Discrete orthogonal Gauss–Hermite transform for optical pulse propagation analysis

Pavlos Lazaridis, Guy Debarge, and Philippe Gallion

Département Communications, Unité de Recherche Associée au Centre National de la Recherche Scientifique, 820 Ecole Nationale Supérieure des Télécommunications, 46 rue Barrault, 75634 Paris Cedex 13, France

Received February 14, 2002; revised manuscript received June 24, 2002

A discrete orthogonal Gauss–Hermite transform (DOGHT) is introduced for the analysis of optical pulse properties in the time and frequency domains. Gaussian quadrature nodes and weights are used to calculate the expansion coefficients. The discrete orthogonal properties of the DOGHT are similar to the ones satisfied by the discrete Fourier transform so the two transforms have many common characteristics. However, it is demonstrated that the DOGHT produces a more compact representation of pulses in the time and frequency domains and needs less expansion coefficients for a given accuracy. It is shown that it can be used advantageously for propagation analysis of optical signals in the linear and nonlinear regimes. © 2003 Optical Society of America

OCIS codes: 190.5530, 060.4370, 060.2330, 320.5550, 060.2430, 060.2360.

1. INTRODUCTION

Numerical optical pulse propagation analysis is important for the design of future light-wave communication systems. In particular, the systems that present nonlinearities need efficient numerical solution methods for the modeling equations to perform a significant number of numerical simulations that are necessary for good design. Discrete Fourier transform (DFT) methods with a fast Fourier transform (FFT) algorithm have been used extensively for the study of linear propagation[1] in optical fibers as well as nonlinear propagation and especially soliton propagation. In the latter case, the split-step Fourier (SSF) method is the most widely used because it is stable and efficient.[2–5] By using the DFT for analysis of a propagating pulse, one is essentially expanding the pulse on the basis of orthogonal complex exponential functions. However, many researchers have recognized that it could be advantageous to expand a pulse on an alternative basis of orthogonal functions, most notably the Gauss–Hermite functions.[6–9] Moreover, advantages of the use of a Gauss–Hermite orthogonal basis in place of the complex exponential Fourier basis appeared not only in optics but in other disciplines as well, most importantly in antenna theory,[10,11] electromagnetics,[12] signal theory,[13] image representation and compression,[14,15] and electrocardiogram analysis and compression.[16,17] These previous studies addressed interesting issues such as the convergence properties and the optimum scaling of Gauss–Hermite expansions. It is important to note, however, that the common feature of previous studies is that the Gauss–Hermite basis used satisfies a continuous orthogonal property (with respect to integration) but not a discrete orthogonal property (with respect to summation) as the one satisfied by the DFT.[18] This creates a number of problems and inconveniencies for numerical implementation of the Gauss–Hermite expansions: the expansion coefficients are given by an integral defined over an infinite domain that has to be approximated in some way, the optimum scaling factors depend on this approximation and cannot be uniquely defined, and, most importantly, there is no exact inverse transform if a finite number of basis functions is used. The discrete orthogonal Gauss–Hermite transform (DOGHT) presented in this paper overcomes the above-mentioned problems that are typical of classical Gauss–Hermite orthogonal expansions and has many of the useful properties of a DFT.

2. DISCRETE ORTHOGONAL GAUSS–HERMITE TRANSFORM

The DOGHT is defined by

$$c_n = \sum_{i=0}^{N-1} w_i f\left(\frac{t_i}{T}\right) h_n(t_i), \tag{1}$$

$$\tilde{f}\left(\frac{t_i}{T}\right) = \sum_{n=0}^{N-1} C_n h_n(t_i), \tag{2}$$

where T is a time-scaling factor and

$$h_n(t) = \frac{1}{(2^n n!\sqrt{\pi})^{1/2}} \exp\left(-\frac{t^2}{2}\right) H_n(t). \tag{3}$$

Moreover

$$\int_{-\infty}^{+\infty} h_m(t) h_n(t) \mathrm{d}t = \delta_{mn} \tag{4}$$

is the continuous orthogonal property of the orthonormal Gauss–Hermite functions of order n, and h_n and H_n are the classical Hermite polynomials of order n. The sampling points $t_0 < t_1 < t_2 < \ldots t_{N-1}$ are the zeros of the Nth-order Hermite polynomial that are commonly used in the context of Gauss–Hermite quadrature theory, and weights w_i are given by the relation

