

A NEW PRECONDITIONED NONLINEAR CONJUGATE GRADIENT METHOD IN REAL ARITHMETIC FOR COMPUTING THE GROUND STATES OF ROTATIONAL BOSE–EINSTEIN CONDENSATE*

TIANQI ZHANG[†] AND FEI XUE[†]

Abstract. We propose a new nonlinear preconditioned conjugate gradient (PCG) method in real arithmetic for computing the ground states of rotational Bose–Einstein condensate, modeled by the Gross–Pitaevskii equation. Our algorithm presents a few improvements of the PCG method in complex arithmetic studied by Antoine, Levitt, and Tang [*J. Comput. Phys.*, 343 (2017), pp. 92–109]. We show that the special structure of the energy functional $E(\phi)$ and its gradient with respect to ϕ can be fully exploited in real arithmetic to evaluate them more efficiently. We propose a simple approach for fast evaluation of the energy functional, which enables exact line search. Most importantly, we derive the discrete Hessian operator of the energy functional and propose a shifted Hessian preconditioner for PCG, with which the ideal preconditioned Hessian has favorable eigenvalue distributions independent of the mesh size. This suggests that PCG with our ideal Hessian preconditioner is expected to exhibit mesh size-independent asymptomatic convergence behavior. In practice, our preconditioner is constructed by incomplete Cholesky factorization of the shifted discrete Hessian operator based on high-order finite difference discretizations. Numerical experiments in two-dimensional (2D) and three-dimensional (3D) domains show the efficiency of fast energy evaluation, the robustness of exact line search, and the improved convergence of PCG with our new preconditioner in iteration counts and runtime, notably for more challenging rotational BEC problems with high nonlinearity and rotational speed.

Key words. PCG, Bose–Einstein condensation, ground states, Hessian preconditioner, exact line search, real arithmetic

MSC codes. 65H17, 65K10, 65F08, 65F15, 35Q90, 35P30

DOI. 10.1137/23M1590317

1. Introduction. The Bose–Einstein condensate (BEC) is referred to as the fifth state of matter, which was first predicted theoretically by Bose and Einstein, before being realized experimentally in 1995 [4, 16, 21, 24]. The literature on BECs has grown rapidly over the last two decades in atomic, molecular, optics, condensed matter physics, and quantum computing; see, e.g., [17, 29, 30, 31, 36] and references therein. In this rapidly growing research area, numerical simulation has been playing an important role in understanding the theories and the experiments. At temperatures T which are much lower than the critical temperature T_c , the macroscopic behavior of a BEC can be well described by a condensate wave function ϕ which is the solution to a Gross–Pitaevskii equation (GPE) [8]. It is very useful to obtain numerical solutions of such a class of equations efficiently. Calculations of stationary states, i.e., ground/excited states, and of the real-time dynamics are the most crucial problems [5, 7, 11, 28, 35]. Numerical methods for approximating the ground states are fundamental to explore the nucleation of vortices, the properties of dipolar gases, bright

*Submitted to the journal’s Numerical Algorithms for Scientific Computing section July 31, 2023; accepted for publication (in revised form) March 6, 2024; published electronically May 23, 2024.

<https://doi.org/10.1137/23M1590317>

Funding: This work was supported in part by the National Science Foundation grant DMS-2111496.

[†]School of Mathematical and Statistical Sciences, Clemson University, Clemson, SC 29634 USA (tianqiz@clemson.edu, fxue@clemson.edu).

beams of coherent matter waves, by studying rotational, dipolar, multicomponent, and spinor BECs.

In this paper, we consider a BEC that can be modeled by the rotational (dimensionless) GPE. In this setting, the computation of a ground state of a d -dimensional BEC takes the form of a constrained minimization problem:

$$(1.1) \quad \text{Find } \phi \in L^2(\mathbb{R}^d) \quad \text{s.t.} \quad \phi \in \arg \min_{\|\phi\|=1} E(\phi)$$

where $\phi \in \mathbb{C}^n$, $\|\phi\| = (\int_{\mathbb{R}^d} |\phi|^2)^{\frac{1}{2}}$ is the standard L^2 -norm and $E(\phi)$ is the associated energy functional defined as

$$(1.2) \quad E(\phi) = \int_{\mathbb{R}^d} \left[\frac{1}{2} |\nabla \phi|^2 + V(\mathbf{x}) |\phi|^2 + \frac{\eta}{2} |\phi|^4 - \Omega \phi^* L_z \phi \right].$$

Here, V is an external potential, η is the nonlinearity strength, Ω is the rotational speed, and $L_z = i(y\partial_x - x\partial_y)$ is the angular momentum operator.

In the literature of numerical solutions to partial differential equations (PDEs), a family of classical methods for computing the ground states of BECs, or similarly the steady-state solution to the Allen–Cahn equation, is based on gradient flows $u_t = -\frac{\partial}{\partial u}(E(u))$ that define a steepest descent curve $u(t)$ of the energy $E(u)$. For example, (1.1) can be solved by the gradient flow with discrete normalization (GFDN, also called ‘imaginary time’ methods in physics) method [2, 6, 7, 11, 12, 15, 19, 20, 49]. These algorithms have been extensively studied, most well-known and widely used for years across disciplines, with mature theoretical support. However, they require solution to a large system of linear equations at each time step, which is usually time-consuming, particularly in three-dimensional (3D) domains with a small mesh size. In addition, these gradient flow-based methods tend to converge slowly with the progress of time steps, since they belong to the class of steepest descent methods that are notably not efficient for numerical optimization. Other methods have been developed, based on numerical solution of the nonlinear eigenvalue problem [25, 47] or on optimization techniques under constraints [14, 18, 22, 23]. In the past few years, new methods have emerged, such as preconditioned conjugate gradient methods [8, 9, 46], the regularized Newton-type method [48] and the regularized density function and accelerated projected gradient (rDF-APG) method [13], which seem successful but falls short of real arithmetic computation and problem-dependent preconditioning.

In [8], the state-of-the-art variant of the preconditioned conjugate gradient (PCG) method was proposed to solve the constrained minimization problem (1.1). However, there are several remaining issues to address, and improvements can be proposed. For example, the complex arithmetic naturally used with Fourier pseudo-spectral methods does not fully exploit the special structure of rotational BEC to speed up certain basic linear algebra computations and to guarantee that the computed energy $E(\phi)$ is real. More importantly, as a most significant component of PCG methods, the preconditioners proposed therein did not take the rotational speed Ω into account, and the convergence rate seems to deteriorate considerably for high-speed rotational problems. In fact, the condition number of the preconditioned Hessian is shown to increase with the domain sizes L and h^{-2} where h is the mesh size. In addition, direct energy evaluation of $E(\phi)$ defined in (1.2) seems costly for each step size along the search direction, which entails the use of approximate line search based on quadratic approximations of $E(\phi)$ or backtracking algorithms.

In this paper, we propose an improved PCG method for computing the ground state of rotational BEC (1.1). Our new method makes exclusive use of real arithmetic

to fully exploit the special structure of the problem and realizes fast numerical linear algebra computations. We discuss a simple approach to achieve fast energy evaluations for many different step sizes along the search direction at little additional cost, which enables exact line search. We derive the explicit expression of the discrete Hessian operator of the energy $E(\phi)$ in real arithmetic and propose an approximate shifted Hessian preconditioner that is quite efficient for tackling high nonlinearity strength η and high rotational speed Ω . We show that the preconditioned Hessian with our ideal preconditioner has favorable eigenvalue distributions independent of the mesh size h . Therefore, given a rotational BEC problem in a specified domain, the PCG method with our ideal preconditioner is expected to exhibit mesh size-independent asymptotic convergence behavior. In addition, we construct a scaling-invariant CG method under the BEC problem setting and provide its global convergence towards a critical point of $E(\phi)$.

The remainder of this paper is organized as follows. We provide an introduction to the mathematical problem and the PDE discretization scheme in section 2. In section 3, we present a detailed description of our method. In section 4, we derive the discrete Hessian operator of energy functional $E(\phi)$ and provide the preconditioning strategy in practice. Section 5 provides an accurate and efficient method to enable fast energy evaluation and exact line search. A proof of the global convergence of a scaling-invariant conjugate gradient (CG) method for BEC is provided in section 6. We study the eigenvalue distribution of the preconditioned Hessian with our ideal preconditioner in section 7. Section 8 provides numerical results in 2D and 3D domains to validate our new developments. We conclude this paper in section 9.

2. Problem description and discretization. The function $\phi \in L^2(\mathbb{R}^d)$ must be discretized in order to find a numerical solution of the minimization problem (1.1). Also, the discretization must be accurate enough to resolve fine details of vortexes in the solution. Several discretization schemes have been used to compute the solution to the GPE, including high-order finite difference schemes, finite element schemes with adaptive meshing strategies [22, 23], the standard pseudo-spectral schemes based on Fast Fourier Transforms (FFTs) [6, 7, 8, 12]. In the literature on numerical methods for BEC computations, the Fourier pseudo-spectral method [11] is the most widely adopted discretization.

The constrained minimization problem (1.1) can be written in the discrete form. Generation of an appropriate mesh on a finite domain $U \subseteq \mathbb{R}^d$ and application of a corresponding discretization to the continuous GPE, the ground state of BEC in discrete form is the global minimizer of the energy functional

$$(2.1) \quad E_{\eta,\Omega} = \left[-\frac{1}{2} \phi^* L_p \phi + \phi^* \text{diag}(V) \phi + \frac{\eta}{2} \phi^* \text{diag}(|\phi|^2) \phi - i\Omega \phi^* L_\omega \phi \right] h^d,$$

with $\|\phi\|_{\ell^2}^2 = h^d \phi^* \phi = 1$, which is a discretized version of (1.2). Here, $-L_p$ (symmetric positive definite) is the negative discrete Laplacian operator, $\text{diag}(V)$ and $\text{diag}(|\phi|^2)$ are diagonal matrices with the values of the external trapping potential $V(\mathbf{x})$ and $|\phi(\mathbf{x})|^2$ at the mesh nodes on the diagonal, L_ω (skew symmetric) is the discrete version of the operator $y\partial_x - x\partial_y$, $\eta > 0$ denotes the repulsive particle interaction, and Ω is the angular momentum rotating speed. A direct evaluation of the gradient of the energy being zero leads to the algebraic nonlinear eigenvalue problem

$$(2.2) \quad -\frac{1}{2} L_p \phi + \text{diag}(V) \phi + \eta \text{diag}(|\phi|^2) \phi - i\Omega L_\omega \phi = \lambda_{\eta,\Omega} \phi, \quad \text{with } h^d \phi^* \phi = 1,$$

where the eigenvalue $\lambda_{\eta,\Omega}$ is defined as

$$(2.3) \quad \lambda_{\eta,\Omega} = \left[-\frac{1}{2} \phi^* L_p \phi + \phi^* \text{diag}(V) \phi + \eta \phi^* \text{diag}(|\phi|^2) \phi - i \Omega \phi^* L_\omega \phi \right] h^d.$$

Our aim is to find the global minimizer of (2.1) numerically. Note that the minimizer of $E_{\eta,\Omega}$ is *not* necessarily the eigenvector associated with the lowest eigenvalue of (2.2); see [11].

In this paper, we adopt the Fourier pseudo-spectral discretization scheme, which is described in two dimensions, and its extension to other dimensions is straightforward. The wave function ϕ is truncated to a rectangular domain $[-L_x, L_x] \times [-L_y, L_y]$ with periodic boundary conditions, and discretized with even number of grid points N_x, N_y in the x - and y - directions, respectively. A uniformly sampled grid is introduced: $\mathcal{D}_{N_x, N_y} := \{\mathbf{x}_{k_1, k_2} = (x_{k_1}, y_{k_2})\}_{k_1, k_2 \in \mathcal{I}_{N_x, N_y}}$, with $\mathcal{I}_{N_x, N_y} := \{0, \dots, N_x - 1\} \times \{0, \dots, N_y - 1\}$, $x_{k_1+1} - x_{k_1} = y_{k_2+1} - y_{k_2} = h$, and with mesh size $h = 2L_x/N_x = 2L_y/N_y$. Define the discrete Fourier frequencies (ξ_p, μ_q) , with $\xi_p = p\pi/L_x$, $-N_x/2 \leq p \leq N_x/2 - 1$, and $\mu_q = q\pi/L_y$, $-N_y/2 \leq q \leq N_y/2 - 1$. The pseudo-spectral approximation $\tilde{\phi}$ of the function ϕ in the x - and y -directions are such that

$$\tilde{\phi}(x, y) = \frac{1}{N_x} \sum_{p=-N_x/2}^{N_x/2-1} \tilde{\phi}_p^*(y) e^{i\xi_p(x+L_x)}, \quad \tilde{\phi}(x, y) = \frac{1}{N_y} \sum_{q=-N_y/2}^{N_y/2-1} \tilde{\phi}_q^*(x) e^{i\mu_q(y+L_y)},$$

where $\tilde{\phi}_p^*(y)$ and $\tilde{\phi}_q^*(x)$ are the Fourier coefficients in the x - and y -directions, respectively; that is,

$$\tilde{\phi}_p^*(y) = \sum_{k_1=0}^{N_x-1} \tilde{\phi}(x_{k_1}, y) e^{-i\xi_p(x_{k_1}+L_x)}, \quad \tilde{\phi}_q^*(x) = \sum_{k_2=0}^{N_y-1} \tilde{\phi}(x, y_{k_2}) e^{-i\mu_q(y_{k_2}+L_y)}.$$

In order to evaluate the action of the discrete Laplacian and the angular rotation operators on vectors in (2.1), we also need to apply the following operators to the approximation $\tilde{\phi}$ of ϕ , for $(k_1, k_2) \in \mathcal{I}_{N_x, N_y}$:

$$\begin{aligned} \partial_x^2 \phi(\mathbf{x}_{k_1, k_2}) &\approx \partial_x^2 \tilde{\phi}(x_{k_1}, y_{k_2}) = -\frac{1}{N_x} \sum_{p=-N_x/2}^{N_x/2-1} \xi_p^2 \tilde{\phi}_p^*(y_{k_2}) e^{i\xi_p(x_{k_1}+L_x)}, \\ \partial_y^2 \phi(\mathbf{x}_{k_1, k_2}) &\approx \partial_y^2 \tilde{\phi}(x_{k_1}, y_{k_2}) = -\frac{1}{N_y} \sum_{q=-N_y/2}^{N_y/2-1} \mu_q^2 \tilde{\phi}_q^*(x_{k_1}) e^{i\mu_q(y_{k_2}+L_y)}, \\ x \partial_y \phi(\mathbf{x}_{k_1, k_2}) &\approx x \partial_y \tilde{\phi}(x_{k_1}, y_{k_2}) = \frac{1}{N_y} \sum_{q=-N_y/2}^{N_y/2-1} i x_{k_1} \mu_q \tilde{\phi}_q^*(x_{k_1}) e^{i\mu_q(y_{k_2}+L_y)}, \\ y \partial_x \phi(\mathbf{x}_{k_1, k_2}) &\approx y \partial_x \tilde{\phi}(x_{k_1}, y_{k_2}) = \frac{1}{N_x} \sum_{p=-N_x/2}^{N_x/2-1} i y_{k_2} \xi_p \tilde{\phi}_p^*(y_{k_2}) e^{i\xi_p(x_{k_1}+L_x)}. \end{aligned}$$

Meanwhile, we also introduce the finite difference discretization scheme [34, 44], which is useful in constructing an approximation to the Hessian preconditioner we propose in section 4. With the same uniform mesh grids in Fourier pseudo-spectral discretization scheme, the matrices for the operators are

$$\begin{aligned} L_p &= D_{2,x} \otimes I + I \otimes D_{2,y}, \\ L_\omega &= \text{diag}(y_0, \dots, y_{N_y-1}) \otimes D_x - D_y \otimes \text{diag}(x_0, \dots, x_{N_x-1}), \end{aligned}$$

where D_x , D_y and $D_{2,x}$, $D_{2,y}$ are sparse matrices containing the coefficients of the central finite difference approximations of the first partial derivative and the second partial derivative with respect to x and y , respectively, [19, 34, 44]. Note that, regardless of the discretization scheme, the discrete negative Laplacian operator $-L_p$ and the discrete angular rotation operator L_ω are real symmetric positive definite and real skew-symmetric, respectively.

