

Numerical Methods for the Inverse Nonlinear Fourier Transform

Stella Civelli, Luigi Barletti
 Dipartimento di Matematica e Informatica “U. Dini”
 Università degli Studi di Firenze
 50134 Firenze, Italy
 Email: stellac@garmentgroup.it, luigi.barletti@unifi.it

Marco Secondini
 TeCIP Institute
 Scuola Superiore Sant’Anna
 56124 Pisa, Italy
 Email: marco.secondini@sssup.it

Abstract—We introduce a new numerical method for the computation of the inverse nonlinear Fourier transform and compare its computational complexity and accuracy to those of other methods available in the literature. For a given accuracy, the proposed method requires the lowest number of operations.

Index Terms—Nonlinear Fourier transform, inverse scattering transform, nonlinear Schrödinger equation, optical fiber communication.

I. INTRODUCTION

The propagation of light in optical fibers is governed by the nonlinear Schrödinger equation (NLSE), which accounts for the interplay of dispersion and nonlinearity. In the lossless case, the NLSE admits an analytical solution based on the inverse scattering transform (also known as nonlinear Fourier transform—NFT) [1]. On this basis, during the ’80s and the ’90s, research on soliton transmission in optical fibers was carried out extensively. However, despite a great effort, soliton systems never caught on commercially and were eventually abandoned in favor of simpler and more effective transmission techniques originally developed for linear channels. Recently, an alternative approach to communication over nonlinear channels—originally proposed in [2] and further generalized in [3] and [4]—is gaining new attention from the optical fiber community. Rather than using solitons as information carriers, the idea is that of encoding information onto the discrete and/or continuous spectrum of the Zakharov-Shabat operator (the scattering data), which evolves trivially and linearly along the fiber, such that the encoded information can be easily and accurately recovered from the optical signal after propagation at any distance, without any impact from dispersion and nonlinearity. This approach requires the computation of the inverse NFT at the transmitter to obtain the transmitted time-domain signal from the modulated scattering data, and the computation of the direct NFT at the receiver to extract the scattering data from the received time-domain signal. Both operations are, in general, quite involved and research on efficient numerical methods for their implementation is in progress [4]–[9].

In this work, we focus on the numerical computation of the inverse NFT. We consider two methods available in the literature [4], [5] and propose an alternative method based on iterated convolutions evaluated through the fast Fourier

transform (FFT) algorithm. All the methods are tested for the focusing NLSE in the soliton-free case (i.e., when only the continuous spectrum of the Zakharov-Shabat operator is present), which is of particular interest for information transmission based on nonlinear inverse synthesis [4]. Finally, we investigate and compare the accuracy and computational complexity of all the methods.

II. NUMERICAL METHODS

We consider the focusing NLSE in the normalized form

$$ju_z + \frac{1}{2}u_{tt} + u|u|^2 = 0 \quad (1)$$

We assume to know the scattering data at a given distance z and we focus on the inverse scattering problem (or inverse NFT) to obtain the corresponding NLSE solution $u(t, z)$.¹ This problem leads to the Gel’fand-Levitan-Marchenko equation (GLME) [10]

$$K(t, y) + F(t+y) + \int_{-\infty}^t \int_{-\infty}^t K(t, r)F^*(r+s)F(s+y)dsdr = 0 \quad (2)$$

where the integral kernel $F(y)$ is a function of the scattering data. Specifically, considering the soliton-free case, $F(y) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} r(\lambda)e^{-j\lambda y}d\lambda$ where $r(\lambda)$ is the reflection coefficient. The solution of the NLSE is finally obtained as $u(t) = 2 \lim_{y \rightarrow t^-} K(t, y)$.