0740-3224/2003/071508-06$15.00

$$w_i = \frac{2}{[h_N(t_i)]^2}. \tag{5}$$

Use of the orthonormal Gauss–Hermite functions is convenient for numerical computations. A three-term recurrence relation is used for the calculation:

$$h_0 = \frac{1}{\pi_{1/4}} \exp\left(-\frac{t^2}{2}\right), \qquad h_1 = \sqrt{2} t h_0,$$

$$h_{n+1} = \frac{1}{\sqrt{n}}(\sqrt{2} t h_n - \sqrt{n-1} h_{n-1}). \tag{6}$$

We used a Newton–Raphson root-finding algorithm for the computation of sampling points t_i by finding the zeros of the h_n functions. Initial approximations for the zeros are provided by the fact that the N zeros of h_n interleave the $N-1$ zeros of h_{n-1} and there is exactly one zero of the former between each two adjacent zeros of the latter. This is a well-known property of all classical orthogonal polynomials. The arithmetic mean (midinterval) of the two adjacent zeros is taken as the initial approximation that is refined to the desired accuracy by the Newton–Raphson algorithm. The derivative necessary for this purpose is calculated by

$$h_n = \sqrt{2n} h_{n-1}, \tag{7}$$

and this in turn is used for the calculation of weights w_i by Eq. (5). Using the above-described, reliable procedure, we easily calculated sampling points at least up to $N = 512$ and with 13–14 digit accuracy in double precision.

The important discrete orthogonal relations satisfied by the Gauss–Hermite functions are

$$\sum_{i=0}^{N-1} w_i h_m(t_i) h_n(t_i) = \delta_{mn}, \tag{8}$$

$$\sum_{n=0}^{N-1} w_j h_n(t_i) h_n(t_j) = \delta_{ij}. \tag{9}$$

Because of Eq. (8) the transform coefficients c_n are calculated by Eq. (1). Furthermore, Eq. (9) leads to the fact that

$$\tilde{f}\left(\frac{t_i}{T}\right) = f\left(\frac{t_i}{T}\right), \tag{10}$$

the collocation property of the DOGHT, and this is demonstrated in detail in the Appendix. These properties are equivalent and are similar to the properties of the DFT and result in an exact inverse DOGHT. The Parseval relation for the DOGHT is

$$\tilde{E} = \frac{1}{T} \sum_{n=0}^{N-1} |c_n|^2 = \frac{1}{T} \sum_{i=0}^{N-1} w_i \left| f\left(\frac{t_i}{T}\right)\right|^2, \tag{11}$$

where $\tilde{E}$ is the approximate pulse energy. The exact pulse energy is

$$E = \int_{-\infty}^{+\infty} |f(t)|^2 \mathrm{d}t. \tag{12}$$

The Parseval relation in Eq. (11) is instructive about the number of sampling points needed for a good approximation of the pulse energy, the correct choice of the time-scaling factor T, as well as the rate of convergence of the transform in general. First, Eq. (11) shows that the pulse energy is approximated by a Gauss–Hermite quadrature rule as opposed to the rectangle rule of the Parseval relation for the DFT. As is well known, a Gauss quadrature rule converges much faster than the simple rectangle rule in a vast majority of cases.

The approximate Fourier transform can be also calculated analytically as

$$\begin{aligned} \tilde{F}(\omega) &= \int_{-\infty}^{+\infty} \tilde{f}(t)\exp(-j\omega t)\mathrm{d}t \\ &= \int_{-\infty}^{+\infty} \sum_{n=0}^{N-1} c_n h_n(T_t)\exp(-j\omega t)\mathrm{d}t \\ &= \frac{\sqrt{2\pi}}{|T|} \sum_{n=0}^{N-1} (-j)^n c_n h_n\left(\frac{\omega}{T}\right) \end{aligned} \tag{13}$$

and is in the form of another orthogonal function expansion that is due to the fact that the Gauss–Hermite functions are their own Fourier transforms multiplied by the $\sqrt{2\pi}(-j)^n$-proportionality factor. Contrary to the Fourier transform of the DFT, which provides only discrete frequency values and no continuous approximation formula, the Fourier transform of the DOGHT in Eq. (13) provides a continuous frequency approximation and thus frequency resolution is less of a problem with the DOGHT than with the DFT.

