Computer Physics Communications 271 (2022) 108185

Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

A complex-valued gradient flow for the entangled bipartite low rank approximation ☆

Moody T. Chu [a], Matthew M. Lin [b,*]

[a] Department of Mathematics, North Carolina State University, Raleigh, NC 27695-8205, United States of America
[b] Department of Mathematics, National Cheng Kung University, Tainan 701, Taiwan

ARTICLE INFO

Article history:
Received 11 January 2021
Received in revised form 22 September 2021
Accepted 27 September 2021
Available online 12 October 2021

Keywords:
Quantum states
Entanglement
Separability
Bipartite system
Low-rank approximation
Gradient dynamics
Wirtinger calculus

ABSTRACT

Entanglement of quantum states in a composite system is of profound importance in many applications. With respect to some suitably selected basis, the entanglement can be mathematically characterized via the Kronecker product of complex-valued density matrices. An approximation to a mixed state can be thought of as calculating its nearest separable state. Such a task encounters several challenges in computation. First, the added twist by the entanglement via the Kronecker product destroys the multi-linearity. The popular alternating least squares techniques for tensor approximation can hardly be applied. Second, there is no clear strategy for selecting a priori a proper low rank for the approximation. Third, the conventional calculus is not enough to address the optimization of real-valued functions over complex variables. This paper proposes a dynamical system approach to tackle low rank approximation of entangled bipartite systems, which has several advantages, including 1) A gradient dynamics in the complex space can be described in a fairly concise way; 2) The global convergence from any starting point to a local solution is guaranteed; 3) The requirement that the combination coefficients of pure states must be a probability distribution can be ensured; 4) The rank can be dynamically adjusted. This paper discusses the theory, algorithms, and presents some numerical experiments.

1. Introduction

Entanglement is perhaps the most basic mode, yet both intricate and critical, when characterizing a complicated phenomenon that involves components interacting with each other. Entanglement arises in nature, in different forms, and in almost all areas, such as computer science, physics, biology, and chemistry. Quantum entanglement is particularly spectacular and intriguing. Although the terminology "quantum entanglement" was invented only later, the thought experiment, i.e., the so-called EPR paradox, by Einstein, Podolsky, and Rosen [1] on that two quantum systems interact in such a way as if both their spatial coordinates and their linear momenta are linked, even when the systems are widely separated in space, is now generally regarded as the vanguard of this important notion. Quantum entanglement plays an increasingly more important role in many tasks employing quantum technologies nowadays because of its potential of transmitting information massively, swiftly, concurrently, and securely. The applications include quantum cryptography [2], quantum teleportation [3], measurement-based quantum computation [4], quantum communication [5–8], dense coding and information retrieval [9–11], to name just a few. Together with the rapid experimental progress on quantum control of both hardware and software, there has been a rapidly growing interest in the entanglement theory [12]. If quantum computing can ultimately be realized, many salient applications stand to benefit from this quantum way of information exchange.

Of vital importance in both theory and practice is the fundamental question of entanglement detection and certification, i.e., how can one be sure that entanglement is indeed happening and, when an entangling operation is taking place, how can one quantify the entanglement in the experiment? For years, these questions have been pursued by many researchers in different directions [13–16]. A variety of possible tactics has been proposed to tackle this verification problem, ranging from Bell inequalities, entanglement witnesses, and spin squeezing inequalities to entropic inequalities, the measurement of nonlinear properties of the quantum state and the approximation of

☆ The review of this paper was arranged by Prof. Hazel Andrew.

* Corresponding author.
E-mail addresses: chu@math.ncsu.edu (M.T. Chu), mhlin@mail.ncku.edu.tw (M.M. Lin).

https://doi.org/10.1016/j.cpc.2021.108185
0010-4655/

positive maps. Some of the tactics have been built into software packages, e.g., QETLAB [17]. It is beyond our capabilities, nor is there room in this paper, to introduce even the basics of the different methods. Out of the numerously many, we mention merely three review articles [15,18,19] whose lists of hundreds of references should be indicative of the breadth and the depth of the vast research endeavors in this area. This work concerns only about a fairly focused subject, namely, given a bipartite mixed state, find numerically its nearest separable state [20–22].

Exactly quantifying the entanglement of a mixed state by finding its nearest separable state is extremely computationally demanding, if at all possible. Even the relaxed problem of determining separability with a margin of error is NP-hard [18,23,24]. See also [25] for general tensor decomposition problems. However, we must not misconceive that an NP-hard problem is forever hopeless and untouchable. A practical example is the problem that expressing $\sqrt{2}$ is NP-hard in theory because it requires infinite complexity on a Turing machine, but we do have polynomial-time algorithms to approximate it to any finite precision. Likewise, the NP-hardness encountered in entanglement quantification do not imply that we cannot approximately solve the problem by numerical means. It is in this spirit that we propose to cast the approximation problem as a constrained optimization problem over complex variables in this paper. We further propose to tackle the optimization numerically by means of a complex-valued gradient flow for several benefits which will be elaborated in the sequel.

Before we set up the problem and detail our method, we ought to stress that the distance measurement depends highly on the assumptions and the applications. See the discussion in [26] and also [27, Page 406]. For example, if the goal is to measure the maximum probability of distinguishability between two quantum states ρ and σ, then resorting to the so-called Kolmogorov-Smirnov (KS) test for comparing random samples is more appropriated. In turn, the trace metric

$$D_T(\rho,\sigma) := \frac{1}{2}\mathrm{Tr}\sqrt{(\rho-\sigma)^2}$$

is perhaps preferred. On the other hand, since repeated measurements are necessary in quantum computation, calculating the minimum number of measurements required to distinguish two different states is also a reasonable metric. In this case, the Bures distance

$$D_B(\rho,\sigma) := \sqrt{2-2\mathrm{Tr}\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}}},$$

which is an analogue of the Fisher information in classical statistics, can be employed. If we regard the density matrix as an integrated entity of the state per se, then the Frobenius norm

$$D_F(\rho,\sigma) = \frac{1}{2}\|\rho-\sigma\|_F = \frac{1}{2}\sqrt{\mathrm{Tr}(\rho-\sigma)^2}$$

measure the geometric difference between two ensembles without any particular assumption. Though all norms are equivalent over finite dimensional spaces, different choices of metrics will lead to a different approximation result and the associated interpretation. Note also that not all distance formulas are easy to use for numerical computation. The Frobenius norm involves the square root of a scalar, but D_T or D_B requires taking the positive square root of a positive definite matrix repeatedly in the computation, which will be costly. Obtaining the gradient information of the objective function is another concern. As a starter, we use the Frobenius norm D_F in this work for its ease of implementation, but it will require separate works to develop new schemes and the pertinent convergence theory for D_T and D_B. A numerical comparison of various measures is worthy of further investigation, but is beyond the scope of this paper.

This paper is organized as follows. We begin in Section 2 with a brief sketch of the notion of entanglement and separability. Our goal is to bring forth the mathematical equivalence of traditional notations used in quantum mechanics by physicists, e.g., the tensor product versus the Kronecker product, the Dirac's ket notation $|\mathbf{x}\rangle$ versus the complex column vector $\mathbf{x}$, and so on. In Section 3 we review the notion of Wirtinger calculus. Such a concept is not new in the literature. However, in order to retrieve the gradient information for the far more complicated objective function in our problem, we have to introduce some tools to relate it to the classical matrix calculus. As a demonstration, we apply our tools to the case of rank-one and derive the projected gradient in a fairly concise way. Our main contribution is at Section 4 where we develop a complex-valued gradient flow for the general bipartite low-rank approximation problem. In contrast to the popular greedy alternating least squares (GALS) method which is hard to analyze and whose convergence is not guaranteed, our gradient flow approach is capable of detecting the infeasibility, automatically satisfying the sum-to-one constraint, dynamically adjusting the needed rank, and most importantly, enjoying the global convergence from established results in the dynamical systems theory. This gradient flow characterized in a compact ordinary differential equation (ODE) makes it easier to analyze the underlying dynamical properties. In Section 5 we implement this method with some existing ODE integrators and carry out some interesting numerical experiments.

2. Basics of quantum mechanics

In this section we outline some background information on quantum mechanics to motivate our optimization problem over complex variables. Our emphasis here is on bridging many of the physics notions to the mathematical terms. For a more thorough and in-depth grasp of the main ideas, we suggest [28–30] and the classic book [27]. Readers who are familiar with the subject and the notion of entanglement might skip to the next section immediately.

One of the basic postulates in quantum mechanics is that each quantum mechanical system is associated with a complex Hilbert space $\mathscr{H}$. Any unit vector in $\mathscr{H}$ is referred to as a pure state which typically denoted by the Dirac's ket notation $|\mathbf{x}\rangle$. Two pure states $|\mathbf{x}\rangle$ and $|\mathbf{y}\rangle$ are considered equivalent if they differ only by a phase change, i.e., if $|\mathbf{x}\rangle = c\,|\mathbf{y}\rangle$ for some $c \in \mathbb{C}$ with $|c| = 1$. For simplicity, we shall limit ourselves to the finite dimensional spaces. Then, with respect to a prescribed basis, we can interpret $|\mathbf{x}\rangle$ as merely a column vector $\mathbf{x} \in \mathbb{C}^d$, where d is the dimension of $\mathscr{H}$.

The orthogonal projection of any $\mathbf{z} \in \mathscr{H}$ onto a given pure state $\mathbf{x}$ plays the role of a critical operator. There is a one-to-one correspondence between the unit vector $\mathbf{x}$, indeed its equivalent class under phase change, and the associated projection operator $\mathcal{D} := \mathbf{x}\mathbf{x}^* \in \mathbb{C}^{d\times d}$, where $\mathbf{x}^*$ denotes the conjugate transpose of $\mathbf{x}$. (In the Dirac's notation, $\mathbf{x}^*$ corresponds to the bra vector $\langle\mathbf{x}|$ and $\mathcal{D} = |\mathbf{x}\rangle\langle\mathbf{x}|$.) We may therefore interchangeably represent a pure state $\mathbf{x}$ by the so called density matrix $\mathcal{D}$.

A mixed quantum state is a probabilistic ensemble of pure states. A mixed state cannot be described with a single ket vector. Instead, it is described by its associated density matrix ρ defined as the convex combination

$$\rho := \sum_i \mu_i \mathbf{x}_i \mathbf{x}_i^*; \quad \sum_i \mu_i = 1; \quad \mu_i \geq 0 \tag{1}$$

of some pure states $\mathbf{x}_i$. The density matrix of a general state in $\mathscr{H}$, therefore, is a positive semi-definite operator with unit trace. It is worth noting that density matrices acting on $\mathscr{H}$ form a convex set whose extreme points are the pure states. Therefore, by the Carathéodory theorem [31, Theorem 2.2.4], the summation in (1) needs at most $d^2 + 1$ terms to express every possible such combination. In fact, by taking the spectral decomposition of ρ as a special case of (1), we need no more than d terms in the summation to represent ρ. This count of needed numbers of terms becomes a complicated issue when entanglement takes place.

Suppose $\mathscr{H}_1$ and $\mathscr{H}_2$ represent two finite dimensional quantum mechanical systems with basis states $\{\mathbf{e}_i\}$ and $\{\mathbf{f}_j\}$, respectively. It is often desirable to describe the composite system in which the entries of quantum states in one system interact in some physical sense with all entries of quantum states in the other system simultaneously. A mathematical way to characterize such a phenomenon is the tensor product space $\mathscr{H}_1 \otimes \mathscr{H}_2$ defined by

$$\mathscr{H}_1 \otimes \mathscr{H}_2 := \{\sum_{i,j} \mathbf{u}_i \otimes \mathbf{v}_j | \mathbf{u}_i \in \mathscr{H}_1, \mathbf{v}_j \in \mathscr{H}_2\},$$

where $\otimes$ denotes the usual outer product in linear algebra. An inner product can be induced via the relationship

$$(\mathbf{x} \otimes \mathbf{y})^* (\mathbf{z} \otimes \mathbf{w}) := (\mathbf{x}^* \mathbf{z})(\mathbf{y}^* \mathbf{w}).$$

With respect to a specified ordering of the basis $\{\mathbf{e}_i \otimes \mathbf{f}_j\}$, any element $\boldsymbol{\psi}$ in the bipartite system $\mathscr{H}_1 \otimes \mathscr{H}_2$ can be represented by a matrix $C = [c_{ij}]$ which can also be identified as a column vectorization $\mathbf{vec}(C)$. A pure state C in $\mathscr{H}_1 \otimes \mathscr{H}_2$ can be written as a complex matrix with $\|C\|_F = 1$. Similar to (1), the density matrix of a mixed state $\rho \in \mathcal{H}_1 \otimes \mathcal{H}_2$ in the bipartite system is an operator acting on matrices in $\mathcal{H}_1 \otimes \mathcal{H}_2$ and, hence, an order-4 tensor which we may write in the form

$$\rho = \sum_i \mu_i C_i \otimes C_i; \quad \sum_i \mu_i = 1; \quad \mu_i \geq 0, \tag{2}$$

where each C_i represents a pure state and $\otimes$ is now interpreted as the Kronecker product.

