

Fiber-Optic Communications Using Nonlinear Fourier Transforms Algorithms and a Bound

Chimmalgi, S.

DOI

[10.4233/uuid:50ed87d8-2ec2-40e4-88b2-02ac63ef8413](https://doi.org/10.4233/uuid:50ed87d8-2ec2-40e4-88b2-02ac63ef8413)

Publication date

2022

Document Version

Final published version

Citation (APA)

Chimmalgi, S. (2022). *Fiber-Optic Communications Using Nonlinear Fourier Transforms: Algorithms and a Bound*. [Dissertation (TU Delft), Delft University of Technology]. <https://doi.org/10.4233/uuid:50ed87d8-2ec2-40e4-88b2-02ac63ef8413>

Important note

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

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.



**FIBER-OPTIC COMMUNICATIONS
USING NONLINEAR FOURIER
TRANSFORMS**

ALGORITHMS AND A BOUND

FIBER-OPTIC COMMUNICATIONS USING NONLINEAR FOURIER TRANSFORMS

Shrinivas Chimmalgi

Shrinivas Chimmalgi

**FIBER-OPTIC COMMUNICATIONS USING
NONLINEAR FOURIER TRANSFORMS**
ALGORITHMS AND A BOUND

**FIBER-OPTIC COMMUNICATIONS USING
NONLINEAR FOURIER TRANSFORMS**
ALGORITHMS AND A BOUND

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
Thursday 8 September 2022 at 10:00 o'clock

by

Shrinivas CHIMMALGI

Master of Science in Systems and Control, Delft University of Technology, the
Netherlands
born in Panvel, India.

This dissertation has been approved by the promotor.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Dr.-Ing. S. Wahls,	Delft University of Technology, promotor
Prof. dr. ir. M. Verhaegen,	Delft University of Technology, promotor

Independent members:

Prof. dr. G. V. Vdovine,	Delft University of Technology
Prof. dr. S. Randoux,	University of Lille, France
Prof. dr. S. K. Turitsyn,	Aston University, United Kingdom
Dr. A. Alvarado,	Eindhoven University of Technology
Prof. dr. ir. B. H. K. De Schutter,	Delft University of Technology, reserve member

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 716669)



European Research Council
Established by the European Commission

Keywords: nonlinear Fourier transforms, nonlinear Schrödinger equation, fiber-optic communications, forward and inverse algorithms, numerical methods, fast algorithms, Riemann theta function, theoretical bounds

Printed by: [Drukkerij Haveka](#)

Front & Back: Numerical approximation of a genus-60 solution of the Korteweg–De Vries equation computed using the Hyperbolic_cross algorithm from Chapter 3 of this dissertation.

Copyright © 2022 by S. Chimmalgi

ISBN 978-94-6384-357-7

An electronic version of this dissertation is available at
<http://repository.tudelft.nl/>.

CONTENTS

Summary	ix
Samenvatting	xiii
Acknowledgements	xvii
1 Introduction	1
1.1 Brief History of Nonlinear Fourier Transforms and the Nonlinear Schrödinger Equation	2
1.2 Nonlinear Fourier Transforms for the Nonlinear Schrödinger Equation . . .	4
1.2.1 Nonlinear Fourier Transform for Vanishing Signals	4
1.2.2 Nonlinear Fourier Transform for Periodic Signals	7
1.3 Nonlinear Schrödinger Equation in Fiber-Optic Communications	10
1.3.1 Soliton Communications	13
1.3.2 Nonlinear Frequency Division Multiplexing System Design	14
1.4 Numerical Aspects of Nonlinear Fourier Transforms	18
1.4.1 Numerical Aspects of Forward NFTs	19
1.4.2 Numerical Aspects of Inverse NFTs	24
1.5 Outline of the Dissertation	25
2 Fast Nonlinear Fourier Transform Algorithms Using Higher Order Exponential Integrators	29
2.1 Introduction	30
2.2 Preliminaries	31
2.3 Numerical Computation of NFT using Higher Order Exponential Integrators	33
2.3.1 Assumptions	33
2.3.2 Numerical Scattering	33
2.3.3 Exponential Integrators	33
2.3.4 Error Metric and Numerical Examples	35
2.4 Fast Fourth-Order NFT	40
2.4.1 Fast Scattering Framework	41
2.4.2 Fast Evaluation	42
2.4.3 Numerical Examples	43
2.5 Main Result: Fast Sixth-order NFT	47
2.5.1 Richardson Extrapolation	48
2.5.2 Numerical Examples	48
2.5.3 Remarks on Computing Eigenvalues	49

2.6	Conclusion	49
2.A	Comparison of FLOP Counts and Execution Times	51
2.B	Interpolation Based on Fourier Transform	51
2.C	Computing Eigenvalues.	53
2.C.1	Existing Approaches	53
2.C.2	Proposed Method	54
2.C.3	Numerical Example	57
3	On Computing High-dimensional Riemann Theta Functions	61
3.1	Introduction	62
3.2	Preliminaries	63
3.2.1	Truncated Riemann Theta Functions	63
3.2.2	Siegel Transform	64
3.2.3	Tensor-Train Decomposition.	65
3.3	Analysis of the Truncation Error.	66
3.3.1	Lower Bounds on the Truncation Error	66
3.3.2	An Upper Bound for $\mathcal{J}^g(\infty, N)$	68
3.3.3	Discussion	70
3.4	Summing over Hypercubes Using Tensor Trains and Scaling and Squaring	70
3.4.1	Complexity Analysis for Fixed K and s	71
3.4.2	Complexity Analysis for a Given Error Bound	71
3.5	Numerical Experiments.	72
3.5.1	Choice of Benchmark Algorithms	73
3.5.2	Generation of Test Data	73
3.5.3	Accuracy for Genus-2 and Genus-6 KdV Solutions	74
3.5.4	Accuracy for Genus-3 NLS Solution	76
3.5.5	Computing High Genus KdV Solutions.	81
3.6	Conclusion	85
3.A	Proof of Theorem 1	85
3.B	Proof of Lemma 2 and Proposition 5	87
4	Theoretical Analysis of Maximum Transmit Power in a b-Modulator	91
4.1	Introduction	92
4.2	Review of b -Modulation	92
4.3	Theoretical Analysis.	93
4.4	Simulations	95
4.5	Conclusion	96
4.A	Improved INFT Algorithm	96
5	Bounds on the Transmit Power of b-Modulated NFDN Systems in Anomalous Dispersion Fiber	99
5.1	Introduction	100
5.2	Review of NFDN	101
5.2.1	Nonlinear Fourier Transform for Vanishing Signals	101
5.2.2	NFDN Signal Generation	102

5.3	Upper Bounds on the Transmit Power of b -Modulators	104
5.3.1	Power Bound for a Fixed Gap to Singularity.	105
5.3.2	Uniform Power Bound for Arbitrary Gaps to Singularity	107
5.4	Conclusions.	114
6	Software Contributions: FNFT and NFDMLab	115
6.1	Introduction	116
6.2	FNFT	116
6.3	NFDMLab.	117
6.4	Conclusion	120
7	Conclusions and Recommendations	123
7.1	Conclusions.	124
7.2	Recommendations for Future Work.	126
	List of Publications	151
	Curriculum Vitae	153

SUMMARY

Due to the ever increasing global connectivity, the demand on the fiber-optic communication infrastructure is projected to keep increasing rapidly. A major factor currently limiting transmission capacity is the fiber nonlinearity. Some researchers have suggested the application of nonlinear Fourier transforms to exploit the fiber nonlinearity rather than ignoring or mitigating it. Nonlinear Fourier transforms allow us to solve certain nonlinear partial differential equations by transforming the complex evolution of the solution in the time-domain to a simple multiplication with a nonlinear frequency response in the nonlinear Fourier domain. This method is analogous to solving linear partial differential equations using the Fourier transform. The nonlinear Schrödinger equation is a suitable model for the propagation of light through a single-mode optical fiber. Its lossless version is solvable through a nonlinear Fourier transform. In recent years, several nonlinear Fourier transform based communication systems have been proposed. Such systems require numerical algorithms to compute the nonlinear Fourier transforms as nonlinear Fourier spectra are known analytically for only a handful of signals, and linear superposition cannot be used to compute the spectrum of a more complex signal. Computationally efficient algorithms are therefore not only essential for the real-time operation of nonlinear Fourier transform based communication systems, but are also important for their simulation. One common way to improve the spectral efficiency of a communication system is to increase the signal power in order to reduce the impact of noise. Another is to increase the signal duration in order to reduce the impact of information-free guard intervals that are inserted between transmissions to deal with the channel memory. Longer signals however require more resources to process them. The numerical problem of computing nonlinear Fourier transforms furthermore gets harder for both higher power and longer durations. Hence in the literature, we observe that the inability to perform efficient communication in these regimes is typically attributed to numerical problems of existing algorithms. In this dissertation we develop new algorithms that require shorter computation times for achieving similar accuracies as existing algorithms. Furthermore, we theoretically investigate whether some of the problems that are commonly attributed to numerical difficulties could occur in the absence of numerical effects.

The nonlinear Fourier transform for signals that decay sufficiently fast is currently the most commonly used transform in nonlinear Fourier transform based communication systems. We developed new algorithms for computing the continuous nonlinear Fourier spectrum which is one part of the nonlinear Fourier spectrum for decaying signals. We demonstrated significant improvements over existing algorithms in multiple numerical benchmarks, and implemented the algorithms in the open source software library FNFT. We also developed NFDMLab, which is a Python based open source simulation environment for nonlinear Fourier transform based communication systems that relies on FNFT. The developed forward nonlinear Fourier transform algorithms are

fast higher-order methods with a complexity of $\mathcal{O}(D \log^2 D)$ for computing the continuous nonlinear Fourier spectrum from D samples of a decaying signal. In the numerical benchmarks, we introduced the trade-off between accuracy and computation time as a new way to compare nonlinear Fourier transform algorithms and found that the newly proposed algorithms perform significantly better than prior work in this regard. We also provided the first counting analysis of a fast nonlinear Fourier transform algorithm.

There is also interest in using the nonlinear Fourier transform for periodic signals, as it is closer to the method used in conventional orthogonal frequency division multiplexing communication systems. The definition of the nonlinear Fourier transform for periodic signals is different from that of decaying signals. Communication systems based on nonlinear Fourier transforms for periodic signals make use of so-called finite-genus solutions of the nonlinear Schrödinger equation. Riemann theta functions are the traditional way to realize inverse nonlinear Fourier transforms that are used to synthesize finite-genus solutions. They are multi-dimensional Fourier series and their numerical computation suffers from the curse of dimensionality. This limits the genus of the signals used in the communication systems and is seen as a major bottleneck. We derived new bounds on the series truncation error and proposed two tensor-train based and a hyperbolic cross index set based algorithms for computing high-dimensional Riemann theta functions. We compared them to existing algorithms in multiple numerical benchmarks. The bounds that we derived on the truncation error of the Riemann theta functions allowed us to rule out several of the existing approaches for the high-dimension regime. We demonstrated that the algorithm based on the hyperbolic cross can compute Riemann theta functions upto 60 dimensions with moderate accuracy which is significantly higher than what was previously feasible.

We also tried to improve the performance of nonlinear Fourier transform based communication systems known as b -modulators in the highly nonlinear regime using improved numerical algorithms. When we did not see improvements, we conducted a theoretical analysis of b -modulation systems. The analysis allowed us to prove theoretically that nonlinear bandwidth, signal duration and power are coupled when singularities in the nonlinear spectrum are avoided. When the nonlinear bandwidth is fixed, the coupling results in an upper bound on the transmit power. The power bound decreases with increasing signal duration which consequently decreases the signal-to-noise ratios for long signals, which explains the observed performance degradation in this regime without resorting to numerical difficulties. This result is the first of its kind as such a behaviour is not known from conventional linear systems. We also demonstrated numerically that the transmit powers achieved by an exemplary b -modulated system are close to its theoretical limits.

Fiber-optic communication systems based on nonlinear Fourier transforms have been proposed to potentially tackle fiber nonlinearity, which is a major factor currently limiting transmission capacity. Efficient numerical algorithms are essential for real-time operation as well as efficient simulations of nonlinear Fourier transform based fiber-optic communication systems. The algorithms presented in this dissertation potentially make already published nonlinear Fourier transform based communication systems more practical and also allow for development of new designs which were previously infeasible. In this dissertation furthermore a limitation on communication system de-

sign imposed by the structure of the nonlinear Fourier transform was identified. It can be used to explain the inability to perform efficient communication with long duration signals, which was previously attributed to numerical problems, and guide the design of future systems.

SAMENVATTING

Vanwege de steeds toenemende wereldwijde verbondenheid voor communicatie zal de vraag naar de infrastructuur voor glasvezelcommunicatie naar verwachting snel blijven toenemen. Een belangrijke factor die momenteel de transmissiecapaciteit beperkt, is de niet-lineariteit van de glasvezelkabels. Sommige onderzoekers hebben de toepassing van niet-lineaire Fouriertransformaties voorgesteld om de niet-lineariteit van de vezels te benutten in plaats van deze te negeren of te verminderen. Niet-lineaire Fouriertransformaties stellen ons in staat om bepaalde niet-lineaire partiële differentiaalvergelijkingen op te lossen door de complexe evolutie van de oplossing in het tijd domein te transformeren naar een eenvoudige vermenigvuldiging met een niet-lineaire frequentierespons in het niet-lineaire Fourierdomein. Deze methode is analoog aan het oplossen van lineaire partiële differentiaalvergelijkingen met behulp van de Fouriertransformatie. De niet-lineaire Schrödinger vergelijking is een geschikt model voor de voortplanting van licht door een single-mode optische vezel. De verliesloze versie ervan is oplosbaar via een niet-lineaire Fouriertransformatie. In de afgelopen jaren zijn verschillende niet-lineaire op Fouriertransformatie gebaseerde communicatiesystemen voorgesteld. Dergelijke systemen vereisen numerieke algoritmen om de niet-lineaire Fouriertransformaties te berekenen, aangezien niet-lineaire Fourierspectra voor slechts een handvol signalen analytisch bekend zijn, en lineaire superpositie niet kan worden gebruikt om het spectrum van een complexer signaal te berekenen. Computationeel efficiënte algoritmen zijn daarom niet alleen essentieel voor de real-time werking van niet-lineaire Fouriertransformatie-gebaseerde communicatiesystemen, maar zijn ook belangrijk voor hun simulatie. Een veelgebruikte manier om de spectrale efficiëntie van een communicatiesysteem te verbeteren, is door het signaalvermogen te vergroten, zodat de invloed van informateloze bewakingsintervallen (die tussen pulsen worden ingevoegd omwille van het kanaalgeheugen) wordt vermindert. Langere signalen vereisen echter meer middelen om ze te verwerken. Het numerieke probleem van het berekenen van niet-lineaire Fouriertransformaties wordt bovendien moeilijker voor zowel hogere vermogens als langere duur. Daarom zien we in de literatuur dat het onvermogen om in deze regimes efficiënt te communiceren doorgaans wordt toegeschreven aan numerieke problemen van bestaande algoritmen. In dit proefschrift ontwikkelen we nieuwe algoritmen die kortere rekentijden nodig hebben om vergelijkbare nauwkeurigheid te bereiken als bestaande algoritmen. Verder onderzoeken we theoretisch of sommige van de problemen die typisch worden toegeschreven aan numerieke problemen, zouden kunnen optreden in de afwezigheid van numerieke effecten.

De niet-lineaire Fouriertransformatie voor signalen die voldoende snel vervallen is momenteel de meest gebruikte transformatie in niet-lineaire op-Fouriertransformatie-gebaseerde communicatiesystemen. We hebben nieuwe algoritmen ontwikkeld voor het berekenen van het continue niet-lineaire Fourierspectrum, dat deel uitmaakt van het niet-lineaire Fourierspectrum voor vervallende signalen. We hebben significante ver-

beteringen aangetoond ten opzichte van bestaande algoritmen in meerdere numerieke benchmarks en hebben de algoritmen geïmplementeerd in de open source softwarebibliotheek FNFT. We hebben ook NFDMLab ontwikkeld, een op-Python-gebaseerde open-source simulatieomgeving voor niet-lineaire Fouriertransformatie-gebaseerde communicatiesystemen die afhankelijk zijn van FNFT. De ontwikkelde voorwaartse niet-lineaire Fouriertransformatiealgoritmen zijn snelle methoden van hogere orde met een complexiteit van $\mathcal{O}(D \log^2 D)$ voor het berekenen van het continue niet-lineaire Fourier spectrum uit D -meetpunten van een vervalend signaal. In de numerieke benchmarks hebben we de afweging tussen nauwkeurigheid en rekentijd als een nieuwe manier om niet-lineaire Fouriertransformatiealgoritmen te vergelijken geïntroduceerd en hebben ontdekt dat de nieuw voorgestelde algoritmen in dit opzicht aanzienlijk beter presteren dan voorgaande methoden. We hebben ook de eerste telanalyse van een snel niet-lineair-Fouriertransformatiealgoritme geleverd.

Er is ook interesse in het gebruik van de niet-lineaire Fouriertransformatie voor periodieke signalen, omdat deze dichter bij de methode ligt die wordt gebruikt in conventionele orthogonale-frequentieverdeling-multiplexingcommunicatiesystemen. De definitie van de niet-lineaire Fouriertransformatie voor periodieke signalen verschilt van die van vervallende signalen. Communicatiesystemen op basis van niet-lineaire Fouriertransformaties voor periodieke signalen maken gebruik van zogenaamde eindige-genusoplossingen van de niet-lineaire Schrödinger-vergelijking. Riemann-thèta-functies zijn de traditionele manier om inverse niet-lineaire Fouriertransformaties te realiseren die worden gebruikt om eindige-genusoplossingen te synthetiseren. De Riemann-thètafuncties zijn multidimensionale Fourierreeksen en de numerieke berekening ervan lijdt onder de vloek van de dimensionaliteit. Dit beperkt de genus van de signalen die in de communicatiesystemen worden gebruikt en wordt gezien als een belangrijk knelpunt. We hebben nieuwe grenzen afgeleid voor de afbreekfout van de reeks en hebben twee op-tensorketen-gebaseerde algoritmes en een op-hyperbolisch-kruis-indexset-gebaseerde algoritmen voor voor het berekenen van hoog-dimensionale Riemann-thètafuncties. We hebben ze met bestaande algoritmen in meerdere numerieke benchmarks vergeleken. De grenzen die we hebben afgeleid op de afbreekfout van de Riemann-thètafuncties stelden ons in staat om verschillende van de bestaande benaderingen voor het regime met hoge dimensies uit te sluiten. Het op-hyperbolische-kruis-gebaseerde algoritme kan Riemann-thètafuncties tot 60 dimensies berekenen met een redelijke nauwkeurigheid, wat aanzienlijk hoger is dan voorheen haalbaar was.

We hebben ook geprobeerd om de prestaties van niet-lineaire-Fouriertransformatie-gebaseerde communicatiesystemen, bekend als b -modulatoren in het sterk niet-lineaire regime, te verbeteren met behulp van verbeterde numerieke algoritmen. Toen we geen verbeteringen zagen, voerden we een theoretische analyse uit van b -modulatiesystemen. De analyse stelde ons in staat om theoretisch te bewijzen dat niet-lineaire bandbreedte, signaalduur en vermogen gekoppeld zijn indien singulariteiten in het niet-lineaire spectrum worden vermeden. Wanneer de niet-lineaire bandbreedte vast ligt resulteert de koppeling in een bovengrens van het zendvermogen. De vermogensgrens neemt af met toenemende signaalduur, waardoor de signaal-ruisverhoudingen voor lange signalen afnemen, wat de waargenomen prestatievermindering in dit regime verklaart zonder toevlucht te nemen tot numerieke problemen. Dit resultaat is het eerste in zijn soort,

aangezien een dergelijk gedrag niet bekend is voor conventionele lineaire systemen. We hebben ook numeriek aangetoond dat de zendvermogens die worden bereikt door een b -gemoduleerd voorbeeldsysteem dicht bij zijn theoretische limieten liggen.

Vezeloptische-communicatiesystemen op basis van niet-lineaire Fouriertransformaties zijn voorgesteld om mogelijk niet-lineariteit van vezels aan te pakken, wat momenteel een belangrijke beperkende factor voor de transmissiecapaciteit. Efficiënte numerieke algoritmen zijn essentieel voor real-time werking, evenals efficiënte simulaties van niet-lineaire-Fouriertransformatie-gebaseerde glasvezelcommunicatiesystemen. De algoritmen die in dit proefschrift worden gepresenteerd, maken mogelijk reeds gepubliceerde op-niet-lineaire-Fouriertransformatie-gebaseerde communicatiesystemen praktischer en maken ook de ontwikkeling mogelijk van nieuwe ontwerpen die voorheen onhaalbaar waren. In deze dissertatie is bovendien een beperking geïdentificeerd op het ontwerp van communicatiesystemen, wat veroorzaakt wordt door de structuur van de niet-lineaire Fouriertransformatie. Dit kan worden gebruikt om het onvermogen om efficiënte communicatie uit te voeren met signalen van lange duur uit te leggen, wat voorheen werd toegeschreven aan numerieke problemen. Ten slotte kan deze beperking ook worden meegenomen in het ontwerp van toekomstige systemen.

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank a number of people that have helped and supported me during the course of my journey towards this PhD.

Firstly I would like to thank my promotors, Sander Wahls and Michel Verhaegen, for giving me the opportunity to do doctoral research, for constructively challenging my ideas and giving me crucial feedback. I am indebted to Sander for believing in me and providing invaluable advice, many a times at short notice. Thank you for nudging me to improve myself and to become comfortable with trying tasks. It has been a pleasure working with both of you and it was a wonderful experience to develop into an independent researcher under your supervision.

My thanks also go out to the colleagues at DCSC for the fun and stimulating environment. Special thanks to Peter, Pascal, Vinod, Markus, Sina, Huichen and Yu-Chen for sharing your insights and critical yet essential feedback. Abhimanyu, Baptiste, Coen, Folkert, Jacques, Laurens, Leonoor, Lukasz, Maarten, Pieter, Raf, Reinier, Tope: thank you all for the wonderful time at DCSC. I will always cherish all the fun talk during lunch, coffee breaks, foosball and board games.

Furthermore, I would like to thank external collaborators: Marius Brehler, Christoph Mahnke, Vahid Aref, Gai Zhou, Alan Pak Tao Lau, for their work and feedback.

Marieke, Heleen, Francy and Erica, thank you all for your incredible support for all practical matters.

A big thank you to my family and friends for their immeasurable love and support.

Thank you Girish, Akshaya, Karthik, Mitesh, Surabhi, Manogna and others in 2D for all the banter and encouragement. I am grateful to have you as my friends for almost ten years now!

Abhilash, thank you for being a great housemate and a friend. From the strenuous leg-days at the gym to the discussions about the plagues of being a PhD student, I couldn't have asked for a better person to share a house with for four years.

Vadiraj, Sarvesh, Udhav and Sharadhi, thank you for the fun-filled weekend get-togethers and moral support. It was always a pleasure to hang out with everyone and de-stress from work.

Vaishnavi, I am eternally grateful for all the love, motivation and support. Thank you for helping me grow into the person I am today.

I cannot thank my parents and sister enough for their unconditional love and kind words. Thank you for believing in me and having my back during all the ups and downs. I couldn't have accomplished all this without you.

Shrinivas Chimmalgi
January 2022

1

INTRODUCTION

CHAPTER ABSTRACT

This chapter gives an introduction to the concept of nonlinear Fourier transforms (NFTs) for the nonlinear Schrödinger equation and their application in the field of fiber-optic communications. We start the introduction with a brief history of NFTs and the nonlinear Schrödinger equation. In the second section, we furnish details of particular NFTs for the nonlinear Schrödinger equation. In the third section, we discuss the role of the nonlinear Schrödinger equation in the development of fiber-optic communication systems. Recent interest and progress of NFT based approaches for fiber-optic communications are also discussed. In the fourth section we review numerical methods for computing NFTs for the nonlinear Schrödinger equation. The final section presents an outline of this dissertation.

1.1. BRIEF HISTORY OF NONLINEAR FOURIER TRANSFORMS AND THE NONLINEAR SCHRÖDINGER EQUATION

Integral transforms are useful tools for solving initial value problems involving linear partial differential equations (PDEs). For a given PDE defined on a domain, applying a suitable integral transform allows it to be expressed in a form that can be operated on more easily than the original one. If a solution is found in the transformed domain, the inverse integral transform can be applied to obtain the solution of the original PDE. Hence, the PDE can be solved by the means of an integral transform using the scheme of *transform-solve-invert*. One of the most known and widely applied transforms is the Fourier transform (FT) [1]. The Fourier transform can be used to decompose a function of, for example, time into a function of temporal frequency. The frequency domain representation can give insights into the solutions of linear PDEs and reduces the evolution of the solutions to simple multiplication with a frequency response [2]. However, the evolution of the frequency domain representation of solutions to nonlinear PDEs typically remains complicated. The nonlinearity furthermore enables new wave phenomena which are not possible in linear systems. The solitary wave solution of the Korteweg-de Vries (KdV) equation is an representative example of such a solution. The solitary wave was first described in 1834 by John Scott Russell who observed the wave in the Union Canal in Scotland. A solitary wave is a wave packet that maintains its shape while it propagates at a constant velocity even after collision with another solitary wave. Figure 1.1 shows a photo of the solitary wave that was recreated on the Union canal in 1995 [3]. In



Figure 1.1: Soliton on the Scott Russell Aqueduct on the Union Canal near Heriot-Watt University, 12 July 1995. Photo from [3]. Reprinted by permission from Springer Nature: Nature Soliton wave receives crowd of admirers. Nature 376, 373 (1995)

1965 Zabusky and Kruskal [4], [5] coined the word “soliton” for these special waves. We now know that solitons are caused by a cancellation of nonlinear and dispersive effects in the medium [6]. They arise as solutions of a widespread class of weakly nonlinear dis-

persive PDEs which describe physical systems [7, p. 481]. While studying one such PDE, the KdV equation, in 1967 Gardner et al. found an ingenious new method for solving the associated Cauchy problem [8]. In the following year, Peter Lax developed the framework of Lax pairs [9], which not only covered the results of Gardner et al., but also were applicable to many other nonlinear PDEs. Soon Lax pairs were identified for other equations such as the nonlinear Schrödinger equation (NLSE) [10], the Sine-Gordon equation [11], and the Toda lattice equation [12]–[14]. The method developed by Gardner et al. is traditionally known as the *inverse scattering transform* (IST) method. The IST method follows the transform-solve-invert scheme. In the first step of the IST method, a spectral representation of the initial condition is computed. The spectral representation is obtained as the spectrum of a linear operator associated with the PDE and the initial condition. This representation is specific to the nonlinear evolution equation and boundary conditions of the solution. The spectral representation evolves in a simple way that can be solved in closed form. In the third step, the spectral representation is inverted to obtain the evolved solution. Evolution equations to which the IST method is applicable belong to a class known as integrable equations. Ablowitz et al. developed a framework to generate integrable equations associated with Lax pairs of a certain structure [11]. A general approach for finding a Lax pair for a given PDE (should it exist) however remains an open problem. The IST methods can be viewed as nonlinear analogues of the Fourier transform and are hence often called nonlinear Fourier transforms (NFTs). NFTs should not only be seen as tools for solving PDEs, but also as tools to analyze data since they transform signals into special representations that reveal nonlinear features such as potentially hidden solitonic components. Conventional linear analysis methods are not able to detect such features.

The nonlinear Schrödinger equation is one of the most widely studied integrable nonlinear PDEs [15]. It finds application in modelling the propagation of weakly nonlinear narrow-band wave trains in diverse fields such as fluid dynamics [16], nonlinear optics [17], magnetic spin waves [18], [19] and plasma physics [20]. A version of the nonlinear Schrödinger equation which models the complex valued envelope of a pulse $A(\tau, z)$ transmitted through an ideal silica fiber without loss is given by [21, Section 2.6]

$$\frac{\partial A}{\partial z} + \frac{i\beta_2}{2} \frac{\partial^2 A}{\partial \tau^2} = i\gamma|A|^2 A. \quad (1.1)$$

Here z denotes distance (in km) along the fiber and the symbol τ represents retarded time (in seconds). The retarded time $\tau = \tilde{t} - \beta_1 z$, where \tilde{t} is ordinary time and β_1 is a constant. The symbol i denotes the imaginary unit. *Chromatic dispersion* is the phenomenon by which different spectral components of a pulse travel at different velocities. The coefficient β_2 (in s^2/km) is called the chromatic dispersion coefficient. The nonlinear term in the NLS equation models *self-phase modulation* (SPM) and the coefficient γ (in $\text{W}^{-1}\text{km}^{-1}$) is the corresponding nonlinearity parameter. SPM occurs due to the *Kerr effect* which is a change in the refractive index of a material in response to an applied electric field. Different forms of the NLS equation can be derived specific to the application. It is often convenient to develop the theory and numerical methods for a normalized NLS equation, as they can be adapted to the particular physical problem using the appropriate change of variables. A commonly used normalized form of the

nonlinear Schrödinger equation is [21, Chap. 2]

$$i \frac{\partial q}{\partial x} + \frac{\partial^2 q}{\partial t^2} + 2\kappa |q|^2 q = 0, \quad (1.2)$$

where $q(t, x)$ denotes a complex valued envelope function, $x \geq 0$ denotes the normalized spatial coordinate and t denotes normalized temporal coordinate. Note that the roles of x and t can be switched depending on the application. We obtain (1.2) from (1.1) by changing the variables as

$$q = \frac{A}{\sqrt{P_0}}, \quad x = -\text{sign}(\beta_2) \frac{z}{L_D}, \quad t = \frac{\tau}{T_0} \quad (1.3)$$

with $L_D = 2T_0^2/|\beta_2|$ and $P_0 = 2/(\gamma L_D)$. The constants L_D and P_0 are usually obtained by choosing a normalization time T_0 , which is a free parameter. The parameter $\kappa = -1$ if β_2 is positive (normal dispersion). In that case (1.2) is known as the defocusing NLS equation. For β_2 negative (anomalous dispersion), $\kappa = 1$ and (1.2) is known as the focusing NLS equation. The equation (1.1) is integrable, while in practical applications, other terms such as loss and higher-order dispersion terms can be present, which break integrability. The integrable approximation (1.2) nevertheless can provide meaningful insights into the nonlinear dynamics.

In an optical fiber, two pulses can be transmitted simultaneously on the same carrier frequency by using waves of two orthogonal polarization states. The NLS equation models the evolution of only one of the polarization states. The evolution of orthogonally polarized pulse envelopes can be approximated by the normalized Manakov equation [22]

$$\begin{aligned} i \frac{\partial q_1}{\partial x} + \frac{\partial^2 q_1}{\partial t^2} + 2\kappa(|q_1|^2 + |q_2|^2)q_1 &= 0, \\ i \frac{\partial q_2}{\partial x} + \frac{\partial^2 q_2}{\partial t^2} + 2\kappa(|q_1|^2 + |q_2|^2)q_2 &= 0 \end{aligned} \quad (1.4)$$

which is a vector valued integrable equation. The normalized signals $q_1(t, x)$ and $q_2(t, x)$ can be converted to physical units using a change of variables similar to that given for the NLS equation in (1.3).

1.2. NONLINEAR FOURIER TRANSFORMS FOR THE NONLINEAR SCHRÖDINGER EQUATION

Nonlinear Fourier transforms for the NLS equation have been studied extensively for solutions with different boundary conditions [10], [23]–[29]. In recent years these NFTs have been used in engineering applications for signal processing and data analysis [30]–[37]. The most widely used ones are the NFTs for solutions with vanishing boundaries and periodic boundaries. In the following Sections 1.2.1 and 1.2.2, we will describe the details of the NFTs for these two boundary conditions.

1.2.1. NONLINEAR FOURIER TRANSFORM FOR VANISHING SIGNALS

The NFT that solves the NLS equation (1.2) was first developed for the case of vanishing signals by Zakharov and Shabat [10]. It transforms any signal $q(t)$ that vanishes suffi-

ciently fast for $t \rightarrow \pm\infty$ from the time-domain to the nonlinear Fourier domain through the analysis of the linear ordinary differential equation (ODE)

$$\frac{\partial \mathbf{V}(t, \lambda)}{\partial t} = \mathbf{P}(t, \lambda) \mathbf{V}(t, \lambda) = \begin{bmatrix} -i\lambda & q(t) \\ -\kappa q^*(t) & i\lambda \end{bmatrix} \mathbf{V}(t, \lambda), \quad (1.5)$$

where $q(t) = q(x_0, t)$ for any fixed x_0 . The system (1.5) can be reorganized into an eigen-equation $\mathbf{L}\mathbf{V} = \lambda\mathbf{V}$ [11], which is one half of the Lax pair for the NLS equation. Since $q(t)$ decays fast for $|t| \rightarrow \infty$, we can single out two sets of solutions, $\mathbf{V}(t, \lambda) = [\boldsymbol{\phi}(t, \lambda) \ \bar{\boldsymbol{\phi}}(t, \lambda)]$ and $\mathbf{V}(t, \lambda) = [\boldsymbol{\psi}(t, \lambda) \ \bar{\boldsymbol{\psi}}(t, \lambda)]$, of the system (1.5) that are distinguished by their behaviour at the boundaries as

$$\begin{aligned} \mathbf{V}(t, \lambda) = [\boldsymbol{\phi}(t, \lambda) \ \bar{\boldsymbol{\phi}}(t, \lambda)] &\rightarrow \begin{bmatrix} e^{-i\lambda t} & 0 \\ 0 & -e^{i\lambda t} \end{bmatrix} \text{ as } t \rightarrow -\infty, \\ \mathbf{V}(t, \lambda) = [\bar{\boldsymbol{\psi}}(t, \lambda) \ \boldsymbol{\psi}(t, \lambda)] &\rightarrow \begin{bmatrix} e^{-i\lambda t} & 0 \\ 0 & e^{i\lambda t} \end{bmatrix} \text{ as } t \rightarrow \infty. \end{aligned} \quad (1.6)$$

The term $\lambda \in \mathbb{C}$ is a spectral parameter similar to the parameter s in the Laplace domain. The matrix $\mathbf{V}(t, \lambda)$ is said to contain the eigenfunctions. The spatial evolution of these eigenfunctions is given by

$$\frac{\partial \mathbf{V}(x, \lambda)}{\partial x} = \mathbf{N}(x, \lambda) \mathbf{V}(x, \lambda) = \begin{bmatrix} -2i\lambda^2 + i\kappa|q(t, x)|^2 & 2\lambda q(t, x) + i\frac{\partial q(t, x)}{\partial t} \\ -2\kappa\lambda q^*(t, x) + i\kappa\frac{\partial q^*(t, x)}{\partial t} & 2i\lambda^2 - i\kappa|q(t, x)|^2 \end{bmatrix} \mathbf{V}(x, \lambda), \quad (1.7)$$

which follows from the second operator in the Lax pair. Note that for $t \rightarrow \pm\infty$, the spatial evolution of the eigenfunctions becomes independent of $q(t, x)$ since the signal is assumed to vanish quickly. The so-called *zero-curvature* condition [38]

$$\frac{\partial \mathbf{P}}{\partial x} - \frac{\partial \mathbf{N}}{\partial t} + \mathbf{P}\mathbf{N} - \mathbf{N}\mathbf{P} = 0$$

then is equivalent to the normalized NLS equation (1.2). Each set of solutions of the system (1.5) for the boundary conditions (1.6) has two linearly independent components and thus can span the space of solutions, except for a discrete set of values of λ . Later we shall see that these values are eigenvalues of the \mathbf{L} operator. For values of λ not in this set, every other solution can be expressed as a linear combination of the components of either the sets in (1.6). These sets of solutions are related via a unique matrix

$$\mathcal{S}(\lambda) = \begin{bmatrix} a(\lambda) & \bar{b}(\lambda) \\ b(\lambda) & -\bar{a}(\lambda) \end{bmatrix}, \quad (1.8)$$

called the scattering matrix, such that [11]

$$[\boldsymbol{\phi}(t, \lambda) \ \bar{\boldsymbol{\phi}}(t, \lambda)] = [\bar{\boldsymbol{\psi}}(t, \lambda) \ \boldsymbol{\psi}(t, \lambda)] \mathcal{S}(\lambda). \quad (1.9)$$

The components $a(\lambda)$, $b(\lambda)$, $\bar{b}(\lambda)$ and $\bar{a}(\lambda)$ are known as the scattering data. The components $a(\lambda)$ and $b(\lambda)$ are sufficient to describe the signal completely due to symmetry in the Zakharov-Shabat problem [10]. The *reflection coefficient* is then defined as

$\rho(\lambda) = b(\lambda)/a(\lambda)$ for $\lambda \in \mathbb{R}$. The function $\rho(\lambda)$ is said to constitute the continuous spectrum as it exists for all $\lambda \in \mathbb{R}$. It is analogous to the frequency spectrum of the linear Fourier transform. Although the evolution of the solution $q(x, t)$ with x is complicated, the evolution of the scattering data is simple due to the fact that the evolution of the eigenfunctions is independent of $q(t, x)$ for $t \rightarrow \pm\infty$. The evolution of the reflection coefficient is consequently simple as well [11],

$$\rho(\lambda, x_1) = \rho(\lambda, x_0)e^{4i\lambda^2(x_1-x_0)}, \quad b(\lambda, x_1) = b(\lambda, x_0)e^{4i\lambda^2(x_1-x_0)}, \quad a(\lambda, x_1) = a(\lambda, x_0). \quad (1.10)$$

Since this is similar to multiplication with a frequency response for linear systems, we can say that the NFT as described above linearizes the evolution of the solution for the NLS equation.

Remark 1. *In the literature, sometimes other normalizations of the nonlinear Schrödinger equation are used. The nonlinear Fourier transforms however remain the same. Only the evolution of the scattering data in (1.10) changes with a scalar factor.*

In the case of the focusing NLS equation ($\kappa = 1$), the nonlinear Fourier spectrum can also contain a so-called *discrete spectrum*. It corresponds to the complex poles of the reflection coefficient in the upper half-plane $\mathbb{H} = \{z | z \in \mathbb{C}, \Im\{z\} > 0\}$, or equivalently to the zeros $\lambda_k \in \mathbb{H}$ of $a(\lambda)$. The zeros λ_k are also known as eigenvalues as they correspond to the finite energy solutions of the system (1.5). The finite energy solutions are thus the eigenfunctions of the associated \mathbf{L} operator. It is often assumed that there are finitely many such zeros and that they all are simple zeros. However, it is possible to have zeros of higher order. For simple zeros a corresponding set of values $\rho_k := b(\lambda_k) / \left. \frac{da(\lambda)}{d\lambda} \right|_{\lambda=\lambda_k}$ are known as residues [11, Appendix 5]. The zeros λ_k correspond to the solitonic components of the signal, with $\Re\{\lambda_k\}$ and $\Im\{\lambda_k\}$ indicating the soliton amplitude and speed respectively. In Figure 1.2 we can see a single soliton solution of the NLS equation. The soliton maintains its shape while moving at a constant speed. Unless some solitons have the same speed, the non-solitonic components will spread out while the solitonic components will separate for $x \rightarrow \infty$. Solutions that have only a discrete spectrum are known as *multi-solitons*. The eigenvalues of a solution $q(t, x)$ are invariant with respect to x and the residues evolve with x as

$$\rho_k(x_1) = \rho_k(x_0)e^{4i\lambda_k^2(x_1-x_0)}.$$

In Figure 1.3 we can see an exemplary nonlinear Fourier spectrum of a solution of the focusing NLS equation. The figure is based on Figure 1 from [35]. The reflection coefficient $\rho(\lambda)$ decays to zero on both sides for all vanishing signals for both the focusing and defocusing NLS equations.

Computing the nonlinear Fourier spectrum for a given signal is known as the forward NFT and corresponds to the *transform* step of the *transform-solve-invert* scheme. The evolution of the scattering data as per (1.10) is the *solve* step and the *invert* step involves computing the signal from a given nonlinear Fourier spectrum. This inverse step is thus known as the inverse NFT. Nonlinear Fourier transforms have also been developed for the Manakov equation (1.4) [22].

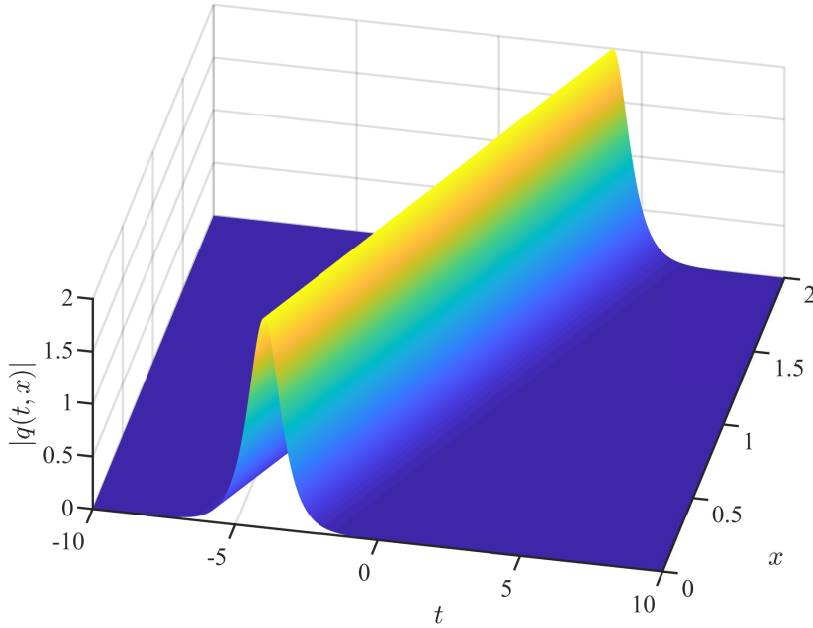


Figure 1.2: A single soliton solution of the nonlinear Schrödinger equation. The soliton maintains its shape while moving at a constant speed.

1.2.2. NONLINEAR FOURIER TRANSFORM FOR PERIODIC SIGNALS

The nonlinear Fourier transform for vanishing signals is suitable to analyze wave packets such as the solitary wave in water or a burst of data in fiber, but in other situations the periodic boundary condition is more appropriate. The study of the NFT for the case of periodic signals started in the mid-seventies [39]–[43] and remains an active area of research [44], [45]. In the following we will describe the periodic NFT for the focusing NLS equation as defined by Kotlyarov and Its [23], [24]. For a periodic signal with period T ,

$$q(t, x) = q(t + T, x).$$

Setting $\mathbf{V}(t, \lambda) = \mathbf{I}_2$ and integrating (1.5) over one period T , we get $\mathbf{V}(t + T, \lambda)$. A part of the nonlinear spectrum known as the *main spectrum* is given by values λ_k that indicate the eigenvalues of the \mathbf{L} operator with (anti-)periodic eigenfunctions. Alternatively, the main spectrum is given by values λ_k for which the monodromy-matrix \mathbf{M} ,

$$\mathbf{M}(t, x, \lambda) := \mathbf{V}(t + T, x, \lambda) \mathbf{V}^{-1}(t, x, \lambda), \quad (1.11)$$

has eigenvalues ± 1 . Any sufficiently smooth periodic solution of the NLS equation can be approximated arbitrarily well on any fixed finite time interval by a periodic *finite-genus* solution [46]. A finite-genus solution has a finite number of main spectrum points. In the focusing case ($\kappa = +1$), the main spectrum always consists of complex conjugate

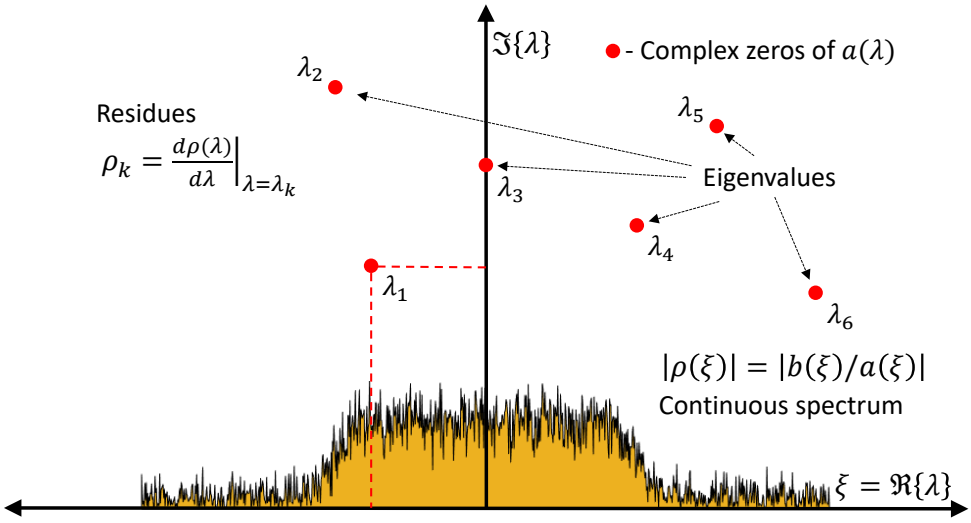


Figure 1.3: Exemplary NF spectrum (anomalous dispersion case), containing solitons (discrete eigenvalues) and continuous nonlinear spectrum. Based on Figure 1 in “Nonlinear Fourier transform for optical data processing and transmission: advances and perspectives,” by S. K. Turitsyn et al. (<https://doi.org/10.1364/OPTICA.4.000307>). Copyright 2017 Optica Publishing Group.

pairs. The main spectrum can thus be described by $g + 1$ points in the upper-half complex plane \mathbb{H} with the value g being the genus of the solution. Similar to the eigenvalues of a vanishing solution, the main spectrum of a periodic solution is invariant. When $\Im\{\lambda_k\} = 0$, the eigenvalue is a degenerate point and does not contribute to the dynamics of $q(t, x)$ [24]. Hence it is not considered a part of the main spectrum.

The dynamics of the signal are captured by the time- and space-dependent part of the spectrum known as the *auxiliary spectrum*. It is formed of g complex points $\mu_j(t, x)$ and g sheet indices $\sigma_j(t, x) = \pm 1$. At $\mu_j(t, x)$ the off-diagonal term \mathbf{M}_{12} of the monodromy-matrix vanishes:

$$\mathbf{M}_{12}(t, x, \mu_j(t, x)) = \mathbf{M}_{21}(t, x, \mu_j^*(t, x)) = 0. \quad (1.12)$$

For finite-genus solutions, the evolution of the auxiliary spectrum is governed by a set of coupled partial differential equations [24]:

$$\frac{\partial \log q(t, x)}{\partial t} = 2i \left(\sum_{j=1}^g \mu_j(t, x) - \frac{1}{2} \sum_{k=1}^{2g+2} \lambda_k \right), \quad (1.13)$$

$$\frac{\partial \mu_j(t, x)}{\partial t} = \frac{-2i\kappa\sigma_j(t, x) \sqrt{\prod_{k=1}^{2g+2} (\lambda_k - \mu_j(t, x))}}{\prod_{l \neq j} (\mu_j(t, x) - \mu_l(t, x))}, \quad (1.14)$$

$$\frac{\partial \mu_j(t, x)}{\partial x} = 4i\kappa\sigma_j(t, x) \left(\sum_{l=1}^g \mu_l - \frac{1}{2} \sum_{k=1}^{2g+2} \lambda_k - \mu_j \right) \frac{\sqrt{\prod_{k=1}^{2g+2} (\lambda_k - \mu_j(t, x))}}{\prod_{l \neq j} (\mu_j(t, x) - \mu_l(t, x))}. \quad (1.15)$$

For a given set of main spectrum points λ_k , the solution to these equations is fully specified by the initial conditions $\mu_j(0, 0)$, $\sigma_j(0, 0)$ and $q(0, 0)$. The absolute value of $q(0, 0)$ is determined by the initial condition for the auxiliary spectrum, while the phase is a free parameter. Not every choice of auxiliary spectrum corresponds to a periodic solution of the NLSE. Although the evolution of the periodic spectrum appears more complicated than that for the vanishing spectrum, it can still be solved in closed form.

Kotlyarov and Its noted that the auxiliary variables $(\mu_j(t, x), \sigma_j(t, x))$ should be interpreted as points on a Riemann surface. A Riemann surface is a one-dimensional complex manifold that can be thought of as a deformed version of the complex plane. Locally near every point on the Riemann surface, it looks like a patch of the complex plane, but its global topology can be quite different like a sphere or a torus [47]. As an example consider the function $f(\lambda) = \sqrt{\lambda}$, $\lambda \in \mathbb{C}$. If we set the discontinuity of the square-root function to be along the negative real line, the associated Riemann surface can be visualized as in Figure 1.4. The figure has been taken from [47]. From the figure we can see that the function takes two values for most λ . We can denote this mathematically as $f(\lambda) = \sigma\sqrt{\lambda}$, where the sheet index $\sigma = \pm 1$ indicates the two solutions of the equation $\lambda = z^2$. Now consider the multi-valued function

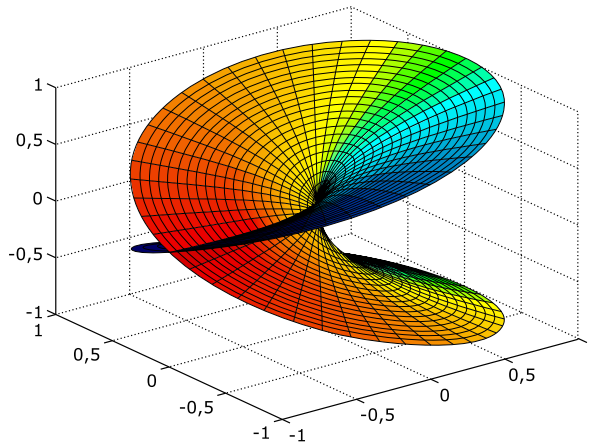


Figure 1.4: Riemann surface for the function $f(\lambda) = \sqrt{\lambda}$. The two horizontal axes represent the real and imaginary parts of λ , while the vertical axis represents the real part of $\sqrt{\lambda}$. The imaginary part of $\sqrt{\lambda}$ is represented by the coloration of the points. For this function, it is also the height after rotating the plot 180° around the vertical axis. Leonid 2, CC BY-SA 3.0 [48], via Wikimedia Commons.

$$P(\lambda) = \sigma \sqrt{\prod_{k=1}^{g+1} (\lambda - \lambda_k)(\lambda - \lambda_k^*)}, \tag{1.16}$$

where $\lambda_1, \dots, \lambda_{g+1}$ are the points in the main spectrum, $\sigma = \pm 1$ and g is the genus. On the associated Riemann surface, the multi-valued function $P(\lambda)$ can be considered a single-valued analytic function. A point on the surface is denoted by $p = (P, \lambda)$. Each value λ of

the projective plane $\mathbb{C} \cup \infty$ corresponds to two points on the Riemann surface. They are distinguished by their sheet indices $\sigma = \pm 1$. The points where $P = 0$ are called branch points and they correspond to points in the main spectrum. As an auxiliary spectrum point μ_j moves over the Riemann surface, the sheet index σ_j keeps track of the sheet changes.

The inverse periodic NFT problem is solved primarily either through the algebro-geometric approach [23], [24], [41], [42] or through the solution of a Riemann-Hilbert problem [44]. Here we present the algebro-geometric approach in which we can write an analytical expression for a finite-genus solution using the Riemann theta function. The Riemann theta function is defined as [49]

$$\theta(\mathbf{z} | \mathbf{\Omega}) = \sum_{\mathbf{n} \in \mathbb{Z}^g} e^{2\pi i (\frac{1}{2} \mathbf{n} \cdot \mathbf{\Omega} \mathbf{n} + \mathbf{n} \cdot \mathbf{z})}. \quad (1.17)$$

The argument $\mathbf{z} \in \mathbb{C}^{g \times 1}$ and the period-matrix $\mathbf{\Omega} \in \mathbb{C}^{g \times g}$, and $\mathbf{\Omega}$ is symmetric ($\mathbf{\Omega}^T = \mathbf{\Omega}$) with a strictly positive definite imaginary part. The scalar product of the integer vector \mathbf{n} with \mathbf{z} is denoted as

$$\mathbf{n} \cdot \mathbf{z} = \sum_{i=1}^g n_i z_i \quad (1.18)$$

and

$$\mathbf{n} \cdot \mathbf{\Omega} \mathbf{n} = \sum_{i,j=1}^g \Omega_{ij} n_i n_j. \quad (1.19)$$

A finite-genus solution of the focusing NLS equation is then given by

$$q(t, x) = K_0 \frac{\theta\left(\frac{\mathbf{z}_-}{2\pi} | \mathbf{\Omega}\right)}{\theta\left(\frac{\mathbf{z}_+}{2\pi} | \mathbf{\Omega}\right)} e^{i\omega_0 t + ik_0 x}, \quad (1.20)$$

with $\mathbf{z}_{\pm} = \mathbf{k}x + \boldsymbol{\omega}t + \boldsymbol{\delta}_{\pm}$, where $\mathbf{\Omega}, \mathbf{k}, \boldsymbol{\omega}, \boldsymbol{\delta}_{\pm}, k_0, \omega_0$ and K_0 are constants. The positive definiteness of $\Im\{\mathbf{\Omega}\}$ guarantees the convergence of (1.17), for all values of \mathbf{z} . The constants are collectively known as the *Riemann spectrum*. The values $\mathbf{\Omega}, \mathbf{k}, \boldsymbol{\omega}, k_0, \omega_0$ and K_0 depend only on the main spectrum [23]. The vectors $\boldsymbol{\delta}_{\pm}$ are determined by the auxiliary spectrum.

1.3. NONLINEAR SCHRÖDINGER EQUATION IN FIBER-OPTIC COMMUNICATIONS

Fiber-optic communication systems are optical communication systems which transmit high frequency (~ 130 THz) electromagnetic carrier waves through optical fibers for information transmission. The availability of low-loss optical fibers and compact optical sources were essential in the development of fiber-optic communication systems [50, Section 8.1]. They allowed for the first-generation of systems to be developed in the 1970s. By 1980 commercial systems were installed. In the modern day, fiber-optic communication systems form the backbone of telecommunications and the internet. In Figure 1.5 we can see a simplified block diagram of a fiber-optic communication system (based on [50, Figure 8.3]). An electrical input with embedded information is converted

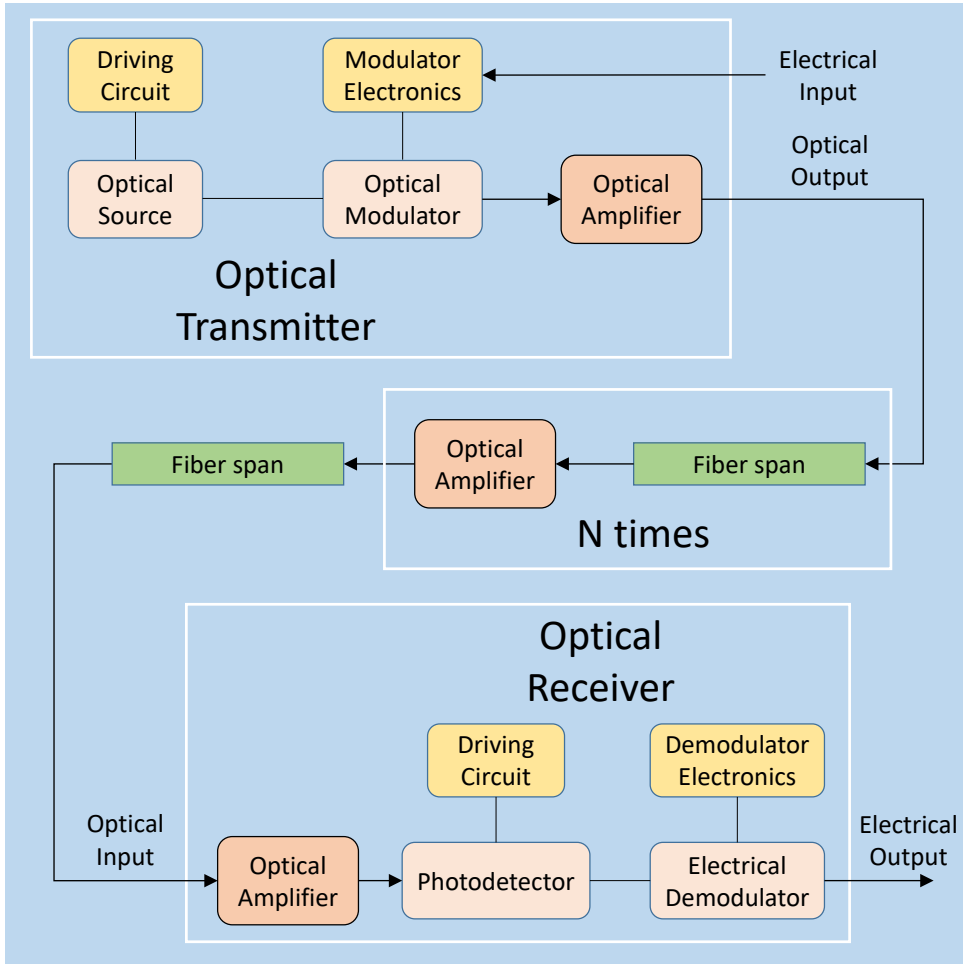


Figure 1.5: Simplified block diagram for a fiber-optic communication system based on [50, Figure 8.3].

into an optical signal by the transmitter which is then injected into the optical fiber. Optical fibers transmit light over long distances by confining the optical wave to the vicinity of a microscopic cylindrical glass core through a phenomenon known as total internal reflection [50, Section 8.2.2]. An optical fiber can be designed to support a single spatial mode by reducing its core diameter to below $10\mu\text{m}$. They are known as single-mode fibers and are made from silica. The envelope of a pulse $A(\tau, z)$ transmitted through a single-mode silica fiber can be approximated by the nonlinear Schrödinger equation [21, Section 2.6]

$$\frac{\partial A}{\partial z} + \frac{i\beta_2}{2} \frac{\partial^2 A}{\partial \tau^2} = i\gamma|A|^2 A - \frac{\alpha}{2} A, \quad (1.21)$$

where fiber losses are included through the α parameter while β_2 accounts for the second-order dispersion effects. The non-linearity parameter γ depends on the wave-

length of transmitted light, physical dimensions and material properties of the fiber. Due to losses in the fiber, the power of the transmitted signal decreases as it moves through the fiber. This loss is compensated by amplifying the signal back to the transmitted power level using optical amplifiers which are placed at regular intervals between sections of the fiber. The sections of the fiber are commonly known as *fiber spans*. The effect of the loss on the signal can be assessed better by rewriting (1.21) as

$$\frac{\partial B}{\partial z} + \frac{i\beta_2}{2} \frac{\partial^2 B}{\partial \tau^2} = i\gamma e^{-\alpha z} |B|^2 B, \quad A(z, \tau) = B(z, \tau) e^{-\alpha z/2}. \quad (1.22)$$

The variations in the signal power due to loss can thus be transformed into variations in the nonlinear parameter γ . We can approximate the varying nonlinear parameter with its average value over a span γ_{avg} . We then have a lossless NLS equation which is known as the path-average model [51]

$$\frac{\partial B}{\partial z} + \frac{i\beta_2}{2} \frac{\partial^2 B}{\partial \tau^2} = i\gamma_{\text{avg}} |B|^2 B. \quad (1.23)$$

The path-average model is an integrable approximation over a single fiber span. Between the spans, amplifiers raise the power such that the power is same at the beginning of each span. *Lumped* and *distributed* amplification schemes are two of the major techniques for loss management [21, Chap. 7]. In lumped amplification, the signal amplification is carried out within the amplifier between two spans. Distributed amplification on the other hand uses the transmission fiber itself for signal amplification by exploiting the nonlinear phenomenon of stimulated Raman scattering (SRS). The path-average approximation (1.23) is a good model when the variation of the signal power over the span is low. Hence it is a better approximation of the signal transmission through a lossy fiber with distributed amplification when compared with lumped amplification. In special fibers, the dispersion parameter β_2 and the nonlinear parameter γ can be varied within the span. For certain profiles of the two parameters, the NLS equation (1.21) remains integrable even in the presence of loss [52]. Such fibers can be combined with lumped amplification to form completely integrable channels over long distances [53].