With regard to the choice of the time-scaling factor T of the DOGHT we first propose, in analogy with the DFT, to choose a computational window $[-t_{\max}, t_{\max}]$, taking into account the temporal extent of the pulses under analysis. Then the time-scaling factor is taken from the relation

$$T = \frac{t_{N-1}}{t_{\max}}, \tag{14}$$

where t_{N-1} is the largest Gaussian quadrature node. In the rest of the paper we will use this time-scaling factor.

As an example the DOGHT transform of the $f(t) = \sec h(t)$ function in the interval $t \in [-5, +5]$ with $N = 8$ is calculated and the results are shown in Table 1. The time domain inverse DOGHT transformed function is given by

Table 1. DOGHT of sech(t) for N=8

i	t_i/T	$f(t_i/T)$	c_i	$\tilde{f}(t_i/T)$
0	5.00	0.01	1.06	0.01
1	3.38	0.07	0.00	0.07
2	1.97	0.27	−0.18	0.27
3	0.65	0.82	0.00	0.82
4	−0.65	0.82	0.10	0.82
5	−1.97	0.27	0.00	0.27
6	−3.38	0.07	−0.03	0.07
7	−5.00	0.01	0.00	0.01

$$\tilde{f}(t_i) = \sum_{n=0}^{N-1} c_n h_n(t_i T). \tag{15}$$

The time-scaling factor is T = 0.586 in this case and the approximate pulse energy as calculated by the DOGHT coefficients c_n [Eq. (11)] is $\tilde{E}$=1.97069 whereas the exact pulse energy is $E = 2$. Furthermore, it is obvious that the collocation property (exact inverse transform) of Eq. (10) holds and that the odd coefficients of the DOGHT are zero because of the symmetry of the hyperbolic secant pulse around the origin.

3. PROPAGATION OF OPTICAL PULSES IN THE LINEAR REGIME

The normalized linear propagation equation in optical fibers, including up to second-order dispersion,[5] is

$$j \frac{\partial \varphi(z,\, t)}{\partial z} = \frac{1}{2} \frac{\partial^2 \varphi(z,\, t)}{\partial t^2}, \tag{16}$$

where $\varphi(z,\, t)$ is the normalized electrical field envelope. Taking the Fourier transform of Eq. (16) and solving the resultant ordinary differential equation yield

$$\Phi(z,\, \omega) = \Phi(0,\, \omega) \exp\left(j \frac{\omega z^2}{2}\right). \tag{17}$$

For $\Phi(0, \omega)$ we then chose[9] the set of orthogonal functions:

$$\Phi_n(0,\, \omega) = \sqrt{2\pi}(-j)^n h_n(\omega). \tag{18}$$

The coefficients in Eq. (18) were chosen to yield normalized functions in the time domain. The inverse Fourier transform of Eq. (18) is

$$\varphi_n(z,\, t) = \frac{h_n\left(\dfrac{t}{\sqrt{1 + z^2}}\right)}{(1 + z^2)^{1/4}} \times \exp\left[j(n + 1/2)\arctan z - \frac{j}{2} \frac{z t^2}{1 + z^2}\right]. \tag{19}$$

The finite series expansion of an arbitrary solution of Eq. (16) is written as

$$u(z,\, t) = \sum_{n=0}^{N-1} c_n \varphi_n(z,\, t), \tag{20}$$

where

$$c_n = \sum_{i=0}^{N-1} w_i u(0, t_i) h_n(t_i). \tag{21}$$

However, from a computational point of view it is more practical to be able to define the computational window that was used to solve Eq. (16) independent of the number of expansion functions N used. We achieved this by using, as previously with the DOGHT, a time-scaling factor T. Moreover, taking advantage of the scaling properties of Eq. (16), we used the initial condition of $u(0, t/T) = f(t/T)$ with $-t_{\max} \leq t/T \leq t_{\max}$, and the solution algorithm for Eq. (16) was as follows:

(1) Define a computational window $[-t_{\max}, +t_{\max}]$.

(2) Choose the number N of expansion functions or sampling points.