We say that the pure state $\boldsymbol{\psi} \in \mathscr{H}_1 \otimes \mathscr{H}_2$ is separable if it can be expressed as

$$\boldsymbol{\psi} = \boldsymbol{\psi}_1 \otimes \boldsymbol{\psi}_2,$$

where $\boldsymbol{\psi}_i \in \mathscr{H}_i$, $i = 1, 2$, are pure states. A pure state in the composite system can be entangled. However, a pure state in the bipartite system can always be decomposed as a linear combination of separable states. This is known as the Schmidt decomposition theorem which we state below.

Lemma 1 *(Schmidt Decomposition Theorem [32]). Any unit vector $\boldsymbol{\psi} \in \mathcal{H}_1 \otimes \mathcal{H}_2$ can be written in the form*

$$\boldsymbol{\psi} = \sum_k \sqrt{p_k} \mathbf{u}_k \otimes \mathbf{v}_k \tag{3}$$

where the vectors $\mathbf{u}_k \in \mathcal{H}_1$ and $\mathbf{v}_k \in \mathcal{H}_2$ are pairwise orthogonal and normalized, and $\{p_k\}$ is a probability distribution.

Suppose that the pure state $\boldsymbol{\psi}$ in Lemma 1 is represented by the matrix C. Then, from the linear algebra viewpoint, the Schmidt decomposition (3) is precisely the classical singular value decomposition of C. Thus far, nothing is particularly new, except that the Schmidt decomposition holds even in a general (separable) Hilbert space and can be proved without employing linear algebra.

We must note, however, that $\boldsymbol{\psi}$, $\mathbf{u}_k$, and $\mathbf{v}_k$ are merely "vectors" in $\mathcal{H}_1 \otimes \mathcal{H}_2$, $\mathcal{H}_1$, and $\mathcal{H}_2$, respectively. The decomposition (3) cannot be readily translated into density matrices. The real challenge in the separability problem is to determine whether a given mixed state density matrix ρ in the bipartite system can be decomposed as

$$\rho = \sum_i \eta_i \mathcal{D}_i^{(1)} \otimes \mathcal{D}_i^{(2)}, \quad \sum_i \eta_i = 1; \quad \eta_i \geq 0, \tag{4}$$

where $\{\mathcal{D}_i^{(1)}\}$ and $\{\mathcal{D}_i^{(2)}\}$ are density matrices of the subsystems $\mathscr{H}_1$ and $\mathscr{H}_2$, respectively. That is, a general (mixed) quantum state $\rho \in \mathcal{H}_1 \otimes \mathcal{H}_2$ is said to be separable if and only if the decomposition (4) holds; otherwise ρ is entangled [14,18,33]. The problem of separability in a composite quantum system is now a problem of structured decomposition in linear algebra. Since $\{\mathcal{D}_i^{(1)}\}$ and $\{\mathcal{D}_i^{(2)}\}$ themselves are probabilistic ensembles of pure states in the form (1), we may further reduce the right-hand side of (4) to

$$\rho = \sum_i \theta_i (\mathbf{x}_i \mathbf{x}_i^*) \otimes (\mathbf{y}_i \mathbf{y}_i^*), \quad \sum_i \theta_i = 1; \quad \theta_i \geq 0, \tag{5}$$

where $\mathbf{x}_i \in \mathscr{H}_1$ and $\mathbf{y}_i \in \mathscr{H}_2$ are unit vectors. A given density matrix ρ of a mixed state over the space $\mathcal{H}_1 \otimes \mathcal{H}_2$ is separable if and only if it is the convex combination of tensor products of density matrices of pure states over $\mathcal{H}_1$ and $\mathcal{H}_2$ [12,14,34,35]. This expression is referred to as ρ being prepared by a coordinated local operations and classical communication (LOCC) protocol.

The collection of all separable states in a bipartite system form a convex set with pure separable states as its extreme points [29]. Given a general mixed state ρ, its nearest separable approximation therefore should be unique. Nevertheless, it should not be confused that there might be infinitely many ways to decompose this unique nearest approximation in the ensemble form (5). For better control of the quantum system in applications, it is of paramount importance to quantify the amount of entanglement [13,26,36,37].

More specifically, let R be a fixed positive integer and $[\![R]\!] := \{1, 2, \ldots, R\}$. The entangled bipartite rank-R approximation to a given positive definite (PD) matrix $\rho \in \mathbb{C}^{mn\times mn}$ with unit trace concerns finding complex vectors $\mathbf{x}_r \in \mathbb{C}^m$, $\mathbf{y}_r \in \mathbb{C}^n$, each with unit length, and nonnegative real number $\lambda_r \in \mathbb{R}_+$ with unit sum, $r \in [\![R]\!]$, such that the distance

$$\|\rho - \sum_{r=1}^{R} \lambda_r (\mathbf{x}_r \mathbf{x}_r^*) \otimes (\mathbf{y}_r \mathbf{y}_r^*)\|_F^2 \tag{6}$$

is minimized.

A problem similar to (6) with real-valued variables has been studied in [20]. See also [37, Sec. VI]. By encapsulating the summation in a singleton matrix without decomposition so as to regard (6) as a convex programming problem in the matrix variable, the first approach is an iterative scheme (the DA algorithm [20]) adapted from the conventional Frank-Wolfe method [38]. The essence of the Frank-Wolfe algorithm is to consider a sequence of linear approximation of the objective function and move towards a minimizer of this linear function. Consequently, the core component in the DA algorithm is to solve a sequence of projective subproblems [20, Sec. 5], which is expensive and slow. To improve the speed of convergence, a second approach is to seek a convex combination of solutions from all proceeding projective subproblems to minimize (6). The new task becomes a quadratic programming problem in the coefficients λ_r, $r \in [\![R]\!]$. Other than suggesting that a variant of the conjugate gradient (CG) method is applied, most implementation details are omitted from [20]. Some comments are due. First, for real variables, the analytic gradient of the objective function (6) is easy to come by. However, in quantum mechanics, involving complex variables is a necessity in order to characterize quantum properties correctly [28,39]. For that purpose, as we shall show, more work is needed for the objective function over complex variables. The claim in [20, Section 7] that the generalization of the DA algorithm from the real case to the complex case is "quite straightforward" is probably based on obtaining the gradient information numerically. Second, when including all solutions from the projective subproblems, the number R of terms involved in the convex combination can potentially become unbounded. The more iterations take place, the more variables are involved, taking the CG method longer to complete and making the new task of finding λ_r harder. Third, and most critically, the convergence theory has been left as an open question [20, Page 721]. Cutting in from different perspectives, our recent work in [40,41] should have helped address the gap. In contrast, the differential equation approach proposed in this paper is a very different idea and enjoys several advantages, including that we can describe the analytic gradient in the complex space in a fairly concise way and that the global convergence from any starting point to a local solution is guaranteed by existing theory. Most interestingly, we have a convenient way to ensure that the combination coefficients of pure states stay to be a probability distribution and that the rank can be dynamically adjusted.

We need to delineate a little bit further the goal of our problem (6). We have already mentioned that, even with the allowance for a margin of error that is inversely polynomial in the dimension of the problem, the determination of entanglement remains NP-hard [18,23,24]. Our approximation problem is not for such a task per se. Rather, it is formulated as a constrained optimization problem, where for a given ρ and a fixed rank R, we only look for a local separable approximation in the form (6). Nonlinear optimization algorithms are rarely discussed from a complexity point of view because even a simple convex programming problem such as minimizing $f(x) = \max(x^2 - 2, 0)$ can have irrational solutions of which writing the output alone requires infinite complexity. Using the complexity models proposed in [42], nonlinear optimization problems that may be cast within a unifying proximity-scaling framework can have polynomial time algorithms. Some NP-hard optimization problems, such as the traveling salesman problem, can be polynomial-time approximated within some fixed multiplicative factor of the optimal answer. To stay focused on our dynamical system framework, we shall not be concerned about the complexity analysis of problem (6).

3. Wirtinger calculus

Optimization of real-valued functions over complex field is an old subject. See [43] and the references contained therein. While we can identify $\mathbb{C}^n \equiv \mathbb{R}^{2n}$, we must take into account that the multiplication of complex numbers involves a twist of real and imaginary parts. For example, suppose $\mathbf{x} = \mathbf{u} + \imath\mathbf{v}$ and $\mathbf{y} = \mathbf{p} + \imath\mathbf{q}$ with $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$ and $\mathbf{p}, \mathbf{q} \in \mathbb{R}^n$. Then the tensor product yields

$$\mathbf{x} \otimes \mathbf{y} = (\mathbf{u} \otimes \mathbf{p} - \mathbf{v} \otimes \mathbf{q}) + \imath(\mathbf{v} \otimes \mathbf{p} + \mathbf{u} \otimes \mathbf{q}),$$

showing a nontrivial intertwinement between the real and the imaginary parts of the variables. Thus, not only that we have to deal with four real-valued vectors per $\mathbf{x} \in \mathbb{C}^m$ and $\mathbf{y} \in \mathbb{C}^n$, but also that we have to consider their intertwinement which complicates the cost function. Since a real-valued objective function is not differentiable with respect to the complex-valued variables, it seems that a tedious calculation of the derivative with respect to the real and the imaginary parts of the variables is inevitable. It turns out that the difficulty can be circumvented by using the so called Wirtinger derivatives [43,44] to manage the needed differential calculus. As a result, the complex-value gradient can be characterized in an explicit and compact form which helps facilitate the description of numerical methods for solving the problem (6).

In this section, we review the basic idea of Wirtinger calculus. For readers who might not be familiar with this type of calculus, we apply the technique to a complex-valued nonlinear eigenvalue maximization problem as a worked-out example. In the meantime, through this demonstration, we also want to point out how the real-version eigenvalue maximization problem discussed in [20] should be generalized to the complex-version. The twist by the complex values is somewhat subtle.

Suppose that $f : \mathbb{C} \to \mathbb{R}$ is a given real-valued function over a complex variable $z = x + \imath y$. By regarding the same function as $f(z) = u(x, y)$, the Wirtinger derivatives are defined by

$$\begin{cases} \frac{\partial f}{\partial z} & := & \frac{1}{2}(\frac{\partial u}{\partial x} - \imath \frac{\partial u}{\partial y}), \\ \frac{\partial f}{\partial \overline{z}} & := & \frac{1}{2}(\frac{\partial u}{\partial x} + \imath \frac{\partial u}{\partial y}), \end{cases} \tag{7}$$

where all the partial derivatives involved are taken formally with respect to the designated variables. In other words, while we maintain the usual complex conjugate relationship throughout all arithmetic operations involved in the definition of $f(z)$, the two symbols z and $\overline{z}$ are formally regarded as independent with respect to each other.

The definition for scalars can be generalized to functions with multiple variables. However, unlike the Fréchet derivative, the Wirtinger derivative is merely a formality and does not carry any geometric meaning of gauging the rate of change. For our application, we prefer to exploit the Fréchet derivative acting as a linear map on some infinitesimal change of the variables, whence the gradient information of a functional can be retrieved from the Riesz representation theorem [45,46]. The following two lemmas serve as useful tools to bridge the gap.

