Improving robustness of the FEAST algorithm and solving eigenvalue problems from graphene nanoribbons

Lukas Krämer

GAMM Annual Meeting, Erlangen

Bergische Universität Wuppertal
—Applied Computer Science Group—

March 12th, 2014

Joint work with

- Martin Galgon
- Bruno Lang

from Bergische Universität Wuppertal,

- Andreas Alvermann
- Holger Fehske
- Andreas Pieper

from EMA Universität Greifswald.

This work was partially supported by DFG within the priority programme 1648—Software for Exascale Computing (SPPEXA) within the project ESSEX (Equipping Sparse Solvers for Exascale).

`http://www.sppexa.de`
`http://blogs.fau.de/essex/`

Outline

Symmetric eigenvalue problem

We aim at solving the problem:

Given: $A \in \mathbb{R}^{n\times n},\ A = A^T$,
an interval

$$I_\lambda = \left[\overline{\lambda},\, \underline{\lambda}\right].$$

Sought: $X \in \mathbb{R}^{n\times m}$, $m \le n$, $X^T X = I$,
$\Lambda = \mathrm{diag}(\lambda_1\ \ldots, \lambda_m)$ such that

$$AX = X\Lambda \text{ and all } \lambda_j \in \left[\overline{\lambda},\, \underline{\lambda}\right].$$

✎ Solve the (partial) symmetric eigenproblem. Find all eigenvalues in a given interval.

(Everything can be generalized to complex matrices and to a pair (A, B) with B hpd.)

Rayleigh-Ritz Approach...

...for solving eigenvalue problems (non-orthogonal version):

- Choose a subspace $\mathcal{U} = \operatorname{span}(U)$
- Compute the Rayleigh quotients $A_U := U^H A U,\ B_U := U^H U$
- Compute the primitive Ritz pairs $(\widetilde{\Lambda}, \widetilde{W})$ of $A_U W = B_U W \Lambda$.
- Return the approximate Ritz pairs $(\widetilde{\Lambda}, U\widetilde{W})$ of $AX = X\Lambda$.

...and hope that $(\widetilde{\Lambda}, U\widetilde{W})$ are good approximations to some eigenpairs.

The subspace in use

[E. Polizzi, Phys. Rev. B, 2009, 79, 115112]
In FEAST: The subspace

$$U := \frac{1}{2\pi \mathbf{i}} \int_{\mathcal{C}} (zI - A)^{-1} dz Y$$

is used.

- $\mathcal{C}$ = curve in complex plane surrounding I_λ.
- Y = starting base of (possibly) random vectors.

It follows

$$\frac{1}{2\pi \mathbf{i}} \int_{\mathcal{C}} (zI - A)^{-1} dz = \text{orth. projector onto } \operatorname{span}(X)$$

The FEAST Algorithm: Skeleton

Input: An interval $I_\lambda := [\overline{\lambda}, \underline{\lambda}]$, an estimate $\widetilde{m}$ of the number of eigenvalues in I_λ.

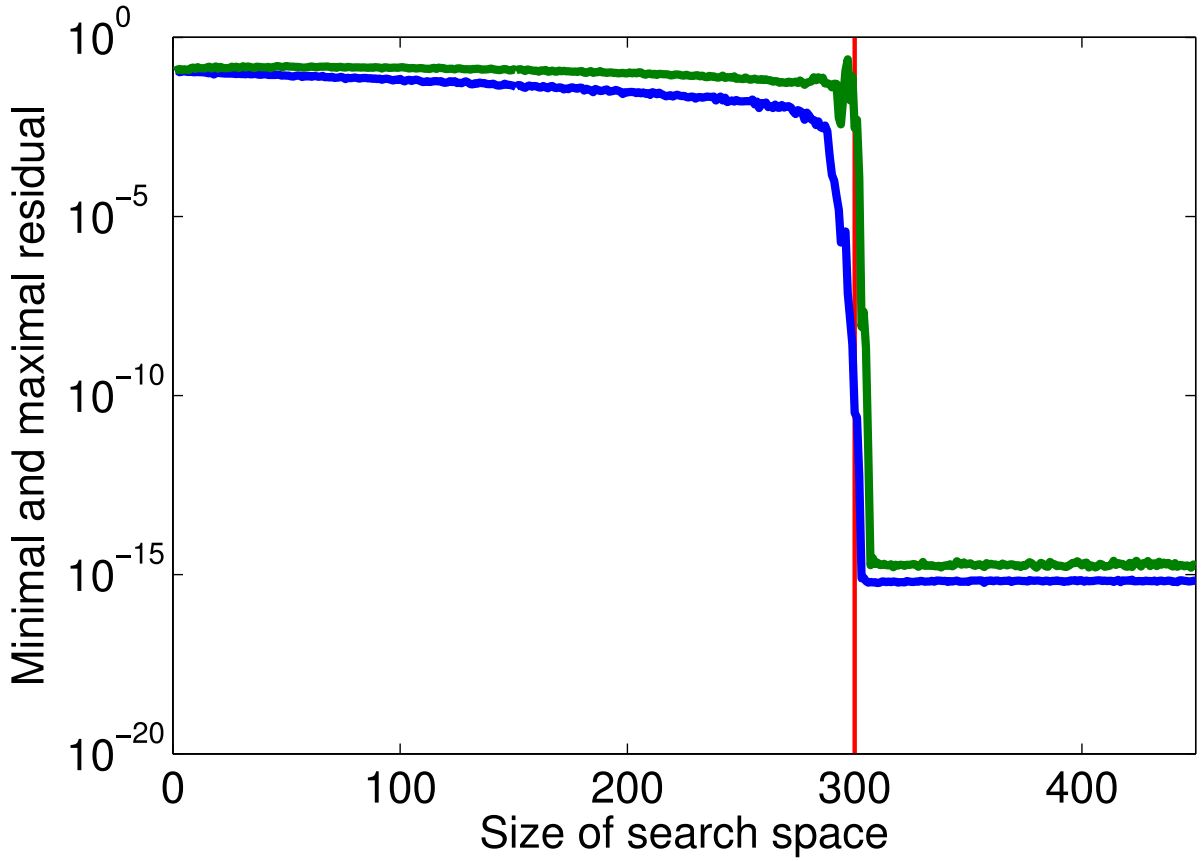
Output $\hat{m} \leq \widetilde{m}$ eigenpairs with eigenvalue in I_λ.

