Fully parallel algorithm for simulating dispersion-managed wavelength-division-multiplexed optical fiber systems

P. M. Lushnikov

Theoretical Division, Los Alamos National Laboratory, MS-B284, Los Alamos, New Mexico 87545, and Landau Institute for Theoretical Physics, Kosygin Street 2, Moscow 117334, Russia

Received November 28, 2001

An efficient numerical algorithm is presented for massively parallel simulations of dispersion-managed wavelength-division-multiplexed optical fiber systems. The algorithm is based on a weak nonlinearity approximation and independent parallel calculations of fast Fourier transforms on multiple central processor units (CPUs). The algorithm allows one to implement numerical simulations $M/2$ times faster than a direct numerical simulation by a split-step method, where M is a number of CPUs in a parallel network. © 2002 Optical Society of America

OCIS codes: 060.2330, 060.5530, 060.4370, 190.5530, 260.2030.

A wavelength-division-multiplexed (WDM) dispersion-managed (DM) optical fiber system is the focus of current research in high-bit-rate optical communications. High capacity of optical transmission is achieved with both wavelength multiplexing and dispersion management (see, e.g., Refs. 1 and 2). Wavelength multiplexing allows the simultaneous transmission of several information channels, modulated at different wavelengths, through the same optical fiber. Dispersion-managed[3–6] optical fiber systems are designed to achieve low (or even zero) path-averaged group-velocity dispersion (GVD) by periodic alternation of the sign of the dispersion along an optical fiber. Second-order GVD (dispersion slope) effects and path-averaged GVD effects cause optical pulses in distinct WDM channels to move with different group velocities. Consequently, modeling of WDM systems requires simulation of a long time interval. Enormous computation resources are necessary to capture accurately the nonlinear interactions between channels, which causes the bit-rate capacity to deteriorate. The large computational resources required for simulating WDM transmission over transoceanic distances make parallel computation necessary. Here an efficient numerical algorithm is developed for massive parallel computation of WDM systems. The required computational time is inversely proportional to the number of parallel processors used. This makes feasible a full-scale numerical simulation of WDM systems on a workstation cluster with a few hundred processors.

Neglecting polarization effects and stimulated Raman scattering and Brillouin scattering, one can describe the propagation of WDM optical pulses in a DM fiber with a scalar nonlinear Schrödinger equation:

$$iA_z - \frac{1}{2}\beta_2(z)A_{tt} - \frac{i}{6}\beta_3(z)A_{ttt} + \sigma(z)|A|^2A = iG(z)A, \quad (1)$$

where $G(z) \equiv \{-\gamma + [\exp(z_a\gamma) - 1]\sum_{k=1}^{N}\delta(z - z_k)\}$; z is the propagation distance along an optical fiber; $A(t, z)$ is the slow amplitude of light; β_2 and β_3 are the first- and second-order GVD, respectively, which are periodic functions of z; $\sigma = (2\pi n_2)/(\lambda_0 A_{\text{eff}})$ is the nonlinear coefficient; n_2 is the nonlinear refractive index; $\lambda_0 = 1.55\ \mu$m is the carrier wavelength; A_{eff} is the effective fiber area; $z_k = kz_a$ ($k = 1, \ldots, N$) are amplifier locations; z_a is the amplifier spacing; and γ is the loss coefficient. Distributed amplification can be also included in $G(z)$.

The change of variables $u = A\exp[-\int_0^z G(z')\mathrm{d}z']$ gives

$$iu_z - \frac{1}{2}\beta_2(z)u_{tt} - \frac{i}{6}\beta_3(z)u_{ttt} + c(z)|u|^2u = 0, \quad (2)$$

where $c(z) \equiv \sigma(z)\exp(2\int_0^z G(z')\mathrm{d}z')$. By applying Fourier transform $\hat{u}(\omega, z) = \int_{-\infty}^{\infty}u(t, z)\exp(i\omega t)\mathrm{d}t$ to Eq. (2), changing variables $\hat{u}(\omega, z) \equiv \hat{\psi}(\omega, z) \times \exp[i\beta(z)]$, $\beta(z)\int_0^z\mathrm{d}z'[\omega^2\beta_2(z') + (\omega^3/3)\beta_3(z')]$ and integrating Eq. (2) over z from z_0 to z, one obtains the following integrodifferential equation:

$$\hat{\psi}(\omega, z) = \hat{\psi}(\omega, z_0) + iR(\hat{\psi}[\omega, z], \omega, z, z_0), \quad (3)$$

where

$$R(\hat{v}[\omega, z], \omega, z, z_0) = \frac{1}{(2\pi)^2}\int \mathrm{d}\omega_1\mathrm{d}\omega_2\mathrm{d}\omega_3\int_{z_0}^{z}\mathrm{d}z' \times c(z')\hat{v}^{(z')}(\omega_1, z')\hat{v}^{(z')}(\omega_2, z')\hat{v}^{*(z')}(\omega_3, z')\exp[-i\beta(z')] \times \delta(\omega_1 + \omega_2 - \omega - \omega_3), \quad \hat{v}^{(z)}(\omega, z) \equiv \hat{v}(\omega, z)\exp[i\beta(z)]. \quad (4)$$

Assume that if the nonlinearity is small, $z_{\text{nl}} \gg z_{\text{disp}}$, where $z_{\text{nl}} \equiv 1/|p|^2$ is a characteristic nonlinear length, $z_{\text{disp}} \equiv \tau^2/|\beta_2|$ is the dispersion length, and p and τ are the typical pulse amplitude and width, respectively. $\hat{\psi}(\omega, z)$ is a slow function of z on any scale $L \ll z_{\text{nl}}$ because all of the fast dependence of $\hat{u}$ is already included in the term $\exp[i\beta(z)]$.[7–9] This term is nothing more than an exact solution of the linear part of Eq. (2). In the first approximation one can neglect the slow dependence of $\hat{\psi}$ on z in the interval $mL \le z < (m + 1)L$; i.e., one can replace $\hat{\psi}[\omega, z]$ with $\hat{\psi}[\omega, mL]$ in the nonlinear term, R (m is an arbitrary nonnegative integer number), and obtain from Eq. (3):

0146-9592/02/110939-03$15.00/0

$$\hat{\psi}[\omega,(m+1)L] = \hat{\psi}(\omega,mL) + iR(\hat{\psi}[\omega,mL],\omega,(m+1)L,mL) + O(L/z_{\mathrm{nl}})^2. \quad (5)$$

The term $O(L/z_{\mathrm{nl}})^2$ indicates the order of accuracy of this approximation. Equation (5) enables one to find $\hat{\psi}[\omega, (m+1)L]$ given $\hat{\psi}(\omega, mL)$. Thus one can recover $u(t, z)$ by using the definition of ψ. However, for WDM simulation, the accuracy $O(L/z_{\mathrm{nl}})^2$ is not always sufficient. The next-order approximation is obtained by inclusion of the first-order correction, $\hat{\psi}^{(1)}(\omega, z)$, in the nonlinear term, R:

$$\hat{\psi}[\omega,(m+1)L] = \hat{\psi}(\omega,mL) + iR(\hat{\psi}^{(1)}[\omega,z],\omega,z,mL) + O(L/z_{\mathrm{nl}})^3, \quad (6)$$

$$\hat{\psi}^{(1)}(\omega,z) \equiv \hat{\psi}(\omega,mL) + iR(\hat{\psi}[\omega,mL],\omega,z,mL). \quad (7)$$

Equations (4), (6), and (7) form a closed set for the approximate calculation of $\hat{\psi}[\omega, (m+1)L]$ given $\hat{\psi}(\omega, mL)$, where $O(L/z_{\mathrm{nl}})^3$ is the accuracy of the approximate solution, which is controlled by the appropriate choice of L. The main obstacle in the numerical integration of Eqs. (4), (6), and (7) is the computation of the integral term $R(\hat{v}[\omega, z], \omega, z, mL)$, which generally requires $M \times N^3$ operations for each iteration, where N is the number of grid points in ω or t space and M is the number of grid points for integration over z. Below, a very efficient numerical algorithm for calculations $R(\hat{v}[\omega, z], \omega, z, mL)$ is presented.