3. The preconditioned conjugate gradient method in real arithmetic.

To develop an efficient solver for problems involving complex numbers, an important strategy in numerical linear algebra is to fully use real arithmetic whenever appropriate. Since $E_{\eta,\Omega}$ in (2.1) is real even though ϕ is complex, computation in real arithmetic is desired, especially for optimization algorithms where $E_{\eta,\Omega}$ needs to be evaluated many times. To the best of our knowledge, nearly all existing algorithms for computing BEC ground states use complex arithmetic, with an exception in [34] that requires solutions of a long sequence of large linear systems that arise in a special nonlinear inverse iteration to solve the nonlinear eigenvalue problem (2.2).

First, we will reformulate the BEC problem in real arithmetic. To develop new methods in real arithmetic, let $\phi = \phi_r + i\phi_g \in \mathbb{C}^n$, where ϕ_r and ϕ_g are the real and imaginary parts of ϕ , with $\|\phi\|_{\ell^2}^2 = \|\phi_r\|_{\ell^2}^2 + \|\phi_g\|_{\ell^2}^2 = 1$. Define $L_s = -\frac{1}{2}L_p + \text{diag}(V)$ (symmetric positive definite). The energy (2.1) in real arithmetic has the form

$$(3.1) \quad E_{\eta,\Omega} = \left[\phi_r^T L_s \phi_r + \phi_g^T L_s \phi_g + \frac{\eta}{2} (\phi_r^2 + \phi_g^2)^T (\phi_r^2 + \phi_g^2) + 2\Omega \phi_r^T L_\omega \phi_g \right] h^d,$$

with $\|\phi\|_{\ell^2} = h^{d/2} \|\phi\|_2 = 1$. Note that ϕ_r^2 is the column vector whose entries are the squares of those of ϕ_r , and ϕ_g^2 is defined similarly. The evaluation of (3.1) takes only half of the arithmetic cost needed to evaluate (2.1) in complex arithmetic. Note that the evaluation of (2.1) in complex arithmetic [8] did not take advantage of the special structure of $-L_p$ and L_ω , which might involve more round-off errors, give a complex energy value with a small imaginary part, and could make the final converged energy $E_{\eta,\Omega}(\phi)$ less accurate in high accuracy demand. To be more specific, given that L_ω is skew-symmetric, we have

$$-(\phi_r + i\phi_g)^* i L_\omega (\phi_r + i\phi_g) = -i \left(\underbrace{\phi_r^T L_\omega \phi_r}_0 + \underbrace{\phi_g^T L_\omega \phi_g}_0 \right) + 2\phi_r^T L_\omega \phi_g.$$

Similar results can be derived for L_s . The evaluation of the energy functional (2.1) in complex arithmetic and real arithmetic give identical results in exact arithmetic, without changing any essence of BEC. However, the evaluation of the energy functional for BEC can be done more efficiently in real arithmetic, because certain quantities equal to zero as shown above do not need to be evaluated, but such savings cannot be exploited in complex arithmetic. Furthermore, the real arithmetic form allows us to derive the Hessian easily and enable exact line search.

In order to employ the PCG method, it is necessary to obtain the gradient of $E_{\eta,\Omega}$ (3.1). Note that the energy expression of $E_{\eta,\Omega}$ (2.1) or (3.1) is valid under the normalization constraint $\|\phi\|_{\ell^2} = 1$. But we may disregard it and derive its gradient formally. The gradient of $E_{\eta,\Omega}$ (3.1) with respect to $\phi = (\phi_r^T \ \phi_g^T)^T$ is

$$(3.2) \quad \frac{\partial E_{\eta,\Omega}}{\partial \phi} = 2 \begin{pmatrix} L_s \phi_r + \eta \text{diag}(\phi_r^2 + \phi_g^2) \phi_r + \Omega L_\omega \phi_g \\ L_s \phi_g + \eta \text{diag}(\phi_r^2 + \phi_g^2) \phi_g - \Omega L_\omega \phi_r \end{pmatrix} h^d.$$

We can disregard the factor h^d and keep the direction $\frac{\partial E_{\eta,\Omega}}{\partial \phi}$. Since ϕ is restricted on the sphere $\|\phi\|_{\ell^2} = 1$, the *effective* gradient is the component of (3.2) that is orthogonal to ϕ :

$$(3.3) \quad r_{\eta,\Omega} = \begin{pmatrix} L_s \phi_r + \eta \text{diag}(\phi_r^2 + \phi_g^2) \phi_r + \Omega L_\omega \phi_g \\ L_s \phi_g + \eta \text{diag}(\phi_r^2 + \phi_g^2) \phi_g - \Omega L_\omega \phi_r \end{pmatrix} - \lambda_{\eta,\Omega} \begin{pmatrix} \phi_r \\ \phi_g \end{pmatrix},$$

where

$$(3.4) \quad \lambda_{\eta,\Omega} = [\phi_r^T L_s \phi_r + \phi_g^T L_s \phi_g + \eta(\phi_r^2 + \phi_g^2)^T (\phi_r^2 + \phi_g^2) + 2\Omega \phi_r^T L_\omega \phi_g] h^d$$

such that $\phi^T r_{\eta,\Omega} = 0$. Note that (3.3), up to a scaling factor, can also be derived by differentiating the scaling-invariant energy $E(\phi)$ as performed in section 4. We call $\lambda_{\eta,\Omega}$ the nonlinear Rayleigh functional of ϕ , which approximates the desired eigenvalue in (2.2). Also, $\lambda_{\eta,\Omega}$ represents the *chemical potential* [11] and

$$(3.5) \quad \lambda_{\eta,\Omega} = E_{\eta,\Omega} + \frac{\eta}{2} (\phi_r^2 + \phi_g^2)^T (\phi_r^2 + \phi_g^2) h^d = E_{\eta,\Omega} + E_{int},$$

where $E_{int} = \frac{\eta}{2} (\phi_r^2 + \phi_g^2)^T (\phi_r^2 + \phi_g^2) h^d$ is the *interaction energy*. Therefore, $r_{\eta,\Omega}$ is the eigenresidual associated with ϕ of the nonlinear eigenvalue problem (2.2).

Now, a nonlinear preconditioned conjugate gradient (PCG) method in real arithmetic could be employed in an effort to find the global minimizer of the energy functional (3.1). Suppose we work with a generic preconditioner M . The standard search direction in nonlinear PCG is

$$(3.6) \quad d_{(k)} = -M^{-1} r_{(k)} + \beta_{(k)} d_{(k-1)},$$

with the Fletcher-Reeves update [32]

$$\beta_{(k)} = \frac{\langle r_{(k)}, M^{-1} r_{(k)} \rangle}{\langle r_{(k-1)}, M^{-1} r_{(k-1)} \rangle}.$$

At iteration k , a regular update formula for $\phi_{(k+1)}$ in PCG could be

$$(3.7) \quad \phi_{(k+1)} = \phi_{(k)} \cos(\theta_{(k)}) + p_{(k)} \sin(\theta_{(k)}).$$

Here, $p_{(k)}$ is the modified search direction, which is orthogonal to $\phi_{(k)}$ in complex arithmetic and normalized in the standard ℓ^2 -space.

Given two complex vectors $\widehat{d_{(k)}} = d_{(k)r} + id_{(k)g}$ and $\widehat{\phi_{(k)}} = \phi_{(k)r} + i\phi_{(k)g} \in \mathbb{C}^n$, the real representation of $\widehat{d_{(k)}}$ and $\widehat{\phi_{(k)}}$ are $d_{(k)} = (d_{(k)r}^T \ d_{(k)g}^T)^T$ and $\phi_{(k)} = (\phi_{(k)r}^T \ \phi_{(k)g}^T)^T$, respectively. Then, the orthogonalization of $\widehat{d_{(k)}}$ against $\widehat{\phi_{(k)}}$ in complex arithmetic, which gives the resulted complex vector $\widehat{p_{(k)}} = p_{(k)r} + ip_{(k)g}$, can be done by their real representations as follows:

$$(3.8) \quad p_{(k)} = d_{(k)} - W(W^T W)^{-1} W^T d_{(k)},$$

where $p_{(k)} = \begin{pmatrix} p_{(k)r} \\ p_{(k)g} \end{pmatrix}$ and $W = \begin{pmatrix} \phi_{(k)r} & -\phi_{(k)g} \\ \phi_{(k)g} & \phi_{(k)r} \end{pmatrix}$. Moreover, the normalization condition $\|p_{(k)}\|_{\ell^2} = 1$ can be easily done by

$$(3.9) \quad p_{(k)} = p_{(k)} / (h^{\frac{d}{2}} \|p_{(k)}\|_2).$$

Note that (3.8) and (3.9) ensure that $\phi_{(k+1)}$ obtained from (3.7) satisfies the normalization constraint such that $\|\phi_{(k+1)}\|_{\ell^2} = 1$ for any $\theta_{(k)}$.

An outline of the nonlinear PCG is given in Algorithm 3.1.

Algorithm 3.1. The PCG method.

```

1: Start with an initial approximation  $\phi_{(0)}$  with  $\|\phi_{(0)}\|_{\ell^2} = 1$ .
2: while not converged do
3:    $\lambda_{(k)} = \lambda_{\eta, \Omega}(\phi_{(k)})$  (see (3.4)).
4:    $r_{(k)} = r_{\eta, \Omega}(\phi_{(k)})$  (see (3.3)).
5:    $\beta_{(k)} = \frac{\langle r_{(k)}, M^{-1}r_{(k)} \rangle}{\langle r_{(k-1)}, M^{-1}r_{(k-1)} \rangle}$ 
6:    $d_{(k)} = -M^{-1}r_{(k)} + \beta_{(k)}d_{(k-1)}$ 
7:    $p_{(k)} = d_{(k)} - W(W^TW)^{-1}W^Td_{(k)}$ 
8:    $p_{(k)} = p_{(k)} / (h^{\frac{d}{2}}\|p_{(k)}\|_2)$ 
9:    $\theta_{(k)} = \arg \min_{\theta} E(\phi_{(k)} \cos(\theta) + p_{(k)} \sin(\theta))$ 
10:   $\phi_{(k+1)} = \phi_{(k)} \cos(\theta_{(k)}) + p_{(k)} \sin(\theta_{(k)})$ 
11:   $k = k + 1$ 
12: end while

```

4. Problem-dependent Hessian preconditioner. One critical problem in the nonlinear PCG is to design a good preconditioner, which can significantly reduce the iteration counts and runtime. In general, the preconditioner for PCG near convergence should be an approximation to the Hessian of the objective function.

4.1. Derivation of the discrete Hessian operator. In this section, we will derive the explicit expression of the discrete Hessian operator for the energy functional $E_{\eta, \Omega}$ (3.1) based on the real arithmetic and introduce the preconditioning strategy in practice. It is crucial to point out the discrete Hessian operator of a real-valued scalar function of n complex variables must be a linear operator that operates on a vector of $2n$ degrees of freedom [39]. In particular, this means that such a Hessian operator may not be represented correctly as a complex matrix of order n . In [39], three definitions of the Hessian are given for such a function based on the real arithmetic or complex arithmetic, and all these Hessian matrices are of order $2n$. For the BEC problems, with our use of real arithmetic, it is also natural to derive the discrete Hessian operator of the energy functional (2.1) as a real symmetric matrix of order $2n$. In order to derive the full expression of discrete Hessian operator of $E_{\eta, \Omega}$, we will follow the scheme in [34] to absorb the normalization constraint to rewrite (3.1) in a form that is invariant with respect to the scaling of the wave function ϕ .

Assume that $\phi = (\phi_r^T \ \phi_g^T)^T$, which leads to the equivalent expression of (3.1)

$$(4.1) \quad E(\phi) = \frac{\phi^T A \phi}{\phi^T \phi} + \frac{\eta \phi^T B(\phi) \phi}{2 h^d (\phi^T \phi)^2},$$

where

$$A = \begin{pmatrix} L_s & \Omega L_\omega \\ -\Omega L_\omega & L_s \end{pmatrix} \quad \text{and} \quad B(\phi) = \begin{pmatrix} \text{diag}(\phi_r^2 + \phi_g^2) & 0 \\ 0 & \text{diag}(\phi_r^2 + \phi_g^2) \end{pmatrix}.$$

Here, A is real symmetric and $E(\phi)$ satisfies the scaling-invariant property, i.e., $E(\alpha\phi) = E(\phi)$ for any $\alpha \in \mathbb{R} \setminus \{0\}$. In this way, the energy functional $E(\phi)$ only depends on the relative strength of ϕ in different locations in Ω , or the direction of the vector ϕ . This would eliminate potential issues how the normalization condition $\|\phi\|_{\ell_2} = 1$ would impact the gradient and the Hessian of $E(\phi)$. Theorem 4.1 provides the complete expression of the gradient and the Hessian for $E(\phi)$ in (4.1).

THEOREM 4.1. *The gradient and Hessian for $E(\phi)$ (4.1) are given by*

$$(4.2) \quad \frac{\partial E(\phi)}{\partial \phi} = \frac{2}{\phi^T \phi} (A(\phi)\phi - \lambda(\phi)\phi)$$

and

$$(4.3) \quad \frac{\partial^2 E(\phi)}{\partial \phi^2} = \frac{2}{\phi^T \phi} \left\{ A + \frac{\eta}{h^d \phi^T \phi} \begin{pmatrix} \text{diag}(3\phi_r^2 + \phi_g^2) & 2\text{diag}(\phi_r \phi_g) \\ 2\text{diag}(\phi_r \phi_g) & \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - \lambda(\phi)I \right. \\ \left. - 2A \frac{\phi \phi^T}{\phi^T \phi} - 2 \frac{\phi \phi^T}{\phi^T \phi} A - 4\eta \frac{B(\phi)}{h^d \phi^T \phi} \frac{\phi \phi^T}{\phi^T \phi} - 4\eta \frac{\phi \phi^T}{\phi^T \phi} \frac{B(\phi)}{h^d \phi^T \phi} \right. \\ \left. + 4 \frac{\phi^T A \phi}{\phi^T \phi} \frac{\phi \phi^T}{\phi^T \phi} + 6\eta \frac{\phi \phi^T}{\phi^T \phi} \frac{\phi^T B(\phi) \phi}{h^d (\phi^T \phi)^2} \right\},$$

where

$$A(\phi) = A + \eta \frac{B(\phi)}{h^d \phi^T \phi} \quad \text{and} \quad \lambda(\phi) = \frac{\phi^T A \phi}{\phi^T \phi} + \eta \frac{\phi^T B(\phi) \phi}{h^d (\phi^T \phi)^2}.$$

Also, if $\phi = \begin{pmatrix} \phi_r \\ \phi_g \end{pmatrix}$ is a stationary point of $E(\phi)$ such that

$$\frac{\partial E(\phi)}{\partial \phi} = \frac{2}{\phi^T \phi} (A(\phi)\phi - \lambda(\phi)\phi) = 0,$$

we have

$$\frac{\partial^2 E(\phi)}{\partial \phi^2} \phi = 0 \quad \text{and} \quad \frac{\partial^2 E(\phi)}{\partial \phi^2} \hat{\phi} = 0,$$

where $\hat{\phi} = \begin{pmatrix} -\phi_g \\ \phi_r \end{pmatrix}$; that is, ϕ and $\hat{\phi}$ are the eigenvectors of $\frac{\partial^2 E(\phi)}{\partial \phi^2}$ associated with the zero eigenvalue.

Proof. The proof is given in the appendix. $\square$

Note that the gradient from Theorem 4.1 is consistent with (3.3) up to a scaling factor. Also, we can see that the discrete Hessian operator of the energy $E_{\eta,\Omega}(\phi)$ should be of order $2n$. Now, we are ready to introduce a shifted Hessian preconditioner for the nonlinear PCG method.

