GSGPEs: a MATLAB code for computing the ground state of systems of Gross–Pitaevskii equations

Marco Caliari[a], Stefan Rainer[b,*]

[a]*Dipartimento di Informatica, Università degli Studi di Verona, Strada Le Grazie 15, I–37134 Verona, Italy.*
[b]*Institut für Mathematik, Leopold–Franzens Universität Innsbruck, Technikerstraße 13/7, A–6020 Innsbruck, Austria.*

Abstract

GSGPEs is a Matlab/GNU Octave suite of programs for the computation of the ground state of systems of Gross–Pitaevskii equations. It can compute the ground state in the defocusing case, for any number of equations with harmonic or quasi-harmonic trapping potentials, in spatial dimension one, two or three. The computation is based on a spectral decomposition of the solution into Hermite functions and direct minimization of the energy functional through a Newton-like method with an approximate line-search strategy.

Keywords: Nonlinear Schrödinger equations, Gross–Pitaevskii equations, Bose–Einstein condensation, Ground state, Spectral methods, Minimization

PROGRAM SUMMARY

Program Title: GSGPEs
Journal Reference:
Catalogue identifier:
Licensing provisions: none.
Programming language: Matlab/GNU Octave.
RAM: about 100 MB for a single three-dimensional equation (test run output).
Keywords: Gross–Pitaevskii equations, ground state.
Classification: 2.7, 4.9.
Nature of problem: A system of Gross–Pitaevskii Equations (GPEs) is used to

---

*Corresponding author.
E-mail address: stefan.rainer@uibk.ac.at

mathematically model a Bose–Einstein Condensate (BEC) for a mixture of different interacting atomic species. The equations can be used both to compute the ground state solution (i.e., the stationary order parameter that minimizes the energy functional) and to simulate the dynamics. For particular shapes of the traps, three-dimensional BECs can be also simulated by lower dimensional GPEs.

Solution method: The ground state of a system of Gross–Pitaevskii equations is computed through a spectral decomposition into Hermite functions and the direct minimization of the energy functional.

Running time: about 30 seconds for a single three-dimensional equation with d.o.f. 40 for each spatial direction (test run output).

1. Introduction

The first experimental realization of Bose–Einstein condensate for atomic gases [1] in 1995 gave rise to various theoretical and numerical investigations on the equation modeling these phenomena, that is the Gross–Pitaevskii Equation

$$
\begin{cases}
i\hbar \partial_t \Psi(x,t) = \left(-\dfrac{\hbar^2}{2m}\Delta + V(x) + \hbar^2 g |\Psi(x,t)|^2 \right) \Psi(x,t), \\
\|\Psi(\cdot,0)\|_{L^2}^2 = N
\end{cases}
\tag{1}
$$

describing the order parameter $\Psi \colon \mathbb{R}^3 \times \mathbb{R}_{\geq 0} \to \mathbb{C}$ of an atomic species with mass m, N being the number of atoms. The atoms are confined in a trapping potential $V \colon \mathbb{R}^3 \to \mathbb{R}$ and g is the coupling constant defined by $\frac{4\pi\sigma}{m}$, where σ is the scattering length. We observe that the normalization condition can also take the form

$$\|\Psi(\cdot,0)\|_{L^2}^2 = 1$$

leading to a different definition of g (see, for instance, [2, 3, 4]). Few years later, in 1997, Bose–Einstein condensation for a mixture of two different interacting atomic species was firstly realized at JILA [5], exhibiting a partial overlap between the wave functions. More recently, around 2003, triplet species states were observed in [6]. The vector nature of the order parameter exhibits some new structures and dynamics that are absent in the single component case. This, again, stimulated various succeeding studies of numerical and theoretical nature [7, 8, 9, 10].

In this paper, we present a suite of programs for the computation of the ground state of a system of ℓ Gross–Pitaevskii equations (or Vector Gross–

Pitaevskii Equation, VGPE)

$$\begin{cases} i\hbar\partial_t \Psi^{(l)} = \left(-\frac{\hbar^2}{2m_l}\Delta + V_l + \hbar^2\sum_{k=1}^{\ell} g_{lk}|\Psi^{(k)}|^2 \right)\Psi^{(l)}, \\ \left\|\Psi^{(l)}\right\|_{L^2}^2 = N_l, \quad l = 1,\dots,\ell \end{cases} \tag{2}$$

that is the order parameters $\left(\Psi^{(1)}(x,t),\dots,\Psi^{(\ell)}(x,t)\right)$, where $\Psi^{(l)}(x,t) = \Phi^{(l)}(x)\mathrm{e}^{-\mathrm{i}\mu_l t}$, of smallest energy

$$\mathcal{E}(\Psi) = \mathcal{E}(\Phi) = \sum_{l=1}^{\ell}\left(\left(-\frac{\hbar^2}{2m_l}\Delta + V_l + \frac{\hbar^2}{2}\sum_{k=1}^{\ell} g_{lk}|\Phi^{(k)}|^2\right)\Phi^{(l)}\Bigg|\Phi^{(l)}\right)_{L^2}$$

with the *chemical potential* μ_l defined by

$$\mu_l N_l = \left(\left(-\frac{\hbar^2}{2m_l}\Delta + V_l + \hbar^2\sum_{k=1}^{\ell} g_{lk}|\Phi^{(k)}|^2\right)\Phi^{(l)}\Bigg|\Phi^{(l)}\right)_{L^2}.$$

