $q = c_1 u$ at optimal c_1, for various levels of the SNR. At high SNR, the reference solution is near-perfect, while the error for $q = c_1 u$ consists only of a model mismatch error. However, at lower SNR, the error in the reference solution grows approximately twice as fast as for $q = c_1 u$.

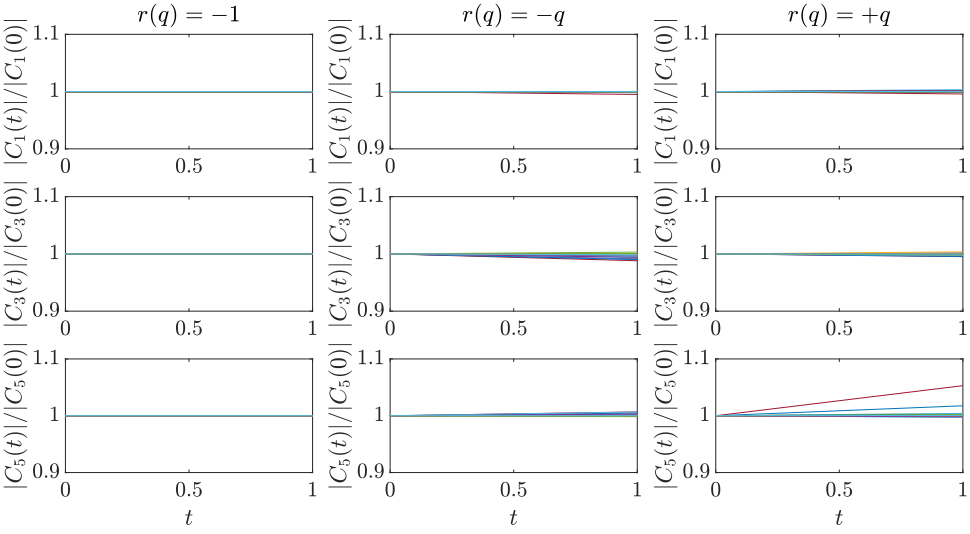
As it should, the proposed method identifies the correct solution if the SNR is large enough. However, it seems likely that the range of sufficiently large SNRs could be expanded by equalizing the impact of noise for the different library functions. For example, one could try to select the library functions for u more carefully. When we removed the library functions u and u^2, and replaced those with the single library function $u + u^2$, the reference solution was identified at SNR = 10^4. We also attempted to replace the library functions u and u^2 with $u + \frac{1}{2}u^2$ and u^2, such that the library still spans the same space as before, but these two library functions are affected in similar amounts by the noise due to their quadratic term. However, the identified solution still converged to approximately $c_1 u = c_1 \left[(u + \frac{1}{2}u^2) - \frac{1}{2}(u^2)\right]$, as the higher noise in the quadratic terms of the two library functions still cancel out for this solution. Nevertheless, there might be better, systematic ways to design the library. Alternatively, one could also try to weight the contributions from the different library functions in the error to compensate for differences in their noise-sensitivity. Choosing these weights however is a non-trivial problem that is beyond the scope of this study.



(a) Noiseless transformed KdV data, signal 1 of 21.



(b) Identified coefficients, assuming $r(q) = -1$



(c) Global quantities at optimal choice for c_1 for various hierarchies $r(q)$.

Figure 6.5: Noiseless KdV data $v(x,t)$, transformed as $u = -\frac{1}{2} + \sqrt{v + \frac{1}{4}}$. The signal domains and parameters were $t \in \{0,1\}$, $N = 21$, $x = -4, -3.95, -3.9, \ldots, 4$, signal bandwidth = 1 Hz. (a) One consisting of a kink-antikink soliton in light-cone coordinates, shown between $t = -3$ and $t = 0$. (b) The initial, identified, and correct coefficients when assuming the relation $r = q^*$, i.e., the correct relation for the fNLSE. (c) The conserved quantities at the identified coefficients for each of the considered relations $q(r)$.

6.4.5. CASE 5: NOISELESS VISCOUS BURGERS' EQUATION WITH COMPLEX VISCOSITY COEFFICIENT

Finally, we would like to illustrate the behavior of our algorithm for an equation that does not seem to fit into our AKNS-ansatz. The equation is a viscous Burgers' equation with a complex viscosity coefficient:

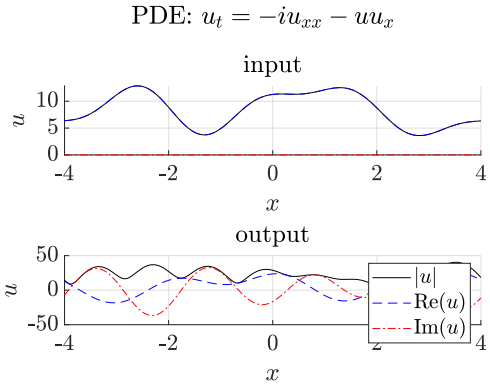
$$u_t + uu_x = \nu u_{xx}, \quad \nu = -i. \tag{6.14}$$

A single realization of the signal is shown in Fig. 6.6a, which was propagated from $t = 0$ to $t = 1$. In total, we generated 41 sets of input-output signals. No noise was added to the signals.

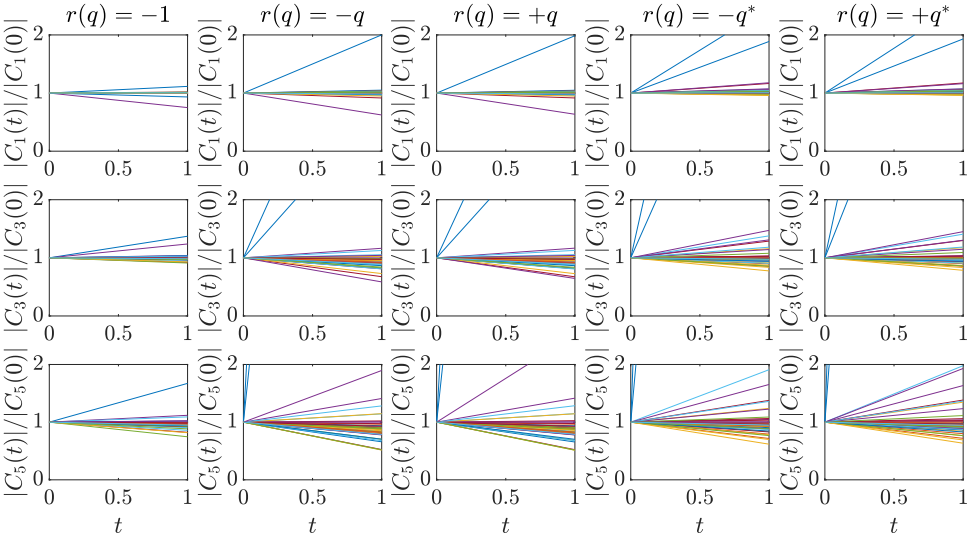
As we apply our algorithm on this data set, we observe in Fig. 6.6b that none of the considered choices for $r(q)$ result in well-conserved quantities. This suggests that this dataset may not fit any of the considered integrable AKNS-type PDEs.

However, the poor conservation of the quantities (6.5) can also arise for other reasons. It could be due to noise, or because the considered space of functions for q and r is not large enough. It could also arise when the starting points in the space for $q(u)$ were not close enough to the correct solution. A large error alone therefore does not necessarily imply that the data does not originate from an AKNS-type system (or a system close to one) in the considered search space, and further investigations are required to eliminate the alternative causes discussed above.

We finally remark that on the other hand, even a perfect conservation of the quantities (6.5) does not guarantee that the system is from the considered class of AKNS systems because only a finite number of conserved quantities is considered for the computation of the error. In such cases, the identified $r(q)$ and $q(u)$ should be further tested w.r.t. other criteria. For example, one could check numerically if the spectrum of the identified L-operator stays constant during propagation. Another possibility would be to determine the PDE that corresponds to the identified Lax pair, and use it to numerically propagate the inputs in the data set. The result should match the corresponding outputs in the data set.