At the receiver, the optical signal is converted back into an electrical signal. Most fiber-optic communication is digital, where the data is represented in terms of bits. In the simplest transmitter, the optical signal is switched on or off for a fixed duration to represent either a 1 or a 0. At the receiver, a photodetector is used to detect changes in the optical signal magnitude which is converted back to an electrical signal. More sophisticated techniques can be implemented using coherent receivers which can measure both the optical signal magnitude and phase. Usually the processing units at both ends of the fiber can transmit as well as receive signals. Such units are known as *transceivers*. The rate of data transmitted through a fiber (bit rate) is measured in bits/s. A commonly used figure of merit for communication systems is the bit rate-distance product BL , where B is the bit rate and L is distance after which the signal fidelity is lost [50, Section 8.1]. Modern systems can achieve BL product of 10^{18} bits/s-km [50, Figure 8.2]. Many bitstreams can be transmitted at the same time by spacing them apart in the frequency domain. This technique is known as *wavelength division multiplexing* (WDM). The (de-)multiplexing of the bitstreams can be done in the optical domain, therewith circumventing the bandwidth constraints of electrical components. The total data rate thus depends upon the

data rate of each bitstream and the frequency window available for all the bitstreams. A figure of merit known as *spectral efficiency*, measured in bits/s/Hz, is commonly used to compare communication systems. Modern transceivers employing sophisticated hardware and modulation techniques can achieve spectral efficiencies up to 8.3 bits/s/Hz over a distance of 960 km [54].

1.3.1. SOLITON COMMUNICATIONS

Access to the internet has grown vastly in the past two decades allowing the world to be more interconnected than ever. The number of users, traffic per user and applications dependent on the internet is projected to keep increasing in the foreseeable future [55]. Keeping up with the rising demand requires deployment of new fibers but also maximizing the data throughput of existing networks. Data throughput could be potentially increased by expanding the bandwidth in WDM systems or by employing more spatial degrees of freedom using space-division multiplexing (SDM) in multi-mode fibers. An increase in the rate of data transmission through a single-mode fiber crudely translates to an increase in transmit power. For low transmit powers, the optical fiber behaves like a linear channel for which the data transmission capacity of conventional systems increases with the power. However at high powers, the nonlinear effects are significant and lead to a reduction in the data transmission capacity of conventional systems [56, Section 10]. The mitigation of nonlinear propagation effects is hence an important topic of research and is expected to allow for a significant increase in the data transmission capacity of optical fibers [56, Section 6].

As early as 1973, Hasegawa and Tappert [57] demonstrated using numerical simulations a novel approach for data transmission based on the concept of optical solitons. Solitons preserve their shape during propagation by counteracting the effect of dispersion through the Kerr nonlinearity as we saw in Figure 1.2. In an ideal lossless fiber, solitons can preserve their shape forever. Due to the losses in real fibers, they however start to degrade with increasing distance. The proposal by Hasegawa and Tappert exploited the nonlinearity rather than just trying to mitigate it. The data was to be embedded in the soliton amplitude. Optical solitons in a fiber were first observed in experiments by Mollenauer et al. in 1980 [58]. They experimentally demonstrated the feasibility of data transmission using solitons in 1988 [59]. These systems came to be known as “soliton transmission” or “soliton communication” systems. Due to the loss in the fiber, the transmitted pulses are not true solitons as their amplitude undergoes large changes. However, the pulse-width almost remains constant. They are called *average solitons* or *dynamic solitons* [60, Section 3] since the average nonlinearity (see Equation (1.23)) balances with the dispersion. Further research into soliton transmission systems identified several problems, the major ones being the Gordon-Haus effect and soliton timing jitter, interaction between two solitons and effects of high-order dispersion [61, Section 3]. Various ingenious techniques were developed to address these problems, allowing for better control on the behaviour of the solitons [61, Section 4]. In the latter half of the 1990s, management of the fiber dispersion emerged as a possible remedy for the many problems intrinsic to soliton transmission systems. The core idea of dispersion management is that the fiber dispersion is varied such that the imbalance between the nonlinear and dispersive effects is minimized [61, Section 5]. The transmit-

ted pulses are known as *dispersion managed solitons* and they behave on average like the ideal solitons [60, Section 3]. Employing dispersion managed solitons, the authors of [60] achieved 1 Tbit/s data transmission rate over a distance of 1500 km. The ideal soliton is a solution of the homogeneous (β_2 and γ are constant) conservative ($\alpha = 0$) NLS equation (1.21). In [62] Ganapathy et al. showed that nonlinear pulses with soliton like behavior can exist even for the inhomogeneous nonconservative NLS equation with periodic amplification. They achieved this by designing special profiles for the dispersion and nonlinearity parameters which allowed them to control the behaviour of the dispersion managed solitons. Controlling more than one aspect of the dispersion managed solitons at the same time however remains a challenging problem. Dispersion management as described above requires special fibers with variable parameters which can be difficult to manufacture. More importantly it means that already deployed homogeneous fibers are not suited for dispersion-managed soliton transmission. In 1993, Hasegawa and Nyu proposed to use multiple solitons as an extension to single soliton transmission systems [63]. Multi-solitons can be viewed as several solitons merged together. This approach indirectly modulates the eigenvalues, which constitute part of the discrete nonlinear Fourier spectrum (see Section 1.2.1). They named this approach “eigenvalue communication”. In their approach a signal with multiple soliton components indicated by different eigenvalues is transmitted. This allows for more information to be transmitted per pulse. In recent years, advancements in optical and digital signal processing (DSP) hardware have led to renewed interest in systems exploiting solitons and more generally in the application of nonlinear Fourier transforms for optical data processing and transmission [35]. Fiber-optic communication systems based on nonlinear Fourier transforms started gaining considerable attention around 2013 [64]–[71]. The idea of modulating the nonlinear Fourier spectrum is now commonly known as *nonlinear frequency division multiplexing* (NFDM) in analogy to OFDM for linear channels.

1.3.2. NONLINEAR FREQUENCY DIVISION MULTIPLEXING SYSTEM DESIGN

The nonlinear Fourier spectrum has multiple degrees of freedom which can be modulated with data. In principle, all of them can be used simultaneously. However, that is not straightforward as it is difficult to control pulse properties such as temporal spread and bandwidth. We will first consider the modulation of the discrete and continuous spectra, i.e., we modulate the nonlinear spectrum for vanishing signals. As discussed in Section 1.2.1, the reflection coefficient $\rho(\lambda) \forall \lambda \in \mathbb{R}$ constitutes the continuous spectrum while the the eigenvalues λ_k and the residues ρ_k form the discrete spectrum. The $b(\lambda)$ scattering coefficient can be modulated instead of the reflection coefficient $\rho(\lambda)$. Similarly, the norming constants b_k can be modulated instead of the residues ρ_k . In Figure 1.6 we can see a simplified block diagram of a NFDM system based on the vanishing NFT. The data to be transmitted is represented by the bits which are embedded by the NFDM modulator into either or both the discrete and continuous spectrum. The inverse NFT is then used to obtain the corresponding vanishing signal. This signal is transmitted through the optical channel. At the receiver, the forward NFT is used to compute the nonlinear spectrum of the received signal. The effect of the channel is compensated in the equalizer using the relation (1.10). The compensation is exact only when the optical channel represents the lossless homogeneous NLS equation (1.2). The NFDM demod-

ulator then extracts the information bits from the equalized spectrum. In the following we will discuss specific modulation strategies that have been devised for the discrete and continuous spectrum modulation.

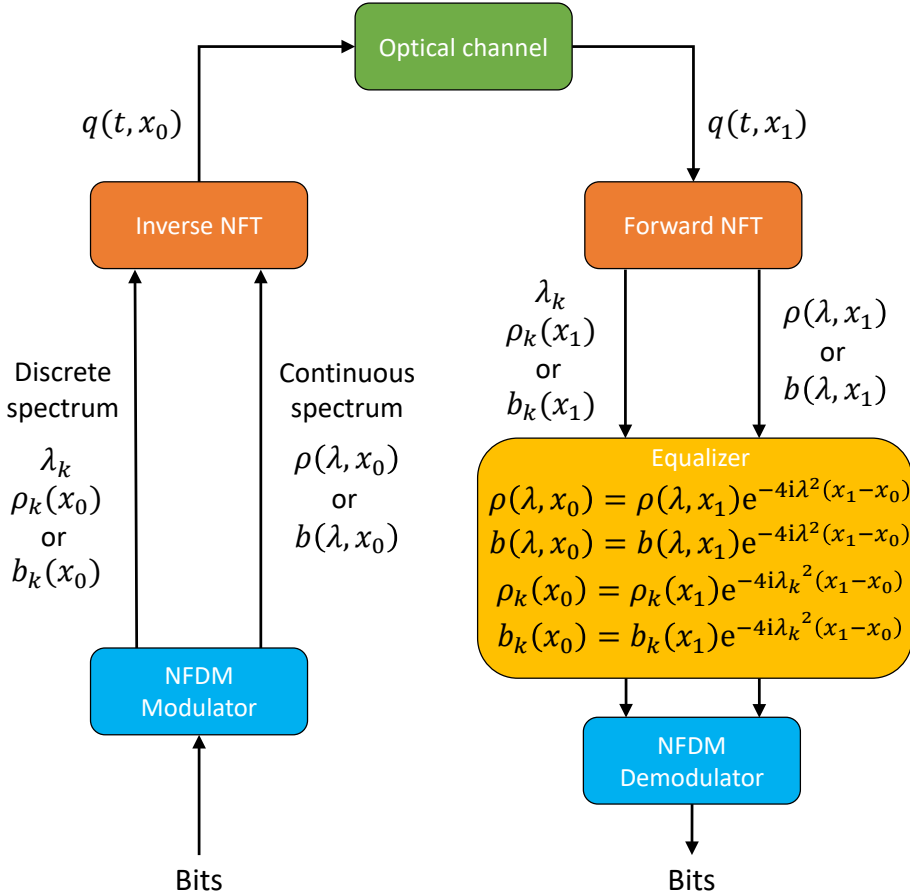


Figure 1.6: Simplified block diagram of a NFDN system based on the vanishing NFDN.

DISCRETE SPECTRUM MODULATION

In systems that modulate only the discrete spectrum, the eigenvalues λ_k and/or the residues ρ_k (or the norming constants b_k) can be modulated. The “eigenvalue communication” approach by Hasegawa and Nyu is a type of discrete spectrum modulation. Early discrete spectrum modulated systems [72], [73] employed on-off keying. In such systems, the bits are mapped to a set of eigenvalues. The eigenvalues are included in the signals depending on the corresponding bits. The spectral efficiency of on-off keying systems was observed to decrease with increasing size of the set [72, Table 1]. This

was partially attributed to the choice of the residues. The authors in [74] improved on the on-off keying design by optimizing the choice of the discrete spectrum. Later works [75]–[77] developed and studied systems which modulated the residues for a fixed set of eigenvalues. These systems allowed for better control over the characteristics of the signals. In [78], authors studied the noise correlation between the scattering coefficients and demonstrated that systems that modulate the norming constants b_k have higher robustness to noise when compared to systems that modulate the residues ρ_k . To develop systems with higher spectral efficiencies, the authors in [79] studied the evolution of the signal duration and bandwidth of multi-soliton pulses during propagation. They optimized the location of eigenvalues and the magnitudes of residues for 2- and 3-solitons in order to minimize the time-bandwidth product. They observed that the time-bandwidth product per eigenvalue improves very slowly with the multi-soliton order. In another interesting observation they noted that the optimal multi-soliton pulses looked similar to a train of single solitons. Interactions of the soliton pulses [80] and the impact of noise [81] have also been studied to aid system design. Optimizing the spectrum to minimize the time-bandwidth product for higher-order solitons remains a highly challenging problem. Joint modulation of the eigenvalues and the residues was proposed in [82] as another way to improve spectral efficiency.

In Section 1.2.1 we mentioned that it is possible to have eigenvalues with higher multiplicity (zeros of higher order). In [83] the modulation of such higher multiplicity eigenvalues was proposed. The multiplicity is an additional degree of freedom that may be beneficial in improving system efficiency.

CONTINUOUS SPECTRUM MODULATION

NFDM systems that modulate the continuous spectrum are inspired by conventional OFDM systems where the data to be transmitted is embedded in the linear frequency domain. The reflection coefficient $\rho(\lambda)$ which constitutes the continuous spectrum is defined for all $\lambda \in \mathbb{R}$. To transmit discrete data, we make use of special functions called *carriers*. A signal $u(\lambda)$ is then defined in the nonlinear frequency domain as the sum of modulated carriers,

$$u(\lambda) = \sum_{k=-M}^M c_k \Psi(\lambda - k\Delta_\lambda), \quad (1.24)$$

where c_k are complex valued symbols from a fixed constellation alphabet and $\Psi(\lambda)$ is the carrier function. The symbols are determined from the bits of data that are to be transmitted. The raised-cosine function and the sinc function are examples of commonly used carrier functions as they are bandlimited and satisfy the Nyquist intersymbol interference criterion. For certain choices of carrier function and spacing Δ_λ , the carriers are orthogonal. In ρ -modulated systems we set $\rho(\lambda) = u(\lambda)$. This construction is carried out in the modulator (Figure 1.6). At the receiver, after equalization, knowledge of the carrier function and the spacing is used to retrieve the transmitted symbols and ultimately the bits of data. The characteristics of ρ -modulated systems have been studied extensively through simulations and experiments [84]–[87]. They have also been combined with discrete spectrum modulation [88]. In practical implementations of NFDM systems, there are many things to consider such as the choice of normalization parameters and scaling of symbols to generate signals of desired characteristics. The vanishing

NFT requires the signals to be transmitted in the so-called *burst mode* where we need to have sufficiently large gaps between the signals to avoid interference due to channel memory. Hence for efficient data transmission, it is beneficial to have long signals that decay quickly on both ends since that reduces the impact of the gap. For ρ -modulated systems with carriers such as the sinc function, the NFDm signals tend to have slowly decaying tails [89, Figure 4]. Truncating these tails prematurely can introduce non-trivial changes to the ρ -coefficient of the signals, which can impact performance. To address this problem, it was proposed by Wahls [90] to instead modulate the b -coefficient, i.e. set $b(\lambda) = u(\lambda)$. For certain carrier functions it was shown that b -modulation leads to signals with finite duration. This idea was proven rigorously in [91]. It has been observed that b -modulated systems perform better than ρ -modulated systems [92]. Next to the improved time-domain pulse shapes, this can also be attributed to the relatively higher robustness of the b -coefficient to noise [78].

The ideas of NFDm have been extended to the Manakov equations [93]–[100]. In [99], a simulated 56GHz dual-polarization b -modulated system was shown to have a spectral efficiency of 3.6bits/s/Hz/Pol with net data rate of 400Gbits/s over 960km. More recently a 5GHz b -modulated system [101] demonstrated a spectral efficiency of 5.51 bits/s/Hz over a distance of 960km. However, the design of high data rate NFDm systems with spectral efficiencies comparable to conventional systems remains a challenge.

PERIODIC SPECTRUM MODULATION

We now discuss modulation of the main and auxiliary spectra which form the nonlinear spectrum for periodic signals. Here we only consider finite-genus solutions. As discussed in Section 1.2.2, the main spectrum of a genus g solution comprises of $g+1$ complex points λ_k and g pairs of values $(\mu_j(t, x), \sigma_j(t, x))$ form the auxiliary spectrum. In Figure 1.7 we can see a simplified block diagram of a NFDm system based on the periodic NFT. The idea of periodic NFT based NFDm was first demonstrated in [102], [103]. Instead of computing the inverse NFT numerically, the authors made use of special periodic signals whose spectrum is known analytically. The data was mapped to certain parameters of the signals which was equivalent to main spectrum modulation. In [104], using such a special signal known as a perturbed plane wave, the authors demonstrated a spectral efficiency of 2.5bits/s/Hz over a distance of 1000km. In [105], a system was designed using genus-2 solutions with certain symmetry in the main spectrum which allows the solution to be expressed analytically in terms of elliptic functions [106]. The work [107] was the first demonstration of a NFDm system that computed the inverse numerically. They made use of the Riemann-Hilbert problem approach [44] to compute genus-1 solutions. The data was embedded in the imaginary part of the main spectrum points. In [108] authors proposed a system which employed the algebro-geometric approach to compute quasi-periodic solutions. The relationship between the main spectrum points and period of the solution is non-trivial for $g \geq 2$. Hence the authors in [109] adjusted the Riemann spectrum of a genus-2 solution such that the resulting solution would have a predefined period and would closely approximate the genus-2 solution. Auxiliary spectrum modulation was demonstrated for the first time in [110]. The Riemann-Hilbert problem approach was employed to compute solutions of upto genus-10. In [111], [112] the authors experimentally demonstrated transmission of a modulated

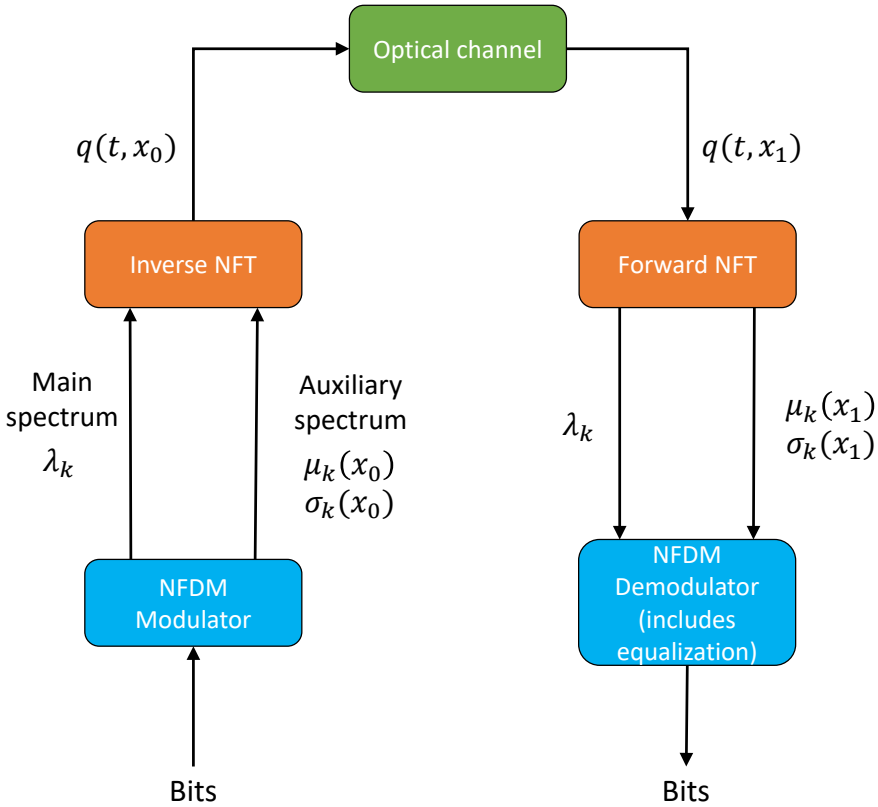


Figure 1.7: Simplified block diagram of a NFDN system based on the periodic NFT.

genus-2 solution over a distance of 2000 km. They could achieve a spectral efficiency of only 0.45 bits/s/Hz. They note that spectral efficiency can be improved by increasing the genus and the length of the signal. They also remark that periodic NFT based NFDN may be extendible to the Manakov equations. Additional research is necessary to investigate whether the systems based on the periodic NFT can achieve similar or better performance compared to those that use the vanishing NFT.

1.4. NUMERICAL ASPECTS OF NONLINEAR FOURIER TRANSFORMS

NFT based communication systems are required to compute the nonlinear Fourier spectrum of complex signals. Like the Fourier transform, computing the nonlinear Fourier

transform analytically is not always possible. In fact there are only a handful of signal types for which the nonlinear Fourier spectrum is known analytically [68], [106], [113]–[115]. As the NFT does not obey the principle of superposition, the spectrum of more complicated waveforms cannot be constructed from these simpler ones by shifting, scaling and adding. Thus numerical algorithms have been developed for performing the forward and inverse NFTs. In this section we present a common framework for typical numerical NFT algorithms for the NLS equation with vanishing and periodic boundary conditions. First, forward NFT algorithms are considered. Then, inverse NFT algorithms are discussed.

1.4.1. NUMERICAL ASPECTS OF FORWARD NFTS

The forward NFT involves the computation of the nonlinear Fourier spectrum of a given signal. We discussed the NFT for vanishing signals in Section 1.2.1 and for periodic signals in Section 1.2.2. For both the vanishing and periodic signals, we need to solve the Zakharov-Shabat (1.5) problem. In the following, we will first discuss numerical methods for solving the Zakharov-Shabat problem. We will then explain the computation of numerical approximations of the nonlinear spectra.

NUMERICALLY SOLVING THE ZAKHAROV-SHABAT SYSTEM

We consider the case where the numerical computation of NFTs is carried out with finitely many discrete signal samples. Computing NFTs numerically requires us to make assumptions which are specific to the signal's boundary condition. For the vanishing case we make the following assumptions:

1. The support of the signal $q(t)$ is truncated to a finite interval, $t \in [T_-, T_+]$, instead of $t \in (-\infty, \infty)$. The values T_{\pm} need to be chosen such that the resulting truncation error is sufficiently small. The approximation is exact if $q(t) = 0 \forall t \notin [T_-, T_+]$.
2. The interval $[T_-, T_+]$ is divided into D subintervals of width $h = (T_+ - T_-)/D$. We assume that the signal is sampled at the midpoints of each subinterval $t_n = T_- + (n + 0.5)h$, $n = 0, 1, \dots, D - 1$ such that $q_n := q(t_n)$.

In Figure 1.8 we can see the example of a vanishing signal and its equispaced sampling in the interval $[-8, 8]$.

Using numerical ODE solvers for equispaced data, we solve (1.5) for an approximation $\hat{\mathbf{V}}(T_+, \lambda)$ of $\mathbf{V}(T_+, \lambda)$. We do this for two initial conditions $\hat{\mathbf{V}}_1(T_-, \lambda) = [1 \ 0]^T$ and $\hat{\mathbf{V}}_2(T_-, \lambda) = [0 \ 1]^T$. Stacking the results we get a numerical approximation

$$\hat{\mathbf{M}}(\lambda) = [\hat{\mathbf{V}}_1(T_+, \lambda) \ \hat{\mathbf{V}}_2(T_+, \lambda)]. \quad (1.25)$$

The matrix $\hat{\mathbf{M}}(\lambda)$ is a direct approximation of the monodromy matrix $\mathbf{M}(\lambda)$ in (1.11) for the periodic NFT, or upon an additional transformation which will be discussed later an approximation of the scattering matrix $\mathcal{S}(\lambda)$ in (1.8) for the vanishing NFT.

One of the first works on forward NFT methods for the NLS equation was by Boffetta and Osborne [116]. Their method is commonly referred to as the “Boffetta-Osborne” or

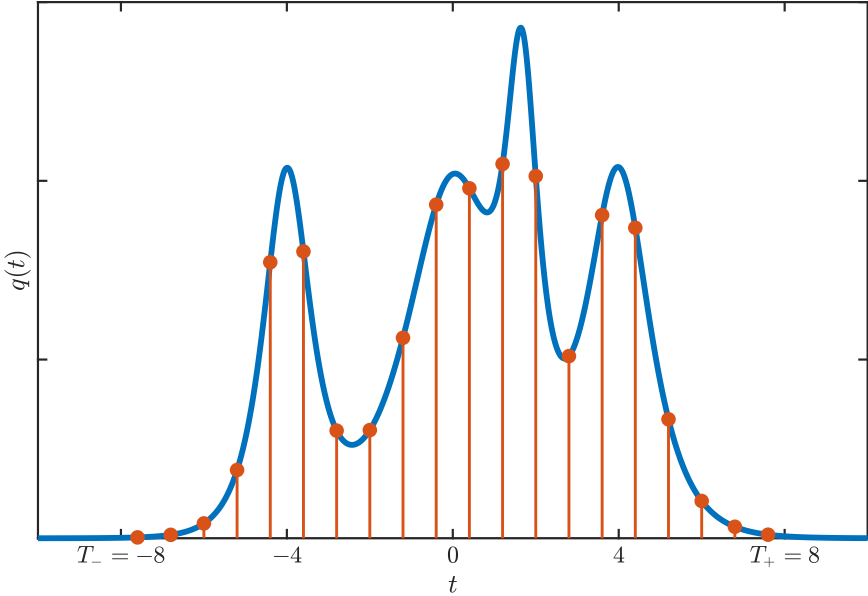


Figure 1.8: Example of a vanishing signal (blue) and its equispaced sampling (red) in the interval $[-8, 8]$.

“BO” method. In this method the approximation $\hat{\mathbf{M}}_s(\lambda)$ has the structure

$$\begin{aligned} \hat{\mathbf{M}}_s(\lambda) &= \left(\prod_{n=0}^{D-1} \mathbf{G}_n(\lambda) \right) \\ &= \mathbf{G}_{D-1}(\lambda) \mathbf{G}_{D-2}(\lambda) \cdots \mathbf{G}_0(\lambda). \end{aligned} \quad (1.26)$$

The terms $\mathbf{G}_n(\lambda)$ are the transition matrices of the individual sampling intervals. In the Boffetta-Osborne method, the signal value is assumed to be constant in each sampling interval. In Figure 1.9 we can see an illustration of this assumption for the signal from Figure 1.8. In each interval, the Zakharov-Shabat system can be solved analytically under this assumption. The transition matrices are given by

$$\begin{aligned} \mathbf{G}_n(\lambda) &= \expm(h\mathbf{C}_n(\lambda)), \\ \mathbf{C}_n(\lambda) &= \begin{bmatrix} -i\lambda & q_n \\ -\kappa q_n^* & i\lambda \end{bmatrix}, \end{aligned} \quad (1.27)$$

where $\expm(\cdot)$ is the matrix exponential. This method is also known as the midpoint exponential integrator. We shall refer to this as the *base numerical method*. For a 2×2 matrix the exponential can be computed in closed form [117]. This allows for simple yet efficient implementations of the BO method. Other authors have experimented with various base numerical methods to develop more efficient algorithms [118]–[123], many of which have the same structure as (1.26).

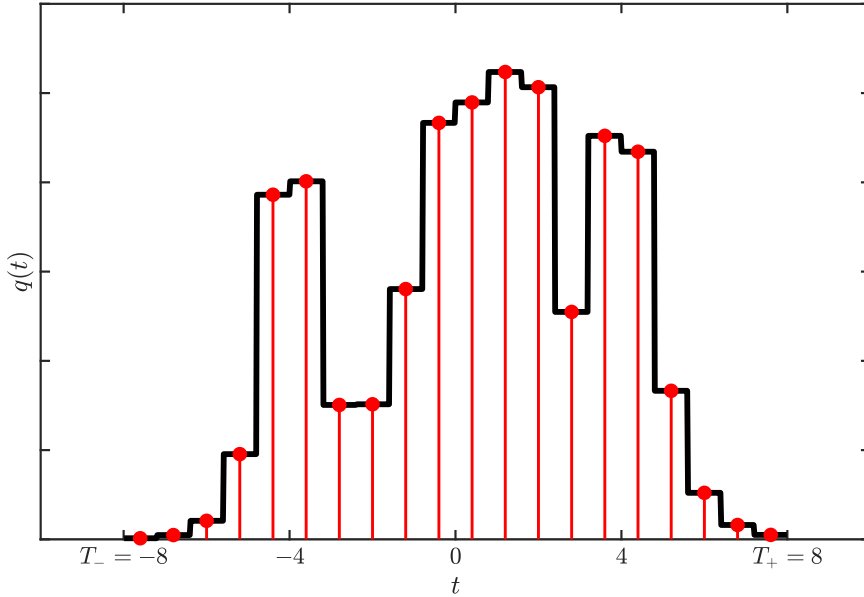


Figure 1.9: Constant signal approximation (black) in each subinterval centered around the sampled location (red).

COMPUTING CONTINUOUS SPECTRUM OF THE VANISHING NFT

The continuous spectrum consists of the reflection coefficient $\rho(\lambda) = b(\lambda)/a(\lambda)$, which we can compute by first computing the scattering coefficients $a(\lambda)$ and $b(\lambda)$ for a desired value of lambda by solving equation (1.9). Using the numerical approximations $[\hat{\mathbf{V}}_1(T_+, \lambda) \quad \hat{\mathbf{V}}_2(T_+, \lambda)]$ from (1.25) and accounting for the boundary conditions (1.6), an approximation of the scattering matrix $\hat{\mathcal{S}}(\lambda)$ can be obtained by solving

$$[\hat{\mathbf{V}}_1(T_+, \lambda) \quad \hat{\mathbf{V}}_2(T_+, \lambda)] \begin{bmatrix} e^{-i\lambda T_-} & 0 \\ 0 & -e^{i\lambda T_-} \end{bmatrix} = \begin{bmatrix} e^{-i\lambda T_+} & 0 \\ 0 & e^{i\lambda T_+} \end{bmatrix} \hat{\mathcal{S}}(\lambda). \quad (1.28)$$

From the definition of the scattering matrix (1.8), we can write the approximations $\hat{a}(\lambda)$ and $\hat{b}(\lambda)$ of the scattering coefficients $a(\lambda)$ and $b(\lambda)$ as

$$\hat{a}(\lambda) = (\hat{\mathbf{V}}_1(T_+, \lambda))_1 e^{i\lambda(T_+ - T_-)}, \quad \hat{b}(\lambda) = (\hat{\mathbf{V}}_1(T_+, \lambda))_2 e^{-i\lambda(T_+ + T_-)}. \quad (1.29)$$

Finally, a numerical approximation of the reflection coefficient is given by $\hat{\rho}(\lambda) = \hat{b}(\lambda)/\hat{a}(\lambda)$.

COMPUTING EIGENVALUES OF THE VANISHING NFT

As discussed in Section 1.2.1, in case of the focusing NLS equation, the eigenvalues are the zeros of $a(\lambda)$ in the complex upper half-plane. If present, they can be approximated by the zeros of $\hat{a}(\lambda)$. This root-finding problem has been tackled through various approaches [69], [116], [118]–[120], [124]–[126]. Boffetta and Osborne [116] proposed the use of the Newton-Raphson method for localizing the zeros. The Newton-Raphson

method is an iterative root-finding method and requires approximation of the derivative $a_\lambda(\lambda)$ which is usually computed by extending the transfer matrices $\mathbf{G}_n(\lambda)$ in (1.26). The use of other root-finding methods like the Muller's method and the secant method that do not require the derivative have also been proposed [118]. The rate of convergence of these methods is generally lower than that of the Newton-Raphson method but they usually have a larger region of convergence. All the iterative methods require initial guesses for the roots within this region to converge successfully. Hence for arbitrary signals the initial guesses are taken to be on a grid or other root-finding methods are used to find them. The authors in [69] propose to compute the eigenvalues of finite matrix approximations of the Zakharov-Shabat operator \mathcal{L}

$$\mathcal{L} = \begin{pmatrix} i \frac{\partial}{\partial t} & -iq(t) \\ -iq^*(t) & -i \frac{\partial}{\partial t} \end{pmatrix}, \quad (1.30)$$

which can be obtained by rearranging (1.5) to an eigenvalue equation of the form $\mathbf{L}\mathbf{V} = \lambda\mathbf{V}$. The proposed methods do not require initial guesses and hence are better suited for arbitrary signals. Their computation cost is however higher than the iterative methods and grows quickly with the number of signal samples. Hence it was proposed to use them with a subset of the samples to find preliminary approximations of the zeros which are used as initial guesses for the iterative methods. In [124], the authors proposed another global root-finder based on special contour integrals. The idea is to approximate $a(\lambda)$ in a region of the complex plane with a polynomial, the zeros of which are then the approximations of the discrete eigenvalues. The coefficients of the polynomial can be computed via certain contour integrals. The method can be very accurate if the polynomial approximation has low degree. If the polynomial degree is high, computing the roots of the polynomial becomes ill-conditioned even if the polynomial is a very accurate approximation of $a(\lambda)$. To overcome this limitation, the authors in [124] suggest the use of multiple local polynomial approximations. This however creates the challenge of choosing these regions and also increases the computation cost. The authors of [125] proposed a similar approach which utilizes a rational approximation. The $a(\lambda)$ coefficient can be written as [10]

$$a(\lambda) = a_c(\lambda) \prod_{k=1}^N \frac{\lambda - \lambda_k}{\lambda - \lambda_k^*}, \quad \forall \lambda \in \mathbb{R}, \quad (1.31)$$

where $a_c(\lambda)$ is the part corresponding to the non-solitonic component of the signal. In the algorithm proposed in [125], the approximation $\hat{a}(\lambda)$ is computed using a standard ODE solver. The approximation $\hat{a}_c(\lambda)$ is computed from $\hat{a}(\lambda)$ using the Hilbert-transform. An optimization algorithm is then used to identify the number of zeros N and locate the zeros $\hat{\lambda}_k$. Assuming the continuous spectrum has already been computed, this approach can be very efficient. The optimization step can however become ill-conditioned when the number of zeros is large as in the case of the algorithm in [124]. The algorithm proposed in [126] shares aspects with those in [124] and [125]. It is based on the theory that for a meromorphic function such as $a(\lambda)$, on a closed non-intersecting contour of a region containing its N zeros, the argument $\arg\{a(\lambda)\} \in (-\pi, \pi]$ has N discontinuities. The algorithm starts by choosing a region in the upper-half complex plane based on the linear Fourier spectrum of the signal. The jumps in $\arg\{a(\lambda)\}$

along a contour which encloses this region are identified. A tracking routine such as marching-squares is used to track the individual discontinuities to the zeros. The algorithm is efficient and is demonstrated to work on signals where other algorithms such as [124] fail. The algorithm heuristically determines the step-sizes for locating the discontinuities and tracking them. Hence it can miss some discontinuities and fail to find approximations for all the eigenvalues. The eigenvalues form only a part of the discrete spectrum. We also need to compute the associated norming constants. Computing the norming constants numerically is a challenging problem on its own. We shall not provide the specifics here. Please see [127] and references therein for recent developments.

FAST NFT ALGORITHMS

Given D samples of a signal $q(t)$, computing the approximation $\hat{\mathbf{M}}(\lambda)$ in (1.26) for D values of λ has a computational cost of $\mathcal{O}(D^2)$. This is akin to the cost of computing the discrete Fourier transform naively. In the work [128], Wahls and Poor presented an approach where the components of $\hat{\mathbf{M}}(\lambda)$ are approximated as rational functions of a variable z related to λ . The transition matrices $\mathbf{G}_n(\lambda)$ in (1.26) are approximated by rational function matrices $\hat{\mathbf{G}}_n(z)$, with $z = z(\lambda)$ being a mapping. We then have the rational matrix function approximation

$$\begin{aligned} \hat{\mathbf{M}}_f(\lambda) &= \left(\prod_{n=0}^{D-1} \hat{\mathbf{G}}_n(z) \right) \\ &= \hat{\mathbf{G}}_{D-1}(z) \hat{\mathbf{G}}_{D-2}(z) \cdots \hat{\mathbf{G}}_0(z). \end{aligned} \quad (1.32)$$

In numerical implementations of this approach, the polynomials are represented by their coefficient vectors. We first compute the products of the pairs of matrices $\hat{\mathbf{G}}_n(z)$ and $\hat{\mathbf{G}}_{n-1}(z)$. After all the pairs are multiplied, the same procedure is applied to their products. The process is repeated until only one matrix is left. The multiplication of polynomials is equivalent to the convolution of their coefficient vectors which can be computed efficiently using the FFT algorithm. The resulting polynomials can be evaluated for multiple values of z efficiently using the chirp z -transform [129]. It was shown that the complexity of computing the continuous spectrum using this approach is $\mathcal{O}(D \log^2 D)$. These algorithms are thus commonly referred to as fast NFT algorithms. For low number of samples the base methods maybe faster than the corresponding fast methods. In case of the fast methods, the rational function matrix $\hat{\mathbf{G}}_n(z)$ can be exactly equal to $\mathbf{G}_n(\lambda)$ for some base numerical methods while being an approximation for others. If it is only an approximation, the accuracy of the fast NFT algorithm maybe lower than that of the base numerical method. Various base numerical methods and rational approximations have been explored by other authors [130], [131].

COMPUTING THE PERIODIC SPECTRUM

As discussed in Section 1.2.2, the periodic spectrum consists of the main spectrum and the auxiliary spectrum. To compute the spectrum of a periodic signal numerically, we make the following assumptions:

1. The period T of the signal is known.

2. The interval $[T_-, T_+] = [T_s, T_s + T]$ is divided into D subintervals of width $h = T/D$. The point T_s is an arbitrary starting point. We assume that the signal is sampled at the points $t_n = T_- + nh$, $n = 0, 1, \dots, D-1$ such that $q_n := q(t_n)$. Note that the sample points are shifted by $h/2$ compared the vanishing case. This is helpful in ensuring that the auxiliary spectrum which depends on t is computed correctly.

As for the vanishing case, we can solve the Zakharov-Shabat system numerically to obtain approximations $\hat{\mathbf{M}}(\lambda)$ of the Monodromy matrix (1.11). Approximations of the main spectrum points $\hat{\lambda}_k$ are found by solving $\hat{\mathbf{M}}_{11}(\hat{\lambda}_k) + \hat{\mathbf{M}}_{22}(\hat{\lambda}_k) \pm 2 = 0$ using root-finding algorithms. Similarly, approximations of the auxiliary spectrum points $\hat{\mu}_j$ are obtained by solving $\hat{\mathbf{M}}_{12}(\hat{\mu}_j) = 0$ (1.12).

1.4.2. NUMERICAL ASPECTS OF INVERSE NFTS

Unlike the discrete Fourier transform where the inverse problem is identical to the forward problem, inverse NFTs are more challenging. Inverting the continuous spectrum in the vanishing case is equivalent to solving a system of integral equations [11], [68]. This has been accomplished numerically by many authors [69], [132]–[137]. The complexity of such algorithms is usually $\mathcal{O}(D^2)$ or higher. Wahls and Poor extended the idea of using rational approximations to the inverse NFT problem [138], resulting in a first fast inverse NFT algorithm for multi-solitons with a complexity of $\mathcal{O}(D \log^2 D)$. The approach was explored further for other cases in [139]–[141]. Most of the inverse NFT algorithms invert the $\rho(\lambda)$ coefficient. In [90] a fast algorithm was proposed for inverting the b -coefficient which could be employed directly in b -modulated NFD systems. In the case of multi-soliton signals, the inverse problem reduces to the solution of a system of linear equations [11], [16], [142], [143]. They can be solved iteratively using so-called Darboux transforms [144], [145]. Darboux transforms can be combined with methods to invert the continuous spectrum to obtain a full inverse NFT algorithm. Numerical implementations of the Darboux transform can invert a discrete spectrum with K eigenvalues in $\mathcal{O}(K^2 D)$ operations [69], [120]. Multiple efforts have been made to further reduce the computational complexity of multi-soliton generation [146]–[148].

Numerical algorithms for the inverse periodic NFT have received relatively less attention when compared to the vanishing counterpart. The proposal to utilize the periodic NFT for fiber-optic communications has renewed interest for their development in recent years. The algorithms for generating finite-genus solutions are based either on the algebro-geometric approach [41] or the Riemann-Hilbert problem (RHP) approach [44]. In [107] the authors developed a periodic NFT based communication system using the RHP approach. They made use of a software package written in Wolfram Mathematica for solving Riemann-Hilbert problems [149]. A Julia based package is also available [150]. The authors in [107] note that solving the RH problem from [44] requires extra caution due to the presence of singularities at the end of jump contours. They provide a transformation of the RH problem to improve the numerical implementation. Nevertheless they remark that the software package [150] is not optimal for this particular application. For the algebro-geometric approach we first need to compute the constants $\mathbf{\Omega}, \mathbf{k}, \boldsymbol{\omega}, \boldsymbol{\delta}_\pm, k_0, \omega_0$ and K_0 , which form the Riemann spectrum (see Section 1.2.2), from the given finite-genus spectrum. Several advancements have been made on this front allowing for the Riemann spectrum to be computed accurately [111], [151]. Given the

Riemann spectrum, computing the finite-genus solution reduces to the evaluation of the Riemann theta function (1.20). The Riemann theta function is a form of a multi-dimensional Fourier series which is computed numerically by truncating the series [49]. For classical implementations [49], [152], [153], the number of coefficients in the summation grows exponentially with the number of dimensions/genus of the solution. This is known famously as the *curse of dimensionality* and it significantly limits the genus of the solutions for which the algebro-geometric approach is practical.

1.5. OUTLINE OF THE DISSERTATION

The principal objective of this dissertation is to understand and address numerical issues that currently complicate the practical use of nonlinear Fourier transform-based fiber-optic communication systems. To achieve this objective, new numerical algorithms are developed and theoretical analysis is performed.

The remaining chapters of the dissertation are structured as follows.

CHAPTER 2

Conventionally, the computation cost for a given number of samples is used as a performance metric for NFT algorithms. However, this does not give the complete picture as the accuracy depends on the underlying discretization of the Zakharov-Shabat problem. In Chapter 2 we look, for the first time in literature, at the accuracy achieved by NFT algorithms at comparable run times. We design new fast forward NFT algorithms that achieve accuracies that are orders of magnitudes better than existing methods, at comparable run times and even for moderate sampling intervals. The new algorithms are compared to existing algorithms in multiple, extensive numerical examples.

This chapter is based on the following publication:

S. Chimmalgi, P. J. Prins and S. Wahls, "Fast Nonlinear Fourier Transform Algorithms Using Higher Order Exponential Integrators," in *IEEE Access*, Vol. 7, No. 1, pp. 145161–145176, Dec. 2019.

CHAPTER 3

It has been suggested in the literature that the spectral efficiency of NFD systems based on the periodic NFT discussed in Section 1.3.2 may be improved by the use of high-genus solutions. The algebro-geometric approach discussed in Section 1.2.2 allows for the computation of finite-genus solutions using the Riemann theta function. The computation of multi-dimensional Riemann theta functions is however a notoriously difficult problem in high dimensions. This limits the tractable genera of solutions. In Chapter 3 we address the challenge of computing high dimensional Riemann theta functions efficiently. We investigate conventional computational methods theoretically and propose new methods. We demonstrate that one of the new methods allows us to compute solutions with genus much higher than what was previously feasible.

This chapter is based on the following publication:

S. Chimmalgi and S. Wahls, "On Computing Riemann Theta Functions," under review.

CHAPTER 4

In certain b -modulated systems such as in [92], a drop in performance was observed with increasing signal duration. This was typically attributed in the literature to limitations of the numerical algorithms of the forward and inverse NFT. We improved the NFT algorithms but were unable to achieve better performance. We speculated that the limitation might be fundamental to the b -modulator and not due to numerical problems. In Chapter 4 we investigate this question further by studying the relationship between the signal duration and transmit power for the b -modulator from [90]. The maximal transmit power is observed to decrease with the signal duration when bandwidth is fixed. We provide a new theoretical explanation for this behavior and validate it in simulations. These results explain that performance will degrade with signal duration even for numerical methods using infinite precision.

This chapter is based on the following publication:

S. Chimmalgi and S. Wahls, "Theoretical analysis of maximum transmit power in a b -modulator", in 45th European Conference on Optical Communication (ECOC2019).

CHAPTER 5

The derivation of the coupling between signal duration and power in Chapter 4 exploited specific features of a b -modulation system that had been used for the experiments in [92]. However, we soon realized that this explanation can be extended to most b -modulators. In Chapter 5 we show that for b -modulators, the nonlinear bandwidth, signal duration and power are coupled when singularities in the nonlinear spectrum are avoided. For fixed nonlinear bandwidth, the coupling results in an upper bound on the transmit power that decreases with increasing signal duration which in-turn leads to decrease in the signal-to-noise ratios conceivably affecting performance. This result is first of its kind as such a phenomenon is not known from conventional linear systems.

This chapter is based on the following publication:

S. Chimmalgi and S. Wahls, "Bounds on the Transmit Power of b -Modulated NFDN Systems in Anomalous Dispersion Fiber," in *Entropy: Special Issue Information Theory of Optical Fiber*, Vol. 22, No. 6, Article 639, 2020.

CHAPTER 6

The implementation of a numerical algorithm for computing a NFT is not as straightforward as for the linear Fourier transform. At the start of this project, there was not a single publicly available software library that implemented numerical NFTs. The absence of a reliable software library for computing NFTs resulted in a high entry cost for the use of NFTs as an engineering tool. For this reason, we have developed the "Fast

Nonlinear Fourier Transforms" (FNFT) software library. The library is open source and includes many tests, examples and extensive documentation. During the research work for this dissertation, we furthermore developed an open source software environment for the simulation of fiber-optic NFDm communication systems that is based on FNFT - NFDMLab [154]. In Chapter 6 we discuss the structure and features of the FNFT library and the NFDMLab simulation environment. We elaborate on the contributions of this dissertation to them and their scope in NFT based signal processing.

This chapter is based on the following publications:

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), 597, 2018.

S. Wahls et al., FastNFT/FNFT: Version 0.4.1, 2020

M. Brehler, C. Mahnke, S. Chimmalgi and S. Wahls, "NFDMLab: Simulating Nonlinear Frequency Division Multiplexing in Python," in *Proceedings of 2019 Optical Fiber Communications Conference and Exhibition (OFC)*, 2019, pp. 1-3.

2

FAST NONLINEAR FOURIER TRANSFORM ALGORITHMS USING HIGHER ORDER EXPONENTIAL INTEGRATORS

CHAPTER ABSTRACT

The nonlinear Fourier transform (NFT) has recently gained significant attention in fiber optic communications and other engineering fields. Although several numerical algorithms for computing the NFT have been published, the design of highly accurate low-complexity algorithms remains a challenge. In this chapter, we present new fast forward NFT algorithms that achieve accuracies that are orders of magnitudes better than current methods, at comparable run times and even for moderate sampling intervals. The new algorithms are compared to existing solutions in multiple, extensive numerical examples.

The text in this chapter has previously appeared in modified form in the open access article S. Chimmalgi, P. J. Prins and S. Wahls, "Fast Nonlinear Fourier Transform Algorithms Using Higher Order Exponential Integrators," IEEE Access, Vol. 7, No. 1, pp. 145161–145176, Dec. 2019. [155]. Its reuse is licensed under CC BY 4.0 [156]

2.1. INTRODUCTION

The fast Fourier transform (FFT) is a well-known success story in engineering. From a numerical point of view, the FFT provides a mere first-order approximation of the discrete-time Fourier transform one is actually interested in. Hence the success of the FFT is quite surprising. Upon closer inspection, it however turns out that approximations based on FFTs are very accurate if the signal is smooth [157]. Recently, *nonlinear* Fourier transforms (NFTs) have been gaining much attention in engineering areas such as fiber-optic communications [35], [56] and coastal engineering [158], [159]. NFTs are generalizations of the conventional Fourier transform that allow to solve specific nonlinear evolution equations in a way that is analogous to how Fourier solved the heat equation [11]. The evolution of the nonlinear Fourier spectrum is, exactly like in the linear case, much simpler than the evolution of the original signal. NFTs also have unique data analysis capabilities that enable the detection of particle-like signal components known as solitons [160].

Recently, a nonlinear variant of the FFT has been derived [128], [161]. These type of fast NFTs (FNFTs) can provide up to second-order accuracy [35]. Unfortunately, unlike for the FFT, the accuracy of the FNFTs in [35], [128], [161] remains (at most) second-order even when the signal is smooth. As a result, engineers currently have to strongly oversample even smooth signals in order to get reliable numerical results [96, Section 4]. Several authors have proposed NFT algorithms with higher orders of accuracy, utilizing either Runge-Kutta [69], [119] or implicit Adams methods [130]. However, even though these methods have higher accuracy orders, they require very small sampling intervals in order to actually perform better than standard second-order method such as [116]. For practically relevant sampling intervals, they are typically not the best choice as they are slower *and* may even perform worse in these regimes. Numerical methods that provide better complexity-accuracy trade-offs in practically relevant regimes have been an open problem until recently.

In [121], the authors introduced a new numerical method that can compute the NFT with accuracies that are orders of magnitudes better than those of standard methods while having comparable run times. The key enabler for this large improvement in the complexity-accuracy trade-off was that, for the first time, a so-called commutator-free exponential integrator [162] of higher order was used to compute the NFT. In a nutshell, the absence of commutator terms drastically reduces the computational cost whereas the excellent performance of exponential integrators is retained. However there is one drawback remaining in [121]: The complexity of the algorithm grows quadratically with the number of signal samples D , which makes the algorithm attractive only if the number of samples is not too high. In other words the algorithm is not fast. In this chapter we overcome this limitation. Our main contribution is the first fast higher-order NFT algorithm based on an exponential integrator. By combining it with Richardson extrapolation scheme, we arrive at an NFT algorithm that requires only $\mathcal{O}(D \log^2 D)$ floating point operations, but achieves a sixth-order $[\mathcal{O}(D^{-6})]$ error decay.¹ To the best of our knowledge no such algorithm has been investigated in the literature before. We show

¹The complexity estimate only contains the cost of computing the so-called continuous spectrum as is usual in the NFT literature. Details on the continuous spectrum will be given later in the text. The cost of computing the discrete spectrum are highly problem dependent.

that the complexity-accuracy trade-off of the proposed algorithm is dramatically better than that of existing standard methods. To give an illustration, we point out that in one of our numerical examples, our new method achieves an accuracy that is *hundred million* times better than the standard second-order method in [116] at a comparable run time.²

The rest of this chapter is structured as follows. In Section 2.2 we recapitulate the required mathematical background of the NFT. In Section 2.3 we derive improved versions of our recently proposed numerical NFT in [121], and compare them with both conventional second-order and other higher-order NFT algorithms in multiple numerical examples. Then, in Section 2.4, we demonstrate how some of our new numerical NFTs can be made fast. The fast versions are compared to their slow counterparts. Next, in Section 2.5, we investigate how Richardson extrapolation can improve the complexity-accuracy trade-off of the fast NFT methods even further. The chapter is finally concluded in Section 2.6.³

NOTATION

Real numbers: \mathbb{R} ; $\mathbb{R}_{\geq 0} := \{x \in \mathbb{R} : x \geq 0\}$; Complex numbers: \mathbb{C} ; Complex numbers with positive imaginary part: \mathbb{H} ; Integers: \mathbb{Z} ; $i := \sqrt{-1}$; Euler's number: e ; Real part: $\text{Re}(\cdot)$; Complex conjugate: $(\cdot)^*$; Floor function: $\lfloor \cdot \rfloor$; Absolute value: $|\cdot|$; Matrix exponential: $\text{expm}(\cdot)$; Matrix product: $\prod_{k=1}^K \mathbf{A}_k := \mathbf{A}_K \mathbf{A}_{K-1} \cdots \mathbf{A}_1$; Matrix element in the i th column and j th row: $[\cdot]_{i,j}$; Fourier transform of the function $f(t)$, $\mathcal{F}(f(t)) = \tilde{f}(\xi) = \int_{-\infty}^{\infty} f(t) e^{-it\xi} dt$; Inverse Fourier transform of the function $\tilde{f}(\xi)$, $\mathcal{F}^{-1}(\tilde{f}(\xi)) = f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{f}(\xi) e^{it\xi} d\xi$.

2.2. PRELIMINARIES

In this section we describe the mathematical machinery behind the nonlinear Fourier transform (NFT). For illustration purposes we will describe the NFT in the context of fiber-optic communications. Let $q(x, t)$ denote the complex envelope of the electric field in an ideal optical fiber, whose evolution in normalized coordinates is described by the nonlinear Schrödinger equation (NSE) [21, Chap. 2]

$$i \frac{\partial q}{\partial x} + \frac{\partial^2 q}{\partial t^2} + 2\kappa |q|^2 q = 0. \quad (2.1)$$

Here, $x \geq 0$ denotes the location in the fiber and t denotes retarded time. The parameter κ determines if the dispersion in the fiber is normal (-1) or anomalous (+1). When $\kappa = +1$, (2.1) is referred to as the focusing NSE and for $\kappa = -1$ (2.1) is referred to as the defocusing NSE. The NFT that solves the NSE (2.1) is due to Zakharov and Shabat [10]. It transforms any signal $q(t)$ that vanishes sufficiently fast for $t \rightarrow \pm\infty$ from the time-domain to the nonlinear Fourier domain through the analysis of the linear ordinary differential equation (ODE)

$$\frac{\partial \mathbf{V}(t, \lambda)}{\partial t} = \mathbf{C}(t, \lambda) \mathbf{V}(t, \lambda) = \begin{bmatrix} -i\lambda & q(t) \\ -\kappa q^*(t) & i\lambda \end{bmatrix} \mathbf{V}(t, \lambda), \quad (2.2)$$

²Compare the error for $\text{CF}_1^{[2]}$ in Figure 2.6 with that of $\text{FCF_RE}_2^{[4]}$ in Figure 2.13 for the execution time 1 sec. We remark that although the execution times are implementation specific, they still give a good indication of the advantages of our proposed algorithm (see Section 2.A).

³Some of the results were presented at the OSA Advanced Photonics Congress, Zurich, July 2018 (SpM4G.5)

where $q(t) = q(x_0, t)$ for any fixed x_0 , subject to the boundary conditions

$$\begin{aligned} \mathbf{V}(t, \lambda) &= [\boldsymbol{\phi}(t, \lambda) \ \bar{\boldsymbol{\phi}}(t, \lambda)] \rightarrow \begin{bmatrix} e^{-i\lambda t} & 0 \\ 0 & -e^{i\lambda t} \end{bmatrix} \text{ as } t \rightarrow -\infty, \\ \mathbf{V}(t, \lambda) &= [\tilde{\boldsymbol{\psi}}(t, \lambda) \ \boldsymbol{\psi}(t, \lambda)] \rightarrow \begin{bmatrix} e^{-i\lambda t} & 0 \\ 0 & e^{i\lambda t} \end{bmatrix} \text{ as } t \rightarrow \infty. \end{aligned} \quad (2.3)$$

The term $\lambda \in \mathbb{C}$ is a spectral parameter similar to s in the Laplace domain. The matrix $\mathbf{V}(t, \lambda)$ is said to contain the eigenfunctions since (2.2) can be rearranged into an eigenvalue equation with respect to λ [11]. One can view the eigenfunctions $V(t, \lambda)$ as being scattered by $q(t)$ as they move from $t \rightarrow -\infty$ to $t \rightarrow \infty$. Hence (2.2) is referred to as the scattering problem [10]. (Many problems in signal processing can be expressed through such scattering problems [133].) For (2.2) subject to boundary conditions (2.3), there exists a unique matrix

$$\mathcal{S}(\lambda) = \begin{bmatrix} a(\lambda) & \bar{b}(\lambda) \\ b(\lambda) & -\bar{a}(\lambda) \end{bmatrix}, \quad (2.4)$$

called the scattering matrix, such that [11]

$$[\boldsymbol{\phi}(t, \lambda) \ \bar{\boldsymbol{\phi}}(t, \lambda)] = [\tilde{\boldsymbol{\psi}}(t, \lambda) \ \boldsymbol{\psi}(t, \lambda)] \mathcal{S}(\lambda). \quad (2.5)$$

The components $a(\lambda)$, $b(\lambda)$, $\bar{b}(\lambda)$ and $\bar{a}(\lambda)$ are known as the scattering data. The components $a(\lambda)$ and $b(\lambda)$ are sufficient to describe the signal completely. Their evolution along the x dimension (along the length of the fiber) is simple [11, Section III]

$$\begin{aligned} a(x, \lambda) &= a(0, \lambda), \\ b(x, \lambda) &= b(0, \lambda)e^{4i\lambda^2 x}. \end{aligned} \quad (2.6)$$

The *reflection coefficient* is then defined as $\rho(\lambda) = b(\lambda)/a(\lambda)$ for $\lambda \in \mathbb{R}$ and it represents the continuous spectrum. In the case of $\kappa = 1$, the nonlinear Fourier spectrum can also contain a so-called *discrete spectrum*. It corresponds to the complex poles of the reflection coefficient in the upper half-plane \mathbb{H} , or equivalently to the zeros $\lambda_k \in \mathbb{H}$ of $a(\lambda)$. It is known that there are only finitely many (N) such poles. The poles λ_k are also referred to as eigenvalues and a corresponding set of values $\rho_k := b(\lambda_k) / \left. \frac{da(\lambda)}{d\lambda} \right|_{\lambda=\lambda_k}$ are known as residues [11, App. 5]. There are different ways to define a nonlinear Fourier spectrum. One possibility is $\{\rho(\lambda)\}_{\lambda \in \mathbb{R}}, (\lambda_k, \rho_k)_{k=1}^N$ [11]. The other is $\{b(\lambda)\}_{\lambda \in \mathbb{R}}, (\lambda_k, b(\lambda_k))_{k=1}^N$ [38]. In this chapter we are primarily interested in computation of $\rho(\lambda)$ but some notes regarding computation of $b(\lambda)$ and the λ_k will also be given. Although we will illustrate our algorithms by applying them to the specific case of NFT of NSE with vanishing boundary condition, it should be noted that we in fact presenting algorithms for solving a class of equations similar to (2.2) [11, Eq. 2]. Hence the algorithms presented in this chapter should carry over to NFTs w.r.t. other nonlinear evolution equations such as the Korteweg–de Vries equation [163] and other boundary conditions.

