Massively parallel algorithm (MPA) for simulation of GMMNLSE

P. M. Lushnikov[1]
[1] *Department of Mathematics and Statistics, University of New Mexico, Albuquerque, NM 87131, USA**

A massively parallel spectral algorithm for GMMNLSE is described below.

We consider the following form of GMMNLSE following "Multimode Nonlinear Fibre Optics.pdf" (Multimode Nonlinear Fibre Optics: Theory and Applications Peter Horak and Francesco Poletti) notation

$$\frac{\partial}{\partial z}A_p = \mathcal{D}A_p$$
$$+\mathrm{i}\frac{n_2\omega_0}{c}\left(1+\frac{\mathrm{i}}{\omega_0}\frac{\partial}{\partial t}\right)\sum_{l,m,n}\left\{(1-f_R)S^k_{plmn}A_lA_mA^*_n + f_RS^R_{plmn}A_l\left[h*(A_mA^*_n)\right]\right\}, \quad (1)$$

where $\mathcal{D}$ is the linear differential operator over time which in Fourier space is the multiplication operator $\widehat{F}[\mathcal{D}A_p(t,z)] = \mathcal{D}_\omega\hat{A}_p(\omega,z)$, where $\mathcal{D}_\omega(z)$ is assumed to be the given function of the frequency ω and z. Here we define Fourier transform (FT) as $\hat{A}_p(\omega,z) = \widehat{F}_\omega[A_p(t,z)] = \int_{-\infty}^{\infty}A_p(t,z)\exp[\mathrm{i}\omega t]dt$. The summation over l,m,n is taken over the total number of P modes.

We use the following change of variable

$$\hat{A}_p(\omega,z) \equiv \hat{\psi}_p(\omega,z)\exp[\tilde{\mathcal{D}}_\omega(z)], \quad (2)$$

where

$$\tilde{\mathcal{D}}_\omega(z) \equiv \int_{z_0}^{z}\mathcal{D}_\omega(z')dz'. \quad (3)$$

In particular, if $\mathcal{D}_\omega(z)$ does not depend on z, then $\tilde{\mathcal{D}}_\omega(z) = (z-z_0)\mathcal{D}_\omega$.

Taking into account all terms in Eq. (1), using Eq. (2), applying FT to Eq. (1) and integrating over z we obtain the exact expression

$$\hat{\psi}_p(\omega,z) = \hat{\psi}_p(\omega,z_0) + \mathrm{i}\frac{n_2\omega_0}{c}\left(1+\frac{\omega}{\omega_0}\right)$$
$$\times\sum_{l,m,n}\int_{z_0}^{z}\widehat{F}_\omega\left\{(1-f_R)S^k_{plmn}A_l(t,z')A_m(t,z')A^*_n(t,z') + f_RS^R_{plmn}A_l(t,z')\left[h*(A_mA^*_n)\right](t,z')\right\}\mathrm{e}^{-\tilde{\mathcal{D}}_\omega(z')}\mathrm{d}z', \quad (4)$$

where we use the following notation for the convolution

$$\left[h*(A_mA^*_n)\right](t,z') = \int_{-\infty}^{\infty}dt'h(t')A_m(t-t',z')A^*_n(t-t',z'). \quad (5)$$

If the nonlinear terms in Eq. (1) are absent, then Eq. (2) provides the exact solution with $\hat{\psi}(\omega,z)$ being independent on z, i.e. $\hat{\psi}(\omega,z) = \hat{\psi}(\omega,z_0)$ in Eq. (4). Taking into account a nonlinearity we obtain that $\hat{\psi}(\omega,z)$ is the slow function of z on a scale $L \ll z_{nl}$ (see e.g. Refs. [1]-[5]), where $z_{nl} \equiv c/(n_2\omega_0|a|^2)$ is the characteristic nonlinear length, and a is the typical optical pulse amplitude. We solve Eq. (4) by iterations for $0 \le z-z_0 \le L$, where we have a freedom of choice of L with the only condition that $L \ll z_{nl}$ (typically L does not have to be very small, e.g. $L \lesssim 0.1z_{nl}$ is often sufficiently small to reach the high precision, see the error estimates below). For the first iteration we set $\hat{\psi}^{(0)}(\omega,z) = \hat{\psi}(\omega,z_0) = \hat{A}(\omega,z_0)$ and, respectively, $\hat{A}(\omega,z) = \hat{\psi}^{(0)}(\omega,z)\mathrm{e}^{\tilde{\mathcal{D}}_\omega(z)}$ in right hand side (r.h.s.) of Eq. (4) which gives the first iteration $\hat{\psi}^{(1)}(\omega,z)$ for left hand side (l.h.s.) of Eq. (4). Similar, substitution of $\hat{\psi}^{(n-1)}(\omega,z)$ in r.h.s. of Eq. (4) gives $\hat{\psi}^{(n)}(\omega,z)$ in l.h.s. of Eq. (4) for $n = 1,2,\ldots$