(a) A single realization of an input ($t = 0$) and output ($t = 1$) signal.



(b) Global quantities at the optimal $q(u)$ for various hierarchies $r(q)$.

Figure 6.6: Results for the viscous Burgers equation $u_t + uu_x = \nu u_{xx}$ with complex viscosity coefficient $\nu = -i$

.

6.5. CONCLUSION

We proposed and demonstrated an automatic, data-driven identification method for the spectral operator of AKNS-type Lax pairs. Given measured data from an unknown, not necessarily Lax-integrable PDE, it finds the spectral operator L of the Lax pair that explains the given data best by matching conserved quantities. To the best of our knowledge, it is the first such method. It enables non-specialists to discover Lax-integrable systems "in the wild", and paves the way to exploiting the strong theoretical properties of Lax-integrable PDEs in new application areas. In contrast to conventional PDE identification methods, the measurements can be taken at time points that are far apart (assuming time is the evolution variable), which is an important practical advantage. Only two different time points are required. We focused on finding the L operator of the Lax pair because together with the easily measurable linear dispersion relation of the underlying PDE, it already completely specifies the Lax-integrable PDE that belongs to the Lax pair, and therefore also the propagation operator A, which is the second part of the Lax pair. The spectral operator is furthermore already sufficient to perform nonlinear Fourier analysis.

The method was demonstrated on noisy measurements from the MKdV, the NLSE and the sine-Gordon equation, as well as on noise-free and noisy data from a transformed KdV. By choosing initial guesses for the coefficient vectors that evoke different parts of the conservation laws with similar strength, and then performing searches in low-dimensional spaces of k-sparse vectors, we were able to identify the correct spectral operator for a variety of (nearly) Lax-integrable systems. We found that the algorithm generally works well if the amount of noise in the data is less than 1% of the signal amplitude, although one case required less noise. We remark that PDE identification methods in general require relatively high signal-to-noise ratios so that e.g. higher-order derivatives of the data can be computed. As a next step, the method should be investigated with real-world data. If the noises in such scenarios turn out to be too strong, approaches such as filtering and/or weak formulations should be incorporated to reduce the impact of the noise on the identification process. Additionally, as discussed in the last example, methods for taking the different noise amplification properties of the library functions into account should be investigated.

7

CONCLUSIONS AND RECOMMENDATIONS

Within this thesis, we investigated data-driven parameter identification of Lax-integrable partial differential equations, with the goal to find a Lax-integrable partial differential equation that explains given experimental measurement data best. Being able to associate a Lax-integrable system to given measurement data allows the underlying system to be thoroughly analyzed using the nonlinear Fourier transform (NFT), a.k.a. the method of forward scattering. While many practical systems will not be exactly Lax-integrable due to the presence of noise, loss or other imperfections, many systems are often still approximated well by Lax-integrable equations. Practical examples of systems that are well approximated by Lax-integrable partial differential equations include fiber optical wave propagation modeled by the nonlinear Schrödinger equation (NLSE), surface wave propagation in shallow water canals modeled by the Korteweg-de Vries equation (KdV) and mechanical wave propagation in a chain of couples pendulums modeled by the sine-Gordon equation. It is thus often possible and beneficial if a Lax-integrable description can be found for an experimental system, but few data-driven Lax-pair identification methods existed before the methods presented within this thesis. Due to the vast space of integrable models, as well as the complexity of many of these models, this thesis is limited to the parameter identification for Lax-integrable partial differential equations of the AKNS-type, which already cover a large class of practically relevant systems.

The first part of this thesis is dedicated to the identification of coefficients for individual Lax-integrable PDEs. The proposed new method focused on the nonlinear Fourier spectrum, of which the eigenvalues remain constant over time, while the scattering coefficients evolve linearly, given that the right parametrization was selected. We validated this method for the KdV equation and the NLSE in both simulations and experiments, in particular by comparing the solitons present at two snapshots.

Chapter 2 considered an approximately KdV-governed system with unknown parameters, and showed that it is indeed possible to determine the normalization constant - directly related to the ratio between linear and nonlinear terms - from just two distant snapshots of wave data. From the normalization constant, the depth of the canal can be directly derived. One of the main strengths of the method is that numerical propagation

is not necessary. Furthermore, the normalization constant and propagation distance can be identified separately, which reduces the complexity with respect to propagation-based methods that must identify these coefficients simultaneously. Finally, we also showed that the normalization constant may also be roughly identified by comparing the first few conserved quantities of the KdV.

Chapter 3 and Chapter 4 considered a similar problem, but for the NLSE applied to optical single-mode fibers. The main parameter of interest is the normalization coefficient - again the ratio between the linear and nonlinear effects. The normalization coefficient can be translated into a nonlinearity coefficient, which has become increasingly relevant for digital signal processing. Similar to the KdV, we showed that the normalization constant can be determined by comparing the soliton content from two snapshots, using the NFT. We showed that the continuous spectrum could also be used, but for focussing single-mode fibers (often used in practice) the discrete spectrum (solitons) was shown to be more reliable in the considered cases. As with KdV, no numerical signal propagation is required, as well as that the normalization and the dispersion coefficient can be determined separately. We also quantitatively compared the proposed NFT-based identification method to a propagation-based method in terms of both accuracy and speed. It turned out that the NFT-based identification method could indeed be faster than the propagation based method for long propagation distances, at which numerical propagation requires more time, whereas the calculation of the solitons does not scale with the optical fiber length. However, the propagation based method in general showed better accuracy, which was also due to the fact that the loss and amplification could be taken into account in the propagation, while this can only be approximated in the NFT-based method.

After comparing the speed of the NFT-based propagation to split-step method propagation in 4, we further investigated methods to speed up the discrete spectrum part of the NLSE-NFT in Chapter 5. Faster calculation of the highest eigenvalues in the discrete spectrum would be of great benefit to the identification method in Chapter 4, but also for many other applications where NLSE-type solitons are analyzed. In Chapter 5, we started by noting that the calculation of the discrete spectrum of the NLSE scaled superlinearly (typically quadratic for most fast methods) in the time-bandwidth product of a signal. It was then shown that the highest eigenvalues of the full signal could be calculated by windowing the full signal in time and frequency, and inspecting the highest eigenvalues of each window. The lower eigenvalues are typically affected due to the windowing, but the resulting spurious eigenvalues could efficiently be removed by thresholding the lower eigenvalues based on the window size. While the lowest eigenvalues could thus not be detected, these only contain a small fraction of the total energy. For longer signals, we can simply use more windows (without changing the time-bandwidth product of each window), and thus this windowing method is linear in the time-bandwidth product of the full signal. The windowing process does distort the lower eigenvalues, so this method is best suited for the higher eigenvalues.

In Chapter 6, we focused again on parameter identification, but now for the Lax pairs directly instead of in the PDE by considering conserved quantities. This allowed to consider a wide range of AKNS-type PDEs simultaneously instead of a single AKNS-type PDE as in the earlier chapters, and efficiently distinguish between all PDEs in the considered

class, and find the most suitable Lax pair. We already demonstrated that the normalization parameter of the KdV (in Chapter 2) and the NLSE (in Chapter 3) could be quickly determined by comparing conserved quantities. We showed that these ideas could be further extended to a variety of AKNS-type PDEs, as well as that many PDEs could be distinguished in this way. We compared five of the most ubiquitous choices for the spectral Lax operator of AKNS-type, and compared which choice would best conserve the AKNS-type conserved quantities. For sufficiently high signal-to-noise ratios, indeed this method could accurately identify the spectral operator, including the correct normalization. One of the main benefits was that only two snapshots were required at two distant locations. The derivative in the direction of propagation was thus not required at all, as opposed to many other identification methods.