In t space Eq. (4) becomes

$$\hat{F}^{-1}[R(\hat{v}[\omega],\omega,z,mL)] = \int_{mL}^{z} dz' c(z') \mathbf{G}^{(z')}[V^{(z')}(t,z')], \quad (8)$$

where $\hat{F}^{-1}$ is the inverse Fourier transform over ω; $V^{(z)}(t, z) \equiv |v^{(z)}(t, z)|^2 v^{(z)}(t, z)$ and $\mathbf{G}^{(z)}$ is the integral operator corresponding to the multiplication operator $\hat{\mathbf{G}}^{(z)}[\hat{\Psi}^{(z)}(\omega, z)] \equiv \exp[-i\beta(z)]\hat{V}^{(z)}(\omega, z)$ in the ω space. It follows from Eqs. (4) and (6)–(8) that numerical calculation of $R(\hat{A}, \omega)$ requires the following eight steps:

(i) The inverse Fourier transform of $\hat{v}^{(z)}(\omega, mL) = \hat{\psi}(\omega, mL)\exp[i\beta(z)]$ for every value of $z[mL < z \le (m+1)L]$.
(ii) A calculation of $V^{(z)}(t, mL)$ from $v^{(z)}(t, mL)$.
(iii) The forward Fourier transform of $V^{(z)}(t, mL)$.
(iv) A numerical integration (summation) of $c(z') \times \exp[-i\beta(z)]\hat{V}^{(z')}(\omega)$ over z' (from $z' = mL$ to $z' = z$) for every values of ω and $z[mL < z \le (m+1)L]$. This integration gives $\hat{\psi}^{(1)}(\omega, z)$ according to Eq. (7).
(v) The inverse Fourier transform of $\hat{v}^{(z)}(\omega, z) = \hat{\psi}^{(1)}(\omega, z)\exp[i\beta(z)]$ for every value of z, $mL < z \le (m+1)L$. [Note that in contrast with step (i) it is necessary to take into account the dependence of $\hat{\psi}^{(1)}$ on z.]
(vi)–(viii) These steps are similar to steps (ii)–(iv) except that the new value of $\hat{v}^{(z)}(\omega, z)$ is used that was obtained in step (v).

The forward and inverse Fourier transforms can be performed with the fast Fourier transform, which requires $N \mathrm{Log}_2(N)$ numerical operations. Steps (i)–(iii) need only the value of $\psi(t, mL)$. These steps can be performed independently and simultaneously in a network of M central processor units (CPUs), shown schematically in Fig. 1. The number of CPUs, M, coincides with the number of grid points for integration over z. Thus the effective computational time equals the time necessary to perform $2N \mathrm{Log}_2(N)$ operations on complex numbers in one CPU. As shown below, to estimate effective computational time one always refers to the number of numerical operations in one CPU if all calculations can be implemented simultaneously in different CPUs without communication between them.