We recall that there is a unique ground state solution with $\Phi^{(l)}(x)$ real-valued and non-negative for $l = 1,\dots,\ell$ (see [7]). In (2), $\Psi^{(l)}$ are the order parameters of atomic species with masses m_l. The number of atoms for each species is N_l. We call g_{ll} *intra-species* coupling constants and $g_{lk} = g_{kl}$, $l \neq k$ *inter-species* coupling constants; g_{lk} equals $2\pi\sigma_{lk}\frac{m_l+m_k}{m_l m_k}$, where σ_{lk} is the scattering length for the l-k species. We restrict ourselves to the defocusing case $\sigma_{lk} \geq 0$. Since for many situations the trapping potentials V_l are harmonic or quasi-harmonic, we suppose that

$$V_l = V_l(x) = V_l(x_1, x_2, x_3) = \frac{m_l}{2}\sum_{i=1}^{3}\omega_{li}^2 x_i^2 + V_l^\varepsilon(x),$$

where $V^\varepsilon(x)$ is a "small" correction to a *standard* harmonic potential.

The suite is able to compute the ground state in the full three-dimensional environment, as well as in the lower dimensional cases, for any number ℓ of components.

The method is based on a spectral decomposition of the solution into Hermite functions and a Newton-like method for constrained minimization.

2. Gross–Pitaevskii system of equations

The three-dimensional Gross–Pitaevskii system can be approximately reduced to a two-dimensional system (disk-shaped condensation) or even to a one-dimensional system (cigar-shaped condensation), depending on the ratio between the trap frequencies ω_{li}, see [11, 2, 4]. In these cases, the systems are formally equivalent to (2), with a different definition of the coupling constants g_{lk}. For this reason, we will consider from now on a general d-dimensional system of Gross–Pitaevskii equations, where $d \in \{1, 2, 3\}$.

By the linear transformation defined by

$$\tau = \omega t \quad \xi_i = \sqrt{c_i} x_i, \quad \psi^{(l)} = \sqrt[4]{c_1 \cdot \ldots \cdot c_d}\, \Psi^{(l)}, \tag{3}$$

where

$$c_i = \frac{m \omega_i}{\hbar}, \quad m = \sqrt[\ell]{m_1 \cdot \ldots \cdot m_\ell}$$

and ω and ω_i are free parameters, system (2) takes the form

$$\begin{cases} \mathrm{i}\partial_\tau \psi^{(l)} = \left(-\frac{m}{2m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \partial_{\xi_j}^2 + U_l + \sum_{k=1}^{\ell} \vartheta_{lk} |\psi^{(k)}|^2 \right) \psi^{(l)}, \\ \left\| \psi^{(l)} \right\|_{L^2}^2 = N_l, \quad l = 1, \ldots, \ell \end{cases} \tag{4}$$

with

$$U_l = U_l(\xi) = U_l(\xi_1, \ldots, \xi_d) = \frac{1}{\omega \hbar} V_l \left(\frac{\xi_1}{\sqrt{c_1}}, \ldots, \frac{\xi_d}{\sqrt{c_d}} \right)$$

and

$$\vartheta_{lk} = \frac{\hbar}{\omega} g_{lk} \sqrt{c_1 \cdot \ldots \cdot c_d}.$$

The ground state $\psi = (\psi^{(1)}, \ldots, \psi^{(\ell)})$ of GPEs system (4)

$$\psi^{(l)}(\xi, t) = \mathrm{e}^{-\mathrm{i}\nu_l t} \varphi^{(l)}(\xi), \quad l = 1, \ldots, \ell$$

minimizes the energy functional

$$E(\varphi) = \sum_{l=1}^{\ell} \left(\left(-\frac{m}{2m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \partial_{\xi_i}^2 + U_l + \frac{1}{2} \sum_{k=1}^{\ell} \vartheta_{lk} |\varphi^{(k)}|^2 \right) \varphi^{(l)} \middle| \varphi^{(l)} \right)_{L^2} \tag{5}$$

and the corresponding chemical potentials ν_l are

$$\nu_l N_l = \left(\left(-\frac{m}{2m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \partial_{\xi_i}^2 + U_l + \sum_{k=1}^{\ell} \vartheta_{lk} |\varphi^{(k)}|^2 \right) \varphi^{(l)} \middle| \varphi^{(l)} \right)_{L^2} = \frac{\mu_l N_l}{\omega \hbar}. \tag{6}$$

4

We observe that the relations with the corresponding physical quantities are

$$E(\varphi) = \frac{\mathcal{E}(\Phi)}{\omega\hbar}, \quad \nu_l = \frac{\mu_l}{\omega\hbar}.$$

Several different examples can be found in literature, about the choice of the free parameters ω and ω_i. For instance, in the single-component case, $\omega = \omega_i = \omega_{11}$, $i = 1, \ldots, d$ is used in [2, 3] (and also in the multi-component case [7]) and $\omega = \omega_i = \min_i\{\omega_{1i}\}$ in [11]. Another possible choice is $\omega = \omega_d$ and $\omega_i = \omega_{1i}$, see [4].

Together with the constraints

$$G_l(\varphi) = 1 - \frac{\left\|\varphi^{(l)}\right\|_{L^2}^2}{N_l} = 0, \quad l = 1, \ldots, \ell, \tag{7}$$

in order to compute the ground state solution it is possible to minimize the Lagrange function

$$L(\varphi, \eta) = E(\varphi) + \sum_{l=1}^{\ell} \eta_l G_l(\varphi),$$

where $\eta = (\eta_1, \ldots, \eta_\ell)$ is the Lagrange multiplier, as suggested in [2]. Since the local minima of L are solutions of $\nabla L(\varphi, \eta) = 0$, we obtain, for $l = 1, \ldots, \ell$,

$$\frac{\eta_l}{N_l} \varphi^{(l)} = \left(-\frac{m}{2m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \partial_{\xi_i}^2 + U_l + \sum_{k=1}^{\ell} \vartheta_{lk} |\varphi^{(k)}|^2 \right) \varphi^{(l)}$$

and, multiplying by $\varphi^{(l)}$ and integrating over $\mathbb{R}^d$ and comparing with (6), we get that the chemical potential ν_l coincides with the Lagrange multiplier η_l divided by N_l.

