LOCALIZED SPECTRUM SLICING
LIN LIN *

Abstract. Given a sparse Hermitian matrix $A$ and a real number $\mu$, we construct a set of sparse vectors, each approximately spanned only by eigenvectors of $A$ corresponding to eigenvalues near $\mu$. This set of vectors spans the column space of a localized spectrum slicing (LSS) operator, and is called an LSS basis set. The sparsity of the LSS basis set is related to the decay properties of matrix Gaussian functions. We present a divide-and-conquer strategy with controllable error to construct the LSS basis set. This is a purely algebraic process using only submatrices of $A$, and is therefore applicable to general sparse Hermitian matrices. We demonstrate that the computational complexity of our method is $O(n)$ under the assumption that the spectral radius of $A$ does not increase with $n$. The LSS basis set leads to sparse projected matrices with reduced sizes, which allows the projected problems to be solved efficiently with techniques using sparse linear algebra. As an example, we demonstrate that the LSS basis set can be used to solve interior eigenvalue problems for a discretized second order partial differential operator in one-dimensional and two-dimensional domains.

Key words. Spectrum slicing; Localization; Decay properties; Basis set; Interior eigenvalue problem

AMS subject classifications. 65F60, 65F50, 65F15, 65N22

1. Introduction. Let $A$ be an $n \times n$ large, sparse, Hermitian matrix. In many applications in science and engineering, one would like to find eigenvalues and eigenfunctions of $A$ near a given real number $\mu$. As a motivating problem, we consider $A$ to be obtained from a certain discretization (e.g. finite difference or finite element discretization) of a second order partial differential operator of the form $-\Delta + V(x)$, where $\Delta$ is the Laplacian operator, and $V(x)$ is a potential function. Depending on the context and the choice of $V$, this type of problems can arise from quantum mechanics, wave propagation, electromagnetism etc.

When $\mu$ locates inside the spectrum of $A$, the eigenvalues to be computed are called interior eigenvalues. These interior eigenvalues and corresponding eigenfunctions are in general difficult to compute. Since $n$ is large and $A$ is sparse, iterative methods such as inverse power method [10], preconditioned conjugate gradient type of methods [6, 13], and shift-inverse Lanczos type of methods [15, 24] are desirable. The effectiveness of such methods often depends on the availability of a good preconditioner that can approximately apply $(A - \mu I)^{-1}$ to vectors, and such preconditioner can be difficult to construct.

Another type of methods that recently receives increasing amount of attention is based on the construction of a matrix function $f_\mu(A)$, where the corresponding scalar function $f_\mu(z)$ only takes significant values on a small interval near $\mu$ on the real line. Such a matrix function $f_\mu(A)$ can be called a spectrum slicing operator, since for any vector $v \in \mathbb{C}^n$, $f_\mu(A)v$ is approximately only spanned by eigenvectors of $A$ corresponding to eigenvalues near $\mu$, and the vector $f_\mu(A)v$ is said to be spectrally localized. The spectrum slicing operator can be simultaneously applied to a set of random vectors $V = [v_1, \ldots, v_p]$. When $p$ is large enough but is still small compared to $n$, the subspace spanned by

$$W = f_\mu(A)V$$

* Department of Mathematics, University of California Berkeley and Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720. Email: lin-lin@math.berkeley.edu
will approximately contain the subspace of all eigenvectors corresponding to eigenvalues near \( \mu \). Let
\[
A_W = W^* AW, \quad B_W = W^* W,
\]
then the desired eigenvalues and eigenvectors can be computed via the solution of a generalized eigenvalue problem
\[
A_W C = B_W C \Theta. \tag{1.1}
\]
In practice \( f_{\mu}(A) \) can be constructed through a relatively high order Chebyshev polynomials \[23\], or contour integral based methods \[20, 22\]. It should be noted that contour integral based methods still require solving equations of the form \((A - zI)^{-1} v\) where \( z \) is close to \( \mu \) in the complex plane, either through direct methods or iterative methods.

In general the spectrum slicing operator \( f_{\mu}(A) \) is a dense matrix. Therefore the matrix \( W = f_{\mu}(A)V \) is in general a dense matrix, regardless of how the initial matrix \( V \in \mathbb{C}^{n \times p} \) is chosen. Furthermore, the matrices \( A_W, B_W \) are in general dense matrices that do not reveal much structure to be further exploited, and the solution of the projected problem (1.1) may still be expensive when \( p \) is large.

1.1. Contribution. In this paper, we consider the use of a simple choice of Gaussian function with a positive number \( \sigma \)
\[
f_{\sigma, \mu}(z) = e^{-\frac{(z-\mu)^2}{\sigma^2}}, \tag{1.2}
\]
and the corresponding matrix Gaussian function \( f_{\sigma, \mu}(A) \) is spectrally localized near \( \mu \) with width proportional to \( \sigma \). We demonstrate that under a proper choice of \( \sigma \), \( f_{\sigma, \mu}(A) \) can have many entries that are small in magnitude, so that after truncating these small entries the resulting matrix is close to be a spectrum slicing operator but is also sparse. In this sense, \( f_{\sigma, \mu}(A) \) is called a localized spectrum slicing (LSS) operator.

We demonstrate that the LSS operator \( f_{\sigma, \mu}(A) \) can be constructed in a divide-and-conquer method with controllable error using only a sequence of submatrices of \( A \) with \( \mathcal{O}(n) \) cost, provided that the spectral radius of \( A \) does not increase as \( n \) increases. The column space of the LSS operator is spanned by a sparse matrix \( U \in \mathbb{C}^{n \times p} \), and the subspace spanned by \( U \) will approximately contain the subspace of eigenvectors to be computed. As a result, the projected matrices
\[
A_U = U^* AU, \quad B_U = U^* U \tag{1.3}
\]
are sparse matrices. In this aspect, the matrix \( U \) can be regarded as a specially tailored basis set for representing the subspace approximately spanned by eigenvectors of \( A \) near \( \mu \), and each column of \( U \) is localized both spectrally and spatially. In the following text \( U \) is called a localized spectrum slicing (LSS) basis set. The LSS basis set can be constructed without explicitly constructing the LSS operator. The generalized eigenvalue problem for the sparse projected matrices \( A_U, B_U \) may be solved both by direct methods, but also by methods using sparse linear algebra techniques. During the construction of the LSS operator and/or the LSS basis set, a good global preconditioner for \((A - \mu I)^{-1}\) is not needed. We demonstrate the construction of the LSS basis set and its use for solving interior eigenvalues problems for matrices obtained from discretizing second order partial differential operators, and find that the use of the LSS basis set can be more efficient than solving the global problem directly for matrices of large sizes.
1.2. Related work. The spectral locality of the LSS operator is valid by construction. Comparatively the spatial locality of the LSS operator is less obvious, and is given more precisely by the decay properties of matrix functions that are analytic in a certain region in the complex plane (see e.g. [1, 2, 3]). The decay properties of matrix functions were first realized for matrix inverse $A^{-1}$ (i.e. $f(z) = z^{-1}$), where $A$ is a banded, positive definite matrix [4, 8]. The method for showing decay properties relies on whether $f(z)$ can be well approximated by a low order Chebyshev polynomial evaluated at the eigenvalues of $A$, and this method is therefore generalizable to any analytic function $f(z)$ for banded matrices $A$. In order to generalize from banded matrices to general sparse matrices, decay properties should be defined using geodesic distances of the graph induced by $A$. These techniques have been shown in [1, 3] and references therein, for demonstrating the decay properties of e.g. Fermi-Dirac operators in electronic structure theory. These techniques are directly used for showing the decay properties of the LSS operator in this work, which then allows the construction of the divide-and-conquer method. In physics literature, such decay property is dubbed “near-sightedness property” and is vastly studied using various models (see e.g. [14, 19, 21]). The decay property is also used for constructing linear scaling algorithms [4, 9] for density functional theory calculations.

1.3. Contents. The rest of this paper is organized as follows. We introduce the decay properties of matrix functions and in particular the localized spectrum slicing operator in section 2. Based on the decay properties, section 3 describes a divide-and-conquer algorithm for constructing the LSS operator and the LSS basis set, and provides the error bound and computational complexity. Section 4 demonstrates the use of the LSS basis set for solving interior eigenvalue problems. We demonstrate numerical results using the LSS basis set for solving interior eigenvalue problems for discretized second order partial differential operator in section 5, and discuss the conclusion and future work in section 6.

