IAM . CI. COMPUT.
. 26, N . 6, . 2133 2159

c 2005 L A M

ALGORITHMS FOR NUMERICAL ANALYSIS IN HIGH DIMENSIONS

form (1.1) is not good enough, then there is no way within this framework to improve the accuracy.

We use the natural extension of separation of variables

$$f(x_1,\dots,x_d)=\sum_{l=1}^{r} s_l\,\phi_1^l(x_1)\cdots\phi_d^l(x_d)+\mathcal{O}(\epsilon), \tag{1.2}$$

which we call a *separated representation*. We set an accuracy goal ϵ first, and then adapt $\phi_i^l(x_i)$, s_l, and r to achieve this goal with minimal separation rank r. The separated representation seems rather simple and familiar, but it actually has a surprisingly rich structure and is not well understood. It is not a projection onto a subspace, but rather a nonlinear method to track a function in a high-dimensional space while using a small number of parameters. In section 2 we develop the separated representation, extending the results in [3] and making connections with other results in the literature. We introduce the concept of the condition number of a separated representation, which measures the potential loss of significant digits due to cancellation errors. We provide analysis and examples to illustrate the structure of this representation, with particular emphasis on the variety of mechanisms that allow it to be surprisingly efficient. Note, however, that the theory is still far from complete.

Many linear algebra operations can be performed while keeping all objects in the form (1.2). We can then perform operations in d dimensions using combinations of one-dimensional operations, and so achieve computational complexity that formally scales *linearly* in d. Of course, the complexity also depends on the separation rank r. The optimal separation rank for a specific function or operator is a theoretical question, and is considered in section 2. The practical question is how to keep the separation rank close to optimal during the course of some numerical algorithm. As we shall see, the output of an operation, such as matrix-vector multiplication, is likely to have larger separation rank than necessary. If we do not control the separation rank, it will continue to grow with each operation. In section 3 we present an algorithm for reducing the separation rank back toward the optimal, and we also present a modified algorithm that avoids ill-conditioned representations. Although the modification required is very simple, it makes the overall algorithm significantly more robust.

In order to use the separated representation for numerical analysis applications, many algorithms and operations need to be translated into this framework. Basic linear algebra operations, such as matrix-vector multiplication, are straightforward and were described in [3], but other operations are not as simple. In section 4 we continue to expand the set of operations that can be performed within this framework by showing how to solve a linear system. Many standard methods (e.g., Gaussian elimination) do not make sense in the separated representation. We take two approaches to solving a linear system. First, we discuss how to use iterative methods designed for large sparse matrices, such as conjugate gradient. Second, we present an algorithm that formulates the system as a least-squares problem, combines it with the least-squares problem used to find a representation with low separation rank, and then solves this joint problem by methods similar to those in section 3. We also discuss how these two general strategies can be applied to problems other than solving a linear system.

One of our target applications is the representation and computation of wavefunctions of the multiparticle Schrödinger equation in quantum mechanics. These wavefunctions have the additional constraint that they must be antisymmetric under exchange of variables, a condition that seems to preclude having low separation rank. In section 5 we present the theory and algorithms for representing and computing with such antisymmetric functions. We construct an antisymmetric separation-rank reduction algorithm, which uses a pseudonorm that is only nonzero for antisymmetric functions. This algorithm allows us to guide an iterative method, such as the power method, to converge to the desired antisymmetric solution.

We conclude the paper in section 6 by briefly describing further steps needed for the development of this methodology.

2. The separated representation. In this section we introduce the separated representation and discuss its properties. In order to emphasize the underlying physical dimension, we define operators and functions in d dimensions. To avoid confusion between, e.g., a "vector in two dimensions" and a "matrix," we clarify our notation and nomenclature. A [illegible] f [illegible] d is a map f : R $\to$ R from d-dimensional Euclidean space to the real numbers. We write f as $f($[illegible]$_1, \ldots,$[illegible]$)$, where [illegible]$_i$ R. A [illegible] F [illegible] d is a discrete representation of a function in dimension d on a rectangular domain. We write it as F = $F(\ _1, \ldots, \ )$, where $\ _i = 1, \ldots, M_i$. A linear [illegible] d is a linear map : $S \to S$, where S is a space of functions in dimension d. A [illegible] A [illegible] d is a discrete representation of a linear operator in dimension d. We write A = $A(\ _1, \ '_1; \ldots; \ , \ ')$, where $\ _i = 1, \ldots, M_i$ and $\ '_i = 1, \ldots, M'_i$. For simplicity we assume $M'_i = M_i = M$ for all .