Perform:

1. Choose $Y \in \mathbb{C}^{n \times \widetilde{m}}$ of rank $\widetilde{m}$ and compute
   $U := \frac{1}{2\pi \mathbf{i}} \int_{\mathcal{C}} (zI - A)^{-1} dz\, Y$,
2. Form the Rayleigh quotients $A_U := U^H A U,\ B_U := U^H U$,
3. Solve the size-$\widetilde{m}$ generalized eigenproblem $A_U \widetilde{W} = B_U \widetilde{W} \widetilde{\Lambda}$,
4. Compute the approximate Ritz pairs $(\widetilde{\Lambda}, \widetilde{X} := U \cdot \widetilde{W})$,
5. If convergence is not reached go to Step 1, with $Y := \widetilde{X}$.

Problem: How to choose $\widetilde{m}$ (initial dimension of $\mathcal{U}$)?

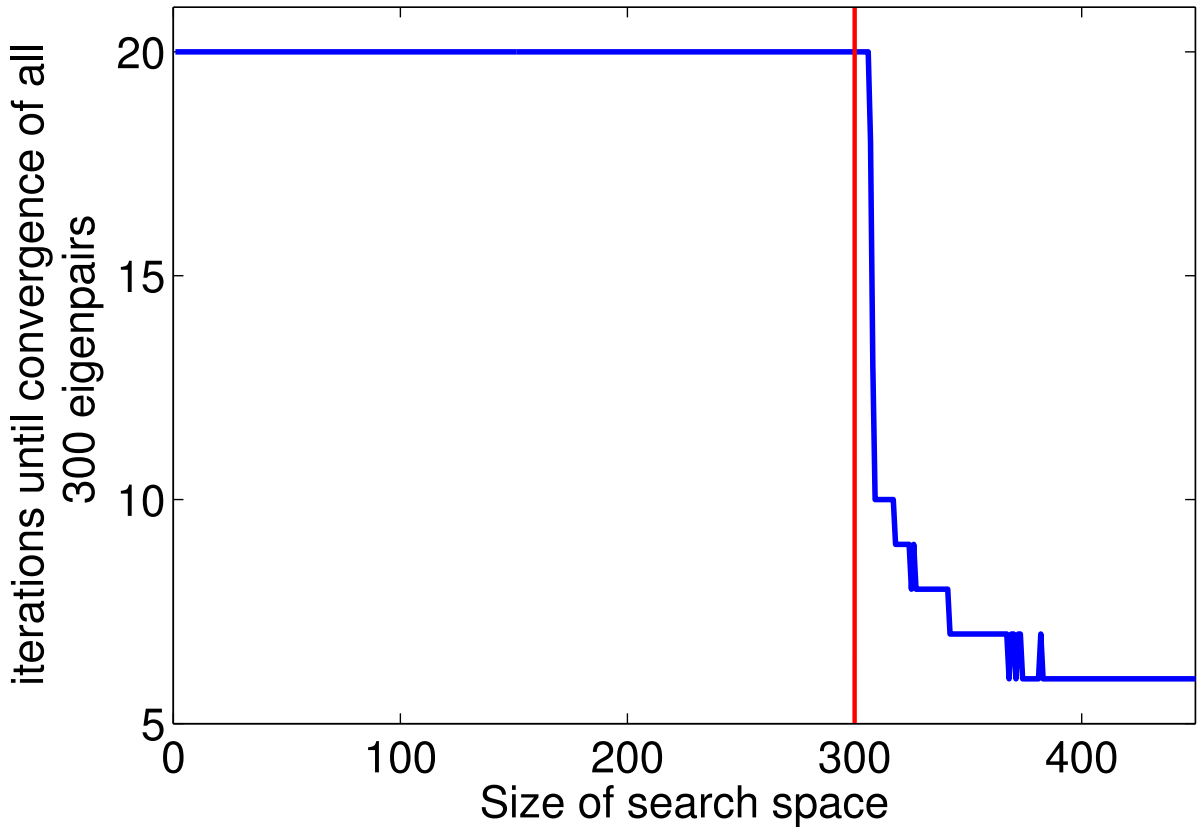
Experiment: Choose $I_\lambda = [\overline{\lambda}, \underline{\lambda}]$ such that it contains the $m = 300$ lowest eigenvalues of size 1059-matrix. Let $\widetilde{m}$ vary, $\widetilde{m} = 1, \ldots, 450$
Residuals*:



*K., Di Napoli, et al.: Dissecting the FEAST algorithm for generalized eigenproblems

Problem: How to choose $\widetilde{m}$ (initial dimension of $\mathcal{U}$)?

Iteration counts[†]:



✎ Never choose $\widetilde{m}$ too small!

What if $\widetilde{m} > m$ (results in too high numerical effort)?

[†]K., Di Napoli, et al.: Dissecting the FEAST algorithm for generalized eigenproblems

Counting eigenvalues

- $\mathrm{rank}(U) = m =$ number of eigenvalues in I_λ
- In exact arithmetic: U has singular values 0 or 1
- numerical rank?

Counting eigenvalues

- $\operatorname{rank}(U) = m =$ number of eigenvalues in I_λ
- In exact arithmetic: U has singular values 0 or 1
- numerical rank?
- It can be shown: If $m \leq \widetilde{m}$ we have $m = \#$ of singular values of U that are $\geq 1/2$.
- Number $1/2$ depends on implementation (integration scheme). E.g., valid for Gauß–Legendre integration, midpoint rule of even order.

Counting eigenvalues: SVD of U

SVD $U = W\Sigma V^T$ with $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_m, 0, \ldots, 0)$

$\Rightarrow$ $\mathrm{span}(W(:, 1:m)) = \mathrm{span}(U)$

$\Rightarrow$ Found the correct subspace, but cost $= \mathcal{O}(n^2 \cdot \widetilde{m})$

SVD of $B_U = U^T U$:

$\Rightarrow$ $B_U = V\Sigma^T\Sigma V^T$ with $V \in \mathbb{R}^{\widetilde{m}\times\widetilde{m}}$, orthogonal. No order n cost.

$\Rightarrow$ $U' = U \cdot V(:, 1:m)$ spans the wanted space $\rightsquigarrow \mathcal{O}(n \cdot m)$.

Counting eigenvalues: SVD of U

SVD $U = W\Sigma V^T$ with $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_m, 0, \ldots, 0)$

$\Rightarrow$ $\mathrm{span}(W(:, 1:m)) = \mathrm{span}(U)$

$\Rightarrow$ Found the correct subspace, but cost $= \mathcal{O}(n^2 \cdot \widetilde{m})$

SVD of $B_U = U^T U$:

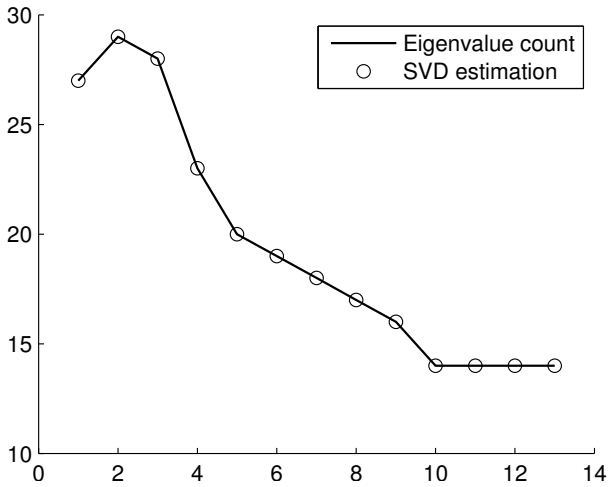
$\Rightarrow$ $B_U = V\Sigma^T\Sigma V^T$ with $V \in \mathbb{R}^{\widetilde{m}\times\widetilde{m}}$, orthogonal. No order n cost.

$\Rightarrow$ $U' = U \cdot V(:, 1:m)$ spans the wanted space $\rightsquigarrow \mathcal{O}(n \cdot m)$.