4.2. Preconditioning strategy. For the nonlinear PCG, it is crucial to apply the preconditioner efficiently. For the ground state solution, i.e., the global minimizer of $E_{\eta,\Omega}$, the first-order optimality condition is $r_{\eta,\Omega} = 0$, and the second-order optimality condition is $H_{\eta,\Omega} \succeq 0$ (positive semidefinite) with the null space spanned by the ground state solution $\phi = (\phi_r^T \ \phi_g^T)^T$ and $\hat{\phi} = (-\phi_r^T \ \phi_g^T)^T$. Suppose that P is the orthogonal projector with null space spanned by the ground state $\phi = (\phi_r^T \ \phi_g^T)^T$ and $\hat{\phi} = (-\phi_r^T \ \phi_g^T)^T$, i.e., $P = I - W(W^T W)^{-1} W^T$, where $W = \begin{pmatrix} \phi_r & -\phi_g \\ \phi_g & \phi_r \end{pmatrix}$.

With the normalization constraint $\phi^T \phi h^d = 1$, we obtain $W^T W = \frac{1}{h^d} I$, so that $P = I - h^d W W^T$. Since that $P\phi = \phi^T P = 0$, the low-rank updates in (4.3) can be cancelled out by multiplying $\frac{\partial^2 E(\phi)}{\partial \phi^2}$ on both sides by P . That is,

$$P \frac{\partial^2 E(\phi)}{\partial \phi^2} P = \frac{2}{\phi^T \phi} P \left\{ A + \frac{\eta}{h^d \phi^T \phi} \begin{pmatrix} \text{diag}(3\phi_r^2 + \phi_g^2) & 2\text{diag}(\phi_r \phi_g) \\ 2\text{diag}(\phi_r \phi_g) & \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - \lambda(\phi)I \right\} P.$$

Therefore, we can define the *effective* Hessian of $E_{\eta,\Omega}$ (3.1), i.e.,

$$(4.4) \quad H_{\eta,\Omega} := PH_pP = P \left\{ \begin{pmatrix} L_s + \eta \text{diag}(3\phi_r^2 + \phi_g^2) & \Omega L_\omega + 2\eta \text{diag}(\phi_r \phi_g) \\ -\Omega L_\omega + 2\eta \text{diag}(\phi_r \phi_g) & L_s + \eta \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - \lambda I_{2n} \right\} P,$$

where ϕ_r, ϕ_g is the column vector whose entries are the product of those of ϕ_r and ϕ_g . Moreover, the projector P adopted can avoid the potential stagnation of the “correction direction” that occurred in Davidson-type eigensolvers [43].

To speed up the convergence of our optimization methods, we define the *shifted Hessian preconditioner* based on (4.4) as

$$(4.5) \quad M_{\eta,\Omega} := PM_pP \\ = P \left\{ \begin{pmatrix} L_s + \eta \text{diag}(3\phi_r^2 + \phi_g^2) & \Omega L_\omega + 2\eta \text{diag}(\phi_r \phi_g) \\ -\Omega L_\omega + 2\eta \text{diag}(\phi_r \phi_g) & L_s + \eta \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - (\lambda - \sigma) I_{2n} \right\} P.$$

The shift $\sigma > 0$ is chosen such that $M_p \succ 0$ (positive definite) near the ground state and ensures that incomplete Cholesky factorization of M_p can be done successfully. A smaller σ lead to M_p closer to the effective Hessian $H_{\eta,\Omega}$ (more effective preconditioning), whereas a larger σ makes M_p less close to $H_{\eta,\Omega}$ (less effective). In practice, σ should be chosen to strike a balance between the chance of success of incomplete Cholesky factorization and the effectiveness of preconditioning. In our numerical experiments, we let $\sigma = (E_{\eta,\Omega} + \lambda_{\eta,\Omega})/2$ for the current iterate $\phi_{(k)}$ by default, though this choice can be easily changed if necessary.

Remark. Given the real formulation of our proposed preconditioner $M_{\eta,\Omega}$ (4.5), one might wonder if we could find a complex Hermitian $\widehat{M}_{\eta,\Omega}$ of order n , and form the vector u in complex arithmetic $\widehat{u} = u_r + iu_g$, such that $M_{\eta,\Omega}^{-1}u = M_{\eta,\Omega}^{-1}(u_r^T \ u_g^T)^T$ and $\widehat{M}_{\eta,\Omega}^{-1}\widehat{u}$ represent the same vector in real and complex arithmetic, respectively. This is equivalent to finding a complex Hermitian $\widehat{M}_p$ of order n such that $M_p^{-1}u$ and $\widehat{M}_p^{-1}\widehat{u}$ represent the same vector in real and complex arithmetic, respectively. It can be shown this is impossible. Suppose $M_p^{-1}u = v$ and $\widehat{M}_p^{-1}\widehat{u} = \widehat{v}$ such that $\widehat{v} = v_r + iv_g$, then M_pv and $\widehat{M}_p\widehat{v}$ represent the same vector in real and complex arithmetic, respectively. Let $\widehat{M}_p = \text{re}(\widehat{M}_p) + i\text{im}(\widehat{M}_p)$, then we have $M_pv = \begin{pmatrix} \text{re}(\widehat{M}_p) & -\text{im}(\widehat{M}_p) \\ \text{im}(\widehat{M}_p) & \text{re}(\widehat{M}_p) \end{pmatrix} v$, which leads to a contradiction, since the (1,2) and (2,1) blocks of M_p are not opposite of each other unless $\eta = 0$. Real arithmetic computation is essential to enable a wide range of options to approximate the action of the Hessian H_p for both Newton-like and preconditioner conjugate gradient-like methods for the minimization of $E_{\eta,\Omega}$.

The difficulty of applying the Hessian preconditioner depends on the discretization scheme used. Under the Fourier pseudo-spectral discretization scheme we adopt, the discrete Hessian operator (4.5) is fully dense, and geometric multigrid (GMG) is a reasonable method to approximate the action of $M_{\eta,\Omega}^{-1}$ on vectors. A more efficient alternative, however, is to construct the shifted Hessian operator (4.5) in finite difference discretization based on the same uniform mesh; this leads to a sparse approximation to the true discrete Hessian operator (4.3) in Fourier pseudo-spectral scheme, to which incomplete Cholesky factorization can be applied efficiently. This preconditioning strategy is reasonable in this setting, since the action of preconditioning usually does not need to be computed to high accuracy. In practice, we apply the eighth-order finite difference approximations to form $M_{\eta,\Omega}$ (4.5), which seems accurate enough to approximate the true discrete Hessian operator (4.3) given that the wave function ϕ has a complex pattern of vortices. In this way, the *shifted Hessian*

preconditioner actually used is sparse and can be applied efficiently. For large problems, as exact matrix factorizations are prohibitive, we apply incomplete Cholesky factorization with fill-reducing permutations such as approximate minimal degree ordering [3] and an appropriate drop tolerance. To further lower the computational cost, we keep the same preconditioner for a certain number of PCG steps before performing a new factorization. Generally, the Combined preconditioner is effective to help PCG proceed closer to the final minimized energy, but it tends to struggle or even stagnate near the convergence, whereas our Hessian preconditioner can help PCG converge to the final energy more rapidly in a robust manner. We propose using the combined preconditioner initially and switch to our Hessian preconditioner later. Note that the timing for switching the preconditioners should depend on the nonlinearity η and rotational speed Ω . Higher value of η or Ω makes the problem more challenging, thus we suggest switching the preconditioners in an early stage (e.g., the relative change in energy drops below 10^{-7}) to speed up convergence.

5. Fast energy functional evaluation and exact line search. For gradient-based optimization methods, it is a common practice to perform an approximate line search following the Armijo–Goldstein or Wolfe conditions [40], since exact line search is prohibitive for large problems. There are a few well-known counterexamples, such as PCG for solving a symmetric positive definite (SPD) linear system and computing the lowest eigenvalue(s) of an SPD matrix, as exact line search can be done efficiently with explicit formula for the optimal step size [33] or by the Rayleigh–Ritz projection [37]. The state-of-the-art variant of PCG for rotational BEC [8] performs line search by approximating $E_{\eta,\Omega}$ by a quadratic function and some complex methodologies based on different conditions and certain default values not explicitly specified. Fortunately, we find that fast exact line search can be enabled, *without* repeated evaluations of the energy functional at different step sizes in the original problem dimension n .

Specifically, let $\phi_{(k)}$ be the current ground state approximation with $\|\phi_{(k)}\|_{\ell^2} = 1$, and let $d_{(k)}$ be a search direction. We orthogonalize $d_{(k)}$ against $\phi_{(k)}$ then normalize it in $\|\cdot\|_{\ell^2}$ norm into $p_{(k)}$ following the process introduced in section 3. Then, the new iterate is $\phi_{(k+1)} = \phi_{(k)} \cos(\theta_{(k)}) + p_{(k)} \sin(\theta_{(k)})$, where $\theta_{(k)}$ is the minimizer $E(\phi_{(k)} \cos(\theta) + p_{(k)} \sin(\theta))$. By construction, we know $\|\phi_{(k+1)}\|_{\ell^2} = 1$. Consider the objective function $E_{\eta,\Omega}$ (3.1), substitute $\phi = (\phi_r^T \ \phi_g^T)^T$ with $(\phi_{(k)r}^T \ \phi_{(k)g}^T)^T \cos \theta + (p_{(k)r}^T \ p_{(k)g}^T)^T \sin \theta$ into $E_{\eta,\Omega}$. Then, by direct algebraic evaluation, we get

$$(5.1) \quad \begin{aligned} E_{\eta,\Omega}(\phi_{(k)} \cos \theta + p_{(k)} \sin \theta) &= [w(\theta)^T L_{s(k)} w(\theta) + 2\Omega w(\theta)^T L_{\omega(k)} w(\theta) \\ &\quad + \frac{\eta}{2} (c_1 \cos^4 \theta + c_2 \cos^3 \theta \sin \theta + c_3 \cos^2 \theta \sin^2 \theta + c_4 \cos \theta \sin^3 \theta + c_5 \sin^4 \theta)] h^d, \end{aligned}$$

where

$$\begin{aligned} w(\theta) &= (\cos \theta \ \sin \theta)^T, \quad L_{\omega(k)} = (\phi_{(k)r} \ p_{(k)r})^T L_{\omega} (\phi_{(k)g} \ p_{(k)g}) \in \mathbb{R}^{2 \times 2}, \\ L_{s(k)} &= (\phi_{(k)r} \ p_{(k)r})^T L_s (\phi_{(k)r} \ p_{(k)r}) + (\phi_{(k)g} \ p_{(k)g})^T L_s (\phi_{(k)g} \ p_{(k)g}) \in \mathbb{R}^{2 \times 2}, \end{aligned}$$

and

$$\begin{aligned} c_1 &= (\phi_{(k)r}^2 + \phi_{(k)g}^2)^T (\phi_{(k)r}^2 + \phi_{(k)g}^2), \quad c_2 = 4(\phi_{(k)r}^2 + \phi_{(k)g}^2)^T (\phi_{(k)r} p_{(k)r} + \phi_{(k)g} p_{(k)g}), \\ c_3 &= 4(\phi_{(k)r} p_{(k)r} + \phi_{(k)g} p_{(k)g})^T (\phi_{(k)r} p_{(k)r} + \phi_{(k)g} p_{(k)g}) + 2(\phi_{(k)r}^2 + \phi_{(k)g}^2)^T (p_{(k)r}^2 + p_{(k)g}^2), \\ c_4 &= 4(\phi_{(k)r} p_{(k)r} + \phi_{(k)g} p_{(k)g})^T (p_{(k)r}^2 + p_{(k)g}^2), \quad c_5 = (p_{(k)r}^2 + p_{(k)g}^2)^T (p_{(k)r}^2 + p_{(k)g}^2). \end{aligned}$$

Here, $\phi_{(k)r}$ and $\phi_{(k)g}$ stand for the real part and imaginary part of $\phi_{(k)}$, respectively; $p_{(k)r}$ and $p_{(k)g}$ are defined similarly, and $\phi_{(k)r}p_{(k)r}$ stands for the column vector whose entries are the product of those of $\phi_{(k)r}$ and $p_{(k)r}$; $\phi_{(k)r}^2$, $\phi_{(k)g}^2$, $p_{(k)r}^2$ and $p_{(k)g}^2$ can be defined similarly.

The key observation is that it only takes six matrix vector multiplications of order n , 18 vector inner product of order n , 6 elementwise vector multiplications of order n and, three vector additions of order n to obtain $L_{s(k)}$, $L_{\omega(k)} \in \mathbb{R}^{2 \times 2}$, and the scalars c_i ($1 \leq i \leq 5$), *no more* computation in the original problem dimension n is needed. Now $E_{\eta,\Omega}(\phi_{(k)} \cos \theta + p_{(k)} \sin \theta) : \mathbb{R} \rightarrow \mathbb{R}$ can be evaluated for any and as many values of θ as needed at little arithmetic cost. We can afford to perform a numerical exact line search to minimize $E_{\eta,\Omega}(\phi_{(k)} \cos \theta + p_{(k)} \sin \theta)$, or find the minimizer by forming $\frac{d}{d\theta} E_{\eta,\Omega}(\phi_{(k)} \cos \theta + p_{(k)} \sin \theta) = 0$ in closed form and solving it for θ . In our implementation, we use MATLAB's `fminsearch` function to find the optimal θ , which can be done rapidly without additional work on dimension n . Note that this procedure is equivalent to the Rayleigh–Ritz procedure in many iterative methods to solve linear or linearized symmetric eigenproblems for the lowest eigenvalues.

Similarly, the fast exact search can be applied to the locally optimal preconditioned conjugate gradient method (LOPCG) [38], which is very successful in nonlinear eigenproblems. Assume that we have determined two search directions $p_{(k)}, f_{(k)}$ such that $\|p_{(k)}\|_{\ell^2}^2 = \|f_{(k)}\|_{\ell^2}^2 = 1$, and $\phi_{(k)}, p_{(k)}, f_{(k)}$ are pairwise orthogonal. To determine the new iterate $\phi_{(k+1)}$ for which the energy functional is minimized, define $\phi_{(k+1)} = \phi_{(k)} \cos \theta + p_{(k)} \sin \theta \cos \gamma + f_{(k)} \sin \theta \sin \gamma$. The simplified expression $E_{\eta,\Omega}(\phi_{(k+1)})$ can be derived and one only needs to compute $L_{s(k)}, L_{\Omega(k)} \in \mathbb{R}^{3 \times 3}$ and 15 scalar coefficients once to evaluate $E_{\eta,\Omega}$ for all values of (θ, γ) efficiently. However, we found that adding more directions in the search subspace does not yield significant gain in runtime consistently.

In [8, 9], a quadratic approximation line search is provided. Without the details of the specified parameters, we find that it is not easy to achieve the fast convergence of the nonlinear PCG. Here, we provide a modified quadratic approximation line search. Assuming $\epsilon_{(k)}$ is the eigenresidual at the step k , we can approximate $E_{\eta,\Omega}$ by a quadratic function, which is evaluated at $\theta_{(k)} = 0$, $\epsilon_{(k)}/2$, and $\epsilon_{(k)}$, respectively. Then, we can use the minimizer $\theta_{(k)}^{opt}$ of the corresponding quadratic function as a trial step size. If the energy $E_{\eta,\Omega}(\theta_{(k)}^{opt})$ is decreased, we accept this step. Otherwise, we reject this step, decrease the interpolation step sizes by a factor of 2 (e.g., 0, $\epsilon_{(k)}/4$ and $\epsilon_{(k)}/2$), and try again, until the energy is decreased, which ensures that $\theta_{(k)}$ is small enough. However, the performance of the nonlinear PCG can still be affected by the choice of the three interpolation points. Furthermore, a backtracking line search with Armijo–Goldstein condition [10] can also be employed here. Compared with these line search methods, our exact line search can avoid tuning parameters and help the nonlinear PCG converge more robustly.