[illegible] .1 (separated representation of a vector). [illegible] F = $F(\ _1, \ _2, \ldots, \ )$ [illegible] d

$$F_1(\ )F_2(\ ) \quad F$$

2 Tf 0 -2.16[7.6ptcj /F11 1 Tf 6.97387-339943 (as)Tj /F11 1 Tf 6.9738 0 0 6.9738 138.1488 4266 273 Tm (r)Tj /F14 1 Tf 9.9626 0 0 9.9626 132.91

=1

In order to maintain significant digits when using the representation (2.1), we cannot allow to be too large. In particular, to achieve (2.2) numerically it suffices to have

$$\mathrm{F} \; < , \tag{2.4}$$

where is the machine roundoff (e.g., 10^{-16}).

The main point of the separated representation is that since we only operate on one-dimensional objects, the computational complexities are formally linear in d rather than exponential. The key, however, is to determine any hidden dependency of on d. We demonstrated in [3], and discuss further in this paper, that in many physically significant problems the separation rank depends benignly on d, so that near-linear complexities can be achieved in practice. We show next how to do the basic operations of addition, inner product, and matrix-vector multiplication. Other basic operations such as scalar multiplication, trace, Frobenius norm, matrix-matrix multiplication, etc. follow a similar pattern. The following statements can be easily verified.

[illegible] (basic linear algebra).

[illegible] $(d \quad M)$ [illegible].

[illegible] $\hat{\mathrm{F}} + \tilde{\mathrm{F}}$: [illegible]

2.1. Analysis and examples. In dimension $d = 2$, the separated representation (2.1) of a vector $F({}_{1}$

2.1.1. Example: Sine of a sum. We next consider an elegant example that illustrates several phenomena we have observed in separated representations.

One early numerical test of the separated representation was to consider a sine wave in the diagonal direction, $\sin(x_1 + \cdots + x_d)$, and attempt to represent it in the separated form, using only real functions. We can use the usual trigonometric formulas for sums of angles to obtain a separated representation, but then we will have restricted to make them bounded. The condition number (Definition 2.2) of (2.11), however, is = (1/), and such a representation may be unusable numerically. Accounting for conditioning, we obtain the following theorem.

> 0 … $0 <$ 1 … . A … d $<$, … i i …

$$(2.12)\qquad = \frac{\log(d\quad /\ )}{\log(1/\ )\ \log(d\quad /\ )}\ .$$

. Consider the auxiliary operator-valued function of the real variable ,

$$(2.13)\qquad (\ ) = \sum_{i=1} \mathrm{I}_i + \frac{\ \ _i}{\ \ } \ ,$$

and note that $'(0) = \sum_i \ _i$. Using an appropriate finite difference formula of order , we approximate

$$(2.14)\qquad '(0) \approx \sum_{=1} (\ ) = \sum_{=1} \prod_{i=1} \mathrm{I}_i + \frac{\ \ _i}{\ \ } \ ,$$

thus providing a separation-rank approximation. If we choose equispaced with stepsize , then the truncation error of this finite difference can be made proportional to $(\ ^r/\ !)\ ^{(r+1)}(\ )$, where (see, e.g., [23]). Pulling out the norm as we did in (2.13) allows us to choose $=\ /d$ for some < 1 and bound the truncation error by d ^{r}. The error due to finite precision arithmetic and loss of significance is proportional to $/\ = d\ /\ $. Adding these two errors and choosing $= (\ d/\ )^{1/(r+1)}$ yields the bound $d\ \ ^{r/(r+1)}$. Setting this equal to and solving for , we obtain (2.12). □

The estimate (2.12) implies that, as long as d / 1/ , the separation rank is (log(d /)).

This example illustrates that

> low separation-rank can sometimes be achieved at the expense of (reasonably) increasing the condition number. For problems in high dimensions it is an excellent trade-off.

2.1.3. Example: Exponential expansions using quadratures. We next consider an example of a methodology for constructing separated representations, built upon the exponential. The exponential function converts sums into products via $e^{\ _1+\ _2+\cdots+\ _d} = e^{\ _1}e^{\ _2}\cdots e^{\ _d}$, valid as long as a_i commute. In this section a_i will be a function or operator in the direction , such as x_i^2 or $\partial^2/\partial x_i^2$.