Let $F'(y) = F(y)$ for $y \geq 0$ and $F'(y) = 0$ otherwise. The GLME can be rewritten in the form of the Marchenko integral equations

$$\begin{aligned} B_1(t, \alpha) &= - \int_0^{+\infty} F'^*(2t-\alpha-\beta)B_2(t, \beta)d\beta \\ B_2(t, \alpha) &= \int_0^{+\infty} F'(2t-\alpha-\beta)B_1(t, \beta)d\beta - F'(2t-\alpha) \end{aligned} \quad (3)$$

and the NLSE solution obtained as $u(t) = 2B_2(t, 0^+)$. In the following we assume to know F on a uniform grid in the interval $[0, 2T]$ with step $\delta_\alpha = \frac{2T}{N_F}$ and we analyze three different numerical methods to solve the system (3), in order to find the solution of (1) for $t \in [0, T]$. Specifically, we

¹Hereafter, the dependence on z is inessential and will be omitted.

consider the uniform grid on $[0, T]$ given by $t_m = (m-1)\delta_t$ for $m = 1, \dots, N_u + 1$ with discretization step $\delta_t = \frac{T}{N_u}$. The complexity of the methods is measured by considering the required number of complex products, assuming that the FFT of a vector of N elements requires $\frac{N}{2} \log_2(N)$ complex products.

An efficient numerical way to compute the inverse NFT, assuming $N_F = N_u$, and therefore $\delta_t = \frac{\delta_\alpha}{2}$, is presented in [4]. The authors, using the Nyström method with a rectangular quadrature scheme to approximate the integrals in (3), find the solution $u(t_m)$ by solving a linear system characterized by a Toeplitz matrix of size $N_F + m$. The method of Trench for the inversion of non-Hermitian Toeplitz matrices [11] is used to recursively compute the solution for every value t_m . For this reason, let us name this method Nyström-Trench (NT). The resulting algorithm computes the solution $u(t)$ in the desired interval with complexity²

$$C \simeq 11N_F^2 \quad (4)$$

However, due to the recursive nature of the algorithm, the complexity remains almost unchanged if the solution is computed in a lower number of points N_u .

Another technique to solve system (3) with $\delta_t = c\frac{\delta_\alpha}{2}$ for $c \in \mathbb{N}$, (i.e., in $N_u = N_F/c$ points in the interval $[0, T]$) is presented in [5], where the Nyström method with composite Simpson's quadrature rule is considered. For a fixed time t_m , the resulting linear system is solved by means of the conjugate gradient method. Therefore, we refer to this scheme as the Nyström conjugate gradient (NCG) method. Since we deal only with diagonal and upper-left triangular Hankel matrices, the complexity of the method can be reduced by using the FFT algorithm to compute the matrix-vector products required by the conjugate gradient method. As a result, the number of operations required to compute the solution in the desired interval is²

$$C \simeq (1 + k_{\max}) \sum_{m=1}^{N_u+1} 6[c(m-1) + 1] \log_2\{2[c(m-1) + 1]\} + N_u N_F (4 + 6k_{\max}) \quad (5)$$

where k_{\max} is the number of iterations of the conjugate gradient method, which is assumed to be the same for every t for the sake of simplicity. In this case, the solution is independently evaluated at each time t_m in the given interval, such that C depends significantly on the number of points N_u .

Now, let us define a new iterative method to solve the system (3) for $t \in [0, T]$ with discretization step $\delta_t = c\frac{\delta_\alpha}{2}$. The main idea is that, by defining the auxiliary functions $B_i'(t, \alpha) = B_i(t, \alpha)$ for $\alpha \geq 0$ and $B_i'(t, \alpha) = 0$ otherwise, with $i = 1, 2$, the integrals in (3) can be seen as convolutions between these functions and the kernel F and efficiently computed by means of the FFT algorithm. Therefore, considering a proper

starting value for B_1 and iteratively updating the values of B_2 and B_1 by alternately computing the integrals in (3), we define an iterative method that, under certain conditions on F and t , converges to the solution of the system. As a starting point for B_1 at t_m , we pick the solution found at the previous point t_{m-1} , as we are dealing with continuous functions. The resulting algorithm is