6. Global convergence of CG. In this section, we explore the convergence of the CG method for computing the ground state of BEC. Under the BEC problem setting, we follow [45] to provide a proof of the global convergence of a special variant of CG based on the equivalent expression of the energy functional $E(\phi)$ (4.1). Here, we need to consider a variant of CG that can normalize ϕ in any manner at each step without changing its behavior, so that the widely adopted proof of global convergence of CG can be extended to our problem setting. For the simplicity of notation, we denote $\frac{\partial E(\phi)}{\partial \phi}$ as $\nabla E(\phi)$ for the remainder of this section. Given the scaling-invariant $E(\phi)$ (4.1), i.e., $E(\alpha\phi) = E(\phi)$ for any nonzero scalars α and vectors ϕ , we can

show that $\frac{1}{\alpha} \nabla E(\phi) = \nabla E(\alpha\phi)$. Therefore, the traditional stopping criterion of CG in the general setting of optimization $\|\nabla E(\alpha\phi)\| \leq \delta$ for some small $\delta > 0$ cannot guarantee that ϕ approximates the desired solution in direction. Actually, due to the dependence on scaling, $\nabla E(\phi)$ does not satisfy the Lipschitz condition $\|\nabla(E(\phi_1)) - \nabla(E(\phi_2))\| \leq L\|\phi_1 - \phi_2\|$ for all $\phi_1, \phi_2 \in \mathbb{R}^{2n} \setminus \{0\}$. In [26], the CG method on Grassman manifolds is developed to address this issue in a similar problem setting. However, the understanding of theoretical properties of these CG methods, especially their convergence, remains far from complete.

Instead, we propose a special Fletcher–Reeves variant of CG with exact line search (Algorithm 6.1), which works independently of the scaling of any iterate ϕ_k . To be more specific, at step 5, we are free to scale each new iterate ϕ_{k+1} by any nonzero factor, which allows us to avoid great effort to tune CG to proceed in a manner consistent with the geometry of the unit sphere for BEC. At step 4, both β_k and p_k in Algorithm 6.1 are scaling-invariant of the CG iterates by construction, i.e., they depend on the directions, instead of the scalings, of $\phi_0, \phi_1, \dots$. Therefore, one can normalize ϕ_k in any convenient manner after each CG step without being concerned about the geometric constraints for BEC. In this way, the CG method under BEC setting is consistent with that in [45] for the nonlinear Hermitian eigenproblems $T(\lambda)v = 0$ with a variational characterization, such that the global convergence can be established. Note that, if $\{\phi_k\}$ in Algorithm 6.1 satisfies the normalization constraint, i.e., $\|\phi_k\|_{\ell^2} = 1$, then it is equivalent to Algorithm 3.1 given the identity preconditioner.

Our main interest is to prove the global convergence of Algorithm 6.1 towards a critical point of $E(\phi)$. Note that a critical point of $E(\phi)$ indicates that $\nabla E(\phi) = 0$, i.e., $(A(\phi) - \lambda(\phi)I)\phi = 0$. At such a point, the matrix $(A(\phi) - \lambda(\phi)I)$ must be singular, since ϕ is a nonzero vector. A complete proof will be long and technical, so we only provide the major steps of it. More details of the analysis can be found in [45]. Several intermediate results are necessary to be established.

DEFINITION 6.1. *The gradient $\nabla E(\phi)$ as given in (4.2) is called Lipschitz continuous in direction if there is a constant $L > 0$ such that $\|\|\phi_1\| \nabla E(\phi_1) - \|\phi_2\| \nabla E(\phi_2)\| \leq L\alpha$ for all $\phi_1, \phi_2 \in \mathbb{R}^{2n} \setminus \{0\}$ that satisfy $\alpha = \angle(\phi_1, \phi_2) \leq \frac{\pi}{2}$.*

For $\nabla E(\phi)$ that is Lipschitz continuous in direction, the following inequality [45] can be derived:

$$(6.1) \quad \|\|\phi_1\| \nabla E(\phi_1) - \|\phi_2\| \nabla E(\phi_2)\| \leq \frac{\pi L \|\phi_1 - \phi_2\|}{2 \max(\|\phi_1\|, \|\phi_2\|)}.$$

Given the above inequality, and the fact that exact line search also satisfies the strong Wolfe conditions, we have the following inequality from [45, Theorem 3.4]:

Algorithm 6.1. A scaling-invariant CG method.

- 1: Start with an initial approximation ϕ_0 .
 - 2: **for** $k = 0, 1, \dots$, until convergence, i.e., $\|\phi_k\| \|\nabla E(\phi_k)\| \leq \delta$ **do**
 - 3: $E_k = E(\phi_k)$ (see (4.1)).
 - 4: If $k=0$, $p_0 = -\|\phi_0\| \nabla E(\phi_0)$ (see (4.2)); otherwise, $p_k = -\|\phi_k\| \nabla E(\phi_k) + \beta_k p_{k-1}$,
 where $\beta_k = \frac{\nabla E(\phi_k)^T \nabla E(\phi_k) \|\phi_k\|^2}{\nabla E(\phi_{k-1})^T \nabla E(\phi_{k-1}) \|\phi_{k-1}\|^2}$.
 - 5: $\phi_{k+1} = \phi_k + \tau_k p_k$ through exact line search, and normalize ϕ_{k+1} if necessary.
 - 6: $k = k + 1$
 - 7: **end for**
-

$$(6.2) \quad \sum_{k=0}^{\infty} \|\nabla E(\phi_k)\|^2 \|\phi_k\|^2 \cos^2 \theta_k < \infty,$$

where $\theta_k = \angle(-\nabla E(\phi_k), p_k)$, such that $\cos \theta_k = \frac{-\nabla E(\phi_k)^T p_k}{\|\nabla E(\phi_k)\| \|p_k\|} > 0$. Moreover, since exact line search is used in Algorithm 6.1, some preliminary results can be immediately obtained in Proposition 6.2.

PROPOSITION 6.2. *Algorithm 6.1 generates $\{p_k\}$ and $\nabla E(\phi_k)$ satisfying*

- (i) $p_k^T \nabla E(\phi_{k+1}) = 0$,
- (ii) $p_k^T \nabla E(\phi_k) = -\|\phi_k\| \|\nabla E(\phi_k)\|^2$,
- (iii) $\|p_k\|^2 = \|\phi_k\|^2 \|\nabla E(\phi_k)\|^2 + \beta_k^2 \|p_{k-1}\|^2$,
- (iv) $\|\phi_k\| \|\nabla E(\phi_k)\| \leq \|p_k\|$.

Proof. The proof is omitted and can be found in [45, Proposition 3.8]. $\square$

With the above preliminary results, we are ready to establish the global convergence of Algorithm 6.1. In Theorem 6.3, $E_{\min}$ and $E_{\max}$ are the lowest and highest value of $E(\phi)$, respectively. These two values are finite as a result of the extreme value theorem [41] of a continuous multivariate function on a closed finite set (unit sphere of ϕ). Moreover, $\{E_i\}$ refer to the discrete energy levels of a given BEC problem, each of which is a critical value of $E(\phi)$, and λ_ℓ is the corresponding chemical potential of E_ℓ ; see (2.1) and (2.3). Note that Theorem 6.3 cannot guarantee the CG method converges to the ground state, i.e., global minimizer of BEC; however, if $E(\phi_0)$ is lower than the second lowest critical value of $E(\phi)$, then Algorithm 6.1 indeed converges to the ground state of BEC.

THEOREM 6.3. *Let $J = (a, b)$ be finite, such that $[E_{\min}, E_{\max}] \subset J$. Let $\phi_0 \neq 0$ be the initial iterate of Algorithm 6.1. Assume that $\nabla E(\phi)$ is Lipschitz continuous in direction in a neighborhood of $S = \{\frac{\phi}{\|\phi\|} \mid E(\phi) \leq E(\phi_0)\}$. Then, there exists $E_\ell \in \{E_i\}$ such that $\lim_{k \rightarrow \infty} E(\phi_k) = E_\ell$ and there is a subsequence $\{\phi_{k_j}\}$ such that $\lim_{j \rightarrow \infty} \angle(\phi_{k_j}, \text{null}(A(\phi_{k_j}) - \lambda_\ell I)) = 0$. That is, $\{\phi_{k_j}\}$ converges to the eigenvector corresponding to the eigenvalue λ_ℓ as defined in (2.3).*

Proof. The outline of our proof is as follows. First, we show that there exists a subsequence $\{\phi_{k_j}\}$ of the CG iterates such that $\lim_{j \rightarrow \infty} \|\phi_{k_j}\| \|\nabla E(\phi_{k_j})\| = 0$. Then, we show that $\lim_{k \rightarrow \infty} E(\phi_k)$ must be a critical value of $E(\phi)$, and that the subsequence $\{\phi_{k_j}\}$ satisfying $\lim_{j \rightarrow \infty} \|\phi_{k_j}\| \|\nabla E(\phi_{k_j})\| = 0$ converges to the critical points of energy functional $E(\phi)$ (4.1).

Note that, at each step, Algorithm 6.1 generates a new iterate ϕ_k satisfying $E(\phi_k) < E(\phi_{k-1})$, such that all iterates ϕ_k belong to the level set S . Consider a continuous function $(\lambda, \phi) \rightarrow \|A(\phi) - \lambda I\|$ defined on the finite and closed domain $[\lambda_{\min}, \lambda_{\max}] \times S$, then there exists an $M > 0$ such that $\|A(\phi) - \lambda I\| \leq M$ for all $\phi \in S$. Note that $\lambda_{\min}, \lambda_{\max}$ are not necessarily the corresponding chemical potential of $E_{\min}, E_{\max}$. Then, we have $\|\phi\| \|\nabla E\|_{\phi \in S} = \frac{2\|(A(\phi) - \lambda(\phi))\phi\|}{\|\phi\|} \leq 2\|A(\phi) - \lambda(\phi)I\| \leq 2M < \infty$. Also, for $\theta_k = \angle(-\nabla E(\phi_k), p_k)$, we have $\cos \theta_k = \frac{-\nabla E(\phi_k)^T p_k}{\|\nabla E(\phi_k)\| \|p_k\|} = \frac{\|\nabla E(\phi_k)\| \|\phi_k\|}{\|p_k\|}$ (Proposition 6.2(ii)). Then, we have

$$(6.3) \quad \sum_{k=0}^{\infty} \|\nabla E(\phi_k)\|^2 \|\phi_k\|^2 \cos^2 \theta_k = \sum_{k=0}^{\infty} \frac{\|\nabla E(\phi_k)\|^4 \|\phi_k\|^4}{\|p_k\|^2} < \infty.$$

Assume by contradiction that there exists a $\gamma > 0$, such that $\|\phi_k\| \|\nabla E(\phi_k)\| \geq \gamma$ for all k . Also, we know $\beta_k^2 \beta_{k-1}^2 \dots \beta_{k-i}^2 = \frac{\|\nabla E(\phi_k)\|^4 \|\phi_k\|^4}{\|\nabla E(\phi_{k-i-1})\|^4 \|\phi_{k-i-1}\|^4}$ by the definition of β_k at step 4 in Algorithm 6.1. It follows from Proposition 6.2(iii) that

$$\begin{aligned} \|p_k\|^2 &= \|\phi_k\|^2 \|\nabla E(\phi_k)\|^2 + \frac{\|\nabla E(\phi_k)\|^4 \|\phi_k\|^4}{\|\nabla E(\phi_{k-1})\|^2 \|\phi_{k-1}\|^2} + \cdots + \frac{\|\nabla E(\phi_k)\|^4 \|\phi_k\|^4}{\|\nabla E(\phi_0)\|^2 \|\phi_0\|^2} \\ &= \|\phi_k\|^4 \|\nabla E(\phi_k)\|^4 \sum_{i=0}^k \frac{1}{\|\nabla E(\phi_i)\|^2 \|\phi_i\|^2} \leq \gamma_k^4 \frac{k+1}{\gamma^2}, \end{aligned}$$

where $\gamma_k = \|\phi_k\| \|\nabla E(\phi_k)\|$ satisfies $0 < \gamma \leq \gamma_k \leq 2M < \infty$. It follows that

$$\sum_{k=0}^m \frac{1}{\|p_k\|^2} \geq \gamma^2 \sum_{k=0}^m \frac{1}{\gamma_k^4 (k+1)} \geq \frac{\gamma^2}{(2M)^4} \sum_{k=0}^m \frac{1}{k+1}$$

and thus

$$(6.4) \quad \sum_{k=0}^{\infty} \frac{1}{\|p_k\|^2} \geq \frac{\gamma^2}{(2M)^4} \sum_{k=0}^{\infty} \frac{1}{k+1} = \infty.$$

However, since $\sum_{k=0}^{\infty} \frac{\|\nabla E(\phi_k)\|^4 \|\phi_k\|^4}{\|p_k\|^2} < \infty$ and $\|\nabla E(\phi_k)\| \|\phi_k\| \geq \gamma > 0$ for all k by assumption, we have $\sum_{k=0}^{\infty} \frac{1}{\|p_k\|^2} < \infty$, which leads to a contradiction. Therefore, such $\gamma > 0$ does not exist, and we have $\lim_{k \rightarrow \infty} \inf \|\phi_k\| \|\nabla E(\phi_k)\| = 0$, which suggests that there exists a subsequence $\{\phi_k\}$, denoted as $\{\phi_{k_j}\}$, such that

$$\lim_{j \rightarrow \infty} \|\phi_{k_j}\| \|\nabla E(\phi_{k_j})\| = 0.$$

Meanwhile, since $E(\phi) \in [E_{\min}, E_{\max}] \subset J$ and $\{E(\phi_k)\}$ is monotonically decreasing, there exists $E^* \in [E_{\min}, E_{\max}]$ such that $\lim_{k \rightarrow \infty} E(\phi_k) = E^*$. Then, we have $\lim_{j \rightarrow \infty} E(\phi_{k_j}) = E^*$. Next, we show E^* is a critical value by contradiction. Define $\lambda^* = \lim_{j \rightarrow \infty} \lambda(\phi_{k_j})$, where $\lambda(\phi)$ is described in (2.3), corresponding to E^* . Assume that E^* is not a critical value, i.e., $\nabla E(\phi) \neq 0$ at E^* , then $\lim_{j \rightarrow \infty} (A(\phi_{k_j}) - \lambda^* I)$ is nonsingular from (4.2). Let $\sigma_{\min}^* > 0$ be the smallest singular value of $\lim_{j \rightarrow \infty} (A(\phi_{k_j}) - \lambda^* I)$ such that $\|\lim_{j \rightarrow \infty} (A(\phi_{k_j})\phi - \lambda^* \phi)\| \geq \sigma_{\min}^* \|\phi\|$ for all $\phi \neq 0$. Then,

$$0 = \lim_{j \rightarrow \infty} \|\phi_{k_j}\| \|\nabla E(\phi_{k_j})\| = \lim_{j \rightarrow \infty} 2 \left\| (A(\phi_{k_j}) - \lambda(\phi_{k_j})) \frac{\phi_{k_j}}{\|\phi_{k_j}\|} \right\| \geq 2\sigma_{\min}^* > 0,$$

which leads to a contradiction. Therefore, $E^* = \lim_{k \rightarrow \infty} E(\phi_k)$ is a critical value, denoted as E_{ℓ} , which may be the energy of the ground state or an excited state.

Finally, let $\{\phi_{k_j}\}$ be a subsequence such that $\lim_{j \rightarrow \infty} \|\phi_{k_j}\| \|\nabla E(\phi_{k_j})\| = 0$. Assume that there is a $\delta > 0$ independent of the iteration count, such that for any $M > 0$, there exists an $m > M$ such that $\angle(\phi_{k_m}, \text{null}(A(\phi_{k_m}) - \lambda_{\ell} I)) \geq \delta$. Then

$$\lim_{j \rightarrow \infty} \|\phi_{k_j}\| \|\nabla E(\phi_{k_j})\| = \lim_{j \rightarrow \infty} \frac{2 \|(A(\phi_{k_j}) - \lambda_{\ell} I) \phi_{k_j}\|}{\|\phi_{k_j}\|} \neq 0,$$

contradicting the assumption about $\{\phi_{k_j}\}$. Thus, there exists a subsequence $\{\phi_{k_j}\}$ of the CG iterates in Algorithm 6.1, such that $\lim_{j \rightarrow \infty} \angle(\phi_{k_j}, \text{null}(A(\phi_{k_j}) - \lambda_{\ell} I)) = 0$. $\square$

7. Expected behavior of PCG near convergence. Given the well-known results about the convergence properties of the CG method for preconditioned linear systems [42] and the steepest descent methods for nonlinear constrained minimization [1], the convergence of the nonlinear PCG relies on the properties of the preconditioned Hessian operator. In this section, we show the pattern of the spectrum for the

preconditioned Hessian with our shifted Hessian preconditioner (4.5), such that the nonlinear PCG is expected to converge quickly when approaching convergence.