2.3. NUMERICAL COMPUTATION OF NFT USING HIGHER ORDER EXPONENTIAL INTEGRATORS

In this section we will start by outlining some assumptions that are required for the numerical methods that will be presented. We will give a brief overview of one of the approaches for computing the NFT and then talk specifically about implementations using commutator-free exponential integrators. To evaluate the methods, we describe examples and performance criteria. We will finally show and compare the results for various methods applied to the mentioned examples.

We remark that only one of the investigated commutator-free exponential integrators can later serve as basis for our new fast method. However, the remaining higher order integrators have their own merits when the number of samples is low, since (asymptotically) slow NFT algorithms can be faster than (asymptotically) fast NFT algorithms in that regime.

2.3.1. ASSUMPTIONS

Just like the FFT, the numerical computation of the NFT is carried out with finitely many discrete data samples. Hence, we need to make the following assumptions:

1. The support of the signal $q(t)$ is truncated to a finite interval, $t \in [T_-, T_+]$, instead of $t \in (-\infty, \infty)$. The values T_{\pm} are chosen such that the resulting truncation error is sufficiently small. The approximation is exact if $q(t) = 0 \forall t \notin [T_-, T_+]$.
2. The interval $[T_-, T_+]$ is divided into D subintervals of width $h = (T_+ - T_-)/D$. We assume that the signal is sampled at the midpoints of each subinterval $t_n = T_- + (n + 0.5)h$, $n = 0, 1, \dots, D - 1$ such that $q_n := q(t_n)$.

2.3.2. NUMERICAL SCATTERING

The main step in numerically computing the NFT is to solve the scattering problem (2.2) for $\phi(T_+, \lambda)$ for different values of λ . We can view the D subintervals as layers which scatter the eigenfunction $\phi(t, \lambda)$ as it moves from $t = T_-$ to $t = T_+$. Using numerical ODE solvers we solve for an approximation $\hat{\phi}(T_+, \lambda)$ of $\phi(T_+, \lambda)$. By taking $\tilde{\psi}(T_+, \lambda)$ and $\psi(T_+, \lambda)$ equal to the limit in (2.3) at $t = T_+$, we can compute with (2.5) a numerical approximation of the scattering data and ultimately the reflection coefficient.

2.3.3. EXPONENTIAL INTEGRATORS

Almost any numerical method available in literature for solving ODEs can be used to solve for $\phi(T_+, \lambda)$ [118], [119]. However, we are particularly interested in so-called exponential type integrators. These methods have been shown to provide a very good trade-off between accuracy and computational cost in several numerical benchmark problems while being fairly easy to implement, see [164] and references therein. We propose to use a special sub-class known as commutator-free quasi-Magnus (CFQM) exponential integrators as some NFT algorithms based on these integrators turn out to have the special structure [161] that is needed to make them fast. We show this in Section 2.4.

The results in [165] provide a scheme to compute a numerical approximation $\hat{\phi}(T_+, \lambda)$

of $\boldsymbol{\phi}(T_+, \lambda)$. We start by fixing $\hat{\boldsymbol{\phi}}(T_+, \lambda) = \mathbf{H}(\lambda)\boldsymbol{\phi}(T_-, \lambda)$, where

$$\begin{aligned} \mathbf{H}(\lambda) &= \left(\prod_{n=0}^{D-1} \mathbf{G}_n(\lambda) \right) \\ &= \mathbf{G}_{D-1}(\lambda) \mathbf{G}_{D-2}(\lambda) \cdots \mathbf{G}_0(\lambda), \end{aligned} \quad (2.7)$$

with n being the index of samples of $q(t)$.

The structure of $\mathbf{G}_n(\lambda)$ depends on the integrator and the exact values depend on the signal samples q_n and the value of λ . For the integrator in [165], $\mathbf{G}_n(\lambda) = \text{CF}_J^{[r]}(t_n, \lambda)$ which leads to the following iterative scheme:

$$\begin{aligned} \hat{\boldsymbol{\phi}}_{n+1}(\lambda) &= \text{CF}_J^{[r]}(t_n, \lambda) \boldsymbol{\phi}_n(\lambda) \\ &= \prod_{j=1}^J \text{expm}(\mathbf{B}_j(t_n, \lambda)) \boldsymbol{\phi}(t_n, \lambda) \\ &= \text{expm}(\mathbf{B}_J(t_n, \lambda)) \cdots \text{expm}(\mathbf{B}_1(t_n, \lambda)) \boldsymbol{\phi}_n(\lambda) \\ &= \boldsymbol{\phi}(t_{n+1}, \lambda) + \mathcal{O}(h^{r+1}), \end{aligned} \quad (2.8)$$

where expm is the matrix exponential

$$\begin{aligned} \mathbf{B}_j(t_n, \lambda) &= h \sum_{k=1}^K a_{jk} \mathbf{C}_k(t_n, \lambda), \quad j \in \{1, \dots, J\}, \\ \mathbf{C}_k(t_n, \lambda) &= \mathbf{C}(t_n + (c_k - 0.5)h, \lambda), \end{aligned} \quad (2.9)$$

where a_{jk} and $c_k \in [0, 1]$ for $k \in \{1, \dots, K\}$ are constants that are specific to the integrator and $\mathbf{C}(t, \lambda)$ as in (2.2). By iterating with (2.7) from $n = 0, 1, \dots, D - 1$, we obtain the numerical approximation of $\boldsymbol{\phi}(T_+, \lambda)$ that we need to compute the NFT.

For an integrator $\text{CF}_J^{[r]}$, r is the order and J is the number of matrix exponentials required for each subinterval. K is the number of points within each subinterval where the signal value needs to be known. We refer the reader to [165] for their derivation.

An integrator of order r has a local error (error in each subinterval) of $\mathcal{O}(h^{r+1})$. Over D ($\propto 1/h$) such subintervals i.e., over the interval $[T_-, T_+]$, the global error will be $\mathcal{O}(h^r)$. This distinction of local and global error will become important when we define the error metric used to compare the various algorithms in Section 2.3.4.

The integrator $\text{CF}_1^{[2]}$ is also sometimes referred to as the exponential midpoint rule. It was used in the context of NFT for the defocusing NSE ($\kappa = -1$) by Yamada and Sakuda [136] and later by Boffetta and Osborne [116]. For $\text{CF}_1^{[2]}$, (2.8) reduces to

$$\begin{aligned} \hat{\boldsymbol{\phi}}_{n+1}(\lambda) &= \mathbf{G}_n(\lambda) \boldsymbol{\phi}_n(\lambda), \quad \text{where} \\ \mathbf{G}_n(\lambda) &= \text{expm}(h\mathbf{C}_n(\lambda)), \\ \mathbf{C}_n(\lambda) &= \begin{bmatrix} -i\lambda & q_n \\ -\kappa q_n^* & i\lambda \end{bmatrix}. \end{aligned} \quad (2.10)$$

This is applied repeatedly as in (2.7) to obtain $\hat{\boldsymbol{\phi}}(T_+, \lambda)$. In [121] we investigated the possibility of using $\text{CF}_2^{[4]}$ (first introduced in [162]) to obtain $\hat{\boldsymbol{\phi}}(T_+, \lambda)$. We were able to show

its advantage over $\text{CF}_1^{[2]}$ when considering the trade-off between an error and execution time. Here we investigate further in this direction and evaluate $\text{CF}_3^{[4]}$, $\text{CF}_3^{[5]}$ and $\text{CF}_4^{[6]}$, which are fourth-, fifth- and sixth-order methods respectively.

The CFQM exponential integrators require multiple non-equispaced points within each subinterval. However, it is unrealistic to assume that signal samples at such non-equispaced points can be obtained in a practical setting. In [121] we used local cubic-spline based interpolation to obtain the non-equispaced points from the mid-points of each subinterval. (We will refer to the samples at these midpoints as the given samples.) However we found that local cubic-spline based interpolation is not accurate enough for higher-order methods. Here, we propose to utilize the Fourier transform and its time-shift property for interpolation, i.e.,

$$q(t - t_s) = \mathcal{F}^{-1} \left(e^{-i\xi t_s} \mathcal{F} (q(t)) \right), \quad (2.11)$$

to obtain the samples on shifted time grids required for (2.9) with $t_s = -(c_k - 0.5)h$. This interpolation rule is also known in signal-processing literature as sinc or bandlimited interpolation [166, Section 7.4.2] and it is accurate when $q(t)$ is sampled in accordance with the Nyquist criterion. As we are working with discrete signal samples, the interpolation can be implemented efficiently using the FFT. The MATLAB code that we used can be found in Section 2.B. We remark that we use band-limited interpolation for all methods that require non-equispaced samples: $\text{CF}_2^{[4]}$, $\text{CF}_3^{[4]}$, $\text{CF}_3^{[5]}$ and $\text{CF}_4^{[6]}$.

2.3.4. ERROR METRIC AND NUMERICAL EXAMPLES

In this subsection, we compare the performance of CFQM exponential integrators $\text{CF}_1^{[2]}$, $\text{CF}_2^{[4]}$, $\text{CF}_3^{[4]}$, $\text{CF}_3^{[5]}$ and $\text{CF}_4^{[6]}$, the two-step Implicit-Adams method (IA₂) introduced in [130] and the fourth-order Runge-Kutta method [119] (RK₄) for computation of the reflection coefficient. The fourth-order Runge-Kutta method ($r = 4$) was the first fourth-order method used for the computation of the reflection coefficient in [69], [119]. We include the third-order two-step Implicit-Adams method ($r = 3$) here as it was the first higher-order method that was introduced in the context of fast nonlinear Fourier transform. The meaning of “fast” will be made precise in Section 2.4.

We are interested in evaluating the trade-off between the increased accuracy and execution time due to use of higher-order methods. We assess the accuracy of different methods using the relative L^2 -error

$$E_\rho = \frac{\sqrt{\sum_{n=0}^{M-1} |\rho(\lambda_n) - \hat{\rho}(\lambda_n)|^2}}{\sqrt{\sum_{n=0}^{M-1} |\rho(\lambda_n)|^2}}, \quad (2.12)$$

where $\rho(\lambda)$ is the analytical reflection coefficient, $\hat{\rho}(\lambda)$ is the numerically computed reflection coefficient and λ_n are M equally-spaced points in $[-\lambda_{\max}, \lambda_{\max}]$. E_ρ is a global error and hence it is expected to be $\mathcal{O}(h^r)$ for an integrator of order r as explained in Section 2.3.3. We compute the reflection coefficient at the same number of points M as the number of given samples D , i.e. $M = D$, unless mentioned explicitly otherwise.

EXAMPLE 1: HYPERBOLIC SECANT, $\kappa = 1$

As the first numerical example we chose the signal $q(t) = \dot{q}e^{-2i\lambda_0 t} \operatorname{sech}(t)$. The closed form of the reflection coefficient is given by applying the frequency-shift property [68, Section D] to the analytical known reflection coefficient of the secant-hyperbolic signal [113],

$$\begin{aligned} \rho(\lambda) &= \frac{b(\lambda)}{a(\lambda)}, \\ b(\lambda) &= \frac{-\sin(\pi)}{\cosh(\pi(\lambda - \lambda_0))}, \\ a(\lambda) &= \frac{\Gamma^2(0.5 - i\lambda)}{\Gamma(0.5 - i(\lambda - \lambda_0 + \dot{q}))\Gamma(0.5 - i(\lambda - \lambda_0) - \dot{q})}, \end{aligned} \tag{2.13}$$

where $\Gamma(\cdot)$ is the gamma function. The discrete spectrum is

$$\lambda_k = \lambda_0 + i(\dot{q} + 0.5 - k), \quad k = 1, 2, \dots, M_D, \tag{2.14}$$

$$b_k = (-1)^k, \quad k = 1, 2, \dots, M_D, \tag{2.15}$$

$$M_D = \lfloor (\dot{q} + 0.5) \rfloor. \tag{2.16}$$

We set $\dot{q} = 5.4$, $\lambda_0 = 3$, $\lambda_{\max} = 10$, and chose $[T_-, T_+] = [-32, 32]$ to ensure negligible truncation error.

EXAMPLE 2: RATIONAL REFLECTION COEFFICIENT WITH ONE POLE, $\kappa = 1$

The signal is given by [114]

$$q(t) = \begin{cases} -2i\gamma \frac{\alpha}{|\alpha|} \operatorname{sech}\left(2\gamma t + \operatorname{arctanh}\left(\frac{\beta}{\gamma}\right)\right), & t \leq 0 \\ 0, & t > 0, \end{cases} \tag{2.17}$$

where α , β are scalar parameters and $\gamma = \sqrt{\alpha\alpha^* + \beta^2}$. We used $\alpha = 1$ and $\beta = -1$. The corresponding reflection coefficient is then known to be

$$\rho(\lambda) = \frac{\alpha}{\lambda - i\beta}. \tag{2.18}$$

We set $\lambda_{\max} = 60$ and chose $[T_-, T_+] = [-12, 0]$. As the signal in (2.17) has a discontinuity, it cannot be interpolated well using bandlimited interpolation. We hence assume only in this example that we can sample the signal at exactly the points that we require.

EXAMPLE 3: HYPERBOLIC SECANT, $\kappa = -1$

The signal is given by

$$q(t) = \frac{\mathcal{Q}}{L} \left(\operatorname{sech}\left(\frac{t}{L}\right) \right)^{1-2i\mathcal{G}}, \tag{2.19}$$

where \mathcal{G} , \mathcal{L} and \mathcal{Q} are scalar parameters. We used $\mathcal{G} = 1.5$, $\mathcal{L} = 0.04$ and $\mathcal{Q} = 5.5$. The corresponding reflection coefficient is known to be [167]

$$\rho(\lambda) = -2^{-2i\mathcal{G}} \mathcal{Q} \frac{\Gamma(d)\Gamma(f_-)\Gamma(f_+)}{\Gamma(d^*)\Gamma(g_-)\Gamma(g_+)}, \tag{2.20}$$

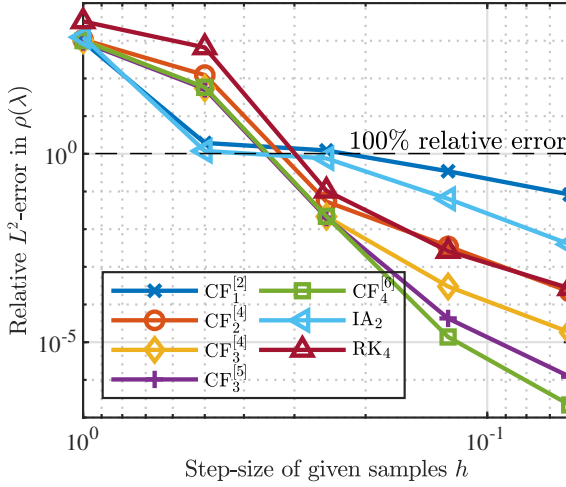


Figure 2.1: Error using slow NFT algorithms for Example 1 with $\lambda_{\max} = 1$.

where $\Gamma(\cdot)$ is the gamma function, $d = 0.5 + i(\lambda\mathcal{L} - \mathcal{G})$, $f_{\pm} = 0.5 - i(\lambda\mathcal{L} \pm \sqrt{\mathcal{G}^2 + \mathcal{Q}^2})$, and $g_{\pm} = 1 - i(\mathcal{G} \pm \sqrt{\mathcal{G}^2 + \mathcal{Q}^2})$. We set $\lambda_{\max} = 250$ and chose $[T_-, T_+] = [-1.5, 1.5]$.

The numerical methods were implemented and tested in 64-bit MATLAB (R2018a) running in Ubuntu 16.04 on a machine with an Intel[®] Core[™] i7-5600U CPU with a maximum clock rate of 3200 MHz and 8192 MB of DDR3 memory at 1600 MHz. The CPU was set to the highest available performance setting and the number of computational threads was set to 1 using the `maxNumCompThreads` function of MATLAB. The closed-form expression of a 2×2 matrix exponential as in [168] was used for the CFQM exponential integrators.

As we are interested in studying the complexity-accuracy trade-off of the NFT algorithms, we need a measure of computational complexity. In the literature, either number of floating point operations (FLOPs) or execution times are used as a measure of the computational complexity. Both are not ideal. Although FLOP counting seems more objective, in practice FLOP counts are (just like execution times) implementation specific and it is hard to determine even the number of FLOP counts of basic operations such as square roots. FLOP counts also do not account for typical capabilities of modern processors and neglect critical issues such as memory access. We will present our results in terms of execution times as we believe that they are more realistic than FLOP counts. However, to ensure that our implementations were equally efficient, we carried out an additional FLOP count analysis in Section 2.A. By comparing the FLOP counts with the measured execution times we show there that the measured execution times agree well with the FLOP counts for medium to high number of samples. We also show there that the FLOP counts are not representative of computation costs for low number of samples.

Execution times were recorded using the MATLAB stopwatch function (`tic-toc`). We report the best execution time among three runs to ensure that we minimize the impact

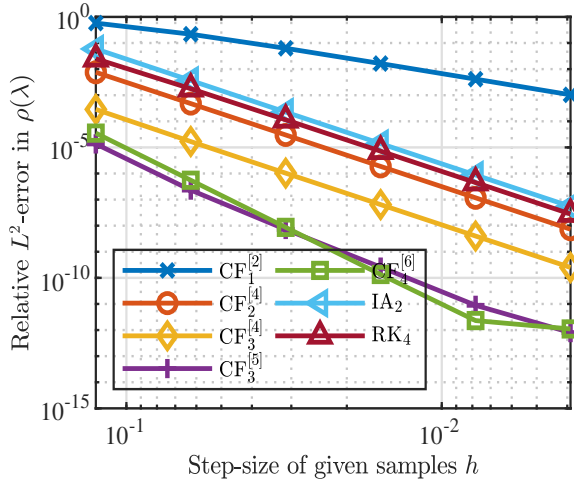


Figure 2.2: Error using slow NFT algorithms for Example 1 with $\lambda_{\max} = 10$.

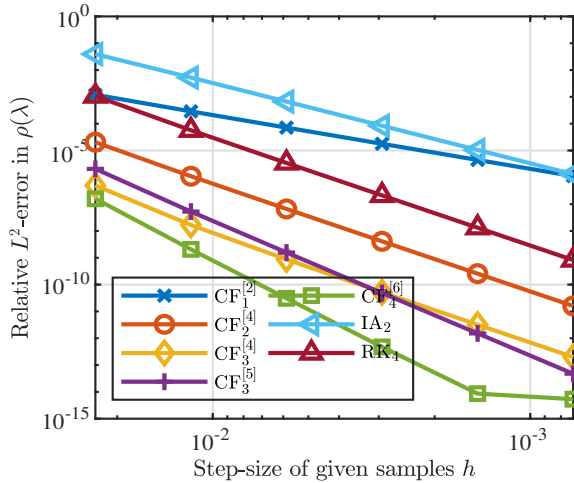


Figure 2.3: Error using slow NFT algorithms for Example 2.

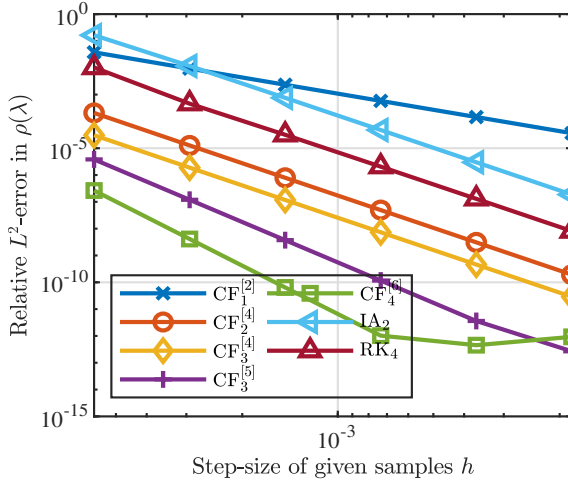


Figure 2.4: Error using slow NFT algorithms for Example 3.

of unrelated background processes.

Figure 2.1 shows the error measure E_ρ , as defined in (2.12), for Example 1 for a range of relatively large step-sizes h . To read such error plots we look at the error achieved by each method for a particular step-size. For the largest two step-sizes, all the errors are above 100 percent and hence a comparison of the methods is not meaningful. The remaining results suggest that the higher-order methods can always be preferred over the lower-order methods.

The error measure E_ρ for smaller sampling intervals h for Example 1, 2, and 3 are shown in Figures 2.2, 2.3 and 2.4 respectively.⁴ For all three examples, the slopes of the error-lines are in agreement with the order r of each method except for IA_2 . For smooth signals IA_2 is seen to have an error of order four rather than the expected three. This observation is in agreement with [130, Figure 2]. However, for the discontinuous signal of Example 2 we see third-order behavior as expected. We can also see that a higher r generally corresponds to better accuracy (lower E_ρ) for the same h . However, that is not necessarily obvious as seen in Figure 2.2, where $CF_3^{[5]}$ is more accurate than $CF_4^{[6]}$ for larger h . The advantage of using three exponentials ($J = 3$) in $CF_3^{[4]}$ instead of two in $CF_2^{[4]}$ is also clear from the figures. The third-order Implicit-Adams method (IA_2 with $r = 3$) and fourth-order Runge-Kutta method (RK_4) may be more accurate than $CF_1^{[2]}$ depending on the signal and other parameters, but have lower accuracy compared to $CF_2^{[4]}$ and $CF_3^{[4]}$.

The error E_ρ reaches a minimum around 10^{-12} and can start rising again as seen in Figure 2.4 for $CF_6^{[4]}$. To understand this behavior, note that the local error in (2.8) is actually $\mathcal{O}(h^{r+1} + \epsilon)$, where ϵ is a small constant due to finite precision effects that can normally be neglected. The global error is thus $\mathcal{O}(h^r + \epsilon h^{-1})$. As h is becoming

⁴To ensure that the discontinuity in Example 2 is faithfully captured, we use $t_n = T_- + nh$ for the Runge-Kutta method and the Implicit-Adams method, instead of the description in Section 2.3.1.

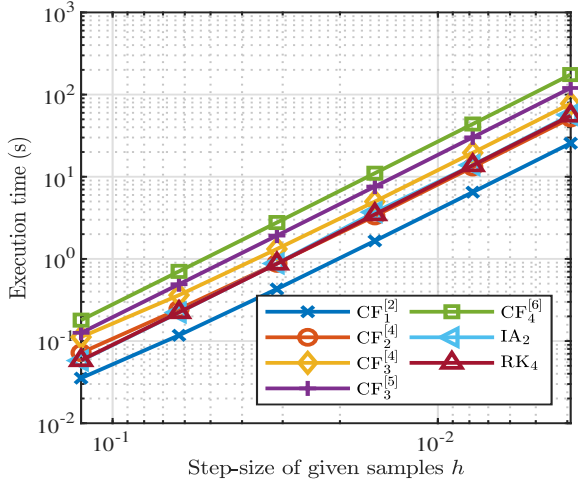


Figure 2.5: Execution time using slow NFT algorithms for Example 1.

smaller and smaller, the second component also known as the arithmetic error, becomes dominant and eventually causes the total error to rise again [169].

For the CFQM exponential integrators, computation of the transfer-matrix $H(\lambda)$ in (2.7) for each λ requires JD multiplications of 2×2 matrices (2.8) for $D(\propto 1/h)$ given samples. If the reflection coefficient is to be computed at D points then the overall computational complexity will be of the order $\mathcal{O}(D^2)$. In Figure 2.5 we plot the execution times of all the methods for Example 1. These execution times are representative for all examples. We can see that the execution time scales quadratically with $1/h$. The execution time of the CFQM exponential integrators is approximately a linear function of J . The IA_2 , RK_4 and $CF_2^{[4]}$ methods have similar execution times. Although both $CF_3^{[4]}$ and $CF_3^{[5]}$ methods require 3 matrix exponentials, the execution times of $CF_3^{[5]}$ are higher because it involves more operations using complex numbers compared to $CF_3^{[4]}$.

To evaluate the trade-off between the execution time and accuracy, we plot the execution time on the x-axis and the error on the y-axis in Figure 2.6 for Example 1. To read such trade-off plots we look at the error achieved by each method for a given amount of time. For Example 1 it turns out that $CF_3^{[5]}$ provides the best trade-off, but we can conclude that extra computation cost of the higher-order methods is generally justified by increased accuracy.

Although performing matrix multiplications of 2×2 matrices is fast, the total cost of the NFT ($\mathcal{O}(D^2)$) is significantly higher when compared to its linear analogue, the FFT, which has a complexity of only $\mathcal{O}(D \log D)$. So the natural question to ask is: Can the complexity be reduced? – Yes, this will be shown in the next section.

2.4. FAST FOURTH-ORDER NFT

In this section we investigate which of the new higher-order NFT algorithms from the previous section can be made fast by using suitable splittings of the matrix exponential.

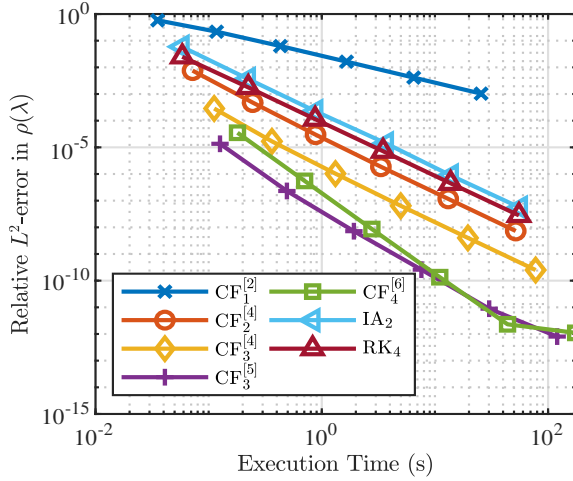


Figure 2.6: Error vs. Execution time trade-off using slow NFT algorithms for Example 1.

We find that only the $\text{CF}_2^{[4]}$ NFT can be made fast. The result is a fast fourth-order NFT algorithm. We then compare the slow $\text{CF}_2^{[4]}$ NFT with its fast variant to ensure that the gain in computational complexity outweighs the loss in accuracy introduced by approximations of the matrix exponential.

2.4.1. FAST SCATTERING FRAMEWORK

In the framework proposed in [161], each matrix $\mathbf{G}_n(\lambda)$ is approximated by a rational function matrix $\hat{\mathbf{G}}_n(z)$, where $z = z(\lambda)$ is a transformed coordinate. By substituting these approximations in (2.7), a rational function approximation $\hat{\mathbf{H}}(z)$ of $\mathbf{H}(\lambda)$ is obtained.

$$\hat{\mathbf{H}}(z) = \prod_{n=0}^{D-1} \hat{\mathbf{G}}_n(z). \quad (2.21)$$

We want to compute the coefficients of the numerator and denominator polynomials, respectively. A straightforward implementation of the matrix multiplication where each entry is a polynomial, has a complexity of $\mathcal{O}(D^2)$. Instead, by using a binary-tree structure and FFTs [161, Alg. 1], the computational complexity can be reduced to $\mathcal{O}(D \log^2 D)$. Hence it is referred to as fast scattering. In [161], the number of samples D was assumed to be a power of two. In cases where D is not a power of two, we use the following approach. We write $D = 2^{D_1} + 2^{D_2} + \dots + 2^{D_m}$, where D_1, D_2, \dots, D_m are non-negative integers. We first choose D_1 as large as possible. Then we choose D_2 as large as possible and repeat until all D_k are fixed. This step splits the D samples into m sets to each of which the fast scattering is applied. The results $\hat{\mathbf{H}}_1(z), \hat{\mathbf{H}}_2(z), \dots, \hat{\mathbf{H}}_m(z)$ are then multiplied using the rule $\hat{\mathbf{H}}(z) = [\dots [[\hat{\mathbf{H}}_1(z)\hat{\mathbf{H}}_2(z)]\hat{\mathbf{H}}_3(z)] \dots]\hat{\mathbf{H}}_m(z)$. Each multiplication is carried out using the same FFT based algorithm as in [161].

The rational function approximation $\hat{\mathbf{H}}(z)$ is explicitly parameterized in z and hence (2.7) is reduced to polynomial evaluations for each z . To elaborate, we again restrict

ourselves to $\mathbf{G}_n(\lambda)$ of the form (2.8). Hence for $\text{CF}_1^{[2]}$, we need to approximate $\mathbf{G}_n(\lambda) = \text{expm}(h\mathbf{C}_n(\lambda))$. The matrix exponential can be approximated to various orders of accuracy using rationals [170] or using splitting-schemes such as the well-known Strang-splitting and the higher-order splitting-schemes developed in [163]. The splitting-schemes map $\lambda \in \mathbb{R}$, the domain of the reflection coefficient, to $z = \exp(i\lambda h/m)$ on the unit circle, where m is a real rational. Such mappings have a certain advantage when it comes to polynomial evaluations which we cover in Section 2.4.2. Note that the mapping $z = \exp(i\lambda h/m)$ is periodic in λ with period $2\pi m/h$. Hence we can resolve the range $|\text{Re}(\lambda)| < \pi m/h$. (See e.g. [135].) This is similar to the Nyquist–Shannon sampling theorem for the FFT.

For a higher-order $\text{CF}_J^{[r]}$ integrator, each $\mathbf{G}_n(\lambda)$ is a product of J matrix exponentials. For example let us look at $\text{CF}_2^{[4]}$. We can write

$$\begin{aligned} \mathbf{G}_n(\lambda) &= \text{expm}(h\mathbf{C}_n^2(\lambda)) \text{expm}(h\mathbf{C}_n^1(\lambda)), \\ \mathbf{C}_n^2(\lambda) &= a_2\mathbf{C}(T_-(n+c_1)h, \lambda) + a_1\mathbf{C}(T_-(n+c_2)h, \lambda), \\ \mathbf{C}_n^1(\lambda) &= a_1\mathbf{C}(T_-(n+c_1)h, \lambda) + a_2\mathbf{C}(T_-(n+c_2)h, \lambda), \end{aligned} \tag{2.22}$$

$$\begin{aligned} a_1 &= \frac{1}{4} + \frac{\sqrt{3}}{6}, & a_2 &= \frac{1}{4} - \frac{\sqrt{3}}{6}, \\ c_1 &= \frac{1}{2} - \frac{\sqrt{3}}{6}, & c_2 &= \frac{1}{2} + \frac{\sqrt{3}}{6}. \end{aligned}$$

Each of the two matrix exponentials can be approximated individually using a splitting-scheme from [163]. $\hat{\mathbf{H}}(z)$ can then be obtained as in (2.21). However, there are a few caveats which prevent extension of this idea to higher-order methods. The splitting-schemes in [163] should not be applied to CFQM exponential integrators with complex coefficients a_{jk} . Complex coefficients mean that $\lambda \in \mathbb{R}$ is no longer mapped to z on the unit circle. Such a mapping is undesirable for polynomial evaluation as will be explained in Section 2.4.2. In addition, we do not even obtain a polynomial structure if there exists no z such that $\exp(i\lambda h \sum_{k=1}^K a_{j,k})$ is an integer power of this z for all j . Furthermore, if such a z exists but only for high co-prime integer powers, $\hat{\mathbf{G}}_n(\lambda)$ will consist of sparse polynomials of high degree, which can significantly hamper the computational advantage of using the approximation. Due to these reasons we restrict ourselves to fast implementations of $\text{CF}_1^{[2]}$ and $\text{CF}_2^{[4]}$ which will be referred to as $\text{FCF}_1^{[2]}$ and $\text{FCF}_2^{[4]}$. Even though we made the $\text{FCF}_1^{[2]}$ algorithm available in the FNFT software library [171] already, accuracy and execution times for it haven't been assessed and published formally anywhere in literature. The $\text{FCF}_2^{[4]}$ algorithm is completely new. For both $\text{FCF}_1^{[2]}$ and $\text{FCF}_2^{[4]}$ we use the fourth-order accurate splitting [163, Eq. 20].

2.4.2. FAST EVALUATION

Once we obtain the rational function approximation $\hat{\mathbf{H}}(z)$ of $\mathbf{H}(\lambda)$ in terms of numerator and denominator coefficients, we only have to evaluate the numerator and denominator polynomials for each value of $z = z(\lambda)$ in order to compute the reflection coefficient. The degree of the polynomials to be evaluated will be at least D which can be in the range of 10^3 – 10^4 . It is known that evaluation of such high-degree polynomials for large values of z

can be numerically problematic [128, Section IV-E]. However, by choosing the mapping $z = \exp(i\lambda h/m)$, which maps the real line to the unit circle, the polynomials need to be evaluated on the unit circle where evaluation of even high-degree polynomials is numerically less problematic. The higher-order splitting schemes in [163] were developed with such a mapping in mind allowing for approximations of the matrix exponentials as rational functions in z . Evaluating any polynomial of degree D using Horner's method has a complexity of $\mathcal{O}(D)$ [128, Section IV-E]. Hence for M values of z , the total cost of fast scattering followed by polynomial evaluation would be $\mathcal{O}(D \log^2 D) + \mathcal{O}(MD)$.

Mapping $\lambda \in \mathbb{R}$ to z on the unit circle has an additional computational advantage. Let $p(z) = p_N z^N + p_{N-1} z^{N-1} + \dots + p_0$ be a polynomial in z of degree N . Evaluation of $p(z)$ at a point z_k can be written as

$$p(z_k) = \sum_{n=0}^N p_n z_k^n = z_k^N \sum_{n=0}^N p_{N-n} z_k^{-n}. \quad (2.23)$$

For M equispaced points z_k , $k = 1, \dots, M$, on a circular arc, this amounts to taking the chirp Z-transform (CZT) of the polynomial coefficients. The CZT can be computed efficiently using the algorithm in [129] which utilizes FFTs. We can also see (2.23) as a non-uniform discrete Fourier transform of the polynomial coefficients which allows us to utilize efficient non-uniform FFT (NFFT) algorithms in [172] for evaluating the polynomial. If the number of evaluation points M is in the same order of magnitude as D , the complexity of evaluation becomes $\mathcal{O}(D \log D)$ and hence the overall complexity of the fast nonlinear Fourier transform (FNFT) is $\mathcal{O}(D \log^2 D)$. In the next section we will see that the error of the FCF₂^[4] algorithm reaches a minimum value and then starts rising. This is again due to the arithmetic error as we already saw in Section 2.3.4. We remark that in numerical tests the CZT was found to perform equally well as the NFFT before the error minimum but the error rise thereafter was significantly steeper. We hence used the NFFT routine from [172] for evaluating the polynomials.

2.4.3. NUMERICAL EXAMPLES

REFLECTION COEFFICIENT

We now compare the implementations of CF₁^[2] and CF₂^[4] presented in Section 2.3.4 and their fast versions FCF₁^[2] and FCF₂^[4] for computing the reflection coefficient $\rho(\lambda)$. We plot the error versus the execution time for Example 1 in Figure 2.7, for Example 2 in Figure 2.8 and for Example 3 in Figure 2.9. In the three figures we can see that the fast FCF algorithms achieve similar errors as their slow CF counterparts in a significantly shorter time. From the other viewpoint, for the same execution time, the FCF algorithms achieve significantly lower errors compared to CF algorithms. Our new algorithm FCF₂^[4] outperforms FCF₁^[2] in the trade-off for all the examples which again highlights the advantage of using higher-order CFQM exponential integrators.

Since the NFT is a nonlinear transform, it changes its form under signal amplification, and computing it typically becomes increasingly difficult when a signal is amplified [118]. Hence it is of interest to study amplification of error with increase in signal amplitude. To test the amplification we use Example 1 and sweep the signal amplitude \hat{q} from 0.4 to 10.4 in steps of 1.0 while keeping all other parameters the same as before. As

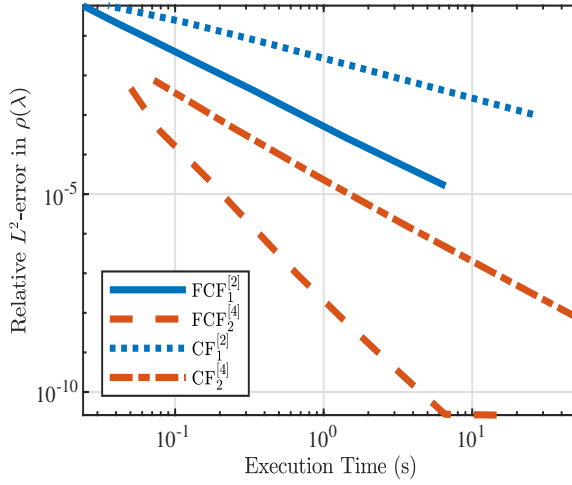


Figure 2.7: Error vs. Execution time trade-off using CF and FCF algorithms for Example 1.

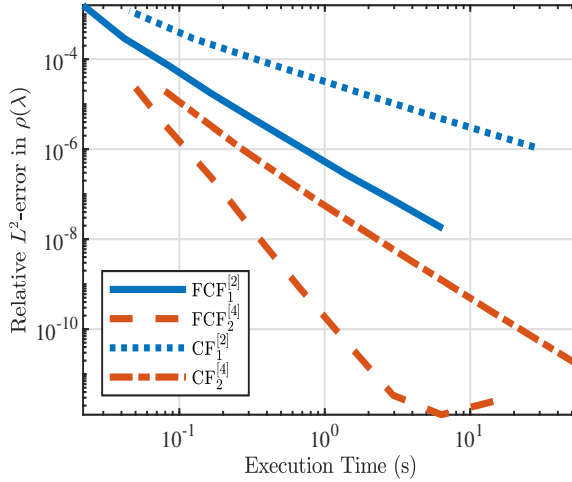


Figure 2.8: Error vs. Execution time trade-off using CF and FCF algorithms for Example 2.

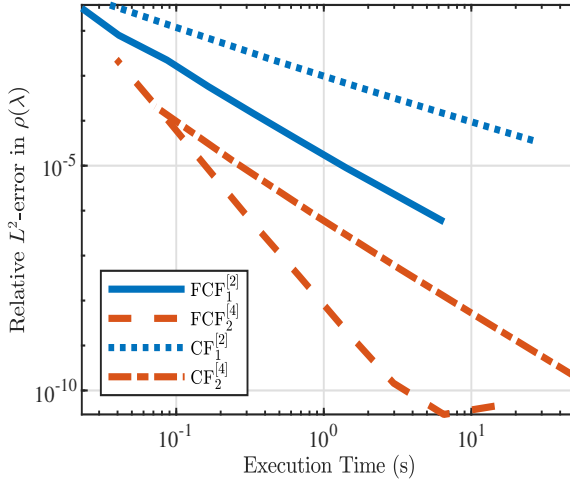


Figure 2.9: Error vs. Execution time trade-off using CF and FCF algorithms for Example 3.

the time-window remains the same, amplification the signal amplitude leads to directly proportional amplification of the signal energy. We compute the error E_ρ for decreasing h for each value of \hat{q} for the CF and FCF algorithms. We plot E_ρ versus the sampling interval h on a log-scale for CF algorithms in Figure 2.10 and for FCF algorithms in Figure 2.11. Instead of plotting individual lines for each value of \hat{q} , we represent the amplitude using different shades of gray. As shown in the colourbar, lighter shades represents lower \hat{q} and darker shades represent higher \hat{q} . The stripes with a higher slope are the higher-order methods. All the four algorithms i.e., $CF_1^{[2]}$, $CF_2^{[4]}$, $FCF_1^{[2]}$ and $FCF_2^{[4]}$ show similar trends for the amplification of error with signal amplitude. The $CF_1^{[2]}$ algorithm was compared with other methods in [118] (where it is referred to as BO), and they conclude that $CF_1^{[2]}$ scales the best with increasing signal amplitude. Hence the results shown in Figure 2.10 are very motivating as the amplification in the error of $CF_2^{[4]}$ is similar to the amplification for $CF_1^{[2]}$. The error of approximations used in the FCF algorithms also depends on \hat{q} . However, comparing Figure 2.10 and Figure 2.11 we can see that the contribution of the approximation error is small. These results combined with the results in the trade-off plots (Figures 2.7, 2.8, and 2.9) make a strong case for our new $FCF_2^{[4]}$ algorithm.

B-COEFFICIENT

The accurate and fast computation of the scattering coefficient $b(\lambda)$ (Section 2.2) is of interest to the fiber-optic communication community, as an efficient FNFT algorithm can be combined with the recently proposed b -modulation [90], [92], [173] scheme to develop a complete NFT based fiber-optic communication system. Hence to test the

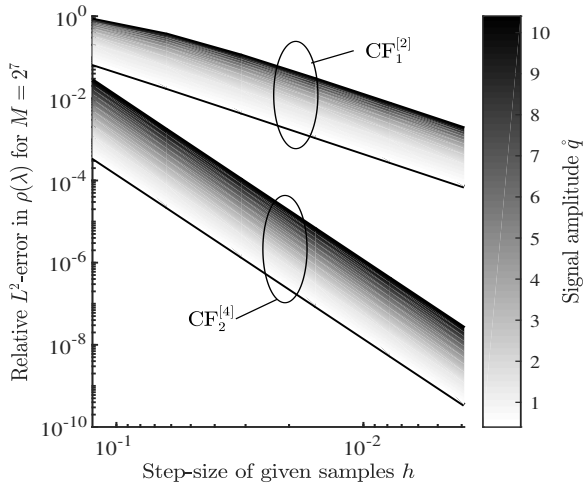


Figure 2.10: Variation of error of CF algorithms with amplitude for Example 1. The fourth-order $CF_2^{[4]}$ algorithm is seen to have gradual increase in error with increase in amplitude similar to the second-order $CF_1^{[2]}$ algorithm.

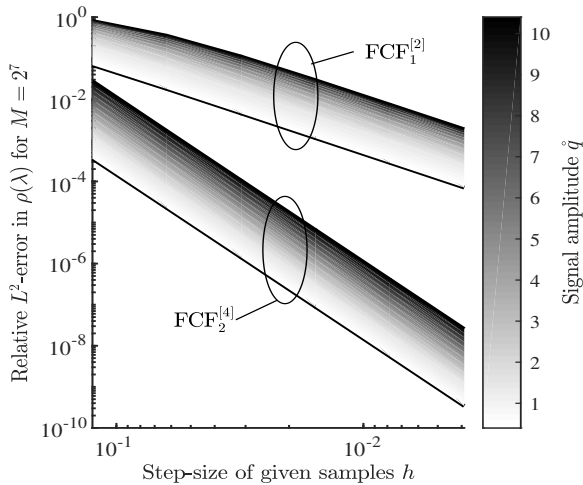


Figure 2.11: Variation of error of FCF algorithms with amplitude for Example 1. Approximating the matrix exponentials with splitting schemes does not significantly affect the amplification of error with increasing amplitude.

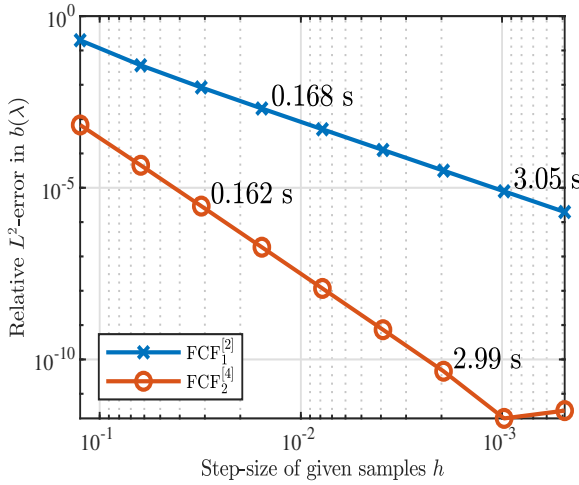


Figure 2.12: Error in b -coefficient using FCF algorithms for Example 1. The execution times for some of the points are shown to give an indication of the computational complexity.

performance of both the FCF algorithms in computation of the b -coefficient, we define

$$E_b = \frac{\sqrt{\sum_{n=0}^{M-1} |b(\lambda_n) - \hat{b}(\lambda_n)|^2}}{\sqrt{\sum_{n=0}^{M-1} |b(\lambda_n)|^2}}, \quad (2.24)$$

where $b(\lambda)$ is the analytically known and $\hat{b}(\lambda)$ is the numerically computed scattering coefficient. For the numerical test we again use the signal from Example 1 as $b(\lambda)$ is known. We plot the error E_b for both the FCF methods for decreasing sampling interval h in Figure 2.12. $\text{FCF}_2^{[4]}$ clearly outperforms $\text{FCF}_1^{[2]}$ even after considering the additional computational cost.

From the results of the numerical tests presented in this section it is clear that approximating $\mathbf{H}(\lambda)$ in (2.7) using rational functions to make the method fast, provides a significant computational advantage: similar accuracy, shorter execution time. However, as mentioned earlier we could only make the fourth-order method $\text{CF}_2^{[4]}$ fast. To further improve the accuracy and order of convergence while restricting ourselves to a fourth-order method, we explore the possibility of using Richardson extrapolation in the next section.

2.5. MAIN RESULT: FAST SIXTH-ORDER NFT

In this section we arrive at our main result by integrating Richardson extrapolation into our new fast fourth-order NFT $\text{FCF}_2^{[4]}$ from the previous section. We show numerically that the resulting algorithm has *sixth-order* accuracy rather than fifth-order as would be expected. We furthermore show that the added complexity due to Richardson extrapolation is outweighed by the gain in accuracy so the complexity-accuracy trade-off of our final algorithm is the best among all methods investigated in this chapter.

2.5.1. RICHARDSON EXTRAPOLATION

Richardson extrapolation is a technique for improving the rate of convergence of a series [174].⁵ Given an r^{th} -order numerical approximation method $\hat{\rho}(\lambda, h)$ for the reflection coefficient $\rho(\lambda)$ that depends on the step-size h , we can write

$$\rho(\lambda) = \hat{\rho}(\lambda, h) + \mathcal{O}(h^r). \quad (2.25)$$

We assume that $\hat{\rho}(\lambda, h)$ has a series expansion in h ,

$$\hat{\rho}(\lambda, h) = \rho(\lambda) + \rho_r(\lambda)h^r + \rho_{r+1}(\lambda)h^{r+1} + \dots \quad (2.26)$$

In Richardson extrapolation [174], we combine two numerical approximations $\hat{\rho}(\lambda, h)$ and $\hat{\rho}(\lambda, 2h)$ as follows,

$$\hat{\rho}^{[\text{RE}]}(\lambda, h) = \frac{2^r \hat{\rho}(\lambda, h) - \hat{\rho}(\lambda, 2h)}{2^r - 1}. \quad (2.27)$$

Using the series expansion, we find that the order of the new approximation $\hat{\rho}^{[\text{RE}]}(\lambda, h)$ is at least $r + 1$:

$$\begin{aligned} \hat{\rho}^{[\text{RE}]}(\lambda, h) &= \frac{2^r(\rho(\lambda) + \rho_r(\lambda)h^r + \rho_{r+1}(\lambda)h^{r+1} + \dots)}{2^r - 1} \\ &\quad - \frac{\rho(\lambda) + \rho_r(\lambda)(2h)^r + \rho_{r+1}(\lambda)(2h)^{r+1} + \dots}{2^r - 1} \\ &= \rho(\lambda) - \frac{2^r}{2^r - 1} \rho_{r+1}(\lambda)h^{r+1} + \mathcal{O}(h^{r+2}). \end{aligned} \quad (2.28)$$

We apply this idea to $\text{FCF}_1^{[2]}$ and $\text{FCF}_2^{[4]}$ to obtain the algorithms $\text{FCF_RE}_1^{[2]}$ and $\text{FCF_RE}_2^{[4]}$ respectively. Note that the range of $|\text{Re}(\lambda)|$ that can be resolved is determined by the larger of the two step-sizes h (see Section 2.4.1). We also remark that Richardson extrapolation can also be applied to the slow algorithms in Section 2.3.3.

2.5.2. NUMERICAL EXAMPLES

We test $\text{FCF_RE}_1^{[2]}$ and $\text{FCF_RE}_2^{[4]}$ against $\text{FCF}_1^{[2]}$ and $\text{FCF}_2^{[4]}$ for all three examples. Since Richardson extrapolation requires us to compute two approximations, which increases the computational complexity, we again evaluate the complexity-accuracy trade-off. We plot the error versus execution time curves for the three examples in the Figures 2.13 to 2.15. In all figures we can see that the FCF_RE algorithms achieve slopes of $r + 2$ rather than the expected slope of $r + 1$. This is an example of superconvergence [175]. Specifically, the error of $\text{FCF_RE}_1^{[2]}$ decreases with slope four and that of $\text{FCF_RE}_2^{[4]}$ decreases with slope six. As seen before in Section 2.3.4, the arithmetic error starts to dominate after a certain point and causes the error to rise. Although the executions times of FCF_RE algorithms are higher for the same step-size h , the error achieved is almost an order of magnitude lower even for large h . From the other viewpoint, for the same execution time, the FCF_RE algorithms achieve significantly lower errors compared to FCF algorithms. $\text{FCF_RE}_2^{[4]}$ outperforms $\text{FCF_RE}_1^{[2]}$ in the trade-off for all the three examples again highlighting the advantage of using higher-order CFQM exponential integrators. These results suggest that Richardson extrapolation should be applied to improve

⁵It was used to improve an inverse NFT algorithm for the defocusing case in [134].

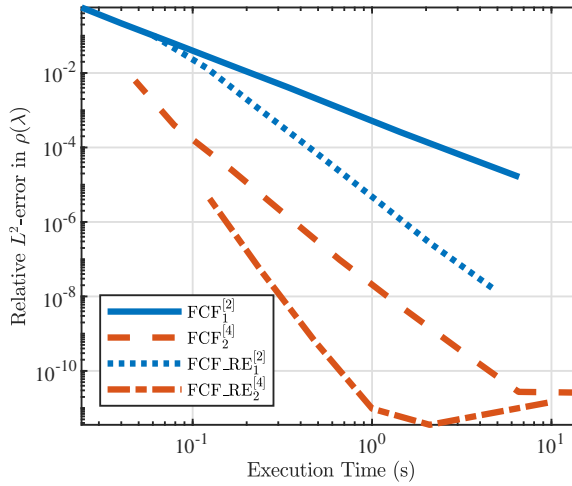


Figure 2.13: Error vs. Execution time trade-off of FCF and FCF_RE algorithms for Example 1.

the considered FNFT algorithms. The $\text{FCF_RE}_2^{[4]}$ algorithm provides the best trade-off among all the algorithms presented in this chapter.

2.5.3. REMARKS ON COMPUTING EIGENVALUES

The main focus of this chapter has been the efficient computation of the reflection coefficient. The computation of the discrete spectrum (see Section 2.2) is more involved and problem specific. The best approach strongly depends on the available a priori knowledge on the number and location of the eigenvalues. In scenarios where little a priori knowledge is available, some of the ideas presented for the reflection coefficient can be applied to the discrete spectrum as well. Some possible approaches are discussed in Section 2.C.

2.6. CONCLUSION

In this chapter, we proposed new higher-order nonlinear Fourier transform algorithms based on a special class of exponential integrators. We also showed that one of these algorithms can be made fast using special higher-order exponential splittings. The accuracy of the fast algorithm was improved even further, to sixth-order, using Richardson extrapolation. To the best of our knowledge this is the first fast sixth-order NFT algorithm ever presented in the literature. Numerical demonstrations showed that the proposed algorithm is highly accurate and provides much better complexity-accuracy trade-offs than existing algorithms. We finally remark that the development of a fast higher-order inverse NFT is an interesting open topic for future research.

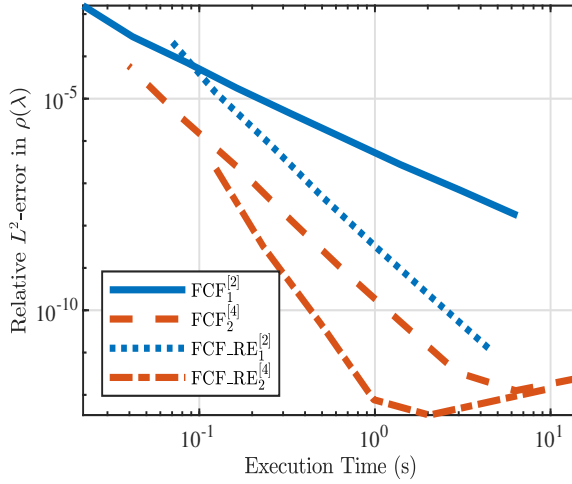


Figure 2.14: Error vs. Execution time trade-off of FCF and FCF_RE algorithms for Example 2.

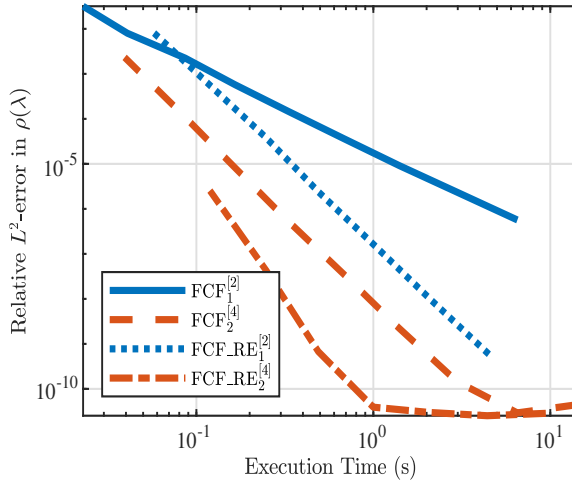


Figure 2.15: Error vs. Execution time trade-off of FCF and FCF_RE algorithms for Example 3.

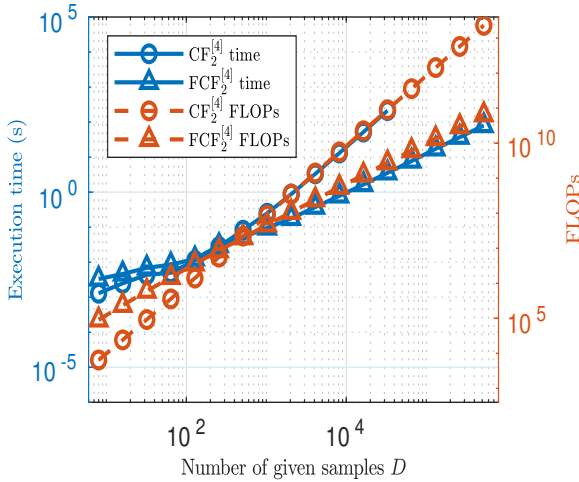


Figure 2.16: The number of FLOPs and measured execution times of a slow fourth-order algorithm and its fast variant.

2.A. COMPARISON OF FLOP COUNTS AND EXECUTION TIMES

In this section we show a comparison between the number of floating-point operations (FLOPs) and execution times of two algorithms proposed in this chapter. We counted all the operations of the slow algorithm and the fast algorithm based on the $CF_2^{[4]}$ integrator. We list the different operations and how often they occur in Table 2.1. For the FFT and CZT, the size of the input is also specified. The number of FLOPs required for each operation type are provided in Table 2.2. Note that these values are only rules of thumb and vary widely across programming languages and CPU architectures. The number of FLOPs required for the fast scattering step (see Section 2.4.1) is given by

$$\begin{aligned}
 & \text{FLOPs(Fast scattering of size } N) \\
 &= \sum_{k=0}^{\lceil \log_2 N \rceil} 2^{\lceil \log_2 N \rceil - k} \left(12 \text{ FLOPs(FFT of size } (2^{k+1} + 1)) \right. \\
 & \quad \left. + (8 \text{ FLOPs(Mult)} + 4 \text{ FLOPs(Add)})(2^{k+1} + 1) \right). \tag{2.29}
 \end{aligned}$$

In Figure 2.16, we plot the total number of FLOPs and the execution times from our MATLAB implementations against the number of given samples D . At medium to high number of samples we see that the MATLAB execution times match the number of FLOPs very well. Moreover the crossover point at which the fast algorithm becomes faster than the slow variant ($D > 300$ in Figure 2.16) is almost the same. At lower number of samples, the execution times deviate from the number of FLOPs. This is due to the unaccounted overheads dominating over the floating-point operations.

