

New Version Announcement

OpenMP Fortran programs for rotating dipolar Bose-Einstein condensates

Denis Mujo^a, Dušan Vudragović^a, Paulsamy Muruganandam^{b,c}, Sadhan K. Adhikari^{d,*}^a Scientific Computing Laboratory, Center for the Study of Complex Systems, Institute of Physics Belgrade, University of Belgrade, Pregrevica 118, Belgrade, 11080, Serbia^b Department of Physics, Bharathidasan University, Palkalaiperur Campus, Tiruchirappalli, Tamilnadu, 620024, India^c Department of Medical Physics, Bharathidasan University, Palkalaiperur Campus, Tiruchirappalli, Tamilnadu, 620024, India^d Instituto de Física Teórica, UNESP – Universidade Estadual Paulista, 01.140-70 São Paulo, São Paulo, Brazil

ARTICLE INFO

The review of this paper was arranged by Prof. Dr. J Ballantyne
Handling Editor: MS Monishm

Keywords:

Dipolar Bose-Einstein condensate
Contact and dipolar interaction
Gross-Pitaevskii equation
Split-step Crank-Nicolson scheme
Fortran programs
Partial differential equation

ABSTRACT

In this paper we present Open Multi-Processing (OpenMP) Fortran 90/95 programs to solve the Gross-Pitaevskii equation for a rotating dipolar Bose-Einstein condensate (BEC) in two and three dimensions, which is a new version of our previous published programs for a dipolar Bose-Einstein condensate without rotation. After the recent experimental study of a rotating dipolar BEC [L. Klaus et al., Nature Phys. 18, 1453 (2022)], the present programs will be useful tools for related theoretical investigation. The algorithm used is the split-step semi-implicit Crank-Nicolson scheme for imaginary- and real-time propagation to obtain stationary states and BEC dynamics, respectively, as in the previous version [L. E. Young-S. et al., Comput. Phys. Commun. 286 (2023) 108669].

New version program summary

Program title: DBEC-ROT-PROGRAM, a program package containing programs imre3d-th.f90, imre2dXY-th.f90, with fftw3.f03 and fftw3.mod.

CPC Library link to program files: <https://doi.org/10.17632/sp3wvbtmnh.2>

Licensing provisions: Apache License 2.0

Programming language: Open Multi-Processing (OpenMP) Fortran 90/95. The program is tested with the GNU, Intel, and Oracle (former Sun) compilers.

Journal Reference of previous version: Comput. Phys. Commun. 286 (2023) 108669, <https://doi.org/10.1016/j.cpc.2023.108669>

Does the new version supersede the previous version?: Yes

Nature of problem: The present OpenMP Fortran 90/95 programs solve the time-dependent nonlinear partial differential Gross-Pitaevskii (GP) equation for a trapped rotating dipolar Bose-Einstein condensate (BEC) in two (2D), and three (3D) spatial dimensions. In 3D the provision for including the higher-order repulsive Lee-Huang-Yang interaction is accommodated.

Solution method: We employ the split-step Crank-Nicolson scheme to discretize the time-dependent GP equation in space and time. The discretized equation is then solved by imaginary- or real-time propagation, employing adequately small space and time steps, to yield the solution of stationary and non-stationary problems, respectively.

Reason for new version: Previously published Fortran [1] and C [2] programs and their OpenMP and Cuda versions [3,4] for solving the time-dependent GP equation have become useful tools in the study of statics and dynamics of a BEC. These programs have been extended to the more complex scenario of dipolar atoms [5–7], and rotating condensates [8]. The available Fortran 90/95 and C programs [1,2] for the solution of the GP equation for a BEC are enjoying widespread use in phenomenological application. The same for the solution of the GP equation of a rotating BEC have played a fundamental role in the theoretical study of the formation of vortices in a rotating BEC. Although, the experimental realization of a rotating nondipolar BEC in a controlled fashion dates back to 1999 [9], the same of a rotating dipolar BEC has been possible only recently in 2022 [10]. In view of this experimental breakthrough, the present OpenMP Fortran 90/95 programs for the solution of the time-dependent GP equation for a rotating dipolar BEC are of paramount interest.

* Corresponding author.

E-mail addresses: denis.mujo@ipb.ac.rs (D. Mujo), dusan.vudragovic@ipb.ac.rs (D. Vudragović), anand@bdu.ac.in (P. Muruganandam), sk.adhikari@unesp.br (S.K. Adhikari).