The preconditioned Hessian operator with the *shifted Hessian preconditioner* is defined as, in the way suggested in [8],

$$(7.1) \quad PM_{\eta,\Omega}^{-1} \frac{\partial^2 E(\phi)}{\partial \phi^2} P.$$

Suppose that $M_{\eta,\Omega} = P(H_p + \sigma I)P = PLL^T P$, where the exact symmetric matrix factor L can be obtained by the exact Cholesky factorization with or without fill-reducing permutation. Then, a symmetric version of the preconditioned Hessian with the shifted Hessian preconditioner can be defined as

$$(7.2) \quad H_c = PL^{-1} P \frac{\partial^2 E(\phi)}{\partial \phi^2} PL^{-T} P.$$

Note that the preconditioned Hessian (7.2) is defined ideally, since we cannot afford the exact Cholesky decomposition for large problems in practice. Nevertheless, this ideal preconditioned Hessian helps us develop insight into the expected favorable behavior of the nonlinear PCG with our practical preconditioner near convergence.

THEOREM 7.1. *The preconditioned Hessian operator with the ideal shifted Hessian preconditioner (4.5) given in (7.2) can be written in the following form:*

$$(7.3) \quad L^{-1}H_p L^{-T} + WW_1^T + L^{-1}WW_2^T + L^{-1}H_p WW_3^T - h^d L^{-1}H_p L^{-T} WW^T$$

for $\sigma > 0$ and

$$(7.4) \quad I + W(W_1 - h^d W)^T + L^{-1}WW_2^T + L^{-1}H_p WW_3^T$$

for $\sigma = 0$. Here, $W_1, W_2, W_3 \in \mathbb{R}^{2n \times 2}$. In other words, the ideal preconditioned Hessian is a rank-6 update of the identity matrix for $\sigma = 0$, and a rank-8 update of $L^{-1}H_p L^{-T}$ that is close to the identity matrix for a small $\sigma > 0$.

Proof. The proof is given in the appendix. $\square$

Theorem 7.1 implies that almost all eigenvalues of the ideal preconditioned Hessian (7.2) are exactly or nearly 1, and there are only six or eight eigenvalues that could be significantly different from 1. Among these six or eight eigenvalues, there are two zero eigenvalues associated with the orthogonal projector P and have no impact on the convergence of the nonlinear PCG. Most importantly, such an observation of the spectrum is independent of the mesh size. Specifically, suppose $\sigma > 0$ and define the rank-8 matrix

$$R_8 = WW_1^T + L^{-1}WW_2^T + L^{-1}H_p WW_3^T - h^d L^{-1}H_p L^{-T} WW^T.$$

Then, by Theorem 7.1, we have

$$(7.5) \quad H_c = I + R_8 - \sigma L^{-1} L^{-T}.$$

Suppose α_i, ρ_i for $1 \leq i \leq 2n$ (satisfying $\alpha_i \leq \alpha_{i+1}, \rho_i \leq \rho_{i+1}$) are the eigenvalues of $I + R_8$ and $I + R_8 - \sigma L^{-1} L^{-T}$, respectively. Then, $2n - 8$ eigenvalues among α_i 's are 1 and at most eight eigenvalues are not 1. Moreover, by the Bauer-Fike theorem [27], we have $|\alpha_i - \rho_i| \leq |\sigma| \|L^{-1} L^{-T}\|_2$.

Also, we know that $L^{-1}L^{-T}$ and $(H_p + \sigma I)^{-1}$ have the same eigenvalues, since $H_p + \sigma I = LL^T$. Assume that the lowest positive eigenvalue of H_p (positive semidefinite) has an infimum $s^* > 0$ that is independent of the mesh size h , as $h \rightarrow 0$. It follows that the spectrum of $L^{-1}L^{-T}$ falls within $(0, \frac{1}{s^* + \sigma}]$, i.e., $\|L^{-1}L^{-T}\|_2 \leq \frac{1}{s^* + \sigma}$. Then, we have

$$(7.6) \quad |\alpha_i - \rho_i| \leq \frac{|\sigma|}{s^* + \sigma},$$

which is guaranteed to be small if σ is small compared to s^* . In other words, for any $\sigma \ll s^*$, the eigenvalues of $I + R_8 - \sigma L^{-1}L^{-T}$ are not much different from those of $I + R_8$. This ensures that the preconditioned Hessian has a favorable eigenvalue distribution such that the nonlinear PCG with our ideal shifted Hessian preconditioner (where σ is sufficiently small) is expected to converge fairly quickly when approaching convergence. With the incomplete Cholesky preconditioner obtained by a fixed drop tolerance, the condition number of $L^{-1}H_pL^{-T}$ deteriorates as the mesh size h decreases, so that nonlinear PCG needs more iterations to converge on a finer mesh.

8. Numerical experiments. In this section, we perform extensive experiments in 2D and 3D domains to validate our method. We compare our Hessian preconditioner with the combined preconditioner proposed in [8]. In the following experiments, we consider the trapping potential: the harmonic plus quartic potential for $d = 2, 3$,

$$(8.1) \quad V(\mathbf{x}) = (1 - \alpha)(\gamma_x^2 x^2 + \gamma_y^2 y^2) + \frac{\kappa(x^2 + y^2)^2}{4} + \begin{cases} 0, & d = 2, \\ \gamma_z^2 z^2, & d = 3. \end{cases}$$

Moreover, we take the initial wave function $\phi_{(0)}$ as the Thomas–Fermi approximation [8, 11]

$$(8.2) \quad \phi_{(0)} = \frac{\phi^{TF}}{\|\phi^{TF}\|_{\ell^2}} \quad \text{with} \quad \phi^{TF}(\mathbf{x}) = \begin{cases} \sqrt{(\mu^{TF} - V(\mathbf{x}))/\eta}, & V(\mathbf{x}) < \mu^{TF} \\ 0 & \text{otherwise,} \end{cases}$$

where

$$(8.3) \quad \mu^{TF} = \frac{1}{2} \begin{cases} (4\eta\gamma_x\gamma_y)^{1/2}, & d = 2, \\ (15\eta\gamma_x\gamma_y\gamma_z)^{2/5}, & d = 3. \end{cases}$$

The stopping criterion we adopt is

$$(8.4) \quad \frac{|E(\phi_{(k+1)}) - E(\phi_{(k)})|}{|E(\phi_{(k)})|} \leq \epsilon = 10^{-14}.$$

Other stopping criterion and comparison between them can be found in [11]. In order to apply our Hessian preconditioner, we perform an inexact Cholesky factorization with the approximate minimal degree ordering, and the drop tolerance is chosen to be 10^{-3} and $10^{-2.5}$ for experiments in 2D and 3D domains, respectively. We use the two stage preconditioning strategy. The combined preconditioner is used at the first stage and our Hessian preconditioner is used at the second stage. We switch the preconditioner when $\frac{|E(\phi_{(k+1)}) - E(\phi_{(k)})|}{|E(\phi_{(k)})|} \leq 10^{-7}$ for the first time. After switching to the Hessian preconditioner, we update the Hessian preconditioner every 100 iterations and 300 iterations for experiments in 2D domains and in 3D domains, respectively.

TABLE 8.1

Comparison of line search for quadratic approximation and exact line search for case I.

$\eta = 100, \Omega = 0.9$	Exact	Quadratic		Backtracking	
	Fast	Fast	Slow	Fast	Slow
PCG iteration	141	142	133	205	205
time (sec)	24.45	25.53	43.1	35.67	60.37

TABLE 8.2

Comparison of line search for quadratic approximation and exact line search for case II.

$\eta = 1000, \Omega = 2$	exact	quadratic		backtracking	
	fast	fast	slow	fast	slow
PCG iteration	302	310	309	579	579
time (sec)	53.91	54.30	86.82	100.65	230.62

8.1. Fast energy evaluation and line search methods. In this section, we perform several experiments to compare the performances of the nonlinear PCG with different line search methods (*with* or *without* fast evaluation of the energy) that we introduce in section 5. Note that we cannot afford to implement the exact line search *without* the fast evaluation of the energy. Therefore, there are five schemes: (a) exact line search *with* fast evaluation; (b) quadratic line search *with* fast evaluation; (c) quadratic line search *without* fast evaluation; (d) backtracking line search *with* fast evaluation; (e) backtracking line search *without* fast evaluation. Note that the exact line search performed by MATLAB's built-in function `fminsearch` is parameter-free, and the quadratic and backtracking line search use a small number of parameters whose values are predetermined, independent of the test problems. Here, we test two cases: (I) $\eta = 100$, $\Omega = 0.9$, and $V(\mathbf{x})$ is chosen with $\gamma_x = \gamma_y = 1$, $\alpha = 0.5$, and $\kappa = 0$; (II) $\eta = 1000$, $\Omega = 2$, and $V(\mathbf{x})$ is chosen with $\gamma_x = \gamma_y = 1$, $\alpha = 1.2$, and $\kappa = 0.3$. The computational domain and mesh size are $\mathcal{D} = [-10, 10]^2$ and $h = \frac{1}{32}$, respectively. To make fair comparison, all the experiments are performed with the combined preconditioner only. The results are summarized in Tables 8.1 and 8.2, respectively. We can see that the exact line search and quadratic line search are more competitive than the backtracking line search. Also, the exact line search could marginally improve the number of iterations compared with the quadratic line search. More importantly, the fast evaluation of the energy is always preferred.

8.2. Partial spectrum of preconditioned Hessian. In this section, we provide numerical examples to illustrate the partial spectrum of preconditioned Hessian with our Hessian preconditioner (7.2) at the converged ground state solution ϕ_c . Here, we have two cases: (I) $\eta = 500$, $\Omega = 0.8$, and $V(\mathbf{x})$ is chosen with $\gamma_x = \gamma_y = 1$, $\alpha = 0.5$, and $\kappa = 0$; (II) $\eta = 5000$ and $\Omega = 1$, and $V(\mathbf{x})$ is chosen with $\gamma_x = \gamma_y = 1$, $\alpha = 1.2$, and $\kappa = 0.3$. We use the MATLAB built-in function `eigs` to compute the partial spectrum for the preconditioned Hessian operators.

8.2.1. Partial spectrum with different shift σ . In this example, we apply the eighth order finite difference scheme to form (7.2), i.e, finite difference method for both the effective Hessian itself (4.4) and the Hessian preconditioner (4.5). Also, we perform an exact Cholesky factorization of the Hessian preconditioner for illustration, which is too expensive for large realistic problems. Here, we fix $h = \frac{1}{8}$ and the shift σ varies. The computational domain is $\mathcal{D} = [-10, 10]^2$. Tables 8.3 and 8.4 list the 10

TABLE 8.3

Partial spectrum of preconditioned Hessian with different shift σ for case I.

$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$
-6.49e-16	1.25	-4.96e-17	1.24	-4.49e-16	1.22
-1.66e-19	1.00	4.84e-17	1.00	-1.90e-16	1.00
2.33e-08	1.00	8.65e-08	1.00	1.62e-07	1.00
2.87e-05	1.00	3.00e-05	1.00	3.86e-05	1.00
5.18e-02	1.00	5.44e-03	1.00	5.53e-04	1.00
1.82e-01	1.00	2.18e-02	1.00	2.22e-03	1.00
5.05e-01	1.00	9.28e-02	1.00	1.01e-02	1.00
5.43e-01	1.00	1.06e-01	1.00	1.17e-02	1.00
8.24e-01	1.00	3.20e-01	1.00	4.49e-02	1.00
8.49e-01	1.00	3.61e-01	1.00	5.35e-02	1.00
(a) $\sigma = 10^{-3}$		(b) $\sigma = 10^{-2}$		(c) $\sigma = 10^{-1}$	

TABLE 8.4

Partial spectrum of preconditioned Hessian with different shift σ for case II.

$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$
-3.69e-16	1.77	-2.96e-16	1.17	-6.90e-17	1.16
2.92e-16	1.17	-8.12e-17	1.02	-4.16e-17	1.00
3.11e-06	1.00	4.33e-07	1.00	9.30e-06	1.00
4.55e-05	1.00	4.71e-05	1.00	4.65e-05	1.00
9.60e-01	1.00	6.97e-01	1.00	1.93e-01	1.00
9.60e-01	1.00	7.11e-01	1.00	1.94e-01	1.00
9.83e-01	1.00	8.53e-01	1.00	3.75e-01	1.00
9.85e-01	1.00	8.66e-01	1.00	4.03e-01	1.00
9.86e-01	1.00	8.90e-01	1.00	4.04e-01	1.00
9.89e-01	1.00	9.12e-01	1.00	4.90e-01	1.00
(a) $\sigma = 10^{-3}$		(b) $\sigma = 10^{-2}$		(c) $\sigma = 10^{-1}$	

smallest eigenvalues and 10 largest eigenvalues of the preconditioned Hessian operator for our Hessian preconditioner with different shifts σ . We can see that most of the eigenvalues of the precondition Hessian with Hessian preconditioner are approximately 1. These observations are consistent with Theorem 7.1.

8.2.2. Partial spectrum with different preconditioners. In this example, we compare the partial spectrum of preconditioned Hessian with the state-of-the-art combined preconditioner [8] and our Hessian preconditioner. To be consistent with [8], we compute the partial spectrum based on the expression of the nonsymmetric preconditioned Hessian (7.1). The Hessian operator (4.4) is discretized in Fourier pseudo-spectral scheme for both cases. Moreover, we apply the combined preconditioner in Fourier pseudo-spectral scheme and the Hessian preconditioner in eighth order finite difference scheme, which is consistent with the preconditioning strategy proposed in section 4. To make fair comparison, we scale the computed eigenvalues so that the largest eigenvalue of both preconditioned Hessian is of the same value for each experiment. Also, we perform an inexact Cholesky factorization with drop tolerance 10^{-3} of the shifted Hessian preconditioner (4.5) with shift $\sigma = 10^{-3}$, permuted by the approximate minimal degree ordering. For case I, we fix $h = \frac{1}{16}$ and vary the domain length L from 4 to 12. For case II, we fix $L = 10$ and vary h from $\frac{1}{4}$ to $\frac{1}{16}$. Tables 8.5 and 8.6 list the 10 smallest scaled eigenvalues and 10 largest scaled eigenvalues of the preconditioned Hessian operator (7.1) for the combined preconditioner and the Hessian preconditioner with different domain $\mathcal{D}$, respectively.

TABLE 8.5
Partial spectrum of preconditioned Hessian with combined preconditioner for case I.

$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$
-1.02e-16	1.00	7.38e-16	1.00	-1.23e-16	1.00
5.09e-17	1.00	3.77e-16	1.00	1.68e-16	1.00
1.25e-03	1.00	1.31e-09	0.98	2.56e-11	0.98
2.18e-03	1.00	3.71e-05	0.98	3.71e-05	0.98
2.70e-03	1.00	3.71e-05	0.98	3.71e-05	0.98
3.48e-03	1.00	2.26e-04	0.98	2.26e-04	0.98
5.18e-03	1.00	3.40e-04	0.98	2.26e-04	0.98
7.27e-03	1.00	4.62e-04	0.98	3.39e-04	0.98
8.22e-03	0.98	7.11e-04	0.98	3.39e-04	0.98
1.06e-02	0.98	7.28e-04	0.98	4.62e-04	0.98
(a) $\mathcal{D} = [-4, 4]^2$		(b) $\mathcal{D} = [-8, 8]^2$		(c) $\mathcal{D} = [-12, 12]^2$	

TABLE 8.6
Partial spectrum of preconditioned Hessian with incomplete Cholesky Hessian preconditioner for case I.