For $k = 0$

$$B_1^{(0)}(\alpha, t_m) = \begin{cases} 0 & \text{if } m = 1 \\ B_1^{(k_{\max})}(\alpha, t_{m-1}) & \text{if } m > 1 \end{cases}$$

For $k = 1, \dots, k_{\max}$

$$\begin{cases} B_2^{(k)}(\alpha, t_m) = -F(2t_m - \alpha) + (F \star B_1^{(k-1)})(2t_m - \alpha) \\ B_1^{(k)}(\alpha, t_m) = -(F^* \star B_2^{(k)})(2t_m - \alpha) \end{cases} \quad (6)$$

where \star denotes convolution, $B_i^{(k)}$ is the k th estimate of the function B_i , and $B_1^{(k)}$ the corresponding auxiliary function. We refer to this method as the iterative convolution (IC) method. The number of operations required to find the solution in the N_u points inside the interval $[0, T]$ is

$$C \simeq k_{\max} \sum_{m=1}^{N_u+1} 3[2c(m-1) + 1] \log_2[2c(m-1) + 1] + 2k_{\max} N_u N_F \quad (7)$$

Also in this case, C depends significantly on the number of points N_u and can be reduced by evaluating the solution with a larger step size.

III. NUMERICAL RESULTS

With the purpose of comparing the methods presented above, we consider the single-pulse signal considered in [4]

$$u(t) = -\frac{4\alpha\nu\sigma(\sigma-1)}{(\sigma-1)^2 e^{-2\sigma\alpha t} + \nu^2 e^{2\sigma\alpha t}} \quad (8)$$

where $\sigma = \sqrt{\nu^2 + 1}$. In this case, the scattering data are known and the GLME kernel is [4]

$$F(y) = \alpha\nu e^{-\alpha y} \quad (9)$$

for $\alpha > 0$ and $-1 \leq \nu \leq 1$. We consider the interval $[0, 3]$ and show the results only for $\alpha = \nu = 1$, as similar results are obtained for different values of the parameters.

In Fig. 1 we compare the analytical solution (8) with the numerical results obtained with $\delta_\alpha = 0.01$ and $\delta_t = 0.005$. At this scale, all the numerical results are practically superimposed (and are, therefore, represented by a single curve to avoid cluttering the figure) and equal the exact solution. Since the new IC method is an iterative method, whose starting point at t_m depends on the previous solution found at t_{m-1} , it is interesting to investigate its convergence properties when the solution is evaluated with a different resolution δ_t (keeping unchanged the resolution δ_α for the evaluation of the integrals). Fig. 2 shows the modulus of the error, i.e., the difference between (8) and the numerical approximation, as a function of t for $\delta_\alpha = 0.01$, different iteration numbers k , and different resolutions δ_t . In all the cases, three iterations

²The computational complexity of the methods presented in [4] and [5] is not indicated by the authors. The values reported here refer to the most efficient implementations that we were able to devise.

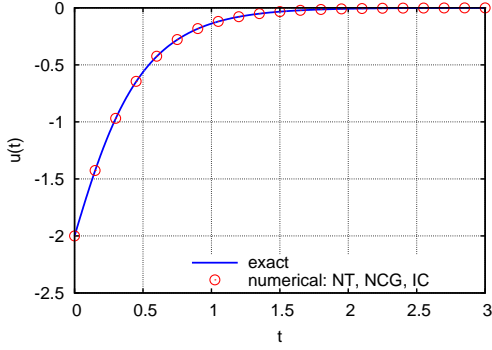


Fig. 1. Comparison between the exact solution (8) and the solution obtained with the NT, NCG, and IC numerical methods.