(3) Calculate the time-scaling factor $T = (t_{N-1}/t_{\max})$.

(4) Calculate N expansion coefficients

$$c_n = \sum_{i=0}^{N-1} w_i u\left(0, \frac{t_i}{T}\right) h_n(t_i).$$

(5) Calculate the solution to Eq. (1) as

$$u(z,\, t) = \sum_{n=0}^{N-1} c_n \varphi_n(z T^2,\, t T).$$

It can easily be seen that for $z = 0$ the above procedure reduces to the DOGHT because $\varphi_n(z = 0, t) = h_n(t)$.

By way of example, the linear propagation of a chirped hyperbolic secant pulse is calculated[9]:

$$u(0, t) = \operatorname{sech}\left(\sqrt{\frac{\pi}{2}} t\right) \exp\left(-j \frac{\alpha t^2}{2}\right). \tag{22}$$

In Eq. (22) $\alpha = -10$, resulting in an initial pulse compression. The complete results are shown in Table 2. The superiority of the DOGHT method, especially at larger distances, is evident. It should also be noted that a smaller computational window is sufficient for the DOGHT, $t \in [-10, +10]$, in comparison with the computational window needed for the DFT–FFT, $t \in [-40, +40]$. This is a general characteristic of the DOGHT and is not specific to this example.

Table 2. Comparison of DOGHT and FFT for the Linear Propagation of a Chirped Hyperbolic Secant Pulse

z	FFT 2048 Points $t \in [-40, +40]$	DOGHT 512 Points $t \in [-10, +10]$	Exact[a]
0.1	3.16	3.16	3.16
1.0	0.329	0.329	0.329
10.0	0.0112	0.0994	0.0994
100.0	0.0682	0.0313	0.0313

[a] From Ref. 9.

4. SPLIT-STEP GAUSS–HERMITE

The split-step Gauss–Hermite algorithm is discussed in Ref. 19. Numerical solution of the nonlinear Schrödinger equation for the simulation of soliton propagation is of great importance to the design of soliton optical communication systems. The finite-difference methods initially used for this purpose[20,21] were rapidly replaced by the SSF method.[2–4] The latter was found to be an order of magnitude faster than the finite-difference methods for the same required precision.[4] The SSF algorithm is an operator splitting method that is used to enable the separate solution of linear and nonlinear parts of an equation. It is a simple, efficient, and stable algorithm that has been used extensively in the study of optical fiber communication systems.[5] A brief description of this algorithm is given in its simplest and most widely used form applied to the nonlinear Schrödinger equation. The nonlinear Schrödinger equation is written in normalized form as

$$\frac{\partial u}{\partial z} = -\frac{i}{2}\frac{\partial^2 u}{\partial t^2} + j|u|^2 u. \tag{23}$$

The SSF algorithm proceeds as follows: First, the nonlinear part of Eq. (23) is solved:

$$\frac{\partial u}{\partial z} = j|u|^2 u, \tag{24}$$

and the solution is propagated by Δz with the formula

$$u_{\mathrm{NL}}(z_0 + \Delta z, t) = \exp[j\Delta z|u(z_0, t)|^2]u(z_0, t). \tag{25}$$

Equation (25) is the exact solution of Eq. (24). Then the linear part of Eq. (23) is solved:

$$\frac{\partial u}{\partial z} = -\frac{j}{2}\frac{\partial^2 u}{\partial t^2} \tag{26}$$

by use of Fourier transforms as

$$u(z_0 + \Delta z, t) = F^{-1}\left[\exp\left(j\frac{\Delta z}{2}\omega\right)F[u_{\mathrm{NL}}(z_0 + \Delta z, t)]\right]. \tag{27}$$

This is also an exact solution. Consequently, Eq. (27) is the approximate solution of Eq. (23), where the linear part and the nonlinear part of Eq. (23) have been solved sequentially. Such a decomposition is, of course, only approximately correct to first order in Δz, its error being of the second order in Δz, according to the operator splitting analysis in Eq. (27). The Fourier transforms in Eq. (27) are approximated by the DFT calculated with the FFT algorithm. This procedure introduces a second approximation error that originates in the discretization of the continuous Fourier transforms by the DFT.