$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$
-2.91e-16	1.00	-1.96e-15	1.00	-1.26e-16	1.00
1.07e-16	1.00	1.40e-16	1.00	1.34e-15	1.00
6.85e-03	1.00	7.02e-09	1.00	1.30e-10	1.00
1.13e-02	1.00	2.05e-04	1.00	1.97e-04	1.00
1.26e-02	1.00	2.23e-04	1.00	2.02e-04	1.00
1.69e-02	1.00	1.33e-03	1.00	1.23e-03	1.00
2.78e-02	1.00	1.38e-03	1.00	1.31e-03	1.00
3.62e-02	1.00	1.80e-03	1.00	1.69e-03	1.00
3.96e-02	1.00	1.92e-03	1.00	1.77e-03	1.00
5.81e-02	1.00	2.77e-03	1.00	2.58e-03	1.00
(a) $\mathcal{D} = [-4, 4]^2$		(b) $\mathcal{D} = [-8, 8]^2$		(c) $\mathcal{D} = [-12, 12]^2$	

Furthermore, Tables 8.7 and 8.8 list the 10 smallest scaled eigenvalues and 10 largest scaled eigenvalues of the preconditioned Hessian operator for the combined preconditioner and Hessian preconditioner with different mesh size h , respectively. From these results, we can see that the conditioning deteriorates for both preconditioners as both the spatial resolution and the size of the domain increase, which is consistent with the observation in [8]. However, the preconditioned Hessian with Hessian preconditioner has a more favorable eigenvalue distribution and smaller condition number compared with the combined preconditioner. More importantly, the nonlinear PCG with the Hessian preconditioner is expected to converge fairly quickly when approaching convergence.

8.3. Numerical experiments in two dimensions. In this section, we apply our method to compute the ground state for some 2D BEC problems with strong repulsive interaction and rotational speed, which are more relevant for real physical problems. We compare our Hessian preconditioner with the state-of-the-art combined preconditioner. The maximum iteration number is set to be 100000. All the experiments are performed on a Ubuntu 22.04 LTS (64 bit) PC-Intel Core i7-4700 CPU 2.40 GHz, 32 GB of DDR3 1600MHz RAM running MATLAB R2022b. Note that the fast energy evaluation and exact line search are used for all experiments in sections 8.3 and 8.4. Our numerical results show that the Hessian preconditioner is more efficient than the combined preconditioner especially for the fast rotating BEC problems.

TABLE 8.7

Partial spectrum of preconditioned Hessian with combined preconditioner for case II.

$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$
-2.66e-17	1.00	-1.38e-17	1.00	-5.37e-17	1
-1.40e-17	1.00	1.99e-17	1.00	-1.55e-17	1
3.58e-04	1.00	1.76e-04	0.99	5.82e-08	0.99
3.58e-04	1.00	1.76e-04	0.99	3.56e-07	0.99
9.42e-04	1.00	4.63e-04	0.98	1.88e-05	0.98
9.75e-04	1.00	5.02e-04	0.98	1.99e-05	0.98
1.40e-03	1.00	5.77e-04	0.98	7.17e-05	0.98
1.78e-03	1.00	8.26e-04	0.98	7.93e-05	0.98
1.78e-03	0.98	8.26e-04	0.98	7.95e-05	0.98
2.75e-03	0.98	9.92e-04	0.98	7.96e-05	0.98
(a) $h = \frac{1}{4}$		(b) $h = \frac{1}{8}$		(c) $h = \frac{1}{16}$	

TABLE 8.8

Partial spectrum of preconditioned Hessian with incomplete Cholesky Hessian preconditioner for case II.

$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$	$\lambda_{\min}$	$\lambda_{\max}$
1.29e-17	1.00	-3.45e-17	1.00	-1.79e-16	1.00
2.76e-17	0.98	2.32e-16	1.00	2.00e-16	1.00
1.21e-01	0.96	1.43e-02	1.00	1.23e-06	1.00
1.30e-01	0.96	1.48e-02	1.00	7.51e-06	1.00
2.28e-01	0.96	3.87e-02	1.00	3.58e-04	1.00
2.28e-01	0.96	4.04e-02	1.00	3.79e-04	1.00
2.31e-01	0.96	6.10e-02	1.00	1.55e-03	1.00
2.31e-01	0.96	6.95e-02	1.00	1.58e-03	1.00
2.31e-01	0.96	7.05e-02	0.98	1.59e-03	1.00
2.31e-01	0.96	8.66e-02	0.98	1.71e-03	1.00
(a) $h = \frac{1}{4}$		(b) $h = \frac{1}{8}$		(c) $h = \frac{1}{16}$	

8.3.1. Example. In this example, $V(\mathbf{x})$ is chosen with $\gamma_x = \gamma_y = 1$, $\alpha = 1.2$, and $\kappa = 0.3$ [8]. The computational domain and mesh sizes are $\mathcal{D} = [-20, 20]^2$ and $h = \frac{1}{32}$. We compute the ground states ϕ_g of rotating BECs with large values of η and Ω . In Table 8.9, η is fixed to be 10000 and Ω is chosen from 1 to 5. In Table 8.10, Ω is fixed to be 5 and η is chosen from 1000 to 20000. Tables 8.9 and 8.10 list the iterations, runtime, and final energy functional our method attain with the combined preconditioner and the Hessian preconditioner, respectively. Also, we underline the lower final energy value obtained by the two preconditioners when there is a significant difference. The contour plots of the density function $|\phi_g(\mathbf{x})|^2$ obtained with the Hessian preconditioner are shown in Figure 8.1. For example, in Table 8.9, when $\eta = 10000, \Omega = 5$, the nonlinear PCG with the combined preconditioner takes 26522 iterations to attain -485.0282069197 in 21126.00 seconds, whereas only 4488 iterations are needed to attain -485.0305526536 in 4681.19 seconds with the Hessian preconditioner. Tables 8.9 and 8.10 show the advantage of our Hessian preconditioner involving Ω over the combined preconditioner that disregards Ω . We can see that with larger values of nonlinearity η and rotating speed Ω , our Hessian preconditioner gains more advantage in runtime. Note that as we mentioned in section 6, the PCG method (local minimization method) cannot guarantee the convergence to the global minimizer of BEC. The choice of the preconditioner also affects the final converged stationary states. From Tables 8.9 and 8.10, we can see that the combined precondi-

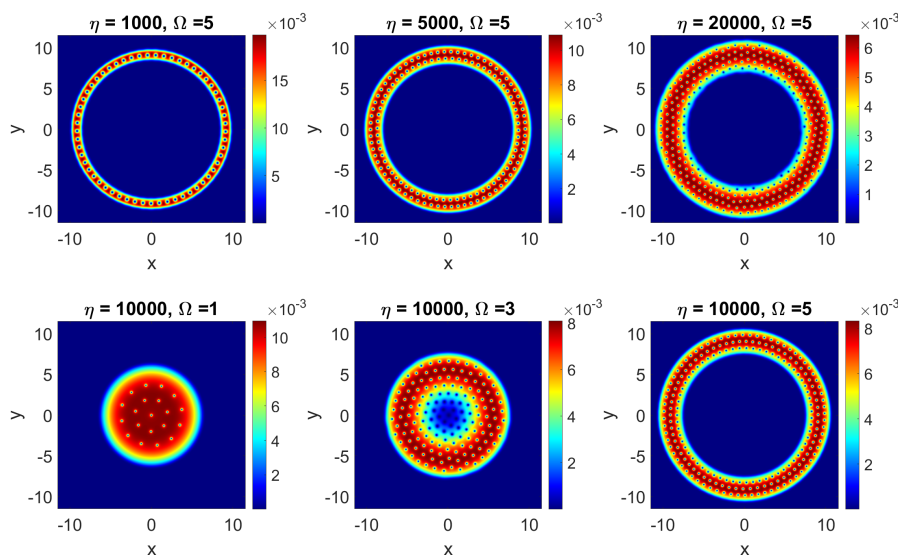


FIG. 8.1. Corresponding contour plots of the density function obtained with the Hessian preconditioner $|\phi_g(\mathbf{x})|^2$ in Tables 8.9 and 8.10.

TABLE 8.9

Performance comparison of PCG with two preconditioners for $\eta = 10000$ and different Ω values.

Ω	PCG iteration		Time (sec)		Final $E_{\eta,\Omega}$	
	Combined	Hessian	Combined	Hessian	Combined	Hessian
1	724	2088	576.51	2052.01	63.02007542539	62.96553732649
1.5	749	697	593.38	583.06	53.26795985753	53.26795985751
2	4929	2443	3885.88	2399.98	37.59961999660	37.59961999657
2.5	5770	3287	4589.90	3137.76	13.63739471900	13.63739471896
3	16435	6226	12885.70	6347.01	-23.48312229660	-23.48295831441
3.5	8653	3612	6895.37	3733.51	-82.54564206625	-82.54564207131
4	25890	6047	20546.26	6430.85	-172.7171085876	-172.7188268092
4.5	18115	3701	14125.19	3868.01	-303.3183033037	-303.3185838060
5	26522	4488	21126.00	4681.19	-485.0282069197	-485.0305526536

tioner achieve lower energies for $\eta = 10000$, $\Omega = 3$, and $\eta = 20000$, $\Omega = 5$. However, the nonlinear PCG with our Hessian preconditioner tends to achieve a lower final energy for most problems.

8.3.2. Example. In this example, we compare the performance of our Hessian preconditioner with the combined preconditioner to solve some more difficult problems. Here, $V(\mathbf{x})$ is chosen with $\gamma_x = 10$, $\gamma_y = 1$, $\alpha = 2$, and $\kappa = 3$. We fix $\eta = 25000$ and vary Ω from 4 to 16. We take $L_x = L_y = 13$, $h = 1/64$. The results are shown in Table 8.11. Figure 8.2 shows the contour plots of the density function $|\phi_g(\mathbf{x})|^2$ obtained with the Hessian preconditioner. Note that the nonlinear PCG with the combined preconditioner does not terminate after 100000 iterations, thus we report the energy and the runtime it attains after the 100000 iterations. More importantly, Table 8.11 shows that our Hessian preconditioner gains significant advantage over the combined preconditioner.

TABLE 8.10
Performance comparison of PCG with two preconditioners for $\Omega = 5$ and different η values.

η	PCG iteration		Time (sec)		Final $E_{\eta,\Omega}$	
	Combined	Hessian	Combined	Hessian	Combined	Hessian
1000	12922	2426	10322.28	2446.36	-522.1631296805	-522.1631296901
2000	10846	2164	8727.81	2192.61	-516.1164313740	-516.1164313839
5000	17673	5349	14009.90	5710.81	-502.4144226059	-502.4145222867
10000	26522	4488	21126.00	4681.19	-485.0282069197	-485.0305526536
20000	60757	6401	47431.78	6956.86	-457.6996232199	-457.6981668537

TABLE 8.11
Performance comparison of PCG with two preconditioners for $\eta = 25000$ and different Ω values.

Ω	PCG iteration		Time (sec)		Final $E_{\eta,\Omega}$	
	Combined	Hessian	Combined	Hessian	Combined	Hessian
4	51864	17454	74290.97	33295.91	141.3951364011	141.3951364033
8	100000+	39510	143848.1+	83809.49	-294.0500897923	-294.0521455922
12	98901	9616	170704.1	20936.99	-1871.149053296	-1871.148855358
16	100000+	30003	168655.4+	68113.02	-5913.255005955	-5913.256644684

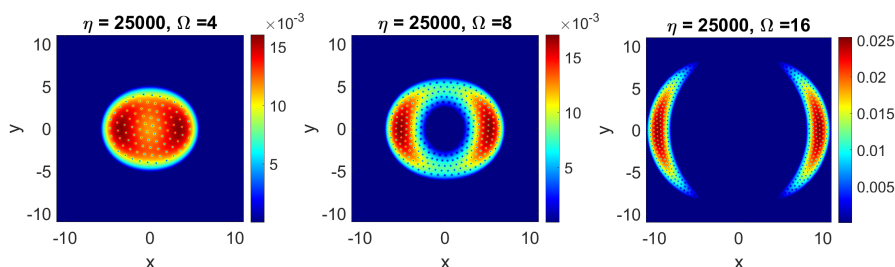


FIG. 8.2. Corresponding contour plots of the density function obtained with the Hessian preconditioner $|\phi_g(\mathbf{x})|^2$ in Table 8.11.

8.4. Numerical experiments in three dimensions. In this section, we apply our method to compute some 3D problems. We perform the 3D experiments on a single node with 16 cores on Clemson Palmetto Cluster running MATLAB R2022a.

In this example, we test four cases: (i) $\eta = 15000$, $\Omega = 4$; (ii) $\eta = 15000$, $\Omega = 5$; (iii) $\eta = 25000$, $\Omega = 4$; (iv) $\eta = 25000$, $\Omega = 6$. The mesh size is $h = \frac{1}{16}$ for all cases. For (i) and (ii), $V(\mathbf{x})$ is chosen with $\gamma_x = \gamma_y = 1, \gamma_z = 1, \alpha = 0.3$, and $\kappa = 1.4$. The computational domain is $\mathcal{D} = [-15, 15]^2 \times [-8, 8]$. For (iii) and (iv), $V(\mathbf{x})$ is chosen with $\gamma_x = \gamma_y = 1, \gamma_z = 3, \alpha = 0.3$, and $\kappa = 1.4$. The computational domain is $\mathcal{D} = [-10, 10]^2 \times [-5, 5]$. We summarize the results in Tables 8.12 and 8.13. Figure 8.3 shows the isosurfaces $|\phi_g(\mathbf{x})|^2 = 10^{-3}$ and surface plots of $|\phi_g(x, y, z = 0)|^2$ obtained with the Hessian preconditioner for all the cases. From these results, we can see that our method works efficiently tackling challenging problems and our Hessian preconditioner is still competitive compared with the combined preconditioner.

9. Conclusions. In this paper, we propose a preconditioned nonlinear CG method in real arithmetic to compute the ground states of the GPE with fast rotation and large nonlinearities that arise in the modeling of BECs. We develop a problem-dependent Hessian preconditioner involving the rotational speed Ω , which is very efficient especially for solving BECs with high nonlinearity and high rotational speeds.

TABLE 8.12
Performance comparison of PCG with two preconditioners for cases I and II.

(η, Ω)	PCG iteration		Time (sec)		Final $E_{\eta, \Omega}$	
	Combined	Hessian	Combined	Hessian	Combined	Hessian
(15000, 4)	14568	3864	378007.4	156913.9	-210.8746065833	-210.8746066226
(15000, 5)	28023	10866	691078.8	448749.6	-529.2941298728	-529.2943293465

TABLE 8.13
Performance comparison of PCG with two preconditioners for cases III and IV.

(η, Ω)	PCG iteration		time (sec)		final $E_{\eta, \Omega}$	
	Combined	Hessian	Combined	Hessian	Combined	Hessian
(25000, 4)	3509	2325	29258.16	23823.73	75.88162274531	75.88162274514
(25000, 6)	16929	7611	140570.9	74431.28	1.258275896279	1.258275895894

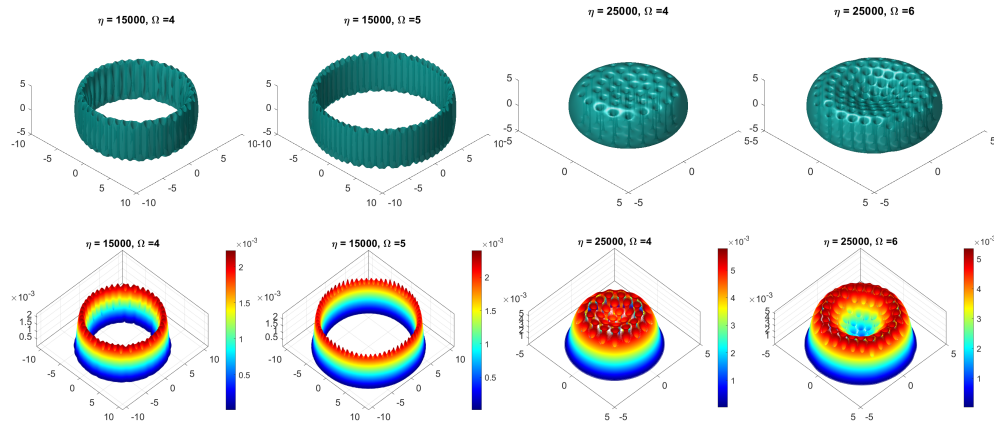


FIG. 8.3. Corresponding isosurface $|\phi_g(\mathbf{x})|^2 = 10^{-3}$ and surface plot of $|\phi_g(x, y, z = 0)|^2$ in Table 8.12 obtained with the Hessian preconditioner.