2. Preliminaries.

2.1. Notation. The $(i, j)$-th element of a matrix $A \in \mathbb{C}^{n \times n}$ is denoted by $A_{ij}$. The submatrix of $A$ corresponding to a set of row indices $\mathcal{I}$ and a set of column indices $\mathcal{J}$ is denoted by $A_{\mathcal{I}, \mathcal{J}}$. Using MATLAB notation, all elements in the $i$-th row of $A$ are denoted by $A_{i, \cdot}$, and all elements in a set of rows $\mathcal{I}$ are denoted by $A_{\mathcal{I}, \cdot}$. Similarly, all elements in the $j$-th column of $A$ are denoted by $A_{\cdot, j}$, and all elements for a set of columns $\mathcal{J}$ are denoted by $A_{\cdot, \mathcal{J}}$. The $k$-th power of $A$ is denoted by $A^k$. The matrix $p$-norm of $A$ is denoted by $\|A\|_p$, and the vector $p$-norm of a vector $u$ is denoted by $\|u\|_p$ ($p \geq 1$). The max norm of a matrix is denoted by $\|A\|_{\max} \equiv \max_{i,j} \{|A_{ij}|\}$, which is the same as the $\infty$-norm of a vector of length $n^2$, formed by all the elements of $A$. The Hermitian conjugate of $A$ is denoted by $A^*$. Depending on the context, we may also refer to a matrix as an operator.

A Hermitian matrix $A$ induces an undirected graph $G = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{i | i = 1, \ldots, n\}$, and $\mathcal{E} = \{(i, j) | A_{ij} \neq 0, \quad 1 \leq i, j \leq n\}$. Each element in $\mathcal{V}$ is called a vertex, and each element in $\mathcal{E}$ is called an edge. The cardinality of a set of indices $\mathcal{I}$ is denoted by $|\mathcal{I}|$.

A Hermitian matrix $A \in \mathbb{C}^{n \times n}$ has the eigen-decomposition

$$AX = X\Lambda.$$  \hfill (2.1)

Here $\Lambda = \text{diag}(|\lambda_1|, \ldots, |\lambda_n|)$ is a diagonal matrix containing the (real) eigenvalues of $A$ and we assume $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ are ordered non-decreasingly. $X = [x_1, \ldots, x_n]$
and $x_i$ is the eigenvector corresponding to the eigenvalue $\lambda_i$. If all eigenvalues (and corresponding eigenvectors) to be computed are within a small interval $(\mu - c, \mu + c)$ on the real line with $\lambda_1 < \mu - c < \mu + c < \lambda_n$, then this problem is called an interior eigenvalue problem.

2.2. Decay property of matrix functions. In this section, we provide a short but self-contained description of the decay properties of $f_{\sigma,\mu}(A)$. More details on the description of the decay properties of general matrix functions can be found in [1] and references therein.

Let $k$ be a non-negative integer, and $P_k$ be the set of all polynomials of degrees less than or equal to $k$ with real coefficients. Without loss of generality we assume the eigenvalues of $A$ are within the interval $(-1, 1)$. For a real continuous function $f$ on $[-1, 1]$, the best approximation error is defined as

$$E_k(f) = \inf_{p \in P_k} \left\{ \|f - p\|_{\infty} \equiv \max_{-1 \leq x \leq 1} |f(x) - p(x)| \right\}. \quad (2.2)$$

Consider an ellipse in the complex plane $C$ with foci in $-1$ and $1$, and $a, b > 0$ be the half axes so that the vertices of the ellipse are $a, -a, ib, -ib$, respectively. Let the sum of the half axes be $\chi = a + b$, then using the identity $a^2 - b^2 = 1$ we have

$$a = \frac{\chi^2 + 1}{2\chi}, \quad b = \frac{\chi^2 - 1}{2\chi}. \quad (2.6)$$

Thus the ellipse is determined only by $\chi$, and such ellipse is denoted by $E_{\chi}$. Then Bernstein’s theorem [18] is stated as follows.

**Theorem 2.1 (Bernstein).** Let $f(z)$ be analytic in $E_{\chi}$ with $\chi > 1$, and $f(z)$ is a real valued function for real $z$. Then

$$E_k(f) \leq \frac{2M(\chi)}{\chi^k(\chi - 1)}, \quad (2.3)$$

where

$$M(\chi) = \sup_{z \in E_{\chi}} |f(z)|. \quad (2.4)$$

Using Theorem 2.1, a more quantitative description of the approximation properties for $f_{\sigma,\mu}(z)$ in Eq. (1.2) is given in Theorem 2.2.

**Theorem 2.2.** Let $f_{\sigma,\mu}(z)$ be a Gaussian function defined in Eq. (1.2), then for any $\alpha > 0$,

$$E_k(f_{\sigma,\mu}) \leq \frac{2}{\alpha \sigma} e^{\alpha^2} (1 + \alpha \sigma)^{-k}. \quad (2.5)$$

**Proof.** For any $\mu \in (-1, 1)$, $\sigma > 0$, the Gaussian function $f_{\sigma,\mu}$ is analytic in any ellipse $E_{\chi}$ with $\chi > 1$, then

$$M(\chi) = \sup_{z \equiv x + iy \in E_{\chi}} |f_{\sigma,\mu}(x + iy)| \leq \sup_{z \equiv x + iy \in E_{\chi}} e^{\frac{2}{\alpha \sigma}} \leq e^{(\chi - 1)^2/(4\alpha^2)}. \quad (2.6)$$

For any $\alpha > 0$, let

$$\chi = 1 + \alpha \sigma. \quad (2.6)$$
then $\chi - \frac{1}{\chi} \leq 2\alpha\sigma$, and

$$M(1 + \alpha\sigma) \leq e^{\alpha^2}. \quad (2.7)$$

Using Theorem 2.1, Eq. (2.5) is the direct consequence of Eq. (2.7) and the choice of $\chi$ in Eq. (2.6). \hfill \Box

For the graph $G = (V, E)$ associated with the matrix $A$ and vertices $i, j \in V$, a path linking $i, j$ is given by a sequence of edges $p = \{(i_0 \equiv i, i_1), (i_1, i_2), \ldots, (i_l, i_{l+1} \equiv j)\}$ where $i_1, \ldots, i_l \in V$, and each element in $p$ is an edge in $E$. The length of the path $p$ is defined to be $l + 1$. If $p = \{(i, j)\}$ then the length of $p$ is 1. The geodesic distance $d(i, j)$ between vertices $i$ and $j$ is defined as the length of the shortest path between $i$ and $j$. It should be noted that for structurally symmetric matrices, i.e. $A_{ij} \neq 0$ implies $A_{ji} \neq 0$ for all indices $i, j$, the geodesic distance is symmetric, i.e. $d(i, j) = d(j, i)$. In particular, Hermitian matrices are structurally symmetric. If $d(i, j) > 1$, then $A_{ij} = 0$. If $d(i, j) = \infty$ then there is no path connecting $i$ and $j$. This statement can be straightforwardly generalized as in Proposition 2.3.

PROPOSITION 2.3. Let $A$ be a sparse and Hermitian matrix. For any positive integer $k$, if $d(i, j) > k$ then $(A^k)_{ij} = 0$, where $A^k$ is the $k$-th power of the matrix $A$.

Proof. The statement is obviously correct for $k = 1$. Assume the statement holds for $k - 1$, then if $d(i, j) > k$, $(A^k)_{ij} = \sum_p A_{ip}(A^{k-1})_{pj}.$

Each term in the summation is nonzero only if $A_{ip} \neq 0$, i.e. $d(i, p) = 1$. Since $d(i, j) > k$, for such $p$ we have $d(p, j) > k - 1$, and we have $(A^{k-1})_{pj} = 0$. Therefore $(A^k)_{ij} = 0$ and the statement holds for $k$. \hfill \Box

The precise statement of the spatial locality of the matrix function $f_{\sigma, \mu}(A)$ is given by the decay properties along the off-diagonal direction in Theorem 2.4. For a given column $j$, the magnitude of each element $f_{\sigma, \mu}(A)_{i, j}$ decays exponentially with respect to the geodesic distance $d(i, j)$.

THEOREM 2.4. Let $A$ be a sparse and Hermitian matrix with all eigenvalues contained in the interval $(-1, 1)$. For any $\alpha > 0, \sigma > 0$, let

$$\rho = (1 + \alpha\sigma)^{-1}, \quad K = \frac{2}{\rho\alpha\sigma} e^{\alpha^2}, \quad (2.8)$$

then for all $d(i, j) \geq 1, i, j = 1, \ldots, n$,

$$|f_{\sigma, \mu}(A)_{ij}| \leq K\rho^{d(i, j)}, \quad (2.9)$$

where $d(i, j)$ is the geodesic distance between vertices $i$ and $j$.