2.1. Hermite spectral decomposition

When the trapping potential is close to a harmonic one, in order to discretize the minimization problem it is convenient to use a spectral approximation based on the eigenfunctions of the quantum harmonic oscillator, i.e. the Hermite functions. We rewrite (4) as

$$\begin{cases} i\partial_\tau \psi^{(l)} = \left(\frac{m}{m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \left(-\frac{\partial_{\xi_i}^2}{2} + \frac{\gamma_i^4}{2} \xi_i^2 \right) + D_l + \sum_{k=1}^{\ell} \vartheta_{lk} |\psi^{(k)}|^2 \right) \psi^{(l)}, \\ \left\|\psi^{(l)}\right\|_{L^2}^2 = N_l, \quad l = 1, \ldots, \ell \end{cases} \tag{8}$$

where

$$D_l = D_l(\xi_1, \ldots, \xi_d) = -\frac{m}{m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \frac{\gamma_i^4}{2} \xi_i^2 + U_l(\xi_1, \ldots, \xi_d).$$

The spectral decomposition is based on the functions $\mathcal{H}_j^\gamma(\xi)$ defined by

$$\mathcal{H}_j^\gamma(\xi) = H_{j_1}^{\gamma_1}(\xi_1) \cdot \ldots \cdot H_{j_d}^{\gamma_d}(\xi_d) \mathrm{e}^{-\frac{1}{2}(\gamma_1^2 \xi_1^2 + \ldots + \gamma_d^2 \xi_d^2)},$$

where $H_{j_i}^{\gamma_i}(\xi_i)$ are the univariate Hermite polynomials normalized in such a way that $\{\mathcal{H}_j^\gamma(\xi)\}_j$ are orthonormal with respect to the $L^2(\mathbb{R}^d)$ scalar product (see, for instance, [10]). Moreover they satisfy

$$\left(\sum_{i=1}^{d} \frac{\omega_i}{\omega} \left(-\frac{\partial_{\xi_i}^2}{2} + \frac{\gamma_i^4 \xi_i^2}{2} \right) \right) \mathcal{H}_j^\gamma(\xi) = \lambda_j \mathcal{H}_j^\gamma(\xi) \tag{9}$$

where

$$\lambda_j = \sum_{i=1}^{d} \frac{\omega_i}{\omega} \gamma_i^2 \left(\frac{1}{2} + j_i \right).$$

By the decomposition of $\varphi^{(l)}$ into a common basis of Hermite functions

$$\varphi^{(l)}(\xi) = \sum_j \varphi_j^{(l)} \mathcal{H}_j^\gamma(\xi) = \sum_{j_1} \cdots \sum_{j_d} \varphi_{j_1 \ldots j_d}^{(l)} \mathcal{H}_{j_1}^{\gamma_1}(\xi_1) \cdot \ldots \cdot \mathcal{H}_{j_d}^{\gamma_d}(\xi_d)$$

in (5) and (7) and truncating the infinite sums, we get

$$E(\varphi) = \sum_{l=1}^{\ell} \left[\frac{m}{m_l} \sum_{j=0}^{J-1} \lambda_j \left(\varphi_j^{(l)} \right)^2 + \right.$$

$$+ \int_{\mathbb{R}^d} D_l(\xi) \left(\sum_{j=0}^{J-1} \varphi_j^{(l)} \mathcal{H}_j^\gamma(\xi) \right)^2 \mathrm{d}\xi +$$

$$\left. + \frac{1}{2} \sum_{k=1}^{\ell} \vartheta_{lk} \int_{\mathbb{R}^d} \left(\sum_{j=0}^{J-1} \varphi_j^{(k)} \mathcal{H}_j^\gamma(\xi) \right)^2 \left(\sum_{j=0}^{J-1} \varphi_j^{(l)} \mathcal{H}_j^\gamma(\xi) \right)^2 \mathrm{d}\xi \right] \to \min \tag{10}$$

$$G_l(\varphi) = 1 - \frac{1}{N_l} \sum_{j=0}^{J-1} \left(\varphi_j^{(l)} \right)^2 = 0$$

where the notation

$$\sum_{j=0}^{J-1} = \sum_{j_1=0}^{J_1-1} \cdots \sum_{j_d=0}^{J_d-1}$$

is used. Given the truncation to J_i coefficients for the spectral representation along the direction i of $\varphi^{(l)}$, in order to be able to compute exactly the integral corresponding to $|\varphi^{(k)}|^2|\varphi^{(l)}|^2$ in the energy functional in (10), it is necessary to apply a Gaussian quadrature formula relative to the weight $\mathrm{e}^{-2\gamma_i\xi_i^2}$ with $2J_i - 1$ nodes (polynomial exactness up to degree $4J_i - 3$). In this way, it is not possible to exactly compute the integral relative to the trap potential $D_l(\xi)$. Anyway, for a general potential $V_l(x)$ it is not possible to have and exact quadrature rule. On the other hand, if $V_l(x)$ is harmonic, an integral of type