<https://doi.org/10.1016/j.cpc.2026.110356>

Received 16 July 2026; Accepted 10 August 2026

Available online 12 August 2026

0010-4655/© 2026 Elsevier B.V. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

Summary of revisions: The program package DBEC-ROT-PROGRAM contains the 3D program `imre3d-th.f90` and 2D program `imre2dXY-th.f90` in the directory `src`, as well as the files `makefile`, `makefile-mur`, `readme.txt`, and `readme-fftw.txt`. The directory output contains examples of matching outputs of imaginary- and real-time propagation programs in sub-directories with a generic name, e.g. `imag3d`, `real2dXY`, etc. These results are not fully converged numerically. The reader is requested to read the file `readme.txt`, which gives instruction to compile and run the programs by the command `make` and also to install the fast Fourier transformation (FFT) routine in Linux/Unix operating systems. The reader is advised to consult Ref. [5] for details. The 3D and 2D programs are OpenMP versions leading to a significant reduction in execution time in a multicore processor. The `makefile` allows an automated compilation of the programs using different supported compilers (gfortran GNU, ifx Intel, macOS) by a simple `make` command, as in Ref. [5]. (However, the 1D and 2D dipolar programs in previous publications [5–7] erroneously used the 3D nonlinearities, in place of corresponding 1D and 2D nonlinearities, in numerical calculations. The corresponding arXiv archive versions do not have these errors.) All input parameters are placed in the beginning of each program in MODULEs `GPE_DATA` and `COMM_DATA`. Most of these input parameters are the same as in Ref. [11]. The new parameter `XDIPOLAR` (= 0) bypasses the dipolar part to give the result for a nondipolar condensate. As in Ref. [8], the parameter `FUNCTION` selects the type of initial Gaussian function (with or without a vortex), and the parameter `RANDOM` (= 1) includes an arbitrary phase on the initial Gaussian function to facilitate the creation of vortices. The parameter `LHY` (= 1) allows the possibility to include the Lee-Huang-Yang interaction in 3D. The imaginary-time programs implement the calculation starting from an initial analytic Gaussian wave function using a non-zero value of the parameter `NSTP`, whereas the real-time programs use the converged solution of the imaginary-time programs as the initial state of calculation employing `NSTP` = 0. The FFT algorithm works faster and more efficiently when we take the number of space grid points along x , y and z directions – NX , NY in 2D and NX , NY , NZ in 3D in MODULE `COMM_DATA` – in powers of 2, e.g. 2^n with n an integer, and should be so chosen for reducing the execution time. The operational scheme for running the codes is identical to that in Ref. [5].

CRediT authorship contribution statement

Denis Mujo: Software, Validation, Visualization, Writing – review & editing; **Dušan Vudragović:** Investigation, Software, Validation, Writing – review & editing; **Paulsamy Muruganandam:** Investigation, Software, Visualization, Writing – review & editing; **Sadhan K. Adhikari:** Conceptualization, Formal analysis, Investigation, Software, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

D. M. and D. V. acknowledge support provided by the Institute of Physics Belgrade, through the grant by the Ministry of Science, Technological Development, and Innovations of the Republic of Serbia. The work of P. M. is supported by the Ministry of Education-Rashtriya Uchchatar Shiksha Abhiyan (MoE RUSA 2.0): Bharathidasan University– Physical Sciences. S. K. A. acknowledges support by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (National Council for Scientific and Technological Development, Brazil) grant 303885/2024-6. Numerical simulations were run on the PARADOX supercomputing facility at the Scientific Computing Laboratory, National Center of Excellence for the Study of Complex Systems, Institute of Physics Belgrade.

Data availability

No data was used for the research described in the article.

Supplementary material

Supplementary material associated with this article can be found in the online version at [10.1016/j.cpc.2026.110356](https://doi.org/10.1016/j.cpc.2026.110356).