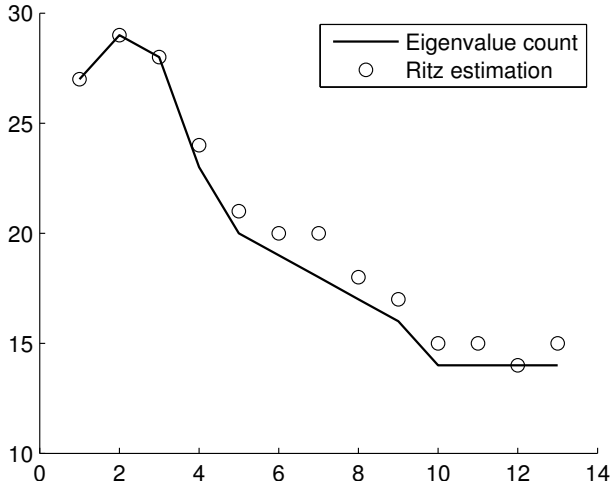
This means the dimension and a basis of the search space can be computed.
Proceed with a subspace slightly larger than m.

Counting eigenvalues: example

SVD count



Number of Ritz values in I_λ



1 Point on abscissa $\equiv$ 1 interval I_λ

$\circ =$ count from respective method

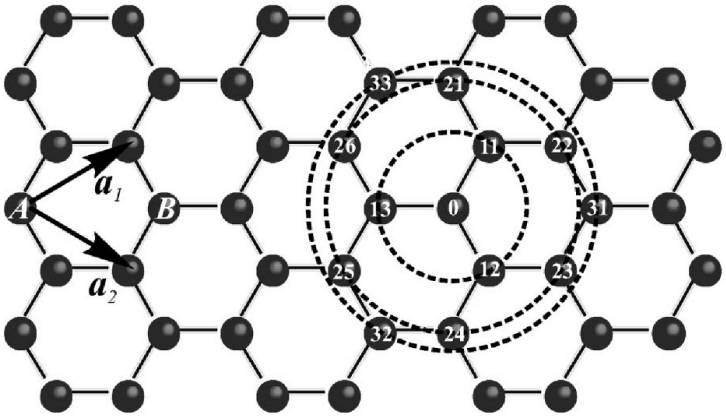
line $=$ exact count

✎ Ritz count is almost for free and not too bad. It does not deliver the subspace. Typically overestimates.

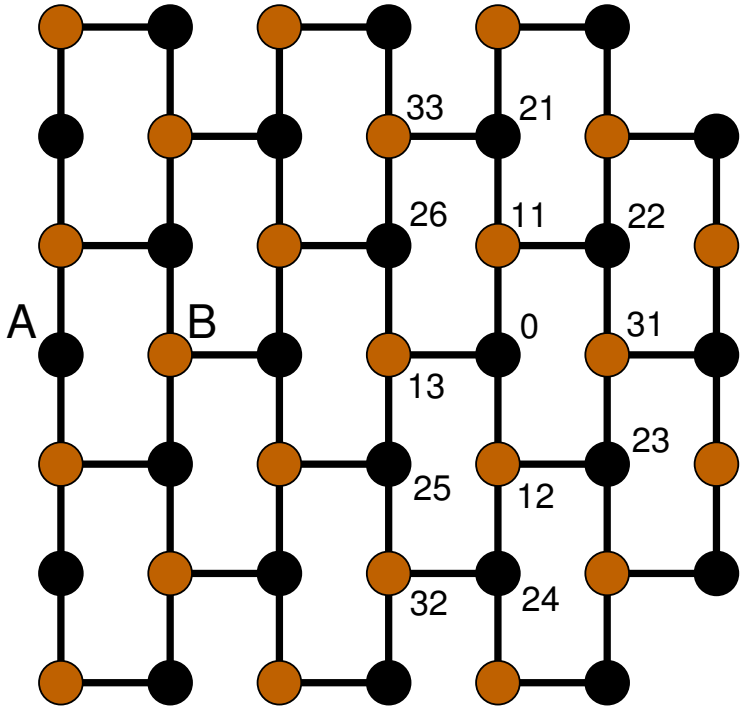
Outline

Graphene nanoribbons...