2.B. INTERPOLATION BASED ON FOURIER TRANSFORM

Table 2.1: Number of operations per type

Operation	Algorithm	
	$CF_2^{[4]}$	$FCF_2^{[4]}$
FFT (no, sz)	$3, D$	$3, D$
Multiplication	$14D + 9M + 18DM$	$86D + 4(M + 1)$
Addition	$4(D + 1) + 10DM$	$34D + 6$
Division	$2DM$	$24D + M$
Conjugation	$2D$	$2D$
Square-root	$2DM$	$2D$
sinh	$2DM$	-
cosh	$2DM$	-
cos	-	$4D$
sinc	-	$4D$
Exponential	$2M$	$2M$
CZT (no, sz)	-	$2, 4D + 1$
Fast scattering (no,sz)	-	$1, 2D$

The abbreviations no and sz are short for number and size respectively. All operations are assumed to be on complex numbers. The number of signal samples (D) is assumed to be greater than the number of reflection coefficient samples (M) being computed, i.e. $D \geq M$.

```
function qs = bandlimited_interp(tn,qn,ts)
% Inputs
% tn - Sorted vector of equispaced points at
%       which qn samples are known
% qn - Vector of signal samples at
%       points tn
% ts - Value by which samples are
%       to be shifted
% Output
% qs - Vector of interpolated signal
%       samples at tn-ts
ep = tn(2)-tn(1);
Qn = fft(qn);
N = length(qn);
Np = floor(N/2);
Nn = -floor((N-1)/2);
Qn = Qn.*exp(2i*pi*[0:Np,Nn:-1]*ts/(N*ep));
qs = ifft(Qn);
end
```


Table 2.2: Number of FLOPs for various operations

Operation	Number of FLOPs
FFT of size N	$5N \log_2 N$
Multiplication	1
Addition	1
Division	4
Conjugation	1
Square-root	4
sinh or cosh	8
sin or cos	8
sinc	12
Exponential	8
CZT of size N	$3(5(2N-1) \log_2(2N-1))$
Fast scattering of size N	See (2.29)

The number of FLOPs for the basic operations have been taken from [176, p. 5]. The number of FLOPs for a sinc are the sum of number of FLOPs for a sin and a division. The number of FLOPs for a FFT are based on the asymptotic number of operations for the radix-2 Cooley-Tukey algorithm [177, p. 3]. The number of FLOPs for a CZT are approximated using three FFTs of size $2N-1$.

2.C. COMPUTING EIGENVALUES

Recall that for the case of focusing NSE ($\kappa = 1$), the nonlinear Fourier spectrum has two parts: a continuous and a discrete part. In this section, we are concerned with the numerical computation of the discrete part. We first mention some of the existing approaches and then show how one of them can be extended to work with the new fast higher-order NFT algorithms. We will also show that Richardson extrapolation can be applied to improve the accuracy at virtually no extra computational cost.

2.C.1. EXISTING APPROACHES

Finding the eigenvalues consists of finding the complex upper half-plane roots of the function $a(\lambda)$. Most of the existing approaches can be classified into four main categories.

1. *Search methods*: Newton's method.
2. *Eigenmethods*: Spectral methods based on the solution of a suitably designed eigenproblem [68].
3. *Gridding methods*: They find λ_k using iterative methods or optimized grid search [68], [119]. Recently a method based on contour integrals was proposed [124].

4. *Hybrid methods*: Any combination of the above. Eigenmethods with rougher sampling can e.g. be used to find initial guesses for a search method [88].

Our proposed method will be a hybrid of a eigenmethod and a search method in the spirit of [88], where an eigenproblem is solved to obtain initial guesses for Newton's method.

2

2.C.2. PROPOSED METHOD

Remember that the discrete spectrum consists of eigenvalues, which are the zeros of $a(\lambda)$ in the complex upper half-plane (\mathbb{H}), and their associated residues. We start with discussing an approximation of $a(\lambda)$ that will be useful for locating the eigenvalues. From (2.3), (2.4) and (2.5) we can write

$$a(\lambda) = \lim_{t \rightarrow \infty} \phi_1(t, \lambda) e^{i\lambda t}. \quad (2.30)$$

Over the finite interval $[T_-, T_+]$ using (2.7) we can see that

$$a(\lambda) \approx H_{1,1}(\lambda) e^{i\lambda T_-} e^{i\lambda T_+}. \quad (2.31)$$

Hence we hope that the zeros of $H_{1,1}(\lambda)$ are approximations of the zeros of $a(\lambda)$ if the signal truncation and discretization errors are small enough. In Section 2.4.1 we explained how $\mathbf{H}(\lambda)$ can be approximated by a rational function in a transformed coordinate z . Hence we can further write

$$a(\lambda) \approx \frac{\hat{a}_{\text{num}}(z)}{\hat{a}_{\text{den}}(z)} e^{i\lambda T_-} e^{i\lambda T_+}, \quad (2.32)$$

where $\hat{a}_{\text{num}}(z)$ and $\hat{a}_{\text{den}}(z)$ are polynomials in $z(\lambda)$. Let $\hat{a}_{\text{num}}(z) = \hat{a}_N z^N + \hat{a}_{N-1} z^{N-1} + \dots + \hat{a}_0$. Thus $\hat{a}_{\text{num}}(z)$ will have N zeros. These zeros or roots of $\hat{a}_{\text{num}}(z)$ can be found using various methods [178]. Of these N zeros, there should be K (typically, $K \ll N$) values that are approximations of zeros of $a(\lambda)$ in \mathbb{H} .

EXAMPLE

We would like to add clarity through a visual representation of the roots. We choose the signal from Example 1 with $D = 2^9$. We plot all the zeros of $\hat{a}_{\text{num}}(z)$ of $\text{FCF}_1^{[2]}$ with 'x' in Figure 2.17. Here $z = \exp(i\lambda h)$. We can then map these zeros back to obtain values of λ . These are plotted with 'x' in Figure 2.18. From the definition of discrete spectrum, we can filter out all the values that are not in \mathbb{H} . Recall that we can resolve the range $|\text{Re}(\lambda)| < \pi/h$. (See Section 2.4.1.) Since we observed that spurious eigenvalues tend to cluster around $|\text{Re}(\lambda)| = \pi/h$, we filter out the corresponding roots of $\hat{a}_{\text{num}}(z)$. More precisely we keep only values of λ for which $|\text{Re}(\lambda)| < 0.9\pi/h$. The filtered roots are plotted in Figures 2.17 and 2.18 with 'o'. For the chosen value of $\hat{q} = 5.4$ the set of eigenvalues is $\Lambda = [3 + 4.9i, 3 + 3.9i, 3 + 2.9i, 3 + 1.9i, 3 + 0.9i]$. From Figure 2.18 we see that the values marked with 'o' are indeed approximations of the values in set Λ . However, there is no guarantee that we will always be able to locate approximations for all values in Λ as that depends on several factors, some of which are the signal magnitude q_0 , signal interval $[T_-, T_+]$, step-size h and values of the eigenvalues themselves.

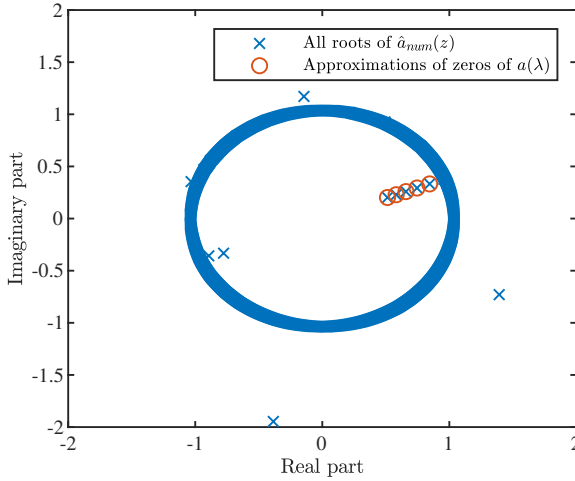


Figure 2.17: Zeros of $\hat{a}_{\text{num}}(z)$ for Example 1 with $D = 2^9$.

For the example chosen in the visual demonstration, the number of zeros is $N = 1024$ and the number of eigenvalues is $K = 5$. For the chosen mapping from $\lambda \rightarrow z$, the K values of interest will always lie inside the unit circle in Figure 2.17 and most other spurious zeros of $\hat{a}_{\text{num}}(z)$ will lie on the unit circle. Even with the best eigenmethods available for polynomial root-finding, which have a complexity of $\mathcal{O}(N^2)$ [179], execution time grows very steeply, making this approach infeasible for large N . To reduce the complexity, it was suggested in [88] to sub-sample the given signal to reduce the dimensionality of the root-finding problem. The algorithm is summarized in Figure 2.19.

We now discuss the three stages of the algorithm in detail.

1. Root finding from a subsampled signal

The given signal q_n is subsampled to give q_n^{sub} with D_{sub} samples. The corresponding step-size is h_{sub} . There are no results for the minimum number of samples that guarantee that all eigenvalues will be found. One choice can be based on limiting the overall computational complexity to $\mathcal{O}(D \log^2 D)$, which is the complexity for the reflection coefficient. For a root-finding algorithm with $\mathcal{O}(D^2)$ complexity, we choose to use $D_{\text{sub}} = \text{round}(\sqrt{D \log^2 D})$ samples. The polynomial $\hat{a}_{\text{num}}(z)$ is then built from these D_{sub} samples. For $\text{FCF}_2^{[4]}$, the non-equispaced samples should be obtained from the original D samples and not the D_{sub} samples. An eigenmethod is then used to find all zeros of $\hat{a}_{\text{num}}(z)$. We used the algorithm in [179]. The values of z are mapped backed to λ and filtered to remove implausible values.

2. Root refinement using Newton method

The Newton method based on the slow CF methods is used for root refinement. The derivative $da(\lambda)/d\lambda$ is calculated numerically along with $a(\lambda)$ as in [116] using all the given samples q_n . The values of λ that remain after filtering in the previous

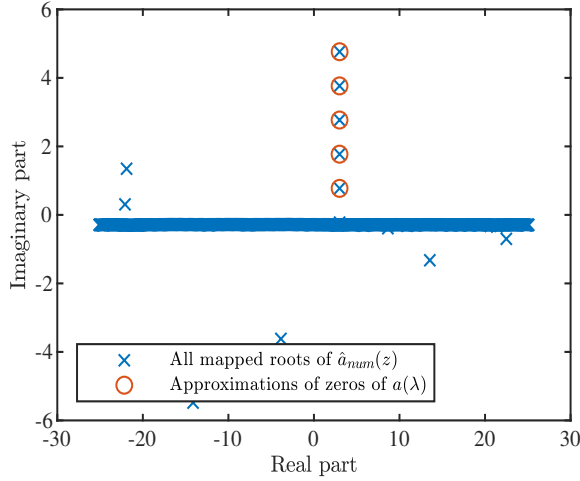


Figure 2.18: Mapped zeros of $\hat{a}_{\text{num}}(z)$ for Example 1 with $D = 2^9$.

Input: Samples $q_0, \dots, q_{D-1}, D_{\text{sub}}, [T_-, T_+]$

Output: Estimated eigenvalues $\hat{\lambda}_k$

- 1: Subsample the given samples $\rightarrow q_n^{\text{sub}} := q_{n \lfloor D/D_{\text{sub}} \rfloor}$, where $\lfloor \cdot \rfloor$ rounds to the closest integer.
Build a polynomial approximation $\hat{a}_{\text{num}}(z)$ using the D_{sub} subsampled samples.
Apply a fast eigenmethod to find the roots of $\hat{a}_{\text{num}}(z)$.
Filter the roots.
- 2: Refine the roots via Newton's method using all samples.
Filter the refined roots.
- 3: Apply Richardson extrapolation to the unrefined and refined roots.

Figure 2.19: **Algorithm** : Subsample and refine

step are used as the initial guesses for the Newton method. We chose to stop the iterations if the change in value goes below 10^{-15} or if a maximum of 15 iterations is reached. The resulting roots are filtered again.

3. Richardson extrapolation

We pair the roots resulting from the Newton step, $\hat{\lambda}_k^{\text{Newton}}$, with the corresponding initial guesses $\hat{\lambda}_k^{\text{init}}$. Then, we apply Richardson extrapolation:

$$\hat{\lambda}_k = \frac{\left(\frac{h_{\text{sub}}}{h}\right)^r \hat{\lambda}_k^{\text{Newton}} - \hat{\lambda}_k^{\text{init}}}{\left(\frac{h_{\text{sub}}}{h}\right)^r - 1}. \quad (2.33)$$

$\hat{\lambda}_k$ is then an improved approximation and constitutes the discrete part of the FCF_RE algorithm. It may so happen that more than one $\hat{\lambda}_k^{\text{init}}$ converge to the same $\hat{\lambda}_k^{\text{Newton}}$. In such cases the value $\hat{\lambda}_k^{\text{init}}$ closest to $\hat{\lambda}_k^{\text{Newton}}$ should be used for

Richardson extrapolation. The other values $\hat{\lambda}_k^{\text{init}}$ that also converged to the same $\hat{\lambda}_k^{\text{Newton}}$ should be treated as spurious values and discarded.

The numerical algorithms may not find particular eigenvalues or find spurious ones. Let $\hat{\mathbf{A}}$ be the set of approximations found by an algorithm. To penalize both missed values and incorrect spurious values at the same time, we define the error

$$E\mathbf{A} = \max \left\{ \max_{\lambda_i \in \mathbf{A}} \min_{\hat{\lambda}_j \in \hat{\mathbf{A}}} |\lambda_i - \hat{\lambda}_j|, \max_{\hat{\lambda}_j \in \hat{\mathbf{A}}} \min_{\lambda_i \in \mathbf{A}} |\lambda_i - \hat{\lambda}_j| \right\}. \quad (2.34)$$

Note that the first term in the outer maximum grows large if an algorithm fails to approximate a part of the set \mathbf{A} while the second term becomes large if an algorithm finds spurious values that have no correspondence with values in \mathbf{A} . $E\mathbf{A}$ is expected to be of order r for an algorithm of order r .

2.C.3. NUMERICAL EXAMPLE

In this section, we compare different variants of our proposed algorithm using Example 1. We compute the error $E\mathbf{A}$ for the following three types of algorithms:

1. Discrete part of FCF algorithms. An eigenmethod is applied to the approximation $\hat{a}_{\text{num}}(z)$ built using all samples. No sub-sampling is used.
2. Discrete part of FCF algorithms with sub-sampling. Only steps 1 and 2 of the algorithm mentioned above.
3. Discrete part of FCF_RE algorithms. All the three steps mentioned above.

To demonstrate the effect of sub-sampling, we show in Figure 2.20 the errors for the second- and fourth-order algorithms of types 1 and 2. For $h > 0.3$ the errors are high either due to failure to find approximations close to the actual eigenvalues or due to spurious values. For $h \leq 0.3$, $\text{FCF}_2^{[4]}$ of type 1 and $\text{FCF}_1^{[2]}$ of type 2 find exactly five values that are close approximations of the values in \mathbf{A} . However $\text{FCF}_1^{[2]}$ of type 1 and $\text{FCF}_2^{[4]}$ of type 2 find good approximations only for $h \leq 0.06$. The error of $\text{FCF}_1^{[2]}$ algorithms decreases with slope two and that of $\text{FCF}_2^{[4]}$ algorithms decreases with slope four as expected from the order of the underlying numerical schemes.

In Figure 2.21 we show the errors for the second- and fourth-order algorithms of type 2 and 3 to indicate the advantage of the extrapolation step. The extrapolation step improves the approximation significantly for $\text{FCF_RE}_1^{[2]}$ while adding negligible computation cost to the algorithm. However, there is only minor improvement in case of $\text{FCF_RE}_2^{[4]}$ over $\text{FCF}_2^{[4]}$.

In Figure 2.22 we plot the execution times for the FCF algorithms of types 1 and 2. The execution times of algorithms of type 3 are almost the same as those of type 2. For type 1 algorithms, these times include the time required to build $\hat{a}_{\text{num}}(z)$ and the time taken by the root-finder. For algorithms of type 2, the additional time required for root-refinement by Newton's method is also included. Even with sub-sampling, we see that

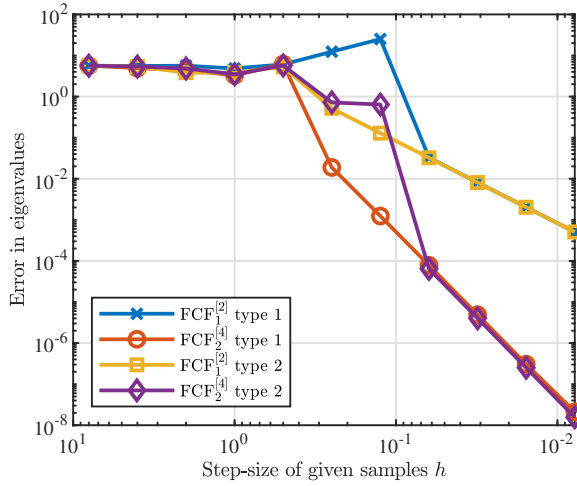


Figure 2.20: Error in approximation of the eigenvalues by the fast second- and fourth-order algorithms of type 1 (no sub-sampling) and type 2 (sub-sample and refine, no Richardson extrapolation).

the execution times are an order of magnitude higher than the execution times for the continuous part. The FCF_RE algorithms seem to provide the best trade-off between accuracy and computation cost similar to the case of continuous part. The overall computational complexity may be decreased by using alternative methods to find the initial guesses.

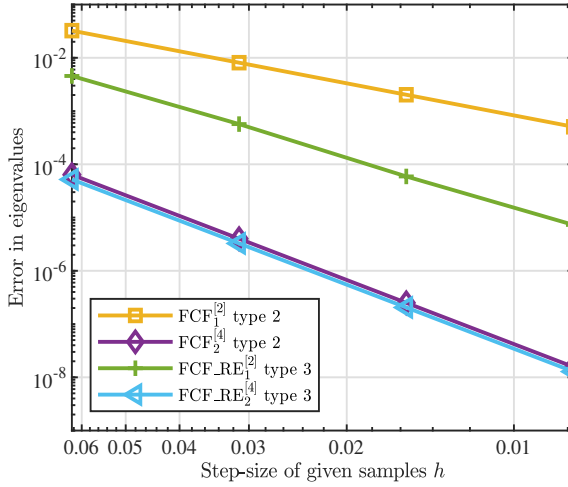


Figure 2.21: Error in approximation of a eigenvalues computed using FCF and FCF_RE algorithms.

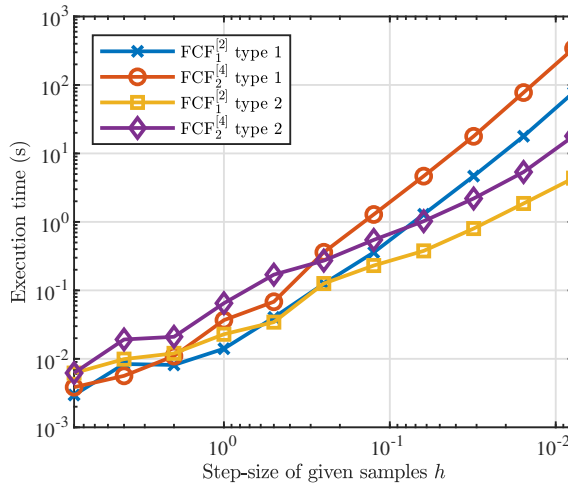


Figure 2.22: Execution time of $FCF_2^{[4]}$ and $FCF_2^{[4]}$ algorithms for computing eigenvalues of Example 1.

3

ON COMPUTING HIGH-DIMENSIONAL RIEMANN THETA FUNCTIONS

CHAPTER ABSTRACT

Riemann theta functions play a crucial role in the field of nonlinear Fourier analysis, where they are used to realize inverse nonlinear Fourier transforms for periodic signals. The practical applicability of this approach has however been limited since Riemann theta functions are multi-dimensional Fourier series whose computation suffers from the curse of dimensionality. In this chapter, we investigate several new approaches to compute Riemann theta functions with the goal of unlocking their practical potential. Our first contributions are novel theoretical lower and upper bounds on the series truncation error. These bounds allow us to rule out several of the existing approaches for the high-dimension regime. We then propose to consider low-rank tensor and hyperbolic cross based techniques. We first examine a tensor-train based algorithm which utilizes the popular scaling and squaring approach. We show theoretically that this approach cannot break the curse of dimensionality. Finally, we investigate two other tensor-train based methods numerically and compare them to hyperbolic cross based methods. Using finite-genus solutions of the Korteweg-de Vries (KdV) and nonlinear Schrödinger (NLS) equations, we demonstrate the accuracy of the proposed algorithms. The tensor-train based algorithms are shown to work well for low genus solutions with real arguments but are limited by memory for higher genera. The hyperbolic cross based algorithm also achieves high accuracy for low genus solutions. Its novelty is the ability to feasibly compute moderately accurate solutions for high dimensions (up to 60). It therefore enables the computation of complex inverse nonlinear Fourier transforms that were so far out of reach.

The text in this chapter is based on the article S. Chimmalgi and S. Wahls, "On Computing Riemann Theta Functions," which has been submitted to IEEE Open Journal of Signal Processing.

3.1. INTRODUCTION

The Riemann theta function

$$\theta(\mathbf{z} | \mathbf{\Omega}) = \sum_{\mathbf{n} \in \mathbb{Z}^g} e^{2\pi i(\frac{1}{2}\mathbf{n} \cdot \mathbf{\Omega} \mathbf{n} + \mathbf{n} \cdot \mathbf{z})}, \quad \mathbf{z} \in \mathbb{C}^g, \quad (3.1)$$

where $\mathbf{\Omega} = \mathbf{\Omega}^T \in \mathbb{C}^{g \times g}$ has a strictly positive definite imaginary part and \cdot indicates the dot product, is a particular multi-dimensional Fourier series that plays a key role in the area of nonlinear Fourier analysis [39], [41]. There, it is used to synthesize periodic signals as part of inverse nonlinear Fourier transforms. Periodic nonlinear Fourier analysis has recently received a lot of attention in nonlinear signal processing problems arising in fiber-optic communications [102], [107], [111], [112], [128] and coastal and ocean engineering [30], [180]–[185]. The Riemann theta function also sees application in quantum coding [186], algebraic geometry [187]–[189], number theory [190], discrete mathematics [191], cryptography [192] and statistics [193]. Despite its applicability in many fields, the practical utility of the Riemann theta function is limited to low number of dimensions due to its high computational cost. While there has been much work on designing efficient algorithms for the computation of the Riemann theta function [30], [49], [152], [194], [195], the complexity of these methods nevertheless increases exponentially with the number of dimensions. Hence, they are limited to low-dimensional problems. In this chapter we propose novel approaches that overcome this limitation and allow us to synthesize high-dimensional non-trivial signals for fiber-optic communications and coastal engineering problems.

Algorithms for computing the Riemann theta function can be primarily classified into two categories. The first category of algorithms concentrate on computing the Riemann theta function value up to a certain number of bits [194]. The second category of algorithms aim at computing the theta function value up to a small threshold. In this chapter we focus on the second category of algorithms [30], [49], [152], [195]. These methods approximate the theta function value by summing a truncated series. The number of terms in the summation grows exponentially with the number of dimensions g . Hence, the computational cost grows exponentially as well. It becomes infeasible to compute the sum even for moderate values of g . This is famously known as the *curse of dimensionality*. The curse is seen in many high-dimensional problems where the number of operations required for a particular action grow exponentially with the underlying dimensionality [196]. In recent years tensor based methods have been increasingly employed to mitigate the curse of dimensionality. They have been applied with great success in signal processing, statistics, data mining, and machine learning [197]–[205]. In particular, they have been used to develop efficient algorithms for computing multi-dimensional Fourier series [206], [207]. Another approach used to reduce the computation cost of the multi-dimensional Fourier series is the utilization of special index sets [208]–[212].

In this chapter we study the applicability of tensor based algorithms and special index sets to efficiently compute approximations of the high-dimensional Riemann theta function. Our main contributions are: 1) Lower and upper bounds on the error introduced from the series truncation for certain index sets, 2) theoretical proof that a standard scaling-and-squaring approach applied to tensor-train approximations cannot

break the curse of dimensionality, and 3) numerical investigations of other tensor-train and hyperbolic-cross based algorithms. To the best of our knowledge, this is the first time that tensor-train and hyperbolic-cross methods have been proposed for the computation of Riemann theta functions. In our numerical experiments, we are able to compute Riemann theta function values for very high number of dimensions ($g = 60$) that dramatically exceed the current state of the art in this area, which significantly extends the range of practical problems to which Riemann theta functions can be applied.

The remainder of this chapter has the following structure. In Section 3.2 we introduce some preliminaries about the Riemann theta function and the tensor-train decomposition. In Section 3.3 we derive some lower bounds and an upper bound on the series truncation error for certain index sets. In Section 3.4 we provide a theoretical analysis to prove that a tensor-train based approach using scaling and squaring cannot break the curse of dimensionality. In Section 3.5 we present two alternatives for the tensor-train based algorithm and use the hyper-elliptic solutions of the KdV and NLS equations as numerical examples to study the accuracies of the algorithms. We conclude our findings in Section 3.6.

NOTATIONS

\mathbb{C} - complex numbers; \mathbb{N} - Natural numbers; \mathbb{Z} - Integers; $\#A$ denotes the number of elements in the set A . We use $e^{(\cdot)}$ and $\exp(\cdot)$ interchangeably to indicate the exponential function (applied element-wise for tensors).

3.2. PRELIMINARIES

In this section, we recapitulate several results related to the computation of the Riemann theta function from the literature. First, the most common ways to truncate the infinite series (3.1) are introduced. Then, a technique to make truncation more efficient by transforming the Riemann matrix Ω that is known as Siegel transform is discussed. Finally, tensor trains are introduced as a potential tool to evaluate the truncated series.

3.2.1. TRUNCATED RIEMANN THETA FUNCTIONS

For numerical purposes, the Riemann theta function (3.1) is typically approximated by a truncated series of the form

$$\hat{\theta}(\mathbf{z} | \Omega) = \sum_{\mathbf{n} \in \mathcal{N}^g(N)} \underline{\mathbf{C}}(\mathbf{n}) e^{2\pi i(\mathbf{n} \cdot \mathbf{z})}, \quad (3.2)$$

where $\underline{\mathbf{C}}$ is a g -dimensional tensor with

$$\underline{\mathbf{C}}(\mathbf{n}) = e^{i\pi \mathbf{n} \cdot \Omega \mathbf{n}}. \quad (3.3)$$

The index set $\mathcal{N}^g = \mathcal{N}^g(N)$ depends on a truncation parameter $N \in \mathbb{N}$. Before we can introduce several popular index sets, the definition of the matrix p -norms has to be recalled.

Definition 1. Let $p \in \mathbb{N} \cup \{\infty\}$. The p -norm of $\mathbf{z} \in \mathbb{C}^g$ is $\|\mathbf{z}\|_p := (|z_1|^p + \dots + |z_g|^p)^{1/p}$ for $p < \infty$ and $\|\mathbf{z}\|_p := \max\{|z_1|, \dots, |z_g|\}$ for $p = \infty$. The corresponding induced p -norm of a matrix $\mathbf{A} \in \mathbb{C}^{g \times g}$ is $\|\mathbf{A}\|_p := \max_{\substack{\mathbf{z} \in \mathbb{C}^g \\ \|\mathbf{z}\|_p \leq 1}} \|\mathbf{A}\mathbf{z}\|_p$.

The first popular family index set, \mathcal{I}^g , is defined as follows.

Definition 2. For any $N \in \mathbb{N}$ and $p \in \mathbb{N} \cup \{\infty\}$, we set

$$\mathcal{I}^g = \mathcal{I}^g(p, N) := \{\mathbf{n} \in \mathbb{Z}^g : \|\mathbf{n}\|_p \leq N\}.$$

The most common choices of p are $p = 1$ (summation over a cross-polytope), $p = 2$ (summation over a hypersphere), and $p = \infty$ (summation over a hypercube). The symmetric hyperbolic cross \mathcal{H}^g is another interesting index set [213, Figure 2.1(b)]. It is defined as follows.

Definition 3. For any $N \in \mathbb{N}$, we set

$$\mathcal{H}^g = \mathcal{H}^g(N) := \left\{ \mathbf{n} \in \mathbb{Z}^g : \prod_{j=1}^g \max(1, 2|n_j|) \leq N \right\}.$$

The hyperbolic cross \mathcal{H}^g can be built recursively using the algorithm given in [214, Section 2.6]. In contrast to the index sets $\mathcal{I}^g(p, N)$, the number of elements in the hyperbolic cross \mathcal{H}^g does not grow exponentially in the genus g if N is fixed.

Lemma 1. The number of elements in $\mathcal{H}^g(N)$ satisfies

$$\#\mathcal{H}^g(N) \leq e^2 N^2 g^{\log_2 N}.$$

Proof. Since $1 + |n_j| \leq \max(1, 2|n_j|)$, we have that $\max(1, 2|n_j|) \leq N \Rightarrow 1 + |n_j| \leq N$. Therefore, $\mathcal{H}^g(N) \subseteq \{\mathbf{n} \in \mathbb{Z}^g : \prod_{j=1}^g (1 + |n_j|) \leq N\}$. The bound [215, Eq. 10.2.3] implies $\#\mathcal{H}^g(N) \leq e^2 N^{2+\log_2 g} = e^2 N^2 N^{\log_2 g}$. The lemma now follows since $N^{\log_2 g} = (2^{\log_2 N})^{\log_2 g} = (2^{\log_2 g})^{\log_2 N} = g^{\log_2 N}$. \square

The Riemann theta function is well-known to converge absolutely and uniformly in \mathbf{z} [216, Ch. II.1]. Therefore, we have the following result.

Theorem 1 (Convergence). Let $Z \subset \mathbb{C}$ be compact. Then

$$\lim_{N \rightarrow \infty} \max_{\mathbf{z} \in Z} |\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| = 0$$

for any of the index sets \mathcal{I}^g and \mathcal{H}^g .

3.2.2. SIEGEL TRANSFORM

For certain Riemann matrices $\mathbf{\Omega}$, it may happen that an index set contains many coefficients that could be neglected during the summation. If we sum over a hypersphere, this issue would for example arise if the indices of the non-negligible coefficients form an hyper-ellipsoid with high eccentricity. An algorithm for finding an hyper-ellipsoid of indices that includes all terms above a threshold is given in [49, Section 4]. The algorithm for identifying the hyper-ellipsoid of indices however unfortunately has a significant computational cost itself that grows sharply with increasing g . Furthermore, even if the hyper-ellipsoid is known, the number of terms inside it can still be very large [49,

p. 1734]. In [49, Section 7], it was therefore proposed to use a modular transform of the form

$$\mathbf{\Omega} \mapsto \hat{\mathbf{\Omega}} = (\mathbf{A}\mathbf{\Omega} + \mathbf{B})(\mathbf{C}\mathbf{\Omega} + \mathbf{D})^{-1}, \quad \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D} \in \mathbb{Z}^g, \quad (3.4)$$

where the integer matrices \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} must satisfy

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{0}_g & \mathbf{I}_g \\ -\mathbf{I}_g & \mathbf{0}_g \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^T = \begin{bmatrix} \mathbf{0}_g & \mathbf{I}_g \\ -\mathbf{I}_g & \mathbf{0}_g \end{bmatrix}, \quad (3.5)$$

known as Siegel transformation to reduce the eccentricity of the set of non-negligible indices. The relation between the two Riemann theta functions that correspond to the two Riemann matrices $\mathbf{\Omega}$ and $\hat{\mathbf{\Omega}}$ is then given by [216, Eq. 5.1]

$$\theta(\mathbf{z} | \mathbf{\Omega}) = \frac{\theta((\mathbf{C}\mathbf{\Omega} + \mathbf{D})^{-1}\mathbf{z} | \hat{\mathbf{\Omega}})}{\zeta \sqrt{\det(\mathbf{C}\mathbf{\Omega} + \mathbf{D})} e^{\pi i \mathbf{z}^T (\mathbf{C}\mathbf{\Omega} + \mathbf{D})^{-1} \mathbf{C} \mathbf{z}}}, \quad (3.6)$$

where ζ is an eighth root of one, i.e., $\zeta^8 = 1$. The main computational step in the construction of the modular transform is the approximation of the shortest vector in a lattice. In [49], the authors employed the LLL algorithm [217] for that purpose. The complexity of the LLL algorithm is only polynomial in the g , but the error in the approximation increases exponentially in g . Several authors therefore investigated replacements for the LLL algorithm [152], [193].

The authors of [153] mention that once a Siegel transform has been applied, the summation can be carried over a hyper-cube instead of the hyper-ellipsoid as it was done in [49], at an additional cost.

3.2.3. TENSOR-TRAIN DECOMPOSITION

The truncated Riemann theta function (3.2) is a multi-dimensional Fourier series. In [206], it was observed that if the coefficient tensor has a low-rank representation in the tensor-train format [218], a multi-dimensional Fourier series can be evaluated with low computational complexity. The idea was to exploit that a multi-dimensional Fourier series constitutes the inner product between the coefficient tensor and the rank one tensor formed by the terms $e^{2\pi i \mathbf{n} \cdot \mathbf{z}}$, which can be then computed efficiently using the method in [218, Section 4.2]. Later in this chapter, we will exploit this idea for the computation of the truncated Riemann theta function. Since this requires us to investigate approximations of the coefficient tensor (3.3), we now quickly recall some facts about the tensor-train format.

Any given tensor $\underline{\mathbf{Y}}$ can be approximated arbitrarily well by a tensor $\underline{\mathbf{X}} \approx \underline{\mathbf{Y}}$ of the form $\underline{\mathbf{X}} = \underline{\mathbf{G}}^{(1)} \times^1 \underline{\mathbf{G}}^{(2)} \times^1 \dots \times^1 \underline{\mathbf{G}}^{(g)}$, [218], where \times^1 is the contracted product [219, Section 2.2] and the $\underline{\mathbf{G}}^{(k)}$ are 3rd-order tensors with sizes $R_{k-1} \times N_k \times R_k$, $k = 1, \dots, g$ and $R_0 = R_g = 1$. Any tensor of this form is said to be a tensor train. The tensors $\underline{\mathbf{G}}^{(k)}$ are called the tensor-train cores, while the integers R_1, \dots, R_{g-1} are called the tensor-train ranks. A tensor-train can alternatively be written entry-wise as a product of slice matrices,

$$\mathbf{x}_{\mathbf{n}} = x_{n_1, n_2, \dots, n_g} = \mathbf{G}_{n_1}^{(1)} \mathbf{G}_{n_2}^{(2)} \dots \mathbf{G}_{n_g}^{(g)}, \quad (3.7)$$

where $\mathbf{G}_{n_k}^{(k)} = \underline{\mathbf{G}}^{(k)}(:, n_k, :)$ $\in \mathbb{C}^{R_{k-1} \times R_k}$ is the lateral slice of the n th tensor-train core, $k = 1, \dots, g$, and $\mathbf{G}_{n_1}^{(1)} \in \mathbb{C}^{1 \times R_1}$ and $\mathbf{G}_{n_g}^{(g)} \in \mathbb{C}^{R_{g-1} \times 1}$. There exist efficient algorithms with which

a tensor-train approximation of a given tensor can be found under accuracy and rank constraints, see e.g. [218], [220].

3.3. ANALYSIS OF THE TRUNCATION ERROR

For the numerical evaluation of the truncated Riemann theta function $\hat{\theta}(\mathbf{z} | \mathbf{\Omega})$, the truncation parameter N has to be chosen large enough such that the truncation error

$$|\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| = \left| \sum_{\mathbf{n} \notin \mathcal{N}^g(N)} c_{\mathbf{n}} e^{2\pi i \mathbf{n} \cdot \mathbf{z}} \right| \quad (3.8)$$

is sufficiently small. The truncation error depends on the choice of the index set $\mathcal{N}^g(N)$. The parameters g and N furthermore determine the complexity of evaluation methods for the truncated Riemann theta function. In this section, we show that the two parameters are connected. We derive lower and upper bounds on the truncation parameter N such that a certain truncation error can be guaranteed. Since we are interested in high dimensional cases, their behavior is studied for large g . The consequences of these studies are discussed.

3.3.1. LOWER BOUNDS ON THE TRUNCATION ERROR

Some authors have proposed to choose the truncation parameter N such that the truncated tensor coefficients satisfy $|c_{\mathbf{n}}| < \varepsilon$, where $\varepsilon > 0$ denotes some small parameter such as machine precision [153, Section 3.3], [152, p. 150], [30]. However, even if the errors in the individual coefficients $c_{\mathbf{n}}$ are very small, the truncation error (3.8) can be large since the number of neglected coefficients grows exponentially with the genus g . (Note that it is possible to sum terms accurately even when the numbers have significantly different orders of magnitude and are smaller than the machine precision [221]. Hence, it would be possible include coefficients below machine precision also in finite precision arithmetic.) The following proposition, which is our first contribution, formalizes this observation for most of the index sets \mathcal{J}^g . It demonstrates that the strategy of truncating coefficients below machine precision can achieve small truncation errors only for small values of g with these index sets since the truncation error in general grows exponentially in the number of dimensions g if N is fixed.

Proposition 1. *Let the index set be $\mathcal{N}^g = \mathcal{J}^g(p, N)$ for any $p \in \mathbb{N}$, $p \geq 2$. The truncation error is then lower bounded as*

$$|\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| \geq e^{g \left(\log(2) - \pi \lambda_{\max} \left[\frac{N+1}{\sqrt{g}} \right]^2 \right)} \quad (3.9)$$

at $\mathbf{z} = \mathbf{0}_{g \times 1}$ whenever $\Re\{\mathbf{\Omega}\} = \mathbf{0}_g$, where λ_{\max} denotes the largest eigenvalue of $\Im\{\mathbf{\Omega}\}$.

Proof. Let $\mathcal{J}^g(N+1)$ denote the index set that contains all $\mathbf{n} \in \mathbb{Z}^g$ of the form $\mathbf{n} = \left[\frac{N+1}{\sqrt{g}} \right] [s_1 \ s_2 \ \cdots \ s_g]^T$ where $s_k = \pm 1$ for $k = 1, 2, \dots, g$. Then

$$\mathbf{n} \in \mathcal{J}^g(N+1) \Rightarrow \|\mathbf{n}\|_p \geq N+1 \Rightarrow \mathbf{n} \notin \mathcal{J}^g(p, N). \quad (3.10)$$

The truncation error thus satisfies

$$\begin{aligned}
 |\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| &= \left| \sum_{\mathbf{n} \notin \mathcal{J}^g(p, N)} c_{\mathbf{n}} \underbrace{\exp(2\pi i \mathbf{n} \cdot \mathbf{z})}_{=1} \right| \\
 (c_{\mathbf{n}} \in \mathbb{R}^+) &= \sum_{\mathbf{n} \notin \mathcal{J}^g(p, N)} c_{\mathbf{n}} \\
 (3.10) &\geq \sum_{\mathbf{n} \in \mathcal{J}^g(N+1)} e^{-\pi \mathbf{n} \cdot \Im\{\mathbf{\Omega}\} \mathbf{n}} \\
 (\mathbf{n} \cdot \Im\{\mathbf{\Omega}\} \mathbf{n} \leq \lambda_{\max} \|\mathbf{n}\|_2^2) &\geq \sum_{\mathbf{n} \in \mathcal{J}^g(N+1)} e^{-\pi \lambda_{\max} g \left\lceil \frac{N+1}{\sqrt[2]{g}} \right\rceil^2} \\
 (\text{since } \#\mathcal{J}^g(N+1) = 2^g) &= 2^g e^{-\pi \lambda_{\max} g \left\lceil \frac{N+1}{\sqrt[2]{g}} \right\rceil^2} \\
 &= e^{g \left(\log(2) - \pi \lambda_{\max} \left\lceil \frac{N+1}{\sqrt[2]{g}} \right\rceil^2 \right)}
 \end{aligned}$$

□

To the best of our knowledge, this is the first lower bound on the approximation error of Riemann theta functions. Note that in order to keep the lower bound (3.9) on the truncation error from blowing up as g increases, the truncation parameter N should grow at least proportionally to $\sqrt[2]{g}$.

The previous result did not cover the cases $p \in \{1, \infty\}$. The next proposition provides a weaker but more general lower bound on the truncation error, which shows that it must grow at least linearly in the genus g .

Proposition 2. *Let the index set be $\mathcal{N}^g = \mathcal{J}^g(p, N)$ for any $p \in \mathbb{N} \cup \{\infty\}$. The truncation error is then lower bounded as*

$$|\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| \geq 2g e^{-\pi \lambda_{\max} (N+1)^2}$$

at $\mathbf{z} = \mathbf{0}_{g \times 1}$ whenever $\Re\{\mathbf{\Omega}\} = \mathbf{0}_g$, where λ_{\max} denotes the largest eigenvalue of $\Im\{\mathbf{\Omega}\}$.

Proof. Let $\mathcal{J}^g(N+1)$ denote the index set that contains all $\mathbf{n} \in \mathbb{Z}^g$ for which exactly one element is non-zero, and this element is either $N+1$ or $-(N+1)$. Then

$$\mathbf{n} \in \mathcal{J}^g(N+1) \Rightarrow \|\mathbf{n}\|_p = N+1 \Rightarrow \mathbf{n} \notin \mathcal{J}^g(p, N). \quad (3.11)$$

The truncation error thus satisfies

$$\begin{aligned}
 |\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| &= \left| \sum_{\mathbf{n} \notin \mathcal{J}^g(p, N)} c_{\mathbf{n}} \underbrace{\exp(2\pi i \mathbf{n} \cdot \mathbf{z})}_{=1} \right| \\
 (c_{\mathbf{n}} \in \mathbb{R}^+) &= \sum_{\mathbf{n} \notin \mathcal{J}^g(p, N)} c_{\mathbf{n}} \\
 (3.11) &\geq \sum_{\mathbf{n} \in \mathcal{J}^g(N+1)} e^{-\pi \mathbf{n} \cdot \Im\{\mathbf{\Omega}\} \mathbf{n}} \\
 (\mathbf{n} \cdot \Im\{\mathbf{\Omega}\} \mathbf{n} \leq \lambda_{\max} \|\mathbf{n}\|_2^2) &\geq \sum_{\mathbf{n} \in \mathcal{J}^g(N+1)} e^{-\pi \lambda_{\max}(N+1)^2} \\
 (\text{since } \#\mathcal{J}^g(N+1) = 2g) &= 2ge^{-\pi \lambda_{\max}(N+1)^2}
 \end{aligned}$$

□

The next proposition finally provides a similar bound for the hyperbolic cross.

Proposition 3. *Let the index set be $\mathcal{N}^g = \mathcal{H}^g(N)$. The truncation error is then lower bounded as*

$$|\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| \geq 2ge^{-\pi \lambda_{\max} \lceil \frac{N+1}{2} \rceil^2}$$

at $\mathbf{z} = \mathbf{0}_{g \times 1}$ whenever $\Re\{\mathbf{\Omega}\} = \mathbf{0}_g$, where λ_{\max} denotes the largest eigenvalue of $\Im\{\mathbf{\Omega}\}$.

Proof. Let $\mathcal{J}^g(N+1)$ denote the index set that contains all $\mathbf{n} \in \mathbb{Z}^g$ for which exactly one element is non-zero, and this element is either $\lceil \frac{N+1}{2} \rceil$ or $-\lceil \frac{N+1}{2} \rceil$. Then

$$\begin{aligned}
 \mathbf{n} \in \mathcal{J}^g(N+1) &\Rightarrow \prod_{j=1}^g \max(1, 2|n_j|) \geq N+1 \\
 &\Rightarrow \mathbf{n} \notin \mathcal{H}^g(N).
 \end{aligned} \tag{3.12}$$

Following the same steps as in the proof of Prop. 2, we arrive at the lower bound on the truncation error for \mathcal{H}^g . □

3.3.2. AN UPPER BOUND FOR $\mathcal{J}^g(\infty, N)$

The lower bounds in the previous subsection have shown that the truncation parameter N in general has to grow with the genus g if small truncation errors are desired. We are now investigating upper bounds. Upper bounds for the truncation errors of transformed Riemann theta functions that are summed over ellipsoids are provided in [49, Thm. 3], [195, Thm. 3.1]. Upper bounds with respect to the hyperbolic cross $\mathcal{H}^g(N)$ and $\mathcal{J}^g(1, N)$ are provided in [212, Ch. 8.1] for the case $\mathbf{z} \in \mathbb{R}^g$, but the influence of the genus on these bounds is unfortunately not investigated.

The following proposition, which is our next contribution, shows that the truncation error can be bounded independently of the genus for the hypercube if the truncation parameter grows slightly faster than $\Omega(\sqrt{g})$ for real \mathbf{z} , or $\Omega(g)$ for non-real \mathbf{z} . This case is later of special interest since it is possible to evaluate the truncated Riemann theta function fast over the hypercube if the coefficient tensor is in some sense low rank.

Proposition 4. *Let the index set be $\mathcal{I}^g(\infty, N)$ and fix any $\delta \in (0, 1)$ and $a > 0$. Then there exists a constant $c > 0$ (independent of g and N) such that the truncation error is upper bounded as*

$$|\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| \leq \sum_{k=N+1}^{\infty} e^{-(\pi/2)\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}k^2}$$

whenever $cN^{2-\delta} \geq g$ and $\Im\{\mathbf{z}\} = \mathbf{0}_{g \times 1}$. If $\|\Im\{\mathbf{z}\}\|_{\infty} \leq a$, the same bound holds whenever $cN \geq g$ (for a different $c > 0$).

Proof. Recall that from the proof of Theorem 1 that

$$\begin{aligned} & |\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| \\ & \leq \sum_{k=N+1}^{\infty} e^{g \log(2k+1) + 2\pi\|\Im\{\mathbf{z}\}\|_{\infty} g k - \pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}k^2}. \end{aligned}$$

We first consider the case $g \leq cN^{2-\delta}$ and $\Im\{\mathbf{z}\} = \mathbf{0}_{g \times 1}$. The constants are then chosen as $M := \max_{x>0} \frac{\log(1+2x)}{x^\delta} > 0$, which is finite because $x^{-\delta} \log(1+2x)$ is continuous and converges to zero for both $x \rightarrow 0$ and $x \rightarrow \infty$, and $c := \frac{\pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}}{2M} > 0$. The bound on the truncation error becomes

$$\begin{aligned} & |\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| \\ & \leq \sum_{k=N+1}^{\infty} e^{cN^{2-\delta} \log(2k+1) - \pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}k^2} \\ & \leq \sum_{k=N+1}^{\infty} e^{ck^{2-\delta} \log(2k+1) - \pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}k^2} \\ & = \sum_{k=N+1}^{\infty} e^{(ck^{-\delta} \log(2k+1) - \pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1})k^2} \\ & \leq \sum_{k=N+1}^{\infty} e^{(cM - \pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1})k^2} \\ & = \sum_{k=N+1}^{\infty} e^{-(\pi/2)\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}k^2}. \end{aligned}$$

In the case $g \leq cN$ and $\Im\{\mathbf{z}\} \neq \mathbf{0}_{g \times 1}$, we instead choose $M := \max_{x>0} \frac{\log(1+2x)}{x} > 0$, $c := \frac{\pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}}{2(M+2\pi a)} > 0$. We arrive at the same bound as before:

$$\begin{aligned} & |\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| \\ & \leq \sum_{k=N+1}^{\infty} e^{cN \log(2k+1) + 2\pi\|\Im\{\mathbf{z}\}\|_{\infty} cNk - \pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}k^2} \\ & \leq \sum_{k=N+1}^{\infty} e^{(cM + 2\pi ac - \pi\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1})k^2} \\ & \leq \sum_{k=N+1}^{\infty} e^{-(\pi/2)\|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1}k^2}. \end{aligned}$$

□

3.3.3. DISCUSSION

In this section, we have found that the truncation parameter N needed to achieve a certain truncation error (3.8) increases with the genus g . The exact behavior depends on the choice of the index set $\mathcal{N}^g(N)$. We have shown that for the hypersphere $\mathcal{S}^g(2, N)$, N has in general to grow *at least* as \sqrt{g} for real \mathbf{z} . Since we have to sum approximately $V_g(N) \sim \frac{1}{\sqrt{g\pi}} \left(\frac{2\pi e}{g}\right)^{g/2} N^g$ terms for the naive evaluation of the truncated Riemann function, we see that this approach quickly becomes infeasible as the genus g increases. For the hypercube $\mathcal{S}^g(\infty, N)$, we have shown that the truncation parameter N has to grow *at most* slightly faster than \sqrt{g} for real \mathbf{z} . Naive evaluation of the truncated Riemann theta function requires us to sum $(2N+1)^g$ terms, which also becomes quickly infeasible. The complexity of evaluating the truncated Riemann theta function over a hypercube however becomes low even for large genus and truncation parameter if the coefficient tensor is approximated well by a low-rank tensor-train [206]. Therefore, we will investigate this case further in the following.

The behavior of the hyperbolic cross is less clear at the moment. We will later investigate its performance numerically.

3.4. SUMMING OVER HYPERCUBES USING TENSOR TRAINS AND SCALING AND SQUARING

The truncated Riemann theta function $\hat{\theta}(\mathbf{z} | \mathbf{\Omega})$ in (3.2) over a hypercube can be represented as the inner product between the tensor $\underline{\mathbb{C}}$ in (3.3) and the tensor $e^{2\pi i \mathbf{n} \cdot \mathbf{z}}$. If the tensor $\underline{\mathbb{C}}$ has a low rank approximation in the tensor-train format, the complexity of computing the Riemann theta function value will be low since the tensor-train rank of the second tensor is one [218, Section 4.2]. The tensor $\underline{\mathbb{C}}$ is the point-wise exponential of the tensor formed from $i\pi \mathbf{n} \cdot \mathbf{\Omega} \mathbf{n}$, which can be represented exactly in the tensor-train format (see Section 3.B). The most common approach to approximate the pointwise exponential of a tensor is the scaling and squaring method. In our case, this means that the truncated Riemann theta function is approximated with

$$\tilde{\theta}(\mathbf{z} | \mathbf{\Omega}) = \sum_{\mathbf{n} \in \mathcal{S}^g(\infty, N)} \tilde{c}_{\mathbf{n}} e^{2\pi i \mathbf{n} \cdot \mathbf{z}}, \quad (3.13)$$

where the coefficient tensor $\underline{\tilde{\mathbb{C}}}$ with terms

$$\tilde{c}_{\mathbf{n}} := \left[\sum_{k=0}^{K-1} \frac{1}{k!} \left(\frac{q_{\mathbf{n}}}{s} \right)^k \right]^s, \quad q_{\mathbf{n}} := \pi i \mathbf{n} \cdot \mathbf{\Omega} \mathbf{n}, \quad K, s \in \mathbb{N},$$

is an approximation of the true coefficient tensor $\underline{\mathbb{C}}$ with terms

$$c_{\mathbf{n}} = e^{q_{\mathbf{n}}} = [e^{q_{\mathbf{n}}/s}]^s = \left[\sum_{k=0}^{\infty} \frac{1}{k!} \left(\frac{q_{\mathbf{n}}}{s} \right)^k \right]^s,$$

in tensor-train format. The scaling and squaring approach is easy to implement and has been proven to work well for matrices [222]. It has been used for computing the

elementwise exponential of tensor-trains and other tensor formats [223], [224]. It is also the method implemented in the `tt_exp` function in the TT-toolbox [225].

In this section, we analyze the applicability of the scaling and squaring method for computing the tensor-train approximation of the tensor $\underline{\mathbf{C}}$ in (3.3). For all its merits we nevertheless show in this section that it is not suitable for our problem.

3.4.1. COMPLEXITY ANALYSIS FOR FIXED K AND s

We now provide a lower bound for the computational cost of evaluating $\tilde{\theta}(\mathbf{z} | \Omega)$ using the efficient inner product approach discussed above. That is, $\tilde{\theta}(\mathbf{z} | \Omega)$ is computed as the inner product between the tensor $\underline{\tilde{\mathbf{C}}}$ and the tensor with terms $e^{2\pi i \mathbf{n} \cdot \mathbf{z}}$ using the inner product algorithm in [218, Section 4.2] for tensor trains. (A full description is provided in Section 3.B.) We start by noticing that the tensor $\underline{\tilde{\mathbf{C}}}$ has the following special diagonal representation in the tensor train format.

Lemma 2. *Let $\hat{g} := \frac{g^2+g}{2}$, $R := \frac{\hat{g}^{K-1}}{\hat{g}-1}$, and $\hat{R} := R^s$. The terms $\tilde{\mathbf{c}}_{\mathbf{n}} = \mathbf{p}_{\mathbf{n}}^s$ of the tensor $\underline{\tilde{\mathbf{C}}}$ have the tensor train representation*

$$\tilde{\mathbf{c}}_{\mathbf{n}} = (\mathbf{P}_{n_1}^{(1)})^{\otimes s} (\mathbf{P}_{n_2}^{(2)})^{\otimes s} \dots (\mathbf{P}_{n_g}^{(g)})^{\otimes s},$$

where the inner cores are diagonal $\hat{R} \times \hat{R}$ matrices, and $\mathbf{A}^{\otimes s} := \mathbf{A} \otimes \dots \otimes \mathbf{A}$ (s times) denotes the s -fold Kronecker product of a matrix with itself.

Proof. See Section 3.B. \square

In the proof of the lemma, it is shown that the rank of the tensor $\underline{\mathbf{Q}}$ is not larger than R . Since $\text{rank}(\mathbf{A}^{\otimes s}) = \text{rank}(\mathbf{A})^s$, the rank of $\underline{\tilde{\mathbf{C}}}$ therefore cannot be larger than $\hat{R} = R^s$. In general, the rank of $\underline{\mathbf{Q}}$ is equal to R . In that case, the rank of $\underline{\tilde{\mathbf{C}}}$ will be equal to \hat{R} . Otherwise, \hat{R} provides an upper bound on the rank of $\underline{\tilde{\mathbf{C}}}$.

Proposition 5. *The computational cost of evaluating $\tilde{\theta}(\mathbf{z} | \Omega)$ using the standard tensor-train inner product algorithm [218, Section 4.2] applied to the diagonal representation in Lemma 1 is lower bounded by $\Omega((g-1)(2N+1)sR^s + (2N+1)R^s)$.*

Proof. See Section 3.B. \square

Note that this lower bound grows exponentially with s even for small parameter values such as $R = 2$ and $N = 1$.

3.4.2. COMPLEXITY ANALYSIS FOR A GIVEN ERROR BOUND

Scaling and squaring can be implemented in different ways, where the freedom mostly lies in the choice of K and s . These parameters should be chosen such that the approximation error is below a given bound. Here we consider a simple strategy for choosing K and s that is similar to the one used in older versions of MATLAB (with Padé approximations instead of Taylor expansions) [226]. The Taylor polynomial

$$P_K(x) = \sum_{k=0}^{K-1} \frac{x^k}{k!} \tag{3.14}$$

is a good approximation of $\exp(x)$ at $x = 0$. The worst case approximation error for $|x| < 1$ can be made arbitrarily small by increasing K . Once such a K is found, the argument z has to be rescaled such that $|z/s| < 1$. Since $e^z = [e^{z/s}]^s$, the terms of the coefficient tensor $\tilde{c}_{\mathbf{n}} = [P_K(q_{\mathbf{n}}/s)]^s$ approximate $c_{\mathbf{n}}$. The strategy for choosing the parameters therefore is as follows.

- Choose K large enough so that $|\exp(x) - P_K(x)| \leq \epsilon$ for all $|x| < 1$, where $\epsilon > 0$ is a error parameter [227].
- Choose s large enough so that $|z/s| < 1 \forall z \in Z \subset \mathbb{C}$.

The following proposition provides a lower bound on the computational complexity of the scaling and squaring method if K and s are chosen in this way.

Proposition 6. *If K and s are chosen as above and we have $\Re\{\Omega\} = \mathbf{0}_g$, then the numerical complexity of evaluating $\tilde{\theta}(\mathbf{z} \mid \Omega)$ using the standard tensor-train inner product algorithm applied to the diagonal representation in Lemma 1 is lower bounded by $o\left((g-1)(2N+1)\pi\lambda_{\min}gN^2R^{\pi\lambda_{\min}gN^2}\right)$ with λ_{\min} being the smallest eigenvalue of $\Im\{\Omega\}$.*

Proof. For the Riemann theta function we have $z = q_{\mathbf{n}} = \pi \mathbf{in} \cdot \Omega_{\mathbf{n}}$. For the index vector $\mathbf{n} = [N \ N \ \dots \ N]^T$, we have $z \leq -\pi\lambda_{\min}gN^2$. Thus, to have $|z/s| < 1$ we need $s \geq \pi\lambda_{\min}gN^2$. Hence, even for a fixed N , s would grow linearly in g . (Recall from Proposition 2 that N actually has to grow with g to keep the error of the truncated Riemann theta sum bounded.) Application of Proposition 3 with $s \geq \pi\lambda_{\min}gN^2$ now provides the following lower bound on the complexity, $o\left((g-1)(2N+1)\pi\lambda_{\min}gN^2R^{\pi\lambda_{\min}gN^2}\right)$. \square

The lower bound on the computational complexity thus grows exponentially in g even if there are lower and upper bounds on the eigenvalues of $\Im\{\Omega\}$ that are independent of g . Since we sum over $(2N+1)^g$ coefficients in the truncated Riemann theta sum, ϵ should actually decrease with g to keep the error introduced by the approximation of the coefficient tensor fixed. The constant K is inversely proportional to ϵ and hence K increases with g increasing the computational complexity even further. Thus, scaling and squaring is not suited for the numerical computation of high dimensional Riemann theta functions.

We finally remark that in practice, a rounding procedure is applied to the tensor-trains to reduce their ranks (like in the implementation `tt_exp` in the `TT-toolbox` [225]). We nevertheless observed in numerical experiments that even with rounding, scaling and squaring was ill-suited for our purposes.

3.5. NUMERICAL EXPERIMENTS

In this section, several methods for computing truncated Riemann functions (3.2) are investigated numerically and compared with respect to accuracy, for both low and high genus cases. First, the choice of algorithms is motivated. Then, the background of the numerical experiments is discussed. Finally, the results are presented.