By using the DOGHT instead of the DFT for solution of the linear part of the nonlinear Schrödinger equation we obtained an efficient solution algorithm for solitonlike pulses. More precisely, the FFT linear step of the SSF algorithm [Eq. (27)] is replaced in the split-step Gauss–Hermite (SSGH) method by

$$c_n = \sum_{i=0}^{N-1} w_i u_{\mathrm{NL}}\left(z_0 + \Delta z, \frac{t_i}{T}\right)h_n(t_i), \tag{28}$$

$$u(z_0 + \Delta z, t) = \sum_{n=0}^{N-1} C_n \varphi_n[(\Delta z)T^2, t_i T]. \tag{29}$$

Equation (28) was used to calculate the Gauss–Hermite expansion coefficients and Eq. (29) was used to propagate the solution linearly by Δz. Compared with the results in Ref. 9, it is obvious that the orthogonal expansion used here is a simplified version of the more complicated chirped expansion used in Ref. 9. Another important aspect of our method is use of the Gauss–Hermite quadrature nodes and weights. As a consequence, the sampling points in the time axis are the zeros of the Hermite polynomial of order N, and they are no longer equispaced as in the FFT method but are nonuniformly spaced in an optimum way. The result is that fewer points are required to represent a propagating pulse accurately.

Moreover, with regard to the actual computations, the matrices

$$w_i h_n(t_i), \tag{30}$$

$$\varphi_n[(\Delta z)T^2, t_i T] \tag{31}$$

were calculated only once at the beginning of the algorithm and are then stored for further use. This greatly reduces the computational complexity of the SSGH algorithm. Furthermore, using the fact that the zeros and weights of the Gauss–Hermite quadrature are symmetrically arranged about the origin, we halved the number of complex multiplications required for each linear step. In comparison, the FFT method requires less complex multiplications for a given N but a much higher number of terms N for the same accuracy. A detailed comparison of the SSF and SSGH methods is listed in Tables 3–6. The problems of first- and second-order soliton propagation are used for comparison of the two methods. These cases are convenient for numerical comparison purposes because well-known exact solutions are available.

The initial pulse is

$$u(z = 0, t) = A \operatorname{sec}h(t), \tag{32}$$

where A is the initial amplitude. If $A = 1$, the exact solution of Eq. (23) in this case would be a first-order soliton:

$$u_{\mathrm{ex}}(z, t) = \exp\left(-j\frac{z}{2}\right)\operatorname{sec}ht \tag{33}$$

with constant amplitude and only a varying phase factor. If $A = 2$ the exact solution of Eq. (23) would be a second-order soliton:

Table 3. SSF First-Order Soliton Propagation Simulation[a]

NFFT $t \in [-20, +20]$	$\lvert u - u_{\mathrm{ex}}\rvert_{\max}$	Energy	CPU (s)
64	0.006	2.0000	0.33
128	0.0015	2.0000	0.66
256	0.0015	2.0000	1.37
512	0.0015	2.0000	2.91

[a] $A = 1, z = 10, \Delta z = 0.01$.

Table 4. SSGH First-Order Soliton Propagation Simulation[a]

NGH $t \in [-10, +10]$	$\lvert u - u_{\mathrm{ex}}\rvert_{\max}$	Energy	CPU (s)
8	0.0079	1.9971	0.07
16	0.0026	1.9999	0.11
32	0.0011	2.0000	0.21
64	0.0018	2.0000	0.94

[a] $A = 1, z = 10, \Delta z = 0.01$.

Table 5. SSF Second-Order Soliton Propagation Simulation[a]

NFFT $t \in [-20, +20]$	$\lvert u - u_{ex} \rvert_{max}$	Energy	CPU (s)
256	0.0087	8.0000	1.37
512	0.0080	8.0000	2.91
1024	0.0080	8.0000	6.05

[a] $A = 2$, $z = 1$, $\Delta z = 0.001$.