Suppose we wish to find a separated representation for the radial function $f(\ \mathbf{x}\ )$ supported on the ball of radius 1 in dimension d. Since physical forces often depend

By substituting $\sum_i x_i^2$ for ^{2} and using the properties of exponentials, we obtain a separated representation for $f(\ \mathrm{x}\ )$ on the ball. In this case we obtain a pointwise relative error bound instead of (2.2). We thus have reduced the problem of finding a separated representation for a radial function in dimension d to the one-dimensional approximation problem (2.15). Usually the minimal to satisfy (2.15) is not the optimal separation rank for $f(\ \mathrm{x}\ )$, but it does provide an excellent upper bound.

The approximation problem (2.15) is addressed in [5] by extending methods in [4]. For certain choices of $f(\ )$ there is a systematic way to approximate it with small , even if the function has a singularity at zero. For example, let us consider $f(\ ) = \ ^{-}$ for > 0.

[illegible] (see [5]). *F*[illegible] > 0, $0 < \ < 1$, [illegible] $0 < \ \le \min \frac{1}{2}, \frac{8}{}$ [illegible] $\le \ \le 1$

$$\left| \frac{1}{} - \sum_{=1} \ e^{-\ _l\ ^2} \right| \le \frac{}{}, \tag{2.16}$$

[illegible]

$$= \log\ ^{-1} \left[c_0 + c_1 \log\ ^{-1} + c_2 \log\ ^{-1} \right], \tag{2.17}$$

[illegible] c_0, c_1, [illegible] c_2 [illegible]. *F*[illegible] $= \ (\log\ ^{-1})$.

The construction uses the integral representation

$$\frac{1}{} = \frac{2}{\Gamma(\ /2)} \int_{-\infty}^{\infty} e^{-\ ^2\ ^{2t} + \ t} d\ , \tag{2.18}$$

as described in [19, 20]. For a given accuracy, the rapid decay of the integrand restricts integration to a finite interval. Using the trapezoidal rule for an appropriately selected interval and stepsize yields a separated representation (see [19, 20]). The resulting representation is then optimized further using the results in [5].

This approach shows that

> there is a general, semianalytic method (based on representations with exponentials) to compute separated representations of radial functions.

These representations for radial functions can be used to construct representations for functions of operators. In particular, we can substitute an operator of the form $\sum_i \ _i$, such as the Laplacian, for ^{2} in (2.15), and obtain a separated representation for $f((\sum_i \ _i)^{1/2})$, valid on a portion of its spectrum. Using Lemma 2.6 with $= 2$ and scaling to an appropriate interval, we can obtain a separated representation for the inverse Laplacian.

[illegible] Δ^{-1}, [illegible] $(\sum_{i=1} \ _i^2)^{1/2}$ [illegible] $[\ D, D]$ [illegible] (2.17).

By applying the Fourier transform, we can obtain the corresponding Green's function

$$\frac{1}{(d - 2)\Omega} \frac{1}{\ ^{\ -2}} \to \sum_{=1} \ \prod_{i=1} \ \frac{1}{4} \exp(-\ _i^2/4\ ), \tag{2.19}$$

where Ω is the surface area of the unit sphere in dimension d.

Notice that the bound in Lemma 2.7 is independent of d. For periodic problems it is more natural to construct a representation on a cube rather than a ball. In dimension d the cube inscribed in the unit ball has side length $2d^{-1/2}$. If we compensate for this effect to maintain a cube of fixed side length, then the separation rank grows as $(\log(d))$.

Robert Harrison (personal communication, 2003) pointed out to us an example that produces a separated representation for a multiparticle Green's function without going through a Fourier-space construction. Following [26], we consider the multiparticle operator ▼ $\ _{i=1}\Delta_i$, where Δ_i is the Laplacian in dimension three. The Green's function for this operator is

$$G(\mathrm{x}) = \frac{1}{(2\ )^{3\ /2}}\,\frac{K_{3\ /2-1}(\ \mathrm{x}\ )}{\mathrm{x}^{\ 3\ /2-1}}\,, \tag{2.20}$$

where $\mathrm{x} = (r_1, r_2, \ldots, r\ )$, r_i R^3, and K is the modified Bessel function. Using its integral representation [18, equation (8.432.6)]

$$K\ (\ ) = \frac{1}{2}\ \frac{}{2}\ \ _0^{\infty}\ -\ ^{-1} e^{-t-\ ^2/4t} d\ , \tag{2.21}$$

we have

$$G(\mathrm{x}) = \ ^{3\ /2}\ _0^{\infty}\ ^{-3\ /2} e^{-t-\|\mathbf{x}\|^2/4t} d\ , \tag{2.22}$$

and changing variables $= e^{-2\ }$, we obtain

$$G(\mathrm{x}) = 2\ ^{3\ /2}\ _{-\infty}^{\infty} e^{-\ ^{-2s}+(3\ -2)\ } e^{-\|\mathbf{x}\|^2\ ^{2s}/4} d\ . \tag{2.23}$$