References

- [1] P. Muruganandam, S.K. Adhikari, Fortran programs for the time-dependent Gross-Pitaevskii equation in a fully anisotropic trap, *Comput. Phys. Commun.* 180 (2009) 1888. [arXiv:0904.3131](https://arxiv.org/abs/0904.3131)
- [2] D. Vudragović, I. Vidanović, A. Balaž, P. Muruganandam, S.K. Adhikari, C programs for solving the time-dependent gross-Pitaevskii equation in a fully anisotropic trap, *Comput. Phys. Commun.* 183 (2012) 2021. [arXiv:1206.1361](https://arxiv.org/abs/1206.1361)
- [3] L.E. Young-S., P. Muruganandam, S.K. Adhikari, V. Loncar, D. Vudragović, Antun Balaž, OpenMP GNU and intel Fortran programs for solving the time-dependent Gross-Pitaevskii equation, *Comput. Phys. Commun.* 220 (2017) 503. [arXiv:1709.04423](https://arxiv.org/abs/1709.04423)
- [4] B. Satrić, V. Slavnić, A. Belić, A. Balža, P. Muruganandam, S.K. Adhikari, Hybrid OpenMP/MPI programs for solving the time-dependent Gross-Pitaevskii equation in a fully anisotropic trap, *Comput. Phys. Commun.* 200 (2016) 411. [arXiv:1601.04641](https://arxiv.org/abs/1601.04641)
- [5] R. Kishor Kumar, L.E. Young-S., D. Vudragović, A. Balaž, P. Muruganandam, S.K. Adhikari, Fortran and C programs for the time-dependent dipolar Gross-Pitaevskii equation in an anisotropic trap, *Comput. Phys. Commun.* 195 (2015) 117. [arXiv:1506.03283](https://arxiv.org/abs/1506.03283)
- [6] V. Lončar, L.E. Young-S., S. Škrbić, P. Muruganandam, S.K. Adhikari, A. Balaž, OpenMP, OpenMP/MPI, and CUDA/MPI C programs for solving the time-dependent dipolar Gross-Pitaevskii equation, *Comput. Phys. Commun.* 209 (2016) 190. [arXiv:1610.05329](https://arxiv.org/abs/1610.05329)
- [7] V. Lončar, A. Balaž, A. Bogojević, S. Škrbić, P. Muruganandam, S.K. Adhikari, CUDA programs for solving the time-dependent dipolar Gross-Pitaevskii equation in an anisotropic trap, *Comput. Phys. Commun.* 200 (2016) 406. [arXiv:1601.04640](https://arxiv.org/abs/1601.04640)
- [8] R.K. Kumar, V. Lončar, P. Muruganandam, S.K. Adhikari, A. Balaž, Fortran OpenMP programs for rotating Bose-Einstein condensates, *Comput. Phys. Commun.* 240 (2019) 74. [arXiv:1906.06327](https://arxiv.org/abs/1906.06327)
- [9] K.W. Madison, F. Chevy, W. Wohlleben, J. Dalibard, Vortex formation in a stirred Bose-Einstein condensate, *Phys. Rev. Lett.* 84 (1999) 806. [arXiv:cond-mat/9912015](https://arxiv.org/abs/cond-mat/9912015)
- [10] L. Klaus, T. Bland, E. Poli, C. Politi, G. Lamporesi, E. Casotti, R.N. Bisset, M.J. Mark, F. Ferlaino, Observation of vortices and vortex stripes in a dipolar condensate, *Nat. Phys.* 18 (2022) 1453. [arXiv:2206.12265](https://arxiv.org/abs/2206.12265)
- [11] L.E. Young-S., P. Muruganandam, A. Balaž, S.K. Adhikari, OpenMP Fortran programs for solving the time-dependent dipolar Gross-Pitaevskii equation, *Comput. Phys. Commun.* 286 (2023) 108669. [arXiv:2301.09383](https://arxiv.org/abs/2301.09383)

OpenMP Fortran programs for rotating dipolar Bose-Einstein condensates

Denis Mujo, Dušan Vudragović, Paulsamy Muruganandam, Sadhan K. Adhikari

Below we are giving, not deriving, the equations which are being solved. The derivation can be found in Refs. [1] and [2]. We also present a set of results in pictorial form.