Proof. From Theorem 2.2 for any integer $k \geq 0$, there exists a polynomial $p_k \in \mathbb{P}_k$ such that

$$\|f_{\sigma, \mu}(A) - p_k(A)\|_2 = \|f_{\sigma, \mu} - p_k\|_\infty = E_k(f_{\sigma, \mu}) \leq K\rho^{k+1}.$$

Now consider all edges $(i, j)$ such that the geodesic distance $d(i, j) = k + 1$, and then $p_k(A)_{ij} = 0$. Therefore

$$|f_{\sigma, \mu}(A)_{ij}| = |f_{\sigma, \mu}(A)_{ij} - p_k(A)_{ij}| \leq \|f_{\sigma, \mu}(A) - p_k(A)\|_2 \leq K\rho^{k+1} = K\rho^{d(i, j)}.$$
Remark 2.5. As suggested in Eq. (2.6), \( \rho, K \) only depend on \( \sigma \) but not on \( \mu \). Therefore the decay properties of the matrix function \( |f_{\sigma,\mu}(A)| \) seem to be independent of the shift \( \mu \). This is because an upper bound for \( M(\chi) \) is given in Theorem (2.2) that is valid for all \( \mu \). Numerical results in section 7 indicate that the preconstant of the exponential decay may have a strong dependency on \( \mu \).

Remark 2.6. In Theorem 2.4 there is an arbitrary positive constant \( \alpha \). For any given \( \alpha > 0 \), the off-diagonal entries of \( |f_{\sigma,\mu}(A)_{ij}| \) should decay exponentially with respect to the geodesic distance. By optimizing \( \alpha \) together with the degree of the Chebyshev polynomial \( k \), the actual decay rate can be slightly faster than exponential. Fig. 2.1 gives an example of the magnitude of the first column \( |f_{\sigma,\mu}(A)_{1,1}| \) where \( A \) is a discretized Laplacian operator in 1D with periodic boundary conditions, with \( \sigma = 1.0, \mu = 2.0 \) and \( \sigma = 1.0, \mu = 10.0 \) respectively.

![Fig. 2.1: Log-scale plot of the magnitude of the first column \( |f_{\sigma,\mu}(A)_{1,1}| \). A is a discretized Laplacian operator in 1D with periodic boundary conditions, with \( \sigma = 1.0 \) and two choices of \( \mu \).](image)

Remark 2.7. When \( \alpha \sigma \ll 1 \),

\[
\rho^{d(i,j)} = (1 + \alpha \sigma)^{-d(i,j)} \approx e^{-\alpha \sigma d(i,j)}.\]

Here \( \sigma \) reflects the spectral locality, and \( d(i,j) \) reflects the spatial locality. Then Theorem 2.4 reveals the balance between the spectral and spatial locality, tuned by one parameter \( \sigma \).

3. Localized spectrum slicing.

3.1. Algorithm. Using the decay properties of the LSS operator \( f_{\sigma,\mu}(A) \) in Theorem 2.4 a set of basis functions called the LSS basis set can be constructed in a divide-and-conquer fashion. Below we demonstrate that if the smearing parameter \( \sigma \) is large enough, then the localized spectrum slicing operator \( f_{\sigma,\mu}(A) \) can be approximately computed using submatrices of \( A \). The size of each submatrix is independent of the size of \( A \). This is important for reducing the computational complexity and for parallel computation.

Proposition 3.1. Let \( A, B \) be \( n \times n \) Hermitian matrices. The graph induced by \( B \) is a spanning subgraph of the graph \( G \) induced by \( A \), and the geodesic distance \( d(i,j) \)
is defined using the graph $G$. We assume for a given integer $j, m$ ($1 \leq j, m \leq n$),

$$A_{il} = B_{il}, \quad \forall i, l \quad \text{s.t.} \quad d(j, i) \leq m, \quad d(j, l) \leq m.$$  

Then for any integer $k$ ($1 \leq k \leq m$),

$$(A^k)_{il} = (B^k)_{il}, \quad \forall i, l \quad \text{s.t.} \quad d(j, i) \leq m - k + 1, \quad d(j, l) \leq m - k + 1.$$  

Proof. The statement is apparently correct for $k = 1$. Assume the statement for $k - 1$ is proved, then

$$(A^k)_{il} = \sum_p A_{ip}(A^{k-1})_{pl}, \quad (B^k)_{il} = \sum_p B_{ip}(B^{k-1})_{pl}.$$  

In the summation above, $A_{ip}(A^{k-1})_{pl}$ is nonzero only if $A_{ip} \neq 0$. Similarly $B_{ip}(B^{k-1})_{pl}$ is nonzero only if $B_{ip} \neq 0$. Since the graph induced by $B$ is a subgraph of the graph induced by $A$, $B_{ip} \neq 0$ implies $A_{ip} \neq 0$, and therefore we only need to consider $p$ such that $A_{ip} \neq 0$, i.e. $d(i, p) = 1$. Consider $i$ such that $d(j, i) \leq m - k + 1$, then $p$ satisfies

$$d(j, p) \leq d(j, i) + d(i, p) = m - k + 2 = m - (k - 1) + 1.$$  

Since $d(j, i) \leq m, d(j, p) \leq m$, we have $A_{ip} = B_{ip}$. Also $d(j, l) \leq m - (k - 1) + 1$, by assumption we have $(A^{k-1})_{pl} = (B^{k-1})_{pl}$. Therefore

$$(A^k)_{il} = \sum_p A_{ip}(A^{k-1})_{pl} = \sum_p B_{ip}(B^{k-1})_{pl} = (B^k)_{il}$$  

is valid for all $i, l$ such that $d(j, i) \leq m - k + 1, d(j, l) \leq m - k + 1$. \[\square\]

Using Proposition 3.1 Theorem 3.2 shows that the $j$-th column of $f_{\sigma, \mu}(A)$ can be accurately computed from $f_{\sigma, \mu}(B)$, as long as $A$ and $B$ are sufficiently close in the vicinity of $j$ in the sense of geodesic distance.

**Theorem 3.2.** Let $A, B$ be $n \times n$ Hermitian matrices with eigenvalues in $(-1, 1)$. For a given $j$ and an even integer $m$ ($1 \leq j, m \leq n$),

$$A_{il} = B_{il}, \quad \forall i, l \quad \text{s.t.} \quad d(j, i) \leq m, \quad d(j, l) \leq m.$$  

Then

$$|f_{\sigma, \mu}(A)_{ij} - f_{\sigma, \mu}(B)_{ij}| \leq 2K\rho^{\frac{m}{2}+1},$$  

for all $i$ such that $d(j, i) \leq m/2 + 1$, where the constants $K, \rho$ are given in Eq. (2.8).

Proof. From Theorem 2.2 for any integer $k \geq 0$, there exists a polynomial $p_k \in \mathbb{P}_k$ such that

$$\|f_{\sigma, \mu}(A) - p_k(A)\|_2 = \|f_{\sigma, \mu} - p_k\|_{\infty} \leq K\rho^{k+1}.$$  

Then for any integer $k \geq 0$, we have

$$|f_{\sigma, \mu}(A)_{ij} - f_{\sigma, \mu}(B)_{ij}| \leq |f_{\sigma, \mu}(A)_{ij} - p_k(A)_{ij}| + |p_k(A)_{ij} - p_k(B)_{ij}| + |f_{\sigma, \mu}(B)_{ij} - p_k(B)_{ij}|.$$  

Take $k = \frac{m}{2}$, then by Proposition 3.1 $p_k(A)_{ij} = p_k(B)_{ij}$ for all $i$ such that $d(j, i) \leq m - k + 1 = \frac{m}{2} + 1$. Also

$$|f_{\sigma, \mu}(A)_{ij} - p_k(A)_{ij}| \leq \|f_{\sigma, \mu}(A) - p_k(A)\|_2 \leq K\rho^{\frac{m}{2}+1},$$  

then

$$|f_{\sigma, \mu}(A)_{ij} - f_{\sigma, \mu}(B)_{ij}| \leq 2K\rho^{\frac{m}{2}+1}.$$  

\[\square\]
\[ |f_{\sigma,\mu}(B)_{ij} - p_k(B)_{ij}| \leq \|f_{\sigma,\mu}(B) - p_k(B)\|_2 \leq K\rho^{2+1}, \]

then
\[ |f_{\sigma,\mu}(A)_{ij} - f_{\sigma,\mu}(B)_{ij}| \leq 2K\rho^{2+1}, \]

for all \( i \) such that \( d(j, i) \leq m/2 + 1 \).

Theorem 3.2 shows that in order to compute any column \( j \) of the matrix \( f_{\sigma,\mu}(A) \) up to a certain accuracy, it is only necessary to have a matrix that is the same as \( A \) up to a certain distance away from \( j \). Together with the decay property of each column of \( f_{\sigma,\mu}(A) \), this allows the \( j \)-th column of \( f_{\sigma,\mu}(A) \) to be constructed in a divide and conquer manner. For instance, for a given integer \( m \) we can define
\[ B_{il} = \begin{cases} A_{il}, & \forall i, l \text{ s.t. } d(j, i) \leq m, d(j, l) \leq m, \\ 0, & \text{otherwise} \end{cases}. \tag{3.1} \]

which is simply a submatrix of \( A \). As a submatrix, \( \|B\|_2 \leq \|A\|_2 \) and the assumption of the spectral radius in Theorem 3.2 is satisfied.