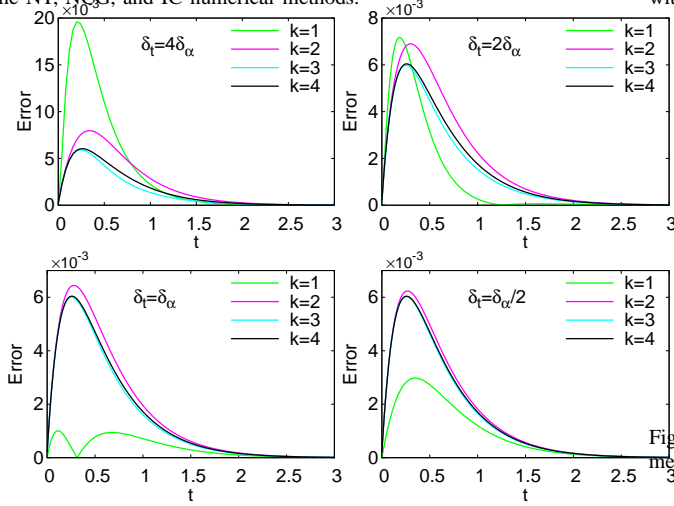


Fig. 2. IC method: modulus of the error for different iterations k , different resolutions δ_t , and a fixed resolution $\delta_\alpha = 0.01$.

are enough to reach the convergence. However, in the case $\delta_t = \delta_\alpha$, the error after the first iteration is significantly lower than the error obtained after the convergence is reached. This quite surprising result has been observed also for different resolutions δ_α and for different pulse shapes (not shown here due to a lack of space). However, we have not been able to find a good explanation for this behavior, which is still under investigation. Given its particular properties, in the rest of the paper we will refer to the IC method with $k_{\max} = 1$ and $\delta_t = \delta_\alpha$ as the IC1 method.

Fig. 3 compares the convergence properties of the IC method (for $\delta_t = \delta_\alpha = 0.01$) to those of the NCG method (for $\delta_\alpha = 0.01$ and any δ_t , as they are independent of δ_t), showing how the root-mean-square error (RMSE) evaluated over the whole interval $[0, 3]$ evolves with the number of iterations. Thanks to the composite Simpson's quadrature rule employed by the NCG method (which is more accurate than the rectangular quadrature rule implicitly employed by the IC method when using the FFT to compute the convolutions), the final RMSE reached by the NCG method is lower than the RMSE reached by the IC method. However, the IC method

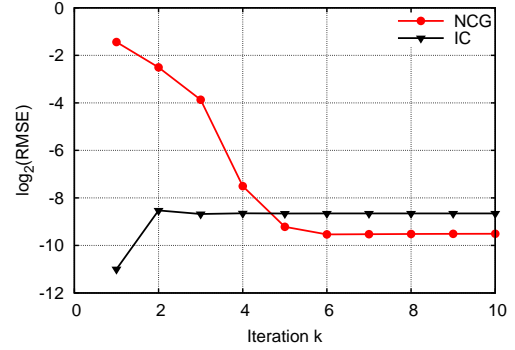


Fig. 3. RMSE versus number of iterations for the NCG and the IC method with $\delta_\alpha = 0.01$ (and $\delta_t = \delta_\alpha$ for IC).

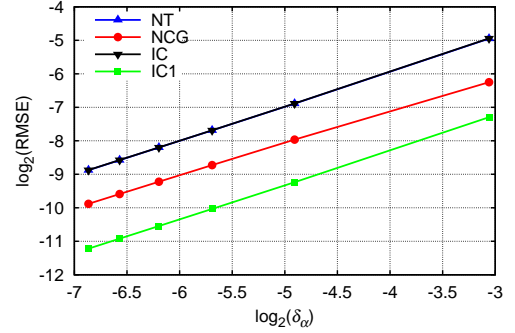


Fig. 4. RMSE as a function of the resolution δ_α for all the numerical methods ($\delta_t = \delta_\alpha/2$ for NT and $\delta_t = \delta_\alpha$ for IC1).

converges faster (after 3 iterations) than the NCG method (after 6 iterations). The same fast convergence of the IC method is observed also for higher values of δ_t (as shown in Fig. 2). However, in the special case $\delta_t = \delta_\alpha$, as observed before, the IC method achieves an even lower RMSE at the first iteration (IC1 method). Such a fast convergence is obtained thanks to the good starting value for B_1 indicated in (6). A convergence of the iterative method is observed in all the considered cases. This is, however, not guaranteed in general, as the convergence depends on specific conditions on F and t that are currently being investigated.