$$\int_{\mathbb{R}} \xi_i^2 p_{2J_i-2}(\xi_i) \mathrm{e}^{-\gamma_i^2\xi_i^2}\, \mathrm{d}\xi_i = \int_{\mathbb{R}} \xi_i^2 p_{2J_i-2}(\xi_i) \mathrm{e}^{\gamma_i^2\xi_i^2} \mathrm{e}^{-2\gamma_i^2\xi_i^2}\, \mathrm{d}\xi_i =$$

$$= \int_{\mathbb{R}} \xi_i^2 p_{2J_i-2}(\xi_i) \sum_{k=0}^{J_i-2} \frac{(\gamma_i^2\xi_i^2)^k}{k!} \mathrm{e}^{-2\gamma_i^2\xi_i^2}\, \mathrm{d}\xi_i +$$

$$+ \int_{\mathbb{R}} \xi_i^2 p_{2J_i-2}(\xi_i) \sum_{k=J_i-1}^{\infty} \frac{(\gamma_i^2\xi_i^2)^k}{k!} \mathrm{e}^{-2\gamma_i^2\xi_i^2}\, \mathrm{d}\xi_i$$

has to be computed, where $p_{2J_i-2}(\xi)$ denotes a polynomial of degree $2J_i - 2$. With our choice of quadrature nodes and weights, the first part is exactly computed (because of degree $4J_i - 4$) and the second is $\mathcal{O}(1/(J_i - 1)!)$. Quadrature nodes and weights are computed by the Matlab tool provided in [12].

The choice of the scale γ is quite important. As suggested in [13], if the support of the solution is contained in $S = [-M_1, M_1] \times \ldots \times [-M_d, M_d]$, then, given the degrees of approximation $J_1, \ldots, J_d$, the scale should be chosen in such a way that the convex hull of the quadrature nodes is S. Of course, we are improperly using the term "support" as the region where the solution is *mainly* concentrated, being exponentially decaying outside this region. A rough estimate of S can be obtained when the intra-species coupling constants are large through the Thomas–Fermi approximation, for which the support is really a compact set. In this way, since the maximum of the quadrature nodes along direction i behaves like $\sqrt{2J_i - 1}/\gamma_i$, it is possible to choose a quite optimal value for γ. On the other hand, when the potentials

are standard harmonic and the coupling constants are small, it is possible to choose γ in order to make vanishing the sum of the D_l which appears in the energy functional. In fact, we have

$$\sum_{l=1}^{\ell} D_l(\xi) = \sum_{l=1}^{\ell} \left(-\frac{m}{m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \frac{\gamma_i^4}{2} \xi_i^2 + U_l(\xi) \right) =$$

$$= \sum_{l=1}^{\ell} \left(-\frac{m}{m_l} \sum_{i=1}^{d} \frac{\omega_i}{\omega} \frac{\gamma_i^4}{2} \xi_i^2 + \frac{m_l}{m} \sum_{i=1}^{d} \frac{\omega_{li}^2}{\omega \omega_i} \frac{\xi_i^2}{2} \right) =$$

$$= \sum_{i=1}^{d} \frac{\omega_i}{\omega} \frac{\xi_i^2}{2} \left(-\gamma_i^4 \sum_{l=1}^{\ell} \frac{m}{m_l} + \sum_{l=1}^{\ell} \frac{m_l}{m} \frac{\omega_{li}^2}{\omega_i^2} \right)$$

which is zero if

$$\gamma_i = \sqrt[4]{\frac{\sum_{l=1}^{\ell} \frac{m_l}{m} \frac{\omega_{li}^2}{\omega_i^2}}{\sum_{l=1}^{\ell} \frac{m}{m_l}}} \tag{11}$$

This choice, in case of a single component, corresponds to using the eigenfunctions of the harmonic oscillator, with

$$\gamma_i = \sqrt{\frac{\omega_{1i}}{\omega_i}}$$

and it is our *initial* suggested choice for γ.

3. Constrained minimization

In order to describe the method used to minimize the energy, we restrict ourselves to the case of one single component, the extension to more than one species being straight forward. The constrained minimization problem (10) takes the form

$$\begin{aligned} E(x) &\rightarrow \min, \\ G(x) &= 0. \end{aligned} \quad x \in \mathbb{R}^{J_1 \times \dots \times J_d} \tag{12}$$

For the numerical solution of (12) we use a Newton-like method based on SQP (Sequential Quadratic Programming). A detailed description of the SQP method can be found, e.g., in [14, 15].

Starting from a given approximation $x^{(k)}$ to the minimizer $x^\star$, we consider the quadratic minimization problem $Q\left(x^{(k)}, H^{(k)}\right)$

$$\nabla E\left(x^{(k)}\right)^{\mathrm{T}} s + \frac{1}{2} s^{\mathrm{T}} H^{(k)} s \to \min \qquad (13a)$$

subject to the linearized constraint

$$G\left(x^{(k)}\right) + \nabla G\left(x^{(k)}\right)^{\mathrm{T}} s = 0, \quad i = 1, \dots, \ell. \qquad (13b)$$

The symmetric positive definite matrix $H^{(k)}$ denotes an approximation to the Hessian of the Lagrangian $L(x^{(k)}, y^{(k)}) = E\left(x^{(k)}\right) + y^{(k)} G\left(x^{(k)}\right)$, $y^{(k)} \in \mathbb{R}$ being the Lagrange multiplier. In order to derive a globally convergent method we need a *merit function*, that is an additional function that measures the progress from one iterate $x^{(k)}$ to the next one $x^{(k+1)}$ towards the solution $x^\star$. Among the several possibilities, we take the so called *exact penalty function*

$$P(x) = E(x) + r\left|G(x)\right|, \qquad (14)$$

with an appropriate parameter $r > 0$. In [16] Han showed that the solution $s^{(k)}$ of (13) is a descent direction of (14), if $H^{(k)}$ is positive definite and the corresponding Lagrange multiplier $y^{(k)}$ is bounded by r. In order to find an appropriate step length λ_k satisfying

$$P\left(x^{(k)} + \lambda_k s^{(k)}\right) < P\left(x^{(k)}\right),$$

we employ a backtracking line-search strategy as described in [17] and define the new approximation $x^{(k+1)}$ by

$$x^{(k+1)} = x^{(k)} + \lambda_k s^{(k)}.$$

The convergence properties of the algorithm can be found in [10], where it is shown that, under mild assumptions on E, on the sequence $H^{(k)}$ and r the method is globally convergent.