3.5.1. CHOICE OF BENCHMARK ALGORITHMS

We so far considered two different approaches to approximate the Riemann theta function (3.1). The first approach is the naive computation of the truncated Riemann theta function (3.2) over the index sets $\mathcal{N}^g(N) = \mathcal{I}^g(p, N)$ and $\mathcal{H}^g(N)$, that were introduced in Section 3.2. Even for $N = 2$, the number of elements in $\mathcal{I}^g(p, N)$ grows exponentially in the genus g for any p . Hence, naive evaluation of (3.1) can be applied when the genus g is small for these index sets, but it is not well-suited for computing high-genus solutions. For the index set $\mathcal{I}^g(\infty, N)$, we will refer to this approach as the Hypercube algorithm. When the Riemann matrix is diagonal, the multi-dimensional Riemann theta function can be computed efficiently using multiple one-dimensional Riemann theta functions. We will refer to the algorithm that neglects the off-diagonal elements in order to exploit this fact to facilitate fast computation as `Hypercube_diag_approx` in the following. This algorithm will allow us to verify that the Riemann matrices used in the examples cannot be approximated well with just the diagonal part of the matrix. We furthermore know from Lemma 1 that the complexity of naive computation over the hyperbolic cross $\mathcal{N}^g(N) = \mathcal{H}^g(N)$ grows at a slower rate. Therefore, it might be better suited for high genus cases. We will refer to this algorithm as `Hyperbolic_cross` in the following.

The second approach to approximate the Riemann theta function is to replace the true coefficient tensor in the truncated Riemann function (3.2) with an approximation in the tensor-train format. The resulting approximation (3.13) of the Riemann theta function can then be evaluated with low numerical complexity since it is an inner product between low-rank tensor-trains. There are different ways how the tensor-train approximation of the coefficient tensor can be obtained. In the previous section, we showed that the popular scaling and squaring approach is not a good choice when then the genus g is high. However, there are other methods for this task. The first method we consider is from the paper [220] and will be referred to as `TT_cross`. Specifically, we employ the `dmrg_cross` routine from the TT-toolbox [225]. We also investigate the performance of the `funcrs` routine from the toolbox, for which unfortunately no reference is provided. This algorithm will be referred to as `TT_funcrs` in the following.

3.5.2. GENERATION OF TEST DATA

Riemann theta functions play, as was already mentioned in the introduction, a fundamental role in the area of nonlinear Fourier analysis [39]. The (quasi-)periodic solutions of many integrable systems can be approximated arbitrarily well using so-called hyperelliptic (or also finite-gap, finite-genus or finite-band) solutions, which have especially simple closed-form representations that involve the Riemann theta function. For the numerical demonstrations we make use of the hyperelliptic solutions of the normalized KdV equation

$$u_t + u_{xxx} + 6uu_x = 0 \quad (3.15)$$

and the normalized NLS equation

$$iq_x + q_{tt} + 2|q|^2q = 0. \quad (3.16)$$

The hyperelliptic solutions of the KdV equation are given in terms of the Riemann theta function as [153, Eq. (19)]

$$u(x, t) = 2 \frac{\partial^2}{\partial x^2} \log \theta(\mathbf{z} | \mathbf{\Omega}), \quad \mathbf{z} = \mathbf{k}x - \boldsymbol{\omega}t + \boldsymbol{\phi}, \quad (3.17)$$

where $\mathbf{\Omega}$ is typically called the period matrix in the literature, and \mathbf{k} , $\boldsymbol{\omega}$ and $\boldsymbol{\phi}$ are constant real vectors. The matrix $\mathbf{\Omega}$ is of size $g \times g$, where g is as before called the genus. The parameters $\mathbf{\Omega}$, \mathbf{k} , $\boldsymbol{\omega}$ and $\boldsymbol{\phi}$ cannot be chosen freely. The theory of integrable systems shows that the parameters provide a valid solution if and only they can be derived from a so-called hyperelliptic Riemann surface, which is a special kind of one-dimensional complex manifold. In our examples, we computed the parameters numerically using the methods given in [153, Section 3.2], [30, Section 14.4]. The hyperelliptic Riemann surfaces that we used are specified by their so-called branch points, which we provide for each example.

The finite-genus solutions of the NLS equation are given by

$$q(x, t) = K_0 \frac{\theta\left(\frac{\mathbf{z}_-}{2\pi} | \mathbf{\Omega}\right)}{\theta\left(\frac{\mathbf{z}_+}{2\pi} | \mathbf{\Omega}\right)} e^{i\omega_0 t + ik_0 z}, \quad \mathbf{z}_{\pm} = \mathbf{k}x + \boldsymbol{\omega}t + \delta_{\pm}, \quad (3.18)$$

where $\mathbf{\Omega}$, \mathbf{k} , $\boldsymbol{\omega}$, δ_{\pm} , k_0 , ω_0 and K_0 are again constant parameters [111]. The vectors δ_{\pm} can be complex valued and hence arguments to the Riemann theta function \mathbf{z}_{\pm} can be complex valued. As before, these parameters have to be obtained from a hyperelliptic Riemann surface, which is specified by branch points. In our test we used the period matrix and parameter vectors provided in [111, Table II].

In the following we will look at three scenarios. In the first case we will assess the accuracy of the algorithms for a genus-2 and a genus-6 solution of the KdV equation. In the second case we will test the accuracy in computing a genus-3 solution of the NLS equation for which the Riemann theta function has complex arguments. In the third case, we compute solutions of the KdV equation up to the very high genus of 60. To the best of our knowledge, a successful computation of non-trivial Riemann theta functions for genera this large has never been reported in the literature before.

3.5.3. ACCURACY FOR GENUS-2 AND GENUS-6 KdV SOLUTIONS

In this scenario, we evaluate the accuracy of computing the Riemann theta function for $\mathbf{z} \in \mathbb{R}^g$ with \mathbf{z} as defined in (3.17). The period matrices $\mathbf{\Omega}$ were derived from the finite-genus KdV solutions in [153, Section 4]. First we have the genus-2 solution of the KdV equation with branch points [0 0.5 1 1.5 5]. We compute the Riemann theta function for 16384 arguments \mathbf{z} corresponding to the grid formed by 128 equispaced values of x in [0, 4] and 128 equispaced values of t in [-0.5, 0.5]. Secondly we have the genus-6 solution with branch points [0 0.5 2 2.5 4 4.5 6 6.5 8 8.5 10 10.5 12]. We again compute the Riemann theta function for 16384 arguments \mathbf{z} corresponding to the grid formed by 128 equispaced values of x in [0, 3] and 128 equispaced values of t in [-0.3, 0.3]. The values of x and t chosen are sufficiently representative for the respective solutions. The phases $\boldsymbol{\phi}$ are set to 0. We then define the point-wise error as

$$E(\mathbf{z}) = \left| \hat{\theta}(\mathbf{z} | \mathbf{\Omega}) - \tilde{\theta}(\mathbf{z} | \mathbf{\Omega}) \right| \quad (3.19)$$

where $\hat{\theta}(\mathbf{z} | \boldsymbol{\Omega})$ is the reference value and $\tilde{\theta}(\mathbf{z} | \boldsymbol{\Omega})$ is the value computed by the other methods. The reference value $\hat{\theta}(\mathbf{z} | \boldsymbol{\Omega})$ is computed using the Hypercube algorithm. The truncation parameter for each example is fixed by starting with $N = 1$ and increasing it until

$$\max_{\mathbf{z}} |\hat{\theta}(\mathbf{z} | \boldsymbol{\Omega})|_{N+1} - \hat{\theta}(\mathbf{z} | \boldsymbol{\Omega})|_N| < 1 \times 10^{-14}, \quad (3.20)$$

where $\hat{\theta}(\mathbf{z} | \boldsymbol{\Omega})|_N$ is the Riemann theta function calculated by the Hypercube algorithm with the truncation parameter N . If equation (3.20) is satisfied for $N = \hat{N}$, then the values $\hat{\theta}(\mathbf{z} | \boldsymbol{\Omega})|_{\hat{N}+1}$ are set as the reference values in (3.19). Note that we can compute the solution classically only because the genera are low in this example. For the algorithms TT_cross and TT_funcrs, we set the accuracy parameter of the coefficient tensor $\underline{\mathbf{C}}$ approximation to 10^{-12} .

In Table 3.1 we see the median and maximum of the error $E(\mathbf{z})$, the truncation parameter N and the number of terms in the summation used for the genus-2 solution. As the coefficient tensor approximation step in the TT_cross and TT_funcrs algorithms is non-deterministic, we report the errors for both the algorithms over 20 runs. For the TT_cross and TT_funcrs algorithms, the number of terms corresponds to the number of non-zero terms in the tensor-train approximation of the coefficient tensor $\tilde{\underline{\mathbf{C}}}$. The * symbol over the number indicates that the value listed is the median value over 20 runs. Correspondingly, we report the maximum, median and minimum over the 20 runs of the median and maximum of the pointwise error $E(\mathbf{z})$. For the Hypercube_diag_approx algorithm we use the same truncation parameter as the reference Hypercube algorithm. For the Hyperbolic_cross algorithm we initially choose the truncation parameter N_{HC} such that the number of terms in the sum is close to that of the Hypercube algorithm. We also run the Hyperbolic_cross algorithm with approximately 50%, 10% and 1% of N_{HC} as the truncation parameter. The errors for the Hypercube_diag_approx algorithm are high indicating that the period matrix has significant non-diagonal components. Both the TT_cross and TT_funcrs algorithms have very low errors and the variation over the multiple runs is very small. The error for the Hyperbolic_cross algorithm with N_{HC} as the truncation parameter is also low. However, it is higher than that of the tensor-train based methods. The tensor-train methods on the other hand use more terms. The errors of the Hyperbolic_cross algorithm increase slowly with decreasing truncation parameter.

In Table 3.2 we again show the errors for the genus-2 case. The only difference being the application of the Siegel transform 3.2.2 in all the algorithms. The process to calculate the truncation parameter is repeated as described previously. For this example the Siegel transform does not lead to a reduction in the truncation parameter. The main differences in between Tables 3.1 and 3.2 are the maximum errors for the tensor-train based methods. The maximum errors are significantly higher when the Siegel transform is applied. The errors for the other algorithms have similar magnitudes.

In Tables 3.3 and Table 3.4 we have the results for the genus-6 example. Without the Siegel transform the truncation parameter was fixed to 5. The Hypercube_diag_approx algorithm has high errors indicating the presence of significant non-diagonal terms. Both TT_cross and TT_funcrs algorithms have low errors while using significantly less number of terms compared to the classical Hypercube algorithm. The Hyperbolic_cross algorithm also performs well but is less accurate when compared to the tensor-train

based methods for the same number of terms. Application of the Siegel transform helps reduce the truncation parameter to 4 in Table 3.4. Both the `TT_cross` and `TT_funcrs` algorithms fail for this genus-6 example when the Siegel transform is applied. We attribute this to numerical ill-conditioning arising from the faster decay of the coefficients c_n . It seems to become harder to approximate the tensor \tilde{C} with a tensor-train as the range of magnitudes of the coefficients c_n increases. For the `Hyperbolic_cross` algorithm, the maximum errors are slightly higher however the median errors are lower when the Siegel transform is applied.

The number of terms listed in Tables 3.1-3.4 indicate the memory requirements of the algorithms and is also related to the computational complexity. The total computation cost for the `Hyperbolic_cross` algorithm is the sum of the cost for generating the index set \mathcal{H}^g and the sum (3.2) over \mathcal{H}^g . Even with the recursive algorithm from [214, Section 2.6], the cost of generating the index set grows quickly with N and g . The cost of the summation depends on the specific implementation and can be made quite efficient using parallelized implementations. For the `TT_cross` and `TT_funcrs` algorithms the computation cost is divided into the cost of computing the tensor-train approximations and the cost of computing the inner-product. Given the rank of the tensor-train approximation of \tilde{C} , the computation cost of the inner-product can be estimated (see Section 3.B or [218, Section 4.2]). The cost of computing the tensor-train approximation \tilde{C} is however non-trivial due to the iterative nature of both the `dmrg_cross` and `funcrs` routines. The actual complexity of all three algorithms depends significantly on the specific implementations. Hence, we have chosen to avoid a detailed comparison of the computational complexity and instead provided only the number of terms as an indicator. Furthermore, they are also an indicator for the memory requirements of the algorithms, which in our experience has been the major limiting factor for larger genera.

3.5.4. ACCURACY FOR GENUS-3 NLS SOLUTION

The solution of the NLS equation is given as the ratio of Riemann theta function values (3.18). We use the genus-3 example from [111, Figure 3]. We did not have to compute the period matrix and parameter vectors from the branch points in this case since they are provided in [111, Table II]. The theta function values are computed for 16384 arguments \mathbf{z}_\pm as defined in (3.18) for 128 equispaced points $x \in [-0.002, 0.002]$ and for 128 equispaced points $t \in [-0.08, 0.08]$. For both the Riemann theta functions in the numerator and denominator of (3.18), the truncation parameter was fixed to be 6 using the procedure described for the KdV solutions in the previous subsection. The tensor-train methods were run 20 times and the truncation parameter for the `Hyperbolic_cross` algorithm was set to have similar number of terms as the `Hypercube` algorithm.

In Tables 3.5 and 3.6, we list the errors in computing the Riemann theta function with arguments \mathbf{z}_- and the Riemann theta function with arguments \mathbf{z}_+ . In this example the arguments \mathbf{z}_- are complex valued while the arguments \mathbf{z}_+ are real valued. Both the tensor-train based algorithms `TT_cross` and `TT_funcrs` fail to compute the Riemann theta function values correctly. We suspect the reason to be the same ill-conditioning that we observed for the genus-6 KdV solution in the previous subsection. The `Hyperbolic_cross` algorithm appears to be moderately accurate even for lower number of terms. Application of the Siegel transform did not lead to a reduction in the

Table 3.1: Error in computing $\theta(\mathbf{z} | \mathbf{\Omega})$ for a genus-2 solution of the KdV equation

Algorithm	N	#terms	$E_1 = \max_{\mathbf{z}}\{E(\mathbf{z})\}$	$E_2 = \text{median}_{\mathbf{z}}\{E(\mathbf{z})\}$
Hypercube	4	81	0 by definition	0 by definition
TT_funcrs	4	125*	$\max\{E_1\}$ over 20 runs: 7.3286×10^{-13} $\text{median}\{E_1\}$ over 20 runs: 7.3286×10^{-13} $\min\{E_1\}$ over 20 runs: 7.3286×10^{-13}	$\max\{E_2\}$ over 20 runs: 5.1681×10^{-13} $\text{median}\{E_2\}$ over 20 runs: 5.1681×10^{-13} $\min\{E_2\}$ over 20 runs: 5.1681×10^{-13}
TT_cross	4	101*	$\max\{E_1\}$ over 20 runs: 7.3319×10^{-13} $\text{median}\{E_1\}$ over 20 runs: 7.3313×10^{-13} $\min\{E_1\}$ over 20 runs: 7.3308×10^{-13}	$\max\{E_2\}$ over 20 runs: 5.1692×10^{-13} $\text{median}\{E_2\}$ over 20 runs: 5.1686×10^{-13} $\min\{E_2\}$ over 20 runs: 5.1675×10^{-13}
Hypercube_diag_approx	4	81	8.3307×10^{-2}	5.3869×10^{-2}
Hyperbolic_cross	20	81	2.4035×10^{-9}	1.5476×10^{-9}
Hyperbolic_cross	10	33	6.2295×10^{-6}	4.3971×10^{-6}
Hyperbolic_cross	2	5	8.5271×10^{-2}	5.9622×10^{-2}
Hyperbolic_cross	1	2	3.3724×10^{-1}	1.1661×10^{-1}

Table 3.2: Error in computing $\theta(\mathbf{z} | \mathbf{Q})$ for a genus-2 solution of the KdV equation (with Siegel transform)

Algorithm	N	#terms	$E_1 = \max_{\mathbf{z}}\{E(\mathbf{z})\}$	$E_2 = \text{median}_{\mathbf{z}}\{E(\mathbf{z})\}$
Hypercube	4	81	1.7763×10^{-15}	2.2216×10^{-16}
TT_funcrs	4	124*	$\max\{E_1\}$ over 20 runs: 1.4664×10^{-5} $\text{median}\{E_1\}$ over 20 runs: 1.4664×10^{-5} $\min\{E_1\}$ over 20 runs: 1.4664×10^{-5}	$\max\{E_2\}$ over 20 runs: 6.6613×10^{-15} $\text{median}\{E_2\}$ over 20 runs: 6.6613×10^{-15} $\min\{E_2\}$ over 20 runs: 6.6613×10^{-15}
TT_cross	4	104*	$\max\{E_1\}$ over 20 runs: 2.9409×10^{-5} $\text{median}\{E_1\}$ over 20 runs: 2.9409×10^{-5} $\min\{E_1\}$ over 20 runs: 2.9409×10^{-5}	$\max\{E_2\}$ over 20 runs: 3.5527×10^{-15} $\text{median}\{E_2\}$ over 20 runs: 3.5527×10^{-15} $\min\{E_2\}$ over 20 runs: 3.5527×10^{-15}
Hypercube_diag_approx	4	81	2.2379	4.3191×10^{-2}
Hyperbolic_cross	20	81	5.5177×10^{-10}	5.1231×10^{-13}
Hyperbolic_cross	10	33	1.6653×10^{-5}	2.9255×10^{-8}
Hyperbolic_cross	2	5	3.5881×10^{-1}	2.5139×10^{-2}
Hyperbolic_cross	1	2	7.7519×10^{-1}	2.7817×10^{-1}

Table 3.3: Error in computing $\theta(\mathbf{z} | \mathbf{\Omega})$ for a genus-6 solution of the KdV equation

Algorithm	N	#terms	$E_1 = \max_{\mathbf{z}}\{E(\mathbf{z})\}$	$E_2 = \text{median}_{\mathbf{z}}\{E(\mathbf{z})\}$
Hypercube	5	1771561	0 by definition	0 by definition
TT_funcrs	5	69005*	max $\{E_1\}$ over 20 runs: 1.3615×10^{-11} median $\{E_1\}$ over 20 runs: 7.8933×10^{-12} min $\{E_1\}$ over 20 runs: 3.6349×10^{-12}	max $\{E_2\}$ over 20 runs: 1.8069×10^{-12} median $\{E_2\}$ over 20 runs: 1.2623×10^{-12} min $\{E_2\}$ over 20 runs: 6.4602×10^{-13}
TT_cross	5	37904*	max $\{E_1\}$ over 20 runs: 7.8628×10^{-9} median $\{E_1\}$ over 20 runs: 1.4163×10^{-9} min $\{E_1\}$ over 20 runs: 1.2142×10^{-9}	max $\{E_2\}$ over 20 runs: 8.7031×10^{-10} median $\{E_2\}$ over 20 runs: 3.4114×10^{-10} min $\{E_2\}$ over 20 runs: 3.1115×10^{-10}
Hypercube_diag_approx	5	1771561	1.027 10	1.1549×10^{-1}
Hyperbolic_cross	1430	1753893	8.2058×10^{-11}	3.9893×10^{-11}
Hyperbolic_cross	715	615521	5.9266×10^{-9}	2.3027×10^{-9}
Hyperbolic_cross	143	47353	8.3491×10^{-6}	1.2015×10^{-6}
Hyperbolic_cross	15	545	1.3741×10^{-1}	1.0142×10^{-2}

Table 3.4: Error in computing $\theta(\mathbf{z} | \mathbf{Q})$ for a genus-6 solution of the KdV equation (with Siegel transform)

Algorithm	N	#terms	$E_1 = \max_{\mathbf{z}}\{E(\mathbf{z})\}$	$E_2 = \text{median}_{\mathbf{z}}\{E(\mathbf{z})\}$
Hypercube	4	531441	3.7750×10^{-14}	5.8950×10^{-15}
TT_funcrs	4	18138*	$\max\{E_1\}$ over 20 runs: 9.7672×10^{12} $\text{median}\{E_1\}$ over 20 runs: 1.7581×10^{12} $\min\{E_1\}$ over 20 runs: 6.0793×10^{11}	$\max\{E_2\}$ over 20 runs: 1.4351 $\text{median}\{E_2\}$ over 20 runs: 7.3642×10^{-1} $\min\{E_2\}$ over 20 runs: 2.3702×10^{-1}
TT_cross	4	13494*	$\max\{E_1\}$ over 20 runs: 1.3213×10^{13} $\text{median}\{E_1\}$ over 20 runs: 5.6749×10^{11} $\min\{E_1\}$ over 20 runs: 1.0188×10^8	$\max\{E_2\}$ over 20 runs: 5.6102×10^{-1} $\text{median}\{E_2\}$ over 20 runs: 1.0112×10^{-1} $\min\{E_2\}$ over 20 runs: 8.6124×10^{-5}
Hypercube_diag_approx	4	531441	1.2994	5.0057×10^{-1}
Hyperbolic_cross	640	549113	2.3065×10^{-7}	4.0635×10^{-14}
Hyperbolic_cross	320	188993	1.6138×10^{-5}	3.5885×10^{-11}
Hyperbolic_cross	64	15241	6.15386×10^{-3}	2.4742×10^{-6}
Hyperbolic_cross	7	97	1.0231	1.6241×10^{-1}

truncation parameter. Hence we have not mentioned specific error values for the same. From the results for the NLS example and the KdV examples in the previous subsection, the `Hyperbolic_cross` algorithm emerges as a practical algorithm for computing the Riemann theta function.

3.5.5. COMPUTING HIGH GENUS KdV SOLUTIONS

In this subsection we will test the ability of the `Hyperbolic_cross` algorithm to compute the Riemann theta function value for high number of dimensions. Unfortunately, both the `TT_funcrs` and `TT_cross` algorithms were limited by memory even for moderate number of dimensions. Hence we only test the `Hyperbolic_cross` algorithm in the following. As there are no non-trivial Riemann matrices for which the Riemann theta function value is known analytically and using the Hypercube algorithm is not feasible, we resort to an alternative approach to verify the correctness of the approximations. We use the hyperelliptic solutions of the KdV equations with the branch points $\lambda_j = 0.5j$, $j = 0, 1, \dots, g$ for different values of the genus g . We would like to remark that to compute the period matrices $\mathbf{\Omega}$ reliably we had to use 1000 bits of precision. We accomplished this using the Julia programming language. For a given g and N the Riemann theta function value is computed for $x_j = -2.0518 + 0.0120j$, $j = 0, 1, \dots, 340$ and $t = [-0.2460 \ -0.2457 \ -0.2455]$. The approximate solution $\tilde{u}(x, t)$ is computed using central-difference to calculate the derivatives in (3.17).

To quantify the error in the calculated solutions, we first compute approximations of the time derivative u_t and space derivatives u_x and u_{xxx} using central differences. We then calculate the relative error

$$E_r = \frac{\sqrt{\sum_i |\text{LHS}_i - \text{RHS}_i|^2}}{\sqrt{\sum_i 0.25(|\text{LHS}_i| + |\text{RHS}_i|)^2}}, \quad (3.21)$$

where $\text{LHS}_i = -u_t(x_i, t)$ and $\text{RHS}_i = u_{xxx}(x_i, t) + 6u(x_i, t)u_x(x_i, t)$. Due to the absence of the true values of LHS_i or RHS_i , we have used the mean value $0.5(|\text{LHS}_i| + |\text{RHS}_i|)$ as the reference value in the relative error. We discard the values at the boundaries for which the numerical derivative cannot be calculated correctly using central-difference. As an example, for the solution computed using the `Hyperbolic_cross` algorithm for $g = 30$ and $N = 4$, we plot $-u_t$ and $u_{xxx} + 6uu_x$ in Figure 3.1. We can see that the lines almost overlap which indicates that the error is small and that the computed values do correspond to a solution of the KdV equation. In Figure 3.2 we show the error E_r for the `Hyperbolic_cross` algorithm for varying values of g and N . The choices of g and N were limited by the available system memory. We can observe that the relative error is small even for high genus solutions. From our understanding, this is the first instance in literature where such high genus solutions have been computed.

We can thus surmise that the `Hyperbolic_cross` algorithm is suited for computing remarkably high genus Riemann theta function with moderate accuracy. The tensor-train based algorithms `TT_funcrs` and `TT_cross` work well for low genus but do not scale well with the number of dimensions. We would like to remark that for both the tensor-train based algorithms and the `Hyperbolic_cross` algorithm, the truncation parameter N can be chosen independently for each dimension to further reduce the computation cost. The impact of such a choice would be a topic for future research.

Table 3.5: Error in computing $\theta\left(\frac{z}{z_0} \mid \mathbf{Q}\right)$ for a genus-3 solution of the NLS equation

Algorithm	N	#terms	$E_1 = \max_{\mathbf{z}}\{E(\mathbf{z})\}$	$E_2 = \text{median}_{\mathbf{z}}\{E(\mathbf{z})\}$
Hypercube	6	2197	0 by definition	0 by definition
TT_funcrs	6	1918*	$\max\{E_1\}$ over 20 runs: 1.4971 $\text{median}\{E_1\}$ over 20 runs: 3.9583×10^{-1} $\min\{E_1\}$ over 20 runs: 2.0053×10^{-3}	$\max\{E_2\}$ over 20 runs: 1.3031 $\text{median}\{E_2\}$ over 20 runs: 3.5048×10^{-1} $\min\{E_2\}$ over 20 runs: 1.35539×10^{-3}
TT_cross	6	1220*	$\max\{E_1\}$ over 20 runs: 4.4819×10^{12} $\text{median}\{E_1\}$ over 20 runs: 2.1824×10^7 $\min\{E_1\}$ over 20 runs: 2.0406×10^{-1}	$\max\{E_2\}$ over 20 runs: 9.9790×10^{11} $\text{median}\{E_2\}$ over 20 runs: 2.0426×10^7 $\min\{E_2\}$ over 20 runs: 1.7638×10^{-1}
Hypercube_diag_approx	6	2197	1.5084	1.1001
Hyperbolic_cross	112	2237	3.7768×10^{-11}	2.9799×10^{-11}
Hyperbolic_cross	56	885	9.3304×10^{-6}	9.3077×10^{-6}
Hyperbolic_cross	12	105	3.4411×10^{-2}	3.0213×10^{-2}
Hyperbolic_cross	2	7	1.5242	1.0956

Table 3.6: Error in computing $\theta(\frac{z}{z^*} | \mathbf{\Omega})$ for a genus-3 solution of the NLS equation

Algorithm	N	#terms	$E_1 = \max_{\mathbf{z}}\{E(\mathbf{z})\}$	$E_2 = \text{median}_{\mathbf{z}}\{E(\mathbf{z})\}$
Hypercube	6	2197	0 by definition	0 by definition
TT_funcrs	6	1918*	max $\{E_1\}$ over 20 runs: 1.4971 median $\{E_1\}$ over 20 runs: 3.9583×10^{-1} min $\{E_1\}$ over 20 runs: 2.0053×10^{-3}	max $\{E_2\}$ over 20 runs: 1.3031 median $\{E_2\}$ over 20 runs: 3.5048×10^{-1} min $\{E_2\}$ over 20 runs: 1.35539×10^{-3}
TT_cross	6	1220*	max $\{E_1\}$ over 20 runs: 4.4819×10^{12} median $\{E_1\}$ over 20 runs: 2.1824×10^7 min $\{E_1\}$ over 20 runs: 2.0406×10^{-1}	max $\{E_2\}$ over 20 runs: 9.9790×10^{11} median $\{E_2\}$ over 20 runs: 2.0426×10^7 min $\{E_2\}$ over 20 runs: 1.7638×10^{-1}
Hypercube_diag_approx	6	2197	9.1129×10^{-2}	3.2034×10^{-2}
Hyperbolic_cross	112	2237	4.8849×10^{-15}	1.3323×10^{-15}
Hyperbolic_cross	56	885	8.9268×10^{-10}	3.9137×10^{-10}
Hyperbolic_cross	12	105	3.6694×10^{-4}	2.5105×10^{-4}
Hyperbolic_cross	2	7	9.1123×10^{-2}	3.2032×10^{-2}

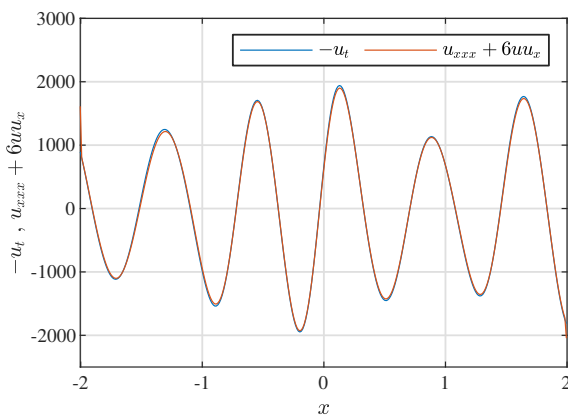


Figure 3.1: Checking the genus-30 solution computed using the `Hyperbolic_cross` algorithm with $N = 4$. We can see that $-u_t \approx u_{xxx} + 6uu_x$ verifying that $u(x, t)$ is a good approximation of the finite-gap KdV solution.

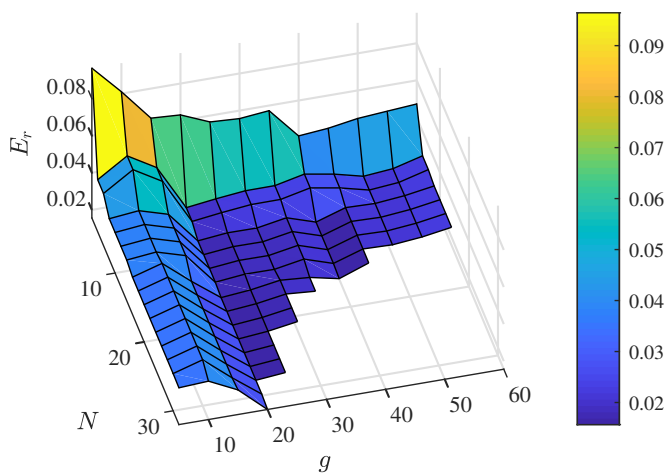


Figure 3.2: The error E_r for KdV solutions computed using the `Hyperbolic_cross` algorithm for varying values of g and N .

3.6. CONCLUSION

The Riemann theta function plays a crucial role in the nonlinear Fourier analysis of signals in fields such as fiber-optic communications and coastal engineering. It is used to synthesize periodic signals through the inverse nonlinear Fourier transforms. Numerical computation of the Riemann theta function as a multi-dimensional Fourier series is challenging due to the *curse of dimensionality*. This significantly limits the practical applicability despite much interest. To better understand the limitations, we derived some lower bounds and an upper bound on the series truncation error for certain index sets in Section 3.3. We investigated a tensor-train method to compute the function which utilizes the scaling and squaring approach for computing the exponential. We theoretically proved in Section 3.4 that such a tensor-train based approach cannot break the curse of dimensionality. Following that we proposed to exploit two other tensor-train based algorithms and another algorithm based on the hyperbolic cross index set. Using hyperelliptic solutions of the KdV and NLS equations as numerical examples, in Section 3.5 we showed that while the two tensor-train based algorithms work for low genus examples with real arguments, they are prone to numerical ill-conditioning. Their memory requirement is a limiting factor for high genera. While the algorithm based on the hyperbolic-cross index set can also achieve high accuracy for the low genus solutions, its novelty is the ability to compute moderately accurate solutions of high genera (up to 60) with relatively low computational cost. It therefore enables the computation of high dimensional inverse nonlinear Fourier transforms that were so far impractical. Similar algorithms based on related yet more general index sets such as the weighted Zaremba cross [214] may provide further reduction in the computation cost of high genus solutions.

3.A. PROOF OF THEOREM 1

Lemma 3. *Let $p \in \mathbb{N} \cup \{\infty\}$ and $\bar{N} \in \mathbb{N}$. Then*

$$\mathbf{n} \notin \mathcal{J}^g(p, \bar{N}) \Rightarrow |c_{\mathbf{n}}| \leq \exp\left(-\pi \frac{(\bar{N} + 1)^2}{\|\Im\{\mathbf{\Omega}\}^{-1}\|_p}\right). \quad (3.22)$$

Proof. The matrix lower bound $\ell(\mathbf{A})$ of a matrix \mathbf{A} is the smallest number m such that $m\|\mathbf{x}\|_p \leq \|\mathbf{y}\|_p$ whenever $\mathbf{y} = \mathbf{A}\mathbf{x}$ [228]. Since $\Im\{\mathbf{\Omega}\}$ is symmetric and positive definite,

$$\begin{aligned} \min_{\mathbf{n} \in \mathcal{J}^g(p, \bar{N})} \mathbf{n} \cdot \Im\{\mathbf{\Omega}\} \mathbf{n} &= \min_{\mathbf{n} \in \mathcal{J}^g(p, \bar{N})} \|\mathbf{n} \cdot \Im\{\mathbf{\Omega}\} \mathbf{n}\|_p \\ &\geq \min_{\mathbf{n} \in \mathcal{J}^g(p, \bar{N})} \ell(\mathbf{n}^T \Im\{\mathbf{\Omega}\}) \|\mathbf{n}\|_p \\ &\geq \min_{\mathbf{n} \in \mathcal{J}^g(p, \bar{N})} \ell(\mathbf{n}^T) \ell(\Im\{\mathbf{\Omega}\}) \|\mathbf{n}\|_p \\ &\geq \min_{\mathbf{n} \in \mathcal{J}^g(p, \bar{N})} \left\| \frac{\mathbf{n}}{\mathbf{n} \cdot \mathbf{n}} \right\|_p^{-1} \|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1} \|\mathbf{n}\|_p \\ &= \min_{\mathbf{n} \in \mathcal{J}^g(p, \bar{N})} \|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1} \mathbf{n} \cdot \mathbf{n} \\ &\geq \|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1} (\bar{N} + 1)^2, \end{aligned}$$

where we used [228, Lem. 4.4] in the third step, and [228, Lem. 2.2] in the fourth step (twice). Hence,

$$\begin{aligned}
 |c_{\mathbf{n}}| &= |\exp(\pi i \mathbf{n} \cdot \mathbf{\Omega} \mathbf{n})| \\
 &= |\exp(\pi i \mathbf{n} \cdot [\Re\{\mathbf{\Omega}\} + i\Im\{\mathbf{\Omega}\}]\mathbf{n})| \\
 &= \underbrace{|\exp(\pi i \mathbf{n} \cdot \Re\{\mathbf{\Omega}\}\mathbf{n})|}_{=1} \exp(-\underbrace{\pi \mathbf{n} \cdot \Im\{\mathbf{\Omega}\}\mathbf{n}}_{>0}) \\
 &\leq \exp(-\pi \|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1} (\bar{N} + 1)^2). \tag{3.23}
 \end{aligned}$$

□

The following results is well-known, but we present an explicit proof for the convenience of the reader. The explicit bounds given in the proof will furthermore be useful later.

Proof of Theorem 1 for $p = \infty$. Let us look at the sets $\mathcal{I}^g(\infty, N + k) \setminus \mathcal{I}^g(\infty, N + k - 1)$, where $k \in \mathbb{N}$. The number of elements satisfy

$$\begin{aligned}
 &\#\mathcal{I}^g(\infty, N + k) \setminus \mathcal{I}^g(\infty, N + k - 1) \\
 &= (2(N + k) + 1)^g - (2(N + k - 1) + 1)^g \\
 &< (2(N + k) + 1)^g, \tag{3.24}
 \end{aligned}$$

Using Lemma 3 with $\bar{N} = N + k - 1$ shows that

$$\begin{aligned}
 &\mathbf{n} \in \mathcal{I}^g(\infty, N + k) \setminus \mathcal{I}^g(\infty, N + k - 1) \\
 &\Rightarrow \mathbf{n} \notin \mathcal{I}^g(\infty, N + k - 1) \\
 &\Rightarrow |c_{\mathbf{n}}| \stackrel{(3.22)}{\leq} \exp(-\pi \|\Im\{\mathbf{\Omega}\}^{-1}\|_p^{-1} (N + k)^2). \tag{3.25}
 \end{aligned}$$

Note that for any column vector \mathbf{x} , the matrix norm $\|\mathbf{x}^T\|_\infty$ is equal to the vector norm $\|\mathbf{x}\|_1$. We thus have

$$\begin{aligned}
 |e^{2\pi i \mathbf{n} \cdot \mathbf{z}}| &= e^{-2\pi \mathbf{n}^T \Im\{\mathbf{z}\}} \leq e^{2\pi \|\mathbf{n}^T\|_\infty \|\Im\{\mathbf{z}\}\|_\infty} = e^{2\pi \|\mathbf{n}\|_1 \|\Im\{\mathbf{z}\}\|_\infty} \\
 &\leq e^{2\pi g(N+k) \|\Im\{\mathbf{z}\}\|_\infty}, \quad \forall \mathbf{n} \text{ as in (3.25)}.
 \end{aligned}$$

With these results, we find that

$$\begin{aligned}
\max_{\mathbf{z} \in Z} |\theta(\mathbf{z} | \mathbf{\Omega}) - \hat{\theta}(\mathbf{z} | \mathbf{\Omega})| &\leq \sum_{\mathbf{n} \notin \mathcal{J}^g(\infty, N)} |e^{2\pi i \mathbf{n} \cdot \mathbf{z}}| |c_{\mathbf{n}}| \\
&= \sum_{k=1}^{\infty} \sum_{\mathbf{n} \in \mathcal{J}^g(\infty, N+k) \setminus \mathcal{J}^g(\infty, N+k-1)} |e^{2\pi i \mathbf{n} \cdot \mathbf{z}}| |c_{\mathbf{n}}| \\
&\leq \sum_{k=1}^{\infty} (2(N+k)+1)^g e^{2\pi \|\mathfrak{I}\{\mathbf{z}\}\|_{\infty} g(N+k) - \pi \|\mathfrak{I}\{\mathbf{\Omega}\}^{-1}\|_p^{-1} (N+k)^2} \\
&= \sum_{k=N+1}^{\infty} (2k+1)^g e^{2\pi \|\mathfrak{I}\{\mathbf{z}\}\|_{\infty} g k - \pi \|\mathfrak{I}\{\mathbf{\Omega}\}^{-1}\|_p^{-1} k^2} \\
&= \sum_{k=N+1}^{\infty} e^{g \log(2k+1) + 2\pi \|\mathfrak{I}\{\mathbf{z}\}\|_{\infty} g k - \pi \|\mathfrak{I}\{\mathbf{\Omega}\}^{-1}\|_p^{-1} k^2} \\
&\rightarrow 0, \quad \text{for } N \rightarrow \infty.
\end{aligned}$$

□

3.B. PROOF OF LEMMA 2 AND PROPOSITION 5

Lemma 4. *The quadratic form tensor*

$$\underline{Q}(n_1, n_2, \dots, n_g) := \pi \mathbf{i} \mathbf{n} \cdot \mathbf{\Omega} \mathbf{n}, \quad \mathbf{n} \in \mathcal{J}^g$$

has tensor-train representation with rank not larger than $\frac{g^2+g}{2}$.

Proof. Recall that the Riemann matrix $\mathbf{\Omega}$ is symmetric, thus we have $\pi \mathbf{i} \mathbf{n} \cdot \mathbf{\Omega} \mathbf{n} = \sum_{k=1}^g \pi \mathbf{i} \Omega_{kk} n_k^2 + \sum_{k=1}^g \sum_{l=k+1}^g 2\pi \mathbf{i} \Omega_{kl} n_k n_l$. The tensors $\underline{S}^{\mathbf{kl}} = \underline{S}_{\mathbf{n}}^{\mathbf{kl}} := \pi \mathbf{i} \Omega_{kl} n_k n_l$, $\mathbf{n} \in \mathcal{J}^g$, can be written in the tensor-train form $\underline{S}_{\mathbf{n}}^{\mathbf{kl}} = \mathbf{S}_{n_1}^{\mathbf{kl}(1)} \mathbf{S}_{n_2}^{\mathbf{kl}(2)} \dots \mathbf{S}_{n_g}^{\mathbf{kl}(g)}$ with

$$\mathbf{S}^{\mathbf{kl}(m)} = \begin{cases} \mathbf{1}, & m \notin \{k, l\} \\ \pi \mathbf{i} \Omega_{kl} \mathbf{u}, & m = k \\ 2\mathbf{u}, & m = l \end{cases} \in \mathbb{C}^{(2N+1) \times 1}$$

for $k \neq l$, $\mathbf{u} = [-N - N + 1 \dots 0 \dots N]^T$ and

$$\mathbf{S}^{\mathbf{kk}(m)} = \begin{cases} \mathbf{1}, & m \notin \{k, l\} \\ \pi \mathbf{i} \Omega_{kl} \mathbf{u}^2, & m = k \end{cases} \in \mathbb{C}^{(2N+1) \times 1}$$

with $\mathbf{u}^2 = [(-N)^2 (-N+1)^2 \dots 0 \dots N^2]^T$. Thus, all $\underline{S}^{\mathbf{kl}}$ are tensor-trains of rank one. It implies that

$$\underline{Q} = \sum_{k=1}^g \underline{S}^{\mathbf{kk}} + \sum_{k=1}^g \sum_{l=k+1}^g \underline{S}^{\mathbf{kl}} \quad (3.26)$$

is a tensor-train with rank at most $\hat{g} = \frac{g^2+g}{2}$. □

Trivial implementation of the tensor-train \underline{Q} contains $((g-2)\hat{g}^2 + 2\hat{g})(2N+1)$ elements. The first and last tensor cores $\underline{Q}^{(1)}$ and $\underline{Q}^{(g)}$ consist of rank one matrices (vectors) while all other cores consist of rank \hat{g}^2 matrices. However, the tensor-train \underline{Q} is a sum of rank one tensors. Therefore each of the rank \hat{g}^2 matrix is a diagonal matrix. Hence the tensor-train \underline{Q} can be represented using only $\hat{g}g(2N+1)$ non-zero elements.

Proof of Lemma 2. The scaling and squaring based approximation of c_n is given by $\tilde{c}_n = p_n^s$, where $p_n := 1 + \frac{1}{1!}\hat{q}_n + \frac{1}{2!}\hat{q}_n^2 + \dots + \frac{1}{(K-1)!}\hat{q}_n^{K-1}$ and $\hat{q}_n = q_n/s$. Let $\hat{q}_n = \hat{Q}_{n_1}^{(1)}\hat{Q}_{n_2}^{(2)}\dots\hat{Q}_{n_g}^{(g)}$ and $p_n := \mathbf{P}_{n_1}^{(1)}\mathbf{P}_{n_2}^{(2)}\dots\mathbf{P}_{n_g}^{(g)}$. Then by the properties of the tensor-train format [218]

$$\mathbf{P}_{n_k}^{(k)} = \begin{bmatrix} 1 & & & \\ & \left(\hat{Q}_{n_k}^{(k)}\right)^{\otimes 1} & & \\ & & \ddots & \\ & & & \left(\hat{Q}_{n_k}^{(k)}\right)^{\otimes (K-1)} \end{bmatrix}$$

for $k \notin \{1, g\}$, and

$$\begin{aligned} \mathbf{P}_{n_1}^{(1)} &= \begin{bmatrix} 1 & \frac{1}{\sqrt{1!}}\left(\hat{Q}_{n_1}^{(1)}\right)^{\otimes 1} & \dots & \frac{1}{\sqrt{(K-1)!}}\left(\hat{Q}_{n_1}^{(1)}\right)^{\otimes (K-1)} \end{bmatrix}, \\ \mathbf{P}_{n_g}^{(g)} &= \begin{bmatrix} 1 & \frac{1}{\sqrt{1!}}\left(\hat{Q}_{n_g}^{(g)}\right)^{\otimes 1} & \dots & \frac{1}{\sqrt{(K-1)!}}\left(\hat{Q}_{n_g}^{(g)}\right)^{\otimes (K-1)} \end{bmatrix}^T. \end{aligned}$$

The inner cores $\mathbf{P}_{n_k}^{(k)}$, $k \notin \{1, g\}$ thus are diagonal $R \times R$ matrices with $R = \left(\frac{\hat{g}^{K-1}}{\hat{g}-1}\right)$. Therefore $\tilde{c}_n = p_n^s$ has the tensor-train representation $\tilde{c}_n = \left(\mathbf{P}_{n_1}^{(1)}\right)^{\otimes s} \left(\mathbf{P}_{n_2}^{(2)}\right)^{\otimes s} \dots \left(\mathbf{P}_{n_g}^{(g)}\right)^{\otimes s}$. It follows that the inner cores are diagonal $\hat{R} \times \hat{R}$ matrices with $\hat{R} = R^s$. \square

Remark 2. As the cores $\underline{P}^{(k)}$ consist of only diagonal matrices, the number of non-zero elements in the tensor-train \underline{P} is only $\frac{\hat{g}^{K-1}}{\hat{g}-1}g(2N+1)$.

Proof of Proposition 5. Computing the approximation of the Riemann theta function is equivalent to the inner product of two tensors in the tensor-train format. We can work it out as the following.

$$\begin{aligned} \tilde{\theta}(\mathbf{z} | \mathbf{\Omega}) &= \sum_{\mathbf{n} \in \{-N, \dots, N\}^g} \tilde{c}_n \exp(2\pi i \mathbf{n} \cdot \mathbf{z}) \\ &= \sum_{\mathbf{n} \in \{-N, \dots, N\}^g} e^{2\pi i n_1 z_1} \left(\mathbf{P}_{n_1}^{(1)}\right)^{\otimes s} \dots e^{2\pi i n_g z_g} \left(\mathbf{P}_{n_g}^{(g)}\right)^{\otimes s} \\ &= \Gamma_1 \Gamma_2 \dots \Gamma_g, \quad \Gamma_k := \sum_{j=-N}^N e^{2\pi i j z_k} \left(\mathbf{P}_j^{(k)}\right)^{\otimes s}. \end{aligned}$$

Note that $\gamma_g := \Gamma_g$ is a column vector. With

$$\gamma_{k-1} := \Gamma_{k-1} \gamma_k = \sum_{j=-N}^N e^{2\pi i j z_{k-1}} \left(\left(\mathbf{P}_j^{(k-1)}\right)^{\otimes s} \gamma_k \right),$$

we have $\tilde{\theta}(\mathbf{z} | \mathbf{\Omega}) = \gamma_1$.

We count the number of multiplications required to compute γ_1 as a measure of the computation cost. For computing $\tilde{\theta}(\mathbf{z} | \mathbf{\Omega})$, we start with γ_g which requires $\Omega((2N+1)R^s)$ multiplications for $R = \frac{\hat{g}^K - 1}{\hat{g} - 1}$. In the next stage, as the matrices $\mathbf{P}_{n_k}^{(g-1)}$ are diagonal, the matrix-vector products $\left(\mathbf{P}_{n_k}^{(g-1)}\right)^{\otimes s} \gamma_g$ can be evaluated as the Hadamard product $\text{diag}\left\{\left(\mathbf{P}_{n_k}^{(g-1)}\right)^{\otimes s}\right\} \odot \gamma_g$. Here $\text{diag}\{\cdot\}$ means the vector of the diagonal elements of a matrix. Using the ideas from [229]–[231] the computation of the term $\left(\mathbf{P}_{n_k}^{(g-1)}\right)^{\otimes s} \gamma_g$ given $\mathbf{P}_{n_k}^{(g-1)} \in \mathbb{C}^{R \times R}$ and $\gamma_g \in \mathbb{C}^{R^s \times 1}$ requires at least $\Omega(sR^s)$ multiplications. Computing γ_{g-1} given γ_g thus requires $\Omega((2N+1)sR^s)$ multiplications. Continuing the same way, we can see that computing γ_1 requires at least $\Omega((g-1)(2N+1)sR^s + (2N+1)R^s)$ multiplications. \square

4

THEORETICAL ANALYSIS OF MAXIMUM TRANSMIT POWER IN A *b*-MODULATOR

CHAPTER ABSTRACT

The optimal transmit power in various nonlinear Fourier transform-based transmission systems has been observed to decrease with the signal duration when bandwidth is fixed. A new theoretical explanation for this behavior is provided for a specific *b*-modulator and validated in simulations.

The text in this chapter has previously appeared in modified form in S. Chimmalgi and S. Wahls, "Theoretical analysis of maximum transmit power in a *b*-modulator", in 45th European Conference on Optical Communication (ECOC2019).[232]

4.1. INTRODUCTION

The nonlinear Fourier transform (NFT) [10] can solve the normalized nonlinear Schrödinger equation (NSE)

$$i \frac{\partial q}{\partial x} + \frac{\partial^2 q}{\partial t^2} + 2|q|^2 q = 0, \quad q = q(x, t), \quad (4.1)$$

which is a model for an ideal lossless single-mode fiber obtained after suitable normalization and path averaging [21, Ch. 5]. Here $q(x, t)$ is the slowly varying pulse envelope, x is the location and t is retarded time, all in normalized units. The nonlinear evolution of the signal along the fiber equals a simple phase rotation in the nonlinear Fourier domain (NFD) [10]. Hence it was suggested to embed data in the NFD and use the NFT to recover the data [66], [68]. This idea is known as nonlinear frequency division multiplexing (NFDm).

NFDm has garnered much attention in recent years and many different NFDm system designs have been proposed [35], [85], [93], [95]. A common problem with many NFDm designs is that the optimum transmit power decreases with signal duration, making it difficult to utilize signals significantly longer than the channel memory [86], [233], [234]. Thus, signals are typically short with a large portion acting as a guard interval that contains no information, leading to low spectral efficiencies. The difficulties with transmitting longer signals at sufficiently high powers are typically attributed to interactions between the signal and noise during the numerical computation of the NFT [86], [98], [233], [234].

In this chapter we provide a new explanation for this phenomenon. We derive an upper bound on transmit power for the specific NFDm system proposed in [92]. The bound decreases with signal duration when the bandwidth is kept constant. Since our implementation achieves transmit powers close to that bound, we show that signal-noise interactions are not a major limiting factor in our setup.

4.2. REVIEW OF b -MODULATION

The NFT is obtained by solving the initial-value problem [10]

$$\frac{\partial \boldsymbol{\phi}(\lambda, t)}{\partial t} = \begin{bmatrix} -i\lambda & q(t) \\ -q^*(t) & i\lambda \end{bmatrix} \boldsymbol{\phi}(\lambda, t), \quad \lim_{t \rightarrow -\infty} \boldsymbol{\phi}(\lambda, t) = \begin{pmatrix} e^{-i\lambda t} \\ 0 \end{pmatrix},$$

where x is considered fixed and thus dropped. The Jost scattering coefficients are defined as $a(\lambda) = \lim_{t \rightarrow -\infty} \phi_1(\lambda, t) e^{i\lambda t}$ and $b(\lambda) = \lim_{t \rightarrow -\infty} \phi_2(\lambda, t) e^{-i\lambda t}$, where λ is a complex parameter. Information can be embedded in the Jost scattering coefficients in various ways. The NFDm technique in which information is embedded in $b(\xi)$, $\xi \in \mathbb{R}$, is known as b -modulation [90]. The advantages of b -modulation are tight control over signal duration and lower sensitivity w.r.t. noise [90], [234]. For b -modulation the energy of the normalized field

$$E[q(t)] = \int_{-\infty}^{\infty} |q(t)|^2 dt \quad (4.2)$$

is equal to [90]

$$E[b(\xi)] = -\frac{1}{\pi} \int_{-\infty}^{\infty} \ln(1 - |b(\xi)|^2) d\xi, \quad (4.3)$$

where $E[\cdot]$ denotes the operation of computing the energy either from the time-domain signal $q(t)$ or the nonlinear Fourier coefficient $b(\xi)$.

POWER CONTROL BY CONSTELLATION RESHAPING (PCCR)

In this chapter we analyze the system from [92]. In such a system the data is transmitted using signal bursts. For each burst, we have

$$b(\xi) = \sum_{k=-N}^N s_k \Psi(\xi - k\Delta_\xi). \quad (4.4)$$

Here, $\Psi(\xi)$ is a specific flat-top carrier (see [92, Eq. 17, Eq. 18 and Figure 1]), Δ_ξ is the carrier spacing and the s_k are information symbols. The average signal power is controlled by using a suitably shaped constellation for the s_k . The constellation is shaped such that $\mathbb{E}\{E[s_k \Psi(\xi)]\} = E_d$, where $\mathbb{E}\{\cdot\}$ denotes expectation w.r.t. the s_k , $E_d > 0$ is a design parameter, and $E[s_n \Psi(\xi)]/E[s_k \Psi(\xi)] = |s_n|^2/|s_k|^2$ for $n \neq k$. This system design ensures that [92]

$$\mathbb{E}\{E[b(\xi)]\} \approx (2N + 1)E_d \quad (4.5)$$

assuming that Δ_ξ is not too small. By definition of the NFT, for the case of anomalous dispersion, $|b(\xi)| < 1$. The parameter E_d thus has to stay below the maximum carrier energy [92]:

$$\text{MCE}[\Psi(\xi)] := \lim_{A \rightarrow (1/\sup_\xi |\Psi(\xi)|)^-} E[A\Psi(\xi)] \geq E_d. \quad (4.6)$$

The MCE has been observed to be finite for common carriers which result in time-limited signals [92, Section 2.4]. The duration of the generated signals can be manipulated by scaling the carrier $\Psi(\xi) \rightarrow \Psi(c\xi) \Rightarrow q(t) \rightarrow q(t/c)$, $c > 0$, [68, pp. IV–D]. To keep utilizing the complete provided bandwidth, we choose $c_N = (2N + 1)c_0$ and increase the number of subcarriers proportionally. We thus consider the following nonlinear spectrum

$$b_N(\xi) = \sum_{k=-N}^N s_k \Psi(c_N \xi - k\Delta_\xi) \quad (4.7)$$

in this chapter. The generated signals have $2N + 1$ subcarriers and are of duration $T_N = (2N + 1)T_0$. The bandwidth stays approximately constant because the subcarriers shrink with N .

4.3. THEORETICAL ANALYSIS

In this section we analyze how the maximum power that the b -modulator discussed in the previous section can achieve depends on the number of subcarriers, which is directly proportional to the signal duration. Our analysis rests on two simple insights. First, since the signals are of finite duration, energy translates directly into power. Second, since $E[\Psi(c_N \xi)] = c_N^{-1} E[\Psi(\xi)]$ by basic integration laws,

$$\text{MCE}[\Psi(c_N \xi)] = c_N^{-1} \text{MCE}[\Psi(\xi)]. \quad (4.8)$$

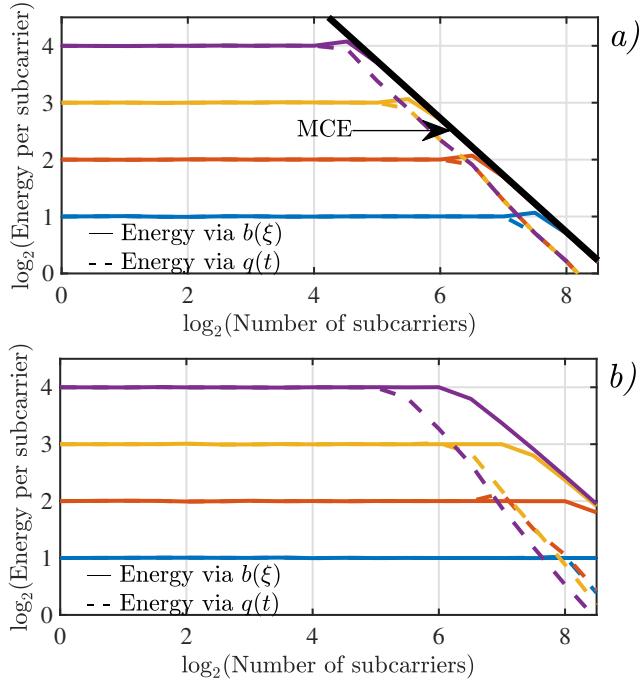


Figure 4.1: Each of the four colours in both subfigures corresponds to the design energy per subcarrier $E_d = \{2, 4, 8, 16\}$. The solid and dashed lines represent the values of mean energy per subcarrier computed from $b(\xi)$ and $q(t)$ respectively for each of the four design values of E_d . *a)* PCCR: The two energy estimates agree almost upto the theoretical maximum. *b)* PCRB: The two estimates also start to decrease with number of subcarriers (signal duration) due to signal truncation and finite precision (similar to [98, Figure 6a]).

We bound the average power P_N achieved with $2N + 1$ subcarriers using (4.5), (4.6), (4.8) and $c_N = (2N + 1)c_0$:

$$\begin{aligned}
 P_N &= \frac{\mathbb{E}\{E[b_N(\xi)]\}}{T_N} \approx \frac{(2N + 1)E_d}{T_N} \\
 &\leq \frac{(2N + 1)\text{MCE}[\Psi(c_N\xi)]}{T_N} = \frac{c_0^{-1}\text{MCE}[\Psi(\xi)]}{T_N}.
 \end{aligned} \tag{4.9}$$

Note that the numerator is a constant independent of the number of subcarriers, so that the bound on the transmit power P_N decreases with the signal duration T_N . To the best of our knowledge this is the first bound on transmit power ever reported for a NFDN system. We will observe in the simulation results that instructing the algorithms to achieve powers beyond this bound leads to numerical failure and rapidly decaying system performance. Hence our bound strongly suggests that for any signal duration T_N there is a finite optimum transmit power. We remark that even though our analysis holds for the particular b -modulation scheme in [92], similar observations have been reported also for other NFDN architectures [86], [233], [234].

4.4. SIMULATIONS

The goal of this section is two-fold. First, we demonstrate that our theoretical bound is useful to describe actual system behavior. Second, we compare the analyzed PCCR approach from Section 4.2 with an alternative scheme from [99] that we call power control by reshaping $b(\xi)$ (PCRB). The advantage of PCRB is that – in theory – arbitrarily large energies can be achieved. The disadvantage is that the generated signals are not of finite duration anymore. We want to investigate whether in practice PCRB can generate higher signal powers than PCCR.