The integral (2.23) is similar to that in (2.18) and, using a similar approach to that in Lemma 2.6, we obtain a separated representation for the Green's function $G(\mathrm{x})$ with

The class of [illegible] functions is considered in [16, 43, 42]. Theorem 1 in [43] says essentially that error [illegible] =

We start with F, an initial approximation of G in (3.1),

$$F = \sum_{\tilde{}=1}^{F} {}^{F}_{\tilde{}} F_1^{\tilde{}} \quad F_2^{\tilde{}} \qquad F^{\tilde{}} . \tag{3.2}$$

If we have no other information, then we start with a random vector with ${}_{F} = 1$. If we are performing an iteration such as the power method, then we use the previous iterate as our starting guess. We then call the core algorithm described in section 3.1, and it improves the approximation F without changing ${}_{F}$. We exit the entire separation-rank reduction successfully if $F\blacktriangledown \; G \; < \;$. If not, we either call the core routine again and repeat the process, or decide that ${}_{FF\blacktriangledown}$ ’

F

Then for a fixed coordinate , form the vector b , with entries

[redacted]

$$b_k(\hat{\ }) = \sum_{=1} {}^{\mathbf{G}}G(\ ) \prod_{i\neq} \ldots . \tag{3.4}$$

The normal equations for the direction and coordinate become

$$\mathsf{B}\mathsf{c}_k = \mathsf{b}_k, \tag{3.5}$$

which we solve for $\mathsf{c}_k = c_k(\tilde{\ })$ as a vector in $\tilde{\ }$. After computing $c_k(\tilde{\ })$ for all , we let ${}^{\mathbf{F}}_{\sim} = c_k(\tilde{\ })$ and $\tilde{F}(\ ) = c_k(\tilde{\ })/ {}^{\mathbf{F}}_{\sim}$, where the norm is taken with respect to the coordinate .

For fixed and , it requires ${}^{2}_{\mathbf{F}}\ d\ M$ operations to compute B, ${}_{\mathbf{F}}\ {}_{\mathbf{G}}\ d\ M$ operations to compute b_k, and ${}^{3}_{\mathbf{F}}$ to solve the system. Since B and the inner products in b_k are independent of , the computation for another value of has incremental cost ${}_{\mathbf{G}}\ {}_{\mathbf{F}} + {}^{2}_{\mathbf{F}}$. Similarly, many of the computations involved in B and b_k are the same for different . Thus, one full alternating least-squares iteration costs $(d\ {}_{\mathbf{F}}({}^{2}_{\mathbf{F}} + {}_{\mathbf{G}}\ M))$. Because this algorithm uses inner products, which can only be computed to within roundoff error , the best accuracy obtainable is $= \bar{\ }$.

3.2. Controlling the condition number. Several of the most efficient mechanisms for producing low separation-rank representations exhibit ill-conditioning (see sections 2.1.1 and 2.1.2), and, thus, we need to control the condition number . Instead of just trying to minimize $\mathbf{F}\blacktriangledown\ \mathbf{G}$, we add a penalty based on and minimize

Without the modification in section 3.2, we indeed see that as we iterate using alternating least squares, increases until the representation is untenable numerically. With the modification, the representation stabilizes at a poor, but tenable, representation, which can then be "grown" into a good approximation. For example, when d = 10 and = 9, and we attempt to force an ill-conditioned representation by performing 1000 iterations, a typical result is approximation error = 0.055 and = 1.3 10^5. When we then allow = 10, we achieve an approximation with = 1.11 10^{-4} and = 1.9 10^4. (Choosing equally spaced in the identity leads to 7.) By allowing = 11, which is more than needed in the identity, we achieve = 1.58 10^{-7} and = 1.3 10^2. Our conclusion from many experiments of this type is that with the modification in section 3.2, the alternating least-squares algorithm is robust enough to use routinely, and provides close-to-optimal separation rank.

4. Solving a linear system.

In this section we discuss how to solve the linear system AF = G for F, where all objects are in the separated representation. One of the standard methods for solving a linear system is Gaussian elimination (LU factorization). In the separated representation, however, we do not act on individual entries in a d-dimensional matrix, so it is not clear if there is a generalization of this approach.