In order to apply the just described method to our constrained minimization problem we have to specify the choice of $H^{(k)}$. The standard choice $H^{(k)} = \nabla^2_{xx} L\left(x^{(k)}, y^{(k)}\right)$ is, from a computational point of view, very unattractive because it leads to the solution of full and, especially in higher dimensions, large linear systems, which are the bottlenecks of such methods. In order to end up with a sparse matrix we take only the main diagonal of

$\nabla_{xx}^2 L\left(x^{(k)}, y^{(k)}\right)$. In principle, the matrix could be not a positive definite matrix and should be modified, taking the absolute values of the entries and replacing all values whose magnitude are smaller than a prescribed threshold by 1, in order to prevent the matrix to become close to singular. Finally, it is possible to scale the matrix in order to avoid undesired large steps resulting in a large number of reductions in the line-search procedure. Let us mention that our numerical experiments showed that the modifications above were never needed, the values remained positive and bounded away form zero and the step length were always reasonable.

We have not specified yet how we choose appropriate starting values for the minimization procedure. To this end let us recall that for a standard harmonic potential and zero coupling constant we can express the ground state exactly using the Hermite spectral decomposition. Hence the idea is to start with this setting and to solve several related problems, using a standard continuation method (see, e.g., [18]), to reach the desired parameters. Let us describe this in more detail. Instead of considering problem (1), we consider

$$\begin{cases} i\partial_\tau \psi = \left(\sum_{i=1}^d \frac{\omega_i}{\omega}\left(-\frac{\partial_{\xi_i}^2}{2} + \frac{\gamma_i^4}{2}\xi_i^2\right) + \hat{\rho}D + \hat{\vartheta}|\psi|^2\right)\psi, \\ \|\psi\|_{L^2}^2 = N \end{cases}$$

where $\hat{\rho}$ and $\hat{\vartheta}$ are used as continuation parameters. We start with $\hat{\rho} = \hat{\vartheta} = 0$ and increase these values step by step until $\hat{\rho} = 1$ and $\hat{\vartheta} = \vartheta$. For the case $\hat{\rho} = \hat{\vartheta} = 0$ we know the exact solution, hence no starting values are needed; for the other problems we use as starting values the ground state of the previous problem. The error estimate is given by the difference of the chemical potential and the Lagrange multipliers divided by N.

These considerations allow us to minimize the energy functional in a fast and efficient way.

4. Description of the programs

4.1. Common functions

4.1.1. init

This function precomputes the initial values for the solution, the continuation parameters and the Hermite functions and corresponding quadrature nodes and weights.

The initial value for $\varphi_l(\xi)$ is $\mathcal{H}_0^\gamma(\xi)$, which, in the single component case with the suggested values for γ (11), corresponds to the the ground state solution in the linear case with standard harmonic potential. For the same reason, the initial value for η_l is the chemical potential $\sum_{i=1}^d \frac{\omega_{li}}{2\omega}$ multiplied by N_l. We use

$$\frac{N_l}{N_l + \int_{\mathbb{R}^d} D_l(\xi) \left(\mathcal{H}_0^\gamma(\xi)\right)^2 \mathrm{d}\xi}.$$

as the initial value for the continuation parameter $\hat{\rho}_l$. This choice depends on the given and the standard harmonic potential. If the potentials are very close to each other, the corresponding initial value for ρ_l is chosen close to 1. Otherwise, it is smaller. As initial value for the coupling constants $\hat{\theta}_{lk}$ we take

$$\min\left\{\theta_{lk}, \frac{1}{N_k}\right\}.$$

4.1.2. groundstate

This function computes the ground state of a system of Gross–Pitaevskii equations, through a continuation technique, using the minimization approach described in Section 3. In particular, it sets the tolerance and the maximum number of iterations.

4.1.3. minimize

This function computes the ground state for the actual setting of the continuation parameters using the method of SQP.

4.1.4. linesearch

This function computes the step length for a given search direction in order to reduce the merit function.

4.1.5. newton

This function computes the minimum for a constrained quadratic minimization problem. In particular, it computes the search direction for the actual step in the method of SQP.

4.1.6. jacobian

Computes the gradient of the Lagrange function and the gradient of the exact penalty function.

4.1.7. linsolv

This function solves the linear systems arising during the application of the modified Newton method in an efficient way.

4.1.8. getscales

This function allows to choose the different ways to normalize the equations described in Section 2. Moreover, it sets the value of γ, according to the suggested value (11).