Hexagonal lattice

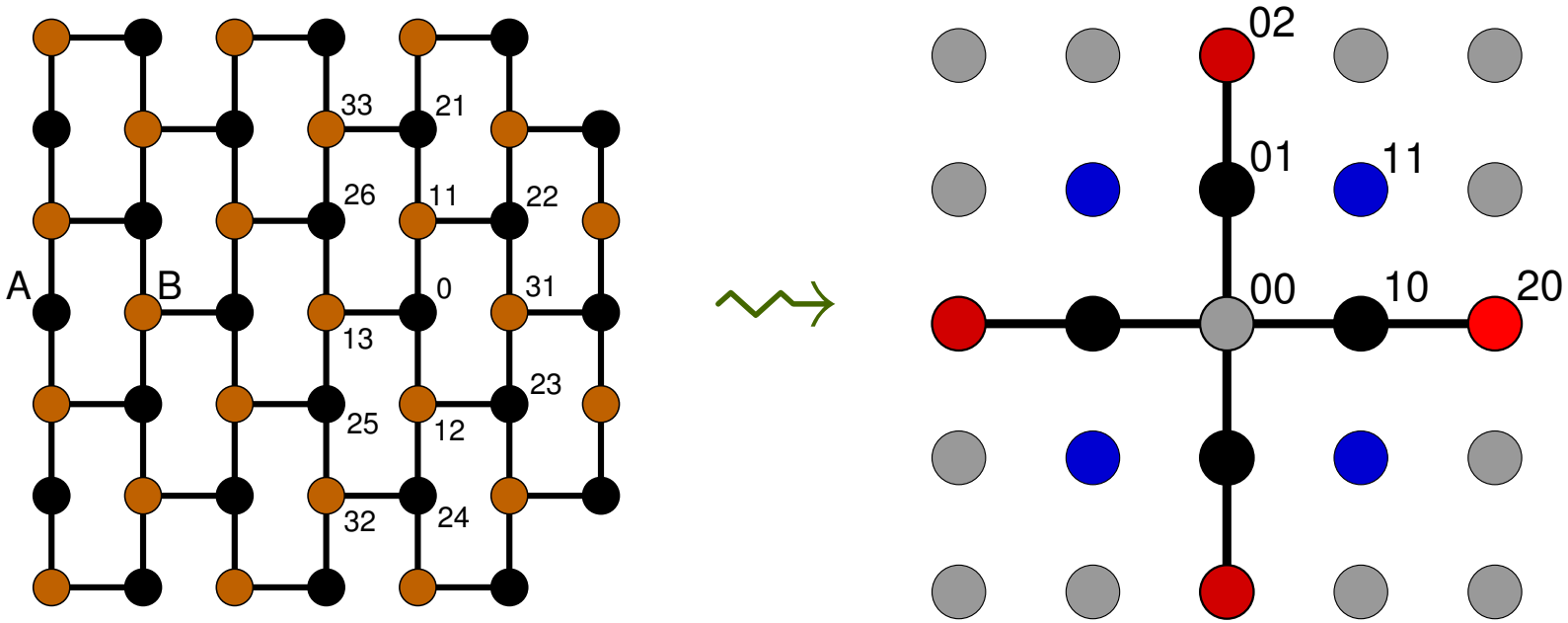


Brick lattice



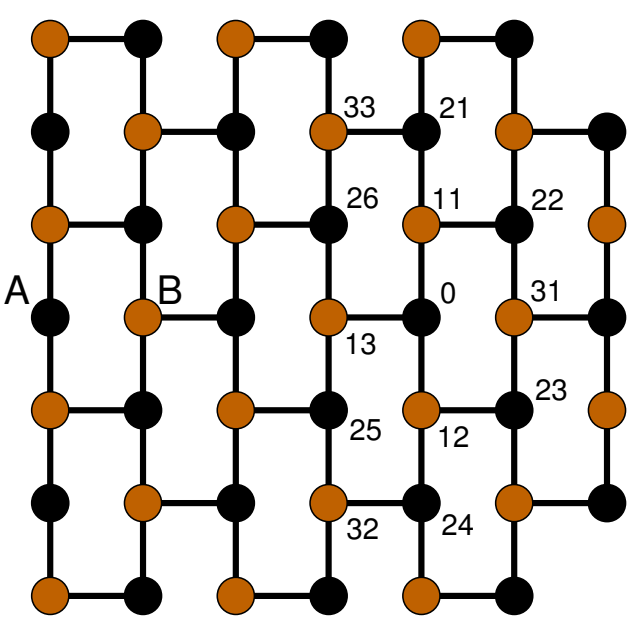
Graphene nanoribbons...

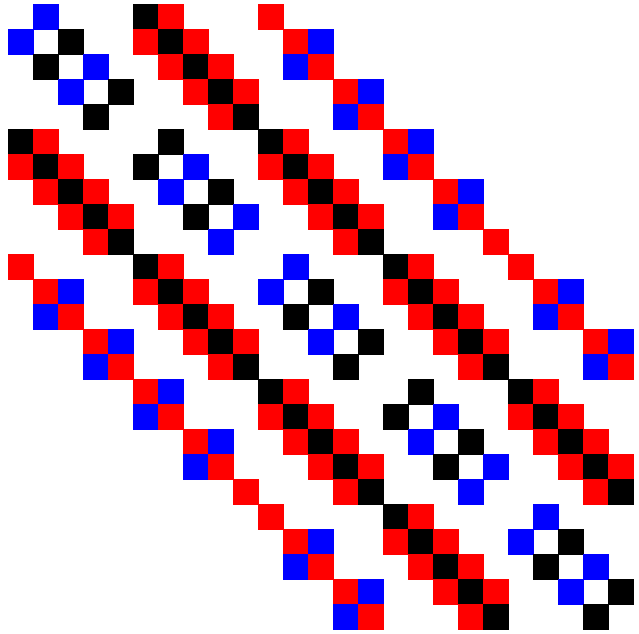
Model connections between neighbored atoms...



Graphene nanoribbons...