In conclusion, we have proposed two new methods of identifying Lax-integrable PDEs of AKNS type: one for finding PDE coefficients by comparing the NFT spectra, and another one for finding Lax pairs by comparing the conserved quantities. The spectral method can use all information of the signal, and can therefore be more accurate, especially in the presence of noise. However, the computational cost of the calculating the NFT spectrum is often the limiting factor. In many cases, split-step propagation may yield a similar or more accurate answer with less computational effort. Spectral based identification methods typically do not scale with propagation distance as opposed to split-step methods, and thus spectral methods may be the method of choice when analyzing signals that have propagated relatively far. NFT-based methods and split-step methods can both be used for propagation and identification, and it is often beneficial to consider which method is more suitable for a given problem. On the other hand, our proposed method comparing conserved quantities does not rely at all on spectral transformations, but condenses the information of a full signal into only a few conserved quantities. Consequently, this method is very fast, but discards much information. As a result these type of methods are typically less accurate and more noise sensitive. Both of our proposed identification methods thus have their advantages and drawbacks, but can also be combined to exploit the advantages of both as demonstrated in this thesis.

7.1. RECOMMENDATIONS

Within this thesis we have mainly focused on developing methods for PDEs for AKNS-type hierarchies, due to the relevance of these PDEs in engineering, but also due to their shared structure, allowing for overarching approaches for PDEs within this hierarchy. For future research, we mainly recommend applying the developed techniques to a PDEs of other Lax-integrable PDEs or PDE hierarchies. We will give several option below.

The coupled nonlinear Schrödinger equation. The coupled nonlinear Schrödinger equation [166] is a generalization of the nonlinear Schrödinger equations, which, among others, describes the propagation of light pulses through an optical fiber when the light inside has not been polarized. In comparison, the normal NLSE assumes polarized light, such that the field has only a single polarization. In fiber optical communication, both polarizations can be used for data transmission, so in practice the coupled NLSE is often of great interest. Fortunately, the coupled NLSE is also a Lax-integrable PDE, although it falls outside of the class of AKNS-type PDEs. However, a Lax-pair is known for the

coupled NLSE, which results in the Manakov system, and we therefore expect many of the techniques developed throughout this thesis to be applicable to the coupled NLSE.

The Boussinesq equation. The Boussinesq equation is a nonlinear PDE that generalizes the KdV equation, and can be used to model waves in shallow water canals, with waves traveling in opposite direction, as opposed to the KdV in which all waves travel in one direction. The Boussinesq equation is also Lax-integrable, but also falls outside the class of AKNS-type PDEs [10]. It would be interesting to investigate whether the techniques developed in this paper could also be applied to identify parameters of the Boussinesq equation. Furthermore, this could also lead to interesting questions such as that whether we are able to distinguish left-going waves from right-going waves, just by considering time-series from just a few distant wave gauges.

Higher-dimensional PDEs Another possible research topic is the application of our developed to higher-dimensional systems, such as the Kadomtsev–Petviashvili (KP) equation (two space dimensions). While the coupled NLSE and Boussinesq are significantly harder to analyze than AKNS-type systems due to their two-dimensional fields, the space they propagate in is still one-dimensional (i.e., a channel). It would be interesting to investigate whether parameter identification by use of the NFT or conserved quantities is also possible for Lax-integrable PDEs on higher-dimensional spaces.

Generalization of Lax-pair identification based on conserved quantities Finally, one of the most interesting questions is whether it is possible to systematically identify Lax-integrable PDEs by a more general method than fitting every known Lax-integrable PDE to measurement data. We already made an attempt at this by using the same method to identify Lax-pairs of AKNS type by considering their conserved quantities, but it would be interesting to investigate if this method can be generalized or applied to other hierarchies of integrable PDEs. We envision an identification method in which a large class of Lax-integrable PDEs can be identified by comparing the conserved quantities for each of several sub-class, of which the AKNS-hierarchy is the first. Each sub-class could then have its own method to calculate and compare the conserved quantities associated with it. While an individual Lax-integrable PDE could be its own sub-class, it would be beneficial if multiple integrable PDEs could be generalized under the same sub-class.

BIBLIOGRAPHY

[1] C. S. Gardner, J. M. Greene, M. D. Kruskal, and R. M. Miura. “Method for solving the Korteweg-deVries equation”. In: *Physical review letters* 19.19 (1967), p. 1095.

[2] D. J. Korteweg and G. De Vries. “On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves”. In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 39.240 (1895), pp. 422–443.

[3] J. Boussinesq. “Théorie des ondes et des remous qui se propagent le long d’un canal rectangulaire horizontal, en communiquant au liquide contenu dans ce canal des vitesses sensiblement pareilles de la surface au fond”. In: *Journal de mathématiques pures et appliquées* 17 (1872), pp. 55–108.

[4] J. S. Russell. *Report on waves. Report of the fourteenth meeting of the British Association for the Advancement of Science*. 1845.

[5] N. J. Zabusky and M. D. Kruskal. “Interaction of" solitons" in a collisionless plasma and the recurrence of initial states”. In: *Physical review letters* 15.6 (1965), p. 240.

[6] A. Kasman. “A brief history of solitons and the KdV equation”. In: *Current Science* 115.8 (2018), pp. 1486–1496.

[7] M. J. Ablowitz, D. J. Kaup, A. C. Newell, and H. Segur. “The inverse scattering transform-Fourier analysis for nonlinear problems”. In: *Studies in Applied Mathematics* 53.4 (1974), pp. 249–315.

[8] P. D. Lax. “Integrals of nonlinear equations of evolution and solitary waves”. In: *Communications on pure and applied mathematics* 21.5 (1968), pp. 467–490.

[9] V. Zakharov and A. Shabat. “Exact theory of two-dimensional self-focusing and one-dimensional self-modulation of waves in nonlinear media”. In: *Sov. Phys. JETP* 34.1 (1972), p. 62.

[10] P. Deift, C. Tomei, and E. Trubowitz. “Inverse scattering and the Boussinesq equation”. In: *Communications on Pure and Applied Mathematics* 35.5 (1982), pp. 567–628.

[11] F. Gesztesy and K. Unterkofler. “Isospectral deformations for Strum-Liouville and Dirac-type operators and associated nonlinear evolution equations”. In: *Reports on mathematical physics* 31.2 (1992), pp. 113–137.

[12] R. Hirota and J. Satsuma. “N-soliton solutions of model equations for shallow water waves”. In: *Journal of the Physical Society of Japan* 40.2 (1976), pp. 611–612.

[13] M. J. Ablowitz, M. Ablowitz, and P. A. Clarkson. *Solitons, nonlinear evolution equations and inverse scattering*. Vol. 149. Cambridge university press, 1991.

[14] B. Fuchssteiner. “Some tricks from the symmetry-toolbox for nonlinear equations: generalizations of the Camassa-Holm equation”. In: *Physica D: Nonlinear Phenomena* 95.3-4 (1996), pp. 229–243.

[15] J. Larue. “Symbolic verification of operator and matrix Lax pairs for some completely integrable nonlinear partial differential equations”. In: *2011-Mines Theses & Dissertations* (2011).

[16] A.-M. Wazwaz. “Multiple-soliton solutions for the KP equation by Hirota’s bilinear method and by the tanh–coth method”. In: *Applied Mathematics and Computation* 190.1 (2007), pp. 633–640.

[17] S. V. Manakov. “On the theory of two-dimensional stationary self-focusing of electromagnetic waves”. In: *Soviet Physics-JETP* 38.2 (1974), pp. 248–253.

[18] J. Ford, S. D. Stoddard, and J. S. Turner. “On the integrability of the Toda lattice”. In: *Progress of Theoretical Physics* 50.5 (1973), pp. 1547–1560.