The resulting values of $V^{(z)}(t, mL)$ [after step (iii)] are a set of vectors $\mathbf{a}_m$ $(m = 1, 2, \ldots, M)$ consisting of N complex numbers each. Every vector $\mathbf{a}_m$ is stored in the memory of the mth CPU (or in memory assigned to mth CPU in shared memory network). To perform step (iv) one replaces these vectors by the new vectors $\mathbf{b}_m$: $\mathbf{b}_m = \sum_{j=1}^{m} \mathbf{a}_j$ $(m = 1, 2, \ldots, M)$. Here a simple parallel algorithm is given. Note that this algorithm can be improved, but this improvement is outside the scope of this Letter. It is assumed that M is a power of 2: $M \equiv 2^{M_e}$, where M_e is an integer. The proposed algorithm requires M_e substeps. The vectors $\mathbf{b}_m^{(k)}$ $(m = 1, 2, \ldots, M)$ are results of the kth substep stored in memory, $\mathbf{b}_m^{(Me)} \equiv \mathbf{b}_m$. The first substep is to sum up every pair of vectors: $\mathbf{a}_{2m} + \mathbf{a}_{2m+1}$ to get $\mathbf{b}_1^{(1)} = \mathbf{a}_1$, $\mathbf{b}_2^{(1)} = \mathbf{a}_1 + \mathbf{a}_2, \ldots, \mathbf{b}_{M-1}^{(1)} = \mathbf{a}_{M-1}$, $\mathbf{b}_M^{(1)} = \mathbf{a}_{M-1} + \mathbf{a}_M$. This summation requires N operations. By induction one can see that after k substeps, $\mathbf{b}_m^{(k)} = \sum_{j=1}^{m} \mathbf{a}_j$ for $1 \le m \le 2^k$, $\mathbf{b}_m^{(k)} = \sum_{j=2^k+1}^{m} \mathbf{a}_j$ for $2^k + 1 \le m \le 2^k + 2^k, \ldots, \mathbf{b}_m^{(k)} = \sum_{j=2^{M_e}-2^k+1}^{m} \mathbf{a}_j$ for $2^{M_e} - 2^k + 1 \le m \le 2^{M_e}$. Note that M vectors are now grouped in $M/2^k$ blocks with the appropriate summation inside each block. To perform the $k+1$th substep, it is necessary to double the block size. This can be done by addition of the last element of each odd block to each element of next even block. To do this, one first creates in memory 2^k copies of the last element of each odd block, which requires kN operations in a parallel CPU network. (A number of copies can be doubled by memory forking after each N operations.) To complete the $k+1$th substep,

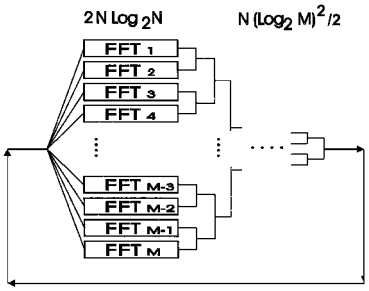

Fig. 1. Schematic of parallel computation algorithm and required number of numerical steps. FFT_1, FFT_2, ... represent fast Fourier transforms in the first CPU, second CPU, etc., respectively. The right-hand side shows schematically calculation of vectors $\mathbf{b}_m$ (see text).

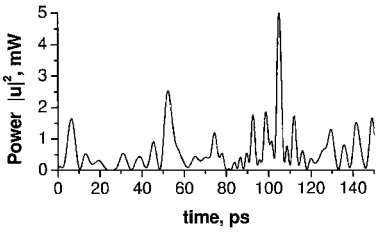

Fig. 2. Power distributions of five WDM channels after propagation of pseudo-random sequences of Gaussian pulses over 10^4 km. Only a small part of the total computational interval of 1000 ps is shown.

it is now enough to simultaneously add 2^k copies to each element in the even block, requiring N operations. The total number of operations for step (iv) is $[1 + 2 + \ldots M_e]N = M_e(M_e + 1)/2$. Steps (v)–(viii) can be done in $\sim N[2\,\mathrm{Log}_2(N) + \mathrm{Log}_2(M)]$ operations. [In step (viii) it is only necessary to calculate $\mathbf{b}_M$ requiring $N\,\mathrm{Log}_2[M]$ operations.] Thus the total number of operation for steps (i)–(viii) is

$$N\{4\,\mathrm{Log}_2(N) + \mathrm{Log}_2(M) + \mathrm{Log}_2(M)\,[\mathrm{Log}_2(M) + 1]/2\}$$

$$\sim N\left[4\,\mathrm{Log}_2(N) + \frac{\mathrm{Log}_2(M)^2}{2}\right]. \qquad (9)$$

Direct solution of Eq. (2) by a split-step method with the same accuracy (for the same size of numerical step, L/M, and the same number of points N in ω space) requires $2MN\,\mathrm{Log}_2(N)$ operations. Comparing this with expression (9), one can conclude that the proposed parallel algorithm allows one to do numerical simulations with the same numerical accuracy $\sim M/2$ times faster using a network of M parallel CPUs. However, the proposed algorithm is approximately two times slower if only one CPU is used.