Lemma 2. *Suppose that a given function $f : \mathbb{C}^n \to \mathbb{R}$ can be expressed in the inner product form $f(\mathbf{z}) = \langle g(\mathbf{z}, \overline{\mathbf{z}}), h(\mathbf{z}, \overline{\mathbf{z}})\rangle$ for some $g, h : \mathbb{C}^n \times \mathbb{C}^n \to \mathbb{C}^n$. Suppose also that the formal partial derivatives of g and h with respect to $\mathbf{z}$ and $\overline{\mathbf{z}}$ exist. Then*

$$\begin{cases} \frac{\partial f}{\partial \mathbf{z}} & = & (\frac{\partial g(\mathbf{z},\overline{\mathbf{z}})}{\partial \mathbf{z}})^\top \overline{h(\mathbf{z},\overline{\mathbf{z}})} + (\frac{\partial \overline{h(\mathbf{z},\overline{\mathbf{z}})}}{\partial \mathbf{z}})^\top g(\mathbf{z},\overline{\mathbf{z}}), \\ \frac{\partial f}{\partial \overline{\mathbf{z}}} & = & (\frac{\partial g(\mathbf{z},\overline{\mathbf{z}})}{\partial \overline{\mathbf{z}}})^\top \overline{h(\mathbf{z},\overline{\mathbf{z}})} + (\frac{\partial \overline{h(\mathbf{z},\overline{\mathbf{z}})}}{\partial \overline{\mathbf{z}}})^\top g(\mathbf{z},\overline{\mathbf{z}}), \end{cases} \tag{8}$$

where $^\top$ stands for the transpose of a matrix.

Proof. The relationships (8) can be obtained directly from the product rule of the Wirtinger derivative that is already known in the literature. However, for later development, we want to offer a different manipulation by treating the symbols $\mathbf{z}$ and $\overline{\mathbf{z}}$ as if we are dealing with real vectors. Our point is that such a procedure justifies that the classical matrix calculus, which we are more accustomed to, can be used formally when f is highly complicated.

Based on the assumption in the lemma, rewrite $f(\mathbf{z})$ as

$$f(\mathbf{z}) = \langle g(\mathbf{z}, \overline{\mathbf{z}}), \overline{h(\mathbf{z}, \overline{\mathbf{z}})}\rangle_{\mathbb{R}},$$

where

$$\langle \mathbf{a}, \mathbf{b}\rangle_{\mathbb{R}} := \sum_{i=1}^{n} a_i b_i$$

stands for the formal inner product as if $\mathbf{a}$ and $\mathbf{b}$ are over the real field. Regarding $\mathbf{z}$ and $\overline{\mathbf{z}}$ as independent variables, we take formally the Fréchet derivative of f as an action on arbitrary $\mathbf{h} \in \mathbb{C}^n$. Therefore, denoting the action by ., we obtain by the conventional product rule that

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{z}}.\mathbf{h} &= \langle \frac{\partial g(\mathbf{z}, \overline{\mathbf{z}})}{\partial \mathbf{z}}.\mathbf{h}, \overline{h(\mathbf{z}, \overline{\mathbf{z}})}\rangle_{\mathbb{R}} + \langle g(\mathbf{z}, \overline{\mathbf{z}}), \frac{\partial \overline{h(\mathbf{z}, \overline{\mathbf{z}})}}{\partial \mathbf{z}}.\mathbf{h}\rangle_{\mathbb{R}}, \\ &= \langle \mathbf{h}, (\frac{\partial g(\mathbf{z}, \overline{\mathbf{z}})}{\partial \mathbf{z}})^\top \overline{h(\mathbf{z}, \overline{\mathbf{z}})}\rangle_{\mathbb{R}} + \langle (\frac{\partial \overline{h(\mathbf{z}, \overline{\mathbf{z}})}}{\partial \mathbf{z}})^\top g(\mathbf{z}, \overline{\mathbf{z}}), \mathbf{h}\rangle_{\mathbb{R}}, \end{aligned} \tag{9}$$

where the second equality is obtained by the adjoint formula of matrices as if over the real field. The first equation in (8) is thus obtained via the formal Riesz representation theorem. □

For the optimization of a general smooth real-valued function, we need to reassemble its complex-valued gradient from the Wirtinger derivatives. This can easily be accomplished as follows.

Lemma 3. *If $f : \mathbb{C}^n \to \mathbb{R}$ is regarded as $f(\mathbf{z}) = f(\mathbf{u}, \mathbf{v})$ for $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, where $\mathbf{z} = \mathbf{u} + \imath \mathbf{v} \in \mathbb{C}^n$. Then the "true" gradient of f is given by*

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial \mathbf{u}} \\ \frac{\partial f}{\partial \mathbf{v}} \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial \mathbf{z}} + \frac{\partial f}{\partial \overline{\mathbf{z}}} \\ \imath(\frac{\partial f}{\partial \mathbf{z}} - \frac{\partial f}{\partial \overline{\mathbf{z}}}) \end{bmatrix}.$$

We now give two examples to illustrate the convenience offered by the above notions of calculation. The first example is meant to show the straightforwardness if we employ (8) as the conventional matrix calculus. It also provides a preview of how the derivatives are to be taken in the second example. The second example is related to the projective subproblem in [20], but shows the subtle distinction in the expression when complex variables are involved.

Example 1. Consider the quadratic form $q(\mathbf{z}) = \mathbf{z}^* H \mathbf{z} = \langle H\mathbf{z}, \mathbf{z}\rangle$, where $\mathbf{z} \in \mathbb{C}^n$ and $H \in \mathbb{C}^{n\times n}$ is a fixed Hermitian matrix. By identifying $g(\mathbf{z}) = H\mathbf{z}$ and $h(\mathbf{z}) = \mathbf{z}$ in Lemma 2, we may take the Wirtinger derivatives formally in the spirit of (8) and obtain easily that $\frac{\partial q}{\partial z} = \overline{H}\overline{\mathbf{z}}$ and $\frac{\partial q}{\partial \overline{z}} = H\mathbf{z}$. The same result can also be obtained by going through the far more complicated entry-by-entry calculation based on (7).

Example 2. Given a fixed positive semi-definite matrix A in $\mathbb{C}^{mn\times mn}$, consider the structured rank-1 approximation in the form

$$\min_{\substack{\lambda\in\mathbb{R}_+,\mathbf{x}\in\mathbb{C}^m,\mathbf{y}\in\mathbb{C}^n\\ \|\mathbf{x}\|=1,\|\mathbf{y}\|=1}} \|A-\lambda(\mathbf{x}\mathbf{x}^*)\otimes(\mathbf{y}\mathbf{y}^*)\|_F^2. \tag{10}$$

We can think of (10) as a special case of (6) with $R=1$, except that λ is free from the sum-to-one constraint, i.e., λ is a variable. A similar problem can be found in [20,37,40,41]. The problem (10) might have multiple local solutions. It is clear that, for given unit vectors $\mathbf{x}\in\mathbb{C}^m$ and $\mathbf{y}\in\mathbb{C}^n$, the objective function in (10) is minimized only if λ is the orthogonal component of A in the direction $(\mathbf{x}\otimes\mathbf{y})(\mathbf{x}\otimes\mathbf{y})^*$. Therefore, the minimization problem (10) is equivalent to the problem of maximizing the function

$$\lambda(\mathbf{x},\mathbf{y}) := \langle A, (\mathbf{x}\otimes\mathbf{y})(\mathbf{x}\otimes\mathbf{y})^*\rangle, \tag{11}$$

subject to the constraints that $\mathbf{x}$ and $\mathbf{y}$ are of unit lengths. For this purpose, define

$$\mathcal{C}(\mathbf{x},\mathbf{y}) := \textbf{reshape}(A(\mathbf{x}\otimes\mathbf{y}),n,m)\in\mathbb{C}^{n\times m}, \tag{12}$$

where "**reshape**" means literally to rearrange the column vector $A(\mathbf{x}\otimes\mathbf{y})$ into an n by m matrix. We now demonstrate how to use the Wirtinger calculus to derive the first order optimality condition of (11) as follows in the proof of the next lemma.

Lemma 4. *Assume that $A\in\mathbb{C}^{mn\times mn}$ is positive semi-definite. Then a critical point $(\mathbf{x},\mathbf{y})\in\mathbb{C}^m\times\mathbb{C}^n$ for the maximization of (11) subject to the unit constraint necessarily satisfies the polynomial system*

$$\begin{cases} \mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}} = \lambda(\mathbf{x},\mathbf{y})\mathbf{x},\\ \mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}} = \lambda(\mathbf{x},\mathbf{y})\mathbf{y}. \end{cases} \tag{13}$$

Proof. To prepare for the Wirtinger calculus, we rewrite $\lambda(\mathbf{x},\mathbf{y})$ with the definition (12) as

$$\lambda(\mathbf{x},\mathbf{y}) = \overline{\mathbf{y}}^\top\mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}}.$$

Taking the Wirtinger derivatives with respect to $\overline{\mathbf{x}}$ and $\overline{\mathbf{y}}$, respectively, we see that

$$\begin{cases} \frac{\partial\lambda}{\partial\overline{\mathbf{x}}} = \mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}},\\ \frac{\partial\lambda}{\partial\overline{\mathbf{y}}} = \mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}}. \end{cases}$$

Since $\lambda(\mathbf{x},\mathbf{y})$ is real, we may also write

$$\lambda(\mathbf{x},\mathbf{y}) = \mathbf{y}^\top\overline{\mathcal{C}(\mathbf{x},\mathbf{y})}\mathbf{x},$$

and, hence,

$$\begin{cases} \frac{\partial\lambda}{\partial\mathbf{x}} = \overline{\mathcal{C}(\mathbf{x},\mathbf{y})}^\top\mathbf{y},\\ \frac{\partial\lambda}{\partial\mathbf{y}} = \overline{\mathcal{C}(\mathbf{x},\mathbf{y})}\mathbf{x}. \end{cases}$$

Suppose $\mathbf{x}=\mathbf{u}+\imath\mathbf{v}$ and $\mathbf{y}=\mathbf{p}+\imath\mathbf{q}$ with $\mathbf{u},\mathbf{v}\in\mathbb{R}^m$ and $\mathbf{p},\mathbf{q}\in\mathbb{R}^n$. Using Lemma 3, we can write the gradient of $\lambda(\mathbf{u},\mathbf{v},\mathbf{p},\mathbf{q})$ as

$$\begin{cases} \frac{\partial\lambda}{\partial\mathbf{u}} = \overline{\mathcal{C}(\mathbf{x},\mathbf{y})}^\top\mathbf{y}+\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}} = 2\,\mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}}),\\ \frac{\partial\lambda}{\partial\mathbf{v}} = \imath(\overline{\mathcal{C}(\mathbf{x},\mathbf{y})}^\top\mathbf{y}-\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}}) = 2\,\mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}}),\\ \frac{\partial\lambda}{\partial\mathbf{p}} = \overline{\mathcal{C}(\mathbf{x},\mathbf{y})}\mathbf{x}+\mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}} = 2\,\mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}}),\\ \frac{\partial\lambda}{\partial\mathbf{q}} = \imath(\overline{\mathcal{C}(\mathbf{x},\mathbf{y})}\mathbf{x}-\mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}}) = 2\,\mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}}), \end{cases}$$

where Re and Im stand for the real and the imaginary parts, respectively. The projection of $[(\frac{\partial\lambda}{\partial\mathbf{u}})^\top,(\frac{\partial\lambda}{\partial\mathbf{v}})^\top]^\top$ onto the unit sphere

$$S^{2m-1} := \left\{\begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix}\in\mathbb{R}^{2m} \mid \|\mathbf{u}\|_2^2+\|\mathbf{v}\|_2^2=1\right\}$$

is given by

$$\begin{bmatrix} 2\,\mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})\\ 2\,\mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})\end{bmatrix} - 2(\mathbf{u}^\top\,\mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})+\mathbf{v}^\top\,\mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}}))\begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix}. \tag{14}$$

A close examination of $\lambda(\mathbf{x},\mathbf{y})$ shows that

$$\begin{aligned}\lambda(\mathbf{x},\mathbf{y}) &= \lambda(\mathbf{x},\mathbf{y})^\top = (\mathbf{u}^\top-\imath\mathbf{v}^\top)(\mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})+\imath\,\mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})\\ &= (\mathbf{u}^\top\,\mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})+\mathbf{v}^\top\,\mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}}))+\imath(\mathbf{u}^\top\,\mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})-\mathbf{v}^\top\,\mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}})).\end{aligned}$$

Therefore,

$$\begin{cases} \mathbf{u}^\top \mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top \overline{\mathbf{y}}) + \mathbf{v}^\top \mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top \overline{\mathbf{y}}) = \lambda(\mathbf{x},\mathbf{y}), \\ \mathbf{u}^\top \mathrm{Im}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top \overline{\mathbf{y}}) - \mathbf{v}^\top \mathrm{Re}(\mathcal{C}(\mathbf{x},\mathbf{y})^\top \overline{\mathbf{y}}) = 0. \end{cases} \tag{15}$$

The first order optimality condition requires that the projected gradient be zero. Together with (14), this condition is equivalent to the expression

$$\mathcal{C}(\mathbf{x},\mathbf{y})^\top \overline{\mathbf{y}} - \lambda(\mathbf{x},\mathbf{y})\mathbf{x} = 0.$$

The second equality in (13) can be proved similarly by considering the projection of $[(\frac{\partial \lambda}{\partial \mathbf{p}})^\top, (\frac{\partial \lambda}{\partial \mathbf{q}})^\top]^\top$ onto the unit sphere S^{2n-1}. □

By merely taking the complex conjugation, we may rewrite (13) as

$$\begin{cases} \mathcal{C}(\mathbf{x},\mathbf{y})^* \mathbf{y} = \lambda(\mathbf{x},\mathbf{y})\overline{\mathbf{x}}, \\ \mathcal{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}} = \lambda(\mathbf{x},\mathbf{y})\mathbf{y}. \end{cases} \tag{16}$$

Since $\lambda(\mathbf{x},\mathbf{y})$ is expected to be maximized, and preferably globally, we find that (16) has an interesting interpretation via the notion of singular value decomposition (SVD) in the following sense.