[19] M. Musette and R. Conte. “Algorithmic method for deriving Lax pairs from the invariant Painlevé analysis of nonlinear partial differential equations”. In: *Journal of mathematical physics* 32.6 (1991), pp. 1450–1457.

[20] H. Chen, Y. Lee, and C. Liu. “Integrability of nonlinear Hamiltonian systems by inverse scattering method”. In: *Physica Scripta* 20.3-4 (1979), p. 490.

[21] I. Habibullin, A. Khakimova, and M. Poptsova. “On a method for constructing the Lax pairs for nonlinear integrable equations”. In: *Journal of Physics A: Mathematical and Theoretical* 49.3 (2015), p. 035202.

[22] V. E. Zakharov *et al. What is integrability?* Vol. 129. Springer, 1991.

[23] S. K. Turitsyn, J. E. Prilepsky, S. T. Le, S. Wahls, L. L. Frumin, M. Kamalian, and S. A. Derevyanko. “Nonlinear Fourier transform for optical data processing and transmission: advances and perspectives”. In: *Optica* 4.3 (2017), pp. 307–322.

[24] A. R. Osborne. “Nonlinear ocean wave and the inverse scattering transform”. In: *Scattering*. Elsevier, 2002, pp. 637–666.

[25] P. D. Lax. “Periodic solutions of the KdV equation”. In: *Communications on pure and applied mathematics* 28.1 (1975), pp. 141–188.

[26] B. Prinari, M. J. Ablowitz, and G. Biondini. “Inverse scattering transform for the vector nonlinear Schrödinger equation with nonvanishing boundary conditions”. In: *Journal of mathematical physics* 47.6 (2006).

[27] G. Boffetta and A. R. Osborne. “Computation of the direct scattering transform for the nonlinear Schrödinger equation”. In: *Journal of computational physics* 102.2 (1992), pp. 252–264.

[28] M. I. Yousefi and F. R. Kschischang. “Information transmission using the nonlinear Fourier transform, Part I: Mathematical tools”. In: *IEEE Transactions on Information Theory* 60.7 (2014), pp. 4312–4328.

[29] S. Chimmalgi, P. J. Prins, and S. Wahls. “Fast nonlinear Fourier transform algorithms using higher order exponential integrators”. In: *IEEE Access* 7 (2019), pp. 145161–145176.

[30] S. Wahls and H. V. Poor. “Fast numerical nonlinear Fourier transforms”. In: *IEEE Transactions on Information Theory* 61.12 (2015), pp. 6957–6974.

[31] A. Vasylchenkova, J. E. Prilepsky, D. Shepelsky, and A. Chattopadhyay. “Direct nonlinear Fourier transform algorithms for the computation of solitonic spectra in focusing nonlinear Schrödinger equation”. In: *Communications in Nonlinear Science and Numerical Simulation* 68 (2019), pp. 347–371.

[32] P. J. Prins and S. Wahls. “An accurate O (N2) floating point algorithm for the Crum transform of the KdV equation”. In: *Communications in Nonlinear Science and Numerical Simulation* 102 (2021), p. 105782.

[33] P. J. Prins. “Reliable numerical algorithms for the Non-linear Fourier Transform of the KdV equation”. PhD thesis. Delft University of Technology, 2022.

[34] P. J. Prins and S. Wahls. “Higher order exponential splittings for the fast non-linear Fourier transform of the Korteweg-de Vries equation”. In: *2018 IEEE International conference on acoustics, speech and signal processing (ICASSP)*. IEEE. 2018, pp. 4524–4528.

[35] S. Wahls, S. Chimmalgi, and P. J. Prins. “FNFT: a software library for computing nonlinear Fourier transforms”. In: *Journal of Open Source Software* 3.23 (2018), p. 597.

[36] A. Hasegawa and F. Tappert. “Transmission of stationary nonlinear optical pulses in dispersive dielectric fibers. II. Normal dispersion”. In: *Applied Physics Letters* 23.4 (1973), pp. 171–172.

[37] P. J. Prins and S. Wahls. “Soliton phase shift calculation for the Korteweg-de Vries equation”. In: *IEEE Access* 7 (2019), pp. 122914–122930.

[38] I. Gelfand and B. Levitan. “Reconstruction of the differential equation by its spectral function”. In: *Izv. Akad. Nauk. SSSR, Ser. Mat* 15.4 (1951), pp. 309–360.

[39] R. M. Miura, C. S. Gardner, and M. D. Kruskal. “Korteweg-de Vries equation and generalizations. II. Existence of conservation laws and constants of motion”. In: *Journal of Mathematical physics* 9.8 (1968), pp. 1204–1209.

[40] P. Scholl, A. Bacho, H. Boche, and G. Kutyniok. “Well-definedness of Physical Law Learning: The Uniqueness Problem”. In: *arXiv preprint arXiv:2210.08342* (2022).

[41] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz. “Data-driven discovery of partial differential equations”. In: *Science advances* 3.4 (2017), e1602614.

[42] S. Krippendorf, D. Lüst, and M. Syvaeri. “Integrability Ex Machina”. In: *Fortschritte der Physik* 69.7 (2021), p. 2100057.

[43] P. de Koster, M. Brühl, and S. Wahls. “Water-depth identification from free-surface data using the KdV-based nonlinear Fourier transform”. In: *International Conference on Offshore Mechanics and Arctic Engineering*. Vol. 85901. American Society of Mechanical Engineers. 2022, V05BT06A056.

[44] M. W. Dingemans. *Water wave propagation over uneven bottoms: Linear wave propagation*. Vol. 13. World Scientific, 1997.

[45] J. L. Hammack and H. Segur. “The Korteweg-de Vries equation and water waves. Part 2. Comparison with experiments”. In: *Journal of Fluid mechanics* 65.2 (1974), pp. 289–314.

[46] D. Crighton. “Applications of KdV”. In: *KdV’95*. Springer, 1995, pp. 39–67.

[47] L. E. Frostick, S. J. McLelland, and T. G. Mercer. *Users guide to physical modelling and experimentation: Experience of the HYDRALAB network*. CRC Press, 2019.

[48] H. E. Williams, R. Briganti, A. Romano, and N. Dodd. “Experimental analysis of wave overtopping: A new small scale laboratory dataset for the assessment of uncertainty for smooth sloped and vertical coastal structures”. In: *Journal of Marine Science and Engineering* 7.7 (2019), p. 217.

[49] B. Jacob and E. V. Stanev. “Understanding the Impact of Bathymetric Changes in the German Bight on Coastal Hydrodynamics: One Step Toward Realistic Morphodynamic Modeling”. In: *Frontiers in Marine Science* 8 (2021), p. 576.

[50] I. Grabemann, L. Gaslikova, T. Brodhagen, and E. Rudolph. “Extreme storm tides in the German Bight (North Sea) and their potential for amplification”. In: *Natural Hazards and Earth System Sciences* 20.7 (2020), pp. 1985–2000.

[51] S. Mitchell, J. P. Thayer, and M. Hayman. “Polarization lidar for shallow water depth measurement”. In: *Applied optics* 49.36 (2010), pp. 6995–7000.

[52] F. Polcyn, W. L. Brown, and I. Sattinger. *The measurement of water depth by remote sensing techniques*. Tech. rep. Michigan University Ann Arbor intitute of Science and Technology, 1970.

[53] V. Vasan, Manisha, and D. Auroux. “Ocean-depth measurement using shallow-water wave models”. In: *Studies in Applied Mathematics* 147.4 (2021), pp. 1481–1518. DOI: https://doi.org/10.1111/sapm.12418. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/sapm.12418. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/sapm.12418.

[54] P. de Koster and S. Wahls. “Dispersion and Nonlinearity Identification for Single-Mode Fibers Using the Nonlinear Fourier Transform”. In: *Journal of Lightwave Technology* 38.12 (2020), pp. 3252–3260.