4.2. Functions for the three-dimensional case

4.2.1. GSGPEs3d

This is a driver function to compute the ground state of the full three-dimensional Gross–Pitaevskii equation

$$\begin{cases} i\hbar\partial_t\Psi(x,t) = \left(-\dfrac{\hbar^2}{2m}\Delta + V(x) + \hbar^2 g|\Psi(x,t)|^2\right)\Psi(x,t), \\ \|\Psi(\cdot,0)\|_{L^2}^2 = N \end{cases}$$

where $m = 1.44 \cdot 10^{-25}$ kg is the atomic mass of ^{87}Rb, $N = 10^4$, $V(x)$ is the standard harmonic potential

$$V(x) = \frac{m}{2}(\omega_1^2 x_1^2 + \omega_2^2 x_2^2 + \omega_3^2 x_3^2), \quad \omega_1 = \omega_2 = \omega_3/\sqrt{8} = 2\pi \times 90 \text{ Hz}$$

and $g = 4\pi\sigma/m$, $\sigma = 5.62 \cdot 10^{-9}$ m. With these choices, the driver reproduces the test run output in [4], taking into account of the different normalization of the L^2 norm to N instead of 1. The number of Hermite functions is set to 40 for each spatial dimension. The values of the parameters and the definition of the potential are set inside the function as follows:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% USER DEFINED PARAMETERS %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Number of components
ell = 1;
% Number of Hermite functions in x and y direction
J = [40, 40, 40];
h = hbar;
% Atomic masses of the species (87Rb)
```

```matlab
m = 1.44e-25;
% Number of particle of the species
N = 1e4;
% Frequency of the harmonic part of the potential
omega = [2*pi, 2*pi, 2*sqrt(8)*pi]*90;
% Scattering length
sigma = 368.8/(4*pi*sqrt(m*omega(1)*omega(2)/h/omega(3)))/N;
% Scale of Hermite functions (see getscales.m)
[gamma,omegai,omegabar,mbar] = getscales(m,omega);
% Function name of the potential
potential{1} = @potential1;


%%% PARAMETERS FOR THE PLOT
% x range (used only for the evaluation).
% The y and z ranges are equal to the x range
% (but can be modified inside function plots).
xmin = -4;
xmax = 4;
% Resolution of the surface (only used for the plot)
M = 201;


%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% POTENTIAL FUNCTION %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%
function f = potential1(x,y,z,m,omega)
% Potential V1 is a function of the coordinates x,y,z,
% the mass m and the harmonic part frequency omega.
% It may depend also on other local parameters.

% possible center of displacement
x0 = 0;
y0 = 0;
z0 = 0;
% possible rotation
Omega = 0;
f = m*((omega(1)*(cos(Omega)*(x-x0)+sin(Omega)*(y-y0))).^2+...
(omega(2)*(sin(Omega)*(x-x0)-cos(Omega)*(y-y0))).^2+...
(omega(3)*(z-z0)).^2)/2;
```

```
%%%%%%%%%%%%%%%%%%
%%% OUTPUT DATA %%%
%%%%%%%%%%%%%%%%%%%%
% The output gs is a structure containing:
% gs.h = Dirac constant
% gs.m = Atomic mass of the species
% gs.N = Number of particles
% gs.sigma = Scattering length
% gs.omega = Frequency of the harmonic part of the potential
% gs.phis{1} = Hermite coefficients of the ground state
% gs.potential{1} = Potential function
% gs.mu = Chemical potential of the ground state
% gs.energy = Energy of the ground state
```

In case of a Vector Gross–Pitaevskii Equation ($\ell > 1$), `m`, `N`, `omega` and `sigma` are arrays, see the two-dimensional example in Section 4.3.1.

4.2.2. plothermite3d

Given the coefficients $\{\varphi_j\}$ and a Cartesian grid of points, it computes the values of the Hermite spectral representation

$$\sum_{j=0}^{J-1} \varphi_j \mathcal{H}_j^\gamma(\xi)$$

at the grid points.

4.2.3. real2spectral3d

Given the values of a function $\varphi^{(l)}(\xi)$ at the quadrature nodes of the Gaussian formula, it computes the corresponding coefficients in the Hermite spectral representation.

4.2.4. phirw3d

Given the Hermite coefficients of the functions $\varphi^{(l)}(\xi)$, $l = 1, \ldots, \ell$, it computes the corresponding values at the quadrature nodes of the Gaussian formula.

4.3. Functions for the two-dimensional case

Here we comment the driver function *GSGPEs2d*, the remaining functions for the two-dimensional case being equivalent to the functions described in sections 4.2.2–4.2.4 for the three-dimensional case.

4.3.1. GSGPEs2d

This is a driver function to compute the ground state of a system of $\ell = 3$ two-dimensional Gross–Pitaevskii equations. The parameters are as follows: $m_1 = m_2 = m_3 = 1.44 \cdot 10^{-25}$ kg, $N_1 = N_2 = N_3 = 10^7$, $g_{lk} = 2\pi\sigma_{lk}(m_l + m_k)/(m_l m_k)$, $\sigma_{lk} = 10^{-6}$, $l \neq k$ and $\sigma_{ll} = 10^{-7}$. The potentials are

$$V_1(x_1, x_2) = \frac{m_1}{2}\left[\omega_{11}^2(x - x_0)^2 + \omega_{12}^2 y^2\right]$$

$$V_2(x_1, x_2) = \frac{m_2}{2}\left[\omega_{21}^2(x\cos\Omega + y\sin\Omega)^2 + \omega_{22}^2(x\sin\Omega - y\cos\Omega)^2\right]$$

$$V_3(x_1, x_2) = \frac{m_3}{2}\left[\omega_{31}^2(x + x_0)^2 + \omega_{32}^2 y^2\right]$$

with

$$\begin{bmatrix}\omega_{11} & \omega_{12} \\ \omega_{21} & \omega_{22} \\ \omega_{31} & \omega_{32}\end{bmatrix} = \begin{bmatrix}\frac{3}{2}\pi & \pi \\ \pi & 2\pi \\ \frac{3}{2}\pi & \pi\end{bmatrix} \text{ Hz,} \quad x_0 = 10^{-5} \text{ m,} \quad \Omega = \frac{\pi}{4}.$$

The number of Hermite functions is set to 100 for each spatial direction. The values of the parameters and the definition of the potential are set inside the function as follows:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% USER DEFINED PARAMETERS %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Number of components
ell = 3;
% Number of Hermite functions in x and y direction
J = [100,100];
h = hbar;
% Atomic masses of each species (87Rb)
m = [1.44e-25; 1.44e-25; 1.44e-25];
% Number of particles of each species
N = [1e7; 1e7; 1e7];
% Frequencies of the harmonic part of the potentials
```

```matlab
omega = [1.5*pi, pi;
           pi, 2*pi;
         1.5*pi, pi];
% Scattering lengths
sigma = [1e-7, 1e-6, 1e-6;
         1e-6, 1e-7, 1e-6;
         1e-6, 1e-6, 1e-7];
% Scale of Hermite functions (see getscales.m)
[gamma,omegai,omegabar,mbar] = getscales(m,omega);
% Function names of the potentials
potential{1} = @potential1;
potential{2} = @potential2;
potential{3} = @potential3;

%%% PARAMETERS FOR THE PLOT
% x range (used only for the plot).
% The y range is equal to the x range
% (but can be modified inside function plots).
% The (function value) range is automatic.
xmin = -5;
xmax = 5;
% Resolution of the surface (only used for the plot)
M = 101;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% POTENTIAL FUNCTIONS %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z = potential1(x,y,m,omega)
% Potential V1 is a function of the coordinates x,y,
% the mass m and the harmonic part frequency omega.
% It may depend also on other local parameters.

% center displacement
x0 = -1e-5;
y0 = 0;
% rotation
Omega = 0;
z = m*((omega(1)*(cos(Omega)*(x-x0)+sin(Omega)*(y-y0))).^2+...
```