Lemma 5. *Assume that A is positive semi-definite and that $(\mathbf{x},\mathbf{y}) \in \mathbb{C}^m \times \mathbb{C}^n$ is the maximizer for (11). Then $(\lambda, \mathbf{y}, \overline{\mathbf{x}})$ is the dominant singular triplets of $\mathcal{C}(\mathbf{x},\mathbf{y})$, where $\mathbf{y}$ is the dominant left singular vector and $\overline{\mathbf{x}}$ is the dominant right singular vector of $\mathcal{C}(\mathbf{x},\mathbf{y})$.*

It must be noted that the matrix $\mathcal{C}(\mathbf{x},\mathbf{y})$ itself depends nonlinearly on $(\mathbf{x},\mathbf{y})$, so (16) is a nonlinear singular value problem. An SVD-like iteration based on (16) and the associated convergence theory have been developed recently in our paper [40].

Though it is not a focus point of this paper, we also mention that the first order optimality condition can be expressed as a pair nonlinear eigenvalue problems. The idea is nothing more than a rearrangement of (13). To see the relationship, partition the matrix A into blocks

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & T_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & T_{m2} & \cdots & A_{mm} \end{bmatrix},$$

where $A_{ij} \in \mathbb{C}^{n\times n}$, and define the so-called $\mathscr{R}$-folding of A via the rearrangement [47,48]

$$\mathscr{R}(A) := \begin{bmatrix} \mathbf{vec}(A_{11})^\top \\ \mathbf{vec}(A_{21})^\top \\ \vdots \\ \mathbf{vec}(A_{mm})^\top \end{bmatrix} \in \mathbb{C}^{m^2\times n^2},$$

where **vec** represents the conventional vectorization of a matrix by its columns. Define the bilinear operators

$$\begin{cases} \mathscr{A}(\mathbf{y},\overline{\mathbf{y}}) := \mathbf{reshape}(\mathscr{R}(A)(\mathbf{y}\otimes\overline{\mathbf{y}}), [m,m]), \\ \mathscr{B}(\mathbf{x},\overline{\mathbf{x}}) := \mathbf{reshape}(\mathscr{R}(A)^\top(\mathbf{x}\otimes\overline{\mathbf{x}}), [n,n]), \end{cases} \tag{17}$$

over the respective unit spheres. Then it can be shown by direct computation [40] that

$$\lambda(\mathbf{x},\mathbf{y}) = \langle A, (\overline{\mathbf{x}}\otimes\overline{\mathbf{y}})(\mathbf{x}\otimes\mathbf{y})^\top\rangle_{\mathbb{R}} = \langle \mathscr{A}(\mathbf{y},\overline{\mathbf{y}})\mathbf{x}, \overline{\mathbf{x}}\rangle_{\mathbb{R}} = \langle \mathscr{B}(\mathbf{x},\overline{\mathbf{x}})\mathbf{y}, \overline{\mathbf{y}}\rangle_{\mathbb{R}}. \tag{18}$$

Observe that for any $\mathbf{p} \in \mathbb{C}^m$ and $\mathbf{q} \in \mathbb{C}^n$, it holds that

$$\begin{aligned} \langle \mathscr{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}}, \mathbf{q}\rangle &= \langle A, (\overline{\mathbf{x}}\otimes\overline{\mathbf{q}})(\mathbf{x}\otimes\mathbf{y})^\top)\rangle_{\mathbb{R}} = \langle \mathscr{B}(\mathbf{x},\overline{\mathbf{x}})\mathbf{y}, \mathbf{q}\rangle, \\ \langle \mathscr{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}}, \mathbf{p}\rangle &= \langle A, (\overline{\mathbf{p}}\otimes\overline{\mathbf{y}})(\mathbf{x}\otimes\mathbf{y})^\top)\rangle_{\mathbb{R}} = \langle \mathscr{A}(\mathbf{y},\overline{\mathbf{y}})\mathbf{x}, \mathbf{p}\rangle. \end{aligned}$$

Therefore, it must be that

$$\begin{cases} \mathscr{C}(\mathbf{x},\mathbf{y})\overline{\mathbf{x}} = \mathscr{B}(\mathbf{x},\overline{\mathbf{x}})\mathbf{y}, \\ \mathscr{C}(\mathbf{x},\mathbf{y})^\top\overline{\mathbf{y}} = \mathscr{A}(\mathbf{y},\overline{\mathbf{y}})\mathbf{x}. \end{cases} \tag{19}$$

Substituting (19) into (13), we obtain a different expression of the first order optimality condition as two nonlinear eigenvalue problems.

Lemma 6. *Assume that $A \in \mathbb{C}^{mn\times mn}$ is positive semi-definite. Then a critical point $(\mathbf{x},\mathbf{y}) \in \mathbb{C}^m \times \mathbb{C}^n$ for the maximization of (11) subject to the unit constraint necessarily satisfies the polynomial system*

$$\begin{cases} \mathscr{A}(\mathbf{y},\overline{\mathbf{y}})\mathbf{x} = \lambda(\mathbf{x},\mathbf{y})\mathbf{x}, \\ \mathscr{B}(\mathbf{x},\overline{\mathbf{x}})\mathbf{y} = \lambda(\mathbf{x},\mathbf{y})\mathbf{y}. \end{cases} \tag{20}$$

At a glance, it might appear that the expression (20) is the complex generalization of [20, Theorem 5.1], which is already described without proof in [37, Equation (85)]. The fundamental difference is at what has been used to define $\mathscr{A}$ and $\mathscr{B}$ in (17). We use the given positive semi-definite matrix A, whereas the special matrices B used in [20] and σ used in [37] are generally not definite. Without the positive definiteness of the underlying matrix, then by the theory developed in [40] we know that the eigenvalue maximization algorithm for the projective subproblem, which is the core to the DA algorithm proposed in [20], may fail to converge. In fact, our investigation in [40] confirms, both analytically and computationally, that a cyclic behavior of iterates between multiple limit points can happen. This sidetracked comparison is not our main contribution in this paper, but should help answer the open question raised in [20].

4. Quantum low-rank separability approximation

In this section, we propose a complex-valued gradient flow approach to tackle the minimization of (6). The Wirtinger calculus will render the projected gradient flow in a concise form. Tactics will be deployed to avoid the infeasibility and automatically satisfy the sum-to-one constraints during the integration. Furthermore, we build in the algorithm the ability to detect redundancy if the rank R is too large to begin with. Our approach then dynamically reduces the rank R from high to low and tries to find the optimal rank which generally is difficult to predetermine in practice.

4.1. Gradient flow for quantum low-rank approximation

We first calculate the projected gradient of the objective function in (6). Though in an entirely different setting, it is interesting to see that its first order optimality condition resembles that in Lemma 4.

For convenience, introduce the abbreviations

$$\begin{aligned}\Theta &= \Theta(\lambda_1,\ldots,\lambda_R,\mathbf{x}_1,\ldots,\mathbf{x}_R,\mathbf{y}_1,\ldots,\mathbf{y}_R)\\ &:= \rho - \sum_{r=1}^{R}\lambda_r(\mathbf{x}_r\otimes\mathbf{y}_r)(\mathbf{x}_r\otimes\mathbf{y}_r)^* \in \mathbb{C}^{mn\times mn},\end{aligned} \tag{21}$$

and, for each $r\in[\![R]\!]$,

$$\begin{cases}\omega_r = \omega_r(\lambda_1,\ldots,\lambda_R,\mathbf{x}_1,\ldots,\mathbf{x}_R,\mathbf{y}_1,\ldots,\mathbf{y}_R) := \langle \mathbf{x}_r\otimes\mathbf{y}_r, \Theta(\mathbf{x}_r\otimes\mathbf{y}_r)\rangle \in \mathbb{R},\\ \mathscr{C}_r = \mathscr{C}_r(\lambda_1,\ldots,\lambda_R,\mathbf{x}_1,\ldots,\mathbf{x}_R,\mathbf{y}_1,\ldots,\mathbf{y}_R) := \textbf{reshape}(\Theta(\mathbf{x}_r\otimes\mathbf{y}_r),n,m)\in\mathbb{C}^{n\times m}.\end{cases} \tag{22}$$

The following results manifest the capability of the Wirtinger calculus to conveniently facilitate an otherwise laborious calculation.

Lemma 7. *Suppose $\mathbf{x}_r=\mathbf{u}_r+\imath\mathbf{v}_r$ and $\mathbf{y}_r=\mathbf{p}_r+\imath\mathbf{q}_r$ with $\mathbf{u}_r,\mathbf{v}_r\in\mathbb{R}^m$ and $\mathbf{p}_r,\mathbf{q}_r\in\mathbb{R}^n$. Suppose that the objective function*

$$g(\lambda_1,\ldots,\lambda_R,\mathbf{x}_1,\ldots,\mathbf{x}_R,\mathbf{y}_1,\ldots,\mathbf{y}_R) := \langle\Theta,\Theta\rangle \tag{23}$$

is expressed as a function of the real variables λ_r, $\mathbf{u}_r$, $\mathbf{v}_r$, $\mathbf{p}_r$, and $\mathbf{q}_r$, $r\in[\![R]\!]$. Then the portions of the gradient ∇g with respect to the respective real variables are given by

$$\begin{cases}\frac{\partial g}{\partial\lambda_r} &= -2\omega_r,\\ \frac{\partial g}{\partial(\mathbf{u}_r,\mathbf{v}_r)} &= -4\lambda_r\begin{bmatrix}\mathrm{Re}(\mathscr{C}_r^\top\overline{\mathbf{y}}_r)\\ \mathrm{Im}(\mathscr{C}_r^\top\overline{\mathbf{y}}_r)\end{bmatrix}, \quad r\in[\![R]\!].\\ \frac{\partial g}{\partial(\mathbf{p}_r,\mathbf{q}_r)} &= -4\lambda_r\begin{bmatrix}\mathrm{Re}(\mathscr{C}_r\overline{\mathbf{x}}_r)\\ \mathrm{Im}(\mathscr{C}_r\overline{\mathbf{x}}_r)\end{bmatrix},\end{cases} \tag{24}$$

Proof. Employing the technique introduced in (9), it can be checked that the Wirtinger partial derivatives of g with respect to the complex variables $\mathbf{x}_r$, $\overline{\mathbf{x}}_r$, $\mathbf{y}_r$, and $\overline{\mathbf{y}}_r$, $[\![R]\!]$ are given by, respectively,

$$\begin{cases}\frac{\partial g}{\partial\mathbf{x}_r} &= -2\lambda_r\overline{\mathscr{C}_r}^\top\mathbf{y}_r,\\ \frac{\partial g}{\partial\overline{\mathbf{x}}_r} &= -2\lambda_r\mathscr{C}_r^\top\overline{\mathbf{y}}_r,\\ \frac{\partial g}{\partial\mathbf{y}_r} &= -2\lambda_r\overline{\mathscr{C}_r}\mathbf{x}_r,\\ \frac{\partial g}{\partial\overline{\mathbf{y}}_r} &= -2\lambda_r\mathscr{C}_r\overline{\mathbf{x}}_r,\end{cases} \quad r\in[\![R]\!]. \tag{25}$$

Based on (25), we can reassemble ∇g block by block with respect to the corresponding real variables λ_r, $\mathbf{u}_r$, $\mathbf{v}_r$, $\mathbf{p}_r$, and $\mathbf{q}_r$ via the formula in Lemma 3 to give (24). □

Since our problem is constrained to the pure states, we need the projected gradient. Since g is defined over the product space, the projection can be obtained by projecting the blocks of ∇g onto the corresponding unit spheres, S^{2m-1} and S^{2n-1}, respectively.