In practice it would be very time consuming to construct an approximate matrix for each column of \( j \). Instead one can partition the domain into disjoint columns sets, and apply the truncated matrix to each column set. We partition \( \mathcal{V} = \{1, \ldots, n\} \) into \( M \) simply connected disjoint sets \( \{E_\kappa\}_{\kappa=1}^M \), i.e.
\[ \mathcal{V} = \bigcup_{\kappa=1}^M E_\kappa, \text{ and } E_\kappa \cap E_{\kappa'} = \emptyset, \ \kappa \neq \kappa'. \tag{3.2} \]

For each \( E_\kappa \) and an integer \( m \), we define an associated set
\[ Q_\kappa = \{i|d(i, j) \leq m, \forall j \in E_\kappa\}. \tag{3.2} \]

Then Theorem 3.2 implies that the submatrix \( (f_{\sigma,\mu}(A))_{E_\kappa} \) can be constructed by using a submatrix of \( A \) defined as
\[ (A_\kappa)_{ij} = \begin{cases} A_{ij}, & i, j \in Q_\kappa, \\ 0, & \text{otherwise}. \end{cases} \tag{3.3} \]

In the following discussion, we refer to \( E_\kappa \) as an element, and to \( Q_\kappa \) as an extended element associated with \( E_\kappa \).

**Remark 3.3.** The choice in Eq. (3.1) takes a submatrix of \( A \) to compute the localized spectrum slicing operator. From the point of view of partial differential operators, this is similar to imposing zero Dirichlet boundary condition on some local domains.

Since \( A \) is Hermitian and sparse, and so is \( A_\kappa \), and the latter has the eigen-decomposition
\[ A_\kappa X_\kappa = X_\kappa D_\kappa. \tag{3.4} \]

Here \( D_\kappa \) is a diagonal matrix. Note that \( A_\kappa \) only takes nonzero values on the extended element \( Q_\kappa \), and therefore the eigen-decomposition \( (3.4) \) only considers \( X_\kappa \) with nonzero values on \( Q_\kappa \). Define
\[ f_{\sigma,\mu}(A_\kappa) \equiv X_\kappa f_{\sigma,\mu}(D_\kappa) X_\kappa^*. \tag{3.5} \]
Using Theorem 3.2, \( f_{\sigma,\mu}(A)_{Q_\kappa,E_\kappa} \) can be approximated by \( f_{\sigma,\mu}(A_\kappa) \), in the sense that

\[
| f_{\sigma,\mu}(A)_{ij} - f_{\sigma,\mu}(A_\kappa)_{ij} | \leq 2K\rho^{\frac{\tau}{2}+1}, \quad \forall i \in Q_\kappa, \ j \in E_\kappa.
\]

Using the spectral locality of \( f_{\sigma,\mu} \), in practice not all eigenvalues and eigenvectors of \( A_\kappa \) as in (3.4) are needed. Instead only a partial eigen-decomposition is needed to compute all eigenvalues of \( A_\kappa \) in the interval \( (\mu - \sigma, \mu + \sigma) \). Due to the fast decay properties of Gaussian functions, in practice \( c \) can be chosen to be \( 2 \sim 4 \) to be sufficiently accurate.

The factorized representation in Eq. (3.5) also allows the computation of a set of vectors approximately spanning the column space of \( f_{\sigma,\mu}(A_\kappa) \), through a local singular value decomposition (SVD) procedure, i.e.

\[
\| f_{\sigma,\mu}(D_\kappa)(X_\kappa)_{Q_\kappa,E_\kappa} - \bar{U}_\kappa\tilde{S}_\kappa\bar{V}_\kappa^* \|_2 \leq \bar{\tau}.
\]

Here \( \bar{\tau} \) is SVD truncation criterion. In practice \( \bar{\tau} \) may also be chosen using a relative criterion as \( \bar{\tau} = \tau(S_\kappa)_{1,1} \) in used in our numerical experiment, where we assume \((\tilde{S}_\kappa)_{1,1}\) is the largest singular value in Eq. (3.6). In practice this can be performed by only keeping the singular values in the diagonal matrix \( \tilde{S}_\kappa \) that are larger than \( \bar{\tau} \).

Then we can define

\[
U_\kappa = X_\kappa\bar{U}_\kappa, \quad V_\kappa = \tilde{S}_\kappa\bar{V}_\kappa^*.
\]

We combine all \( U_\kappa \) together

\[
U \equiv [U_1, \ldots, U_M],
\]

and \( U \) is the LSS basis set that is both spectrally localized and spatially localized. We denote by \( n_b \) the total number of columns of \( U \), which is also referred to as the size of the LSS basis set. Using the LSS basis set, an approximation to the LSS operator is defined as

\[
\bar{f}_{ij} = \begin{cases} (U_\kappa)_{i,:}(V_\kappa)_{:,j}, & i \in Q_\kappa, j \in E_\kappa, \text{ for some } \kappa, \\ 0, & i \notin Q_\kappa, j \in E_\kappa, \text{ for some } \kappa. \end{cases}
\]

\( \bar{f} \) is an \( n \times n \) sparse matrix, and the error in the max norm for approximating the LSS operator \( f_{\sigma,\mu}(A) \) is given in Theorem 3.4

**Theorem 3.4.** Let \( A \) be an \( n \times n \) Hermitian matrix with eigenvalues in \((-1,1)\), and the induced graph is partitioned into \( M \) elements \( \{E^\kappa\} \). For each element \( E_\kappa \), there is an extended element \( Q_\kappa \) given in (3.2), a submatrix \( A_\kappa \) given in (3.3), and matrices \( U_\kappa, V_\kappa \) satisfying (3.6) and (3.7). Let \( \bar{f} \) be an \( n \times n \) matrix defined in Eq. (3.9), then

\[
\| f_{\sigma,\mu}(A) - \bar{f} \|_{\text{max}} \leq 2K\rho^{\frac{\tau}{2}+1} + \bar{\tau}.
\]

**Proof.** For each element \( \kappa \), from Eq. (3.6) we have

\[
\max_{i \in Q_\kappa,j \in E_\kappa} | f_{\sigma,\mu}(A_\kappa)_{ij} - \bar{f}_{ij} | = \max_{i \in Q_\kappa,j \in E_\kappa} | f_{\sigma,\mu}(A_\kappa)_{ij} - (U_\kappa)_{i,:}(V_\kappa)_{:,j} | \leq \| f_{\sigma,\mu}(A_\kappa) - U_\kappa V_\kappa \|_2 \leq \| X_\kappa \|_2 \bar{\tau} = \tilde{\tau}.
\]
Using Theorem 3.2 and the definition of the extended element (3.2)

$$\max_{i \in Q_\kappa, j \in E_\kappa} |f_{\sigma, \mu}(A)_{ij} - f_{\sigma, \mu}(A_\kappa)_{ij}| \leq 2K\rho^{2\kappa+1}. \quad (3.12)$$

For vertices $i \notin Q_\kappa, j \in E_\kappa$, $\tilde{f}_{ij} = 0$. Then from Theorem 2.4 and use $\rho < 1$

$$\max_{i \notin Q_\kappa, j \in E_\kappa} |f_{\sigma, \mu}(A)_{ij} - \tilde{f}_{ij}| = |f_{\sigma, \mu}(A)_{ij}| \leq K\rho^{m+1} \leq K\rho^{m/2+1}. \quad (3.13)$$

Combining Eqs. (3.11), (3.12), (3.13), we have

$$\|f_{\sigma, \mu}(A) - \tilde{f}\|_{\max} = \max_{1 \leq i, j \leq n} |f_{\sigma, \mu}(A)_{ij} - \tilde{f}_{ij}|$$

$$= \max_\kappa \left\{ \max_{i \in Q_\kappa, j \in E_\kappa} |f_{\sigma, \mu}(A)_{ij} - \tilde{f}_{ij}|, \max_{i \notin Q_\kappa, j \in E_\kappa} |f_{\sigma, \mu}(A)_{ij} - \tilde{f}_{ij}| \right\}$$

$$= \max_\kappa \left\{ \max \{2K\rho^{2\kappa+1} + \tilde{\tau}, K\rho^{m/2+1}\} \right\} = 2K\rho^{2\kappa+1} + \tilde{\tau}. \quad \square$$

Remark 3.5. Theorem 3.4 indicates that in order to accurately approximate the LSS operator, the SVD truncation criterion $\tilde{\tau}$ must be small enough. However, this may not necessarily be the case for approximating interior eigenvalues. This will be discussed in section 5.

Finally, we summarize the algorithm for finding the divide-and-conquer method for constructing the LSS basis set in Algorithm 1.

**Algorithm 1:** Localized spectrum slicing basis set.

**Input:**
1. Sparse Hermitian matrix $A$, center $\mu$, width $\sigma$, SVD truncation tolerance $\tilde{\tau}$.
2. Number of elements $M$, partition of elements $\{E_\kappa\}_{\kappa=1}^M$ and extended elements $\{Q_\kappa\}_{\kappa=1}^M$.