[55] P. de Koster, J. Koch, O. Schulz, S. Pachnicke, and S. Wahls. “Experimental validation of nonlinear Fourier transform-based Kerr-nonlinearity identification over a 1600 km SSMF link”. In: *Optical Fiber Communication Conference*. Optica Publishing Group. 2022, W2A–39.

[56] A. Osborne. “Nonlinear Fourier analysis for the infinite-interval Korteweg-de Vries equation I: An algorithm for the direct scattering transform”. In: *Journal of Computational Physics* 94.2 (1991), pp. 284–313.

[57] P. C. Schuur. *Asymptotic analysis of soliton problems: an inverse scattering approach*. Vol. 1232. Springer, 2006.

[58] M. Bruehl, P. Prins, S. Ujvary, I. Barranco, S. Wahls, and P. Liu. “Comparative analysis of bore propagation over long distances using conventional linear and KdV-based nonlinear Fourier transform”. In: *UNDER REVIEW* 1.1 (202), pp. 1–50.

[59] T. A. Driscoll, N. Hale, and L. N. Trefethen. *Chebfun guide*. 2014.

[60] V. E. Zakharov and L. D. Faddeev. "Korteweg-de Vries equation: A completely integrable Hamiltonian system". In: *Funktsional'nyi Analiz i ego Prilozheniya* 5.4 (1971), pp. 18–27.

[61] M. J. Ablowitz and H. Segur. *Solitons and the inverse scattering transform*. SIAM, 1981.

[62] G. P. Agrawal. *Fiber-optic communication systems*. Vol. 222. John Wiley & Sons, 2012.

[63] E. Agrell, M. Karlsson, A. Chraplyvy, D. J. Richardson, P. M. Krummrich, P. Winzer, K. Roberts, J. K. Fischer, S. J. Savory, B. J. Eggleton, *et al.* "Roadmap of optical communications". In: *Journal of Optics* 18.6 (2016), p. 063002.

[64] P. J. Winzer, D. T. Neilson, and A. R. Chraplyvy. "Fiber-optic transmission and networking: the previous 20 and the next 20 years". In: *Optics express* 26.18 (2018), pp. 24190–24239.

[65] P. de Koster and S. Wahls. "Fibre model identification for nonlinear Fourier transform-based transmission". In: *Proceedings of the 45th European Conference on Optical Communication*. IEEE. 2019.

[66] F. N. Hauske, C. Xie, Z. Zhang, C. Li, L. Li, and Q. Xiong. "Frequency domain chromatic dispersion estimation". In: *2010 Conference on Optical Fiber Communication (OFC/NFOEC), collocated National Fiber Optic Engineers Conference*. IEEE. 2010, pp. 1–3.

[67] Q. Sui, A. P. T. Lau, and C. Lu. "Fast and robust blind chromatic dispersion estimation using auto-correlation of signal power waveform for digital coherent systems". In: *Journal of Lightwave Technology* 31.2 (2013), pp. 306–312.

[68] C.-Y. Lin, A. Napoli, B. Spinnler, V. Sleiffer, D. Rafique, M. Kuschnerov, M. Bohn, and B. Schmauss. "Adaptive digital back-propagation for optical communication systems". In: *Optical Fiber Communication Conference*. Optical Society of America. 2014, pp. M3C–4.

[69] M. Piels, E. P. da Silva, D. Zibar, and R. Borkowski. "Performance emulation and parameter estimation for nonlinear fibre-optic links". In: *2016 21st European Conference on Networks and Optical Communications (NOC)*. IEEE. 2016, pp. 1–5.

[70] L. Prigent and J.-P. Hamaide. "Measurement of fiber nonlinear Kerr coefficient by four-wave mixing". In: *IEEE Photonics Technology Letters* 5.9 (1993), pp. 1092–1095.

[71] L. Jiang, L. Yan, A. Yi, Y. Pan, M. Hao, W. Pan, B. Luo, and Y. Jaouën. "Chromatic Dispersion, Nonlinear Parameter, and Modulation Format Monitoring Based on Godard's Error for Coherent Optical Transmission Systems". In: *IEEE Photonics Journal* 10.1 (2018), pp. 1–12.

[72] Y. Namihira, A. Miyata, and N. Tanahashi. "Nonlinear coefficient measurements for dispersion shifted fibres using self-phase modulation method at 1.55 μm". In: *Electronics Letters* 30.14 (1994), pp. 1171–1172.

[73] A. Boskovic, S. Chernikov, J. Taylor, L. Gruner-Nielsen, and O. Levring. “Direct continuous-wave measurement of n_2 in various types of telecommunication fiber at 1.55 µm”. In: *Optics letters* 21.24 (1996), pp. 1966–1968.

[74] P. André and J. Pinto. “Simultaneous measurement of the nonlinear refractive index and chromatic dispersion of optical fibers by four-wave mixing”. In: *Microwave and Optical Technology Letters* 34.4 (2002), pp. 305–307.

[75] S. T. Le, J. E. Prilepsky, and S. K. Turitsyn. “Nonlinear inverse synthesis technique for optical links with lumped amplification”. In: *Optics express* 23.7 (2015), pp. 8317–8328.

[76] M. I. Yousefi and F. R. Kschischang. “Information transmission using the nonlinear Fourier transform, Part III: Spectrum modulation”. In: *IEEE Transactions on Information Theory* 60.7 (2014), pp. 4346–4369.

[77] X. Yangzhang, V. Aref, S. T. Le, H. Bülow, and P. Bayvel. “400 Gbps dual-polarisation non-linear frequency-division multiplexed transmission with b-modulation”. In: *2018 European Conference on Optical Communication (ECOC)*. IEEE. 2018, pp. 1–3.

[78] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright. “Convergence properties of the Nelder–Mead simplex method in low dimensions”. In: *SIAM Journal on optimization* 9.1 (1998), pp. 112–147.

[79] J. Satsuma and N. Yajima. “B. Initial value problems of one-dimensional self-modulation of nonlinear waves in dispersive media”. In: *Progress of Theoretical Physics Supplement* 55 (1974), pp. 284–306.

[80] E. Fagerstrom. “On the nonlinear Schrodinger equation with nonzero boundary conditions”. PhD thesis. University at Buffalo, State University of New York, 2015.

[81] M. Klaus and J. Shaw. “On the eigenvalues of Zakharov–Shabat systems”. In: *SIAM journal on mathematical analysis* 34.4 (2003), pp. 759–773.

[82] A. Hasegawa and Y. Kodama. “Guiding-center soliton in optical fibers”. In: *Optics letters* 15.24 (1990), pp. 1443–1445.

[83] H. Buelow, V. Aref, and W. Idler. “Transmission of waveforms determined by 7 eigenvalues with PSK-modulated spectral amplitudes”. In: *ECOC 2016; 42nd European Conference on Optical Communication*. VDE. 2016, pp. 1–3.

[84] H. W. Kuhn. “The Hungarian method for the assignment problem”. In: *Naval research logistics quarterly* 2.1-2 (1955), pp. 83–97.

[85] M. Brehler, C. Mahnke, S. Chimmalgi, and S. Wahls. “NFDMLab: Simulating Nonlinear Frequency Division Multiplexing in Python”. In: *Optical Fiber Communication Conference*. Optical Society of America. 2019, M3Z–13.

[86] S. Turitsyn, E. Sedov, A. Redyuk, and M. Fedoruk. “Nonlinear spectrum of conventional OFDM and WDM return-to-zero signals in nonlinear channel”. In: *Journal of Lightwave Technology* 38.2 (2019), pp. 352–358.

[87] P. de Koster, O. Schulz, J. Koch, S. Pachnicke, and S. Wahls. “Fast Single-Mode Fiber Nonlinearity Monitoring: An Experimental Comparison Between Split-Step and Nonlinear Fourier Transform-Based Methods”. In: *IEEE Photonics Journal* (2023).