The situation is better with iterative algorithms. The first approach is to apply one of the iterative methods designed for large sparse systems. We also use this opportunity to describe how to combine other iterative algorithms with the separated representation. The second approach is to formulate the system as a least-squares problem, combine it with the least-squares problem used to find a representation with low separation rank, and then solve this joint problem by methods similar to those in section 3. This approach incorporates a separation-rank constraint into the formulation of the problem, and can serve as a model for how to approach other problems. We give a numerical example to illustrate this algorithm.

4.1. Iterative algorithms.

Under the assumption that A <

vectors, the performance varied from run to run. We report a typical result for each test. All tests were performed on a laptop with a 1.7 GHz processor, had $d = 20$ and $M = 30$, and requested approximation within $= 10^{-6}$. For simplicity of explanation, we used very low separation ranks, but as long as we avoid the separable case ($_{\mathbf{F}_0} = 1$ or $_{\mathbb{A}} = 1$; see below), these tests illustrate the general behavior.

The purpose of the first test was to check the correctness of the algorithm for random inputs. We generated a random F_0 with $_{\mathbf{F}_0} = 2$ and a random A with $_{\mathbb{A}} = 6$, 1

Achieved errors and cumulative time used (in seconds) for solving a linear system involving the Laplacian in dimension .

$r_{\mathbf{F}}$	A$\mathbf{F}$ − $\mathbf{G}$ / $\mathbf{G}$	
	. . $^{-2}$	
	. . $^{-3}$	
	6. . $^{-4}$	
	. . $^{-5}$	'
	. . $^{-6}$	
	. . $^{-7}$	6

two iterations at each value of $_{\mathbf{F}}$, and present the achieved error and time used for selected $_{\mathbf{F}}$ in Table 1.

5. Antisymmetry. Motivated by the goal of computing the wavefunctions of the multiparticle Schrödinger equation, in this section we describe how to deal with functions that satisfy the antisymmetry constraint. This constraint is that the function must be odd under the exchange of any pair of variables (i.e., $f(x_1, x_2) = -f(x_2, x_1)$). We first sketch in section 5.1 the electronic N-particle Schrödinger equation in quantum mechanics. Since electrons are fermions, we are interested in the antisymmetric solutions, which we call wavefunctions. Although our motivation is the Schrödinger equation, the technique presented in this section demonstrates how a weak formulation can be used to represent functions that otherwise would have excessively large separation rank.

The straightforward construction of an antisymmetric function (via Slater determinants) yields separation rank $(N!)$. We avoid this problem by instead representing a "proto-wavefunction," that is, a function whose antisymmetrization yields the wavefunction itself. A representation of a proto-wavefunction allows us to do operations involving the wavefunction, such as computing its inner product with arbitrary functions, without constructing the wavefunction explicitly. The tools that make such an approach possible are well known in physics (e.g., Löwdin rules), and we review them in section 5.2.

The key to our approach is how we guide the algorithm for solving the electronic multiparticle Schrödinger equation to an antisymmetric solution. We work within the simplest solution algorithm, the power method applied to the shifted Hamiltonian. We modify the power method to find the largest eigenvalue that has an antisymmetric eigenfunction, rather than the largest eigenvalue overall. Applying the projector onto antisymmetric functions after each iteration would accomplish this goal, but so each electron has an additional discrete spin variable taking the values $\blacktriangledown \frac{1}{2}, \frac{1}{2}$. We denote the combined variables (,) by γ . Without changing our basic formalism, we will consider the combined variable γ to be a single direction when using the separated representation.

The Hamiltonian operator for the multiparticle Schrödinger equation is the sum of three terms, = + + . The kinetic energy term $= \blacktriangledown \frac{1}{2} \;\; {}^2$ is defined by

$$\blacktriangledown^2 = (\Delta_1 \quad {}_2 \quad\quad {}_N) + \quad + (\ {}_1 \quad\quad \Delta_N), \tag{5.1}$$

where the three-dimensional Laplacian

$$\Delta_i = \frac{\partial^2}{\partial \ {}_i^2} + \frac{\partial^2}{\partial \ {}_i^2} + \frac{\partial^2}{\partial \ {}_i^2} \tag{5.2}$$

corresponds to electron . The nuclear potential portion is given by

$$= (\ {}_1 \quad {}_2 \quad\quad {}_N) + \quad + (\ {}_1 \quad\quad {}_N), \tag{5.3}$$

where ${}_i$ is the operator that multiplies by the function (${}_i$), which includes nuclear potential terms as well as any external potentials. The electron-electron interaction portion of the Hamiltonian is defined by

$$= \overset{N\square 1}{\underset{i=1}{}} \ \overset{N}{\underset{m=i+1}{}} \quad {}_{im}\,, \tag{5.4}$$

where ${}_{im}$ is multiplication by the electron-electron interaction (Coulomb) potential $(\ {}_i \blacktriangledown \ {}_m) = c/\ {}_i \blacktriangledown \ {}_m$.