**Output:** LSS basis set $\{U_\kappa\}_{\kappa=1}^M$.

For $\kappa = 1, \ldots, M$ do

- Compute the (partial) eigen-decomposition according to (3.4);
- Compute the local SVD decomposition according to (3.6) and only keep singular vectors with singular values larger than $\tilde{\tau}$;
- Compute $U_\kappa$ with matrix multiplication according to (3.7);

end

3.2. Complexity. In order to simplify the analysis of the complexity of the Algorithm 1 for finding the LSS basis set, we make the assumption that the set of $n$ vertices is equally divided into $M$ elements, so that $|E_\kappa| = \frac{n}{M} \equiv |E|$, $|Q_\kappa| = \frac{cQ|E|}{M} \equiv cQ|E|$, where $cQ$ is a small number denoting the ratio between the size of the extended element and the size of the element. Denote by $s_\kappa$ the column dimension of $X_\kappa$ in the partial eigen-decomposition of $A_\kappa$. Denote by $t_\kappa$ the column dimension of $U_\kappa$, and $t_\kappa \leq s_\kappa$. For simplicity we assume $\{s_\kappa\}, \{t_\kappa\}$ are uniform so that $s_\kappa = s, t_\kappa = t, \kappa = 1, \ldots, M$. If $A_\kappa$ is treated as a dense matrix for the computation of the local eigen-decomposition of $A_\kappa$, then the cost is $c_{\text{Eig}}|Q_\kappa|^3$. The cost of the SVD decomposition is $c_{\text{SVD}}|E_\kappa|^2s_\kappa^2$. The cost of matrix multiplication to obtain $U_\kappa$ is $c_{\text{MM}}|Q_\kappa|s_\kappa t_\kappa$. So the
Localized spectrum slicing

The total cost for finding the LSS basis set is proportional to

$$\sum_{\kappa=1}^{M} c_{Eig,i} |Q_{\kappa}|^3 + c_{SVD} |E_{\kappa}| s_{\kappa}^2 + c_{MM} |Q_{\kappa}| s_{\kappa} t_{\kappa} = n \left( c_{Eig,i} |Q_{\kappa}|^3 + c_{SVD} s_{\kappa}^2 + c_{MM} st_{\kappa} \right).$$

(3.14)

If we assume that as $n$ increases, the spectral radius of $A$ does not increase, then all constants in the parenthesis in the right hand side of Eq. (3.14) are independent of $n$, and the overall computational complexity for finding the LSS basis set is $O(n)$.

In practice the constant for the finding the local eigen-decomposition can be large due to the term $|E| s_{\kappa}^2$ in Eq. (3.14). Since $A_{\kappa}$ is still a sparse matrix on $Q_{\kappa}$, iterative methods can be used to reduce the computational cost to $c_{Eig,i} |Q_{\kappa}| s_{\kappa}^2$. This modifies the overall complexity to be

$$n \left( c_{Eig,i} |Q_{\kappa}| s_{\kappa}^2 + c_{SVD} s_{\kappa}^2 + c_{MM} st_{\kappa} \right).$$

However, it should be noted that the preconstant $c_{Eig,i}$ might be larger than $c_{Eig,d}$. Whether direct or iterative method should be used to solve the local eigenvalue problem may depend on a number of practical factors such as the size of the local problem, the availability of efficient preconditioner on the local domain etc.

4. Compute interior eigenvalues. Using the LSS basis set in (3.8), one may compute the interior eigenvalues near $\mu$ together with its associated eigenvectors. This can be done by using the projected matrices $A_{U}, B_{U}$ according to Eq. (1.3). Due to the spatial sparsity of $U$, $A_{U}, B_{U}$ are also sparse matrices, and can be assembled efficiently with local computation. First, the matrix multiplication $Z = AU$ can be performed locally. This is because each column of $U_{\kappa}$ is localized in $Q_{\kappa}$. In particular, if each column of $U_{\kappa}$ vanishes near the boundary of $Q_{\kappa}$, then

$$Z_{\kappa} = AU_{\kappa} = A_{\kappa} U_{\kappa}. \quad (4.1)$$

Second, denote by

$$(A_{U})_{\kappa', \kappa} = U_{\kappa'}^* Z_{\kappa}, \quad (B_{U})_{\kappa', \kappa} = U_{\kappa'}^* U_{\kappa},$$

then for each $\kappa$ it is sufficient to loop over elements $E_{\kappa'}$ so that $Q_{\kappa'} \cap Q_{\kappa}$ is non-empty. The details for constructing the projected matrices are as given in Algorithm 2.

After $A_{U}, B_{U}$ are assembled, the eigenvalues and corresponding eigenvectors near $\mu$ can be solved in various ways. When the size of the LSS basis set $n_b$ is small, one can treat $A_{U}, B_{U}$ as dense matrices and solve the generalized eigenvalue problem

$$A_{U} C = B_{U} C \Theta, \quad (4.2)$$

and only keep the Ritz values $\Theta = \text{diag}[\theta_1, \ldots, \theta_{n_b}]$ and corresponding Ritz vectors $C$ near $\mu$. Each column of the Ritz vector $C_j$ can be partitioned according to the element partition $\{E_{\kappa}\}$ as

$$C_j = [C_{1,j}, \ldots, C_{M,j}]^T.$$

Then an approximate eigenvector for $A$ can be computed as

$$\tilde{X}_j = UC_j = \sum_{\kappa} U_{\kappa} C_{\kappa, j}. \quad (4.3)$$
**Algorithm 2:** Assembly of the projected matrices.

\[ \text{(1) Sparse Hermitian matrix } A. \]

**Input:**
\[ \text{(2) Number of elements } M, \text{ partition of elements } \{ E_\kappa \}_{\kappa=1}^M, \text{ extended elements } \{ Q_\kappa \}_{\kappa=1}^M, \text{ submatrices } \{ A_\kappa \}_{\kappa=1}^M, \text{ LSS basis set } \{ U_\kappa \}_{\kappa=1}^M \text{ with total number of basis functions } n_b. \]

**Output:** Projected matrices \( A_U, B_U \).

Let \( A_U, B_U \) be zero matrices of size \( n_b \times n_b \).

\[
\text{for } \kappa = 1, \ldots, M \text{ do}
\]

\[
\text{Compute } Z_\kappa \leftarrow A_\kappa U_\kappa;
\]

\[
\text{for } \kappa' \text{ so that } Q_\kappa' \cap Q_\kappa \neq \emptyset \text{ do}
\]

\[
\text{Compute } (A_U)_{\kappa',\kappa} \leftarrow U_{\kappa'}^* Z_\kappa;
\]

\[
\text{Compute } (B_U)_{\kappa',\kappa} \leftarrow U_{\kappa'}^* U_\kappa;
\]

\[
\text{end}
\]

\[
\text{end}
\]

\[
\text{Symmetrize } A_U \leftarrow \frac{1}{2} (A_U + A_U^*), \quad B_U \leftarrow \frac{1}{2} (B_U + B_U^*).
\]

We remark that in the computation of interior eigenvalues, spurious eigenvalues may appear. A spurious eigenvalue is a Ritz value \( \theta_j \) near the vicinity of \( \mu \) as obtained from Eq. (4.2), but the corresponding vector \( \tilde{X}_j \) as given in Eq. (4.3) is not an approximate eigenvector. The appearance of spurious eigenvalue is also referred to as spectral pollution [11, 12], and can be identified by computing the residual

\[
R_j = A \tilde{X}_j - \tilde{X}_j \theta_j.
\]

(4.4)

A Ritz value \( \theta_j \) corresponding to large residual norm \( \| R_j \|_2 \) should be removed. Note that the residual can also be computed with local computation

\[
R_j = \sum_{\kappa} (Z_\kappa C_{\kappa,j} - U_\kappa C_{\kappa,j} \theta_j),
\]

(4.5)

where \( Z_\kappa \) is given in [4.1].

### 5. Numerical results.

In this section we demonstrate the accuracy and efficiency of the divide-and-conquer procedure for computing the LSS operator and the LSS basis set, and for computing interior eigenvalues. All the computation is performed on a single computational thread of an Intel i7 CPU processor with 64 gigabytes (GB) of memory using MATLAB. The matrix \( A \) is obtained from a discretized second order partial differential operator \(-\Delta + V\) in one-dimension (1D) and in two-dimension (2D) with periodic boundary conditions, respectively.