[88] M. Secondini, D. Marsella, and E. Forestieri. “Enhanced split-step Fourier method for digital backpropagation”. In: *2014 The European Conference on Optical Communication (ECOC)*. IEEE. 2014, pp. 1–3.

[89] R. Dar and P. J. Winzer. “Nonlinear interference mitigation: Methods and potential gain”. In: *Journal of Lightwave Technology* 35.4 (2017), pp. 903–930.

[90] E. Ip and J. M. Kahn. “Compensation of dispersion and nonlinear impairments using digital backpropagation”. In: *Journal of Lightwave Technology* 26.20 (2008), pp. 3416–3425.

[91] R.-J. Essiambre, G. Kramer, P. J. Winzer, G. J. Foschini, and B. Goebel. “Capacity limits of optical fiber networks”. In: *Journal of Lightwave Technology* 28.4 (2010), pp. 662–701.

[92] T. Kato, Y. Suetsugu, M. Takagi, E. Sasaoka, and M. Nishimura. “Measurement of the nonlinear refractive index in optical fiber by the cross-phase-modulation method with depolarized pump light”. In: *Optics Letters* 20.9 (1995), pp. 988–990.

[93] J. K. Andersen, J. W. Lou, G. A. Nowak, T. Xia, M. N. Islam, R. M. Fortenberry, and S. A. Newton. “Path average measurements of optical fiber nonlinearity using solitons”. In: *Journal of lightwave technology* 16.12 (1998), p. 2328.

[94] P. de Koster, J. Koch, S. Pachnicke, and S. Wahls. “Experimental Investigation of Nonlinear Fourier Transform Based Fibre Nonlinearity Characterisation”. In: *2021 European Conference on Optical Communication (ECOC)*. IEEE. 2021, pp. 1–4.

[95] G. P. Agrawal. “Nonlinear fiber optics, 5th edition”. In: *Nonlinear fiber optics*. Elsevier, 2010.

[96] R. Pleban, A. Azari, R. Salem, and T. E. Murphy. *SSPROP*. https://photonics.umd.edu/software/ssprop/. Accessed: 06-2022.

[97] S. Wahls, S. T. Le, J. E. Prilepsk, H. V. Poor, and S. K. Turitsyn. “Digital backpropagation in the nonlinear Fourier domain”. In: *2015 IEEE 16th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE. 2015, pp. 445–449.

[98] S. Chimmalgi and S. Wahls. “Bounds on the transmit power of b-modulated NFDM systems in anomalous dispersion fiber”. In: *Entropy* 22.6 (2020), p. 639.

[99] T. Gui, G. Zhou, C. Lu, A. P. T. Lau, and S. Wahls. “Nonlinear frequency division multiplexing with b-modulation: shifting the energy barrier”. In: *Optics express* 26.21 (2018), pp. 27978–27990.

[100] S. Pachnicke. *Fiber-optic transmission networks: efficient design and dynamic operation*. Springer, 2011.

[101] P. Kabaciński, T. M. Kardaś, Y. Stepanenko, and C. Radzewicz. “Nonlinear refractive index measurement by SPM-induced phase regression”. In: *Optics express* 27.8 (2019), pp. 11018–11028.

[102] T. D. De Menezes, C. Tu, V. Besse, M. O’Sullivan, V. S. Grigoryan, C. R. Menyuk, and I. T. Lima Jr. “Nonlinear Spectrum Modulation in the Anomalous Dispersion Regime Using Second-and Third-Order Solitons”. In: *Photonics*. Vol. 9. 10. MDPI. 2022, p. 748.

[103] P. de Koster and S. Wahls. “Fast and Reliable Detection of Significant Solitons in Signals with Large Time-Bandwidth Products”. In: *Journal of Lightwave Technology* (2023).

[104] L. Liu, L. Li, Y. Huang, K. Cui, Q. Xiong, F. N. Hauske, C. Xie, and Y. Cai. “Intrachannel nonlinearity compensation by inverse Volterra series transfer function”. In: *Journal of Lightwave Technology* 30.3 (2011), pp. 310–316.

[105] G. P. Agrawal. *Fiber-optic communication systems, 5th Edition*. John Wiley, 2021.

[106] A. C. Scott, F. Chu, and D. W. McLaughlin. “The soliton: a new concept in applied science”. In: *Proceedings of the IEEE* 61.10 (1973), pp. 1443–1483.

[107] F. Mitschke, C. Mahnke, and A. Hause. “Soliton content of fiber-optic light pulses”. In: *Applied Sciences* 7.6 (2017), p. 635.

[108] V. Zakharov and A. Shabat. “Exact theory of two-dimensional self-focusing and one-dimensional self-modulation of waves in nonlinear media”. In: *Soviet physics JETP* 34.1 (1972), p. 62.

[109] G. Zhou, L. Sun, C. Lu, and A. P. T. Lau. “Multi-symbol digital signal processing techniques for discrete eigenvalue transmissions based on nonlinear Fourier transform”. In: *Journal of Lightwave technology* 39.17 (2021), pp. 5459–5467.

[110] S. Wahls. “Shortening solitons for fiber-optic transmission”. In: *2021 17th International Symposium on Wireless Communication Systems (ISWCS)*. IEEE. 2021, pp. 1–6.

[111] Z. Dong, S. Hari, T. Gui, K. Zhong, M. I. Yousefi, C. Lu, P.-K. A. Wai, F. R. Kschischang, and A. P. T. Lau. “Nonlinear frequency division multiplexed transmissions based on NFT”. In: *IEEE Photonics Technology Letters* 27.15 (2015), pp. 1621–1623.

[112] S. T. Le, V. Aref, and H. Buelow. “Nonlinear signal multiplexing for communication beyond the Kerr nonlinearity limit”. In: *Nature Photonics* 11.9 (2017), pp. 570–576.

[113] V. Aref, S. T. Le, and H. Buelow. “Modulation over nonlinear Fourier spectrum: Continuous and discrete spectrum”. In: *Journal of Lightwave Technology* 36.6 (2018), pp. 1289–1295.

[114] J. Koch, R. Weixer, and S. Pachnicke. “Equalization of soliton transmission based on nonlinear Fourier transform using neural networks”. In: *45th European Conference on Optical Communication (ECOC 2019)*. IET. 2019, pp. 1–4.

[115] G. Zhou, T. Gui, C. Lu, A. P. T. Lau, and P.-K. A. Wai. “Improving soliton transmission systems through soliton interactions”. In: *Journal of Lightwave Technology* 38.14 (2019), pp. 3563–3572.

[116] S. Sugavanam, M. K. Kopae, J. Peng, J. E. Prilepsky, and S. K. Turitsyn. “Analysis of laser radiation using the Nonlinear Fourier transform”. In: *Nature communications* 10.1 (2019), pp. 1–10.

[117] Y. Wang, S. Fu, C. Zhang, X. Tang, J. Kong, J. H. Lee, and L. Zhao. “Soliton distillation of pulses from a fiber laser”. In: *Journal of Lightwave Technology* 39.8 (2021), pp. 2542–2546.

[118] Y. Wang, S. Fu, J. Kong, A. Komarov, M. Klimczak, R. Buczyński, X. Tang, M. Tang, Y. Qin, and L. Zhao. “Nonlinear Fourier transform enabled eigenvalue spectrum investigation for fiber laser radiation”. In: *Photonics Research* 9.8 (2021), pp. 1531–1539.

[119] F. Chen, H. Lan, Y. Wang, M. Klimczak, R. Buczynski, X. Tang, M. Tang, H. Zhu, and L. Zhao. “Characterization of sidebands in fiber lasers based on nonlinear Fourier transformation”. In: *Optics Express* 31.5 (2023), pp. 7554–7563.