*Electronic address: plushnik@math.unm.edu

Assume that $A(\omega, z_0)$ is given from the previous step and that the interval $z_0 \le z \le z_0 + L$ includes $M+1$ equally spaced points $z_0,\ z_1, \dots\ z_M = z_0 + L$. We solve Eq. (4) numerically by iterations at the large step $z_0 \le z \le z_0 + L$ subdivided to the small elementary step $\Delta z \equiv L/M$ along z. We perform a total number of iterations n_{tot} to approximate $\hat{\psi}(\omega, z_0 + L)$ as $\hat{\psi}^{(n_{tot})}(\omega, z_0 + L)$ with the desired precision, see the error estimates below. We assume that a distributed memory cluster with M nodes is used then jth node keeps/processes the data for the point z_j with $j = 1, 2, \dots, M$ (for the computers with shared memory there is obviously more freedom where to keep these data which can be e.g. optimized for the most efficient use of the supercomputer cache). MPA is performed on each of n_{tot} iterations as follows:

1. $\hat{A}(\omega, z_0) = \hat{\psi}(\omega, z_0)$, copy the values of $\hat{\psi}(\omega, z_0)$ into $\hat{\psi}(\omega, z)$ at all z, i.e. create M copies of $\hat{\psi}(\omega, z_0)$.

2. Find $\hat{A}(\omega, z) = \hat{\psi}(\omega, z)\mathrm{e}^{\hat{\mathcal{D}}_\omega(z)}$ at all z.

3. In order to return to t-domain, calculate independent Fourier transforms $A(t, z) = \hat{F}^{-1}[\hat{A}(\omega, z)]$ at all z.

4. Calculate values of $U_p(t, z) \equiv (1 - f_R) \sum\limits_{l,m,n} S^k_{plmn} A_l(t, z) A_m(t, z) A^*_n(t, z)$ for all z and p. In a similar way, calculate values of $V_{pl}(t, z) \equiv \sum\limits_{m,n} f_R S^R_{plmn} A_m(\omega, z) A^*_n(\omega, z)$ for all z, l and p.

5. Use values from the previous step to calculate independent Fourier transforms $\hat{U}_p(\omega, z) \equiv \widehat{F}_\omega\left[U_p(t, z)\right]$ for all z and p. In a similar way, calculate independent Fourier transforms $\hat{V}_{pl}(\omega, z) \equiv \widehat{F}_\omega\left[V_{pl}(t, z)\right]$ for all z, l and p.

6. Calculate the convolution (5) by the multiplication $\hat{h}_\omega \widehat{V}_{pl}$ in Fourier space and inverse FT as $[h * V_{pl}](t, z) = \widehat{F}^{-1}\left[\hat{h}_\omega \widehat{V}_{pl}\right]$ for all z, l and p.

7. Perform the remaining summation over l of the result of the previous step to find $Q_p(t, z) \equiv \sum\limits_l A_l(t, z)[h * V_{pl}](t, z)$ for all z and p.

8. Calculate independent Fourier transforms $\hat{U}_p(\omega, z) + \hat{Q}_p(\omega, z) = \hat{F}_\omega[U_p(t, z) + Q_p(t, z)]$ at all z.

9. Numerical integration (summation) by trapezoidal rule of the integral in (4) using $\hat{U}_p(\omega, z) + \hat{Q}_p(\omega, z)$ from previous step, i.e. use the integrand $\mathrm{i}\frac{n_2\omega_0}{c}\left(1 + \frac{\omega}{\omega_0}\right)[\hat{U}_p(\omega, z) + \hat{Q}_p(\omega, z)]\mathrm{e}^{-\hat{\mathcal{D}}_\omega(z)}$ for the numerical summation over z. Save intermediate results of integration at every z.

10. For the second, third etc. iterations go to step 2. Repeat that until finishing n_{tot}th iteration.

11. Reconstruct $\hat{A}_p(\omega, z_M)$ on the right edge of the interval from $\hat{\psi}_p(\omega, z_M)$ to complete the integration over distance L.