#### 5.1. One-dimensional case.

In the 1D case, the global domain is \( \Omega = [0, L] \). The Laplacian operator is discretized using a 3-point finite difference stencil. The domain is uniformly discretized into \( n = c_n M \) grid points so that \( x_i = (i - 1)h \), with the grid spacing \( h \equiv L/n = 0.1 \). All the \( n \) grid points (vertices) are uniformly partitioned into \( M \) elements \( \{ E_\kappa \}_{\kappa=1}^M \). For simplicity let \( Q_\kappa \) be the union of \( E_\kappa \) and its two neighbors taking into account the periodic boundary condition, i.e.

\[
Q_\kappa = \begin{cases} 
E_M \cup E_1 \cup E_2, & \kappa = 1, \\
\bigcup_{\kappa' = \kappa-1}^{\kappa+1} E_{\kappa'}, & \kappa = 2, \ldots, M - 1, \\
E_{M-1} \cup E_M \cup E_1, & \kappa = M.
\end{cases}
\]
The potential \( V(x) \) is given by the sum of \( n_w \) exponential functions as

\[
V(x) = -\sum_{i=1}^{n_w} a_i e^{-\text{dist}(x, R_i) \delta_i}.
\]  

(5.1)

Here \( \{R_i\} \) are a set of equally spaced points. The distance between two points \( x \) and \( x' \) is defined to be the minimal distance between \( x \) and all the periodic images of \( x' \), i.e.

\[
\text{dist}(x, x') = \min_{\tilde{x}' = x' + kL, k \in \mathbb{Z}} |x - \tilde{x}'|.
\]

In order to study the performance of the algorithm for systems of increasing sizes, we set \( L = 20n_w \) so that the length of the computational domain is proportional to the number of potential wells \( n_w \). To show that we do not take advantage of the periodicity of the potential, we introduce some randomness in each exponential function. We choose \( a_i \sim \mathcal{N}(5.0, 1.0) \), which is a Gaussian random variable with a mean value 5.0 and a standard deviation 1.0. Similarly the width of the exponential function \( \delta_i \sim \mathcal{N}(2.0, 0.2) \). One realization of the potential with \( n_w = 8 \) is given in Fig. 5.1 (a), with the partition of elements indicated by black dashed lines. For the choice of parameter \( \mu = 2.0 \) and \( \sigma = 1.0 \), Fig. 5.1 (b) shows the function \( f_{\sigma, \mu}(\lambda) \) evaluated on the eigenvalues of \( A \), and the LSS operator \( f_{\sigma, \mu}(A) \) is spectrally localized.

Fig. 5.1: (a) One realization of the 1D potential with \( n_w = 8 \). The domain is partitioned into 8 equally sized elements separated by black dashed lines. (b) The function \( f_{\sigma, \mu}(A) \) with \( \sigma = 1.0 \) and \( \mu = 2.0 \) viewed spectrally. Each circle represents an eigenvalue of \( A \). The spectral radius of \( A \) is 199.89.

Fig. 5.2 (a)-(c) demonstrates the behavior of the exact LSS operator \( f_{\sigma, \mu}(A) \) with \( \sigma = 1.0 \) and increasing value of \( \mu \). In Fig. 5.2 (d)-(f) \( f_{\sigma, \mu}(A))(x, y) \) should be interpreted using its discretized matrix element \( [f_{\sigma, \mu}(A)]_{ij} \) for \( x = (i-1)h, y = (j-1)h \). We find that as \( \mu \) increases, the off-diagonal elements of \( f \) decays rapidly and remains to be well approximated by a banded (and therefore sparse) matrix with increasing bandwidth. Fig. 5.2 (d)-(f) demonstrates the quality of the divide-and-conquer approximation \( \tilde{f} \) to the LSS operator. Here we first demonstrate the accuracy of \( \tilde{f} \) without the truncation using SVD decomposition (i.e. the SVD truncation criterion \( \tilde{\tau} = 0 \) as in Eq. 3.6). When \( \mu = -2.0 \), the approximation is nearly exact, while when \( \mu \) increases to 20.0
varying $\sigma$ exponentially with the increase of $\sigma$ estimation leads to large error. As spectrally, but the matrix is almost dense. Therefore the divide-and-conquer approximation leads to large error. As $\sigma$ increases above 10, the vectors spanning columns of $f_{\sigma,\mu}(A)$ are approximately linear combination of high frequency Fourier modes, and Fig. 5.3 (a) shows that the Fourier modes are increasingly more difficult to localize as the frequency increases. Fig. 5.3 (c)-(d) shows similar behavior for $\sigma = 2.0$. The profile of the error with respect to $\mu$ closely resembles a Gaussian function. Compared to the case with $\sigma = 1.0$ the error significantly reduces for all $\mu$, indicating the balance between spatial locality and spectral locality with varying $\sigma$.

Fig. 5.4 (a) demonstrates the max norm error of the LSS operator for $\mu = 2.0$ with increasing value of $\sigma$. When $\sigma$ is less than 0.25 the LSS operator is very localized spectrally, but the matrix is almost dense. Therefore the divide-and-conquer approximation leads to large error. As $\sigma$ increases above 0.25, the max norm error decreases exponentially with the increase of $\sigma$. We observe that the choice of $\sigma$ is crucial: by varying $\sigma$ from 0.5 to 1.5, the error is reduced by over 6 orders of magnitude from

Fig. 5.2: The LSS operator $f_{\sigma,\mu}(A)$ with $\sigma = 1.0$ for (a) $\mu = -2.0$ (b) $\mu = 2.0$ (c) $\mu = 20.0$. The max error between the LSS operator and its divide-and-conquer approximation $\tilde{f}_{\sigma,\mu}(A)$ (d) $\mu = -2.0$ (e) $\mu = 2.0$ (f) $\mu = 20.0$. The relative error is around 10% since the support size of each column of $f$ already extends beyond each extended element $Q_\kappa$. A more complete picture of the $\mu$-dependence for approximating the LSS operator is given in Fig. 5.3. Fig. 5.3 (a) shows the max norm error of the divide-and-conquer approximation $\tilde{f}_{\sigma,\mu}(A)$ for $\mu$ traversing the entire spectrum of $A$ from $-3.0$ to 200.0. The error increases rapidly as $\mu$ initially increases, achieves its maximum at $\mu = 100$ and then starts to decrease. Fig. 5.3 (b) shows the same picture but zooms into the interval near $\mu = 0$. As $\mu$ increases above 10.0, the vectors spanning columns of $f_{\sigma,\mu}(A)$ are approximately linear combination of high frequency Fourier modes, and Fig. 5.3 (a) shows that the Fourier modes are increasingly more difficult to localize as the frequency increases. Fig. 5.3 (c)-(d) shows similar behavior for $\sigma = 2.0$. The profile of the error with respect to $\mu$ closely resembles a Gaussian function. Compared to the case with $\sigma = 1.0$ the error significantly reduces for all $\mu$, indicating the balance between spatial locality and spectral locality with varying $\sigma$. Fig. 5.4 (a) demonstrates the max norm error of the LSS operator for $\mu = 2.0$ with increasing value of $\sigma$. When $\sigma$ is less than 0.25 the LSS operator is very localized spectrally, but the matrix is almost dense. Therefore the divide-and-conquer approximation leads to large error. As $\sigma$ increases above 0.25, the max norm error decreases exponentially with the increase of $\sigma$. We observe that the choice of $\sigma$ is crucial: by varying $\sigma$ from 0.5 to 1.5, the error is reduced by over 6 orders of magnitude from
10^{-4} to below 10^{-10}.

Next we study the effect of grid refinement by varying the grid size from $h = 0.20$ to $h = 0.033$. For 3-point finite difference stencil the spectral radius of $A$, denoted by $\Delta E$ is proportional to $1/h^2$, and in practice $\Delta E$ increases from 50 to 1800. We note that Theorem 2.2 indicates that the error should be determined by the ratio $\sigma/\Delta E$, and therefore the size of the extended element as characterized by the geodesic distance $m$ should increase proportionally to $\Delta E$ to preserve accuracy. Here instead we fix the number of elements to be 8 as the grid refines. Therefore $m \sim 1/h \sim \sqrt{\Delta E}$, and we should expect that the error increases as the grid refines. Fig. 5.4 (b) shows that max norm error of the LSS operator for $\mu = 2.0, \sigma = 1.0$, with increasing $\Delta E$. As the ratio $\sigma/\Delta E$ decreases over one order of magnitude, the max norm error does not increase, but rather decreases by more than a factor of 2. We note that this numerical result does not contradict the theoretical prediction, since Theorem 2.2 only provide an upper bound of the decay rate, and the actual decay rate can be faster. Note that as the grid refines, the change towards the high end of the spectrum is often larger than the change at the low end of the spectrum. Fig. 5.4 indicates that the accuracy of the LSS operator is relatively insensitive to the change in the high end of the spectrum, and it may be possible to construct the LSS operator with improved
discretization scheme, without sacrificing too much in terms of the spatial locality.

![Graph](image1)

Fig. 5.4: Max norm error of the LSS operator for (a) $\mu = 2.0, \Delta E = 199.89$ and increasing value of $\sigma$. (b) $\mu = 2.0, \sigma = 1.0$ and increasing value of $\Delta E$.