Lemma 8. *In terms of complex vectors $\mathbf{x}_r$ and $\mathbf{y}_r$, $r \in [\![R]\!]$, the projected gradients of objective function g can be condensed into the expressions*

$$\begin{cases} \mathrm{Proj}_{S^{2m-1}} \frac{\partial g}{\partial(\mathbf{u}_r, \mathbf{v}_r)} &= -4\lambda_r(\mathscr{C}_r^\top \overline{\mathbf{y}}_r - \omega_r \mathbf{x}_r), \\ \mathrm{Proj}_{S^{2n-1}} \frac{\partial g}{\partial(\mathbf{u}_p, \mathbf{v}_q)} &= -4\lambda_r(\mathscr{C}_r \overline{\mathbf{x}}_r - \omega_r \mathbf{y}_r), \end{cases} \quad r \in [\![R]\!]. \tag{26}$$

Proof. For each $r \in [\![R]\!]$, it only remains to calculate the orthogonal component the corresponding block of ∇g in the direction of the pure states. The procedure is analogous to that in the proof of Lemma 4. (For instance, the role of $\mathcal{C}$ of (14) is replaced by the role of $\mathscr{C}_r$.) Similar to (15), we also find from the definition of ω_r in (22) that

$$\omega_r = \mathbf{u}_r^\top \mathrm{Re}(\mathscr{C}_r^\top \overline{\mathbf{y}}_r) + \mathbf{v}_r^\top \mathrm{Im}(\mathscr{C}_r^\top \overline{\mathbf{y}}_r) = \mathbf{p}_r^\top \mathrm{Re}(\mathscr{C}_r \overline{\mathbf{x}}_r) + \mathbf{q}_r^\top \mathrm{Im}(\mathscr{C}_r \overline{\mathbf{x}}_r). \tag{27}$$

(That is, the role of λ is replaced by the role of ω_r.) The analogy holds for each $r \in [\![R]\!]$. The simple expression (26) indeed represents the projected gradient. □

Using the negative projected gradient of the objective function in (6), we now define the complex-valued differential system

$$\begin{cases} \frac{d\lambda_r}{dt} &= 2\omega_r, \\ \frac{d\mathbf{x}_r}{dt} &= 4\lambda_r(\mathscr{C}_r^\top \overline{\mathbf{y}}_r - \omega_r \mathbf{x}_r), \quad r \in [\![R]\!], \\ \frac{d\mathbf{y}_r}{dt} &= 4\lambda_r(\mathscr{C}_r \overline{\mathbf{x}}_r - \omega_r \mathbf{y}_r), \end{cases} \tag{28}$$

where t stands for a dimensionless parameter of time. Note that (28) is a highly coupled differential system because Θ involved all variables. Unlike the iterative algorithm in [20,37], the differential system updates all variables simultaneously at every given time t.

The behavior of gradient dynamics has been well studied in the literature [49–51]. For completion, we mention a few key points related to our problem. First, suppose $\boldsymbol{\xi}(t)$ represents the gradient flow

$$\frac{\partial \boldsymbol{\xi}}{\partial t} = -\nabla h(\boldsymbol{\xi}); \quad \boldsymbol{\xi}(0) = \boldsymbol{\xi}_0 \tag{29}$$

of a smooth objective function $h : U \subset \mathbb{R}^n \to \mathbb{R}$. Then the flow will continue so long as $\nabla h(\boldsymbol{\xi})$ is defined. Suppose $\boldsymbol{\xi}(t)$ stays bounded for all $t \geq 0$. The set of accumulation points

$$\omega(\boldsymbol{\xi}_0) := \left\{ \boldsymbol{\xi}^* \in \mathbb{R}^n \,|\, \boldsymbol{\xi}(t_\nu) \to \boldsymbol{\xi}^* \text{ for some sequence } t_\nu \to \infty \right\}$$

is a non-empty, compact, and connected subset of stationary points

$$\mathcal{C} := \left\{ \boldsymbol{\xi} \in \mathbb{R}^n \,|\, \nabla h(\boldsymbol{\xi}) = 0 \right\}.$$

If ∇h is real analytic in $\boldsymbol{\xi}$, then the well known Łojasiewicz gradient inequality implies that any bounded gradient trajectory is necessarily of finite length [52–54]. The gradient flow therefore must converge globally to a singleton as its limit point [55, Theorem 2.2]. For completeness, we include one such a result below.

Theorem 1. *Suppose that $h : U \to \mathbb{R}$ is real analytic in an open set $U \subset \mathbb{R}^n$. Then for any bounded semi-orbit of the gradient flow (29), there exists a point $\boldsymbol{\xi}^*$ such that $\boldsymbol{\xi}(t) \to \boldsymbol{\xi}^*$ as $t \to \infty$.*

With respect to our differential system, if we separate the real and the imaginary parts of $\mathbf{x}_r$ and $\mathbf{y}_r$, $r \in [\![R]\!]$, then together with λ_r, there are a total of $2(m+n)R + R$ real variables in which the right hand side of (28) is a polynomial system which is real analytic. Note also that, by construction, the vector field $\frac{d\mathbf{x}_r(t)}{dt}$ is tangent to the unit sphere S^{2m-1}, i.e.,

$$\langle \frac{d\mathbf{x}_r(t)}{dt}, \mathbf{x}_r(t) \rangle = \langle 4\lambda_r(\mathscr{C}_r^\top \overline{\mathbf{y}}_r - \omega_r \mathbf{x}_r), \mathbf{x}_r \rangle = 0, \quad r \in [\![R]\!].$$

The trajectories of $\mathbf{x}_r(t)$ and $\mathbf{y}_r(t)$, $r \in [\![R]\!]$, stay on the unit sphere S^{2m-1} and S^{2n-1} for all t, respectively.

If we can guarantee that $0 \leq \lambda_r(t) \leq 1$, $r \in [\![R]\!]$ while staying on an analytic descend flow, then by Theorem 1 the flow $(\lambda_r(t), \mathbf{x}_r(t), \mathbf{y}_r(t))$, $r \in [\![R]\!]$ must converge to a single limit point. Indeed, for our problem, it is necessary to satisfy the nonnegativity condition $\lambda_r(t) \geq 0$ and the sum-to-one condition $\sum_{r=1}^R \lambda_r(t) = 1$. Toward this end, we now explain how to slightly modify (28) to meet these conditions while keeping an analytic gradient flow and, hence, enjoy the global convergence.

4.2. Maintaining nonnegativity and rank reduction

We first address the task of keeping $\lambda_r(t) \geq 0$ for all $r \in [\![R]\!]$. In fact, the need of maintaining nonnegativity in solutions when solving ordinary differential equations has been widely observed in many other applications. See [56] for a brief history of development on this subject and a collection of examples that illuminate the difficulties of imposing nonnegativity. A variety of strategies for enforcing nonnegativity have long been proposed and tested in the literature. The numerical ODE package in the popular computing platform MATLAB, for example, has the option of NonNegative that implements several carefully thought-through and articulated devices, such as redefining the differential equations outside the feasible region, imposing an additional absolute error tolerance on the components that violate the constraint, performing continuous interpolation for the solution between mesh points, and moving along the constraint [57,58].

For our application, we do not need to employ the full scope of these devices. Instead, we adopt the following strategies:

1. **Event detection:** Use an event function to detect when any $\lambda_r(t)$, $r \in [\![R]\!]$ becomes zero during the integration. Most modern programming languages using array operations have the ability to efficiently test whether a barrier of zero is crossed. The option Events of the odeset function in MATLAB, for example, can determine if $\lambda_r(t)$ is decreasing from a positive value to zero and report the time $\widehat{t}$ up to the specified precision when such a crossing happens. When such an event occurs, we terminate the integration and call for a restart from the current position with an adjustment described below.
2. **Rank deduction:** Generally we do not know a priori of an appropriate value of the rank R. Rank determination is in fact an open question. When the event $\lambda_r(\widehat{t}) = 0$ is detected for one particular value r and time $\widehat{t}$, the term $\lambda_r(\mathbf{x}_r \otimes \mathbf{y}_r)(\mathbf{x}_r \otimes \mathbf{y}_r)^*$ contributes nothing to the objective value g in (23) at that instant. In the meantime, the entire vector field in (28) as a whole continues to decrease, at least momentarily, the objective value g *without* the contribution of the term $\lambda_r(\mathbf{x}_r \otimes \mathbf{y}_r)(\mathbf{x}_r \otimes \mathbf{y}_r)^*$. Furthermore, by the definition of an event, we should have the behavior that $\lambda_r(t)$ is decreasing from a positive value to zero, i.e., $\frac{d\lambda_r}{dt}(\widehat{t}) \le 0$. Continuing the integration even with a small step size will certainly violate the nonnegative constraint. Continuing to move along the constraint with $\lambda_r(t) = 0$ is also of no point because it does not help improve the objective value. For these reasons, we decide to drop this term entirely. The rank R in the summation of (21) is decreased by 1. In this way, we build in our algorithm the capability of starting with a rather large R and then dynamically lowering the rank R when a certain component is not needed. Using a large number of R to begin with may seem redundant first, but it enhances the flexibility of searching in more directions, which might give the flow an advantageous vantage to move into a better solution. The rank reduction mechanism, on the other hand, helps to filter out unnecessary directions.
3. **Ratchet restart:** When an event $\lambda_r(\widehat{t}) = 0$ is detected and after the corresponding component is dropped, we use the remaining $(\lambda_s(\widehat{t}), \mathbf{x}_s(\widehat{t}), \mathbf{y}_s(\widehat{t}))$, $s \in [\![R]\!] \backslash \{r\}$, as the initial values and restart the integration. In this way, the currently attained objective value will continue to be ratcheted downward.

4.3. Maintain sum-to-one

Suppose that initially $\lambda_r(0) > 0$, $r \in [\![R]\!]$, and $\sum_{r=1}^R \lambda_r(0) = 1$. To satisfy the constraint $\sum_{r=1}^R \lambda_r(t) = 1$ for all $t \ge 0$, it is necessary to impose the consistency condition

$$\sum_{r=1}^{R} \frac{d\lambda_r(t)}{dt} = 0, \quad \text{for all } t \ge 0, \tag{30}$$

which may not hold in the dynamical system defined by (28). We propose to remedy the situation by modifying the flow for $\lambda_r(t)$ to

$$\frac{d\lambda_r}{dt} = 2(\omega_r - \widetilde{\omega}), \quad r \in [\![R]\!], \tag{31}$$

where $\widetilde{\omega} := \frac{\sum_{r=1}^R \omega_r}{R}$, while the original governing equations for $\frac{d\mathbf{x}_r}{dt}$ and $\frac{d\mathbf{y}_r}{dt}$, $r \in [\![R]\!]$ are kept invariant. By (31), the condition (30) is satisfied, but the resulting system is no longer in the steepest descent direction. It is important to show that with (31) we still have a descent flow.

Lemma 9. *Let $Z(t)$ denote the newly defined flow*

$$Z(t) := (\lambda_1(t), \ldots, \lambda_R(t), \mathbf{x}_1(t), \ldots, \mathbf{x}_R(t), \mathbf{y}_1(t), \ldots, \mathbf{y}_R(t)).$$

Then the objection value of g is descending along the trajectory $Z(t)$.