16

```matlab
(omega(2)*(sin(Omega)*(x-x0)-cos(Omega)*(y-y0))).^2)/2;

function z = potential2(x,y,m,omega)
% Potential V2 is a function of the coordinates x,y,
% the mass m and the harmonic part frequency omega.
% It may depend also on other local parameters.

% center displacement
x0 = 0;
y0 = 0;
% rotation
Omega = pi/4;
z = m*((omega(1)*(cos(Omega)*(x-x0)+sin(Omega)*(y-y0))).^2+...
(omega(2)*(sin(Omega)*(x-x0)-cos(Omega)*(y-y0))).^2)/2;

function z = potential3(x,y,m,omega)
% Potential V3 is a function of the coordinates x,y,
% the mass m and the harmonic part frequency omega.
% It may depend also on other local parameters.

% center displacement
x0 = 1e-5;
y0 = 0;
% rotation
Omega = 0;
z = m*((omega(1)*(cos(Omega)*(x-x0)+sin(Omega)*(y-y0))).^2+...
(omega(2)*(sin(Omega)*(x-x0)-cos(Omega)*(y-y0))).^2)/2;

%%%%%%%%%%%%%%%%%%%%
%%% OUTPUT DATA %%%
%%%%%%%%%%%%%%%%%%%%
% The output gs is a structure containing:
% gs.h = Dirac constant
% gs.m = Atomic masses of each species
% gs.N = Number of particles of each species
% gs.sigma = Scattering lenghts of each species
% gs.omega = Frequency of the harmonic part of the potentials
% gs.phis{1:ell} = Hermite coefficients of the ground state
```

```
% gs.potential{1:ell} = Potential functions
% gs.mu = Chemical potentials of the ground state
% gs.energy = Energy of the ground state
```

4.4. Functions for the one-dimensional case

Here we comment the driver function *GSGPE1d*, the remaining functions for the one-dimensional case being equivalent to the functions described in sections 4.2.2–4.2.4 for the three-dimensional case.

4.4.1. GSGPEs1d

This is a driver function to compute the ground state of a system of $\ell = 2$ one-dimensional Gross–Pitaevskii equations. The parameters are as follows: $m_1 = m_2 = 1.44 \cdot 10^{-25}$ kg, $N_1 = N_2 = 10^7$, $g_{lk} = 2\pi\sigma_{lk}(m_l + m_k)/(m_l m_k)$, $\sigma_{lk} = 10^{-2}$ m^{-1}. The potentials are

$$V_1(x_1) = \frac{m_1}{2}\omega_{11}^2(x - x_0)^2, \quad V_2(x_1) = \frac{m_2}{2}\omega_{21}^2(x + x_0)^2$$

with $\omega_{11} = \omega_{21} = \pi$ Hz and $x_0 = 10^{-5}$ m. The number of Hermite functions is set to 100. The values of the parameters and the definition of the potential are set inside the function as follows:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% USER DEFINED PARAMETERS %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Number of components
ell = 2;
% Number of Hermite functions in x direction
J = 100;
h = hbar;
% Atomic masses of each species (87Rb)
m = [1.44e-25; 1.44e-25];
% Number of particles of each species
N = [1e7; 1e7];
% Frequencies of the harmonic part of the potentials
omega = [pi;
         pi];
% Scattering lengths of each species
sigma = [1e-2, 1e-2;
```

```matlab
          1e-2, 1e-2];
% Scale of Hermite functions (see getscales.m)
[gamma,omegai,omegabar,mbar] = getscales(m,omega);
% Function names of the potentials
potential{1} = @potential1;
potential{2} = @potential2;

%%% PARAMETERS FOR THE PLOT
% x range (used only for the plot).
% The (function value) range is automatic.
xmin = -10;
xmax = 10;
% How much fine is the surface (only used for the plot)
M = 101;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% POTENTIAL FUNCTIONS %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function V = potential1(x,m,omega)
% Potential V1 is a function of the coordinate x,
% the mass m and the harmonic part frequency omega.
% It may depend also on other local parameters.