The antisymmetric eigenfunctions of represent electronic states of the system and we refer to them as "wavefunctions." The bound-state wavefunctions have negative eigenvalues. The ground-state multiparticle Schrödinger problem is to compute the wavefunction with the most negative eigenvalue. We note that the most negative eigenvalue has an eigenspace with no antisymmetric eigenfunctions, and thus is not associated with a wavefunction.

We need to discretize the operator to form its matrix representation H. For the purposes of this paper the particular choice of discretization is not important, although it is very important for actual implementations. We let M denote the number of degrees of freedom used for each electron, and, as we will see later, the antisymmetry constraint will force $N \;\; M$. We also need to put H in the separated representation, which we do using techniques based on Theorem 2.5. A partial analysis of this construction, without the antisymmetry condition, appeared in [3], and a more detailed analysis is in progress.

To construct the wavefunction we will use the power method. The power method repeatedly applies a given matrix A to a test vector F_0, using the iteration

$$\begin{aligned} \mathsf{G}_m &= \mathsf{A}\mathsf{F}_m, \\ \mathsf{F}_{m+1} &= \mathsf{G}_m / \ \mathsf{G}_m \ , \qquad = 0, 1, \ldots . \end{aligned} \tag{5.5}$$

If A is diagonalizable and has a distinct eigenvalue largest in magnitude, then for arbitrary F_0, the iterates F_m will converge to the corresponding eigenvector (up to a sign). To use H within the power method, we choose $c \quad \mathsf{H}$ and set $\mathsf{A} = c\mathsf{I} \blacktriangledown \mathsf{H}$, so that the eigenvalue that we want is positive and largest in magnitude.

Matrix-vector multiplication produces a vector with separation rank $\mathbb{A}\ \mathbf{F}$. As we iterate within the power method, the separation rank of F_m, if unattended, would grow rapidly. To avoid this, we apply the separation-rank reduction after each iteration. The power method, however, does not take into account the antisymmetry constraint on the wavefunction. We do not seek the largest eigenvalue of A, but rather the largest eigenvalue that has an antisymmetric eigenfunction. In the following sections we describe how to incorporate this antisymmetry constraint.

5.2. **The antisymmetrizer and the Slater determinant.** Given a function of N

5.3. Antisymmetric separation-rank reduction. If we applied the power method to A instead of A, it would produce the wavefunction. As noted above, applying has complexity $(N!)$, and produces a vector with separation rank $(N!)$. In this section we show how to apply the power method to A, while never actually applying . The key is to incorporate into the separation-rank reduction algorithm, and then use (5.8) to evaluate its effect.

The separation-rank reduction algorithm in section 3 is, at heart, the minimization of F▼ G with G fixed and a constraint on the separation rank of F. The algorithm in section 4.3 for solving a linear system is, at heart, the minimization of AF▼ G . We now use the same principle to construct an antisymmetric separation-rank reduction algorithm that minimizes (F▼ G) . We accomplish this without applying directly by using the pseudonorm $_{\mathcal{A}}$ = ( ) for the approximation error bound (2.2) and the power method normalization (5.5).

Let us drop the index in the power method and consider the problem of reducing the separation rank of G = G$_{m}$/ G$_{m}$ to obtain F = F$_{m+1}$. We begin with a fixed vector G and an approximation F, and will again fix a direction and refine in that direction, as in section 4. For simplicity we describe the = 1 case. A straightforward calculation, which we omit, produces the normal equations for this linear least-squares problem. We form the matrix B with entries

$$B((\ ,\ ),(\ ,\ )) = \left|\begin{array}{ccc} & \hat{F}_2(\ ) & \hat{F}_N(\ ) \\ (\ ) & \hat{F}_2(\ ) & \hat{F}_2(\ ) \end{array}\right|$$

0 6.9738 266.751 453.9237 Tm ( )Tj /F11 0089 0.2039 TD (l)Tj/31.0238 516.8082 Tm (m)Tj3/F14803Tf 9.9626 0 0( )F10 1 Tf 0.00.46

1 column0.9171.7(and)-365291n/ F1291n
▼

[illegible] 2153

on the left and ∪ on the right, and then remove the first N▼ 1 columns and rows, which now repH(-365s)-365.7(anu01.8(v)27.8(e)-36n)-36501.8(v)2 17.50(-365s)nSimilarly.7(40on)nu0N

the zero pivot occurred, and thus remove it. The second term in (5.14) is modified to compensate for these changes, but then the same general procedure is followed.