Proof. Observe that

$$\begin{aligned}
\frac{dg(Z(t))}{dt} &= \nabla g(Z(t)).\frac{dZ(t)}{dt} \\
&= \sum_{r=1}^{R} \frac{dg}{d\lambda_r}\frac{d\lambda_r}{dt} + \sum_{r=1}^{R} \langle \frac{dg}{d(\mathbf{u}_r, \mathbf{v}_r)}, \begin{bmatrix} \frac{d\mathbf{u}_r}{dt} \\ \frac{d\mathbf{v}_r}{dt} \end{bmatrix} \rangle + \sum_{r=1}^{R} \langle \frac{dg}{d(\mathbf{p}_r, \mathbf{q}_r)}, \begin{bmatrix} \frac{d\mathbf{p}_r}{dt} \\ \frac{d\mathbf{q}_r}{dt} \end{bmatrix} \rangle \\
&= \sum_{r=1}^{R} \frac{dg}{d\lambda_r}\frac{d\lambda_r}{dt} - 16\sum_{r=1}^{R} \lambda_r^2(\|\mathscr{C}_r^\top \overline{\mathbf{y}}_r\|^2 - \omega_r^2) - 16\sum_{r=1}^{R} \lambda_r^2(\|\mathscr{C}_r \overline{\mathbf{x}}_r\|^2 - \omega_r^2).
\end{aligned}$$

By (27), each term in the last two summations is nonnegative. On the other hand, observe the first summation that

$$\sum_{r=1}^{R} \frac{dg}{d\lambda_r}\frac{d\lambda_r}{dt} = -4\sum_{r=1}^{R} \omega_r(\omega_r - \widetilde{\omega}) = -4(\sum_{r=1}^{R} \omega_r^2 - \frac{1}{R}(\sum_{r=1}^{R} \omega_r)^2) \le 0,$$

where the last inequality follows from the Cauchy-Schwarz inequality and by the fact that $\omega_r \in \mathbb{R}$, $r \in [\![R]\!]$. In all, we have proved that $g(Z(t))$ is a decreasing function in t. □

5. Numerical experiment

In this section we carry out some numerical experiments to demonstrate the different features in our algorithm, i.e., the descent property of the objective function, the dynamical adjustment of the rank, and the sum-to-one constraint. Since all desired properties have been built in our differential system, any available ODE integrator can be used as a computational tool. The following experiments are performed on a MacBook Pro laptop with Quad-Core Intel Core i5 @ 2.4GHz processor and 16GB RAM by using MATLAB, version 2020b, as the computing platform.

For demonstration purposes, we employ the ODE suite in MATLAB for our experiments. We make no attempt to fine tune the code for efficiency, so the only options we have chosen are to turn on the event finder Events and to set the local error tolerance at AbsTol = 10^{-12} and RelTol = 10^{-12}. Since we know in theory that our flow will converge, it is expected that eventually the integrator can choose a fairly large step size even at this high precision. We purposefully let the integration go for a long period of time to demonstrate the general dynamics.

We should point out that the framework we have set up in the work is for a general bipartite quantum mechanical system with states $\mathbf{x}_r \in \mathbb{C}^m$ and $\mathbf{y}_r \in \mathbb{C}^n$, $r \in [\![R]\!]$, where m and n are arbitrary positive integers. For quantum information applications, the number of qubits used in the system is more relevant. In this case, the dimensions of the underlying Hilbert spaces will grow exponentially in the number of qubits. Recall that a d-qubit system is represented by $(\mathbb{C}^2)^{\otimes d} = \mathbb{C}^2 \otimes \ldots \otimes \mathbb{C}^2$. A state in a d-qubit system can be thought of as a complex vector of dimension 2^d. One could regard $(\mathbb{C}^2)^{\otimes d}$ as an d-partite entangled system of $\mathbb{C}^2$ which we have not considered yet in this paper. Or, if we split $d = p + q$, then we could consider $(\mathbb{C}^2)^{\otimes d} = (\mathbb{C}^2)^{\otimes p} \otimes (\mathbb{C}^2)^{\otimes q}$ as a bipartite entanglement of $(\mathbb{C}^2)^{\otimes p}$ and $(\mathbb{C}^2)^{\otimes q}$. In the latter case, if we take $m = 2^p$ and $n = 2^q$, then our method is applicable.

We test out the different features of our method through the following four experiments. The first two experiments are cast against general m and n. The last two experiments are against a 4-qubit system and a 2-qubit system, respectively.

Experiment 1. This experiment is designed to examine the dynamical adjustment of R. We first generate a test matrix

$$\rho = \sum_{r=1}^{6} \lambda_r (\mathbf{x}_r \mathbf{x}_r^*) \otimes (\mathbf{y}_r \mathbf{y}_r^*),$$

with randomly generated unit vectors $\mathbf{x}_r, \mathbf{y}_r \in \mathbb{C}^5$ and $\lambda_r > 0$, $r \in [\![6]\!]$, satisfying $\sum_{r=1}^{6} \lambda_r = 1$, as the target. In other words, from the beginning, $\rho \in \mathbb{C}^{25\times 25}$ is already separable in itself with rank 6. We are interested in finding out whether ρ can be completely recovered by our method. Pretending that we do not know this exact rank, we consider the approximation problem (6) with artificially high values of R. During the integration, our code monitors the event to see if $\lambda_r(\widehat{t}) = 0$ for some $r \in [\![R]\!]$ at some $\widehat{t}$. When such an event is detected, our code lowers the value of R by 1 and restarts.

Plotted in Fig. 1 are results from an experiment with $R = 20$ initially. The descent property of the objective function g defined in (23) along the solution trajectory is shown in Fig. 1(a). Also in the drawing are a total of 14 circular marks indicating the specific time $\widehat{t}$ at which one of the λ_r values becomes zero. The total dynamics of $\lambda_r(t)$, $r \in [\![20]\!]$, is recorded in Fig. 1(b). Note the termination of some of the $\lambda_r(t)$ curves because an event is detected for that particular $\lambda_r(t)$. The same event detection criteria are applied and the event time can be determined accurately via high-order interpolation [58]. However, partly due to the logarithmic scale in the drawing and partly due to the variable steps in the integrator, the resolution may not manifest the fact that all downward $\lambda_r(t)$ curves are stopped at the same level of diminishing zero. At the end of integration, the rank is indeed reduced to $R = 6$ and the objective value is nearly zero in this particular example.

We do need to point out that, while the original (separable) ρ is almost perfectly approximated, even with the same number ($R = 6$) of components, the ultimate reconstructed data $\{(\lambda_r, \mathbf{x}_r, \mathbf{y}_r)\}$ are not necessarily the same as those used to generate the test matrix ρ. This is because the decomposition of ρ is not unique.

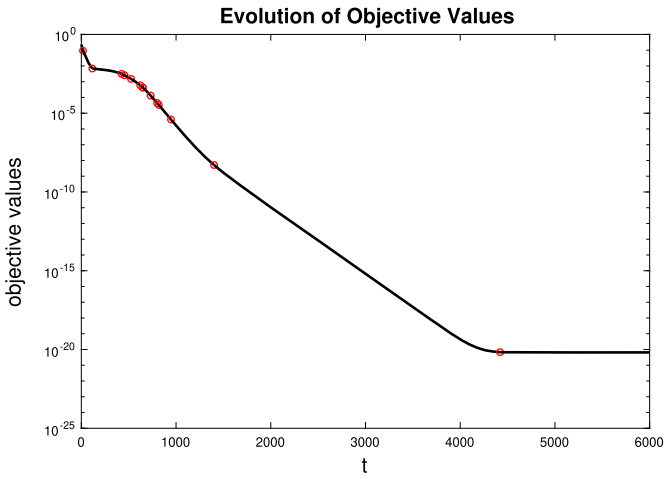
We should also point out that in some other runs with different starting values, the rank R is still reduced as expected but may be to an integer slightly larger than the original number 6. This observation indicates that the problem (6), being nonlinear, might have multiple local solutions and that the limit points depend on the starting values.

Experiment 2. This experiment is designed to examine the preservation of sum-to-one property. We generate a positive definite matrix $\rho \in \mathbb{C}^{40\times 40}$ randomly and seek its approximation by $\mathbf{x}_r \in \mathbb{C}^8$ and $\mathbf{y}_r \in \mathbb{C}^5$. Suppose we preset $R = 10$ in the approximation. Since it is unlikely that the randomly generated 40×40 matrix ρ can be separated by less than 10 components, we do not expect that the rank R will be reduced.

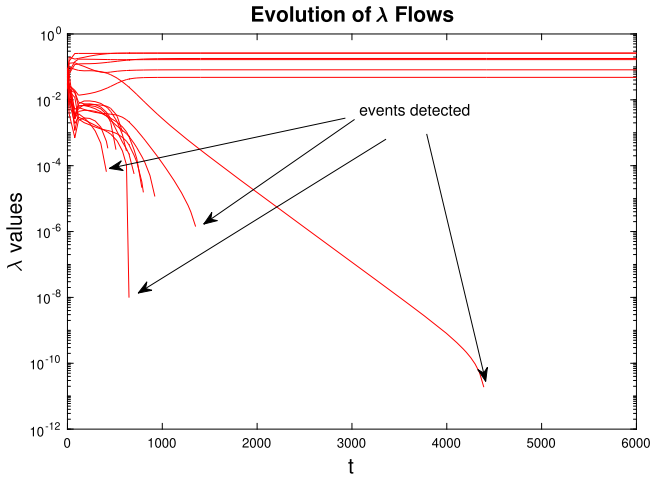
Starting with four sets of randomly generated starting values, the evolution of the objective values is plotted in Fig. 2(a). This is a hard problem in that at $t = 10^4$ the flows have not reached convergence yet, but their descent property is clear. It is also likely they will converge to different optimal values.

What is most interesting is that, despite a long time of integration, we find in Fig. 2(b) that within a fairly narrow window of approximately 10^{-8}, the property $\sum_{r=1}^{10} \lambda_r = 1$ is reasonably preserved. The non-smooth fluctuations at the beginning of each integral curve might cause misgivings but can easily be explained. First, this drawing is at a microscopic scale of 10^{-8} along the y-axis. Second, the numerical integrator is using variable step sizes along the x-axis. Such a fluctuation is within the range of expectation. This experiment confirms that our strategy for maintaining both sum-to-one and descent achieves its goal.

Experiment 3. This experiment is designed to examine the effect of R on the quality of approximation. Most ODE integrators are implemented to handle real-valued problems. To integrate our modified complex-valued gradient flow, we must deal with $(2m + 2n + 1)R$ real variables. In this experiment, we choose to work on a 4-qubit system where $d = 4$ is split as $2 + 2$. Therefore, $m = n = 2^2$ and this is an entanglement of two 2-qubit subsystems. We randomly generate a positive matrix $\rho \in \mathbb{C}^{16\times 16}$. We increase R from 5 to 20 by 1 and compare the objective values. Understandably, this is not a fair comparison because each problem requires $17R$ randomly generated starting values, which thus are unavoidably distinct, and the solution trajectories vary according to the starting values. However, assuming



(a) Descending behavior of the objective value; each circle indicates an event occurs.



(b) Dynamics of $\lambda_r(t)$, $r \in [\![20]\!]$

Fig. 1. History of objective values and variation of λ_r values.

that the trend in Experiment 2 is typical, we would like to attain a general impression on the effect when different values of R are used. Note that the number of variables for the ODEs varies from 85 to 340, so even for an entangled 2-qubit bipartite system the calculation is demanding.

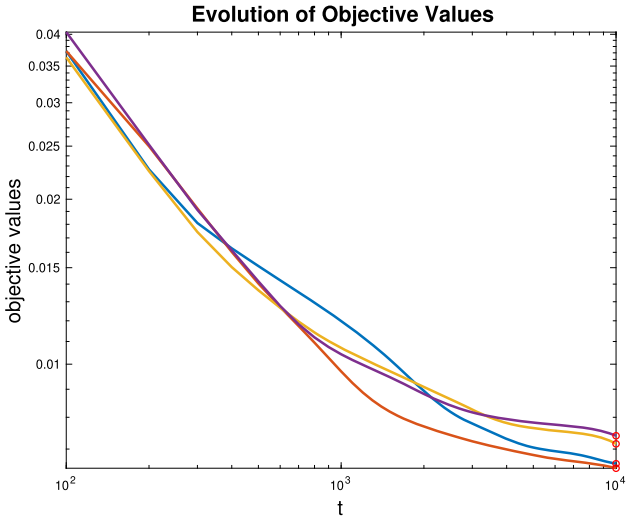
The evolution of the objective values is recorded in Fig. 3. Colors refer to different choices of R, including some restarts when an event is detected and R is reduced. Some initial segments of the curves are not shown due to the scaling. The trend of these curves is in agreement with what is already expected. What matters is the summary in Table 1 of the optimal objective values, where R_{final} denotes the final rank. We see that in order to approximate this random matrix $\rho \in \mathbb{C}^{16\times 16}$, no event happens if we start with a rank $R \leq 14$. We also notice that the optimal objective values recorded in the third row get larger when R gets lower. On the other hand, if we begin with $R \geq 15$, then events do happen and our rank reduction mechanism kicks in. Not all of the tests return the same R_{final}. In this particular case $R = 15$, we observe that the rank is reduced to 13 but with a slightly larger optimal value. This is perhaps an indication that the flow is trapped at a local solution. It seems that $R \approx 15$ is the borderline case.

Experiment 4. This final experiment is designed for two purposes. One is to examine how the dynamical approach can serve as a tool to qualify the entanglement of a realistic problem. The other is to make a comparison of the converging behavior with the DA algorithm.