MPA is schematically shown in Fig. 1. It is assumed that there are N points for the discretization in ω (and t). Each vertical bar represent a required memory for storing $\propto PN$ complex values of $A_p(t, z_j)$ and different intermediate complex values like $U_p(t, z)$. Also storing $V_{pl}(t, z)$ needs a memory $\propto P^2N$ complex values. But if the available memory is the issue then the summation over l can be done to use only $\propto PN$ memory. Respectively, taking into account all vertical bars $j = 0, \dots, M-1$ suggests that the total required memory is either $\propto P^2NM$ or $\propto PNM$ (depending on the summation over l). Most steps of MPA are computed in parallel without communications between different vertical bars. It is then natural to assume that each vertical bar corresponds to the computer node (if all P modes cannot fit into the memory of the single node then of course the additional memory allocation between different nodes would have to be done). The communications between different nodes occur at step 1 when M identical copies of ψ have to be created as well as at step 9 where the summation over z is performed (that summation is shown schematically by dashed horizontal line in Fig. 1). That summation can be also done with the additional parallelization through a hierarchy of pairwise summation between different vertical bars as suggested in Ref. [4].

Note that many of steps of MPA algorithm (e.g. steps 4-8) are the same/similar to the standard Split-Step or Runge-Kutta methods. So the existing code can be used for these steps except it has to be executed simultaneously on multiple nodes corresponding to different values of z_j.

In simulations we successively perform many large steps of length L. E.g. if we use the same L at each large step, then we use $\hat{\psi}(\omega, z_0)$ with $z_0 = mL$ for a given $m = 0, 1, \dots$ to perform a total number of iterations n_{tot} to approximate $\hat{\psi}(\omega, z_0 + L)$ as $\hat{\psi}^{(n_{tot})}(\omega, z_0 + L)$. Then we use that approximate value as starter for the next spatial interval by setting $z_0 = (m+1)L$ and proceeding in a similar way. However, if the spectrum of the solution widens/shrinks during the current L step then the next large step can be performed with the updated value of L.

Numerical errors of MPA. The accuracy of MPA was found in Refs. [4, 5]. In short, there are two sources of the numerical error. First source is the elementary step $\Delta z \equiv L/M$ along z. The corresponding error is the same as the error $r_1 \propto (\Delta z)^2$ of the M steps of the second order Split-Step method with the elementary step Δz (however, contrary to MPA, Split-step method cannot be parallelized along z). The second error $r_2 \propto (L/z_{nl})^{n_{tot}+1}$ comes from the large step L while being independent of M. Respectively, to keep that error below the desired tolerance one has either choose L small enough or increase the number of iterations n_{tot}. E.g., if the relative error 10^{-5} is desired then one can choose $L/z_{nl} \sim 0.1$ and $n_{tot} = 4$. For practical calculations the width of the spectrum N and the available

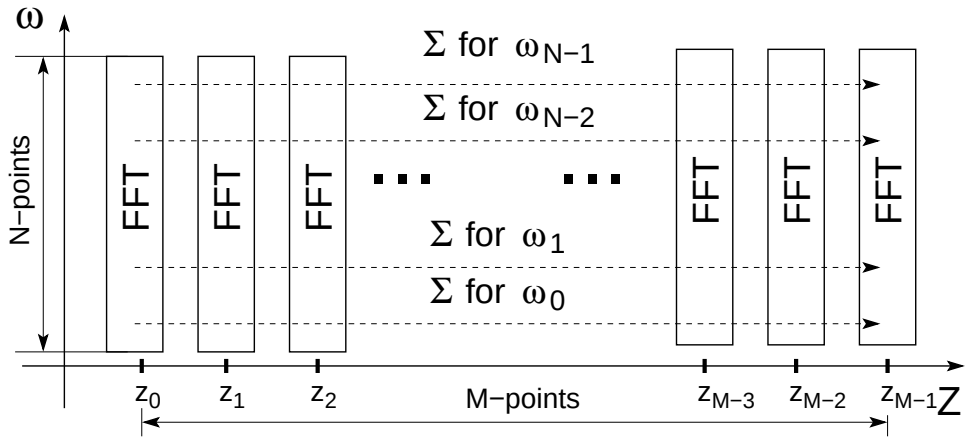

FIG. 1: Schematic representation of MPA for N points in ω and M parallel points in z direction.