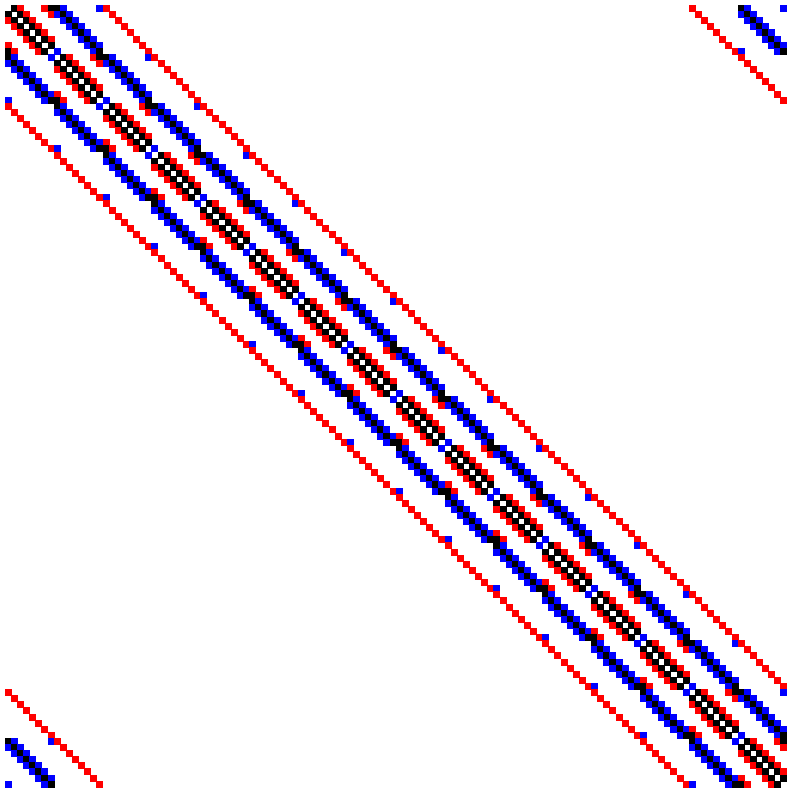
Model connections between neighbored atoms...