If a zero pivot is detected before the last step, then we can conclude that $A(\, , \,) = 0$ for all $(\, , \,)$ and not compute them. One way to see this is to note that the rows of E span a subspace of dimension at most $N - 3$. Augmenting with one extra coordinate via x_i can increase the span to dimension $N - 2$, and including the vector $[\; _i \; \mathrm{w} \;]$ can increase the span to dimension $N - 1$, but this still leaves a singular matrix.

5.4. Numerical example: Schrödinger wavefunction. In this section we illustrate the effect of the antisymmetric separation-rank reduction by computing the wavefunction for an academic model of the multiparticle Schrödinger equation with one-dimensional particles and simplified potentials. Our model is certainly not realistic, but the results do show some similarities to phenomena observed in physics. We provide tools to interpret the results, and show that the wavefunctions that we compute are consistent with the intuition developed by CI methods (see, e.g., [39]).

For our example we will let $\gamma = x$ be a one-dimensional, periodic, spinless variable. We choose the nuclear potential $(x) = c \; \cos(2 \; x)$ and the electron interaction potential $(x_i - x_m) = c \; \cos(2 \; (x_i - x_m))$. Our Hamiltonian is thus

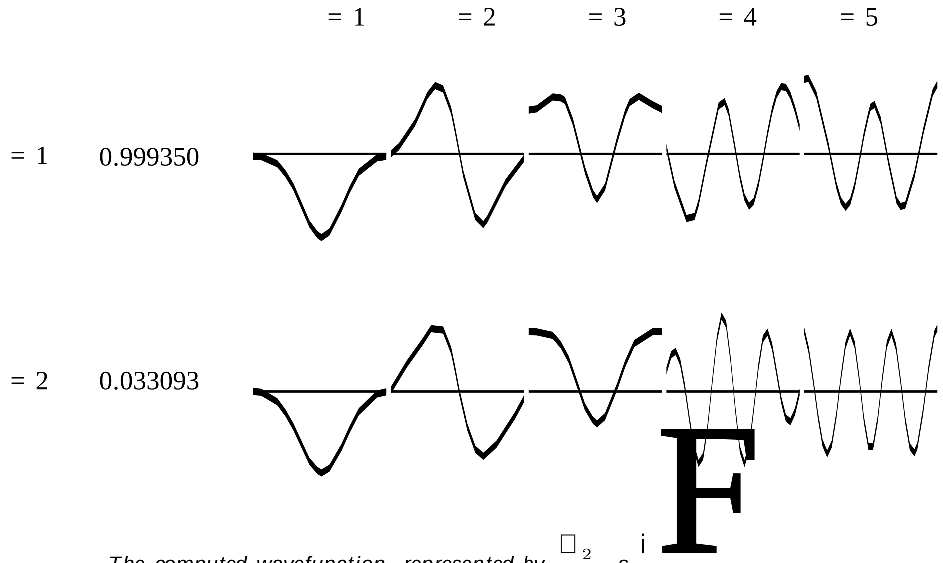
$$= \; + c \sum_{i=1}^{N} \cos(2 \; x_i) + c \sum_{i=1}^{N-1} \sum_{m=i+1}^{N} \cos(2 \; (x_i - x_m)). \tag{5.15}$$

We discretize by sampling the variable x at M equispaced points to form the vector x. The second derivatives $\partial^2/\partial x^2$ in are represented with a 9-point centered finite difference with order 8, which we denote D^2. We choose the shift $c \quad \mathsf{H} \; /2$ and set $\mathsf{A} = c\mathsf{I} - \mathsf{H}$. We combine $c\mathsf{I} - \mathsf{T} - \mathsf{V}$ into $\sum_i^N [(c/N)\mathsf{I}_i - \mathsf{D}_i^2 - c \; \cos(2 \; \mathrm{x}_i)]$, and then represent it using the construction in (2.14) with stepsize and a -point finite difference in the auxiliary parameter. The electron interaction is first separated using a trigonometric identity as $\cos(2 \; (\mathrm{x}_i - \mathrm{x}_m)) = \cos(2 \; \mathrm{x}_i) \cos(2 \; \mathrm{x}_m) - \sin(2 \; \mathrm{x}_i) \sin(2 \; \mathrm{x}_m)$, and then each term is represented using a second derivative version of (2.14) with stepsize and a -point finite difference. We use the parameters $N = 5$, $M = 30$, $c = 100$, $c = 5$, $c = 14000$, $= 0.1$, $= 6$, $= 0.1$, and $= 5$, which were chosen not for realism, but simply to provide an elegant example. With these parameters, we have a separation rank $= 16$ approximation for A with relative error less than 10^{-7}. As discussed in the previous sections, given a working precision of 10^{-16}, an allowance for the conditioning of the representation, and the need to compute square roots in order to compute norms, this is the smallest error that we are able to measure.