The 3D GP equation

In 3D, we solve the following GP equation, in dimensionless units, including the Lee-Huang-Yang (LHY) interaction

$$i\frac{\partial\phi(\mathbf{r},t)}{\partial t} = \left[-\frac{1}{2}\nabla^2 + \frac{1}{2}(\gamma^2 x^2 + \nu^2 y^2 + \lambda^2 z^2) + 4\pi aN|\phi(\mathbf{r},t)|^2 + 3a_{dd}N \int U_{dd}(\mathbf{R})|\phi(\mathbf{r}',t)|^2 d\mathbf{r}' + \gamma_{\text{LHY}} N^{3/2} |\phi(\mathbf{r},t)|^3 - \Omega l_z \right] \phi(\mathbf{r},t), \quad (1)$$

$$\gamma_{\text{LHY}} = \frac{128}{3} \sqrt{\pi a^5} \times \frac{(3\epsilon_{dd})^{5/2}}{48} \Re \left[(8 + 26\kappa + 33\kappa^3) \sqrt{1 + \kappa} + 15\kappa^3 \ln \left(\frac{1 + \sqrt{1 + \kappa}}{\sqrt{\kappa}} \right) \right], \quad \kappa = \frac{1 - \epsilon_{dd}}{3\epsilon_{dd}} \quad (2)$$

$$U_{dd}(\mathbf{R}) = \frac{1 - 3 \cos^2 \theta}{|\mathbf{R}|^3}, \quad \epsilon_{dd} = \frac{a_{dd}}{a}, \quad \mathbf{R} = \mathbf{r} - \mathbf{r}', \quad (3)$$

where $\Re$ denotes the real part, γ, ν, λ are the harmonic trap parameters along x, y, z directions, respectively, $l_z \equiv i(y\partial/\partial x - x\partial/\partial y)$ the z component of angular momentum, Ω the angular frequency of rotation, a is the scattering length, a_{dd} the dipolar length, N the number of atoms, $\mathbf{r} \equiv \{x, y, z\}$ and $\mathbf{r}' \equiv \{x', y', z'\}$ are the positions of two atoms, θ is the angle made by the vector $\mathbf{R}$ with the polarization z direction of dipolar atoms.

The quasi-2D GP equation

In 2D, we solve the following GP equation, in dimensionless units,

$$i\frac{\partial\phi_{2D}(\boldsymbol{\rho},t)}{\partial t} = \left[-\frac{1}{2}\nabla_{\boldsymbol{\rho}}^2 + \frac{1}{2}(\gamma^2 x^2 + \nu^2 y^2) + \frac{4\pi aN|\phi_{2D}(\boldsymbol{\rho},t)|^2}{\sqrt{2\pi}d_z} + \frac{a_{dd}N}{\pi} \int d\mathbf{k}_p \exp(-i\mathbf{k}_p \cdot \boldsymbol{\rho}) \tilde{n}(\mathbf{k}_p, t) h_{2D} \left(\frac{k_p d_z}{\sqrt{2}} \right) - \Omega l_z \right] \phi_{2D}(\boldsymbol{\rho},t), \quad (4)$$

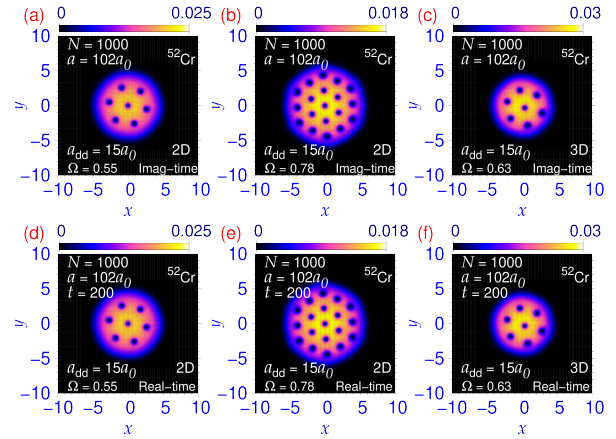


Figure 1: Contour plot of density of a dipolar BEC of $N = 1000$ ^{52}Cr atoms for (a) $\Omega = 0.55$ in a quasi-2D trap (b) $\Omega = 0.78$ in a quasi-2D trap, (c) $\Omega = 0.63$ in a 3D trap obtained by imaginary-time propagation. In plots (d)-(f) we show the densities obtained in real-time propagation corresponding to the states of plots (a)-(c), respectively, at times $t = 200, 200$, and 30 where we use the converged imaginary-time wave functions as the initial states. In 3D, for larger t , dynamical instability sets in and the vortex lattice is destroyed quickly, which is not due to numerical trouble.