SETUP

We used the carrier waveform [92, (18)] with $T = 0.5$ for both systems. We chose the constants c_0 , T_0 and Δ_ξ from Section 4.2 as 1.0, 0.5 and 135 respectively. The signals generated using PCRB were truncated to duration T_N . The signals generated using PCCR were of duration T_N by design. The symbols s_k were drawn from a QPSK constellation. The simulations were carried out using NFDMLab [154], which uses the software library FNFT [128] to compute (inverse) NFTs. To avoid algorithm breakdowns, all $b(\xi)$ were clipped such that $|b(\xi)| \leq 1 - 2.2204 \times 10^{-15}$. Furthermore a numerical improvement to the inverse NFT (INFT) described in the appendix was used. The fiber parameters were $\beta_2 = -5 \text{ ps}^2 \text{ km}^{-1}$, $\gamma = 1.2 \text{ W}^{-1} \text{ km}^{-1}$ and $\alpha = 0.2 \text{ dB km}^{-1}$. The signal duration in real-world units was $1.25 T_N$ ns. The transmitted and received signals were low-pass filtered to 40 GHz. Amplification was carried out using EDFAs with a 6 dB noise figure.

IMPACT OF FINITE PRECISION

Recall that the energy of the signal can be computed from $b(\xi)$ via (4.3) or from $q(t)$. Since $b(\xi)$ is the input to the INFT and $q(t)$ is the output, we compare these two energies in order to assess the accuracy of the INFT. In Figure 4.1, we show the average signal energy (taken over 20 signal realizations) divided by the number of subcarriers for both PCCR and PCRB. In Figure 4.1a we see that both energy estimates stay close together which hints that numerical effects in the INFT are not the major limiting factor for PCCR. The numerical values also confirm that the achievable energy per subcarrier is always less than the maximum carrier energy for PCCR (4.6). In Figure 4.1b we see that the energy via $b(\xi)$ eventually starts decreasing even though it should stay constant in theory. This behavior is due to the clipping mentioned above. The gap to energy via $q(t)$ is larger than in Figure 4.1a, which we attribute to the signal truncation mentioned above. The energies via $q(t)$ in Figure 4.1b are higher than those in Figure 4.1a but show a similar decay.

BACK-TO-BACK (B2B)

We simulated 1m of fiber followed by a single amplifier to add noise. After demodulation and equalization, estimates \hat{s}_k of the transmitted symbols s_k were obtained. For PCCR, error vector magnitudes (EVMS) between \hat{s}_k and s_k were computed. For PCRB the inverse of the map $R(u) = \sqrt{1 - e^{-|u|^2}} e^{i\angle u}$ [99, Figure 2a] was applied to \hat{s}_k before computing the EVM. This was done to ensure similarly shaped noise balls. We can see examples of the noise balls for both systems in Figures 4.3 and 4.4. The EVM results are shown in Figures 4.2a-b. In Figure 4.2a, PCCR eventually breaks down due to ill-conditioning in

the INFT algorithms. However, this happens only when trying to exceed the theoretical limit depicted in Figure 4.1a. The initial decay of the curves seems to be a result of the nonlinear nature of the NFT which squeezes the noise balls [235, Figure 2]. From Figure 4.2b we observe that PCRB performs better than PCCR at low subcarrier energies but worse at the high one.

TRANSMISSION

We finally compared PCCR and PCRB in a transmission scenario. To make the scenario more realistic, we added precompensation as in [233], [234] and truncated the signals to $0.556T_{Ns}$ before transmission so that now both methods suffer from truncation errors. The signals were transmitted over a 8×80 km link. The results are shown in Figures 4.2c-d. By comparing Figures 4.2a and 4.2c, we observe that the breakdowns still occur approximately for the same number of subcarriers, which hints that the transmit power limitation from our theoretical analysis is a dominant effect even in our transmission setup. The spike in the $E_d = 16$ curve seems to occur due to numerical issues. We remark that it occurs when the maximum number of subcarriers for this E_d has been exceeded. Similar observations can be made when comparing Figures 4.2b and 4.2d.

4.5. CONCLUSION

We derived an upper bound on the achievable transmit power for the b -modulator from [92] (PCCR) that decreases with signal duration for fixed bandwidth. It seems to be the first such bound for NFDm. In simulations we achieved transmit powers close to the theoretical bound which suggests that – in our setup – numerical signal-noise interactions were a minor issue. We found that another b -modulator from [99] (PCRB) suffers from similar limitations even though the maximum carrier energy argument used in our analysis does not apply to that case. We plan to use our analysis to improve NFDm system designs.

4.A. IMPROVED INFT ALGORITHM

The INFT was computed as described in [90] with FNFT [128]. To run the algorithm we require D samples of

$$B(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} b(\xi) e^{i\tau\xi} d\xi \approx \frac{e^{i\tau\xi_-}}{2\pi} \int_{\xi_-}^{\xi_+} b(\xi) e^{i\tau(\xi-\xi_-)} d\xi \quad (4.10)$$

on a grid $\tau_k = \tau_- + k\delta$. We approximate the right integral by

$$\begin{aligned} \hat{B}(\tau_k) &= \frac{e^{i\tau_k\xi_-}}{2\pi} \sum_{n=0}^{M-1} b(\xi_n) e^{i\tau_k(\xi_n-\xi_-)} \epsilon \\ &= \frac{\epsilon e^{-i\tau_k\xi_-}}{2\pi} \sum_{n=0}^{M-1} b(\xi_n) \left(e^{i\tau_k\epsilon} \left(e^{i\delta\epsilon} \right)^{-k} \right)^{-n}, \end{aligned} \quad (4.11)$$

where $\xi_n = \xi_- + n\epsilon$. The $B(\tau_k)$ are computed fast by applying chirp z-transform [236] to the last sum.

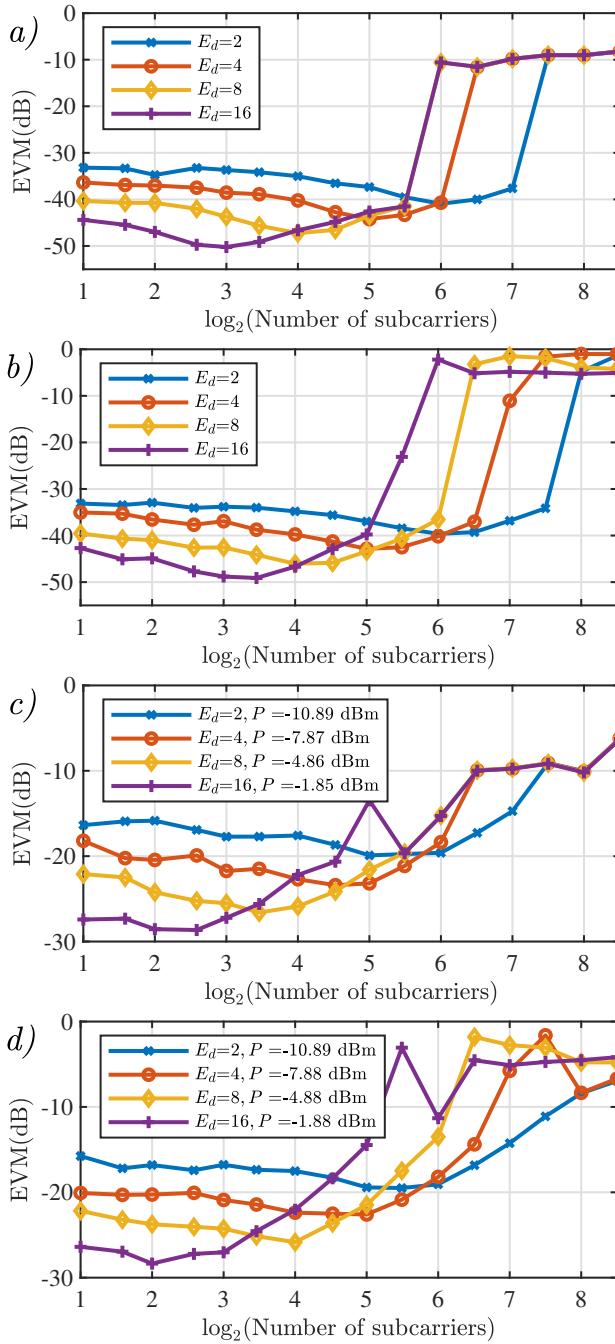


Figure 4.2: a) B2B for PCCR b) B2B for PCRb c) Transmission for PCCR d) Transmission for PCRb

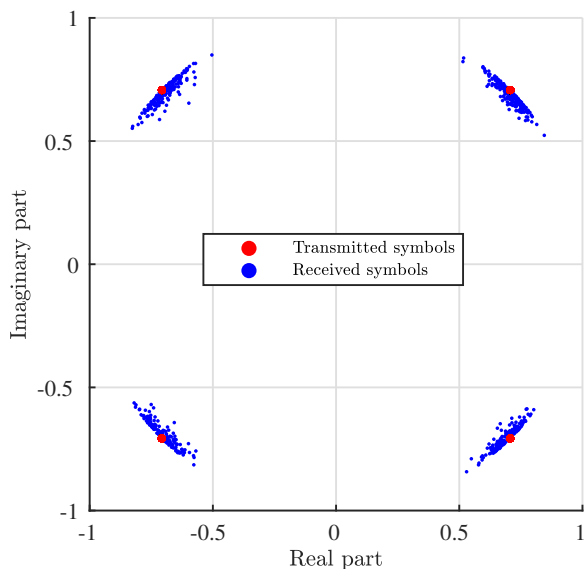


Figure 4.3: PCCR: Transmitted and received symbols after transmission with $E_d = 16$ and 16 subcarriers. The noise balls are squeezed due to the nonlinear nature of the NFT.

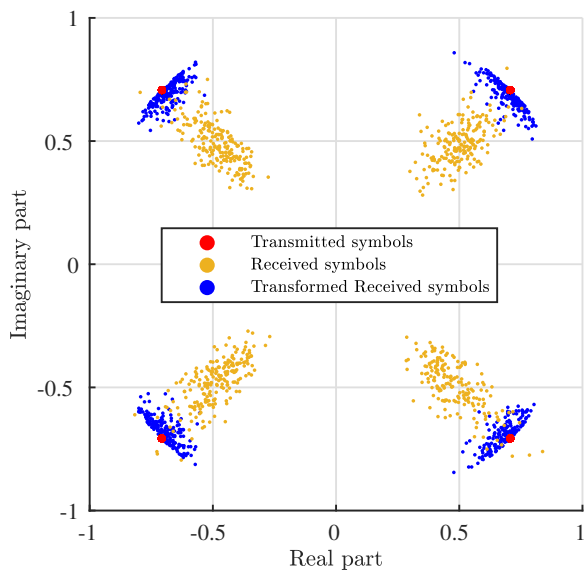


Figure 4.4: PCRB: Transmitted, received and transformed received symbols after transmission with $E_d = 16$ and 16 subcarriers. The noise balls of the received symbol are very different compared to the case of PCCR. However, after the inverse mapping is applied, the transformed received symbols have similar noise balls.

5

BOUNDS ON THE TRANSMIT POWER OF b -MODULATED NFDM SYSTEMS IN ANOMALOUS DISPERSION FIBER

CHAPTER ABSTRACT

The performance of various nonlinear frequency division multiplexed (NFDM) fiber-optic transmission systems has been observed to decrease with increasing signal duration. For a class of NFDM systems known as b -modulators, we show that the nonlinear bandwidth, signal duration, and power are coupled when singularities in the nonlinear spectrum are avoided. When the nonlinear bandwidth is fixed, the coupling results in an upper bound on the transmit power that decreases with increasing signal duration. Signal-to-noise ratios are consequently expected to decrease, which can help explain drops in performance observed in practice. Furthermore, we show that there is often a finite bound on the transmit power of b -modulators even if spectral singularities are allowed.

The text in this chapter has previously appeared in modified form in the open access article S. Chimmalgi and S. Wahls, "Bounds on the Transmit Power of b -Modulated NFDM Systems in Anomalous Dispersion Fiber," Entropy, Vol. 22, No. 6, Article 639, 2020. Special Issue Information Theory of Optical Fiber [237]. Its reuse is licensed under CC BY 4.0 [156]

5.1. INTRODUCTION

The nonlinear Fourier transform (NFT) [10] is a mathematical tool to solve the normalized nonlinear Schrödinger equation (NSE)

$$i \frac{\partial q}{\partial z} + \frac{1}{2} \frac{\partial^2 q}{\partial t^2} + \kappa |q|^2 q = 0, \quad q = q(z, t), \quad (5.1)$$

which is a model for an ideal lossless single-mode fiber obtained after suitable normalization and path averaging [17]. (The path average can be avoided by using tapered fibers [53].) Here, $q(z, t)$ is the slowly varying pulse envelope, z is the location, and t is retarded time, all in normalized units. The parameter κ determines if the dispersion in the fiber is normal (-1) or anomalous ($+1$). The nonlinear evolution of the signal according to the NSE equals a simple phase rotation in the nonlinear Fourier domain (NFD) [10]. Hence, it was suggested to embed data in the NFD at the transmitter and use the NFT to recover the data at the receiver [66], [68]. This idea is known as nonlinear frequency division multiplexing (NFDm).

NFDm has garnered much attention in recent years and many different NFDm system designs have been proposed [35], [53], [80], [85], [86], [88], [93]–[95], [98], [100], [234], [238]. A common problem with many NFDm designs is that the optimum transmit power decreases with signal duration, making it difficult to utilize signals significantly longer than the channel memory [86], [232]–[234], [239]. Thus, signals are typically short with a relatively large portion acting as a guard interval that contains no information, leading to low spectral efficiencies. The difficulties with transmitting longer signals have been suspected to be caused by limitations of numerical NFT algorithms and increased signal-noise interactions [233, p.3], [234, Section 3.3], [98, Section 4].

In Chapter 4 (first reported in [232]), we discovered a new factor contributing to this phenomenon when we derived an upper bound on the transmit power of one specific NFDm system proposed in [92]. It was shown that the transmit power bound decreases with signal duration when the nonlinear bandwidth is kept constant. Since signals with lower power are more susceptible to corruption by noise, this leads to reduced transmission performance. In this paper, we look at a class of systems where only a part of the nonlinear Fourier spectrum known as continuous spectrum is modulated, which is the nonlinear analogue of linear frequency division multiplexing. More specifically, we look at so-called b -modulators in the case of anomalous dispersion. The paper is organized as follows. In Section 5.2, we briefly review nonlinear frequency division multiplexing (NFDm). In Section 5.3, we derive two different upper bounds on the transmit power of b -modulated systems. We conclude our findings in Section 5.4.

NOTATION

Real numbers: \mathbb{R} ; $\mathbb{R}_{\geq 0} := \{x \in \mathbb{R} : x \geq 0\}$; Complex numbers: \mathbb{C} ; Complex numbers with positive imaginary part: \mathbb{H} ; Integers: \mathbb{Z} ; Natural numbers: \mathbb{N} ; $i := \sqrt{-1}$; Euler's number: e ; Real part: $\Re(\cdot)$; Imaginary part: $\Im\{\cdot\}$; Complex conjugate: $(\cdot)^*$; Absolute value: $|\cdot|$; Lebesgue spaces: $\mathcal{L}^p(X)$ contains all measurable complex-valued functions f on X for which

$$\|f\|_p := \begin{cases} (\int_X |f(x)|^p dx)^{1/p}, & \text{if } 1 \leq p < \infty \\ \sup_{x \in X} |f(x)|, & \text{if } p = \infty \end{cases} < \infty. \quad (5.2)$$

5.2. REVIEW OF NFDM

In this section, we describe the mathematical machinery behind the nonlinear Fourier transform (NFT) and review the idea of nonlinear frequency domain multiplexing (NFDM).

5.2.1. NONLINEAR FOURIER TRANSFORM FOR VANISHING SIGNALS

The nonlinear Fourier transform (NFT) that solves the NSE (5.1) is due to Zakharov and Shabat [10]. It transforms any signal $q(t)$ that vanishes sufficiently quickly for $t \rightarrow \pm\infty$ from the time domain to the nonlinear Fourier domain through the analysis of the linear ordinary differential equation (ODE)

$$\frac{\partial \mathbf{V}(t, \lambda)}{\partial t} = \mathbf{C}(t, \lambda) \mathbf{V}(t, \lambda) = \begin{bmatrix} -i\lambda & q(t) \\ -\kappa q^*(t) & i\lambda \end{bmatrix} \mathbf{V}(t, \lambda). \quad (5.3)$$

The term $\lambda \in \mathbb{C}$ is a spectral parameter similar to the parameter s in the Laplace domain. Since $|q(t)| \rightarrow 0$ fast for $t \rightarrow \pm\infty$, the ODE has solutions that fulfill the boundary conditions

$$\begin{aligned} \mathbf{V}(t, \lambda) &= [\boldsymbol{\phi}(t, \lambda) \ \bar{\boldsymbol{\phi}}(t, \lambda)] \rightarrow \begin{bmatrix} e^{-i\lambda t} & 0 \\ 0 & -e^{i\lambda t} \end{bmatrix} \text{ as } t \rightarrow -\infty, \\ \mathbf{V}(t, \lambda) &= [\bar{\boldsymbol{\psi}}(t, \lambda) \ \boldsymbol{\psi}(t, \lambda)] \rightarrow \begin{bmatrix} e^{-i\lambda t} & 0 \\ 0 & e^{i\lambda t} \end{bmatrix} \text{ as } t \rightarrow \infty. \end{aligned} \quad (5.4)$$

The matrix $\mathbf{V}(t, \lambda)$ is said to contain (generalized) eigenfunctions since Equation (5.3) can be rearranged into an eigenvalue equation with respect to λ [11]. For the solutions Equation (5.4) of Equation (5.3), there exists a unique matrix

$$\mathcal{S}(\lambda) = \begin{bmatrix} a(\lambda) & \bar{b}(\lambda) \\ b(\lambda) & -\bar{a}(\lambda) \end{bmatrix}, \quad (5.5)$$

called the scattering matrix, such that [11]

$$[\boldsymbol{\phi}(t, \lambda) \ \bar{\boldsymbol{\phi}}(t, \lambda)] = [\bar{\boldsymbol{\psi}}(t, \lambda) \ \boldsymbol{\psi}(t, \lambda)] \mathcal{S}(\lambda). \quad (5.6)$$

The components $a(\lambda)$, $b(\lambda)$, $\bar{b}(\lambda)$, and $\bar{a}(\lambda)$ are known as the *scattering coefficients*. The scattering coefficients satisfy [11] (p. 260 and p. 271)

$$\bar{b}(\lambda) = \kappa b^*(\lambda^*), \quad \bar{a}(\lambda) = a^*(\lambda^*), \quad a(\lambda)\bar{a}(\lambda) + b(\lambda)\bar{b}(\lambda) = 1. \quad (5.7)$$

The evolution of the scattering coefficients along the location z in the fiber is simple: [11, Section III]

$$\begin{aligned} a(z, \lambda) &= a(0, \lambda), \\ b(z, \lambda) &= b(0, \lambda) e^{4i\lambda^2 z}. \end{aligned} \quad (5.8)$$

The *reflection coefficient* is then defined as $\rho(\lambda) = b(\lambda)/a(\lambda)$ for $\lambda \in \mathbb{R}$. It provides a representation of the *continuous spectrum*. In the anomalous dispersion case $\kappa = 1$ considered in this paper, the nonlinear Fourier spectrum can also contain a so-called *discrete spectrum*. It corresponds to the complex poles of the reflection coefficient in the upper half-plane \mathbb{H} , or equivalently to the zeros $\lambda_k \in \mathbb{H}$ of $a(\lambda)$. Usually, there are only

finitely many (N) such poles, all simple [11, Section VI]. The poles λ_k are also referred to as *eigenvalues* and a corresponding set of values $\rho_k := b(\lambda_k)/\frac{da}{d\lambda}(\lambda_k)$ are known as residues [11, App. 5]. The eigenvalues correspond to the solitonic components of the signal. There are different ways to define a nonlinear Fourier spectrum. One possibility is $\{\rho(\lambda)\}_{\lambda \in \mathbb{R}}, (\lambda_k, \rho_k)_{k=1}^N$ [11]. Another is $\{b(\lambda)\}_{\lambda \in \mathbb{R}}, (\lambda_k, b(\lambda_k))_{k=1}^N$ [38]. In the case of anomalous dispersion ($\kappa = 1$), the energy of the signal $q(t)$ is related to the components of the nonlinear spectrum as [68, p.9]

$$\int_{-\infty}^{\infty} |q(t)|^2 dt = \frac{1}{\pi} \int_{-\infty}^{\infty} \log(1 + |\rho(\xi)|^2) d\xi + 4 \sum_{k=1}^N \Im\{\lambda_k\}. \quad (5.9)$$

Substituting $\rho(\xi) = b(\xi)/a(\xi)$ and using $|a(\xi)|^2 + |b(\xi)|^2 = 1$ for $\xi \in \mathbb{R}$ due to (5.7) we equivalently have

$$\int_{-\infty}^{\infty} |q(t)|^2 dt = -\frac{1}{\pi} \int_{-\infty}^{\infty} \log(1 - |b(\xi)|^2) d\xi + 4 \sum_{k=1}^N \Im\{\lambda_k\}. \quad (5.10)$$

Note that Equation (5.7) implies that $|b(\xi)| \leq 1$ for real ξ . When $|b(\xi)| = 1$ for some real ξ , then the integrand in Equation (5.10) is undefined at that point. Such points are known as *spectral singularities* in the literature [240]. Even though simple signals such as the rectangle and hyperbolic secant can have isolated spectral singularities [241, Chapter 2], most of the literature on NFTs assumes that $|b(\xi)| < 1$ for all real ξ . From here on, ξ will be used to denote the spectral parameter if it is strictly real and λ if otherwise.

Information can be embedded in the scattering coefficients in various ways. In this paper, we consider the techniques where information is embedded only in $b(\xi)$ for $\xi \in \mathbb{R}$, i.e., we consider signals without solitons. The idea of embedding information in $b(\xi)$ is known as b -modulation [90]. The advantages of b -modulation are tight control over signal duration and lower sensitivity w.r.t. noise [90], [234]. If the signals are of infinite duration, they are truncated to some finite interval $[T_1, T_2]$. From Equation (5.10), we can see that the energy of a b -modulated signal can be controlled by varying $b(\xi)$. This indirectly allows us to control the average power of the truncated signal. In this chapter, we will concentrate on b -modulation in the case of anomalous dispersion ($\kappa = 1$).

5.2.2. NFDM SIGNAL GENERATION

As in any digital transmission scheme, the data to be transmitted are fed to the transmitter as a stream of bits. The system then takes a block of $N_b \in \mathbb{N}$ bits and generates a signal for transmission through the optical fiber channel. This is the process of modulation. At the receiver, the effect of the channel on the nonlinear spectrum is first reverted using Equation (2.6). Then, the block of bits is recovered. The NFDM transmission scheme is illustrated in Figure 5.1. In order to be able to make concise statements in the coming sections, we now introduce formal definitions for a modulator and a b -modulator. An illustrating block diagram is shown in Figure 5.2.

Definition 4. A modulator is a function-valued function

$$\mathcal{M} : \{0, 1\}^{N_b} \rightarrow \mathcal{L}^2([T_1, T_2]) \quad (5.11)$$

that maps vectors of N_b bits to transmit signals of finite energy and duration.

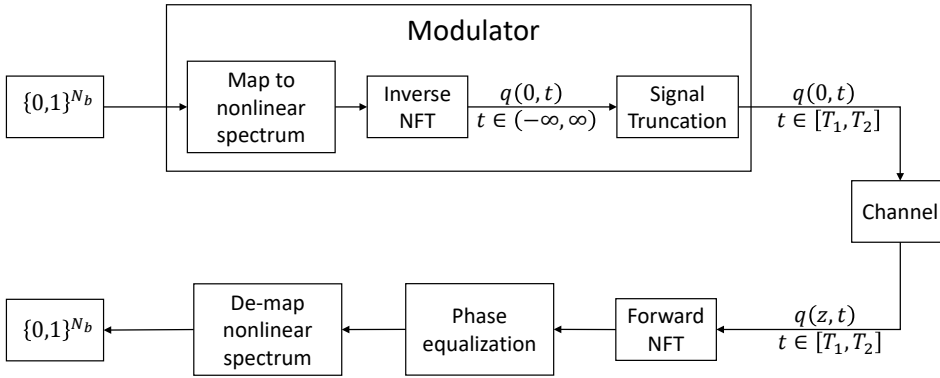


Figure 5.1: Nonlinear frequency domain multiplexing (NFDm) transmission of one block of N_b bits.

This definition of a modulator makes no assumptions about how data are embedded in the signal $q(t)$ and is thus very general. A b -modulator on the other hand is a specific type of modulator that embeds data in the scattering coefficient $b(\xi)$ that was defined in Equation (5.5).

Definition 5. A b -modulator is a modulator of the form

$$\mathcal{M}(\mathbf{v}) = \mathcal{T}(\mathcal{Q}(\mathcal{B}(\mathbf{v}))), \quad (5.12)$$

where \mathcal{B} maps vectors of bits to nonlinear spectra $b(\xi)$ with $\xi \in \mathbb{R}$, \mathcal{Q} is the inverse NFT that maps scattering coefficients $b(\xi)$ to the corresponding time-domain signals $q(t)$, $t \in \mathbb{R}$, without solitonic components, and

$$\mathcal{T} : \mathcal{L}^2(\mathbb{R}) \rightarrow \mathcal{L}^2([T_1, T_2]), \quad [\mathcal{T}(q)](t) = q(t) \quad \forall t \in [T_1, T_2], \quad (5.13)$$

simply truncates infinite duration signals to a finite duration. We assume that $b = \mathcal{B}(\mathbf{v})$ and $q = \mathcal{Q}(b)$ satisfy

$$\|b\|_\infty \leq 1, \quad \int_{-\infty}^{\infty} |q(t)|^2 dt = -\frac{1}{\pi} \int_{-\infty}^{\infty} \log(1 - |b(\xi)|^2) d\xi < \infty, \quad \forall \mathbf{v} \in \{0, 1\}^{N_b}. \quad (5.14)$$

Remark 3. The first assumption in Equation (5.14) is necessary since $|a(\xi)|^2 + |b(\xi)|^2 = 1$ on the real axis for any NFT (see Equation (5.7)). Note that we do not make the common stronger assumption that $\|b\|_\infty < 1$ (no spectral singularities). The second assumption in Equation (5.14) is simply Equation (5.10) specialized to nonlinear spectra without solitonic components. It is known to hold in the absence of spectral singularities. We expect this result to hold even in the presence of spectral singularities. However, as we could not find this result in the literature, we are stating it as an assumption here. We remark that, even if b -modulators that satisfy Equation (5.14) with $\|b\|_\infty = 1$ would not exist, our results still apply to any b -modulator that ensures $\|b\|_\infty < 1$. This still includes all cases in the current literature.

Remark 4. For sufficiently rapidly decaying $b(\xi)$ with $\|b\|_\infty < 1$, it is possible to verify that the second integral in Equation (5.14) will be finite. However, when $\|b\|_\infty = 1$, the integrand will have singularities at which it becomes infinite. The integral may or may not be infinite in such cases. It was observed in [92] that it remains finite in specific cases. In this chapter, we will show in Lemma 5 that this behavior is the norm, not the exception.

Remark 5. The results that will be derived in this chapter for b -modulated systems for anomalous dispersion fiber also apply to ρ -modulated NFDN systems with normal dispersion ($\kappa = -1$, see, e.g., [87], [97]) when b is replaced by ρ . Let us check that the assumptions in Equation (5.14) are fulfilled by ρ in that case. For normal dispersion, $|a(\xi)|^2 - |b(\xi)|^2 = 1$ [16, p. 25]. Thus, $\rho = b/a$ satisfies $\|\rho\|_\infty \leq 1$. Using [16, Equations 1.6.7 and 1.6.21b], the signal energy is found to satisfy

$$\int_{-\infty}^{\infty} |q(t)|^2 dt = -\frac{1}{\pi} \int_{-\infty}^{\infty} \log(1 - |\rho(\xi)|^2) d\xi. \tag{5.15}$$

5

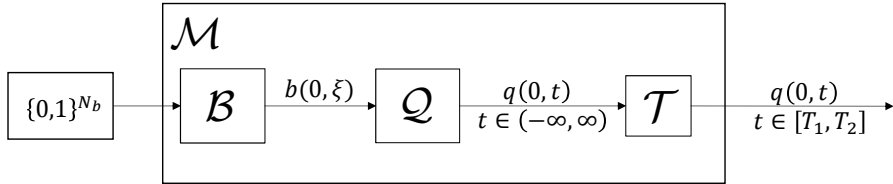


Figure 5.2: Transmitter side components of a NFDN transmission scheme employing b -modulation.

In the next section, we will derive two different bounds on the transmit power of information bearing signals $q(0, t)$ that are generated by b -modulators at the fiber input.

5.3. UPPER BOUNDS ON THE TRANSMIT POWER OF b -MODULATORS

With fiber-optic transmission systems that modulate the conventional “linear” Fourier spectrum, the power of the transmit signal can theoretically be made arbitrarily high without increasing the bandwidth or duration of the signal, simply by scaling (amplifying) it in the time domain. Although b -modulated systems are in many ways similar to such linear systems, there are also some important differences. Scaling the signal in time domain distorts its nonlinear Fourier spectrum in complicated ways. For example, scaling a signal without solitonic components can give rise to many solitons. In linear systems, bandwidth and signal duration are coupled, but the transmit power is independent of the two. We show in the following that, under certain conditions the nonlinear bandwidth, duration, and transmit power in b -modulators are coupled. We already showed this for one specific b -modulator in [232]. In this section, we will derive two more general bounds on the power of b -modulated systems that apply to many systems considered in the literature. In Section 5.3.1, we derive and discuss a bound for systems that have no spectral singularities. In Section 5.3.2, we will show that, even

in the presence of spectral singularities, the power still remains bounded for a class of b -modulators.

5.3.1. POWER BOUND FOR A FIXED GAP TO SINGULARITY.

As already discussed earlier, in the case of anomalous dispersion, it is required that $|b(\xi)| \leq 1$ for real ξ . In special cases, even simple signals such as the rectangle and hyperbolic secant can have isolated spectral singularities at which $|b(\xi)| = 1$ [241, Chapter 2]. In the presence of spectral singularities, the usual theory behind the NFT unfortunately breaks down and has to be extended in a quite complicated manner [240]. Many algorithms available in literature for computing the time-domain signal starting from $b(\xi)$ break down in their presence [90], [242]. In practice, to avoid the complications arising from the spectral singularities, a “gap to singularity” $\varepsilon_b := 1 - \|b\|_\infty^2 > 0$ is typically enforced by either clipping [173], [232]¹, scaling [90], [173], and/or reshaping [98] of $b(\xi)$. In [92], the constellation was reshaped. The gap to singularity ε_b cannot be made arbitrarily small as the numerical algorithms are limited by the computing precision. Since any number that is closer to one than the machine precision is rounded to one, gaps to singularity smaller than machine precision cannot be represented with floating point numbers. As soon as the gap to singularity is never zero, the following power bound applies.

Theorem 2. *Let \mathcal{M} be any b -modulator (see Definition 5) with a gap to singularity. That is,*

$$\varepsilon := 1 - \max_{\mathbf{v} \in \{0,1\}^{N_b}, b = \mathcal{B}(\mathbf{v})} \|b\|_\infty^2 > 0. \quad (5.16)$$

Then, the maximum transmit power of the modulator is upper bounded as

$$P_{\max} = \max_{q \in \mathcal{M}(\mathbf{v}), \mathbf{v} \in \{0,1\}^{N_b}} \frac{1}{T_2 - T_1} \int_{T_1}^{T_2} |q(t)|^2 dt \leq \frac{-2W \log \varepsilon}{\pi \gamma (T_2 - T_1)}, \quad (5.17)$$

where $0 < \gamma < 1$ can be chosen arbitrarily and $W > 0$ is any finite constant such that

$$\gamma E := -\frac{\gamma}{\pi} \int_{-\infty}^{\infty} \log(1 - |b(\xi)|^2) d\xi \leq E_W := -\frac{1}{\pi} \int_{-W}^W \log(1 - |b(\xi)|^2) d\xi, \quad \forall b = \mathcal{B}(\mathbf{v}). \quad (5.18)$$

It is always possible to find such a W .

Remark 6. *Note that $2W$ is a bound on the nonlinear $\gamma \times 100$ -percent bandwidth of the modulator, which is illustrated in Figure 5.3 together with the gap to singularity ε_b . Figure 5.4 illustrates the decay of the power bound.*

Remark 7. *The most important implication of Theorem 2 is that as soon as there is a nonlinear bandwidth constraint ($\gamma E \leq E_W$) and the gap to singularity cannot be made arbitrarily small (e.g., due to clipping or finite precision effects), the transmit power of any b -modulator producing long transmit signals must be low. Longer signals are preferred as they are more data dense. However, making the signals longer decreases the SNRs. Hence, one expects there to be a finite optimal signal duration.*

¹The use of clipping in [173] was reported in [98, p. 1574], not in the paper itself.

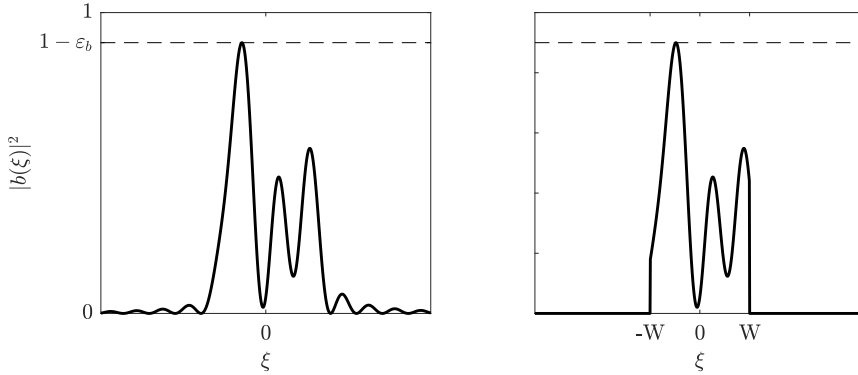


Figure 5.3: In this example, $2W$ is exactly the 90% bandwidth: 90% ($\gamma = 0.9$) of the energy corresponding to the left spectrum (E) are equal to the energy corresponding to right spectrum (E_W). That is, $\gamma E = E_W$.

5

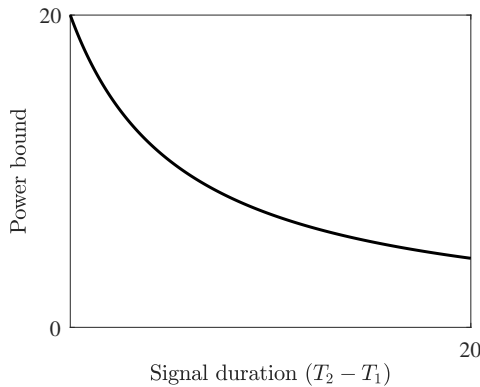


Figure 5.4: The power bound from Theorem 1 for $W = 6.0338$, $\gamma = 0.9$ and $\epsilon \geq 0.1$. The transmit power of any b -modulator with these fundamental parameters must approach zero for long durations.

Proof of Theorem 2. Let $0 < \gamma < 1$ be fixed. We start with finding $W > 0$. For any fixed $b = \mathcal{B}(\mathbf{v})$, the assumptions in Equation (5.14) ensure that $\gamma E \leq E_{W_b}$ for some finite $W_b > 0$. Since the number of bit vectors $\mathbf{v} \in \{0, 1\}^{N_b}$ that can be passed to the modulator is finite, there is only a finite number of nonlinear spectra $b = \mathcal{B}(\mathbf{v})$. Hence, Equation (5.18) is fulfilled if we choose W to be the largest of the W_b .

For any fixed $q = \mathcal{M}(\mathbf{v})$ with corresponding $b = \mathcal{B}(\mathbf{v})$, the transmit power satisfies

$$\begin{aligned}
 P_b &= \frac{1}{T_2 - T_1} \int_{T_1}^{T_2} |q(t)|^2 dt \\
 &\leq \frac{1}{T_2 - T_1} \underbrace{\int_{-\infty}^{\infty} |q(t)|^2 dt}_{=E} \\
 &\stackrel{(5.18)}{\leq} \frac{1}{T_2 - T_1} \frac{E_W}{\gamma} \\
 &= \frac{1}{T_2 - T_1} \frac{-1}{\pi\gamma} \int_{-W}^W \log(1 - |b(\xi)|^2) d\xi \\
 &\stackrel{(5.16)}{\leq} \frac{1}{T_2 - T_1} \frac{-1}{\pi\gamma} \int_{-W}^W \log(1 - (1 - \varepsilon)) d\xi \\
 &= \frac{1}{T_2 - T_1} \frac{-1}{\pi\gamma} 2W \log \varepsilon.
 \end{aligned} \tag{5.19}$$

Since this bound is independent of \mathbf{v} , we obtain Equation (5.17). \square

5

5.3.2. UNIFORM POWER BOUND FOR ARBITRARY GAPS TO SINGULARITY

The bound derived in Theorem 2 describes many practical situations and applies to most of the b -modulators currently seen in the literature. However, the bound is not meaningful in the limit $\varepsilon \rightarrow 0$ as it diverges to infinity. It is interesting to know if we could achieve arbitrary powers in scenarios where the gap to singularity could be made arbitrarily small. In the following Theorem 3, we show that the power will still remain bounded for many typical b -modulators even in the limit $\varepsilon \rightarrow 0$.

Theorem 3. *Let \mathcal{M} be a b -modulator (see Definition 5) such that any $b = \mathcal{B}(\mathbf{v})$ is of the form*

$$b(\xi) = A \sum_{n=-N}^N s_n \Psi(\xi - n\Delta\xi), \quad s_{-N}, \dots, s_N \in S_*, \quad A, \Delta\xi > 0, \tag{5.20}$$

where $S_* \subset \mathbb{C}$ is a finite constellation alphabet and $\Psi \in \mathcal{L}^2(\mathbb{R})$ is a real-analytic carrier waveform with

$$\lim_{\xi \rightarrow \pm\infty} \Psi(\xi) = 0 \quad \text{and} \quad \sup_{k=2,3,\dots} \left\| \frac{d^k \Psi}{d\xi^k} \right\|_{\infty} < \infty. \tag{5.21}$$

The power control factor $A \geq 0$ and the symbols s_n in Equation (5.20) may depend on the bit vector \mathbf{v} . All other quantities are assumed independent of \mathbf{v} . Then, the maximum transmit power of the modulator is bounded as

$$P_{\max} = \max_{q = \mathcal{M}(\mathbf{v}), \mathbf{v} \in \{0,1\}^{N_b}} \frac{1}{T_2 - T_1} \int_{T_1}^{T_2} |q(t)|^2 dt \leq \frac{\bar{E}}{T_2 - T_1} < \infty,$$

where the constant \bar{E} is independent of the power control factors $A = A(\mathbf{v})$ and data symbols $s_n = s_n(\mathbf{v})$ in Equation (5.20), as well as of the temporal domain $[T_1, T_2]$.

The proof of Theorem 3 requires us to establish some lemmas first, which will be given later in this section. Before we proceed to the lemmas, let us first discuss the theorem.

Theorem 3 is formulated such that it is applicable to the carriers typically used in NFDm systems. One of the commonly used carriers is the sinc function [86], [91], [98], [234]

$$\Psi(\xi) = \text{sinc}(\xi) = \begin{cases} 1, & \xi = 0 \\ \frac{\sin(\xi)}{\xi}, & \text{otherwise} \end{cases}. \quad (5.22)$$

The function $\text{sinc}(\xi)$ is real-analytic [243], square-integrable, and decays to zero as $\xi \rightarrow \pm\infty$. To apply Theorem 3, we need to show that $\sup_{k=2,3,\dots} \left\| \frac{d^k}{d\xi^k} \text{sinc}(\xi) \right\|_{\infty} < \infty$. To check this, we first note that $\text{sinc} \in \mathcal{L}^{\infty}(\mathbb{R})$ with $\|\text{sinc}\|_{\infty} = 1$. The Fourier transform of $\text{sinc}(\xi)$ is furthermore a rectangle function,

$$\mathcal{F}\{\text{sinc}\}(\omega) = \begin{cases} \pi, & |\omega| < 1 \\ 0, & \text{otherwise} \end{cases}. \quad (5.23)$$

The set of ω for which the Fourier transform is non-zero thus satisfies $\text{supp } \mathcal{F}\{\text{sinc}\}(\omega) \subset [-1, 1]$, where ‘‘supp’’ is short for support. Then, [244, Theorem 4] tells us that $\left\| \frac{d^k}{d\xi^k} \text{sinc} \right\|_{\infty} \leq 1^k \|\text{sinc}\|_{\infty}$ so that Equation (5.21) is indeed fulfilled. Theorem 3 now tells us that the b -modulator is bounded in transmit power.

Remark 8. *The argument above for showing that the $\text{sinc}(\xi)$ carrier satisfies the conditions in Theorem 3 exploits that the support of its Fourier transform is contained in the interval $[-1, 1]$. For a b -modulator \mathcal{M} as in Theorem 3 where the Fourier transform is not contained in $[-1, 1]$, but in some larger interval $[-\alpha, \alpha]$, the following method can be applied. For any $b = \mathcal{B}(v)$, we define $b_{\alpha}(\xi) = b(\xi/\alpha)$ and $q_{\alpha} := \mathcal{Q}(b_{\alpha})$. By basic properties of the NFT, we have that $q_{\alpha}(t) = \alpha q(\alpha t)$. The b -modulator*

$$\mathcal{M}_{\alpha} : \{0, 1\}^{N_b} \rightarrow \mathcal{L}^2([T_1/\alpha, T_2/\alpha]), \quad v \mapsto q_{\alpha}, \quad (5.24)$$

has the maximum transmit power

$$P_{\alpha} := \frac{1}{T_2/\alpha - T_1/\alpha} \int_{T_1/\alpha}^{T_2/\alpha} |q_{\alpha}(t)|^2 dt \quad (5.25)$$

$$= \frac{\alpha}{T_2 - T_1} \int_{T_1/\alpha}^{T_2/\alpha} |\alpha|^2 |\alpha q(t)|^2 dt \quad (5.26)$$

$$= \frac{|\alpha|^2}{T_2 - T_1} \int_{T_1}^{T_2} |q(\tilde{t})|^2 d\tilde{t} = |\alpha|^2 P, \quad (5.27)$$

where we used the substitution $\tilde{t} := \alpha t$, $d\tilde{t} = dt/\alpha$. The carrier waveform of \mathcal{M}_{α} will by construction have a Fourier transform with support in $[-1, 1]$, so that the argument given above for $\text{sinc}(\xi)$ can again be made. Thus the power P of the modulator \mathcal{M} will also be bounded.

Similar arguments show that raised cosine carriers [90] and flat-top carriers [245] also fulfill the conditions of Theorem 3. Having seen that Theorem 3 is applicable to many b -modulated systems, we now prove two lemmas which we need to prove the theorem.

Lemma 5. Let $b(\xi)$ be any real-analytic function for $\xi \in \mathbb{R}$ with

$$\|b\|_\infty \leq 1, \quad \lim_{\xi \rightarrow \pm\infty} b(\xi) = 0 \quad \text{and} \quad \sup_{k=2,3,\dots} \left\| \frac{d^k}{d\xi^k} b(\xi) \right\|_\infty < \infty. \quad (5.28)$$

Then, the energy contained in any finite interval $[-W, W]$ is finite:

$$E_W := -\frac{1}{\pi} \int_{-W}^W \log(1 - |b(\xi)|^2) d\xi < \infty. \quad (5.29)$$

Proof. Let us set $f(\xi) := b(\xi)\bar{b}(\xi)$, where $\bar{b}(\xi) = b^*(\xi^*)$. If $b(\xi)$ is real-analytic, then $\bar{b}(\xi)$ is real-analytic which implies $f(\xi)$ is also real-analytic [246, Proposition 1.1.4]. For $\xi \in \mathbb{R}$, $f(\xi) := |b(\xi)|^2$. Let ξ_0 denote any spectral singularity (i.e., $|b(\xi_0)| = 1$). We are interested in showing that the contribution of the singularity to the signal energy is finite, i.e.,

$$I := \int_{\xi_0 - \delta/2}^{\xi_0 + \delta/2} \log(1 - f(\xi)) d\xi > -\infty \quad (5.30)$$

for $\delta > 0$ small enough. Since $f(\xi) \in [0, 1]$, this would imply that I is real and not positive. In a interval $(\xi_0 - \delta/2, \xi_0 + \delta/2)$ with $\delta > 0$ small enough, we can write ([246, Corollary 1.1.10])

$$f(\xi) = f(\xi_0) + \frac{f^{(1)}(\xi_0)}{1!} (\xi - \xi_0) + \frac{f^{(2)}(\xi_0)}{2!} (\xi - \xi_0)^2 + \frac{f^{(3)}(\xi_0)}{3!} (\xi - \xi_0)^3 + \dots, \quad (5.31)$$

where $f^{(k)} := \frac{d^k}{d\xi^k} f$. The derivative test tells us that ξ_0 will be an isolated maximum point of f only if $f^{(k)}(\xi_0) = 0$ for $k = 1, \dots, n$ with n odd and $f^{(n+1)}(\xi_0) < 0$. Plugging these into Equation (5.31), we get

$$f(\xi) = 1 + \frac{f^{(n+1)}(\xi_0)}{(n+1)!} (\xi - \xi_0)^{(n+1)} + \frac{f^{(n+2)}(\xi_0)}{(n+2)!} (\xi - \xi_0)^{(n+2)} + \dots \quad (5.32)$$

(Spectral singularities are maximum points because $f(\xi) = |b(\xi)|^2 \leq 1$ for all ξ . They must be isolated because otherwise $f(\xi) = |b(\xi)|^2 = 1$ for all $\xi \in \mathbb{R}$ since f is real-analytic [246, Corollary 1.2.6], which contradicts the second condition in Equation (5.28).)

For showing Equation (5.30), let us define a second integral II by substituting only

the first two non-zero terms of the expansion Equation (5.32) for $f(\xi)$ in Equation (5.30):

$$II := \int_{\xi_0-\delta/2}^{\xi_0+\delta/2} \log \left(1 - \left(1 + \frac{f^{(n+1)}(\xi_0)}{(n+1)!} (\xi - \xi_0)^{(n+1)} \right) \right) d\xi \quad (5.33)$$

$$= \int_{\xi_0-\delta/2}^{\xi_0+\delta/2} \log \left(- \frac{f^{(n+1)}(\xi_0)}{(n+1)!} (\xi - \xi_0)^{(n+1)} \right) d\xi \quad (5.34)$$

$$= \int_{\xi_0-\delta/2}^{\xi_0+\delta/2} \log \left(- \frac{f^{(n+1)}(\xi_0)}{(n+1)!} \right) d\xi + \int_{\xi_0-\delta/2}^{\xi_0+\delta/2} \log \left((\xi - \xi_0)^{(n+1)} \right) d\xi \quad (5.35)$$

$$= \delta \log \left(- \frac{f^{(n+1)}(\xi_0)}{(n+1)!} \right) + \int_{-\delta/2}^{\delta/2} \log \left((\xi^2)^{(n+1)/2} \right) d\xi \quad (5.36)$$

$$= \delta \log \left(- \frac{f^{(n+1)}(\xi_0)}{(n+1)!} \right) + 2 \frac{n+1}{2} \int_0^{\delta/2} \log(\xi^2) d\xi \quad (5.37)$$

$$= \delta \left(\log \left(- \frac{f^{(n+1)}(\xi_0)}{(n+1)!} \right) + (n+1) \left(\log \left(\frac{\delta}{2} \right) - 1 \right) \right). \quad (5.38)$$

5

For any $\delta > 0$, II is real and finite.

Our next goal is to show that the term $III := I - II$ is also finite for $\delta > 0$ small enough. We start by bounding $S := \frac{f^{(n+2)}(\xi_0)}{(n+2)!} + \frac{f^{(n+3)}(\xi_0)}{(n+3)!} (\xi - \xi_0)^1 + \dots$

$$\begin{aligned} |S| &= \left| \frac{f^{(n+2)}(\xi_0)}{(n+2)!} + \frac{f^{(n+3)}(\xi_0)}{(n+3)!} (\xi - \xi_0)^1 + \dots \right| \\ &\leq \left| \frac{f^{(n+2)}(\xi_0)}{(n+2)!} \right| + \left| \frac{f^{(n+3)}(\xi_0)}{(n+3)!} (\xi - \xi_0)^1 \right| + \dots \end{aligned} \quad (5.39)$$

The largest value for $|\xi - \xi_0|$ we have to consider is $|\xi - \xi_0| = \delta/2$. Hence,

$$|S| \leq \left| \frac{f^{(n+2)}(\xi_0)}{(n+2)!} \right| + \left| \frac{f^{(n+3)}(\xi_0)}{(n+3)!2} \right| \delta + \left| \frac{f^{(n+4)}(\xi_0)}{(n+4)!4} \right| \delta^2 + |\dots| \quad (5.40)$$

Since, by assumption $\sup_{k=n+2, n+3, \dots} |f^{(k)}(\xi_0)| < \infty$, we find that

$$\begin{aligned} |S| &\leq \sup_{k=n+2, n+3, \dots} |f^{(k)}(\xi_0)| \left(\frac{1}{(n+2)!} + \frac{\delta}{(n+3)!2} + \frac{\delta^2}{(n+4)!4} + \dots \right) \\ &\rightarrow \sup_{k=n+2, n+3, \dots} |f^{(k)}(\xi_0)| \frac{1}{(n+2)!} \text{ for } \delta \rightarrow 0, \end{aligned} \quad (5.41)$$

Hence,

$$|S| < \frac{2}{(n+2)!} \sup_{k=n+2, n+3, \dots} |f^{(k)}(\xi_0)| < \infty \quad (5.42)$$

for $\delta > 0$ small enough. The integral

$$III := I - II \tag{5.43}$$

$$= \int_{\xi_0 - \delta/2}^{\xi_0 + \delta/2} \log(1 - f(\xi)) - \log\left(-\frac{f^{(n+1)}(\xi_0)}{(n+1)!}(\xi - \xi_0)^{(n+1)}\right) d\xi \tag{5.44}$$

$$= \int_{\xi_0 - \delta/2}^{\xi_0 + \delta/2} \log\left(\frac{1 - f(\xi)}{-\frac{f^{(n+1)}(\xi_0)}{(n+1)!}(\xi - \xi_0)^{(n+1)}}\right) d\xi \tag{5.45}$$

$$= \int_{\xi_0 - \delta/2}^{\xi_0 + \delta/2} \log\left(\frac{-\frac{f^{(n+1)}(\xi_0)}{(n+1)!}(\xi - \xi_0)^{(n+1)} - \frac{f^{(n+2)}(\xi_0)}{(n+2)!}(\xi - \xi_0)^{(n+2)} + \dots}{-\frac{f^{(n+1)}(\xi_0)}{(n+1)!}(\xi - \xi_0)^{(n+1)}}\right) d\xi \tag{5.46}$$

$$= \int_{\xi_0 - \delta/2}^{\xi_0 + \delta/2} \log\left(1 + \frac{\frac{f^{(n+2)}(\xi_0)}{(n+2)!}(\xi - \xi_0)^{(n+2)} + \dots}{\frac{f^{(n+1)}(\xi_0)}{(n+1)!}(\xi - \xi_0)^{(n+1)}}\right) d\xi \tag{5.47}$$

$$= \int_{\xi_0 - \delta/2}^{\xi_0 + \delta/2} \log\left(1 + (\xi - \xi_0) \frac{(n+1)!}{f^{(n+1)}(\xi_0)} S\right) d\xi, \tag{5.48}$$

is, in light of Equation (5.42), thus indeed finite for $\delta > 0$ small enough. Earlier, we already found that the integral II is finite for any $\delta > 0$. However, then, the integral $I = II + III$ has to be finite as well for $\delta > 0$ small enough since the Lebesgue integrable functions form a vector space.

As $f(\xi)$ is real-analytic on \mathbb{R} , there can be only a finite number of points $\xi_1^\circ, \xi_2^\circ, \dots, \xi_M^\circ$ in $[-W, W]$ at which $f(\xi_m^\circ) = 1$ [246, Corollary 1.2.6]. (An infinite sequence $\xi_1^\circ, \xi_2^\circ, \dots$ of spectral singularities in a finite interval $[-W, W]$ would have an accumulation point. Similarly to before, this would imply $f(\xi) = |b(\xi)|^2 = 1$ for all ξ , which contradicts Equation (5.28).) As shown above, we can choose $\delta_1, \delta_2, \dots, \delta_M > 0$ small enough such that

$$I_m := \int_{\xi_m^\circ - \delta_m/2}^{\xi_m^\circ + \delta_m/2} \log(1 - f(\xi)) d\xi > -\infty \tag{5.49}$$

for all m . The set

$$X := [-W, W] \setminus \bigcup_{m=1}^M (\xi_m^\circ - \delta_m/2, \xi_m^\circ + \delta_m/2) \tag{5.50}$$

is compact. The function $f(\xi)$ thus attains a maximum on X , which has to be smaller than one since we removed all points where $f(\xi) = 1$ from X . Summarizing, we find that

$$\begin{aligned} E_W &= -\frac{1}{\pi} \left(\int_X \log(1 - f(\xi)) d\xi + \int_{[-W, W] \setminus X} \log(1 - f(\xi)) d\xi \right) \\ &\leq -\frac{1}{\pi} \left(\underbrace{\min_{\xi \in X} \log(1 - f(\xi))}_{> -\infty} \int_X d\xi + \sum_{m=1}^M I_m \right) < \infty. \end{aligned} \tag{5.51}$$

□

Lemma 1 implies that there is a bound on the energy of $b(\xi)$ when the nonlinear bandwidth is fixed. This leads us to the following lemma.

Lemma 6 (Energy bound for b -modulation). *We consider the b -modulator in Theorem 3. Let $0 < W < \infty$. Then, there exists a finite constant \bar{E}_W such that the energy of any generated $b(\xi)$ in $[-W, W]$ satisfies*

$$E_W = -\frac{1}{\pi} \int_{-W}^W \log(1 - |b(\xi)|^2) d\xi \leq \bar{E}_W. \tag{5.52}$$

The constant \bar{E}_W depends on $\Psi, S_*, N, \Delta\xi$, and W , but is independent of A and the choice of the s_n .

Figure 5.5 presents a graphical illustration of Lemma 6.

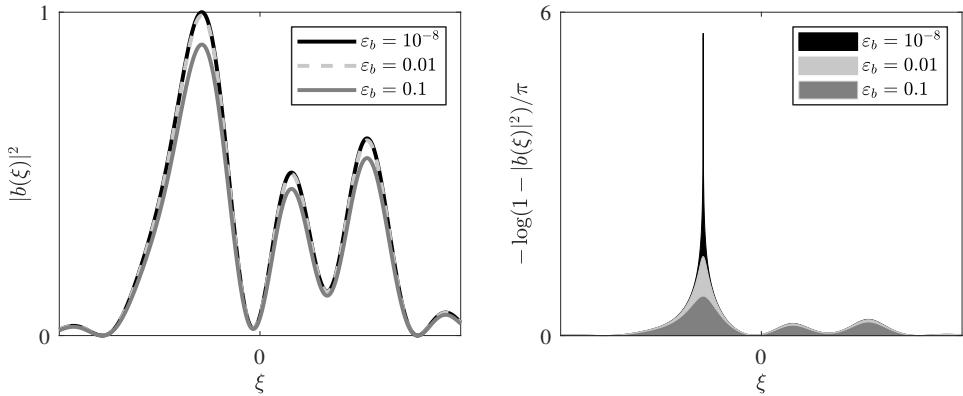


Figure 5.5: The left plot shows $|b(\xi)|^2$ of the form Equation (5.20) for several values of the power control factor A , resulting in different gaps to singularity $\varepsilon_b = 1 - \|b\|_\infty^2$. The right plot shows the corresponding integrand in Equation (5.52). The shaded areas thus represent the signal energy E_W in the shown interval. Lemma 6 tells us that E_W will stay below a finite bound no matter how small the gap to singularity becomes.

Proof of Lemma 6. Let us fix $s_{-N}, \dots, s_N \in S_*$. For a real-analytic $\Psi(\xi)$, $s_n \Psi(\xi)$ will also be real-analytic [246, Proposition 1.1.4]. The sum of real-analytic functions is also real-analytic [246, Proposition 1.1.4], thus $b(\xi)$ will be real-analytic. Since A is assumed admissible, the first condition in Equation (5.28) is fulfilled. The first assumption in Equation (5.21) ensures that the second condition in Equation (5.28) is fulfilled as well. By applying the triangle inequality to Equation (5.28) and using Equation (5.21) to bound the individual summands, we find that also the third condition in Equation (5.28) is fulfilled. Hence, we can apply Lemma 5. The admissible A that results in the largest energy in $[-W, W]$ is given by²

$$A_* = 1 / \max_{\xi \in \mathbb{R}} \left| \sum_{n=-N}^N s_n \Psi(\xi - n\Delta\xi) \right|. \tag{5.53}$$

Lemma 5 shows that E_W is finite for the choice $A = A_*$. Since E_W can only be lower for other admissible choices of A , we have obtained a finite upper bound on E_W for the

²We assume without loss of generality that the denominator in Equation (5.53) is not zero. In such cases, the energy is zero for all $A \geq 0$.

chosen s_{-N}, \dots, s_N that is independent of A . Since our constellation alphabet is finite, there is only a finite number of choices for the s_{-N}, \dots, s_N . By taking the maximum over the upper bounds on E_W for each possible choice, we obtain an upper bound on E_W that is independent of both A and the s_n . \square

Now that we have proved the existence of an energy bound for the modulator in Theorem 3, we shall proceed to prove the power bound.