computational resources (i.e. M) often force $L/z_{nl} \ll 1$, then 1-2 iterations of MPA are often sufficient to match the error of Split-Step method $\propto (\Delta z)^2$. A more accurate estimate of both errors is given by the relative difference of the exact solution $A_{exact}(L)$ at the interval L and its approximation $A^{(n_{tot})}(L)$ by n_{tot} iterations of MPA as follows [4, 5]

$$\left|\frac{A_{exact}(L) - A^{(n_{tot})}(L)}{A_{exact}(L)}\right| = r_1 + r_2 \sim \frac{L(\Delta z)^2}{z_{\mathcal{D}}^2 z_{nl}} + \left(\frac{L}{z_{nl}}\right)^{n_{tot}+1}, \tag{6}$$

were $z_{\mathcal{D}} \equiv |1/\hat{\mathcal{D}}_{\omega_w}|$ is the dispersion with ω_w being the typical width of the spectrum (e.g. FWHM of the spectrum). It is assumed in the derivation of the error (6) that $z_{\mathcal{D}} \lesssim z_{nl}$.

The recommended choice of Δz. The condition of the absence of weak (algebraic) instabilities in GMMNLSE (occurs due to the frequency discretization) requires the highest Fourier harmonic to change in phase for less than π during one elementary step Δz. More precisely, if the simulated time interval is $-T/2 \le t \le T/2$ then the frequency span is $-\omega_{max} \le \omega \le \omega_{max}$, $\omega_{max} = (N/2)(2\pi/T)$. It implies the condition

$$\Delta z < \pi/|\hat{\mathcal{D}}_{\omega_{max}}|. \tag{7}$$

Because the corresponding instability is weak (only algebraic), the condition might be sometimes ignored (7) but still has to be kept in mind.

Adaptive stepping in z. If the spectrum significantly changes (widens) at the current long step of length L then one can increase the number of harmonics in ω by the spectral interpolation (i.e. by adding Fourier harmonics of zero amplitudes for lager $|\omega|$ beyond the current ω_{max}). Respectively, the elementary step in z has to be decreased according to the condition (7). The good practise is to keep the corresponding ratio

$$\epsilon \equiv \frac{\Delta z|\hat{\mathcal{D}}_{\omega_{max}}|}{\pi}. \tag{8}$$

constant (and below one) while adjusting ω_{max} and L during the simulation/propagation of pulse. In practise ω_{max} is often large which forces Δz to be very small to satisfy the condition (7). Then ϵ in (8) can be chosen very close to 1 because the estimate (6) ensures the smallness of the error due to the discretization in z. Note that the condition (7) is the analog of CFL condition for the numerical stepping in GMMNLSE.

The trapezoidal rule of the integration. Consider the step 9 of MPA and denote $\hat{R}_j^{(n)} \equiv \mathrm{i}\frac{n_2\omega_0}{c}\left(1+\frac{\omega}{\omega_0}\right)[\hat{U}_p^{(n)}(\omega, z_j) + \hat{Q}_p^{(n)}(\omega, z_j)]\mathrm{e}^{-\tilde{\mathcal{D}}_\omega(z)}$. Here $\hat{U}_p^{(n)}$ and $\hat{Q}_p^{(n)}$ correspond to quantities $\hat{U}_p$ and $\hat{Q}_p$ defined at steps 4-8 with the superscript n designating the number of iteration. Then the (composite) trapezoidal rule of the numerical integration at each of N discrete values of ω is given by

$$\hat{\psi}_{p,q}^{(n+1)}(\omega) = \hat{\psi}_{p,0}^{(n)}(\omega) + \frac{\hat{R}_0^{(n)}}{2} + \sum_{j=1}^{l-1}\hat{R}_j^{(n)} + \frac{\hat{R}_q^{(n)}}{2}, \tag{9}$$

where $\hat{\psi}_{p,q}^{(n+1)}(\omega) \equiv \hat{\psi}_{p}^{(n+1)}(\omega, z_q)$ and $q = 1, 2, \ldots, M,\ z_j = z_0 + j\Delta z, \quad j = 0, 1, \ldots, M.$

---

[1] P. M. Lushnikov, Opt. Lett. **26**, 1535 (2001).
[2] I. R. Gabitov and P. M. Lushnikov, Opt. Lett. **27**, 113 (2002).
[3] I. R. Gabitov, R. Indik, P. M. Lushnikov, L. Mollenauer, and M. Shkarayev, Opt. Lett. **32**, 605 (2007).
[4] P. M. Lushnikov, Opt. Lett. **27**, 939 (2002).
[5] A.O. Korotkevich, and P.M. Lushnikov, Optics Letters, **36**, 1851-1853 (2011).