where $d_z = 1/\sqrt{\lambda}$, $\boldsymbol{\rho} \equiv \{x, y\}$, $\mathbf{k} \equiv \{k_x, k_y\}$, $k_p = \sqrt{k_x^2 + k_y^2}$, and

$$\tilde{n}(\mathbf{k}_p, t) = \int d\boldsymbol{\rho} \exp(i\mathbf{k}_p \cdot \boldsymbol{\rho}) |\phi_{2D}(\boldsymbol{\rho}, t)|^2, \quad (5)$$

$$\tilde{n}(k_z) = \int dz \exp(ik_z z) |\phi(z)|^2 = e^{-k_z^2 d_z^2/4}, \quad (6)$$

$$h_{2D}(\xi) = \frac{1}{2\pi} \int_{-\infty}^{\infty} dk_z \left[\frac{3k_z^2}{k^2} - 1 \right] |\tilde{n}(k_z)|^2 \quad (7)$$

$$= \frac{1}{\sqrt{2\pi}d_z} \left[2 - 3\sqrt{\pi}\xi \exp(\xi^2) \{1 - \text{erf}(\xi)\} \right], \quad \xi = \frac{k_p d_z}{\sqrt{2}}. \quad (8)$$

Numerical Results

In numerical demonstration of the present programs (imre2dXY.f90 and imre3D.f90), we consider a dipolar BEC of ^{52}Cr atoms in a quasi-two-dimensional (quasi-2D) trap ($\gamma = \lambda = 1, \lambda = 100$), which facilitates the formation of vor-

tices in the x - y plane. The parameters in the present numerical study are $a = 102a_0/l$, $a_{dd} = 15a_0/l$, a_0 is the Bohr radius, $l = 1 \mu\text{m}$ is the length scale, $d_z = 0.1$, $N = 1000$, $\gamma_{\text{LHY}} = 0$. First, we solve the quasi-2D equation (4) by imaginary-time propagation with $\Omega = 0.55$ and number of space grids in the x, y directions $NX=NY=256$, which is adequate to accommodate the condensate with vortices in the x - y plane. The initial state in imaginary-time propagation was taken as a Gaussian function with one vortex and random phase imprinted on it. The contour plot of the resultant density is presented in Fig. 1(a), which shows seven vortices on a triangular lattice. Now we consider the same rotating condensate with an increased angular frequency of rotation $\Omega = 0.78$. In imaginary-time propagation the initial state is the final converged wave function of Fig. 1(a); the space mat is now of size $NX=NY=384$. In Fig. 1(b), we plot the resultant density. With the increased angular frequency of rotation, the condensate now generates nineteen vortices arranged on a triangular lattice.

Next we consider a numerical solution of the 3D equation (1) by imaginary-time propagation with $\Omega = 0.63$ and with the number of space grids in x, y, z directions $NX=NY=384$ and $NZ=32$. In this case the initial state in imaginary-time propagation is taken as a Gaussian function with one vortex and random phase imprinted on it. The contour plot of the z -integrated resultant density $n(x, y) = \int dx dy |\psi(x, y, z)|^2$ is displayed in Fig. 1(c), which shows seven vortices on a triangular lattice. The supplied programs – `imre2dXY.f90` and `imre3d.f90` –, with the included correct parameters, will generate the condensates of Figs. 1(a) and (c). This will allow the user to familiarize with the programs.

The same programs `imre2dXY.f90` and `imre3d.f90` also provide results of real-time propagation when so selected through the parameters and DT in the beginning of the programs. If the final converged wave functions of the condensates of Figs. 1(a)-(c) are used as the initial functions in real-time simulation the stable vortex lattice pattern continues to exist. The final densities obtained in real-time propagation corresponding to the condensates of Figs. 1(a)-(c) are shown in Figs. 1(d)-(f), respectively.

References

- [1] R. Kishor Kumar, L.E. Young-S., D. Vudragović, A. Balaž, P. Muruganandam, S.K. Adhikari, Fortran and C programs for the time-dependent dipolar Gross-Pitaevskii equation in an anisotropic trap, *Comput. Phys. Commun.* 195 (2015) 117; arXiv:1506.03283.
- [2] R. K. Kumar, V. Lončar, P. Muruganandam, S. K. Adhikari, A. Balaž, C and Fortran OpenMP programs for rotating Bose–Einstein condensates, *Comput. Phys. Commun.* 240 (2019) 74; arXiv:1906.06327.