Table 6. SSGH Second-Order Soliton Propagation Simulation[a]

NGH $t \in [-6, +6]$	$\lvert u - u_{ex} \rvert_{max}$	Energy	CPU (s)
72	0.0084	7.9999	1.37
80	0.0071	8.0000	1.76
100	0.0069	8.0000	2.86

[a] $A = 2$, $z = 1$, $\Delta z = 0.001$.

$$u_{ex}(z,\, t) = 4 \exp\left(-j\frac{z}{2}\right)\frac{[\cosh 3t + 3 \exp(-4jz)\cosh t]}{\cosh 4t + 4 \cosh 2t + 3 \cos 4z}, \tag{34}$$

with a varying amplitude and phase factor. In Tables 3 and 4 the calculations are shown for $A = 1$ and a spatial discretization step of $\Delta z = 0.01$. The maximum absolute error calculated on the sampling points at a normalized distance $z = 10$ is our comparison criterion for first-order soliton propagation:

$$\max\lvert u(z = 10, t_i) - u_{ex}(z = 10, t_i)\rvert. \tag{35}$$

A computational window of $-20 \leqslant t \leqslant +20 = t_{max}$ is used for the SSF method whereas a computational window of $-10 \leqslant t \leqslant +10 = t_{max}$ is used for the SSGH method. All the computations were performed on an Intel mobile Celeron processor-based laptop computer running at 650 MHz.

Table 3 shows that a 128-point FFT is enough, because an additional increase in the number of points does not increase accuracy, the residual error being due to the operator splitting approximation. In comparison, in this case, results of the same accuracy are obtained with the SSGH algorithm with only 32 points and approximately three times faster (see Table 4). Going now to the next example, the maximum absolute error calculated on the sampling points at a normalized distance $z = 1$ is our comparison criterion for second-order soliton propagation:

$$\max\lvert u(z = 1, t_i) - u_{ex}(z = 1, t_i)\rvert. \tag{36}$$

A computational window of $-20 \leqslant t \leqslant +20 = t_{max}$ is used for the SSF method whereas a much narrower computational window of $-6 \leqslant t \leqslant +6 = t_{max}$ is used for the SSGH method.

Tables 5 and 6 show that the SSGH is superior to the SSF also for second-order soliton propagation although the speed advantage has been lost because of the larger number of sampling points required in this case. Furthermore, column 3 in Tables 3–6 show that both methods present good conservation of energy properties. It should also be noted that the SSGH method, unlike the SSF method, does not require N to be a power of 2.

5. CONCLUSIONS

A discrete orthogonal Gauss–Hermite transform and its applications to the analysis of optical pulses have been presented. The DOGHT can be used as an alternative to the DFT for the analysis of optical signals and presents several advantages: more compact representation of signals in the time and frequency domains (less expansion functions and sampling points N required), continuous function and continuous frequency approximation for the Fourier spectrum (as opposed to discrete frequency values for the DFT), and energy approximation of the optical signal by Gaussian quadrature rather than by the less accurate rectangle rule for the DFT. The application of the DOGHT to linear pulse propagation further demonstrated its better accuracy, especially at larger propagation distances, in the example of linear chirped pulse compression. Finally, the split-step Gauss–Hermite algorithm has been presented by use of the DOGHT for a linear step solution of the nonlinear Schrödinger equation. Comparison with the standard split-step Fourier algorithm for first-order and second-order soliton propagation shows the advantages of the DOGHT. Throughout, guidelines for the correct scaling of the DOGHT as well as the choice of adequate computational windows have been given.

APPENDIX

To prove the collocation property of the DOGHT, we substituted the coefficients c_n from Eq. (1) into Eq. (2):

$$\tilde{f}\left(\frac{t_i}{T}\right) = \sum_{n=0}^{N-1} C_n h_n(t_i) = \sum_{n=0}^{N-1}\sum_{j=0}^{N-1} w_j f\left(\frac{t_j}{T}\right) h_n(t_j) h_n(t_i). \tag{A1}$$

Changing the order of summation and rearranging yield

$$\tilde{f}\left(\frac{t_i}{T}\right) = \sum_{j=0}^{N-1} f\left(\frac{t_j}{T}\right)\sum_{n=0}^{N-1} w_j h_n(t_i) h_n(t_j). \tag{A2}$$

The second summation is seen to be equal to δ_{ij} by virtue of the orthogonal property in Eq. (9), so that Eq. (A2) becomes

$$\tilde{f}\left(\frac{t_i}{T}\right) = \sum_{j=0}^{N-1} f\left(\frac{t_j}{T}\right)\delta_{ij} = f\left(\frac{t_i}{T}\right). \tag{A3}$$

Thus the proof is complete.