So far the numerical results are obtained for the divide-and-conquer approximation to the LSS operator with $\tilde{\tau} = 0$. Next we apply the SVD truncation to obtain the LSS basis set $\{U_\kappa\}_{\kappa=1}^M$ for varying SVD relative truncation criterion. In our numerical experiments, we use $\tau$ as the relative SVD truncation criterion with respect to the largest singular value of $\tilde{S}_\kappa$. Fig. 5.5 shows the error of the approximation to the LSS operator with $\tau$ being 0.001, 0.01, 0.1, respectively. As indicated in Eq. (3.6), the max norm error of the approximation of the LSS operator is approximately proportional to $\tau$, as $\tau$ becomes dominant in Eq. (3.10).

![Graph](image2)

Fig. 5.5: Error of the divide-and-conquer approximation to the LSS operator with $\sigma = 1.0, \mu = 2.0$ and different SVD relative truncation criterion (a) $\tau = 10^{-3}$ (b) $\tau = 10^{-2}$ (c) $\tau = 10^{-1}$.

The LSS basis set comes from the SVD decomposition of $\tilde{f}$ on each element. Fig. 5.6 (a) shows the 1-st LSS basis function on two elements $\kappa = 2$ and $\kappa = 6$, respectively, and Fig. 5.6 (b) shows the 5-th LSS basis function on the same two elements for $\mu = 2.0, \sigma = 1.0$. It is clear that each LSS basis function is well localized in each extended element $Q_\kappa$ and its center is in $E_\kappa$. 

![Graph](image3)
Localized spectrum slicing

Fig. 5.6: Example of the LSS basis function on two elements $\kappa = 2$ and $\kappa = 6$ for (a) the 1-st LSS basis function and (b) the 5-th LSS basis function.

Fig. 5.5 seems to suggest that in order to accurately compute the interior eigenvalues, a very tight SVD criterion $\tau$ is needed. However, we note that many of the LSS basis functions associated with the small singular values actually corresponds to the tail of the Gaussian function in (1.2) which are away from $\mu$. Therefore in order to compute the interior eigenvalues near $\mu$ accurately, it is possible to use a much larger value of $\tau$. Fig. 5.7 (a) shows the difference between the 24 eigenvalues of $A$ within the interval $(\mu - 0.5\sigma, \mu + 0.5\sigma)$ and the corresponding Ritz values of $A$ with $\tau = 0.1$. The computed Ritz values are highly accurate and the maximum error is under $5 \times 10^{-6}$ even though a large SVD truncation criterion $\tau$ is used. Section 4 discusses the identification of spurious eigenvalues using the residual for each computed Ritz value. Indeed within the interval $(\mu - 0.5\sigma, \mu + 0.5\sigma)$ we find 25 Ritz values, and the 1 additional Ritz value should be a spurious eigenvalue. Fig. 5.7 (b) shows $\|R_j\|_2$ for each Ritz value, and we identify that the 11-th Ritz value has a much larger residual than the rest and should be removed. After removing this spurious eigenvalue, the remaining Ritz values become accurate approximation to the eigenvalues as indicated in Fig. 5.7 (a).

While the accuracy of the divide-and-conquer approximation to the LSS operator improves as the SVD truncation criterion $\tau$ decreases, using a very small value of $\tau$ may result in ill-conditioned projection matrices $A_U$ and $B_U$, i.e. some of the LSS basis functions can be approximately represented as the linear combination of other LSS basis functions. Fig. 5.8 (a) shows the condition number of $A_U$, $B_U$ with respect to $\tau$. The condition numbers are below $10^4$ when $\tau \geq 0.1$, and increase very rapidly to $10^{13}$ for $\tau = 10^{-3}$. In the latter case, numerical results obtained from the generalized eigenvalue solver cannot be trusted. Decreasing $\tau$ also leads to increase of the size of the LSS basis set. As $\tau$ decreases from $10^{-1}$ to $10^{-3}$, the number of LSS basis functions increase from 87 to 173. The accuracy of the LSS basis set for different values of $\tau$ is given in Table 5.1. When $\tau$ is too small, the number of computed Ritz values is less than 24 due to the very large condition number of the generalized eigenvalue problem, and the difference between the eigenvalues and the Ritz values is not a meaningful quantity to report and is reported as N/A. The error of the Ritz values reaches its minimum near $\tau = 0.032$ at only $7.59 \times 10^{-8}$, and then starts
to increase as $\tau$ increases. We observe that even if $\tau = 0.316$, the absolute (and relative) error of the Ritz values is still within 0.2%. For this case the dimension of the projected generalized eigenvalue problem is 62, which is much smaller compared to the dimension of $A$ which is 1600.

Even for the 1D simple example, the LSS basis set can be an efficient way to compute interior eigenvalue problems compared to the solution of the eigenvalue problem directly. For comparison of efficiency and accuracy, MATLAB’s sparse eigenvalue solver $\text{eigs}$ is used for the matrix $A$. We acknowledge that $\text{eigs}$ may not be the best eigensolver to use for large interior eigenvalue problems, and other choices such as preconditioned conjugate gradient type of solvers, or Jacobi-Davidson type of solvers may give better results. We also remark that the current implementation of the LSS solver is only for proof of principle, and many of its components can be further optimized before a more thorough performance study is to be performed. Here we consider systems of increasing size by changing $n_w$ in the potential function in
Localized spectrum slicing

| $\tau$  | # Ritz values | $\max_j |\lambda_j - \theta_j|$ |
|---------|---------------|-------------------------------|
| 0.001   | 1             | N/A                           |
| 0.003   | 19            | N/A                           |
| 0.010   | 24            | $2.49 \times 10^{-6}$        |
| 0.032   | 24            | $7.59 \times 10^{-8}$        |
| 0.100   | 24            | $4.40 \times 10^{-6}$        |
| 0.316   | 24            | $1.50 \times 10^{-3}$        |

Table 5.1: The number of computed Ritz values in the interval $(\mu - 0.5\sigma, \mu + 0.5\sigma)$ with $\sigma = 1.0, \mu = 2.0$ (spurious eigenvalues removed). If the number of Ritz values match the number of eigenvalues in the interval (24), then the third column gives the maximum difference between the eigenvalues and the Ritz values. Otherwise the third column gives N/A.

Eq. (5.1) from 8 to 256. Correspondingly the number of grid points $n$ increases from 1600 to 51200, and the number of elements increases proportionally from 8 to 256. $\mu = 2.0, \sigma = 1.0, \tau = 3 \times 10^{-2}$ is used for all systems to compute the eigenvalues within the interval $(\mu - 0.5\sigma, \mu + 0.5\sigma)$. Fig. 5.9 shows the time for computing the interior eigenvalues near $\mu$ using MATLAB’s sparse eigenvalue solver eigs (“Global total”), and the time using the LSS basis set (“LSS total”). The tolerance for eigs is set to $10^{-5}$. The breakdown of the time cost for the LSS solver includes the time for constructing the LSS basis set (“LSS basis”), the time for assembling the projected matrix (“Assembly”), and the time for solving the projected eigenvalue problem (“LSS solve”). Fig. 5.10 shows the sparsity pattern for $A_U$ for $n = 6400$, and the sparsity pattern for $B_U$ is by definition the same. The number of nonzero elements is 15.6% of the total number of elements in $A_U$. The sparsity of the projected matrices is not used in our example here, but can be exploited using alternative methods.

Since the size of the local problem is small, the local eigenvalue problem on each $Q_\kappa$ is performed using MATLAB’s dense eigenvalue solver eig, and so is the solution of the generalized eigenvalue problem for the projected matrix. The time for the global solver scales cubically with respect to $n$, and the constructing the LSS basis and the assembly of the projected matrix increases linearly with respect to $n$. The solution of the generalized eigenvalue problem also scales cubically with respect to $n$, and therefore does not dominate in the LSS solver until $n = 51200$. The cross-over time between the LSS solver and the global solver is around $n = 10000$. For $n = 51200$, the LSS solver costs 46.6 sec, which is 11.2 times faster than the global solver which costs 520.8 sec.

Fig. 5.9 (b) shows the accuracy of the LSS solver. The Ritz values remain as accurate approximation to the eigenvalues as the number of eigenvalues in the interval increases from 24 to 706.

5.2. Two-dimensional case. The setup of the 2D example is similar to that in 1D. The global domain is $\Omega = [0, L] \times [0, L]$, and the Laplacian operator is discretized using a 5-point finite difference stencil. The grid spacing is chosen to be $h = 1.0$. The potential function $V(x, y)$ is given by sum of periodized exponential functions with random heights and widths. One realization of this potential is given in Fig. 5.11.