For all the methods, both the accuracy and the complexity depend on the resolution δ_α employed for the quadrature rules. Fig. 4 shows the RMSE as a function of δ_α for all the considered methods. In the NT and IC1 methods, the resolution δ_t is set to $\delta_\alpha/2$ and δ_α by construction, respectively. On the other hand, in the NCG and IC methods, δ_t can be selected as an integer multiple of $\delta_\alpha/2$ without affecting the accuracy. The RMSE values obtained with the NT and IC methods are practically the same, as they are both based on a rectangular quadrature rule for the computation of the integrals in (3). The NCG method, thanks to the more accurate composite Simpson's quadrature rule, achieves a lower RMSE for the same resolution δ_α . Finally, the IC1 method achieves an even lower RMSE.

Fig. 4 can be used to determine, for each method, what is

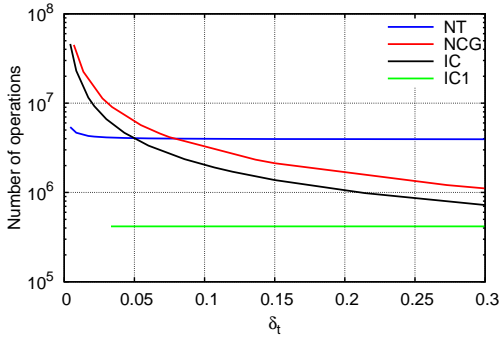


Fig. 5. Number of operations required by the four methods for a fixed RMSE of 2×10^{-3} and a variable desired resolution δ_t .

the resolution δ_α (and, correspondingly, the number of points $N_F = 2T/\delta_\alpha$) required for the computation of the integrals in order to find the solution $u(t)$ with a given accuracy. This, in turn, will affect the complexity of the method according to (4), (5), or (7). Another relevant parameter that affects the computational complexity of the NCG and IC methods (but not that of the NT and IC1 methods) is the resolution $\delta_t = c\delta_\alpha/2$ with which the solution $u(t)$ is needed (or, equivalently, the number of points $N_u = N_F/c$ in which the solution must be computed). Fig. 5 compares the computational complexity required by the various methods, computed according to (4), (5), and (7), to obtain the solution $u(t)$ with a desired RMSE of 2×10^{-3} (about 2^{-9} in Fig. 4), as a function of the desired resolution δ_t . According to Fig. 4, the required resolution is $\delta_\alpha \simeq 0.0086$ for the NT and IC methods, $\delta_\alpha \simeq 0.014$ for the NCG method, and $\delta_\alpha \simeq 0.033$ for the IC1 method. The number of iterations required to reach the convergence is $k_{\max} = 6$ for the NCG method and $k_{\max} = 3$ for the IC method. By construction, the minimum resolution δ_t is $\delta_\alpha/2$ for the NT, NCG, and IC methods, and δ_α for the IC1 method. However, if desired, the resolution δ_t can be increased by computing the solution $u(t)$ in a lower number of points N_u in the given interval $[0, T]$. This will significantly reduce the computational complexity of the NCG and IC methods, but will leave the complexity of the NT and IC1 methods almost unaffected.³ For any resolution δ_t , the NCG and IC methods have a similar complexity. In particular, the latter is slightly more convenient, though the former could be probably improved by using a more accurate quadrature rule or employing preconditioning. Both methods are more complex than the NT method when employed at full resolution, but become less complex when the solution is required in a lower number of points (the IC method for $\delta_t > 0.05$ and the NCG method for $\delta_t > 0.075$). Finally, the IC1 method is significantly less complex than the NT method (by about one order of magnitude) and less complex than the NCG and IC methods even at high resolutions δ_t . However, the behavior