Also, we provide an efficient method to perform fast energy functional evaluation without repeated computation in the original problem dimension. Exact line search can be enabled by fast energy evaluation at many different step sizes at little extra cost, which tends to result in more rapid and robust convergence compared to inexact line search. Furthermore, our methodologies can be extended to solve other different types of BEC in the future.

Appendix A. Proof of Theorem 4.1.

Proof. The derivation follows from several applications of the product rule and quotient rule. More specifically, we have

$$\frac{\partial E(\phi)}{\partial \phi} = \frac{\partial \frac{\phi^T A \phi}{\phi^T \phi}}{\partial \phi} + \frac{\eta}{2h^d} \frac{\partial \frac{\phi^T B(\phi) \phi}{(\phi^T \phi)^2}}{\partial \phi}.$$

It is easy to obtain

$$(A.1) \quad \frac{\partial \frac{\phi^T A \phi}{\phi^T \phi}}{\partial \phi} = \frac{2}{\phi^T \phi} \left(A\phi - \frac{\phi^T A \phi}{\phi^T \phi} \phi \right)$$

and

$$(A.2) \quad \frac{\partial \frac{\phi^T B(\phi)\phi}{(\phi^T \phi)^2}}{\partial \phi} = \frac{1}{(\phi^T \phi)^2} \left(\frac{\partial \phi^T B(\phi)\phi}{\partial \phi} \right) - 4 \frac{\phi^T B(\phi)\phi}{(\phi^T \phi)^3} \phi.$$

From [34, Theorem 5.1], we have

$$\begin{aligned} \frac{\partial B(\phi)\phi}{\partial \phi} &= B(\phi) + 2\text{diag}(\phi)^2 + 2 \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} \text{diag}(\phi) \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} \text{diag}(\phi) \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} \\ &= B(\phi) + 2 \begin{pmatrix} \text{diag}(\phi_r^2) & \text{diag}(\phi_r \phi_g) \\ \text{diag}(\phi_r \phi_g) & \text{diag}(\phi_g^2) \end{pmatrix}. \end{aligned}$$

Then, it follows that

$$(A.3) \quad \begin{aligned} \frac{\partial \phi^T B(\phi)\phi}{\partial \phi} &= B(\phi)\phi + \left(\frac{\partial B(\phi)\phi}{\partial \phi} \right)^T \phi \\ &= B(\phi)\phi + \left(B(\phi) + 2 \begin{pmatrix} \text{diag}(\phi_r^2) & \text{diag}(\phi_r \phi_g) \\ \text{diag}(\phi_r \phi_g) & \text{diag}(\phi_g^2) \end{pmatrix} \right)^T \phi = 4B(\phi)\phi. \end{aligned}$$

It follows from (A.2),

$$(A.4) \quad \frac{\eta}{2h^d} \frac{\partial \frac{\phi^T B(\phi)\phi}{(\phi^T \phi)^2}}{\partial \phi} = \frac{2\eta}{h^d} \left(\frac{B(\phi)\phi}{(\phi^T \phi)^2} - \frac{\phi^T B(\phi)\phi}{(\phi^T \phi)^3} \phi \right).$$

Combining (A.1) and (A.4), we get

$$(A.5) \quad \begin{aligned} \frac{\partial E(\phi)}{\partial \phi} &= \frac{2}{\phi^T \phi} \left(A\phi - \frac{\phi^T A\phi}{\phi^T \phi} \phi \right) + \frac{2\eta}{h^d} \left(\frac{B(\phi)\phi}{(\phi^T \phi)^2} - \frac{\phi^T B(\phi)\phi}{(\phi^T \phi)^3} \phi \right) \\ &= \frac{2}{\phi^T \phi} \left(A\phi + \eta \frac{B(\phi)\phi}{h^d \phi^T \phi} - \frac{\phi^T A\phi}{\phi^T \phi} \phi - \eta \frac{\phi^T B(\phi)\phi}{h^d (\phi^T \phi)^2} \phi \right) = \frac{2}{\phi^T \phi} (A(\phi)\phi - \lambda(\phi)\phi). \end{aligned}$$

Next, we know that

$$(A.6) \quad \begin{aligned} \frac{\partial^2 E(\phi)}{\partial \phi^2} &= \frac{2}{(\phi^T \phi)^2} \left(\phi^T \phi \frac{\partial (A(\phi)\phi - \lambda(\phi)\phi)}{\partial \phi} - 2(A(\phi)\phi - \lambda(\phi)\phi)\phi^T \right) \\ &= \frac{2}{\phi^T \phi} \left(\frac{\partial (A(\phi)\phi - \lambda(\phi)\phi)}{\partial \phi} - \frac{2(A(\phi)\phi - \lambda(\phi)\phi)\phi^T}{\phi^T \phi} \right) \end{aligned}$$

Again, from [34, Theorem 5.1], we have

$$(A.7) \quad \frac{\partial A(\phi)\phi}{\partial \phi} = A + \frac{\eta}{h^d \phi^T \phi} \left[\begin{pmatrix} \text{diag}(3\phi_r^2 + \phi_g^2) & 2\text{diag}(\phi_r \phi_g) \\ 2\text{diag}(\phi_r \phi_g) & \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - \frac{2}{\phi^T \phi} B(\phi)\phi\phi^T \right].$$

Also, we have

$$(A.8) \quad \begin{aligned} \frac{\partial \lambda(\phi)\phi}{\partial \phi} &= \phi \left(\frac{\partial \lambda(\phi)}{\partial \phi} \right)^T + \lambda(\phi)I \\ &= \frac{2}{\phi^T \phi} \phi \left(\phi^T A + 2\eta \frac{\phi^T B(\phi)}{h^d \phi^T \phi} - \phi^T \frac{\phi^T A\phi}{\phi^T \phi} - 2\eta \phi^T \frac{\phi^T B(\phi)\phi}{h^d (\phi^T \phi)^2} \right) + \lambda(\phi)I. \end{aligned}$$

Combine (A.7) and (A.8), we get

$$\begin{aligned}
 & \frac{\partial(A(\phi)\phi - \lambda(\phi)\phi)}{\partial\phi} \\
 &= A + \frac{\eta}{h^d\phi^T\phi} \left[\begin{pmatrix} \text{diag}(3\phi_r^2 + \phi_g^2) & 2\text{diag}(\phi_r\phi_g) \\ 2\text{diag}(\phi_r\phi_g) & \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - \frac{2}{\phi^T\phi} B(\phi)\phi\phi^T \right] \\
 & \quad - \frac{2}{\phi^T\phi} \phi \left(\phi^T A + 2\eta \frac{\phi^T B(\phi)}{h^d\phi^T\phi} - \phi^T \frac{\phi^T A\phi}{\phi^T\phi} - 2\eta \phi^T \frac{\phi^T B(\phi)\phi}{h^d(\phi^T\phi)^2} \right) - \lambda(\phi)I \\
 &= A + \frac{\eta}{h^d\phi^T\phi} \begin{pmatrix} \text{diag}(3\phi_r^2 + \phi_g^2) & 2\text{diag}(\phi_r\phi_g) \\ 2\text{diag}(\phi_r\phi_g) & \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - \lambda(\phi)I \\
 & \quad - 2\eta \frac{B(\phi)}{h^d\phi^T\phi} \frac{\phi\phi^T}{\phi^T\phi} - 2 \frac{\phi\phi^T}{\phi^T\phi} A - 4\eta \frac{\phi\phi^T}{\phi^T\phi} \frac{B(\phi)}{h^d\phi^T\phi} + 2 \frac{\phi^T A\phi}{\phi^T\phi} \frac{\phi\phi^T}{\phi^T\phi} + 4\eta \frac{\phi\phi^T}{\phi^T\phi} \frac{\phi^T B(\phi)\phi}{h^d(\phi^T\phi)^2}.
 \end{aligned}$$

Also, we have

$$\begin{aligned}
 & \frac{2(A(\phi)\phi - \lambda(\phi)\phi)\phi^T}{\phi^T\phi} \\
 &= \frac{2 \left((A + \frac{\eta}{h^d} \frac{B(\phi)}{\phi^T\phi})\phi - (\frac{\phi^T A\phi}{\phi^T\phi} + \frac{\eta}{h^d} \frac{\phi^T B(\phi)\phi}{(\phi^T\phi)^2})\phi \right) \phi^T}{\phi^T\phi} \\
 &= 2A \frac{\phi\phi^T}{\phi^T\phi} + 2\eta \frac{B(\phi)}{h^d\phi^T\phi} \frac{\phi\phi^T}{\phi^T\phi} - 2 \frac{\phi^T A\phi}{\phi^T\phi} \frac{\phi\phi^T}{\phi^T\phi} - 2\eta \frac{\phi^T B(\phi)\phi}{h^d(\phi^T\phi)^2} \frac{\phi\phi^T}{\phi^T\phi}.
 \end{aligned}$$

Combining (A.9) and (A.10), it follows from (A.6) that

$$\begin{aligned}
 \frac{\partial^2 E(\phi)}{\partial\phi^2} &= \frac{2}{\phi^T\phi} \left\{ A + \frac{\eta}{h^d\phi^T\phi} \begin{pmatrix} \text{diag}(3\phi_r^2 + \phi_g^2) & 2\text{diag}(\phi_r\phi_g) \\ 2\text{diag}(\phi_r\phi_g) & \text{diag}(\phi_r^2 + 3\phi_g^2) \end{pmatrix} - \lambda(\phi)I \right. \\
 & \quad - 2A \frac{\phi\phi^T}{\phi^T\phi} - 2 \frac{\phi\phi^T}{\phi^T\phi} A - 4\eta \frac{B(\phi)}{h^d\phi^T\phi} \frac{\phi\phi^T}{\phi^T\phi} - 4\eta \frac{\phi\phi^T}{\phi^T\phi} \frac{B(\phi)}{h^d\phi^T\phi} \\
 & \quad \left. + 4 \frac{\phi^T A\phi}{\phi^T\phi} \frac{\phi\phi^T}{\phi^T\phi} + 6\eta \frac{\phi\phi^T}{\phi^T\phi} \frac{\phi^T B(\phi)\phi}{h^d(\phi^T\phi)^2} \right\}.
 \end{aligned}$$

Next we will show that ϕ and $\hat{\phi}$ are the eigenvectors of $\frac{\partial^2 E(\phi)}{\partial\phi^2}$ associated with the zero eigenvalue. Note that $B(\phi)\phi = (\frac{\phi_r^3 + \phi_r\phi_g^2}{\phi_r^2\phi_g + \phi_g^3}) \in \mathbb{R}^{2n}$. Then, we have

$$\begin{aligned}
 \frac{\partial^2 E(\phi)}{\partial\phi^2} \phi &= \frac{2}{\phi^T\phi} \left\{ A\phi \frac{\eta}{h^d\phi^T\phi} \begin{pmatrix} 3\phi_r^3 + 3\phi_r\phi_g^2 \\ 3\phi_r^2\phi_g + 3\phi_g^3 \end{pmatrix} - \lambda(\phi)\phi - 2A\phi - 2 \frac{\phi^T A\phi}{\phi^T\phi} \phi \right. \\
 & \quad - \frac{4\eta}{h^d\phi^T\phi} \begin{pmatrix} \phi_r^3 + \phi_r\phi_g^2 \\ \phi_r^2\phi_g + \phi_g^3 \end{pmatrix} - \frac{4\eta}{h^d\phi^T\phi} \frac{\phi^T B(\phi)\phi}{\phi^T\phi} \phi \\
 & \quad \left. + 4 \frac{\phi^T A\phi}{\phi^T\phi} \phi + \frac{6\eta}{h^d\phi^T\phi} \frac{\phi^T B(\phi)\phi}{\phi^T\phi} \phi \right\} \\
 &= \frac{2}{\phi^T\phi} \left\{ -A\phi - \frac{\eta}{h^d\phi^T\phi} \begin{pmatrix} \phi_r^3 + \phi_r\phi_g^2 \\ \phi_r^2\phi_g + \phi_g^3 \end{pmatrix} + 2 \frac{\phi^T A\phi}{\phi^T\phi} \phi \right. \\
 & \quad \left. + \frac{2\eta}{h^d\phi^T\phi} \frac{\phi^T B(\phi)\phi}{\phi^T\phi} \phi - \lambda(\phi)\phi \right\} \\
 &= \frac{2}{\phi^T\phi} \left\{ -A\phi - \frac{\eta}{h^d\phi^T\phi} B(\phi)\phi + \lambda(\phi)\phi \right\} = \frac{2}{\phi^T\phi} \{ -(A(\phi)\phi - \lambda(\phi)\phi) \},
 \end{aligned}$$

which is zero since $\frac{\partial E(\phi)}{\partial \phi} = \frac{2}{\phi^T \phi} (A(\phi)\phi - \lambda(\phi)\phi) = 0$, i.e., ϕ is a stationary point of $E(\phi)$ (local or global minimum, or saddle point). On the other hand, it is easy to see that $\phi^T \hat{\phi} = 0$, then we have

$$(A.11) \quad \phi^T A \hat{\phi} = -\phi_r^T L_s \phi_g + \Omega \phi_r^T L_\omega \phi_r + \Omega \phi_g^T L_\omega \phi_g + \phi_g^T L_s \phi_r = 0,$$

since L_s is symmetric and L_ω is skew-symmetric such that $u^T L_\omega u = 0$ for any $u \in \mathbb{R}^n$. Also, we have

$$(A.12) \quad \phi^T B(\phi) \hat{\phi} = -(\phi_r^3)^T \phi_g - \phi_r^T \phi_g^3 + \phi_g^T \phi_r^3 + (\phi_g^3)^T \phi_r = 0.$$

Combining (A.11) and (A.12), we can easily obtain

$$\begin{aligned} \frac{\partial^2 E}{\partial \phi^2} \hat{\phi} &= \frac{2}{\phi^T \phi} \left\{ A \hat{\phi} + \frac{\eta}{h^d \phi^T \phi} \begin{pmatrix} -\phi_r^2 \phi_g - \phi_g^3 \\ \phi_r \phi_g^2 + \phi_r^3 \end{pmatrix} - \lambda(\phi) \hat{\phi} \right\} \\ &= \frac{2}{\phi^T \phi} \left\{ \begin{pmatrix} -L_s \phi_g + \Omega L_\omega \phi_r \\ L_s \phi_r + \Omega L_\omega \phi_g \end{pmatrix} + \frac{\eta}{h^d \phi^T \phi} \begin{pmatrix} -\phi_r^2 \phi_g - \phi_g^3 \\ \phi_r \phi_g^2 + \phi_r^3 \end{pmatrix} - \lambda(\phi) \begin{pmatrix} -\phi_g \\ \phi_r \end{pmatrix} \right\} \\ &= \frac{2}{\phi^T \phi} \begin{pmatrix} 0 & -I \\ I & 0 \end{pmatrix} \left\{ \begin{pmatrix} L_s \phi_r + \Omega L_\omega \phi_g \\ L_s \phi_g - \Omega L_\omega \phi_r \end{pmatrix} + \frac{\eta}{h^d \phi^T \phi} \begin{pmatrix} \phi_r^3 + \phi_r \phi_g^2 \\ \phi_r^2 \phi_g + \phi_g^3 \end{pmatrix} - \lambda(\phi) \begin{pmatrix} \phi_r \\ \phi_g \end{pmatrix} \right\} \\ &= \begin{pmatrix} 0 & -I \\ I & 0 \end{pmatrix} (A(\phi)\phi - \lambda(\phi)\phi) = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \frac{\partial^2 E(\phi)}{\partial \phi^2} \phi. \end{aligned}$$

Therefore, if ϕ is a stationary point of $E(\phi)$ such that $\frac{\partial^2 E}{\partial \phi^2} \phi = A(\phi)\phi - \lambda(\phi)\phi = 0$, we also have $\frac{\partial^2 E}{\partial \phi^2} \hat{\phi} = 0$. $\square$

Proof of Theorem 7.1.