Let the number of elements $M$ is a square number and the number of grid points $n$ is divisible by $M$. Then all $n$ grid points (vertices) are uniformly partitioned into $\sqrt{M} \times \sqrt{M}$ elements. We also assume each extended element $Q_\kappa$ contains $E_\kappa$ and
its 8 nearest neighbor elements. Fig. 5.11 shows the partition of the 2D domain into $8 \times 8 = 64$ elements separated by black dashed lines.

We compare the accuracy of the LSS basis set by comparing the eigenvalues within the interval $(\mu - \sigma, \mu + \sigma)$ with $\mu = 0.0, \sigma = 1.0$. The SVD relative truncation criterion $\tau$ is set to be $10^{-1}$. Fig. 5.12 (a) shows the error of Ritz values compared to all the 828 eigenvalues within the interval, and the error of all Ritz values is very small, within $5 \times 10^{-5}$. Fig. 5.12 (b) shows the residual of the Ritz values. For all the Ritz values the residual are below $4 \times 10^{-3}$ and no spurious eigenvalue is identified for this case.

Finally we demonstrate the performance of the LSS solver for a 2D problem with increasing size. The number of grid points $n$ increases from 1600 to 25600, and the number of elements increases proportionally from 16 to 256. $\mu = 0.0, \sigma = 1.0, \tau = 1 \times 10^{-1}$ is used for all systems to compute the eigenvalues within the interval $(\mu - \sigma, \mu + \sigma)$. Fig. 5.13 shows the time for computing the interior eigenvalues near $\mu$ using MATLAB’s sparse eigenvalue solver \texttt{eigs} ("Global total"), and the time using the LSS basis set ("LSS total"). The tolerance for \texttt{eigs} is set to $10^{-5}$. The breakdown of the LSS solver includes the time for constructing the LSS basis set ("LSS basis"),
Localized spectrum slicing

Fig. 5.11: One realization of the 2D potential. The domain is partitioned into $8 \times 8 = 64$ elements separated by black dashed lines.

Fig. 5.12: (a) Error of the Ritz values (b) The 2-norm of residual corresponding to Ritz values for the 2D problem with $\mu = 0.0$, $\sigma = 1.0$, $\tau = 10^{-1}$.

the time for assembling the projected matrix (“Assembly”), and the time for solving the projected eigenvalue problem (“LSS solve”). Again the local eigenvalue problem on each $Q_\kappa$ is performed using MATLAB’s dense eigenvalue solver `eig`, and so is the solution of the generalized eigenvalue problem for the projected matrix. The crossover point between the global solver and the LSS solver is around $n = 3000$. For $n = 25600$, the LSS solver costs 419 sec, which is 11.6 times faster than the global solver which costs 4867 sec.

Fig. 5.13 (b) shows the accuracy of the LSS solver. The Ritz values remain as accurate approximation to the eigenvalues as the number of eigenvalues in the interval increases from 200 to 3355, and no spurious eigenvalue is observed for all cases.

6. Concluding. In this paper, we present a method for constructing a novel basis set called the localized spectrum slicing (LSS) basis set. Each function in the LSS basis set is localized both spectrally and spatially, and therefore can be used as an efficient way for representing eigenvectors of a general sparse Hermitian matrix corresponding to a relatively narrow range of eigenvalues. The LSS basis set uses the decay properties of analytic matrix functions, and can be constructed in a divide-and-conquer method. We show that by carefully tuning one parameter $\sigma$, spatial
locality and spectral locality of the basis functions can be balanced. The projected matrices are both sparse and have reduced sizes, which leads to a more efficient way for finding interior eigenvalues without the knowledge of the inverse operator in the form of \((A - \mu I)^{-1}\) on the global domain.

In terms of the future work, the Gaussian function used in the LSS operator is a smooth approximation to the \(\delta\) function. The same concept of locality can be used to approximate other matrix functions, such as matrix sign functions. This aspect is, e.g., closely related to the recently developed adaptive local basis functions [16] and element orbitals [17] for constructing efficient basis functions for solving the Kohn-Sham density functional theory. The LSS basis set can also be used to efficiently characterize the eigenvectors close to the null space of \(A\), which can then be used to construct preconditioners to accelerate linear solves for indefinite problems.

From efficiency point of view, in the current implementation, the local eigenvalue problem is solved using a dense eigenvalue solver. This is still feasible for the 1D and 2D model problems presented in the numerical section in this paper, but for 3D problems this is going to be too expensive. Efficient iterative solvers, or local Chebyshev expansion based schemes should be used instead. Another practical issue is to control the condition number of the LSS basis set when the SVD truncation criterion is small. An efficient way to identify a subset of well conditioned LSS basis functions is currently under development.

The balance between spatial and spectral locality is an important topic in Fourier analysis and multi-resolution analysis. Because the construction of the LSS basis set is completely algebraic and can be applied to any sparse Hermitian matrix, it is possible to extend the current work to construct multi-resolution basis functions tailored for given matrices, or multi-resolution basis functions for operators on graphs.

**Acknowledgments.** This work was supported by Laboratory Directed Research and Development (LDRD) funding from Berkeley Lab, provided by the Director, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.
REFERENCES

[1] M. Benzi, P. Boito, and N. Razouk, Decay properties of spectral projectors with applications to electronic structure, SIAM Rev., 55 (2013), pp. 3–64.
[2] M. Benzi and G. H. Golub, Bounds for the entries of matrix functions with applications to preconditioning, BIT, 39 (1999), pp. 417–438.
[3] M. Benzi and N. Razouk, Decay bounds and $O(n)$ algorithms for approximating functions of sparse matrices, Electron. Trans. Numer. Anal., 28 (2007), pp. 16–39.
[4] D. R. Bowler and T. Miyazaki, $O(N)$ methods in electronic structure calculations, Rep. Prog. Phys., 75 (2012), p. 036503.
[5] W. W.Bradbury and R. Fletcher, New iterative methods for solution of the eigenproblem, Numer. Math., 9 (1966), pp. 259–267.
[6] E.R. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices, J. Comput. Phys., 17 (1975), pp. 87–94.
[7] S. Demko, Inverses of band matrices and local convergence of spline projections, SIAM J. Numer. Anal., 14 (1977), pp. 616–619.
[8] S. Demko, W. F. Moss, and P. W. Smith, Decay rates for inverses of band matrices, Math. Comput., 43 (1984), pp. 491–499.
[9] S. Goedecker, Linear scaling electronic structure methods, Rev. Mod. Phys., 71 (1999), pp. 1085–1123.
[10] G. H. Golub and C. F. Van Loan, Matrix computations, Johns Hopkins Univ. Press, Baltimore, third ed., 1996.
[11] R. Gruber and J. Rappaz, Finite element methods in linear ideal magnetohydrodynamics, Springer Series in Computational Physics, Springer-Verlag, New York, 1985.
[12] A. Knyazev, New estimates for Ritz vectors, Math. Comput., 66 (1997), pp. 985–995.
[13] A. V. Knyazev, Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method, SIAM J. Sci. Comp., 23 (2001), pp. 517–541.
[14] W. Kohn, Density functional and density matrix method scaling linearly with the number of atoms, Phys. Rev. Lett., 76 (1996), pp. 3168–3171.
[15] R. Lehoucq and D. Sorensen, Implicitly restarted Lanczos method (section 4.5), in Templates for the Solution of Algebraic Eigenvalue Problems: a Practical Guide, Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, eds., Philadelphia, 2000, SIAM, pp. 67–81.
[16] L. Lin, J. Lu, L. Ying, and W. E, Adaptive local basis set for Kohn-Sham density functional theory in a discontinuous Galerkin framework I: Total energy calculation, J. Comput. Phys., 231 (2012), pp. 2140–2154.
[17] L. Lin and L. Ying, Element orbitals for Kohn-Sham density functional theory, Phys. Rev. B, 85 (2012), pp. 235144–235153.
[18] G. Meinardus and L. L. Schumaker, Approximation of functions: Theory and numerical methods, Springer, 1967.
[19] G. Nenciu, Existence of the exponentially localised Wannier functions, Comm. Math. Phys., 91 (1983), pp. 81–85.
[20] E. Polizzi, Density-matrix-based algorithm for solving eigenvalue problems, Phys. Rev. B, 79 (2009), pp. 115112–115117.
[21] E. Prodan and W. Kohn, Nearsightedness of electronic matter, Proc. Natl. Acad. Sci., 102 (2005), pp. 11635–11638.
[22] T. Sakurai and H. Sugiuira, A projection method for generalized eigenvalue problems, J. Comput. Appl. Math., 159 (2003), pp. 119–128.
[23] G. Schofield, J. R. Chelekosky, and Y. Saad, A spectrum slicing method for the Kohn-Sham problem, Comp. Phys. Comm., 183 (2012), pp. 497–505.
[24] H. Zhang, B. Smith, M. Sternberg, and P. Zapol, $SIPs$: Shift-and-invert parallel spectral transformations, ACM Trans. Math. Software, 33 (2007), pp. 9–19.