REFERENCES

1. D. Marcuse, “Pulse distortion in single-mode fibers,” Appl. Opt. **19**, 1653–1660 (1980).
2. R. H. Hardin and F. D. Tappert, “Applications of the split-step Fourier method to the numerical solution of nonlinear and variable coefficient wave equations,” SIAM (Soc. Ind. Appl. Math.) Rev. **15**, 423 (1973).
3. A. Hasegawa and F. Tappert, “Transmission of stationary nonlinear optical pulses in dispersive dielectric fibers. I. Anomalous dispersion,” Appl. Phys. Lett. **23**, 142–144 (1973).

4. T. R. Taha and M. J. Ablowitz, "Analytical and numerical aspects of certain nonlinear evolution equations. II. Numerical, nonlinear Schrödinger equation," J. Comput. Phys. **55**, 203–230 (1984).
5. G. P. Agrawal, *Nonlinear Fiber Optics*, 2nd ed. (Academic, San Diego, Calif., 1995).
6. H. da Silva and J. O'Reilly, "Optical pulse modeling with Hermite–Gaussian functions," Opt. Lett. **14**, 526–528 (1989).
7. S. Dijali, A. Dienes, and J. Smith, "ABCD matrices for dispersive pulse propagation," IEEE J. Quantum Electron. **26**, 1158–1164 (1990).
8. T. Tang, "The Hermite spectral method for Gaussian-type functions," SIAM (Soc. Ind. Appl. Math.) J. Sci. Stat. Comput. **14**, 594–606 (1993).
9. P. Lazaridis, G. Debarge, and P. Gallion, "Exact solutions for linear propagation of chirped pulses using a chirped Gauss–Hermite orthogonal basis," Opt. Lett. **22**, 685–687 (1997).
10. A. Bogush and R. Elkins, "Gaussian field expansions for large aperture antennas," IEEE Trans. Antennas Propag. **34**, 228–243 (1986).
11. J. Murphy and R. Padman, "Phase centers of horn antennas using Gaussian beam mode analysis," IEEE Trans. Antennas Propag. **38**, 1306–1310 (1990).
12. M. Rao, T. Sarkar, T. Anjali, and R. S. Adve, "Simultaneous extrapolation in time and frequency domains using Hermite expansions," IEEE Trans. Antennas Propag. **47**, 1108–1115 (1999).
13. T. Oliveira e Silva and H. J. W. Belt, "On the determination of the optimal center and scale factor for truncated Hermite series," presented at the European Conference on Signal Processing (EUSIPCO-96) Trieste, Italy, September 10–13, 1996, paper PFT.11.
14. J.-B. Martens, "The Hermite transform theory," IEEE Trans. Acoust., Speech, Signal Process. **38**, 1595–1606 (1990).
15. A. M. van Dijk and J.-B. Martens, "Image representation and compression with steered Hermite transforms," Signal Process. **56**, 1–16 (1997).
16. L. Lo Conte, R. Merletti, and G. Sandri, "Hermite expansions of compact support waveforms: applications to myoelectric signals," IEEE Trans. Biomed. Eng. **41**, 1147–1159 (1994).
17. A. I. Rasiah, R. Togneri, and Y. Attikiouzel, "Modelling 1-D signals using Hermite basis functions," IEE Proc. Vision Image Signal Process. **144**(6), 345–354 (1997).
18. A. J. Jerri, "The application of general discrete transforms to computing orthogonal series and solving boundary value problems," Bull. Calcutta Math. Soc. **71**, 177–187 (1979).
19. P. Lazaridis, G. Debarge, and P. Gallion, "Split-step-Gauss-Hermite algorithm for fast and accurate simulation of soliton propagation," Int. J. Numer. Model. **14**, 325–329 (2001).
20. N. J. Zabusky and M. D. Kruskal, "Interaction of 'solitons' in a collisionless plasma and the recurrence of initial states," Phys. Rev. Lett. **15**, 240–243 (1965).
21. J. Satsuma and N. Yajima, "Initial value problems of one-dimensional self-modulation of nonlinear waves in dispersive media," Prog. Theor. Phys. Suppl. **55**, 284–306 (1974).