Proof of Theorem 3. Our first goal is to bound the energy corresponding to the nonlinear spectrum

$$b(\xi) = Ab_0(\xi) := A \sum_{n=-N}^N s_n \Psi(\xi - n\Delta\xi).$$

As the energy is always zero if $\|b_0\|_\infty = 0$, we assume without loss of generality that $\|b_0\|_\infty > 0$. Let us fix an arbitrary $0 < \delta < \|b_0\|_\infty$. Since $\Psi(\xi) \rightarrow 0$ for $\xi \rightarrow \pm\infty$, also $b_0(\xi) \rightarrow 0$ for $\xi \rightarrow \pm\infty$. Hence, we can choose $0 < W < \infty$ such that $|b_0(\xi)| < \delta$ for all $|\xi| > W$. Since $A\|b_0\|_\infty \leq 1$ by Equation (5.14), we obtain

$$A\delta \leq \frac{\delta}{\|b_0\|_\infty} =: \eta < 1$$

for any admissible $A \geq 0$. Choose now \bar{E}_W as in Lemma 6. Then,

$$E = -\frac{1}{\pi} \int_{-\infty}^{\infty} \log(1 - A^2 |b_0|^2) d\xi \quad (5.54)$$

$$= -\frac{1}{\pi} \int_{\mathbb{R} \setminus [-W, W]} \log(1 - A^2 |b_0|^2) d\xi - \frac{1}{\pi} \int_{-W}^W \log(1 - A^2 |b_0|^2) d\xi \quad (5.55)$$

$$\leq -\frac{1}{\pi} \int_{\mathbb{R} \setminus [-W, W]} \log(1 - A^2 |b_0|^2) d\xi + \bar{E}_W \quad (5.56)$$

$$= \frac{1}{\pi} \int_{\mathbb{R} \setminus [-W, W]} \left(A^2 |b_0(\xi)|^2 + \frac{1}{2} A^4 |b_0(\xi)|^4 + \frac{1}{3} A^6 |b_0(\xi)|^6 + \dots \right) d\xi + \bar{E}_W \quad (5.57)$$

$$\leq \frac{1}{\pi} \int_{\mathbb{R} \setminus [-W, W]} A^2 |b_0(\xi)|^2 \left(1 + \frac{1}{2} A^2 \delta^2 + \frac{1}{3} A^4 \delta^4 + \dots \right) d\xi + \bar{E}_W \quad (5.58)$$

$$\leq \frac{1}{\pi} \frac{\eta^2}{\delta^2} (1 + \eta^2 + \eta^4 + \dots) \int_{\mathbb{R} \setminus [-W, W]} |b_0(\xi)|^2 d\xi + \bar{E}_W \quad (5.59)$$

$$\leq \frac{1}{\pi} \frac{\eta^2}{\delta^2} (1 + \eta^2 + \eta^4 + \dots) \|b_0\|_2^2 + \bar{E}_W < \infty. \quad (5.60)$$

In Equation (5.57), the Taylor expansion $-\log(1 - \xi^2) = \xi^2 + \xi^4/2 + \xi^6/3 + \dots$ was used. In Equation (5.59), it was used that $A\delta \leq \eta$. In the last line, we used $\|b_0\|_2^2 < \infty$, which follows from $\Psi \in \mathcal{L}^2(\mathbb{R})$, and $0 < \eta < 1$.

The bound on E in Equation (5.60) is independent of A but still depends on the choice of the $s_{-N}, \dots, s_N \in S_*$ used to construct $b_0(\xi)$. Since the set S_* is finite, there is only a finite number of possible $b_0(\xi)$. Let \bar{E} denote the largest value of Equation (5.60) over all possible $b_0(\xi)$. By construction, \bar{E} is finite and independent of both A and the choice of s_{-N}, \dots, s_N . From Equation (5.14), we find $\int_{T_1}^{T_2} |q(t)|^2 dt \leq \int_{-\infty}^{\infty} |q(t)|^2 dt = E \leq \bar{E}$. Thus, $P \leq \bar{E}/(T_2 - T_1)$ with \bar{E} independent of A , the choice of the s_n , and of course the duration $T_2 - T_1$. \square

5.4. CONCLUSIONS

The NFDN technique of b -modulation has received much attention in the last few years. We have shown, for the first time, that, for b -modulators, the nonlinear bandwidth, signal duration, and power are coupled when, as it is the case in most practical implementations, the gap to singularity is bounded. For fixed nonlinear bandwidth, this results in a bound on the transmit power that decreases with signal duration. This decrease in the transmit power implies that the supremum of the achievable signal-to-noise ratios (SNRs) decreases as the signals become longer. Hence, we established a new factor that contributes to the observed performance degradation of b -modulated systems for long signals [86], [234]. Furthermore, we showed that, even in the presence of spectral singularities, the transmit powers of many b -modulators cannot be made arbitrarily large. The results in this chapter also apply to NFDN systems that modulate the reflection coefficient in fibers with normal dispersion when b is replaced with ρ since the underlying mathematical structure is the same. The cases of b -modulation in normal dispersion fiber and ρ -modulation in anomalous dispersion fiber require further research.

6

SOFTWARE CONTRIBUTIONS: FNFT AND NFDMLAB

CHAPTER ABSTRACT

In this chapter we discuss the structure and features of the FNFT software library and NFDMLab simulation environment. We elaborate on the contributions of this dissertation to them and their impact on NFT based signal processing.

Parts of this chapter are based on the following publications:

S. Wahls et al., “FNFT: A Software Library for Computing Nonlinear Fourier Transforms”, *Journal of Open Source Software*, vol. 3, no. 23, p. 597, 23 Mar. 2018. [171]

S. Wahls et al., *FastNFT/FNFT: Version 0.4.1*, 2020 [247]

M. Brehler et al., “NFDMLab: Simulating Nonlinear Frequency Division Multiplexing in Python”, in *Optical Fiber Communication Conference (OFC) 2019, OSA*, 2019. [154]

6.1. INTRODUCTION

The application of nonlinear Fourier transforms (NFTs) to fiber-optic communications has received significant attention in the recent years. Nonlinear frequency division multiplexing (NFDm) research is a young field and has many interesting research avenues. NFTs as engineering tools also have many other potential applications. Having a software library with tested implementations of NFT algorithms makes it easier to enter the field of NFDm research or test NFTs in other applications. At the beginning of the research for this dissertation, no software library of NFT algorithms was available. Wahls et al. therefore started work on the FNFT software library with the goal of making the fast NFT algorithms developed by him in [128], [139], [161], [248] available to the public. The software library is open access and hosted on GitHub. The releases of the FNFT are furthermore archived on Zenodo [247]. To showcase FNFT we furthermore collaborated with external partners and developed a software environment for the simulation of NFDm systems called NFDMLab [154], which is an open source environment for simulating NFDm systems in Python.

In the following Section 6.2 we will highlight some salient features of the FNFT software library and list contributions from this dissertation. In Section 6.3 we describe structure of NFDMLab and its features.

6.2. FNFT

The initial focus of the FNFT library was to make fast NFT algorithms openly available. However it has been extended to include more conventional algorithms which are often used as reference algorithms. FNFT is written in the C programming language [249] for speed and interoperability and comes with a MATLAB [250] interface. The MATLAB interface allows for faster prototyping in many applications. The source code for FNFT is publicly available on GitHub at <https://github.com/FastNFT/FNFT> under the terms of the GNU General Public License, version 2 [251]. The associated publication [171] can be used to cite the software library. An external collaborator furthermore develops a Python interface for FNFT known as FNFTpy [252]. It simplifies using FNFT in Python projects such as NFDMLab. FNFT can be installed on Windows, Linux or Mac operating systems.

Below we quote, from the README file, the transforms for the NLS equation supported by version 0.4.1 of the library [247].

Forward Transforms

- Vanishing boundary conditions
 - Reflection coefficient and/or scattering coefficients (a and b)
 - Bound states (eigenvalues)
 - Norming constants and/or residues
- (Quasi-)Periodic boundary conditions
 - Main spectrum

- Auxiliary spectrum

Inverse Transforms

- Vanishing boundary conditions
 - Inversion of reflection coefficients, b-scattering coefficients or the inverse Fourier transform of the b-coefficient
 - Bound states (eigenvalues) can be added with arbitrary norming constants or residuals

The library has multiple algorithms for most of the transforms listed above. The algorithms have been implemented using ideas from several publications which are mentioned in the extensive documentation of each of the routines. The documentation can be built from the source code. Alternatively the documentation of the release versions can be found at <https://fastnft.github.io/FNFT/>. All of the algorithms developed in Chapter 2 have been integrated into the library. Ideas from [116], [122], [123], [127], [128], [161], [163] have also been used in algorithms for computing the forward NFTs. The routines for the inverse NFTs are based primarily on the works [90], [120], [139], [253], [254]. Work is ongoing to add more transforms of the NLS and KdV equations. The library is also being extended to support forward NFTs of the 3×3 system [255] associated with the Manakov equation [22] which is a model for propagation of light pulse in a birefringent fiber (see Section 1.3.2).

With multiple contributors and several thousand lines of code, it is necessary to conduct comprehensive testing to ensure robustness of the library. This is accomplished through tests for individual auxiliary routines, complete NFT routines and others to make sure bugs stay fixed. Users can run these tests to ensure that the library has been installed properly. The tests also help developers check if their changes unexpectedly break something. The library also comes with several example files in both C and MATLAB that demonstrate the use of the various features. In Figure 6.1 we can see a screenshot of one of the MATLAB examples which recreates Figure 2.13 from Chapter 2.

The FNFT library has been used by many groups as a toolbox for computing the NFTs [33], [108], [200], [256]–[258]. Some researchers have also used FNFT as a platform to develop new NFT algorithms [259]. FNFT furthermore forms the basis of NFDMLab, which we will discuss in the next section.

6.3. NFDMLAB

NFDMLab is a highly reconfigurable open source simulation environment for NFDMLab systems. It is written in the Python programming language [260]. It can also be controlled using a simple and intuitive graphical environment built on the celebrated Jupyter notebook technology [261]. NFDMLab is available at <https://github.com/FastNFT/NFDMLab>. It is also provided under the terms of the GNU General Public License, version 2. The installation and testing instructions are documented at <https://fastnft.github.io/NFDMLab/>.

The structure of NFDMLab is illustrated in Figure 6.2. The core of NFDMLab realizes all of the simulation and visualization tasks, using standard libraries that can be installed

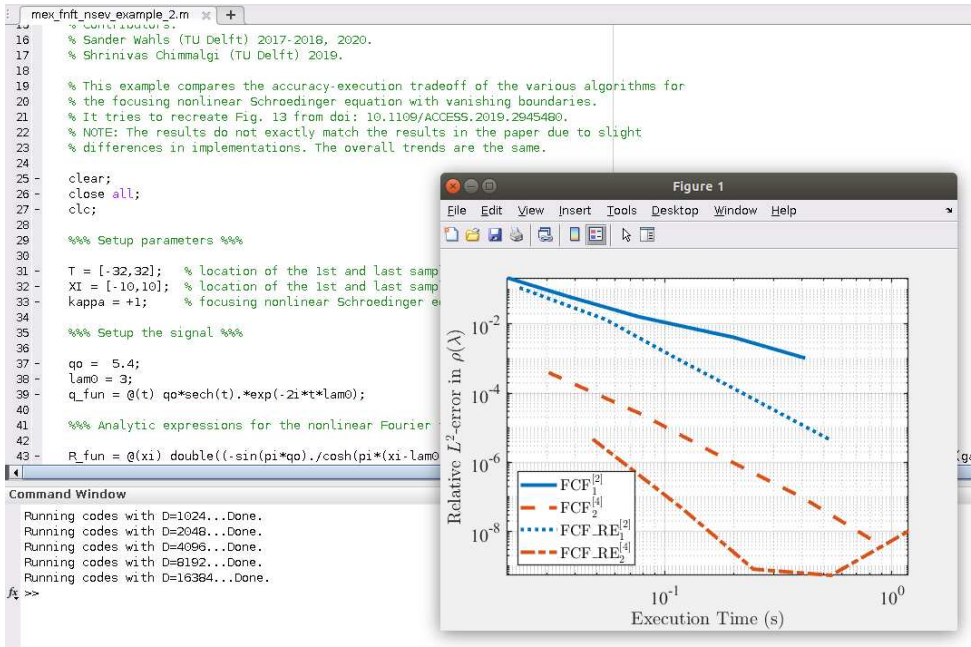


Figure 6.1: Figure 2.13 recreated using `mex_fnft_nsev_example_2` MATLAB example from the FNFT software library.

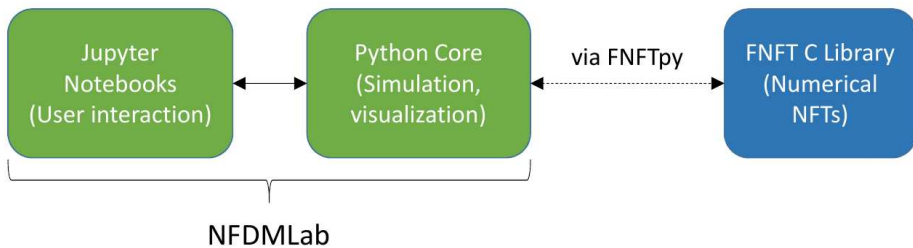


Figure 6.2: Overview of components in NFDMLab. Recreation of Figure 2 in "NFDMLab: Simulating Nonlinear Frequency Division Multiplexing in Python," by M. Brehler et al. (<https://doi.org/10.1364/OFC.2019.M3Z.13>)

with Python's package manager pip (numpy [262], scipy [263], matplotlib [264]). It is organized in modules, each of which provides a typical functionality of a fiber-optic communication system. A typical simulation workflow is depicted in Figure 6.3, where the cyan-colored boxes indicate individual modules. The NFD (de-)modulator modules make use of the FNFT library through FNFTpy.

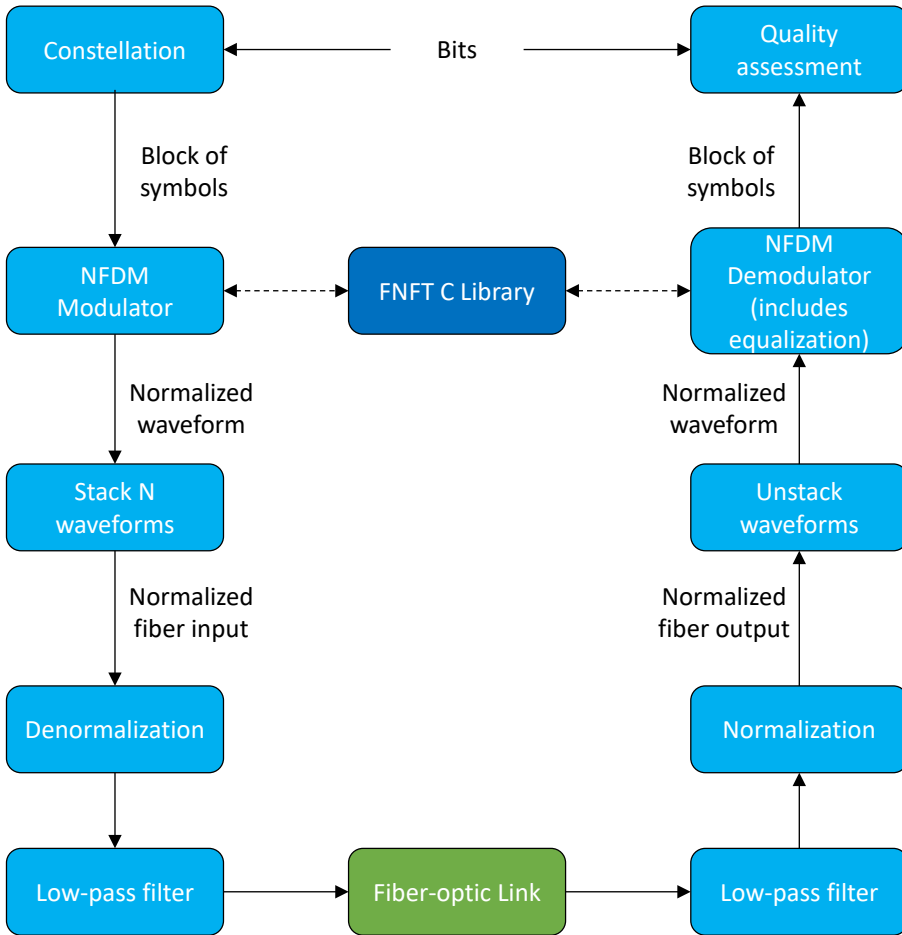


Figure 6.3: Schematic representation of the Python core. The arrows indicate a typical workflow. Recreation of Figure 3 in "NFDMLab: Simulating Nonlinear Frequency Division Multiplexing in Python," by M. Brehler et al. (<https://doi.org/10.1364/OFC.2019.M3Z.13>)

The fiber-optic link is able to simulate EDFA amplification (with flat gain and optionally with noise) and Raman amplification (without noise) in multi-span single-mode fiber links. The nonlinear propagation within the link is simulated using a Python port of the split-step Fourier algorithm SSPROP [265]. The (de-)normalization modules support path-averaging techniques that were discussed in Section 1.3. The quality assessment

module provides standard quality measures such as constellation diagrams or (uncoded) bit error ratios.

NFDMLab comes with NFDM system examples that loosely reproduce experiments reported in the papers [77], [86], [92]. These examples can be run using just Python or with graphical interaction through a Jupyter notebook. In Figure 6.4 we can see a screenshot of the constellation and modulation settings in the Jupyter notebook for the LeArefBuelow2017 example [86]. In Figure 6.5 we can see a fiber input of five blocks and the corresponding fiber output from the same Jupyter notebook. The modular structure of NFDMLab allows researchers to use these examples as a base for their own investigations. The simulations reported in Chapter 4 were performed in NFDMLab. Other colleagues have modified NFDMLab for their work on NFDM systems with special fibers [53] and for fiber parameter estimation [31].

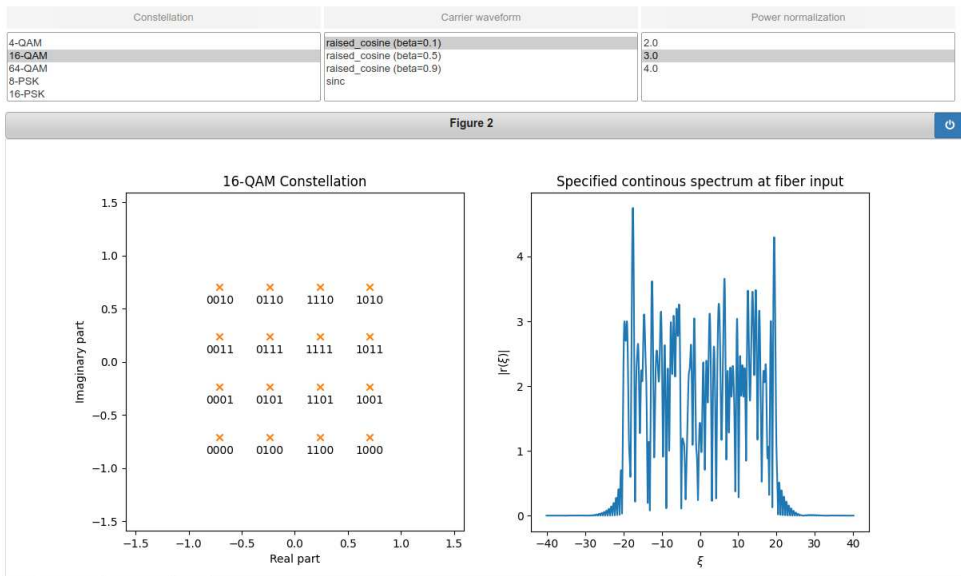


Figure 6.4: Constellation and modulation settings in the Jupyter notebook for the LeArefBuelow2017 example in NFDMLab.

6.4. CONCLUSION

The FNFT open source software library provides several numerical algorithms for nonlinear Fourier transforms. FNFT is quite performant since it is written in the C programming language. It reduces the cost of trying nonlinear Fourier transforms as an engineering tool by providing tested implementations. The NFDMLab simulation environment makes use of the FNFT software library and functions as a platform for developing NFDM systems. It is hoped that these software contributions will also aid in the application of nonlinear Fourier transforms to other interesting potential applications.

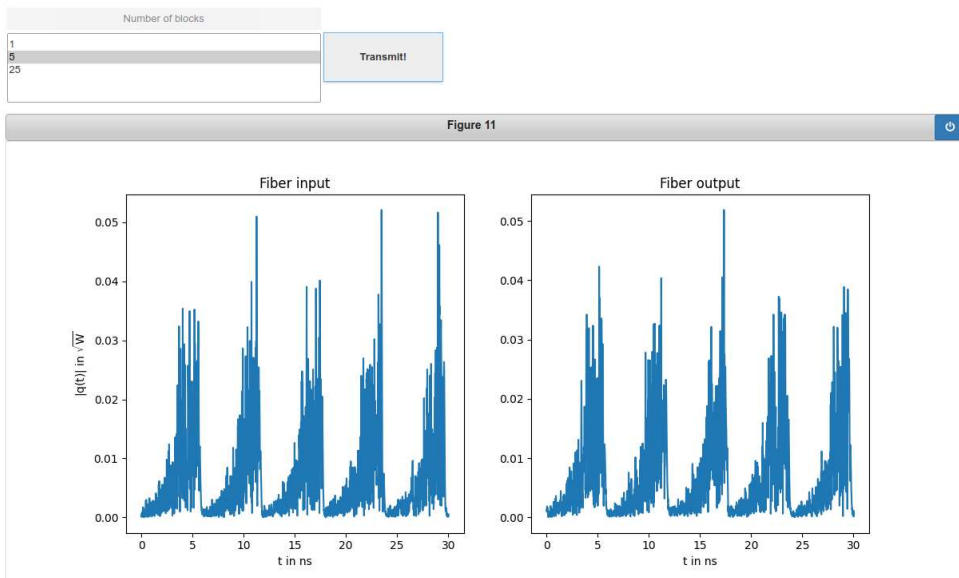


Figure 6.5: Visualization of the fiber input and fiber output in the Jupyter notebook for the LeArefBuelow2017 in NFDMLab.

7

CONCLUSIONS AND RECOMMENDATIONS

CHAPTER ABSTRACT

In this final chapter we evaluate the progress achieved through this dissertation with regards to some practical aspects of nonlinear Fourier transforms for fiber-optic communications. Additionally, some recommendations for further research are discussed.

7.1. CONCLUSIONS

The principal objective of this dissertation was to understand and address numerical issues that currently complicate the practical use of nonlinear Fourier transform-based fiber-optic communication systems. To achieve this objective, we developed new fast numerical algorithms that addressed different open issues in the literature. The first are novel fast forward NFT algorithms for the NLSE with vanishing boundary conditions that offered a vastly better accuracy-complexity trade-off than prior methods. The second is an inverse NFT algorithm for the periodic NFT that is able to generate high genus solutions. The investigation of issues with fiber-optic b -modulation communication systems that were attributed to numerical problems in the literature furthermore lead to the discovery of a novel coupling between nonlinear bandwidth, duration and power for these systems that has no analog in the linear case. Our results explain the observed issues with b -modulation systems without resorting to numerical problems. In the following, we will briefly recapitulate the motivation, progress and likely impact of the work for each of the chapters.

Nonlinear Fourier transforms do not obey the principle of superposition. Hence we require numerical algorithms to compute the transforms of more complicated signals. In Chapter 2, we proposed new higher-order fast forward NFT algorithms based on a special class of exponential integrators. The proposed algorithms were compared with existing algorithms in numerical examples and were shown to have a much better complexity-accuracy trade-off. It was the first time that NFT algorithms were compared from this point of view. Furthermore, for the first time in literature we presented an exact count of the various basic mathematical operations for a fast NFT algorithm. Since the publication of these algorithms, several other higher-order algorithms have been proposed [123], [131], [259], [266] in the literature with focus on complexity-accuracy trade-off. We have observed that the algorithms proposed in this dissertation remain fairly competitive to these newer algorithms. The authors in [266] have suggested that sixth-order methods may offer the best complexity-accuracy trade-off and increasing the order further may not be favorable. The development of new algorithms for computing the nonlinear Fourier spectrum is ongoing.

The majority of NFT based fiber-optic communication systems employ the vanishing NFT. Systems based on the periodic NFT, which is closer to the commonly used linear OFDM, have been investigated as well. However, currently proposed systems are not competitive in terms of spectral efficiency. One reason for this lack of performance is the use of low genus solutions. Higher performance may be achievable with high genus solutions, but the numerical complexity of existing inverse NFT methods grows rapidly with the genus. Finite-genus solutions are typically computed by either the algebrogeometric approach or by solving Riemann-Hilbert problems. We studied the algebrogeometric approach in which finite-genus solutions are represented in terms of the Riemann theta function. The Riemann theta function is a multi-dimensional Fourier series whose numerical computation is challenging due to the curse of dimensionality. This limits the genus of the solutions used in periodic NFT based systems. In Chapter 3 we investigated multiple conventional methods theoretically and concluded that none of them are suitable for the high-dimensional regime. We then proposed to consider low-rank tensor and hyperbolic cross-based techniques. We evaluated the proposed al-

gorithms in multiple numerical benchmarks. The algorithm based on the hyperbolic cross index set allows for the computation of high-genus solutions that were so far out of reach. This removes one major hurdle in the development of periodic NFT based systems with possibly higher performance.

Of the several NFDN system architectures proposed in the literature, b -modulated systems have emerged as the currently preferred design. b -modulated systems allow for control over the signal duration and have been shown to be more robust to noise. The system spectral efficiency can be theoretically increased by utilizing signals significantly longer than the channel memory. However in practice, bit errors were observed to increase which outweighed the advantage of using longer signals. This observation has been typically attributed to increased numerical signal-noise interactions in the NFT for longer signals [86], [98], [233], [234]. In Chapter 4 we considered the b -modulated system from [92] and studied the transmission of long signals. We were able to derive a theoretical bound on the transmit power of the signals which decreases with signal duration for fixed bandwidth. This was the first instance where such a bound on the transmit power of an NFDN system was derived. Long signals are thus low power signals which “drown” in noise. This explains the observed increase in bit errors and ultimately drop in performance even in the absence of numerical issues. Using an improved numerical algorithm for the inverse NFT, we performed numerical simulations in which we achieved transmit powers close to the theoretical bound. This suggested that in our setup, numerical signal-noise interactions was not the major limiting factor. In other numerical simulations we found that the b -modulator from [99], which makes use of a special nonlinear mapping for the b -coefficient with the hope of avoiding the energy barrier, also suffers from similar limitations even though the exact argument used in our analysis does not apply to that case.

We had hoped that the analysis and new insights from Chapter 4 would allow us to design improved b -modulation systems that can generate long signals with high powers. However, we soon discovered that the bound on the transmit power was a fundamental limitation of a wide class of b -modulated systems in the anomalous dispersion regime. We discussed this in detail in Chapter 5. We showed that for b -modulators the nonlinear bandwidth, signal duration and power are coupled when the gap to singularity is bounded, i.e. $1 - |b(\xi)| \geq \epsilon \forall \xi \in \mathbb{R}$. For a fixed nonlinear bandwidth this results in a bound on the transmit power that decreases with signal duration. Furthermore, we showed that even in the presence of spectral singularities, that is points at which $|b(\xi)| = 1$, the transmit powers of many b -modulators cannot be made arbitrarily large. The underlying mathematical structure for the reflection coefficient ρ in the normal dispersion regime is the same as the b -coefficient in the anomalous regime. Hence the bounds also apply to NFDN systems that modulate the reflection coefficient ρ in fibers with normal dispersion [87], [97]. Such a fundamental limitation on the transmit power is not seen in the case of conventional linear modulators. It is a limitation imposed on NFDN system design by the structure of the nonlinear Fourier transform. Knowing the presence of such constraints is crucial for the development of spectrally efficient NFDN systems.

The field of NFDN research is still young with many interesting research opportunities. Having a software library with tested implementations of NFT algorithms makes it easier for more researchers to enter the field of NFDN research. At the beginning of

research for this dissertation, no software library of NFT algorithms was available. Wahls et al. therefore started work on the open source software library FNFT. In Chapter 6 we explained the structure and some salient features of the FNFT library. The library has already made a mark with many researchers using it as a toolbox [33], [108], [257] and others as a platform for developing their own algorithms [259]. We also reviewed the structure of the simulation environment NFDMLab, which is a Python based environment for simulating NFD systems that utilizes FNFT. Although several simulation environments existed for simulation of fiber-optic communication systems, many were not developed on freely available platforms and integrating NFT based designs in them was challenging. NFDMLab was developed with the goal of reducing the barrier to NFD research. It is open source and has a modular structure which allows for straightforward extension to new system designs. NFDMLab also helps to replicate results from the literature and already includes multiple examples. New system designs developed using NFDMLab can be replicated with low effort or shared effortlessly across research groups which will hopefully be beneficial to the field of NFD research.

7.2. RECOMMENDATIONS FOR FUTURE WORK

In this section we discuss some of the possible directions for further research on the topic of nonlinear Fourier transforms in fiber-optic communications.

Fast higher-order inverse NFT algorithms for vanishing signals Although we have made significant progress in the area of forward NFT algorithms, there hasn't been adequate progress in case of the inverse NFT algorithms. It was already shown by Wahls and Poor in [138] that their idea of employing rational polynomial function approximations can be used for developing fast inverse NFT algorithms for vanishing signals. However, it has not been possible so far to extend the idea of higher-order methods to the case of fast inverse algorithms. Being able to compute the inverse NFT efficiently is as important as the forward case and hence the lack of good algorithms can be a limiting factor in engineering applications. The development of fast higher-order inverse NFT algorithms is thus an interesting open topic for future research.

Spectrally efficient NFD systems While the spectral efficiencies of NFD systems have risen to greater than 50% of highly tuned conventional systems, more work remains to be done. We now know that the transmit power of b -modulators in the anomalous dispersion regime, which modulate only the continuous spectrum, is limited. The limitation on the transmit power translates to a limitation on the achievable spectral efficiency. One possible approach to overcome this limitation is the use of strategically placed solitons that can help increase the signal power without losing the time-limitedness of the signals [91]. It may also be possible that b -modulated systems in the normal dispersion regime and ρ -modulated systems in the anomalous dispersion regime do not have such a limitation. These are two of the many possible directions in which NFD system design can be explored.

NFD systems based on NFTs for other boundary conditions Almost all the research on NFT based fiber-optic communications is centered around vanishing signals with only a handful of works employing periodic signals. However, the theory of the nonlinear Fourier transforms for the nonlinear Schrödinger equation extends to signals with other boundary conditions. One of the notable ones is the non-zero boundary condition

[241]. It would be interesting to explore the possibility of utilizing such signals for communication. The development of NFT algorithms for the specific boundary conditions would have to be a precursor for research in that direction.

BIBLIOGRAPHY

- [1] A. Dominguez, “Highlights in the History of the Fourier Transform [Retrospectroscope]”, *IEEE Pulse*, vol. 7, no. 1, pp. 53–61, Jan. 2016. DOI: [10.1109/mpul.2015.2498500](https://doi.org/10.1109/mpul.2015.2498500).
- [2] D. Duffy, *Transform methods for solving partial differential equations*. Boca Raton: Chapman & Hall/CRC, 2004, ISBN: 9781584884514.
- [3] “Soliton wave receives crowd of admirers”, *Nature*, vol. 376, no. 6539, pp. 373–373, Aug. 1995. DOI: [10.1038/376373b0](https://doi.org/10.1038/376373b0).
- [4] N. J. Zabusky and M. D. Kruskal, “Interaction of "Solitons" in a Collisionless Plasma and the Recurrence of Initial States”, *Physical Review Letters*, vol. 15, no. 6, pp. 240–243, Aug. 1965. DOI: [10.1103/physrevlett.15.240](https://doi.org/10.1103/physrevlett.15.240).
- [5] N. Zabusky and M. Porter, “Soliton”, *Scholarpedia*, vol. 5, no. 8, p. 2068, 2010, revision #186585. DOI: [10.4249/scholarpedia.2068](https://doi.org/10.4249/scholarpedia.2068).
- [6] M. Williams, *Solitons : interactions, theoretical and experimental challenges and perspectives*. New York: Nova Publishers, 2013, ISBN: 9781626182349.
- [7] A.-M. Wazwaz, *Partial Differential Equations and Solitary Waves Theory*. Springer-Verlag GmbH, Aug. 2009, 700 pp., ISBN: 3642002501. [Online]. Available: https://www.ebook.de/de/product/8322135/abdul_majid_wazwaz_partial_differential_equations_and_solitary_waves_theory.html.
- [8] C. S. Gardner, J. M. Greene, M. D. Kruskal, and R. M. Miura, “Method for Solving the Korteweg-de Vries Equation”, *Physical Review Letters*, vol. 19, no. 19, pp. 1095–1097, Nov. 1967. DOI: [10.1103/physrevlett.19.1095](https://doi.org/10.1103/physrevlett.19.1095).
- [9] P. D. Lax, “Integrals of nonlinear equations of evolution and solitary waves”, *Communications on Pure and Applied Mathematics*, vol. 21, no. 5, pp. 467–490, Sep. 1968. DOI: [10.1002/cpa.3160210503](https://doi.org/10.1002/cpa.3160210503).
- [10] V. E. Zakharov and A. B. Shabat, “Exact Theory of Two-Dimensional Self-Focusing and One-Dimensional Self-Modulation of Waves in Nonlinear Media”, *Journal of Theoretical Physics*, vol. 34, no. 1, p. 62, Jan. 1972. [Online]. Available: <http://www.jetp.ac.ru/cgi-bin/e/index/e/34/1/p62?a=list>.
- [11] M. J. Ablowitz, D. J. Kaup, A. C. Newell, and H. Segur, “The Inverse Scattering Transform-Fourier Analysis for Nonlinear Problems”, *Studies in Applied Mathematics*, vol. 53, no. 4, pp. 249–315, Dec. 1974. DOI: [10.1002/sapm1974534249](https://doi.org/10.1002/sapm1974534249).
- [12] S. Manakov, “Complete Integrability and Stochastization of Discrete Dynamic Systems”, *Journal of Experimental and Theoretical Physics*, vol. 40, no. 2, pp. 269–274, Aug. 1975.

- [13] H. Flaschka, “The Toda lattice. II. Existence of integrals”, *Physical Review B*, vol. 9, no. 4, pp. 1924–1925, Feb. 1974. DOI: [10.1103/physrevb.9.1924](https://doi.org/10.1103/physrevb.9.1924).
- [14] —, “On the Toda Lattice. II: Inverse-Scattering Solution”, *Progress of Theoretical Physics*, vol. 51, no. 3, pp. 703–716, Mar. 1974, ISSN: 0033-068X. DOI: [10.1143/ptp.51.703](https://doi.org/10.1143/ptp.51.703).
- [15] M. Ablowitz and B. Prinari, “Nonlinear Schrodinger systems: continuous and discrete”, *Scholarpedia*, vol. 3, no. 8, p. 5561, 2008, revision #137230. DOI: [10.4249/scholarpedia.5561](https://doi.org/10.4249/scholarpedia.5561).
- [16] M. J. Ablowitz and H. Segur, “Solitons and the Inverse Scattering Transform”, in *Solitons and the Inverse Scattering Transform*, Society for Industrial and Applied Mathematics, Jan. 1981. DOI: [10.1137/1.9781611970883](https://doi.org/10.1137/1.9781611970883).
- [17] G. P. Agrawal, *Nonlinear fiber optics*. San Diego: Academic Press, 2001, ISBN: 9780080479743.
- [18] A. Zvezdin and A. Popkov, “Contribution to the nonlinear theory of magnetostatic spin waves”, *Sov. Phys. JETP*, vol. 2, no. 350, p. 150, 1983.
- [19] M. Chen, M. A. Tsankov, J. M. Nash, and C. E. Patton, “Backward-volume-wave microwave-envelope solitons in yttrium iron garnet films”, *Physical Review B*, vol. 49, no. 18, pp. 12 773–12 790, May 1994. DOI: [10.1103/physrevb.49.12773](https://doi.org/10.1103/physrevb.49.12773).
- [20] V. E. Zakharov *et al.*, “Collapse of Langmuir waves”, *Sov. Phys. JETP*, vol. 35, no. 5, pp. 908–914, 1972.
- [21] G. P. Agrawal, *Fiber-Optic Communication Systems*, 4th. Hoboken, N.J., USA: Wiley, 2013, ISBN: 978-0-470-50511-3.
- [22] S. V. Manakov, “On the theory of two-dimensional stationary self-focusing of electromagnetic waves”, *Journal of Experimental and Theoretical Physics*, vol. 38, no. 2, pp. 248–253, Feb. 1974.
- [23] A. R. Its and V. P. Kotlyarov, “Explicit Formulas for Solutions of the Schrödinger nonlinear Equation”, *Doklady Akad.Nauk Ukrainian SSR, ser.A*, no. 10, pp. 965–968, 1976, English transl. in arXiv:1401.4445v1[nlin:SI].
- [24] V. Kotlyarov, “Periodic problem for the Schrödinger nonlinear equation”, *Voprosy Matematicheskoi Fiziki i Funkcionalnogo Analiza, 1, Naukova Dumka, Kiev*, pp. 121–131, 1976, English transl. in arXiv:1401.4445v1[nlin:SI].
- [25] G. Biondini and A. Bui, “On the Nonlinear Schrödinger Equation on the Half Line with Homogeneous Robin Boundary Conditions”, *Studies in Applied Mathematics*, vol. 129, no. 3, pp. 249–271, May 2012. DOI: [10.1111/j.1467-9590.2012.00553.x](https://doi.org/10.1111/j.1467-9590.2012.00553.x).
- [26] G. Biondini and G. Kovačič, “Inverse scattering transform for the focusing nonlinear Schrödinger equation with nonzero boundary conditions”, *Journal of Mathematical Physics*, vol. 55, no. 3, p. 031 506, Mar. 2014. DOI: [10.1063/1.4868483](https://doi.org/10.1063/1.4868483).
- [27] G. Biondini, E. Fagerstrom, and B. Prinari, “Inverse scattering transform for the defocusing nonlinear Schrödinger equation with fully asymmetric non-zero boundary conditions”, *Physica D: Nonlinear Phenomena*, vol. 333, pp. 117–136, Oct. 2016. DOI: [10.1016/j.physd.2016.04.003](https://doi.org/10.1016/j.physd.2016.04.003).

- [28] A. S. Fokas and A. R. Its, “The Nonlinear Schrödinger Equation on the Interval”, *Journal of Physics A: Mathematical and General*, vol. 37, no. 23, pp. 6091–6114, May 2004. DOI: [10.1088/0305-4470/37/23/009](https://doi.org/10.1088/0305-4470/37/23/009).
- [29] A. S. Fokas, A. R. Its, and L.-Y. Sung, “The nonlinear Schrödinger equation on the half-line”, *Nonlinearity*, vol. 18, no. 4, pp. 1771–1822, May 2005. DOI: [10.1088/0951-7715/18/4/019](https://doi.org/10.1088/0951-7715/18/4/019).
- [30] A. R. Osborne, *Nonlinear ocean waves and the inverse scattering transform*. Elsevier, AP, 2010, ISBN: 9780080925103.
- [31] P. de Koster and S. Wahls, “Dispersion and Nonlinearity Identification for Single-Mode Fibers Using the Nonlinear Fourier Transform”, *Journal of Lightwave Technology*, vol. 38, no. 12, pp. 3252–3260, Jun. 2020. DOI: [10.1109/jlt.2020.2979285](https://doi.org/10.1109/jlt.2020.2979285).
- [32] S. Wahls, M. Bruehl, Y.-M. Fan, and C.-J. Huang, “Nonlinear Fourier Analysis of Free-Surface Buoy Data Using the Software Library FNFT”, in *Volume 6B: Ocean Engineering*, American Society of Mechanical Engineers, Aug. 2020. DOI: [10.1115/omae2020-18676](https://doi.org/10.1115/omae2020-18676).
- [33] S. Sugavanam, M. K. Kopae, J. Peng, J. E. Prilepsky, and S. K. Turitsyn, “Analysis of laser radiation using the Nonlinear Fourier transform”, *Nature Communications*, vol. 10, no. 1, Dec. 2019. DOI: [10.1038/s41467-019-13265-4](https://doi.org/10.1038/s41467-019-13265-4).
- [34] S. K. Turitsyn, I. S. Chekhovskoy, and M. P. Fedoruk, “Nonlinear Fourier transform for analysis of optical spectral combs”, *Phys. Rev. E*, vol. 103, p. L020202, 2 Feb. 2021. DOI: [10.1103/PhysRevE.103.L020202](https://doi.org/10.1103/PhysRevE.103.L020202). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.103.L020202>.
- [35] 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”, *Optica*, vol. 4, no. 3, pp. 307–322, Mar. 2017. DOI: [10.1364/optica.4.000307](https://doi.org/10.1364/optica.4.000307).
- [36] S. K. Turitsyn, I. S. Chekhovskoy, and M. P. Fedoruk, “Nonlinear Fourier transform for characterization of the coherent structures in optical microresonators”, *Optics Letters*, vol. 45, no. 11, p. 3059, May 2020. DOI: [10.1364/ol.390630](https://doi.org/10.1364/ol.390630).
- [37] I. Chekhovskoy, O. Shtyrina, M. P. Fedoruk, S. Medvedev, and S. Turitsyn, “Characterization of coherent structures in dissipative systems using nonlinear Fourier transform”, in *Nonlinear Optics and its Applications 2020*, A. C. Peacock, N. G. R. Broderick, and J. M. Dudley, Eds., SPIE, Apr. 2020. DOI: [10.1117/12.2555076](https://doi.org/10.1117/12.2555076).
- [38] L. F. Leon Takhtajan, *Hamiltonian Methods in the Theory of Solitons*. Springer-Verlag GmbH, Aug. 10, 2007, 592 pp., ISBN: 9783540699699. [Online]. Available: <https://www.springer.com/gp/book/9783540698432>.
- [39] B. A. Dubrovin, “Theta functions and non-linear equations”, *Russian Mathematical Surveys*, vol. 36, no. 2, pp. 11–92, Apr. 1981. DOI: [10.1070/rm1981v036n02abeh002596](https://doi.org/10.1070/rm1981v036n02abeh002596).
- [40] Y.-C. Ma and M. J. Ablowitz, “The Periodic Cubic Schrödinger Equation”, *Studies in Applied Mathematics*, vol. 65, no. 2, pp. 113–158, Oct. 1981. DOI: [10.1002/sapm1981652113](https://doi.org/10.1002/sapm1981652113).

- [41] E. D. Belokolos, V. Z. Enol'skii, A. R. Its, and V. B. Matveev, *Algebro-geometric approach to nonlinear integrable equations*. Berlin New York: Springer-Verlag, 1994, ISBN: 9783540502654.
- [42] E. R. Tracy and H. H. Chen, "Nonlinear self-modulation: An exactly solvable model", *Physical Review A*, vol. 37, no. 3, pp. 815–839, Feb. 1988. DOI: [10.1103/physreva.37.815](https://doi.org/10.1103/physreva.37.815).
- [43] V. B. Matveev, "30 years of finite-gap integration theory", *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 366, no. 1867, pp. 837–875, Jun. 2007. DOI: [10.1098/rsta.2007.2055](https://doi.org/10.1098/rsta.2007.2055).
- [44] V. Kotlyarov and D. Shepelsky, "Planar unimodular Baker-Akhiezer function for the nonlinear Schrödinger equation", *Annals of Mathematical Sciences and Applications*, vol. 2, no. 2, pp. 343–384, 2017. DOI: [10.4310/amsa.2017.v2.n2.a6](https://doi.org/10.4310/amsa.2017.v2.n2.a6). [Online]. Available: <https://doi.org/10.4310/amsa.2017.v2.n2.a6>.
- [45] B. Deconinck, A. S. Fokas, and J. Lenells, "The implementation of the unified transform to the nonlinear Schrödinger equation with periodic initial conditions", *Letters in Mathematical Physics*, vol. 111, no. 1, Feb. 2021. DOI: [10.1007/s11005-021-01356-7](https://doi.org/10.1007/s11005-021-01356-7).
- [46] P. G. Grinevich, "Approximation theorem for the self-focusing Nonlinear Schrödinger Equation and for the periodic curves in R^3 ", *Physica D: Nonlinear Phenomena*, vol. 152-153, pp. 20–27, May 2001. DOI: [10.1016/s0167-2789\(01\)00155-5](https://doi.org/10.1016/s0167-2789(01)00155-5).
- [47] *Riemann surface*, Jul. 2021. [Online]. Available: https://en.wikipedia.org/wiki/Riemann_surface.
- [48] *Creative Commons Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0) License*, WebPage. [Online]. Available: <https://creativecommons.org/licenses/by-sa/3.0/>.
- [49] B. Deconinck, M. Heil, A. Bobenko, M. van Hoeij, and M. Schmies, "Computing Riemann theta functions", *Mathematics of Computation*, vol. 73, no. 247, pp. 1417–1443, Dec. 2003. DOI: [10.1090/s0025-5718-03-01609-0](https://doi.org/10.1090/s0025-5718-03-01609-0).
- [50] G. P. Agrawal, "Optical Communication: Its History and Recent Progress", in *Optics in Our Time*, Springer International Publishing, 2016, pp. 177–199. DOI: [10.1007/978-3-319-31903-2_8](https://doi.org/10.1007/978-3-319-31903-2_8).
- [51] A. Hasegawa and Y. Kodama, "Guiding-center soliton in optical fibers", *Optics Letters*, vol. 15, no. 24, p. 1443, Dec. 1990. DOI: [10.1364/ol.15.001443](https://doi.org/10.1364/ol.15.001443).
- [52] V. N. Serkin and T. L. Belyaeva, "Optimal control of optical soliton parameters: Part 1. The Lax representation in the problem of soliton management", *Quantum Electronics*, vol. 31, no. 11, pp. 1007–1015, Nov. 2001. DOI: [10.1070/qe2001v031n11abeh002093](https://doi.org/10.1070/qe2001v031n11abeh002093).
- [53] V. Bajaj, S. Chimmalgi, V. Aref, and S. Wahls, "Exact NFDm Transmission in the Presence of Fiber-Loss", *Journal of Lightwave Technology*, vol. 38, no. 11, pp. 3051–3058, Jun. 2020. DOI: [10.1109/jlt.2020.2984041](https://doi.org/10.1109/jlt.2020.2984041).

- [54] M. Mazur, J. Schröder, A. Lorences-Riesgo, T. Yoshida, M. Karlsson, and P. A. Andrekson, “Overhead-optimization of pilot-based digital signal processing for flexible high spectral efficiency transmission”, *Optics Express*, vol. 27, no. 17, p. 24 654, Aug. 2019. DOI: [10.1364/oe.27.024654](https://doi.org/10.1364/oe.27.024654).
- [55] M. Roser, H. Ritchie, and E. Ortiz-Ospina, “Internet”, *Our World in Data*, 2015, <https://ourworldindata.org/internet>.
- [56] E. Agrell, M. Karlsson, A. R. Chraplyvy, D. J. Richardson, P. M. Krummrich, P. Winzer, K. Roberts, J. K. Fischer, S. J. Savory, B. J. Eggleton, M. Secondini, F. R. Kschischang, A. Lord, J. Prat, I. Tomkos, J. E. Bowers, S. Srinivasan, M. Brandt-Pearce, and N. Gisin, “Roadmap of Optical Communications”, *Journal of Optics*, vol. 18, no. 6, p. 063 002, May 2016. DOI: [10.1088/2040-8978/18/6/063002](https://doi.org/10.1088/2040-8978/18/6/063002).
- [57] A. Hasegawa and F. Tappert, “Transmission of stationary nonlinear optical pulses in dispersive dielectric fibers. I. Anomalous dispersion”, *Applied Physics Letters*, vol. 23, no. 3, pp. 142–144, Aug. 1973. DOI: [10.1063/1.1654836](https://doi.org/10.1063/1.1654836).
- [58] L. F. Mollenauer, R. H. Stolen, and J. P. Gordon, “Experimental Observation of Picosecond Pulse Narrowing and Solitons in Optical Fibers”, *Physical Review Letters*, vol. 45, no. 13, pp. 1095–1098, Sep. 1980. DOI: [10.1103/physrevlett.45.1095](https://doi.org/10.1103/physrevlett.45.1095).
- [59] L. F. Mollenauer and K. Smith, “Demonstration of Soliton Transmission Over More Than 4000 Km in Fiber with Loss Periodically Compensated by Raman Gain”, *Optics Letters*, vol. 13, no. 8, p. 675, Aug. 1988. DOI: [10.1364/ol.13.000675](https://doi.org/10.1364/ol.13.000675).
- [60] M. Nakazawa, “Solitons for breaking barriers to terabit/second WDM and OTDM transmission in the next millennium”, *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 6, no. 6, pp. 1332–1343, Nov. 2000. DOI: [10.1109/2944.902187](https://doi.org/10.1109/2944.902187).
- [61] A. Hasegawa, “An historical review of application of optical solitons for high speed communications”, *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 10, no. 3, pp. 475–485, Sep. 2000. DOI: [10.1063/1.1286914](https://doi.org/10.1063/1.1286914).
- [62] R. Ganapathy, K. Porsezian, A. Hasegawa, and V. N. Serkin, “Soliton Interaction Under Soliton Dispersion Management”, *IEEE Journal of Quantum Electronics*, vol. 44, no. 4, pp. 383–390, Apr. 2008. DOI: [10.1109/jqe.2007.914778](https://doi.org/10.1109/jqe.2007.914778).
- [63] A. Hasegawa and T. Nyu, “Eigenvalue Communication”, *Journal of Lightwave Technology*, vol. 11, no. 3, pp. 395–399, Mar. 1993. DOI: [10.1109/50.219570](https://doi.org/10.1109/50.219570).
- [64] E. G. Turitsyna and S. K. Turitsyn, “Digital signal processing based on inverse scattering transform”, *Optics Letters*, vol. 38, no. 20, p. 4186, Oct. 2013. DOI: [10.1364/ol.38.004186](https://doi.org/10.1364/ol.38.004186).
- [65] J. E. Prilepsky, S. A. Derevyanko, and S. K. Turitsyn, “Nonlinear spectral management: Linearization of the lossless fiber channel”, *Optics Express*, vol. 21, no. 20, p. 24 344, Oct. 2013. DOI: [10.1364/oe.21.024344](https://doi.org/10.1364/oe.21.024344).

- [66] J. E. Prilepsky, S. A. Derevyanko, K. J. Blow, I. Gabitov, and S. K. Turitsyn, “Non-linear Inverse Synthesis and Eigenvalue Division Multiplexing in Optical Fiber Channels”, *Physical Review Letters*, vol. 113, no. 1, p. 013901, 1 Jul. 2014. DOI: [10.1103/physrevlett.113.013901](https://doi.org/10.1103/physrevlett.113.013901).
- [67] H. Terauchi and A. Maruta, “Eigenvalue modulated optical transmission system based on digital coherent technology”, in *2013 18th OptoElectronics and Communications Conference held jointly with 2013 International Conference on Photonics in Switching (OECC/PS)*, 2013, pp. 1–2.
- [68] M. I. Yousefi and F. R. Kschischang, “Information Transmission Using the Non-linear Fourier Transform, Part I: Mathematical Tools”, *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4312–4328, Jul. 2014. DOI: [10.1109/tit.2014.2321143](https://doi.org/10.1109/tit.2014.2321143).
- [69] —, “Information Transmission Using the Nonlinear Fourier Transform, Part II: Numerical Methods”, *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4329–4345, Jul. 2014. DOI: [10.1109/tit.2014.2321151](https://doi.org/10.1109/tit.2014.2321151).
- [70] —, “Information Transmission Using the Nonlinear Fourier Transform, Part III: Spectrum Modulation”, *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4346–4369, Jul. 2014. DOI: [10.1109/tit.2014.2321155](https://doi.org/10.1109/tit.2014.2321155).
- [71] M. I. Yousefi, “Information Transmission Using the Nonlinear Fourier Transform”, Ph.D. dissertation, University of Toronto, 2013.
- [72] S. Hari, M. I. Yousefi, and F. R. Kschischang, “Multieigenvalue Communication”, *Journal of Lightwave Technology*, vol. 34, no. 13, pp. 3110–3117, Jul. 2016. DOI: [10.1109/jlt.2016.2551638](https://doi.org/10.1109/jlt.2016.2551638).
- [73] 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”, *IEEE Photonics Technology Letters*, vol. 27, no. 15, pp. 1621–1623, Aug. 2015. DOI: [10.1109/lpt.2015.2432793](https://doi.org/10.1109/lpt.2015.2432793).
- [74] V. Aref, Z. Dong, and H. Buelow, “Design aspects of multi-soliton pulses for optical fiber transmission”, in *2016 IEEE Photonics Conference (IPC)*, IEEE, Oct. 2016. DOI: [10.1109/ipcon.2016.7831052](https://doi.org/10.1109/ipcon.2016.7831052).
- [75] V. Aref, H. Bulow, K. Schuh, and W. Idler, “Experimental demonstration of nonlinear frequency division multiplexed transmission”, in *2015 European Conference on Optical Communication (ECOC)*, IEEE, Sep. 2015. DOI: [10.1109/ecoc.2015.7341903](https://doi.org/10.1109/ecoc.2015.7341903).
- [76] A. Geisler and C. G. Schaeffer, “Experimental Nonlinear Frequency Division Multiplexed Transmission Using Eigenvalues with Symmetric Real Part”, in *ECOC 2016; 42nd European Conference on Optical Communication*, 2016, pp. 1–3.
- [77] 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*, 2016, pp. 1–3.

- [78] T. Gui, T. H. Chan, C. Lu, A. P. T. Lau, and P.-K. A. Wai, "Alternative Decoding Methods for Optical Communications Based on Nonlinear Fourier Transform", *Journal of Lightwave Technology*, vol. 35, no. 9, pp. 1542–1550, May 2017. DOI: [10.1109/jlt.2017.2654493](https://doi.org/10.1109/jlt.2017.2654493).
- [79] A. Span, V. Aref, H. Bulow, and S. T. Brink, "On time-bandwidth product of multi-soliton pulses", in *2017 IEEE International Symposium on Information Theory (ISIT)*, IEEE, Jun. 2017. DOI: [10.1109/isit.2017.8006490](https://doi.org/10.1109/isit.2017.8006490).
- [80] G. Zhou, T. Gui, C. Lu, A. P. T. Lau, and P.-K. A. Wai, "Improving Soliton Transmission Systems Through Soliton Interactions", *Journal of Lightwave Technology*, vol. 38, no. 14, pp. 3563–3572, Jul. 2020. DOI: [10.1109/jlt.2019.2932332](https://doi.org/10.1109/jlt.2019.2932332).
- [81] F. D. Ros, S. Gaiarin, and D. Zibar, "Impact of Transmitter Phase Noise on NFDMM Transmission With Discrete Spectral Modulation", *IEEE Photonics Technology Letters*, vol. 31, no. 22, pp. 1767–1770, Nov. 2019. DOI: [10.1109/lpt.2019.2946451](https://doi.org/10.1109/lpt.2019.2946451).
- [82] T. Gui, C. Lu, A. P. T. Lau, and P. K. A. Wai, "High-order modulation on a single discrete eigenvalue for optical communications based on nonlinear Fourier transform", *Optics Express*, vol. 25, no. 17, p. 20286, Aug. 2017. DOI: [10.1364/oe.25.020286](https://doi.org/10.1364/oe.25.020286).
- [83] F. J. Garcia-Gomez, "Communication using eigenvalues of higher multiplicity of the nonlinear fourier transform", *Journal of Lightwave Technology*, vol. 36, no. 23, pp. 5442–5450, Dec. 2018. DOI: [10.1109/jlt.2018.2875557](https://doi.org/10.1109/jlt.2018.2875557).
- [84] S. T. Le, S. Wahls, D. Lavery, J. E. Prilepsky, and S. K. Turitsyn, "Reduced Complexity Nonlinear Inverse Synthesis for Nonlinearity Compensation in Optical Fiber Links", in *2015 European Conference on Lasers and Electro-Optics - European Quantum Electronics Conference*, Optical Society of America, 2015. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=CLEO_Europe-2015-CI_3_2.
- [85] S. Le, V. Aref, and H. Buelow, "Nonlinear Signal Multiplexing for Communication beyond the Kerr Nonlinearity Limit", *Nature Photonics*, vol. 11, no. 9, pp. 570–576, Jul. 2017. DOI: [10.1038/nphoton.2017.118](https://doi.org/10.1038/nphoton.2017.118).
- [86] S. T. Le, V. Aref, and H. Buelow, "High Speed Precompensated Nonlinear Frequency-Division Multiplexed Transmissions", *Journal of Lightwave Technology*, vol. 36, no. 6, pp. 1296–1303, Mar. 2018. DOI: [10.1109/jlt.2017.2787185](https://doi.org/10.1109/jlt.2017.2787185).
- [87] W. A. Gemechu, M. Song, Y. Jaouen, S. Wabnitz, and M. I. Yousefi, "Comparison of the Nonlinear Frequency Division Multiplexing and OFDM in Experiment", in *2017 European Conference on Optical Communication (ECOC)*, IEEE, Sep. 2017, pp. 1–3. DOI: [10.1109/ecoc.2017.8346229](https://doi.org/10.1109/ecoc.2017.8346229).
- [88] V. Aref, S. T. Le, and H. Buelow, "Modulation Over Nonlinear Fourier Spectrum: Continuous and Discrete Spectrum", *Journal of Lightwave Technology*, vol. 36, no. 6, pp. 1289–1295, Mar. 2018. DOI: [10.1109/jlt.2018.2794475](https://doi.org/10.1109/jlt.2018.2794475).