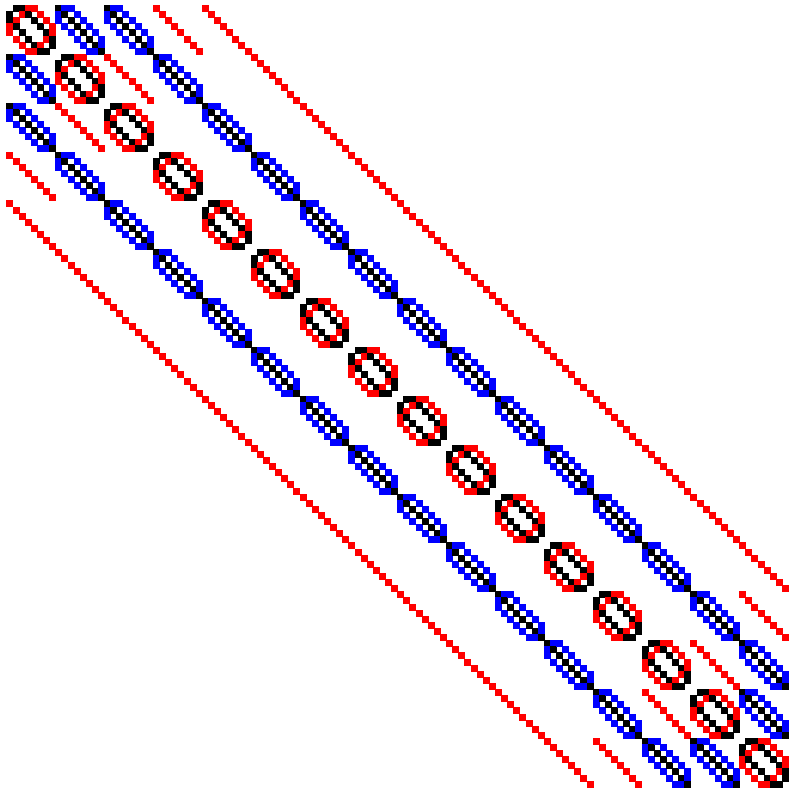


. . . lead to sparse eigenvalue problems

Graphene 8×16, periodic BC



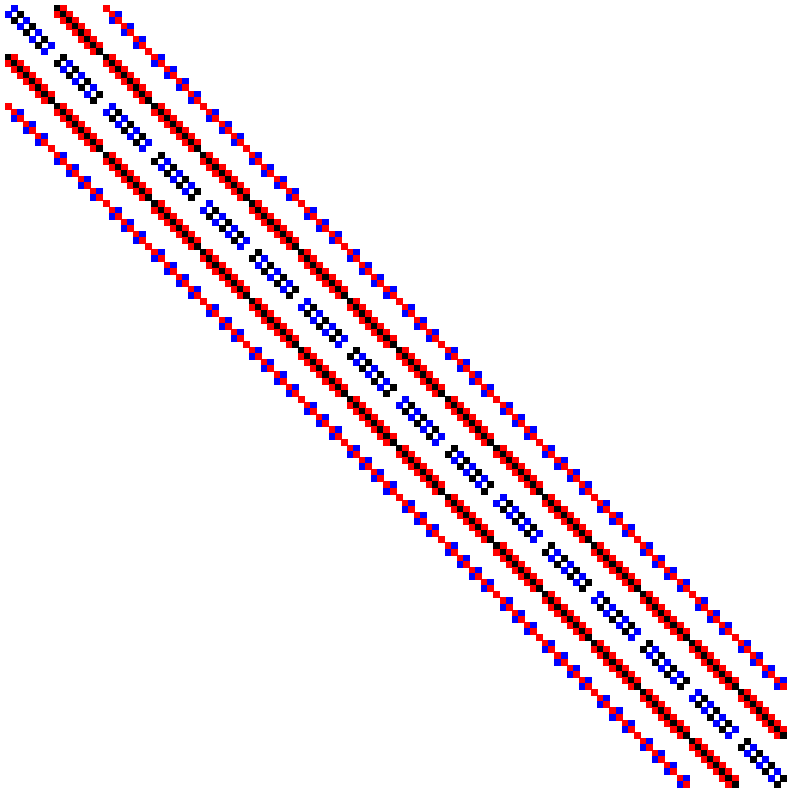
reordering



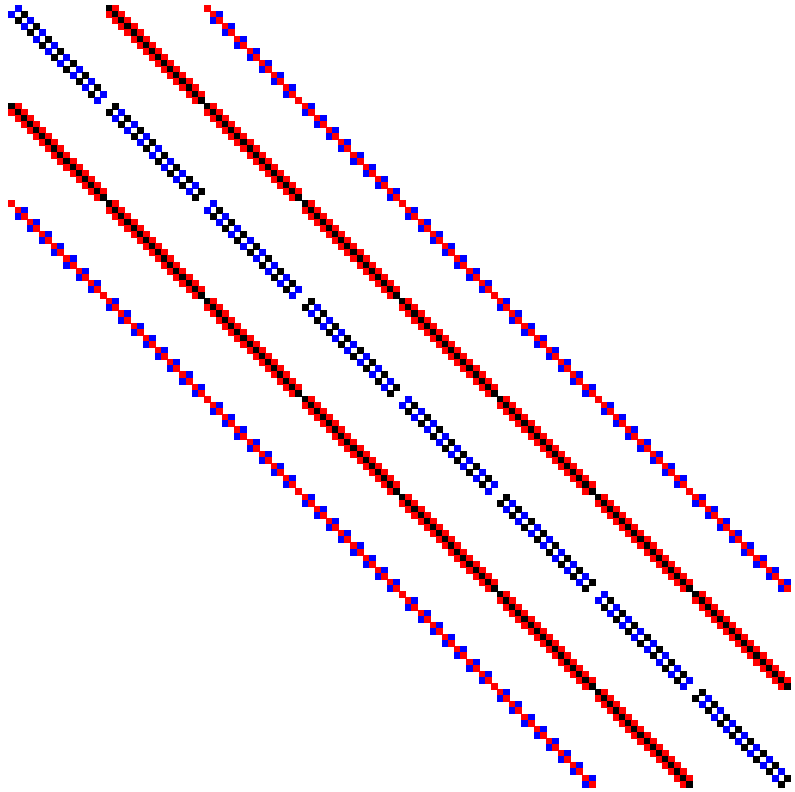
Nearest neighbor, next nearest neighbor, second nearest neighbor

. . . lead to sparse eigenvalue problems

Graphene 8×16



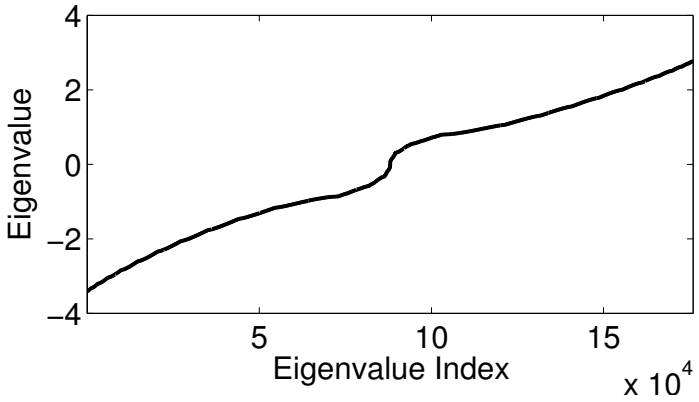
Graphene 16×8



Nearest neighbor, next nearest neighbor, second nearest neighbor
$\Rightarrow$ sparse, symmetric matrix, width of graphene strip is reflected in bandwidth.