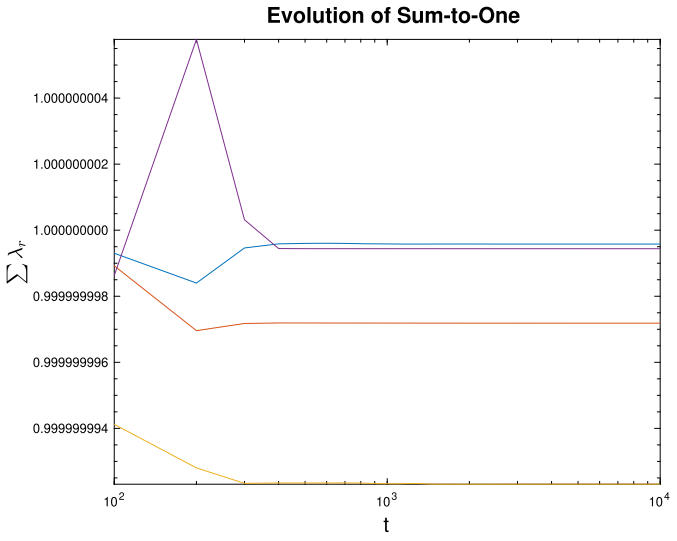
Consider the 2-qubit system $\mathbb{C}^2 \otimes \mathbb{C}^2$ with the natural basis

$$|00\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad |01\rangle = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad |10\rangle = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad |11\rangle = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \tag{32}$$

In quantum information applications, it is often prefer to use the Bell states or the EPR pairs



(a) Distinct trajectories lead to distinct objective values.



(b) Preservation of $\sum_{r=1}^{10} \lambda_r(t) = 1$.

Fig. 2. Effect of starting values and preservation of $\sum_{r=1}^{10} \lambda_r = 1$.

$$\begin{cases} |\Phi^+\rangle := \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \\ |\Phi^-\rangle := \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle), \\ |\Psi^+\rangle := \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), \\ |\Psi^-\rangle := \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle), \end{cases}$$

which exemplify perhaps the simplest but maximally entangled pure states [14,27,59–61]. With respect to the natural basis (32), the matrix representations of the states $|\Phi^+\rangle, |\Phi^-\rangle, |\Psi^+\rangle, |\Psi^-\rangle$ are

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \frac{1}{\sqrt{2}}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \frac{1}{\sqrt{2}}\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

respectively. The corresponding density matrices $\rho_{|\Phi^+\rangle} = |\Phi^+\rangle\langle\Phi^+|$ and so on should be order-4 tensors but in terms of the Kronecker product can be expressed as

$$\frac{1}{2}\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \frac{1}{2}\begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}, \frac{1}{2}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \frac{1}{2}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

respectively. We shall use the probability ensemble

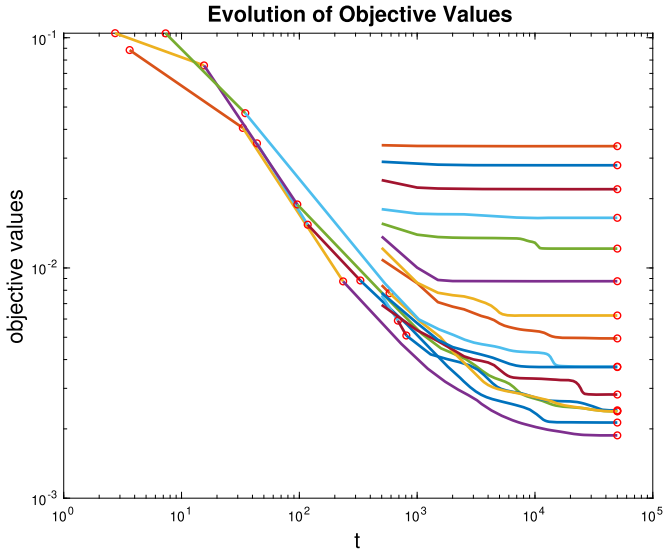


Fig. 3. Effect of R on the approximation.

Table 1
Final ranks (R_{final}) and optimal objective values (*optimal*) obtained from different values of R.

R	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
R_{final}	no change										13	15	15	15	16	17
optimal ($\times 10^{-4}$)	338	279	220	165	121	88	62	49	37	28	66	24	24	24	21	19

$$\rho := \frac{1}{5}\rho_{|\Phi^+\rangle} + \frac{2}{5}\rho_{|\Phi^-\rangle} + \frac{2}{5}\rho_{|\Psi^+\rangle} = \begin{bmatrix} 0.3 & 0 & 0 & -0.1 \\ 0 & 0.2 & 0.2 & 0 \\ 0 & 0.2 & 0.2 & 0 \\ -0.1 & 0 & 0 & 0.3 \end{bmatrix}$$

of three Bell states as our target matrix [15]. It can be checked that the density matrix ρ is still not separable [14,17].

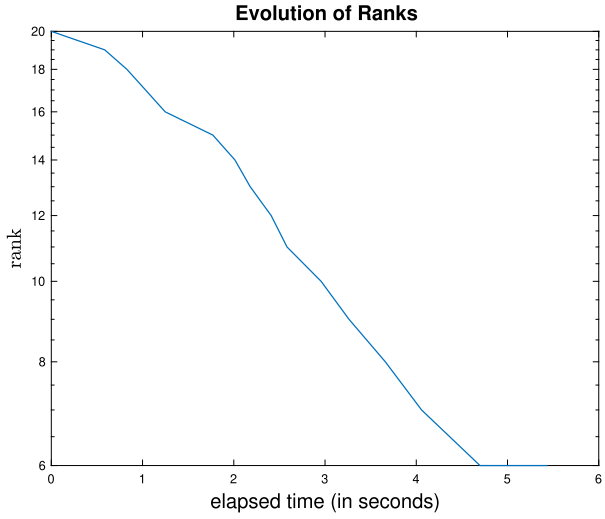
Stating from $R = 20$ and randomly generated complex-valued initial values, we call for our flow approach to estimate the nearest separable state of ρ. The evolutions of the rank reduction and objective values are plotted in Fig. 4(a) and (b), respectively. While the objective values keep descending, the rank is reduced from $R = 20$ to $R_{\text{final}} = 6$.

We are curious about how the DA algorithm proposed in [20,37] will handle this problem. Since not many implementation details are given in the original paper, we have to develop our own code by following the general ideas described in [20,37]. Our understanding is that the DA algorithm consists of the recurrence of two parts each of which is fairly involved. The first part is to solve alternatively a sequence of projective subproblems for eigenvalue maximization. This is the place where we have commented earlier in Sections 2 and after Lemma 6 that the convergence could become an issue, both theoretically and numerically. Regardless, the original stopping criterion proposed in [20] is when two successive iterates $(\mathbf{x}^{[p]} \otimes \mathbf{y}^{[p]})$ and $(\mathbf{x}^{[p+1]} \otimes \mathbf{y}^{[p+1]})$ are sufficiently aligned. In view of (20), we adopt a more rigorous stopping criterion by requiring

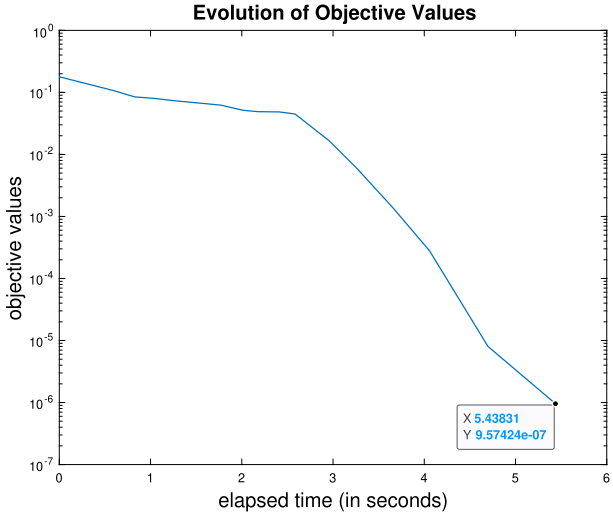
$$\left\| \begin{bmatrix} \mathscr{A}(\mathbf{y}^{[p+1]}, \overline{\mathbf{y}}^{[p+1]})\mathbf{x}^{[p+1]} - \lambda(\mathbf{x}^{[p+1]}, \mathbf{y}^{[p+1]})\mathbf{x}^{[p+1]} \\ \mathscr{B}(\mathbf{x}^{[p+1]}, \overline{\mathbf{x}}^{[p+1]})\mathbf{y}^{[p+1]} - \lambda(\mathbf{x}^{[p+1]}, \mathbf{y}^{[p+1]})\mathbf{y}^{[p+1]} \end{bmatrix} \right\|_F \le 10^{-12}$$

in our simulation. After collecting all vectors $(\mathbf{x}_r, \mathbf{y}_r)$ from the preceding calculation, the second part is to find their best possible convex combination to minimize the objective function. If any coefficient becomes zero, the corresponding term is removed; otherwise, the rank R will grow whenever new vectors from the first part of calculation are added. To facilitate the calculation, we employ the existing MATLAB optimization package to deal with the constrained minimization and mimic the rule of removing the triple $(\lambda_r, \mathbf{x}_r, \mathbf{y}_r)$ whenever $\lambda_r < 10^{-12}$ for some $r \in [\![R]\!]$. The evolutions of the rank growth and objective values are plotted in Fig. 5(a) and (b), respectively. While the objective values have the general trend of descending, the rank grows to $R_{\text{final}} = 241$.

Though the comparison is only preliminary, some observations are worth mentioning. It appears that the DA algorithm can reduce the objective value more rapidly than the gradient dynamics approach at the beginning (Fig. 4). However, it comes at the cost of increasing the rank R sharply. By the time it reaches convergence, the $R_{\text{final}} = 241$ for the DM algorithm is significantly larger the $R_{\text{final}} = 6$ for our gradient flow approach. In contrast to the gradient flow, note also that the DA algorithm cannot guarantee to decrease the objective value monotonically. This probably can be explained by the fact that the projective subproblem can only receive a local solution which, thus, propagates to a non-optimal result. In contrast, we see that the objective values obtained from the gradient flow approach keep descending



(a) Change of ranks with respect to elapsed time.



(b) Change of objective values with respect to elapsed time.

Fig. 4. Low rank approximation via the gradient dynamics.

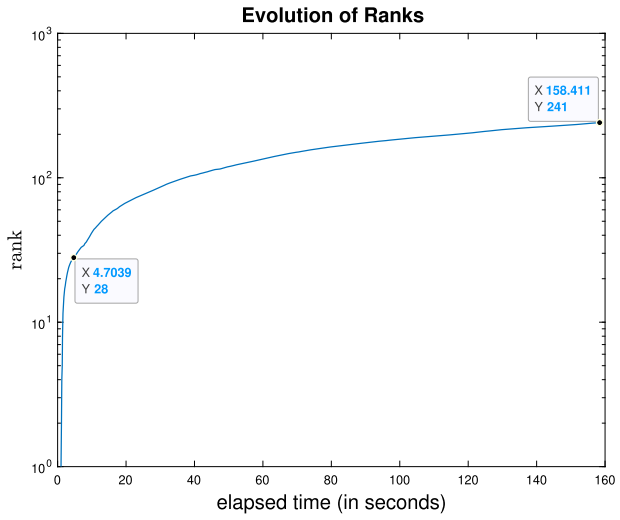
monotonically. By the time we terminate the integration, the objective value $\approx 10^{-6}$ is already one order lower than the optimal value by the DA algorithm. It suggests that this gradient flow approach is easy to implement, is more robust, and can find a better approximation at a relatively low rank.

6. Conclusion

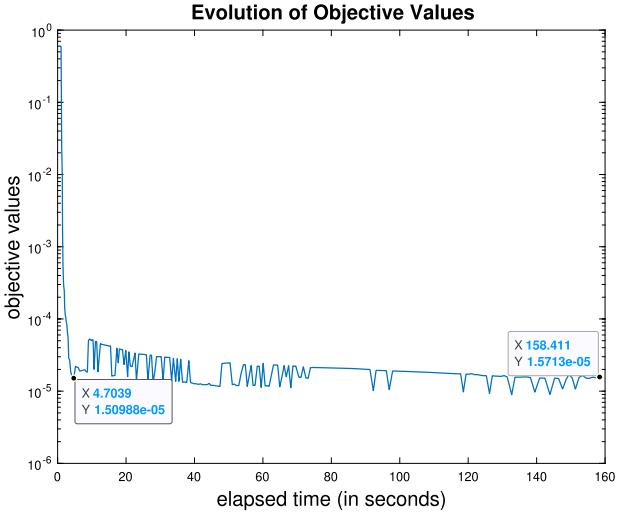
Low rank approximation for entangled bipartite quantum systems is an interesting but challenging task. It is interesting because of its potential application as a way to certify the quality of an entanglement. It is challenging because the computation is complicated by the involvement of complex variables and the curse of dimensionality. Conventional ALS-type methods generally are ineffective.