[120] S. K. Turitsyn, I. S. Chekhovskoy, and M. P. Fedoruk. “Nonlinear Fourier transform for characterization of the coherent structures in optical microresonators”. In: *Optics Letters* 45.11 (2020), pp. 3059–3062.

[121] J. Pan, T. Huang, Y. Wang, Z. Wu, J. Zhang, and L. Zhao. “Numerical investigations of cavity-soliton distillation in Kerr resonators using the nonlinear Fourier transform”. In: *Physical Review A* 104.4 (2021), p. 043507.

[122] S. Turitsyn, I. Chekhovskoy, and M. Fedoruk. “Nonlinear Fourier transform for analysis of optical spectral combs”. In: *Physical Review E* 103.2 (2021), p. L020202.

[123] M. I. Yousefi and F. R. Kschischang. “Information transmission using the nonlinear Fourier transform, Part II: Numerical methods”. In: *IEEE Transactions on Information Theory* 60.7 (2014), pp. 4329–4345.

[124] J. He, L. Jianping, Y. Qin, N. Lin, X. Yu, Y. He, O. Xu, D. Peng, M. Xiang, G. Zhou, *et al.* “Adaptive trust-region-based algorithm for the discrete eigenvalue evaluation of the direct nonlinear Fourier transform”. In: *Optics Letters* 47.16 (2022), pp. 4195–4198.

[125] S. Wahls and H. V. Poor. “Introducing the fast nonlinear Fourier transform”. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2013, pp. 5780–5784.

[126] A. Vasylchenkova, J. E. Prilepsky, and S. K. Turitsyn. “Contour integrals for numerical computation of discrete eigenvalues in the Zakharov–Shabat problem”. In: *Optics letters* 43.15 (2018), pp. 3690–3693.

[127] E. V. Sedov, I. S. Chekhovskoy, J. E. Prilepsky, and M. P. Fedoruk. “Application of neural networks to determine the discrete spectrum of the direct Zakharov–Shabat problem”. In: *Quantum Electronics* 50.12 (2020), p. 1105.

[128] I. Chekhovskoy, S. B. Medvedev, I. Vaseva, E. Sedov, and M. P. Fedoruk. “Introducing phase jump tracking–a fast method for eigenvalue evaluation of the direct Zakharov-Shabat problem”. In: *Communications in Nonlinear Science and Numerical Simulation* 96 (2021), p. 105718.

[129] V. Aref, S. T. Le, and H. Buelow. “An efficient nonlinear Fourier transform algorithm for detection of eigenvalues from continuous spectrum”. In: *Optical Fiber Communication Conference*. Optica Publishing Group. 2019, pp. M1I–5.

[130] S. Hari, M. I. Yousefi, and F. R. Kschischang. “Multieigenvalue communication”. In: *Journal of Lightwave Technology* 34.13 (2016), pp. 3110–3117.

[131] J. Koch, K. Chan, C. G. Schaeffer, and S. Pachnicke. “Signal processing techniques for optical transmission based on eigenvalue communication”. In: *IEEE Journal of Selected Topics in Quantum Electronics* 27.3 (2020), pp. 1–14.

[132] R. Zhang, L. Xi, J. Wei, J. Deng, S. Du, W. Zhang, X. Zhang, and X. Xiao. “Optimal Design of Eigenvalues for the Full-spectrum Modulated Nonlinear Frequency Division Multiplexing Transmission System”. In: *IEEE Photonics Journal* (2023).

[133] G. Roberti, G. El, A. Tovbis, F. Copie, P. Suret, and S. Randoux. “Numerical spectral synthesis of breather gas for the focusing nonlinear Schrödinger equation”. In: *Physical Review E* 103.4 (2021), p. 042205.

[134] P. Suret, M. Dufour, G. Roberti, G. El, F. Copie, and S. Randoux. “Soliton refraction through an optical soliton gas”. In: *arXiv preprint arXiv:2303.13421* (2023).

[135] S. Randoux, P. Suret, and G. El. “Inverse scattering transform analysis of rogue waves using local periodization procedure”. In: *Scientific reports* 6.1 (2016), pp. 1–11.

[136] A. R. Osborne, M. Onorato, and M. Serio. “The nonlinear dynamics of rogue waves and holes in deep-water gravity wave trains”. In: *Physics Letters A* 275.5-6 (2000), pp. 386–393.

[137] A. Slunyaev. “Analysis of the nonlinear spectrum of intense sea wave with the purpose of extreme wave prediction”. In: *Radiophysics and Quantum Electronics* 61.1 (2018), pp. 1–21.

[138] M. Onorato, L. Cavaleri, S. Randoux, P. Suret, M. I. Ruiz, M. de Alfonso, and A. Benetazzo. “Observation of a giant nonlinear wave-packet on the surface of the ocean”. In: *Scientific Reports* 11.1 (2021), p. 23606.

[139] J. Koch, S. Li, and S. Pachnicke. “Transmission of higher order solitons created by optical multiplexing”. In: *Journal of Lightwave Technology* 37.3 (2018), pp. 933–941.

[140] A. Hasegawa and Y. Kodama. *Solitons in optical communications*. 7. Oxford University Press on Demand, 1995, pp. 151–171.

[141] L. M. Alonso. “Effect of the radiation component on soliton motion”. In: *Physical Review D* 32.6 (1985), p. 1459.

[142] M. Borghese, R. Jenkins, and K. D.-R. McLaughlin. “Long time asymptotic behavior of the focusing nonlinear Schrödinger equation”. In: *Annales de l’Institut Henri Poincaré C, Analyse non linéaire*. Vol. 35. Elsevier. 2018, pp. 887–920.

[143] S. Medvedev, I. Vaseva, I. Chekhovskoy, and M. Fedoruk. “Exponential fourth order schemes for direct Zakharov-Shabat problem”. In: *Optics Express* 28.1 (2020), pp. 20–39.

[144] P. de Koster and S. Wahls. “Data-driven identification of the spectral operator in AKNS Lax pairs using conserved quantities”. In: *Wave Motion* (2024), p. 103273.

[145] M. Wadati. “The modified Korteweg-de Vries equation”. In: *Journal of the Physical Society of Japan* 34.5 (1973), pp. 1289–1296.

[146] A. R. Osborne. “Nonlinear Ocean Waves and the Inverse Scattering Transform”. In: vol. 97. International Geophysics. Academic Press, 2010.

[147] S. Trillo, G. Deng, G. Biondini, M. Klein, G. Clauss, A. Chabchoub, and M. Onorato. “Experimental observation and theoretical description of multisoliton fission in shallow water”. In: *Physical review letters* 117.14 (2016), p. 144102.

[148] M. Brühl and H. Oumeraci. “Analysis of long-period cosine-wave dispersion in very shallow water using nonlinear Fourier transform based on KdV equation”. In: *Applied Ocean Research* 61 (2016), pp. 81–91.

[149] I. Teutsch, M. Brühl, R. Weisse, and S. Wahls. “Contribution of solitons to enhanced rogue wave occurrence in shallow water: a case study in the southern North Sea”. In: *Natural Hazards and Earth System Sciences Discussions* (2022). Accepted for publication in April 2023., pp. 1–37.

[150] M. Brühl, P. J. Prins, S. Ujvary, I. Barranco, S. Wahls, and P. L.-F. Liu. “Comparative analysis of bore propagation over long distances using conventional linear and KdV-based nonlinear Fourier transform”. In: *Wave Motion* 111 (2022), p. 102905.

[151] A. Hasegawa and T. Nyu. “Eigenvalue communication”. In: *Journal of lightwave technology* 11.3 (1993), pp. 395–399.

[152] X. Yangzhang, V. Aref, S. T. Le, H. Buelow, D. Lavery, and P. Bayvel. “Dual-polarization non-linear frequency-division multiplexed transmission with *b*-modulation”. In: *Journal of Lightwave Technology* 37.6 (2019), pp. 1570–1578.