- [89] S. T. Le, I. D. Philips, J. E. Prilepsky, P. Harper, A. D. Ellis, and S. K. Turitsyn, "Demonstration of Nonlinear Inverse Synthesis Transmission Over Transoceanic Distances", vol. 34, no. 10, pp. 2459–2466, May 2016. DOI: [10.1109/jlt.2016.2536780](https://doi.org/10.1109/jlt.2016.2536780).
- [90] S. Wahls, "Generation of Time-Limited Signals in the Nonlinear Fourier Domain via b -Modulation", in *2017 European Conference on Optical Communication (ECOC)*, IEEE, Sep. 2017, pp. 1–3. DOI: [10.1109/ecoc.2017.8346231](https://doi.org/10.1109/ecoc.2017.8346231).
- [91] D. Shepelsky, A. Vasylenkova, J. E. Prilepsky, and I. Karpenko, "Nonlinear Fourier Spectrum Characterization of Time-Limited Signals", *IEEE Transactions on Communications*, vol. 68, no. 5, pp. 3024–3032, May 2020. DOI: [10.1109/tcomm.2020.2973265](https://doi.org/10.1109/tcomm.2020.2973265).
- [92] T. Gui, G. Zhou, C. Lu, A. P. T. Lau, and S. Wahls, "Nonlinear frequency division multiplexing with b -modulation: shifting the energy barrier", *Optics Express*, vol. 26, no. 21, pp. 27 978–27 990, Oct. 2018. DOI: [10.1364/oe.26.027978](https://doi.org/10.1364/oe.26.027978).
- [93] S. Gaiarin, A. M. Perego, E. P. da Silva, F. D. Ros, and D. Zibar, "Dual-polarization nonlinear Fourier transform-based optical communication system", *Optica*, vol. 5, no. 3, pp. 263–270, Mar. 2018. DOI: [10.1364/optica.5.000263](https://doi.org/10.1364/optica.5.000263).
- [94] R. Yu, Z. Zheng, X. Zhang, S. Du, L. Xi, and X. Zhang, "Hybrid Probabilistic-Geometric Shaping in DP-NFDM Systems", in *2019 18th International Conference on Optical Communications and Networks (ICOON)*, IEEE, Aug. 2019, pp. 1–3. DOI: [10.1109/icocn.2019.8934217](https://doi.org/10.1109/icocn.2019.8934217).
- [95] J.-W. Goossens, M. I. Yousefi, Y. Jaouën, and H. Hafermann, "Polarization-division multiplexing based on the nonlinear Fourier transform", *Optics Express*, vol. 25, no. 22, pp. 26 437–26 452, Oct. 2017. DOI: [10.1364/OE.25.026437](https://doi.org/10.1364/OE.25.026437).
- [96] S. Civelli, S. K. Turitsyn, M. Secondini, and J. E. Prilepsky, "Polarization-multiplexed nonlinear inverse synthesis with standard and reduced-complexity NFT processing", *Optics Express*, vol. 26, no. 13, pp. 17 360–17 377, Jun. 2018. DOI: [10.1364/oe.26.017360](https://doi.org/10.1364/oe.26.017360).
- [97] W. A. Gemechu, T. Gui, J.-W. Goossens, M. Song, S. Wabnitz, H. Hafermann, A. P. T. Lau, M. I. Yousefi, and Y. Jaouen, "Dual Polarization Nonlinear Frequency Division Multiplexing Transmission", *IEEE Photonics Technology Letters*, vol. 30, no. 18, pp. 1589–1592, Sep. 2018. DOI: [10.1109/lpt.2018.2860124](https://doi.org/10.1109/lpt.2018.2860124).
- [98] X. Yangzhang, S. T. Le, V. Aref, H. Buelow, D. Lavery, and P. Bayvel, "Experimental Demonstration of Dual-Polarization NFDM Transmission With b -Modulation", *IEEE Photonics Technology Letters*, vol. 31, no. 11, pp. 885–888, Jun. 2019. DOI: [10.1109/lpt.2019.2911600](https://doi.org/10.1109/lpt.2019.2911600).
- [99] 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", *Journal of Lightwave Technology*, vol. 37, no. 6, pp. 1570–1578, Mar. 2019. DOI: [10.1109/jlt.2019.2902961](https://doi.org/10.1109/jlt.2019.2902961).

- [100] F. D. Ros, S. Civelli, S. Gaiarin, E. P. da Silva, N. D. Renzis, M. Secondini, and D. Zibar, “Dual-Polarization NFDm Transmission With Continuous and Discrete Spectral Modulation”, *Journal of Lightwave Technology*, vol. 37, no. 10, pp. 2335–2343, May 2019. DOI: [10.1109/jlt.2019.2904102](https://doi.org/10.1109/jlt.2019.2904102).
- [101] Q. Zhang and F. Kschischang, “Correlation-Aided Nonlinear Spectrum Detection”, *Journal of Lightwave Technology*, vol. 39, no. 15, pp. 4923–4931, Aug. 2021. DOI: [10.1109/jlt.2021.3078700](https://doi.org/10.1109/jlt.2021.3078700).
- [102] M. Kamalian, J. E. Prilepsky, S. T. Le, and S. K. Turitsyn, “Periodic nonlinear Fourier transform for fiber-optic communications, Part I: theory and numerical methods”, *Optics Express*, vol. 24, no. 16, p. 18 353, Aug. 2016. DOI: [10.1364/oe.24.018353](https://doi.org/10.1364/oe.24.018353).
- [103] —, “Periodic nonlinear Fourier transform for fiber-optic communications, Part II: eigenvalue communication”, *Optics Express*, vol. 24, no. 16, p. 18 370, Aug. 2016. DOI: [10.1364/oe.24.018370](https://doi.org/10.1364/oe.24.018370).
- [104] —, “Spectral efficiency estimation in periodic nonlinear Fourier transform based communication systems”, in *2017 Optical Fiber Communications Conference and Exhibition (OFC)*, 2017, pp. 1–3.
- [105] M. Kamalian, A. Vasylychenkova, J. Prilepsky, D. Shepelsky, and S. Turitsyn, “Communication System Based on Periodic Nonlinear Fourier Transform with Exact Inverse Transformation”, in *2018 European Conference on Optical Communication (ECOC)*, IEEE, Sep. 2018. DOI: [10.1109/ecoc.2018.8535278](https://doi.org/10.1109/ecoc.2018.8535278).
- [106] A. O. Smirnov, “Periodic two-phase “Rogue waves””, *Mathematical Notes*, vol. 94, pp. 897–907, 2013.
- [107] M. Kamalian, A. Vasylychenkova, D. Shepelsky, J. E. Prilepsky, and S. K. Turitsyn, “Signal Modulation and Processing in Nonlinear Fibre Channels by Employing the Riemann–Hilbert Problem”, *Journal of Lightwave Technology*, vol. 36, no. 24, pp. 5714–5727, Dec. 2018. DOI: [10.1109/jlt.2018.2877103](https://doi.org/10.1109/jlt.2018.2877103).
- [108] J.-W. Goossens, H. Hafermann, and Y. Jaouën, “Experimental realization of Fermi-Pasta-Ulam-Tsingou recurrence in a long-haul optical fiber transmission system”, *Scientific Reports*, vol. 9, no. 1, Dec. 2019. DOI: [10.1038/s41598-019-54825-4](https://doi.org/10.1038/s41598-019-54825-4).
- [109] J.-W. Goossens, Y. Jaouen, and H. Hafermann, “Experimental Demonstration of Data Transmission Based on the Exact Inverse Periodic Nonlinear Fourier Transform”, in *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, 2019, pp. 1–3.
- [110] M. Kamalian-Kopae, A. Vasylychenkova, D. Shepelsky, J. E. Prilepsky, and S. K. Turitsyn, “Full-Spectrum Periodic Nonlinear Fourier Transform Optical Communication Through Solving the Riemann-Hilbert Problem”, *Journal of Lightwave Technology*, vol. 38, no. 14, pp. 3602–3615, Jul. 2020. DOI: [10.1109/jlt.2020.2979322](https://doi.org/10.1109/jlt.2020.2979322).

- [111] J.-W. Goossens, H. Hafermann, and Y. Jaouen, “Data Transmission Based on Exact Inverse Periodic Nonlinear Fourier Transform, Part I: Theory”, *Journal of Lightwave Technology*, vol. 38, no. 23, pp. 6499–6519, Dec. 2020. DOI: [10.1109/jlt.2020.3013148](https://doi.org/10.1109/jlt.2020.3013148).
- [112] —, “Data Transmission Based on Exact Inverse Periodic Nonlinear Fourier Transform, Part II: Waveform Design and Experiment”, *Journal of Lightwave Technology*, vol. 38, no. 23, pp. 6520–6528, Dec. 2020. DOI: [10.1109/jlt.2020.3013163](https://doi.org/10.1109/jlt.2020.3013163).
- [113] J. Satsuma and N. Yajima, “B. Initial Value Problems of One-Dimensional Self-Modulation of Nonlinear Waves in Dispersive Media”, *Progress of Theoretical Physics Supplement*, vol. 55, pp. 284–306, 1974. DOI: [10.1143/ptps.55.284](https://doi.org/10.1143/ptps.55.284).
- [114] D. E. Rourke and P. G. Morris, “Half solitons as solutions to the Zakharov-Shabat eigenvalue problem for rational reflection coefficient with application in the design of selective pulses in nuclear magnetic resonance”, *Physical Review A*, vol. 46, no. 7, pp. 3631–3636, Oct. 1992. DOI: [10.1103/physreva.46.3631](https://doi.org/10.1103/physreva.46.3631).
- [115] G. Biondini, J. Oregero, and A. Tovbis, “On the spectrum of the focusing Zakharov-Shabat operator with periodic potentials”, Oct. 2020. arXiv: [2010.04263](https://arxiv.org/abs/2010.04263) [math.SP].
- [116] G. Boffetta and A. R. Osborne, “Computation of the Direct Scattering Transform for the Nonlinear Schroedinger Equation”, *Journal of Computational Physics*, vol. 102, no. 2, pp. 252–264, Oct. 1992. DOI: [10.1016/0021-9991\(92\)90370-e](https://doi.org/10.1016/0021-9991(92)90370-e).
- [117] C. Moler and C. V. Loan, “Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later”, *SIAM Review*, vol. 45, no. 1, pp. 3–49, Jan. 2003. DOI: [10.1137/s00361445024180](https://doi.org/10.1137/s00361445024180).
- [118] A. Vasylychenkova, J. E. Prilepsy, D. Shepelsky, and A. Chattopadhyay, “Direct nonlinear Fourier transform algorithms for the computation of solitonic spectra in focusing nonlinear Schrödinger equation”, *Communications in Nonlinear Science and Numerical Simulation*, vol. 68, pp. 347–371, Mar. 2019. DOI: [10.1016/j.cnsns.2018.09.005](https://doi.org/10.1016/j.cnsns.2018.09.005).
- [119] S. Burtsev, R. Camassa, and I. Timofeyev, “Numerical Algorithms for the Direct Spectral Transform with Applications to Nonlinear Schrödinger Type Systems”, *Journal of Computational Physics*, vol. 147, no. 1, pp. 166–186, Nov. 1998. DOI: [10.1006/jcph.1998.6087](https://doi.org/10.1006/jcph.1998.6087).
- [120] V. Aref, “Control and Detection of Discrete Spectral Amplitudes in Nonlinear Fourier Spectrum”, May 18, 2016. arXiv: [1605.06328](https://arxiv.org/abs/1605.06328) [math.NA].
- [121] S. Chimmalgi, P. J. Prins, and S. Wahls, “Nonlinear Fourier Transform Algorithm Using a Higher Order Exponential Integrator”, in *Advanced Photonics 2018*, OSA, 2018, SpM4G.5. DOI: [10.1364/sppcom.2018.spm4g.5](https://doi.org/10.1364/sppcom.2018.spm4g.5).
- [122] S. Medvedev, I. Vaseva, I. Chekhovskoy, and M. Fedoruk, “Numerical Algorithm with Fourth-order Accuracy for the Direct Zakharov-shabat Problem”, *Optics Letters*, vol. 44, no. 9, p. 2264, Apr. 2019. DOI: [10.1364/ol.44.002264](https://doi.org/10.1364/ol.44.002264).
- [123] —, “Exponential Fourth Order Schemes for Direct Zakharov-shabat Problem”, *Optics Express*, vol. 28, no. 1, p. 20, Dec. 2019. DOI: [10.1364/oe.377140](https://doi.org/10.1364/oe.377140).

- [124] A. Vasylichenkova, J. E. Prilepsky, and S. K. Turitsyn, “Contour integrals for numerical computation of discrete eigenvalues in the Zakharov–Shabat problem”, *Optics Letters*, vol. 43, no. 15, pp. 3690–3693, Aug. 2018. DOI: [10.1364/ol.43.003690](https://doi.org/10.1364/ol.43.003690).
- [125] 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 (OFC) 2019*, OSA, 2019. DOI: [10.1364/ofc.2019.m1i.5](https://doi.org/10.1364/ofc.2019.m1i.5).
- [126] I. Chekhovskoy, S. B. Medvedev, I. A. Vaseva, E. V. Sedov, and M. P. Fedoruk, “Introducing phase jump tracking - a fast method for eigenvalue evaluation of the direct Zakharov–Shabat problem”, *Communications in Nonlinear Science and Numerical Simulation*, vol. 96, p. 105718, May 2021. DOI: [10.1016/j.cnsns.2021.105718](https://doi.org/10.1016/j.cnsns.2021.105718).
- [127] P. J. Prins and S. Wahls, “Soliton Phase Shift Calculation for the Korteweg–De Vries Equation”, *IEEE Access*, vol. 7, pp. 122914–122930, 2019. DOI: [10.1109/access.2019.2932256](https://doi.org/10.1109/access.2019.2932256).
- [128] S. Wahls and H. V. Poor, “Fast Numerical Nonlinear Fourier Transforms”, *IEEE Transactions on Information Theory*, vol. 61, no. 12, pp. 6957–6974, Dec. 2015. DOI: [10.1109/tit.2015.2485944](https://doi.org/10.1109/tit.2015.2485944).
- [129] L. Rabiner, R. Schafer, and C. Rader, “The Chirp Z-transform Algorithm”, *IEEE Transactions on Audio and Electroacoustics*, vol. 17, no. 2, pp. 86–92, Jun. 1969. DOI: [10.1109/tau.1969.1162034](https://doi.org/10.1109/tau.1969.1162034).
- [130] V. Vaibhav, “Higher Order Convergent Fast Nonlinear Fourier Transform”, *IEEE Photonics Technology Letters*, vol. 30, no. 8, pp. 700–703, Apr. 2018. DOI: [10.1109/lpt.2018.2812808](https://doi.org/10.1109/lpt.2018.2812808).
- [131] —, “Efficient Nonlinear Fourier Transform Algorithms of Order Four on Equi-spaced Grid”, *IEEE Photonics Technology Letters*, vol. 31, no. 15, pp. 1269–1272, Aug. 2019. DOI: [10.1109/lpt.2019.2925052](https://doi.org/10.1109/lpt.2019.2925052).
- [132] O. V. Belai, L. L. Frumin, E. V. Podivilov, and D. A. Shapiro, “Efficient numerical method of the fiber Bragg grating synthesis”, *Journal of the Optical Society of America B*, vol. 24, no. 7, pp. 1451–1457, Jul. 2007. DOI: [10.1364/josab.24.001451](https://doi.org/10.1364/josab.24.001451).
- [133] A. Bruckstein and T. Kailath, “An inverse scattering framework for several problems in signal processing”, *IEEE ASSP Magazine*, vol. 4, no. 1, pp. 6–20, Jan. 1987. DOI: [10.1109/massp.1987.1165567](https://doi.org/10.1109/massp.1987.1165567).
- [134] Y. Choi, J. Chun, T. Kim, and J. Bae, “The Schur Algorithm Applied to the One-Dimensional Continuous Inverse Scattering Problem”, *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3311–3320, Jul. 2013. DOI: [10.1109/tsp.2013.2259487](https://doi.org/10.1109/tsp.2013.2259487).
- [135] J. Skaar and O. H. Waagaard, “Design and characterization of finite-length fiber gratings”, *IEEE Journal of Quantum Electronics*, vol. 39, no. 10, pp. 1238–1245, Oct. 2003. DOI: [10.1109/jqe.2003.817581](https://doi.org/10.1109/jqe.2003.817581).

- [136] M. Yamada and K. Sakuda, "Analysis of almost-periodic distributed feedback slab waveguides via a fundamental matrix approach", *Applied Optics*, vol. 26, no. 16, pp. 3474–3478, Aug. 1987. DOI: [10.1364/ao.26.003474](https://doi.org/10.1364/ao.26.003474).
- [137] S. Civelli, L. Barletti, and M. Secondini, "Numerical methods for the inverse nonlinear fourier transform", in *2015 Tyrrhenian International Workshop on Digital Communications (TIWDC)*, IEEE, Sep. 2015. DOI: [10.1109/tiwdc.2015.7323325](https://doi.org/10.1109/tiwdc.2015.7323325).
- [138] S. Wahls and H. V. Poor, "Fast Inverse Nonlinear Fourier Transform for Generating Multi-solitons in Optical Fiber", in *2015 IEEE International Symposium on Information Theory (ISIT)*, IEEE, Jun. 2015, pp. 1676–1680. DOI: [10.1109/isit.2015.7282741](https://doi.org/10.1109/isit.2015.7282741).
- [139] S. Wahls and V. Vaibhav, "Fast Inverse Nonlinear Fourier Transforms for Continuous Spectra of Zakharov-Shabat Type", Jul. 2016. arXiv: [1607.01305 \[cs.IT\]](https://arxiv.org/abs/1607.01305).
- [140] V. Vaibhav and S. Wahls, "Introducing the Fast Inverse NFT", in *Optical Fiber Communication Conference*, OSA, 2017. DOI: [10.1364/ofc.2017.tu3d.2](https://doi.org/10.1364/ofc.2017.tu3d.2).
- [141] V. Vaibhav, "Fast inverse nonlinear Fourier transform", *Physical Review E*, vol. 98, no. 1, p. 013 304, Jul. 2018. DOI: [10.1103/physreve.98.013304](https://doi.org/10.1103/physreve.98.013304).
- [142] R. Hirota, *The Direct Method in Soliton Theory*, A. Nagai, J. Nimmo, and C. Gilson, Eds. Cambridge, U.K.: Cambridge University Press, Jul. 2004. DOI: [10.1017/cbo9780511543043](https://doi.org/10.1017/cbo9780511543043).
- [143] G. Neugebauer and R. Meinel, "General N-soliton Solution of the AKNS Class on Arbitrary Background", *Physics Letters A*, vol. 100, no. 9, pp. 467–470, Feb. 1984. DOI: [10.1016/0375-9601\(84\)90827-2](https://doi.org/10.1016/0375-9601(84)90827-2).
- [144] J. Lin, "Evolution of the Scattering Data under the Classical Darboux Transform for $su(2)$ Soliton Systems", *Acta Mathematicae Applicatae Sinica, English Series*, vol. 6, no. 4, pp. 308–316, Oct. 1990. DOI: [10.1007/bf02015338](https://doi.org/10.1007/bf02015338).
- [145] C. Gu, H. Hu, and Z. Zhou, "Darboux transformations in integrable systems", in *Darboux Transformations in Integrable Systems*. Dordrecht: Springer, Jul. 9, 2006, ch. "1+1 Dimensional Integrable Systems", ISBN: 9781402030888. [Online]. Available: <https://www.springer.com/gp/book/9781402030871>.
- [146] V. Vaibhav, "Fast inverse nonlinear Fourier transformation using exponential one-step methods: Darboux transformation", *Physical Review E*, vol. 96, no. 6, p. 063 302, Dec. 2017. DOI: [10.1103/physreve.96.063302](https://doi.org/10.1103/physreve.96.063302).
- [147] S. Chimmalgi and S. Wahls, "Discrete Darboux based Fast Inverse Nonlinear Fourier Transform Algorithm for Multi-solitons", in *2017 European Conference on Optical Communication (ECOC)*, IEEE, Sep. 2017. DOI: [10.1109/ecoc.2017.8346226](https://doi.org/10.1109/ecoc.2017.8346226).
- [148] S. Chimmalgi, "Improved Fast Inverse Nonlinear Fourier Transform for Multi-solitons: A Discrete Darboux Based Approach", M.S. thesis, TU Delft Delft Center for Systems and Control, Aug. 2017. [Online]. Available: <http://resolver.tudelft.nl/uuid:adf4d14a-fecd-400f-a1be-3e6c32d23a06>.

- [149] S. Olver, *RHPackage for MATHEMATICA*, Online. [Online]. Available: <http://www.maths.usyd.edu.au/u/olver/%20projects/RHPackage.html>.
- [150] —, *A Julia package for solving Riemann–Hilbert problems*, Online. [Online]. Available: <https://github.com/JuliaHolomorphic/RiemannHilbert.jl>.
- [151] J. Frauendiener and C. Klein, “Algebraic Curves and Riemann Surfaces in Matlab”, in *Computational Approach to Riemann Surfaces*, A. I. Bobenko and C. Klein, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 125–162, ISBN: 978-3-642-17413-1. DOI: [10.1007/978-3-642-17413-1_3](https://doi.org/10.1007/978-3-642-17413-1_3). [Online]. Available: https://doi.org/10.1007/978-3-642-17413-1_3.
- [152] J. Frauendiener, C. Jaber, and C. Klein, “Efficient computation of multidimensional theta functions”, *Journal of Geometry and Physics*, vol. 141, pp. 147–158, Jul. 2019. DOI: [10.1016/j.geomphys.2019.03.011](https://doi.org/10.1016/j.geomphys.2019.03.011).
- [153] J. Frauendiener and C. Klein, “Hyperelliptic Theta-Functions and Spectral Methods: KdV and KP Solutions”, *Letters in Mathematical Physics*, vol. 76, no. 2-3, pp. 249–267, Apr. 2006. DOI: [10.1007/s11005-006-0068-4](https://doi.org/10.1007/s11005-006-0068-4).
- [154] M. Brehler, C. Mahnke, S. Chimmalgi, and S. Wahls, “NFDMLab: Simulating Nonlinear Frequency Division Multiplexing in Python”, in *Optical Fiber Communication Conference (OFC) 2019*, OSA, 2019. DOI: [10.1364/ofc.2019.m3z.13](https://doi.org/10.1364/ofc.2019.m3z.13).
- [155] S. Chimmalgi, P. J. Prins, and S. Wahls, “Fast Nonlinear Fourier Transform Algorithms Using Higher Order Exponential Integrators”, *IEEE Access*, vol. 7, pp. 145 161–145 176, 2019. DOI: [10.1109/access.2019.2945480](https://doi.org/10.1109/access.2019.2945480).
- [156] *Creative Commons Attribution 4.0 International (CC BY 4.0) License*, WebPage. [Online]. Available: <https://creativecommons.org/licenses/by/4.0/>.
- [157] C. L. Epstein, “How well does the finite Fourier transform approximate the Fourier transform?”, *Communications on Pure and Applied Mathematics*, vol. 58, no. 10, pp. 1421–1435, 2005. DOI: [10.1002/cpa.20064](https://doi.org/10.1002/cpa.20064).
- [158] A. R. Osborne, “Nonlinear Fourier Methods for Ocean Waves”, *Procedia IUTAM*, vol. 26, pp. 112–123, 2018. DOI: [10.1016/j.piutam.2018.03.011](https://doi.org/10.1016/j.piutam.2018.03.011).
- [159] M. Brühl, “Direct and Inverse Nonlinear Fourier Transform Based on the Korteweg-devries Equation (kdv-nlft) - a Spectral Analysis of Nonlinear Surface Waves in Shallow Water”, en, Ph.D. dissertation, Feb. 2014. DOI: [10.24355/DBBS.084-201411210916-0](https://doi.org/10.24355/DBBS.084-201411210916-0). [Online]. Available: https://publikationsserver.tu-braunschweig.de/receive/dbbs_mods_00058144.
- [160] A. C. Singer, A. V. Oppenheim, and G. W. Wornell, “Detection and estimation of multiplexed soliton signals”, *IEEE Transactions on Signal Processing*, vol. 47, no. 10, pp. 2768–2782, Oct. 1999. DOI: [10.1109/78.790658](https://doi.org/10.1109/78.790658).
- [161] S. Wahls and H. V. Poor, “Introducing the fast nonlinear Fourier transform”, in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, May 2013, pp. 5780–5784. DOI: [10.1109/icassp.2013.6638772](https://doi.org/10.1109/icassp.2013.6638772).

- [162] M. Thalhammer, “A fourth-order commutator-free exponential integrator for nonautonomous differential equations”, *SIAM Journal on Numerical Analysis*, vol. 44, no. 2, pp. 851–864, Jan. 2006. DOI: [10.1137/05063042](https://doi.org/10.1137/05063042). eprint: <https://doi.org/10.1137/05063042>. [Online]. Available: <https://doi.org/10.1137/05063042>.
- [163] 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, Apr. 2018. DOI: [10.1109/ICASSP.2018.8461708](https://doi.org/10.1109/ICASSP.2018.8461708).
- [164] S. Blanes, F. Casas, J. A. Oteo, and J. Ros, “The Magnus expansion and some of its applications”, *Physics Reports*, vol. 470, no. 5-6, pp. 151–238, Jan. 2009. DOI: [10.1016/j.physrep.2008.11.001](https://doi.org/10.1016/j.physrep.2008.11.001).
- [165] S. Blanes, F. Casas, and M. Thalhammer, “High-order commutator-free quasi-Magnus exponential integrators for non-autonomous linear evolution equations”, *Computer Physics Communications*, vol. 220, pp. 243–262, Nov. 2017. DOI: [10.1016/j.cpc.2017.07.016](https://doi.org/10.1016/j.cpc.2017.07.016).
- [166] A. Oppenheim, *Signals and systems*. Essex, England: Pearson Education Limited, 2013, ISBN: 9781292025902.
- [167] E. V. Podivilov, D. A. Shapiro, and D. A. Trubitsyn, “Exactly solvable profiles of quasi-rectangular Bragg filter with dispersion compensation”, *Journal of Optics A: Pure and Applied Optics*, vol. 8, no. 9, pp. 788–795, Jul. 2006. DOI: [10.1088/1464-4258/8/9/013](https://doi.org/10.1088/1464-4258/8/9/013).
- [168] D. S. Bernstein and W. So, “Some explicit formulas for the matrix exponential”, *IEEE Transactions on Automatic Control*, vol. 38, no. 8, pp. 1228–1232, Aug. 1993. DOI: [10.1109/9.233156](https://doi.org/10.1109/9.233156).
- [169] E. H.-L. Liu, *Fundamental Methods of Numerical Extrapolation with Applications*, MIT open courseware, Massachusetts Institute Of Technology, Accessed: 1-11-2018, 2006. [Online]. Available: <http://web.mit.edu/ehliu/Public/Spring2006/18.304/extrapolation.pdf>.
- [170] J. L. Blue and H. K. Gummel, “Rational approximations to matrix exponential for systems of stiff differential equations”, *Journal of Computational Physics*, vol. 5, no. 1, pp. 70–83, Feb. 1970. DOI: [10.1016/0021-9991\(70\)90053-7](https://doi.org/10.1016/0021-9991(70)90053-7).
- [171] S. Wahls, S. Chimmalgi, and P. J. Prins, “FNFT: A Software Library for Computing Nonlinear Fourier Transforms”, *Journal of Open Source Software*, vol. 3, no. 23, p. 597, 23 Mar. 2018. DOI: [10.21105/joss.00597](https://doi.org/10.21105/joss.00597).
- [172] J. Keiner, S. Kunis, and D. Potts, “Using NFFT 3—A Software Library for Various Nonequispaced Fast Fourier Transforms”, *ACM Transactions on Mathematical Software*, vol. 36, no. 4, pp. 1–30, Aug. 2009. DOI: [10.1145/1555386.1555388](https://doi.org/10.1145/1555386.1555388).

- [173] S. T. Le, K. Schuh, F. Buchali, and H. Buelow, "100 Gbps b-modulated Nonlinear Frequency Division Multiplexed Transmission", in *Optical Fiber Communication Conference*, OSA, Mar. 2018, pp. 1–3. DOI: [10.1364/ofc.2018.w1g.6](https://doi.org/10.1364/ofc.2018.w1g.6). [Online]. Available: <https://www.osapublishing.org/abstract.cfm?URI=OFC-2018-W1G.6>.
- [174] A. Sidi, *Practical Extrapolation Methods*, 1st. New York, NY, USA: Cambridge University Press, Jun. 2003, ISBN: 9780511546815. DOI: [10.1017/cbo9780511546815](https://doi.org/10.1017/cbo9780511546815).
- [175] I. H. Sloan, "Superconvergence", in *Numerical Solution of Integral Equations*, M. A. Golberg, Ed. Boston, MA: Springer US, 1990, pp. 35–70, ISBN: 978-1-4899-2593-0. DOI: [10.1007/978-1-4899-2593-0_2](https://doi.org/10.1007/978-1-4899-2593-0_2).
- [176] J. E. Huss and J. A. Pennline, *Comparison of five benchmarks*, Feb. 1987. [Online]. Available: <https://www.osti.gov/biblio/6574703>.
- [177] M. Frigo and S. G. Johnson, "The Design and Implementation of FFTW3", *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, Feb. 2005. DOI: [10.1109/jproc.2004.840301](https://doi.org/10.1109/jproc.2004.840301).
- [178] J. M. McNamee and V. Y. Pan, "Efficient Polynomial Root-refiners: A Survey and New Record Efficiency Estimates", *Computers & Mathematics with Applications*, vol. 63, no. 1, pp. 239–254, Jan. 2012. DOI: [10.1016/j.camwa.2011.11.015](https://doi.org/10.1016/j.camwa.2011.11.015).
- [179] J. L. Aurentz, T. Mach, R. Vandebril, and D. S. Watkins, "Fast and Backward Stable Computation of Roots of Polynomials", *SIAM Journal on Matrix Analysis and Applications*, vol. 36, no. 3, pp. 942–973, Jan. 2015. DOI: [10.1137/140983434](https://doi.org/10.1137/140983434).
- [180] S. Randoux, P. Suret, and G. El, "Inverse scattering transform analysis of rogue waves using local periodization procedure", *Scientific Reports*, vol. 6, no. 1, Jul. 2016. DOI: [10.1038/srep29238](https://doi.org/10.1038/srep29238).
- [181] M. Brühl and M. Becker, "Analysis of Subaerial Landslide Data Using Nonlinear Fourier Transform Based on Korteweg-deVries Equation (KdV-NLFT)", *Journal of Earthquake and Tsunami*, vol. 12, no. 02, p. 1 840 002, Jun. 2018. DOI: [10.1142/s179343111840002x](https://doi.org/10.1142/s179343111840002x).
- [182] G. Jeans, W. Xiao, A. R. Osborne, C. R. Jackson, and D. A. Mitchell, "The Application of Nonlinear Fourier Analysis to Soliton Quantification for Offshore Engineering", in *Volume 1: Offshore Technology*, American Society of Mechanical Engineers, Jun. 2017. DOI: [10.1115/omae2017-61943](https://doi.org/10.1115/omae2017-61943).
- [183] A. R. Osborne and S. P. de León, "Properties of Rogue Waves and the Shape of the Ocean Wave Power Spectrum", in *Volume 3A: Structures, Safety and Reliability*, American Society of Mechanical Engineers, Jun. 2017. DOI: [10.1115/omae2017-62217](https://doi.org/10.1115/omae2017-62217).
- [184] S. Randoux, P. Suret, A. Chabchoub, B. Kibler, and G. El, "Nonlinear spectral analysis of Peregrine solitons observed in optics and in hydrodynamic experiments", *Physical Review E*, vol. 98, no. 2, p. 022 219, Aug. 2018. DOI: [10.1103/physreve.98.022219](https://doi.org/10.1103/physreve.98.022219).

- [185] A. R. Osborne, D. T. Resio, A. Costa, S. P. de León, and E. Chirivi, “Highly nonlinear wind waves in Currituck Sound: dense breather turbulence in random ocean waves”, *Ocean Dynamics*, vol. 69, no. 2, pp. 187–219, Dec. 2018. DOI: [10.1007/s10236-018-1232-y](https://doi.org/10.1007/s10236-018-1232-y).
- [186] B. Q. Baragiola, G. Pantaleoni, R. N. Alexander, A. Karanjai, and N. C. Menicucci, “All-Gaussian Universality and Fault Tolerance with the Gottesman-Kitaev-Preskill Code”, *Physical Review Letters*, vol. 123, no. 20, p. 200502, Nov. 2019. DOI: [10.1103/physrevlett.123.200502](https://doi.org/10.1103/physrevlett.123.200502).
- [187] A. I. Bobenko, “All constant mean curvature tori in R^3 , S^3 , H^3 in terms of theta-functions”, *Mathematische Annalen*, vol. 290, no. 1, pp. 209–245, Mar. 1991. DOI: [10.1007/bf01459243](https://doi.org/10.1007/bf01459243).
- [188] B. Deconinck and M. van Hoeij, “Computing Riemann matrices of algebraic curves”, *Physica D: Nonlinear Phenomena*, vol. 152–153, pp. 28–46, May 2001. DOI: [10.1016/s0167-2789\(01\)00156-7](https://doi.org/10.1016/s0167-2789(01)00156-7).
- [189] C. Birkenhake, *Complex Abelian Varieties*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, ISBN: 9783662063071.
- [190] M. Eichler and D. Zagier, *The Theory of Jacobi Forms*. Birkhäuser Boston, 1985. DOI: [10.1007/978-1-4684-9162-3](https://doi.org/10.1007/978-1-4684-9162-3).
- [191] O. Regev and N. Stephens-Davidowitz, “An Inequality for Gaussians on Lattices”, *SIAM Journal on Discrete Mathematics*, vol. 31, no. 2, pp. 749–757, Jan. 2017. DOI: [10.1137/15m1052226](https://doi.org/10.1137/15m1052226).
- [192] P. Gaudry, “Fast Genus 2 Arithmetic Based on Theta Functions”, *Journal of Mathematical Cryptology*, vol. 1, no. 3, Jan. 2007. DOI: [10.1515/jmc.2007.012](https://doi.org/10.1515/jmc.2007.012).
- [193] D. Agostini and C. Améndola, “Discrete Gaussian Distributions via Theta Functions”, *SIAM Journal on Applied Algebra and Geometry*, vol. 3, no. 1, pp. 1–30, Jan. 2019. DOI: [10.1137/18m1164937](https://doi.org/10.1137/18m1164937).
- [194] H. Labrande and E. Thomé, “Computing theta functions in quasi-linear time in genus two and above”, *LMS Journal of Computation and Mathematics*, vol. 19, no. A, pp. 163–177, 2016. DOI: [10.1112/s1461157016000309](https://doi.org/10.1112/s1461157016000309).
- [195] D. Agostini and L. Chua, “Computing theta functions with Julia”, *Journal of Software for Algebra and Geometry*, vol. 11, no. 1, pp. 41–51, Mar. 2021. DOI: [10.2140/jsag.2021.11.41](https://doi.org/10.2140/jsag.2021.11.41).
- [196] L. Chen, “Curse of Dimensionality”, in *Encyclopedia of Database Systems*, Springer US, 2009, pp. 545–546. DOI: [10.1007/978-0-387-39940-9_133](https://doi.org/10.1007/978-0-387-39940-9_133).
- [197] N. D. Sidiropoulos, L. D. Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, “Tensor Decomposition for Signal Processing and Machine Learning”, *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017. DOI: [10.1109/tsp.2017.2690524](https://doi.org/10.1109/tsp.2017.2690524).
- [198] N. Ghadermarzy, Y. Plan, and O. Yilmaz, “Learning Tensors From Partial Binary Measurements”, *IEEE Transactions on Signal Processing*, vol. 67, no. 1, pp. 29–40, Jan. 2019. DOI: [10.1109/tsp.2018.2879031](https://doi.org/10.1109/tsp.2018.2879031).

- [199] C. I. Kanatsoulis, X. Fu, N. D. Sidiropoulos, and M. Akcakaya, “Tensor Completion From Regular Sub-Nyquist Samples”, *IEEE Transactions on Signal Processing*, vol. 68, pp. 1–16, 2020. DOI: [10.1109/tsp.2019.2952044](https://doi.org/10.1109/tsp.2019.2952044).
- [200] J. Chen, Y. Chen, Y. Duan, D. Liu, and M. Tang, “Geometric Shaping Optimization of 64-APSK Eigenvalue Transmission based on Nonlinear Fourier Transform”, in *2021 Optical Fiber Communications Conference and Exhibition (OFC)*, 2021, pp. 1–3.
- [201] H. Chen, F. Ahmad, S. Vorobyov, and F. Porikli, “Tensor Decompositions in Wireless Communications and MIMO Radar”, *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 3, pp. 438–453, Apr. 2021. DOI: [10.1109/jstsp.2021.3061937](https://doi.org/10.1109/jstsp.2021.3061937).
- [202] D. Muti and S. Bourennane, “Multidimensional signal processing using lower-rank tensor approximation”, in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)*, IEEE, DOI: [10.1109/icassp.2003.1199510](https://doi.org/10.1109/icassp.2003.1199510).
- [203] J. H. de Morais Goulart and P. Comon, “A novel non-iterative algorithm for low-multilinear-rank tensor approximation”, in *2017 25th European Signal Processing Conference (EUSIPCO)*, IEEE, Aug. 2017. DOI: [10.23919/eusipco.2017.8081288](https://doi.org/10.23919/eusipco.2017.8081288).
- [204] L. Yuan, Q. Zhao, and J. Cao, “High-Order Tensor Completion for Data Recovery via Sparse Tensor-Train Optimization”, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, Apr. 2018. DOI: [10.1109/icassp.2018.8462592](https://doi.org/10.1109/icassp.2018.8462592).
- [205] P. Gelß, “The Tensor-Train Format and Its Applications”, Ph.D. dissertation, 2017. DOI: [10.17169/REFUBIUM-7566](https://doi.org/10.17169/REFUBIUM-7566).
- [206] S. Wahls, V. Koivunen, H. V. Poor, and M. Verhaegen, “Learning multidimensional Fourier series with tensor trains”, in *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, IEEE, Dec. 2014. DOI: [10.1109/globalsip.2014.7032146](https://doi.org/10.1109/globalsip.2014.7032146).
- [207] N. Kargas and N. D. Sidiropoulos, “Supervised Learning and Canonical Decomposition of Multivariate Functions”, *IEEE Transactions on Signal Processing*, vol. 69, pp. 1097–1107, 2021. DOI: [10.1109/tsp.2021.3055000](https://doi.org/10.1109/tsp.2021.3055000).
- [208] K. Hallatschek, “Fouriertransformation auf dünnen Gittern mit hierarchischen Basen”, *Numerische Mathematik*, vol. 63, no. 1, pp. 83–97, 1992.
- [209] M. Döhler, S. Kunis, and D. Potts, “Nonequispaced Hyperbolic Cross Fast Fourier Transform”, *SIAM Journal on Numerical Analysis*, vol. 47, no. 6, pp. 4415–4428, Jan. 2010. DOI: [10.1137/090754947](https://doi.org/10.1137/090754947).
- [210] L. Kämmerer, “Reconstructing Hyperbolic Cross Trigonometric Polynomials by Sampling along Rank-1 Lattices”, *SIAM Journal on Numerical Analysis*, vol. 51, no. 5, pp. 2773–2796, Jan. 2013. DOI: [10.1137/120871183](https://doi.org/10.1137/120871183).

- [211] L. Kämmerer, D. Potts, and T. Volkmer, “Approximation of multivariate periodic functions by trigonometric polynomials based on sampling along rank-1 lattice with generating vector of Korobov form”, *Journal of Complexity*, vol. 31, no. 3, pp. 424–456, Jun. 2015. DOI: [10.1016/j.jco.2014.09.001](https://doi.org/10.1016/j.jco.2014.09.001).
- [212] G. Plonka, D. Potts, G. Steidl, and M. Tasche, *Numerical Fourier Analysis*. Springer International Publishing, 2018. DOI: [10.1007/978-3-030-04306-3](https://doi.org/10.1007/978-3-030-04306-3).
- [213] L. Kämmerer, S. Kunis, and D. Potts, “Interpolation lattices for hyperbolic cross trigonometric polynomials”, *Journal of Complexity*, vol. 28, no. 1, pp. 76–92, Feb. 2012. DOI: [10.1016/j.jco.2011.05.002](https://doi.org/10.1016/j.jco.2011.05.002).
- [214] R. Cools, F. Y. Kuo, and D. Nuyens, “Constructing lattice rules based on weighted degree of exactness and worst case error”, *Computing*, vol. 87, no. 1-2, pp. 63–89, Jan. 2010. DOI: [10.1007/s00607-009-0076-1](https://doi.org/10.1007/s00607-009-0076-1).
- [215] D. Dũng, V. Temlyakov, and T. Ullrich, *Hyperbolic Cross Approximation*, S. Tikhonov, Ed. Springer International Publishing, 2018. DOI: [10.1007/978-3-319-92240-9](https://doi.org/10.1007/978-3-319-92240-9).
- [216] D. Mumford, *Tata Lectures on Theta I*. Springer-Verlag GmbH, Jun. 2007, 236 pp., ISBN: 0817645772. [Online]. Available: https://www.ebook.de/de/product/11432679/david_mumford_tata_lectures_on_theta_i.html.
- [217] A. K. Lenstra, H. W. Lenstra, and L. Lovász, “Factoring polynomials with rational coefficients”, *Mathematische Annalen*, vol. 261, no. 4, pp. 515–534, Dec. 1982. DOI: [10.1007/bf01457454](https://doi.org/10.1007/bf01457454).
- [218] I. V. Oseledets, “Tensor-Train Decomposition”, *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, Jan. 2011. DOI: [10.1137/090752286](https://doi.org/10.1137/090752286).
- [219] N. Lee and A. Cichocki, “Fundamental tensor operations for large-scale data analysis using tensor network formats”, *Multidimensional Systems and Signal Processing*, vol. 29, no. 3, pp. 921–960, Mar. 2017. DOI: [10.1007/s11045-017-0481-0](https://doi.org/10.1007/s11045-017-0481-0).
- [220] D. Savostyanov and I. Oseledets, “Fast adaptive interpolation of multi-dimensional arrays in tensor train format”, in *The 2011 International Workshop on Multidimensional (nD) Systems*, IEEE, Sep. 2011. DOI: [10.1109/nds.2011.6076873](https://doi.org/10.1109/nds.2011.6076873).
- [221] S. M. Rump, T. Ogita, and S. Oishi, “Accurate Floating-Point Summation Part I: Faithful Rounding”, *SIAM Journal on Scientific Computing*, vol. 31, no. 1, pp. 189–224, Jan. 2008. DOI: [10.1137/050645671](https://doi.org/10.1137/050645671).
- [222] N. J. Higham, “The Scaling and Squaring Method for the Matrix Exponential Revisited”, *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 4, pp. 1179–1193, Jan. 2005. DOI: [10.1137/04061101x](https://doi.org/10.1137/04061101x).
- [223] S. M. Greene and V. S. Batista, “Tensor-Train Split-Operator Fourier Transform (TT-SOFT) Method: Multidimensional Nonadiabatic Quantum Dynamics”, *Journal of Chemical Theory and Computation*, vol. 13, no. 9, pp. 4034–4042, Aug. 2017. DOI: [10.1021/acs.jctc.7b00608](https://doi.org/10.1021/acs.jctc.7b00608).

- [224] S. Dolgov and B. Khoromskij, “Two-Level QTT-Tucker Format for Optimized Tensor Calculus”, *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 2, pp. 593–623, Jan. 2013. DOI: [10.1137/120882597](https://doi.org/10.1137/120882597).
- [225] I. Oseledets, *TT-Toolbox version 2.2.2*, <https://github.com/oseledets/TT-Toolbox>, Online; accessed 3 March 2021.
- [226] *A Balancing Act for the Matrix Exponential*, <https://blogs.mathworks.com/cleve/2012/07/23/a-balancing-act-for-the-matrix-exponential/>, Accessed: 2021-08-06.
- [227] G. M. Phillips and P. J. Taylor, “TAYLOR’s POLYNOMIAL AND SERIES”, in *Theory and Applications of Numerical Analysis*, Elsevier, 1996, pp. 39–51. DOI: [10.1016/b978-012553560-1/50004-5](https://doi.org/10.1016/b978-012553560-1/50004-5).
- [228] J. F. Grcar, “A matrix lower bound”, *Linear Algebra and its Applications*, vol. 433, no. 1, pp. 203–220, Jul. 2010. DOI: [10.1016/j.laa.2010.02.014](https://doi.org/10.1016/j.laa.2010.02.014).
- [229] R. Schweiger, Y. Erlich, and S. Carmi, “FactorialHMM: fast and exact inference in factorial hidden Markov models”, *Bioinformatics*, vol. 35, no. 12, R. Schwartz, Ed., pp. 2162–2164, Nov. 2018. DOI: [10.1093/bioinformatics/bty944](https://doi.org/10.1093/bioinformatics/bty944).
- [230] A. Williams, *Efficient computation of a Kronecker - vector product (with multiple matrices)*. Online; accessed 29 April 2021. [Online]. Available: <https://gist.github.com/ahwillia/f65bc70cb30206d4eade857b98c4065>.
- [231] *Vector Multiplication with Multiple Kronecker Products*, Online; accessed 29 April 2021. [Online]. Available: <https://math.stackexchange.com/questions/1879933/vector-multiplication-with-multiple-kronecker-products%7D>.
- [232] S. Chimmalgi and S. Wahls, “Theoretical analysis of maximum transmit power in a b-modulator”, in *45th European Conference on Optical Communication (ECOC 2019)*, Institution of Engineering and Technology, 2019, pp. 1–3. DOI: [10.1049/cp.2019.0941](https://doi.org/10.1049/cp.2019.0941).
- [233] S. Civelli, E. Forestieri, and M. Secondini, “Why Noise and Dispersion May Seriously Hamper Nonlinear Frequency-Division Multiplexing”, *IEEE Photonics Technology Letters*, vol. 29, no. 16, pp. 1332–1335, Aug. 2017. DOI: [10.1109/lpt.2017.2722040](https://doi.org/10.1109/lpt.2017.2722040).
- [234] S. T. Le and H. Buelow, “High Performance NFDm Transmission with *b*-modulation”, in *Photonic Networks; 19th Itg-symposium*, Berlin: VDE Verlag GmbH, Jun. 2018, pp. 1–6, ISBN: 9783800746842. [Online]. Available: <https://ieeexplore.ieee.org/document/8436129>.
- [235] S. Wahls, “Second Order Statistics of the Scattering Vector Defining the D-T Nonlinear Fourier Transform”, in *SCC 2017; 11th International ITG Conference on Systems, Communications and Coding*, 2017, pp. 1–6.
- [236] L. Bluestein, “A Linear Filtering Approach to the Computation of Discrete Fourier Transform”, *IEEE Transactions on Audio and Electroacoustics*, vol. 18, no. 4, pp. 451–455, Dec. 1970. DOI: [10.1109/tau.1970.1162132](https://doi.org/10.1109/tau.1970.1162132).

- [237] S. Chimmalgi and S. Wahls, "Bounds on the Transmit Power of b -Modulated NFD Systems in Anomalous Dispersion Fiber", *Entropy*, vol. 22, no. 6, p. 639, Jun. 2020. DOI: [10.3390/e22060639](https://doi.org/10.3390/e22060639).
- [238] V. Aref, S. T. Le, and H. Buelow, "Does the Cross-Talk Between Nonlinear Modes Limit the Performance of NFD Systems?", in *2017 European Conference on Optical Communication (ECOC)*, IEEE, Sep. 2017, pp. 1–3. DOI: [10.1109/ecoc.2017.8346042](https://doi.org/10.1109/ecoc.2017.8346042).
- [239] S. Civelli, E. Forestieri, and M. Secondini, "Nonlinear Frequency Division Multiplexing: Immune to Nonlinearity but Oversensitive to Noise?", in *Optical Fiber Communication Conference (OFC) 2020*, OSA, 2020, pp. 1–3. DOI: [10.1364/ofc.2020.w4a.7](https://doi.org/10.1364/ofc.2020.w4a.7). [Online]. Available: <https://www.osapublishing.org/viewmedia.cfm?uri=OFC-2020-W4A.7&seq=0>.
- [240] X. Zhou, "Direct and inverse scattering transforms with arbitrary spectral singularities", *Communications on Pure and Applied Mathematics*, vol. 42, no. 7, pp. 895–938, Oct. 1989. DOI: [10.1002/cpa.3160420702](https://doi.org/10.1002/cpa.3160420702).
- [241] E. Fagerstrom, "On the Nonlinear Schrodinger Equation with Nonzero Boundary Conditions", Ph.D. dissertation, State University of New York at Buffalo, Jan. 2015. [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2015PhDT.....37F>.
- [242] S. Wahls, S. Chimmalgi, and P. J. Prins, "Wiener-Hopf Method for b -Modulation", in *Optical Fiber Communications Conference and Exhibition (OFC)*, Place of publication not identified: IEEE, 2019, pp. 1–3, ISBN: 9781728136202. [Online]. Available: <https://ieeexplore.ieee.org/document/8696460>.
- [243] W. B. Gearhart and H. S. Shultz, "The Function $\sin x/x$ ", *The College Mathematics Journal*, vol. 21, no. 2, p. 90, Mar. 1990, ISSN: 0746-8342. DOI: [10.2307/2686748](https://doi.org/10.2307/2686748). [Online]. Available: <https://www.maa.org/programs/maa-awards/writing-awards/the-function-sinx>.
- [244] N. B. Andersen, "Entire L_p -functions of exponential type", *Expositiones Mathematicae*, vol. 32, no. 3, pp. 199–220, 2014. DOI: [10.1016/j.exmath.2013.10.003](https://doi.org/10.1016/j.exmath.2013.10.003).
- [245] K. Duda, T. P. Zielinski, and S. H. Barczentewicz, "Perfectly Flat-top and Equiripple Flat-top Cosine Windows", *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 7, pp. 1558–1567, Jul. 2016. DOI: [10.1109/tim.2016.2534398](https://doi.org/10.1109/tim.2016.2534398).
- [246] S. Krantz, "A Primer of Real Analytic Functions", in *A Primer of Real Analytic Functions*. Boston, MA: Birkhäuser Boston Imprint Birkhäuser, 2002, ch. "Elementary Properties", ISBN: 9780817681340. [Online]. Available: <https://link.springer.com/book/10.1007/978-0-8176-8134-0>.
- [247] S. Wahls, S. Chimmalgi, P. J. Prins, and M. Brehler, *FastNFT/FNFT: Version 0.4.1*, 2020. DOI: [10.5281/ZENODO.3941976](https://doi.org/10.5281/ZENODO.3941976).

- [248] S. Wahls and H. V. Poor, “Inverse Nonlinear Fourier Transforms Via Interpolation: The Ablowitz-Ladik Case”, in *Proc. Int. Symp. Math. Theory Networks Systems (MTNS)*, Groningen. The Netherlands, Jul. 2014, pp. 1848–1855.
- [249] B. Kernighan, *The C programming language*. Englewood Cliffs, N.J: Prentice-Hall, 1978, ISBN: 9780131101630.
- [250] D. J. Higham and N. J. Higham, *MATLAB guide*. Siam, 2016, vol. 150.
- [251] *GNU General Public License*, version 2, Free Software Foundation, Jun. 1991. [Online]. Available: <https://www.gnu.org/licenses/old-licenses/gpl-2.0.html>.
- [252] C. Mahnke, *A Python wrapper for FNFT*, Accessed: 30-Sept-2021, 2021. [Online]. Available: <https://github.com/xmhk/FNFTpy>.
- [253] J. Skaar, L. Wang, and T. Erdogan, “On the synthesis of fiber Bragg gratings by layer peeling”, *IEEE Journal of Quantum Electronics*, vol. 37, no. 2, pp. 165–173, 2001. DOI: [10.1109/3.903065](https://doi.org/10.1109/3.903065).
- [254] J. Mertsching, “Quasiperiodic Solutions of the Nonlinear Schrödinger Equation”, *Fortschritte der Physik/Progress of Physics*, vol. 35, no. 7, pp. 519–536, 1987. DOI: [10.1002/prop.2190350704](https://doi.org/10.1002/prop.2190350704).
- [255] L. de Vries, “Fast Numerical Nonlinear Fourier Transform Algorithms for the Manakov Equation”, M.S. thesis, Delft University of Technology, Oct. 2021. [Online]. Available: <http://resolver.tudelft.nl/uuid:0276e693-3408-4472-9749-b754c2114183>.
- [256] O. Kotlyar, M. Pankratova, M. Kamalian-Kopae, A. Vasylichenkova, J. E. Prilepsy, and S. K. Turitsyn, “Combining nonlinear Fourier transform and neural network-based processing in optical communications”, *Optics Letters*, vol. 45, no. 13, p. 3462, Jun. 2020. DOI: [10.1364/ol.394115](https://doi.org/10.1364/ol.394115).
- [257] Z. Zheng, X. Zhang, R. Yu, L. Xi, and X. Zhang, “Frequency offset estimation for nonlinear frequency division multiplexing with discrete spectrum modulation”, *Optics Express*, vol. 27, no. 20, p. 28 223, Sep. 2019. DOI: [10.1364/oe.27.028223](https://doi.org/10.1364/oe.27.028223).
- [258] E. Bidaki and S. Kumar, “Enhanced-power NFDm transmission system with mid-point optical phase conjugation”, *Optics Letters*, vol. 45, no. 17, p. 4682, Aug. 2020. DOI: [10.1364/ol.400815](https://doi.org/10.1364/ol.400815).
- [259] S. Medvedev, I. Chekhovskoy, I. Vaseva, and M. Fedoruk, “Conservative multi-exponential scheme for solving the direct Zakharov–Shabat scattering problem”, *Optics Letters*, vol. 45, no. 7, p. 2082, Mar. 2020. DOI: [10.1364/ol.387436](https://doi.org/10.1364/ol.387436).
- [260] G. Van Rossum and F. L. Drake Jr, *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.
- [261] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing, “Jupyter Notebooks – a publishing format for reproducible computational workflows”, in *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, F. Loizides and B. Schmidt, Eds., IOS Press, 2016, pp. 87–90.

- [262] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, “Array programming with NumPy”, *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. DOI: [10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2). [Online]. Available: <https://doi.org/10.1038/s41586-020-2649-2>.
- [263] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, Í. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. 1. Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”, *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2).
- [264] J. D. Hunter, “Matplotlib: A 2D graphics environment”, *Computing In Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007. DOI: [10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55).
- [265] *SSPROP homepage*, Accessed: 30-Sept-2021, 2021. [Online]. Available: <https://photonics.umd.edu/software/ssprop/>.
- [266] S. Medvedev, I. Chekhovskoy, I. Vaseva, and M. Fedoruk, “Fast sixth-order algorithm based on the generalized Cayley transform for the Zakharov-Shabat system in optical applications”, Nov. 2020. arXiv: [2011.11380 \[math.NA\]](https://arxiv.org/abs/2011.11380).

LIST OF PUBLICATIONS

JOURNAL ARTICLES

4. **S. Chimmalgi** and S. Wahls, "Bounds on the Transmit Power of b-Modulated NFDN Systems in Anomalous Dispersion Fiber," in *Entropy: Special Issue Information Theory of Optical Fiber*, Vol. 22, No. 6, Article 639, 2020.
3. V. Bajaj, **S. Chimmalgi**, V. Aref and S. Wahls, "Exact NFDN Transmission in the Presence of Fiber-Loss," in *IEEE/OSA Journal of Lightwave Technology*, Vol. 38, No. 11, pp. 3051–3058, Jun. 2020.
2. **S. Chimmalgi**, P. J. Prins and S. Wahls, "Fast Nonlinear Fourier Transform Algorithms Using Higher Order Exponential Integrators," in *IEEE Access*, Vol. 7, No. 1, pp. 145161–145176, Dec. 2019.
1. S. Wahls, **S. Chimmalgi** and P. J. Prins, "FNFT: A Software Library for Computing Nonlinear Fourier Transforms," in *The Journal of Open Source Software*, vol. 3, no. 23, paper 597, 2018.

CONFERENCE PROCEEDINGS

6. **S. Chimmalgi** and S. Wahls, "Theoretical Analysis of Maximum Transmit Power in a b-Modulator," in *Proc. 45th European Conference on Optical Communication (ECOC)*, Dublin, Ireland, Sep. 2019.
5. V. Bajaj, **S. Chimmalgi**, V. Aref and S. Wahls, "Exact nonlinear frequency division multiplexing in lossy fibers," in *Proc. 45th European Conference on Optical Communication (ECOC)*, Dublin, Ireland, Sep. 2019.
4. S. Wahls, **S. Chimmalgi**, P. J. Prins, "Wiener-Hopf Method for b-Modulation," in *Proc. Optical Fiber Communication Conference and Exhibition (OFC)*, San Diego, USA, Mar. 2019.
3. M. Brehler, C. Mahnke, **S. Chimmalgi** and S. Wahls, "NFDMLab: Simulating Nonlinear Frequency Division Multiplexing in Python," in *Proc. Optical Fiber Communication Conference and Exhibition (OFC)*, San Diego, USA, Mar. 2019.
2. **S. Chimmalgi**, P. J. Prins and S. Wahls, "Nonlinear Fourier Transform Algorithm Using a Higher Order Exponential Integrator," in *Proc. OSA Advanced Photonics Congress*, Zuerich, Switzerland, Jul. 2018.
1. **S. Chimmalgi** and S. Wahls, "Discrete Darboux based Fast Inverse Nonlinear Fourier Transform Algorithm for Multi-solitons," in *Proc. European Conference on Optical Communication (ECOC)*, Gothenburg, Sweden, Sep. 2017.

CURRICULUM VITÆ

Shrinivas CHIMMALGI

30-10-1993 Born in Panvel, India

EDUCATION

2011–2015 Bachelor of Technology in Mechanical Engineering
National Institute of Technology Karnataka, Surathkal, India

2015–2017 MSc Systems & Control Engineering
Delft University of Technology, Delft, The Netherlands

2017–2021 PhD Systems & Control
Delft University of Technology, Delft, The Netherlands
Thesis: Fiber-Optic Communications using Nonlinear
Fourier Transforms: Algorithms and a Bound
Promoters: Dr.-Ing. S. Wahls & Prof. dr. ir. M. Verhaegen

AWARDS

2015 Institute medal, 1986 Batch Gold Medal and Prof. Shuichi Torii
Gold Medal for securing the first rank among 151 students in the
four year Bachelors program

2015 Delft Center for Systems and Control Academic Excellence Fellow-
ship 2015- 2016