This paper describes a complex-valued gradient dynamics for the low rank approximation problem. Using the Wirtinger calculus, we effectively avoid the otherwise tedious algebraic manipulations. Any existing ODE integrators can readily be used to obtain numerical results that might help shed some light into the entangled bipartite systems. The global convergence from any starting point to a local solution is guaranteed by existing mathematical theory. Also built in the gradient flow is the capability of continuously enforcing the probability distribution of the pure states and adaptively reducing the terms used in the approximation. Exactly determining whether a density matrix is entangled or not is NP hard. Our numerical approach, though finding possibly only a local optimum, offers some quantitative information for gauging the quality of entanglement. Our technique might serve as a basic tool when exact quantification is hard to come by.

There are still many unsettled issues. Future work includes exploiting the structure of the projected gradient to gain more computational efficiency, experimenting with maximally allowable d-qubit systems subject to the memory constraint, and generalizing the technique to multipartite systems. When the latter is ready, we may also have a tool to study the effect on various splittings of a given d-qubit systems.



(a) Change of ranks with respect to elapsed time.



(b) Change of objective values with respect to elapsed time.

Fig. 5. Low rank approximation via the DA algorithm.

Declaration of competing interest

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

Acknowledgements

The first author's research is supported in part by the National Science Foundation under grant DMS-1912816, and the corresponding author's research is supported in part by the National Center for Theoretical Sciences of Taiwan and by the Ministry of Science and Technology of Taiwan under grant 110-2636-M-006-006.

References

[1] A. Einstein, B. Podolsky, N. Rosen, Phys. Rev. 47 (1935) 777–780, https://doi.org/10.1103/PhysRev.47.777.
[2] A.K. Ekert, Phys. Rev. Lett. 67 (1991) 661–663, https://doi.org/10.1103/PhysRevLett.67.661.
[3] C.H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, W.K. Wootters, Phys. Rev. Lett. 70 (1993) 1895–1899, https://doi.org/10.1103/PhysRevLett.70.1895.
[4] R. Raussendorf, H.J. Briegel, Phys. Rev. Lett. 86 (2001) 5188–5191, https://doi.org/10.1103/PhysRevLett.86.5188.
[5] J. Barrett, A. Kent, S. Pironio, Phys. Rev. Lett. 97 (2006) 170409, https://doi.org/10.1103/PhysRevLett.97.170409.
[6] N.J. Cerf, M. Bourennane, A. Karlsson, N. Gisin, Phys. Rev. Lett. 88 (2002) 127902, https://doi.org/10.1103/PhysRevLett.88.127902.
[7] S. Gröblacher, T. Jennewein, A. Vaziri, G. Weihs, A. Zeilinger, New J. Phys. 8 (5) (2006) 75, https://doi.org/10.1088/1367-2630/8/5/075.
[8] M. Huber, M. Pawłowski, Phys. Rev. A 88 (2013) 032309, https://doi.org/10.1103/PhysRevA.88.032309.
[9] M. Hayashi, Quantum Information Theory, 2nd edition, Graduate Texts in Physics, Springer-Verlag, Berlin, 2017, mathematical foundation.
[10] M. Melucci, Introduction to Information Retrieval and Quantum Mechanics, The Information Retrieval Series, vol. 35, Springer, Heidelberg, 2015.

[11] M.M. Wilde, Quantum Information Theory, 2nd edition, Cambridge University Press, Cambridge, 2017.
[12] B.M. Terhal, Theor. Comput. Sci. 287 (1) (2002) 313–335, https://doi.org/10.1016/S0304-3975(02)00139-1, natural computing.
[13] N.J. Cerf, C. Adami, R.M. Gingrich, Phys. Rev. A 60 (1999) 898–909, https://doi.org/10.1103/PhysRevA.60.898.
[14] K. Chen, L.-A. Wu, Quantum Inf. Comput. 3 (3) (2003) 193–202.
[15] R. Horodecki, P. Horodecki, M. Horodecki, K. Horodecki, Rev. Mod. Phys. 81 (2) (2009) 865–942, https://doi.org/10.1103/RevModPhys.81.865.
[16] A. Peres, Phys. Rev. Lett. 77 (8) (1996) 1413–1415, https://doi.org/10.1103/PhysRevLett.77.1413.
[17] N. Johnston, QETLAB: a MATLAB toolbox for quantum entanglement, version 0.9, https://doi.org/10.5281/zenodo.44637, Jan. 2016.
[18] N. Friis, G. Vitagliano, M. Malik, M. Huber, Nature Reviews Physics 1 (1) (2019) 72–87, https://doi.org/10.1038/s42254-018-0003-5.
[19] O. Gühne, G. Tóth, Phys. Rep. 474 (1–6) (2009) 1–75, https://doi.org/10.1016/j.physrep.2009.02.004.
[20] G. Dahl, J.M. Leinaas, J. Myrheim, E. Ovrum, Linear Algebra Appl. 420 (2–3) (2007) 711–725, https://doi.org/10.1016/j.laa.2006.08.026.
[21] S.-H. Kye, Rep. Math. Phys. 69 (3) (2012) 419–426, https://doi.org/10.1016/S0034-4877(13)60007-5.
[22] W. Thirring, R.A. Bertlmann, P. Köhler, H. Narnhofer, Eur. Phys. J. D 64 (2) (2011) 181–196, https://doi.org/10.1140/epjd/e2011-20452-1.
[23] S. Gharibian, Quantum Inf. Comput. 10 (3) (2010) 343–360.
[24] L. Gurvits, J. Comput. Syst. Sci. 69 (3) (2004) 448–484, https://doi.org/10.1016/j.jcss.2004.06.003.
[25] C.J. Hillar, L.-H. Lim, J. ACM 60 (6) (2013) 45, https://doi.org/10.1145/2512329.
[26] L. Chen, M. Aulbach, M. Hajdušek, Phys. Rev. A 89 (2014) 042305, https://doi.org/10.1103/PhysRevA.89.042305.
[27] M.A. Nielsen, I.L. Chuang, Quantum Computation and Quantum Information: 10th Anniversary Edition, Cambridge University Press, 2010.
[28] S. Aaronson, Quantum Computing Since Democritus, Cambridge University Press, Cambridge, 2013.
[29] F. Hiai, D. Petz, Introduction to Matrix Analysis and Applications, Universitext, Springer/Hindustan Book Agency, Cham/New Delhi, 2014.
[30] M. Nakahara, T. Ohmi, Quantum Computing: From Linear Algebra to Physical Realizations, CRC Press, Boca Raton, FL, 2008.
[31] R. Webster, Convexity, Oxford Science Publications, The Clarendon Press, Oxford University Press, New York, 1994.
[32] A. Ekert, P.L. Knight, Am. J. Phys. 63 (5) (1995) 415–423, https://doi.org/10.1119/1.17904.
[33] R.F. Werner, Phys. Rev. A 40 (1989) 4277–4281, https://doi.org/10.1103/PhysRevA.40.4277.
[34] E. Chitambar, C.A. Miller, Y. Shi, J. Math. Phys. 51 (7) (2010) 072205, https://doi.org/10.1063/1.3459069.
[35] L.M. Ioannou, B.C. Travaglione, D.C. Cheung, A.K. Ekert, Phys. Rev. A 70 (2004) 060303(R), https://doi.org/10.1103/PhysRevA.70.060303.
[36] M. Horodecki, P. Horodecki, R. Horodecki, Phys. Lett. A 283 (1–2) (2001) 1–7, https://doi.org/10.1016/S0375-9601(01)00142-6.
[37] J.M. Leinaas, J. Myrheim, E. Ovrum, Phys. Rev. A 74 (2006) 012313, https://doi.org/10.1103/PhysRevA.74.012313.
[38] D.P. Bertsekas, Nonlinear Programming, 3rd edition, Athena Scientific Optimization and Computation Series, Athena Scientific, Belmont, MA, 2016.
[39] R. Karam, Am. J. Phys. 88 (1) (2020) 39–45, https://doi.org/10.1119/10.0000258.
[40] M.T. Chu, M.M. Lin, SIAM J. Sci. Comput. 0 (0) (2021) S448–S474, https://doi.org/10.1137/20M1336059.
[41] B. Dong, N. Jiang, M.T. Chu, Numer. Math. 144 (4) (2020) 729–749, https://doi.org/10.1007/s00211-020-01100-8.
[42] D.S. Hochbaum, Ann. Oper. Res. 153 (2007) 257–296, https://doi.org/10.1007/s10479-007-0172-6.
[43] L. Sorber, M. Van Barel, L. De Lathauwer, SIAM J. Optim. 22 (3) (2012) 879–898, https://doi.org/10.1137/110832124.
[44] D.H. Brandwood, Proc. IEE-H 130 (1) (1983) 11–16, https://doi.org/10.1049/ip-h-1.1983.0004.
[45] A. Friedman, Foundations of Modern Analysis, Dover Publications, Inc., New York, 1982, reprint of the 1970 original.
[46] R.E. Moore, M.J. Cloud, Computational Functional Analysis, second edition, Woodhead Publishing, 2007.
[47] C.F. Van Loan, J. Comput. Appl. Math. 123 (1–2) (2000) 85–100, https://doi.org/10.1016/S0377-0427(00)00393-9.
[48] C.F. Van Loan, N. Pitsianis, in: Linear Algebra for Large Scale and Real-Time Applications, Leuven, 1992, in: NATO Adv. Sci. Inst. Ser. E Appl. Sci., vol. 232, Kluwer Acad. Publ., Dordrecht, 1993, pp. 293–314.
[49] U. Helmke, J.B. Moore, Optimization and Dynamical Systems, Communications and Control Engineering Series, Springer-Verlag London, Ltd., London, 1994, with a foreword by R. Brockett.
[50] K. Kurdyka, T. Mostowski, A. Parusiński, Ann. Math. (2) 152 (3) (2000) 763–792, https://doi.org/10.2307/2661354.
[51] S. Wiggins, Introduction to Applied Nonlinear Dynamical Systems and Chaos, 2nd edition, Texts in Applied Mathematics, vol. 2, Springer-Verlag, New York, 2003.
[52] R. Chill, J. Funct. Anal. 201 (2) (2003) 572–601, https://doi.org/10.1016/S0022-1236(02)00102-7.
[53] S. Łojasiewicz, in: Geometry Seminars, 1982–1983, Bologna, 1982/1983, Univ. Stud. Bologna, Bologna, 1984, pp. 115–117.
[54] S. Łojasiewicz, M.-A. Zurro, Bull. Pol. Acad. Sci., Math. 47 (2) (1999) 143–145.
[55] P.-A. Absil, R. Mahony, B. Andrews, SIAM J. Optim. 16 (2) (2005) 531–547, https://doi.org/10.1137/040605266.
[56] L.F. Shampine, S. Thompson, J.A. Kierzenka, G.D. Byrne, Appl. Math. Comput. 170 (1) (2005) 556–569, https://doi.org/10.1016/j.amc.2004.12.011.
[57] R. Ashino, M. Nagase, R. Vaillancourt, Comput. Math. Appl. 40 (4–5) (2000) 491–512, https://doi.org/10.1016/S0898-1221(00)00175-9.
[58] L.F. Shampine, M.W. Reichelt, SIAM J. Sci. Comput. 18 (1) (1997) 1–22, https://doi.org/10.1137/S1064827594276424, dedicated to C. William Gear on the occasion of his 60th birthday.
[59] J.S. Bell, Phys. Phys. Fiz. 1 (1964) 195–200, https://doi.org/10.1103/PhysicsPhysiqueFizika.1.195.
[60] M. Horodecki, P. Horodecki, R. Horodecki, Phys. Lett. A 223 (1–2) (1996) 1–8, https://doi.org/10.1016/S0375-9601(96)00706-2.
[61] N. Chandra, Quantum Entanglement in Electron Optics: Generation, Characterization, and Applications, Springer Series on Atomic, Optical, and Plasma Physics, Springer, Berlin, 2013.