Proof. First, it is easy to obtain that $P\phi = \phi^T P = 0$. Then we have

$$(A.13) \quad PL^{-1}P \frac{\partial^2 E(\phi)}{\partial \phi^2} PL^{-T}P = 2h^d PL^{-1}PH_p PL^{-T}P.$$

Since $P = I - h^d WW^T$, we have

$$\begin{aligned} (A.14) \quad PL^{-1}PH_p PL^{-T} &= L^{-1}H_p L^{-T} - h^d L^{-1}H_p WW^T L^{-T} - h^d L^{-1}WW^T H_p L^{-T} \\ &\quad + h^{2d} L^{-1}WW^T H_p WW^T L^{-T} - h^d WW^T L^{-1}H_p L^{-T} \\ &\quad + h^{2d} WW^T L^{-1}H_p WW^T L^{-T} + h^{2d} WW^T L^{-1}WW^T H_p L^{-T} \\ &\quad - h^{3d} WW^T L^{-1}WW^T H_p WW^T L^{-T} \end{aligned}$$

and

$$\begin{aligned} (A.15) \quad PL^{-1}PH_p PL^{-T}h^d WW^T &= h^d L^{-1}H_p L^{-T}WW^T - h^{2d} L^{-1}H_p WW^T L^{-T}WW^T \\ &\quad - h^{2d} L^{-1}WW^T H_p L^{-T}WW^T + h^{3d} L^{-1}WW^T H_p WW^T L^{-T}WW^T \\ &\quad - h^{2d} WW^T L^{-1}H_p L^{-T}WW^T + h^{3d} WW^T L^{-1}H_p WW^T L^{-T}WW^T \\ &\quad + h^{3d} WW^T L^{-1}WW^T H_p L^{-T}WW^T - h^{4d} WW^T L^{-1}WW^T H_p WW^T L^{-T}WW^T. \end{aligned}$$

Then, we obtain

$$\begin{aligned}
& PL^{-1}PH_pPL^{-T}P \\
&= L^{-1}H_pL^{-T} + W(h^{2d}W^TL^{-1}H_pWW^TL^{-T} - h^dW^TL^{-1}H_pL^{-T} \\
&\quad + h^{2d}W^TL^{-1}WW^TH_pL^{-T} - h^{3d}W^TL^{-1}WW^TH_pWW^TL^{-T} \\
&\quad + h^{2d}W^TL^{-1}H_pL^{-T}WW^T - h^{3d}W^TL^{-1}H_pWW^TL^{-T}WW^T \\
&\quad - h^{3d}W^TL^{-1}WW^TH_pL^{-T}WW^T + h^{4d}W^TL^{-1}WW^TH_pWW^TL^{-T}WW^T) \\
&\quad + L^{-1}W(h^{2d}W^TH_pWW^TL^{-T} - h^dW^TH_pL^{-T} + h^{2d}W^TH_pL^{-T}WW^T \\
&\quad - h^{3d}W^TH_pWW^TL^{-T}WW^T) + L^{-1}H_pW(h^{2d}W^TL^{-T}WW^T - h^dW^TL^{-T}) \\
&\quad - h^dL^{-1}H_pL^{-T}WW^T.
\end{aligned}$$

Therefore, $PL^{-1}PH_pPL^{-T}P$ is a rank-8 update of $L^{-1}H_pL^{-T}$. Also, W_1 , W_2 , and $W_3 \in \mathbb{R}^{2n \times 2}$ can be obtained based on the above expression, respectively.

On the other hand, we have $L^{-1}H_pL^{-T} = L^{-1}(H_p + \sigma I - \sigma I)L^{-T} = I - \sigma L^{-1}L^{-T}$. Then, $h^dL^{-1}H_pL^{-T}WW^T = h^d(I - \sigma L^{-1}L^{-T})WW^T$. If $\sigma = 0$, then $L^{-1}H_pL^{-T} = I$ and we have

$$\begin{aligned}
& PL^{-1}PH_pPL^{-T}P \\
&= I + W(h^{2d}W^TL^TWW^TL^{-T} - h^dW^TL^TL^{-T} + h^{2d}W^TL^{-1}WW^TL \\
&\quad - h^{3d}W^TL^{-1}WW^TH_pWW^TL^{-T} + h^{2d}W^T - h^{3d}W^TL^{-1}H_pWW^TL^{-T}WW^T \\
&\quad - h^{3d}W^TL^{-1}WW^TLWW^T + h^{4d}W^TL^{-1}WW^TH_pWW^TL^{-T}WW^T - h^dW^T) \\
&\quad + L^{-1}W(h^{2d}W^TH_pWW^TL^{-T} - h^dW^TL + h^{2d}W^TLWW^T \\
&\quad - h^{3d}W^TH_pWW^TL^{-T}WW^T) + L^{-1}H_pW(h^{2d}W^TL^{-T}WW^T - h^dW^TL^{-T}).
\end{aligned}$$

Therefore, $PL^{-1}PH_pPL^{-T}P$ is a rank-6 update of I . $\square$

REFERENCES

- [1] P.-A. ABSIL, R. MAHONY, AND R. SEPULCHRE, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, Princeton, NJ, 2008.
- [2] S. K. ADHIKARI, *Numerical solution of the two-dimensional Gross-Pitaevskii equation for trapped interacting atoms*, Phys. Lett. A, 265 (2000), pp. 91–96.
- [3] P. R. AMESTOY, T. A. DAVIS, AND I. S. DUFF, *An approximate minimum degree ordering algorithm*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 886–905, <https://doi.org/10.1137/S0895479894278952>.
- [4] M. H. ANDERSON, J. R. ENSHER, M. R. MATTHEWS, C. E. WIEMAN, AND E. A. CORNELL, *Observation of Bose-Einstein condensation in a dilute atomic vapor*, Science, 269 (1995), pp. 198–201.
- [5] X. ANTOINE, W. BAO, AND C. BESSE, *Computational methods for the dynamics of the non-linear Schrödinger/Gross-Pitaevskii equations*, Comput. Phys. Commun., 184 (2013), pp. 2621–2633.
- [6] X. ANTOINE AND R. DUBOSCQ, *Robust and efficient preconditioned Krylov spectral solvers for computing the ground states of fast rotating and strongly interacting Bose-Einstein condensates*, J. Comput. Phys., 258 (2014), pp. 509–523.
- [7] X. ANTOINE AND R. DUBOSCQ, *Modeling and computation of Bose-Einstein condensates: Stationary states, nucleation, dynamics, stochasticity*, in Nonlinear Optical and Atomic Systems: At the Interface of Physics and Mathematics, 2015, pp. 49–145.
- [8] X. ANTOINE, A. LEVITT, AND Q. TANG, *Efficient spectral computation of the stationary states of rotating Bose-Einstein condensates by preconditioned nonlinear conjugate gradient methods*, J. Comput. Phys., 343 (2017), pp. 92–109.
- [9] X. ANTOINE, Q. TANG, AND Y. ZHANG, *A preconditioned conjugated gradient method for computing ground states of rotating dipolar Bose-Einstein condensates via kernel*

- truncation method for dipole-dipole interaction evaluation, *Commun. Comput. Phys.*, 24 (2018), pp. 966–988.
- [10] L. ARMILJO, *Minimization of functions having Lipschitz continuous first partial derivatives*, *Pac. J. Math.*, 16 (1966), pp. 1–3.
 - [11] W. BAO AND Y. CAI, *Mathematical theory and numerical methods for Bose-Einstein condensation*, *Kinet. Relat. Mod.*, 6 (2013), pp. 1–135, <https://doi.org/10.3934/krm.2013.6.1>.
 - [12] W. BAO AND Q. DU, *Computing the ground state solution of Bose-Einstein condensates by a normalized gradient flow*, *SIAM J. Sci. Comput.*, 25 (2004), pp. 1674–1697, <https://doi.org/10.1137/S1064827503422956>.
 - [13] W. BAO AND X. RUAN, *Computing ground states of Bose-Einstein condensates with higher order interaction via a regularized density function formulation*, *SIAM J. Sci. Comput.*, 41 (2019), pp. B1284–B1309, <https://doi.org/10.1137/19M1240393>.
 - [14] W. BAO AND W. TANG, *Ground-state solution of Bose-Einstein condensate by directly minimizing the energy functional*, *J. Comput. Phys.*, 187 (2003), pp. 230–254.
 - [15] D. BAYE AND J.-M. SPARENBERG, *Resolution of the Gross-Pitaevskii equation with the imaginary-time method on a Lagrange mesh*, *Phys. Rev. E*, 82 (2010), 056701.
 - [16] C. C. BRADLEY, C. SACKETT, J. TOLLETT, AND R. G. HULET, *Evidence of Bose-Einstein condensation in an atomic gas with attractive interactions*, *Phys. Rev. Lett.*, 75 (1995), 1687.
 - [17] T. BYRNES, K. WEN, AND Y. YAMAMOTO, *Macroscopic quantum computation using Bose-Einstein condensates*, *Phys. Rev. A*, 85 (2012), 040306.
 - [18] M. CALIARI, A. OSTERMANN, S. RAINER, AND M. THALHAMMER, *A minimisation approach for computing the ground state of Gross-Pitaevskii systems*, *J. Comput. Phys.*, 228 (2009), pp. 349–360.
 - [19] M. M. CERIMELE, M. L. CHIOFALO, F. PISTELLA, S. SUCCI, AND M. P. TOSI, *Numerical solution of the Gross-Pitaevskii equation using an explicit finite-difference scheme: An application to trapped Bose-Einstein condensates*, *Phys. Rev. E*, 62 (2000), 1382.
 - [20] M. L. CHIOFALO, S. SUCCI, AND M. TOSI, *Ground state of trapped interacting Bose-Einstein condensates by an explicit imaginary-time algorithm*, *Phys. Rev. E*, 62 (2000), 7438.
 - [21] F. DALFOVO, S. GIORGINI, L. P. PITAEVSKII, AND S. STRINGARI, *Theory of Bose-Einstein condensation in trapped gases*, *Rev. Modern Phys.*, 71 (1999), 463.
 - [22] I. DANAILA AND F. HECHT, *A finite element method with mesh adaptivity for computing vortex states in fast-rotating Bose-Einstein condensates*, *J. Comput. Phys.*, 229 (2010), pp. 6946–6960.
 - [23] I. DANAILA AND P. KAZEMI, *A new Sobolev gradient method for direct minimization of the Gross-Pitaevskii energy with rotation*, *SIAM J. Sci. Comput.*, 32 (2010), pp. 2447–2467, <https://doi.org/10.1137/100782115>.
 - [24] K. B. DAVIS, M.-O. MEWES, M. R. ANDREWS, N. J. VAN DRUTEN, D. S. DURFEE, D. KURN, AND W. KETTERLE, *Bose-Einstein condensation in a gas of sodium atoms*, *Phys. Rev. Lett.*, 75 (1995), 3969.
 - [25] C. M. DION AND E. CANCE, *Ground state of the time-independent Gross-Pitaevskii equation*, *Comput. Phys. Commun.*, 177 (2007), pp. 787–798.
 - [26] A. EDELMAN, T. A. ARIAS, AND S. T. SMITH, *The geometry of algorithms with orthogonality constraints*, *SIAM J. Matrix Anal. Appl.*, 20 (1998), pp. 303–353, <https://doi.org/10.1137/S0895479895290954>.
 - [27] S. C. EISENSTAT AND I. C. IPSEN, *Three absolute perturbation bounds for matrix eigenvalues imply relative bounds*, *SIAM J. Matrix Anal. Appl.*, 20 (1998), pp. 149–158, <https://doi.org/10.1137/S0895479897323282>.
 - [28] A. L. FETTER, B. JACKSON, AND S. STRINGARI, *Rapid rotation of a Bose-Einstein condensate in a harmonic plus quartic trap*, *Phys. Rev. A*, 71 (2005), 013605.
 - [29] D. G. FRIED, T. C. KILLIAN, L. WILLMANN, D. LANDHUIS, S. C. MOSS, D. KLEPPNER, AND T. J. GREYTAK, *Bose-Einstein condensation of atomic hydrogen*, *Phys. Rev. Lett.*, 81 (1998), 3811.
 - [30] I. GEORGESCU, *25 years of BEC*, *Nat. Rev. Phys.*, 2 (2020), pp. 396–396.
 - [31] A. GRIFFIN, D. W. SNOKE, AND S. STRINGARI, *Bose-Einstein Condensation*, Cambridge University Press, Cambridge, 1996.
 - [32] W. W. HAGER AND H. ZHANG, *A survey of nonlinear conjugate gradient methods*, *Pac. J. Optim.*, 2 (2006), pp. 35–58.
 - [33] M. R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, *J. Res. Natl. Inst. Stand.*, 49 (1952), pp. 409–436.

- [34] E. JARLEBRING, S. KVAAL, AND W. MICHIELS, *An inverse iteration method for eigenvalue problems with eigenvector nonlinearities*, SIAM J. Sci. Comput., 36 (2014), pp. A1978–A2001, <https://doi.org/10.1137/130910014>.
- [35] B.-W. JENG, Y.-S. WANG, AND C.-S. CHIEN, *A two-parameter continuation algorithm for vortex pinning in rotating Bose-Einstein condensates*, Comput. Phys. Commun., 184 (2013), pp. 493–508.
- [36] J. KASPRZAK, M. RICHARD, S. KUNDERMANN, A. BAAS, P. JEAMBRUN, J. M. J. KEELING, F. MARCHETTI, M. SZYMAŃSKA, R. ANDRÉ, J. STAEHLI, V. SAVONA, P. B. LITTLEWOOD, B. DEVEAUD, AND L. S. DANG, *Bose-Einstein condensation of exciton polaritons*, Nature, 443 (2006), pp. 409–414.
- [37] A. KNYAZEV, *A preconditioned conjugate gradient method for eigenvalue problems and its implementation in a subspace*, in Numerical Treatment of Eigenvalue Problems Vol. 5/Numerische Behandlung von Eigenwertaufgaben Band 5: Workshop in Oberwolfach, 1990/Tagung in Oberwolfach, 1990, Springer, 1991, pp. 143–154.
- [38] A. V. KNYAZEV, *Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method*, SIAM J. Sci. Comput., 23 (2001), pp. 517–541, <https://doi.org/10.1137/S1064827500366124>.
- [39] K. KREUTZ-DELGADO, *The Complex Gradient Operator and the CR-Calculus*, preprint, <https://arxiv.org/abs/0906.4835>, 2009.
- [40] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer-Verlag, New York, 1999.
- [41] W. RUDIN, ET AL., *Principles of Mathematical Analysis*, Vol. 3, McGraw-Hill New York, 1976.
- [42] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.
- [43] G. L. SLEIJPEN AND H. A. VAN DER VORST, *A Jacobi–Davidson iteration method for linear eigenvalue problems*, SIAM Rev., 42 (2000), pp. 267–293, <https://doi.org/10.1137/S0036144599363084>.
- [44] J. C. STRIKWERDA, *Finite Difference Schemes and Partial Differential Equations*, SIAM, Philadelphia, 2004.
- [45] D. SZYLD AND F. XUE, *Preconditioned eigensolvers for large-scale nonlinear Hermitian eigenproblems with variational characterizations. I. Extreme eigenvalues*, Math. Comp., 85 (2016), pp. 2887–2918.
- [46] T. TIAN, Y. CAI, X. WU, AND Z. WEN, *Ground states of spin-F Bose-Einstein condensates*, SIAM J. Sci. Comput., 42 (2020), pp. B983–B1013, <https://doi.org/10.1137/19M1271117>.
- [47] Y.-S. WANG, B.-W. JENG, AND C.-S. CHIEN, *A two-parameter continuation method for rotating two-component Bose-Einstein condensates in optical lattices*, Commun. Comput. Phys., 13 (2013), pp. 442–460.
- [48] X. WU, Z. WEN, AND W. BAO, *A regularized Newton method for computing ground states of Bose-Einstein condensates*, J. Sci. Comput., 73 (2017), pp. 303–329.
- [49] R. ZENG AND Y. ZHANG, *Efficiently computing vortex lattices in rapid rotating Bose-Einstein condensates*, Comput. Phys. Commun., 180 (2009), pp. 854–860.