³Even if the solution is needed in a lower number of points $N_u < N_F$, the NT and IC1 algorithms must be anyway executed for N_F points. In the NT case, however, some operations can be saved, slightly reducing the computational complexity with respect to what is reported in (4).

of the IC1 method is still under investigation and its superior performance, though observed also in other cases (not reported in this paper for a lack of space), has not been fully explained nor demonstrated for a generic signal.

IV. CONCLUSIONS

A new numerical method for the computation of the inverse nonlinear Fourier transform has been proposed. The solution is obtained after iterated convolutions with the GLME kernel, which are efficiently computed through the FFT algorithm. The accuracy and computational complexity of the proposed method have been investigated and compared to those of two other methods available in the literature, both exploiting the Nyström method in combination with either the conjugate gradient algorithm [5] or the Trench algorithm [4]. The obtained results show that, for a desired accuracy, the proposed method requires the lowest number of operations. The relation between the discretization step with which the integrals of the GLME are approximated and the time resolution with which the solution is computed (which need not be the same) is also discussed for all the methods, and the dependence of accuracy and complexity on those two parameters is investigated. When the two parameters are equal, the proposed method shows a very good but peculiar behavior which is still under investigation. Further investigation is also required to compare the algorithms in different scenarios, such as for the defocusing NLSE or in the presence of discrete eigenvalues in the nonlinear spectrum of the signal.

ACKNOWLEDGMENT

This work was supported in part by the Italian MIUR under the FIRB project COTONE

REFERENCES

- [1] V. E. Zakharov and A. B. Shabat, "Exact theory of two-dimensional self-focusing and one-dimensional self-modulation of waves in nonlinear media," *Sov. Phys.-JETP*, vol. 34, pp. 62–69, 1972.
- [2] A. Hasegawa and T. Nyu, "Eigenvalue communication," *J. Lightwave Technol.*, vol. 11, no. 3, pp. 395–399, Mar. 1993.
- [3] M. I. Yousefi and F. R. Kschischang, "Information transmission using the nonlinear Fourier transform, Part III: Spectrum modulation," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4346–4369, Jul. 2014.
- [4] S. T. Le, J. E. Prilepsky, and S. K. Turitsyn, "Nonlinear inverse synthesis for high spectral efficiency transmission in optical fibers," *Optics express*, vol. 22, no. 22, pp. 26 720–26 741, 2014.
- [5] A. Aricò, G. Rodriguez, and S. Seatzu, "Numerical solution of the nonlinear Schrödinger equation, starting from the scattering data," *Calcolo*, vol. 48, no. 1, pp. 75–88, 2011.
- [6] M. I. Yousefi and F. R. Kschischang, "Information transmission using the nonlinear Fourier transform, Part II: Numerical methods," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4329–4345, Jul. 2014.
- [7] S. Wahls and H. V. Poor, "Fast numerical nonlinear Fourier transforms," *arXiv preprint arXiv:1402.1605*, 2014.
- [8] L. Fermo, C. van der Mee, and S. Seatzu, "Scattering data computation for the Zakharov-Shabat system," *arXiv preprint arXiv:1502.04628*, 2015.
- [9] S. Wahls and H. V. Poor, "Fast inverse nonlinear Fourier transform for generating multi-solitons in optical fiber," *arXiv preprint arXiv:1501.06279*, 2015.
- [10] M. J. Ablowitz and H. Segur, *Solitons and the inverse scattering transform*. SIAM, 1981, vol. 4.
- [11] S. Zohar, "Toeplitz matrix inversion: The algorithm of W. F. Trench," *Journal of the ACM (JACM)*, vol. 16, no. 4, pp. 592–601, 1969.