% center displacement
x0 = 1e-5;
V = m*omega^2*(x-x0).^2/2;


function V = potential2(x,m,omega)
% Potential V2 is a function of the coordinate x,
% the mass m and the harmonic part frequency omega.
% It may depend also on other local parameters.

% center displacement
x0 = -1e-5;
V = m*omega^2*(x-x0).^2/2;

%%%%%%%%%%%%%%%%%%%%%
%%% OUTPUT DATA %%%
```

```
%%%%%%%%%%%%%%%%%%%%
% The output gs is a structure containing:
% gs.h = Dirac constant
% gs.m = Atomic masses of each species
% gs.N = Number of particles of each species
% gs.sigma = Scattering lenghts of each species
% gs.omega = Frequency of the harmonic part of the potentials
% gs.phis{1:ell} = Hermite coefficients of the ground state
% gs.potential{1:ell} = Potential functions
% gs.mu = Chemical potentials of the ground state
% gs.energy = Energy of the ground state
```

5. Acknowledgments

Appendix A. Test run output

Considering the driver function *GSGPEs3d* described in Section 4.2.1, the output will look like

```
>> tic,GSGPEs3d,toc
Iteration 1 in the continuation method
Iteration 2 in the continuation method
Iteration 3 in the continuation method

[...]

Iteration 27 in the continuation method
Iteration 28 in the continuation method
Iteration 29 in the continuation method
Computing the ground state
mu and energy as in test run output [DC07]
mu =
    3.9006
E =
    2.8752
```

```
ans =
          h: 1.0546e-34
          m: 1.4400e-25
          N: 10000
      sigma: 5.6169e-09
      omega: [565.4867 565.4867 1.5994e+03]
       phis: {[40x40x40 double]}
  potential: {[1x1 function_handle]}
         mu: 6.5792e-31
     energy: 4.8496e-27
Elapsed time is 30.527741 seconds.
```

The elapsed time is measured with Matlab 7.6.0 (R2008a) on a 2.20 GHz Intel Core2 Duo.

References

[1] M. H. Anderson, J. R. Ensher, M. R. Matthews, C. E. Wieman, E. A. Cornell, Observation of Bose–Einstein condensation in a dilute atomic vapor, Science 269 (5221) (1995) 198–201.

[2] W. Bao, W. Tang, Ground-state solution of Bose–Einstein condensate by directly minimizing the energy functional, J. Comp. Phys. 187 (2003) 230–254.

[3] R. P. Tiwari, A. Shukla, A basis-set based Fortran program to solve the Gross–Pitaevskii equation for dilute Bose gases in harmonic and anharmonic traps, Comput. Phys. Commun. 174 (12) (2006) 966–982.

[4] C. M. Dion, E. Cancès, Ground state of the time-independent Gross–Pitaevskii equation, Comput. Phys. Commun. 177 (10) (2007) 787–798.

[5] C. J. Myatt, E. A. Burt, R. W. Ghrist, E. A. Cornell, C. E. Wieman, Production of two overlapping Bose–Einstein condensates by sympathetic cooling, Phys. Rev. Lett. 78 (1997) 586–589.

[6] C. Rüegg, N. Cavadini, A. Furrer, H.-U. Güdel, K. Krämer, H. Mutka, A. Wildes, K. Habicht, P. Vorderwisch, Bose–Einstein condensation of the triplet states in the magnetic insulator TlCuCl3, Nature 423 (2003) 62–65.

[7] W. Bao, Ground states and dynamics of multicomponent Bose–Einstein condensates, Multiscale Model. Simul. 2 (2) (2004) 210–236.

[8] M. Caliari, M. Squassina, Location and phase segregation of ground and excited states for 2D Gross–Pitaevskii systems, Dyn. Partial Differ. Equ. 5 (2) (2008) 117–137.

[9] M. Caliari, M. Squassina, Spatial patterns for the three species Gross–Pitaevskii system in the plane, Electron. J. Diff. Eqns. 2008 (79) (2008) 1–15.

[10] M. Caliari, A. Ostermann, S. Rainer, M. Thalhammer, A minimisation approach for computing the ground state of Gross–Pitaevskii systems, J. Comput. Phys. 228 (2009) 349–360.

[11] W. Bao, J. Shen, A fourth-order time-splitting Laguerre–Hermite pseudospectral method for Bose–Einstein condensates, SIAM J. Sci. Comput. 26 (6) (2003) 2010–2028.

[12] W. Gautschi, Orthogonal polynomials (in Matlab), J. Comput. Appl. Math. 178 (1–2) (2005) 215–234.

[13] T. Tang, The Hermite spectral method for Gaussian-type functions, SIAM J. Sci. Comput. 14 (3) (1993) 594–606.

[14] R. Fletcher, Practical Methods of Optimization, 2nd Edition, John Wiley & Sons, New York, 1987.

[15] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, C. A. Sagastizábal, Numerical Optimization: Theoretical and Practical Aspects, 2nd Edition, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[16] S. P. Han, A globally convergent method for nonlinear programming, J. Optimiz. Theory App. 22 (1977) 297–309.

[17] J. E. Dennis, R. B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, SIAM, Philadelphia, 1998.

[18] P. Deuflhard, Newton Methods for Nonlinear Problems, Springer, Berlin, 2004.