[153] J. E. Prilepsky, S. A. Derevyanko, K. J. Blow, I. Gabitov, and S. K. Turitsyn. “Nonlinear inverse synthesis and eigenvalue division multiplexing in optical fiber channels”. In: *Physical review letters* 113.1 (2014), p. 013901.

[154] S. Wahls. “Generation of time-limited signals in the nonlinear Fourier domain via b-modulation”. In: *2017 European Conference on Optical Communication (ECOC)*. IEEE. 2017, pp. 1–3.

[155] S. L. Brunton, J. L. Proctor, and J. N. Kutz. “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”. In: *Proceedings of the national academy of sciences* 113.15 (2016), pp. 3932–3937.

[156] S. Lin and Y. Chen. “A two-stage physics-informed neural network method based on conserved quantities and applications in localized wave solutions”. In: *Journal of Computational Physics* 457 (2022), p. 111053.

[157] H. D. Wahlquist and F. B. Estabrook. “Prolongation structures of nonlinear evolution equations”. In: *Journal of Mathematical Physics* 16.1 (1975), pp. 1–7.

[158] N. Asano and Y. Kato. “Spectrum method for a general evolution equation”. In: *Progress of Theoretical Physics* 58.1 (1977), pp. 161–174.

[159] R. Dodd and A. Fordy. “The prolongation structures of quasi-polynomial flows”. In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 385.1789 (1983), pp. 389–429.

[160] M. Ito. “A REDUCE program for evaluating a Lax pair form”. In: *Computer physics communications* 34.3 (1985), pp. 325–331.

[161] M. Hickman, W. Hereman, J. Larue, and Ü. Göktaş. “Scaling invariant Lax pairs of nonlinear evolution equations”. In: *Applicable Analysis* 91.2 (2012), pp. 381–402.

[162] I. T. Habibullin and A. R. Khakimova. “A direct algorithm for constructing recursion operators and Lax pairs for integrable models”. In: *Theoretical and Mathematical Physics* 196.2 (2018), pp. 1200–1216.

[163] M. Syväri. *Symmetries in string theory: Application of machine learning to theoretical physics*. Doctoral thesis, Ludwig-Maximilians-Universität München, Germany. 2021. URL: http://nbn-resolving.de/urn:nbn:de:bvb:19-287407.

[164] The MathWorks Inc. *System Identification Toolbox*. Online. 2024. URL: https://www.mathworks.com/products/sysid.html.

[165] F. Calogero and M. Nucci. “Lax pairs galore”. In: *Journal of mathematical physics* 32.1 (1991), pp. 72–74.

[166] G. P. Agrawal. “Nonlinear fiber optics”. In: Academic Press, 2013.

[167] J. Cuevas-Maraver, P. G. Kevrekidis, and F. Williams. “The sine-Gordon model and its applications”. In: *Nonlinear systems and complexity* 10 (2014).

[168] P. de Koster and S. Wahls. “Fibre model identification for nonlinear Fourier transform-based transmission”. In: *45th European Conference on Optical Communication (ECOC 2019)*. IET. 2019, pp. 1–4.

ACKNOWLEDGEMENTS

I would like to thank several people, who either directly or indirectly helped me on my journey towards completing my doctoral research.

First of all, I would like to thank my promoter and supervisor Sander Wahls. He has patiently has supported me though the entire process of becoming a full-fledged researcher. Thank you helping, advising, encouraging and inspiring me. Furthermore, I would like to thank my second promoter, Michel Verhaegen, for overseeing the entire process of my PhD.

I would also like thank my external collaborators, Jonas Koch, Olaf Schulz and Stephan Pachnicke for all the work they put into getting experimental results and providing feedback.

Next, I would like to thank my former colleagues Shrinivas Chimmalgi and Peter Prins, whom I have always considered as my mentors. Both of you started about one year before me, and helped me get started in the topic, ranging from installing software, getting acquainted with the nonlinear Fourier transform, and discussing new ideas.

I would also like to thank Vinod, Yu-Chen, Markus, Huichen, Shahzeb and Hamed who have supported me with fruitful scientific discussions.

Some special thanks goes towards Jacques and Edoardo, who helped me get a working spot during my hospitality period after my contract ended, and stuck up with me asking for coffee.

I would like to thank both the already mentioned colleagues and many other colleagues at DCSC who have all contributed to making the work floor a fun, social and inspiring environment. Thanks for all the coffee moments, talks, discussion, board games, and table football matches: Atin, Baptiste, Carlos, Coen, Leonoor, Lukasz, Maarten, Pieter, Paul, Roger, Reinier, Sander, Shreesma, and all others that I forgot but should also be in this list.

I would also like to thank many friends outside the PhD environment that have supported me indirectly by helping me relax, sharing my hobbies of board games and partner dancing, and being there for me when I needed them: Daniël, Frank, Remco, Bo, Marieke, Lennart, Roel, Marcella, Wout, Rachida, Joris, Chak Shun, Jaap, Daan, Anita, Trifon, Faye, Lars, Erin, Casper, Cecília, Dante, Jitske, Wessel, Michael, Charlotte, Katja, Sven, Noortje, Anne-Sophie, Laura, Rosanne, Bart, Julia, Daphne, Marieke, Zhengping, Raquel, Karla, Jing, Nick, Lisa, Yoerie, Priya, Bart, Armin, Anne, Colin, Robin, Tineke, Elizabeta, Matthijs, and many others. Thank you all for the time and fun we spent together on sharing our hobbies.

Finally, I would thank my family, who have always loved and supported me. As last person, I would like to thank Esmée for unconditionally loving me and standing with me through all the ups and downs.

Pascal de Koster, March 2024

List of Publications

Journal Articles

4. P. de Koster and S. Wahls. “Data-driven identification of the spectral operator in AKNS Lax pairs using conserved quantities”. In: *Wave Motion* (2024), p. 103273

3. P. de Koster and S. Wahls. “Fast and Reliable Detection of Significant Solitons in Signals with Large Time-Bandwidth Products”. In: *Journal of Lightwave Technology* (2023)

2. P. de Koster, O. Schulz, J. Koch, S. Pachnicke, and S. Wahls. “Fast Single-Mode Fiber Nonlinearity Monitoring: An Experimental Comparison Between Split-Step and Nonlinear Fourier Transform-Based Methods”. In: *IEEE Photonics Journal* (2023)

1. P. de Koster and S. Wahls. “Dispersion and Nonlinearity Identification for Single-Mode Fibers Using the Nonlinear Fourier Transform”. In: *Journal of Lightwave Technology* 38.12 (2020), pp. 3252–3260

Conference Proceedings

4. P. de Koster, M. Brühl, and S. Wahls. “Water-depth identification from free-surface data using the KdV-based nonlinear Fourier transform”. In: *International Conference on Offshore Mechanics and Arctic Engineering*. Vol. 85901. American Society of Mechanical Engineers. 2022, V05BT06A056

3. P. de Koster, J. Koch, O. Schulz, S. Pachnicke, and S. Wahls. “Experimental validation of nonlinear Fourier transform-based Kerr-nonlinearity identification over a 1600 km SSMF link”. In: *Optical Fiber Communication Conference*. Optica Publishing Group. 2022, W2A–39

2. P. de Koster, J. Koch, S. Pachnicke, and S. Wahls. “Experimental Investigation of Nonlinear Fourier Transform Based Fibre Nonlinearity Characterisation”. In: *2021 European Conference on Optical Communication (ECOC)*. IEEE. 2021, pp. 1–4

1. P. de Koster and S. Wahls. “Fibre model identification for nonlinear Fourier transform-based transmission”. In: *45th European Conference on Optical Communication (ECOC 2019)*. IET. 2019, pp. 1–4