Numerical simulations of the WDM system were performed by use of both the split-step method for nonlinear Schrödinger equation (2) and the numerical algorithm given by Eqs. (4), (6), and (7) to demonstrate the accuracy of the proposed numerical scheme. Simulations were performed for five WDM channels (20 Gbits/s per channel) over a typical transoceanic distance of 10^4 km. The channel spacing was 0.6 nm. The GVD periodically alternates between spans of standard monomode fiber [$\beta_2^{(1)} = -20.0$ ps^2/km, $\beta_3^{(1)} = 0.1$ ps^2/km, $\sigma_1 = 0.0013$ (km mW)$^{-1}$, length $L_1 = 40$ km] and dispersion-compensating fiber [$\beta_2^{(2)} = 103.9$ ps^2/km, $\beta_3^{(2)} = -0.3$ ps^3/km, $\sigma_2 = 0.00405$ (km mW)$^{-1}$, length $L_2 = -\beta_2^{(1)} L_1/\beta_2^{(2)}$ km] so that the average GVD is zero. Fiber losses and amplifiers were not considered. However, they can be easily included in the coefficient $c(z)$. A pseudo-random binary sequence of length 20 was used for every WDM channel. The boundary conditions are periodic in time. Each binary 1 was represented by an initially zero-chirp Gaussian pulse (return-to-zero format) of 10-ps width and peak power $|u|^2 = 1$ mW at the beginning ($z = 0$) of the fiber line, which is taken at the middle of a standard monomode fiber span. The integration length, L [see Eqs. (4), (6), and (7)], is set to be equal to $(L_1 + L_2)/4$; $M = 2^9$; and $N = 2^{11}$. Figure 2 shows the pulse power distribution (simultaneously in all five channels) after the pulses propagate 10^4 km, obtained from both the split-step and the proposed parallel algorithms. The differences in power distribution between these two simulations are less than 1%, so the two curves are indistinguishable in Fig. 2. Numerical simulations were performed on the usual workstation without the use of parallel computations. The objective of this numerical example is to demonstrate the relative accuracy of the numerical algorithm. Hardware implementation of the parallel simulation for numerical algorithm (4), (6), and (7) is beyond this Letter.

One can conclude that the proposed parallel numerical algorithm allows one to implement numerical simulations of Eq. (1) $\sim M/2$ times faster than a direct numerical simulation of that equation by the split-step method with the same accuracy. The absence of communications between parallel CPUs during the computation of the fast Fourier transform allows one to implement the proposed massive parallel algorithm on workstation clusters.

The author (e-mail address: lushnikov@cnls.lanl.gov) thanks M. Chertkov, I. R. Gabitov, and S. Tretiak for helpful discussions. Support was provided by the U.S. Department of Energy under contract W-7405-ENG-36.

References

1. F. Neddam, P. Le Lourec, B. Biotteau, P. Poignant, E. Pinceman, J.-P. Hamaide, O. Audouin, E. Desurvire, T. Georges, and F. Favre, Electron. Lett. **35,** 1093 (1999).
2. L. F. Mollenauer, P. V. Mamyshev, J. Gripp, M. J. Neubelt, N. Mamysheva, L. Grüner-Nielsen, and T. Veng, Opt. Lett. **25,** 704 (2000).
3. C. Lin, H. Kogelnik, and L. G. Cohen, Opt. Lett. **5,** 476 (1980).
4. C. Kurtzke, IEEE Photon. Technol. Lett. **5,** 1250 (1993).
5. A. R. Chraplyvy, A. H. Gnauck, R. W. Tkach, and R. M. Derosier, IEEE Photon. Technol. Lett. **5,** 1233 (1993).
6. N. J. Smith, F. M. Knox, N. J. Doran, K. J. Blow, and I. Bennion, Electron. Lett. **32,** 54 (1996).
7. I. R. Gabitov and S. K. Turitsyn, Opt. Lett. **21,** 327 (1996).
8. I. Gabitov and S. K. Turitsyn, JETP Lett. **63,** 861 (1996).
9. P. M. Lushnikov, Opt. Lett. **26,** 1535 (2001).