We first construct a separable approximation F_0 to the wavefunction by running the power method algorithm while forcing the separation-rank reduction algorithm to yield the best approximation with separation rank one. For separable functions, one does not need the full antisymmetric separation-rank reduction, so we instead orthogonalize the vectors F_i^1 after the ordinary reduction. By always orthogonalizing in the order of increasing , the vectors naturally order themselves from low to high "energy" (frequency). We conjecture that this process produces the Hartree–Fock solution, but we have not studied this issue in detail. In this example, we compute F_0 using 10000 iterations.

We then perform the main method with F_0 as our starting guess, using $= 10^{-4}$ and 1000 iterations. It took about a half an hour to run on a laptop with a 1.7 GHz processor and 640 MB of memory, using double precision with machine roundoff about 10^{-16}

Separation rank, achieved approximation, and eigenvalue estimates for the separable ($\mathbf{F}_0$*) and main (*$\mathbf{F}$



. , *The computed wavefunction, represented by* $\square_{l=1}^{2} s_l$ i

[illegible]

The amount of the ground state orbitals present in the two terms of the wavefunction in Figure , and their net excitations.

[illegible]	[illegible]

Finally, we are working to resolve the critical question as to what extent separated representations can represent functions and operators in general.

Acknowledgments. We would like to thank Dr. Robert Harrison (ORNL and University of Tennessee) and Dr. Lucas Monzón (University of Colorado) for many useful conversations about this project.

REFERENCES

[illegible] *Adaptive Control Processes: A Guided Tour*, [illegible] 6 [illegible]

[illegible] *Fast spectral projection algorithms for density-matrix computations*, [illegible] ( ) [illegible]

[illegible] *Numerical operator calculus in higher dimensions*, [illegible] ( ) [illegible] 6 [illegible]

[illegible] *On generalized Gaussian quadratures for exponentials and their applications*, [illegible] ( ) [illegible]

[illegible] *On approximation of functions by exponential sums*, [illegible]

6 [illegible]

6 [illegible] *Monte Carlo calculations of the ground state of three- and four-body nuclei* [illegible] ( ) (6)

[illegible] *Orthogonal tensor decompositions* [illegible] ( )

[illegible] *Principal component analysis of three-mode data by means of alternating least squares algorithms* [illegible] ( ) 6

[illegible] *Canonical correlation analysis when the data are curves* [illegible] ( )

[illegible] *On the systematic improvement of fixed-node diffusion quantum Monte Carlo energies using natural orbital CI guide functions* [illegible] ( ) 6

[illegible] *A q-analog of the Gauss summation theorem for hypergeometric series in* (n) [illegible] ( )

[illegible] *Trigonometric identities and sums of separable functions* [illegible]

[illegible] *Canonical purification of the density matrix in electronic-structure theory* [illegible] ( )

[illegible] *The Symmetric Group in Quantum Chemistry* [illegible]

[illegible] *Efficient computer manipulation of tensor products with applications to multidimensional approximation* [illegible] ( ) 6

6 [illegible] *Finite Sums Decompositions in Mathematical Analysis* [illegible] ( ) &

[illegible] *Configuration Interaction with Non-orthogonal Slater Determinants Applied to the Hubbard Model, Atoms, and Small Molecules* [illegible]

[illegible] *Iterative Berechnung der reziproken Matrix* [illegible] ( )

[illegible] *The configuration interaction method: Advances in highly correlated approaches* [illegible] ( ) 6

[illegible] *Multi-way Analysis: Applications in the Chemical Sciences* [illegible] &

[illegible] *Estimates of best bilinear approximations of periodic functions* [illegible] ( )

[illegible] *Kronecker-product approximations for some function-related matrices* [illegible] ( )

[illegible] *Tensor approximations of matrices generated by asymptotically smooth functions* [illegible] ( ) 6

[illegible] *Approximation with Kronecker products* [illegible] ( ) [illegible]

[illegible] *A generalized one-dimensional fast multipole method with application to filtering of spherical harmonics* [illegible] ( ) 6

6 [illegible]