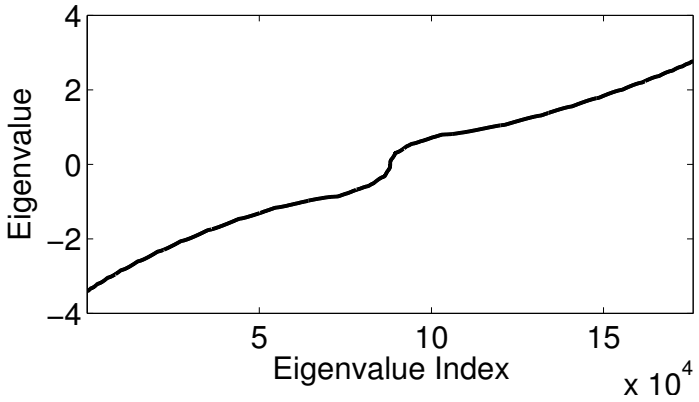
Wanted: Some eigenpairs with eigenvalue close to zero.

Graphene matrix of size $176{,}000$:

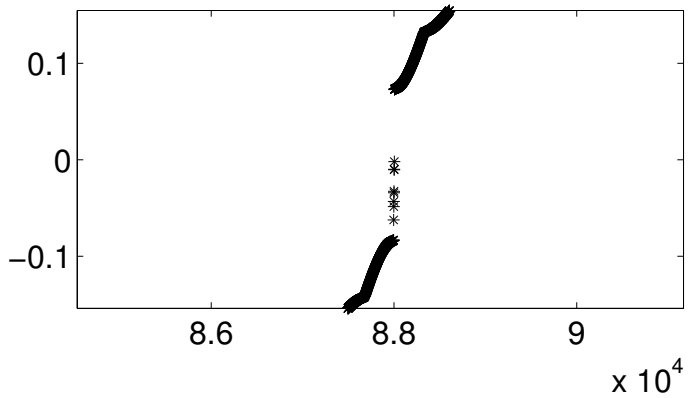


Wanted: Some eigenpairs with eigenvalue close to zero.

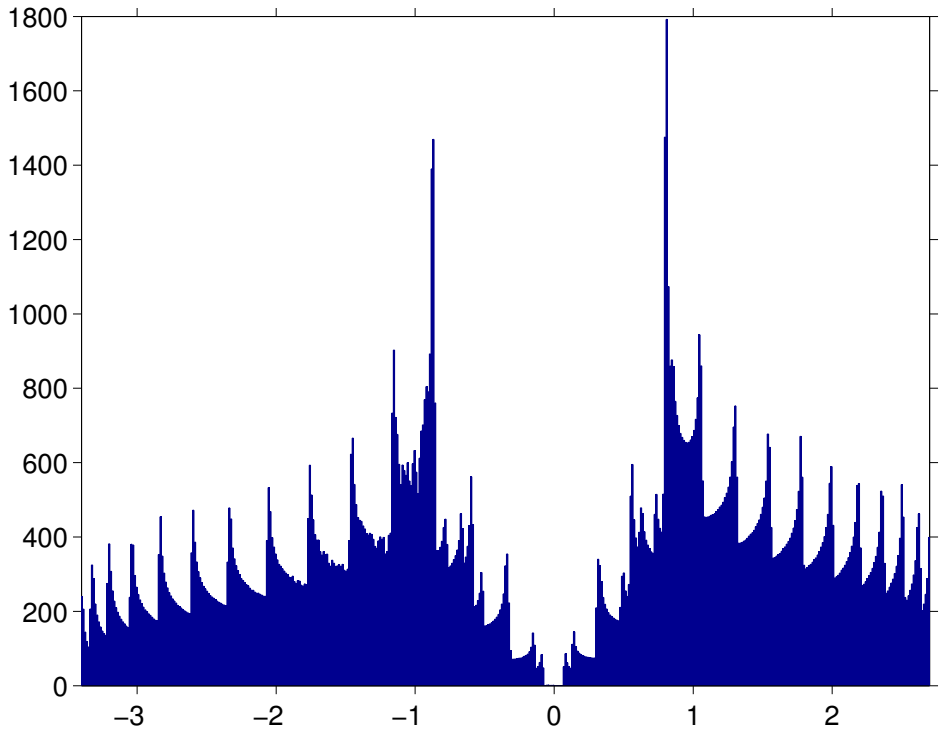
Graphene matrix of size $176{,}000$:



Zoom on eigenvalues around zero:



Spectral density



Outline

Implementation

- Parallel C/MPI implementation incorporating Feast with the discussed eigenvalue counting
- $U = \frac{1}{2\pi \mathbf{i}} \int_{\mathcal{C}} (zI - A)^{-1} dz Y \approx \sum_{j=0}^{p} \omega_j (z_j I - A)^{-1} Y$

  (Gauß–Legendre integration).
- 8 points used
- Solution of $(z_j I - A)V = Y$ via a partitioning method[‡] and recursive doubling (Hockney-Golub) for banded systems.

[‡] D. H. Lawrie, A. H. Sameh, The computation and communication complexity of a parallel banded system solver, ACM TOMS 10(2), 1984.

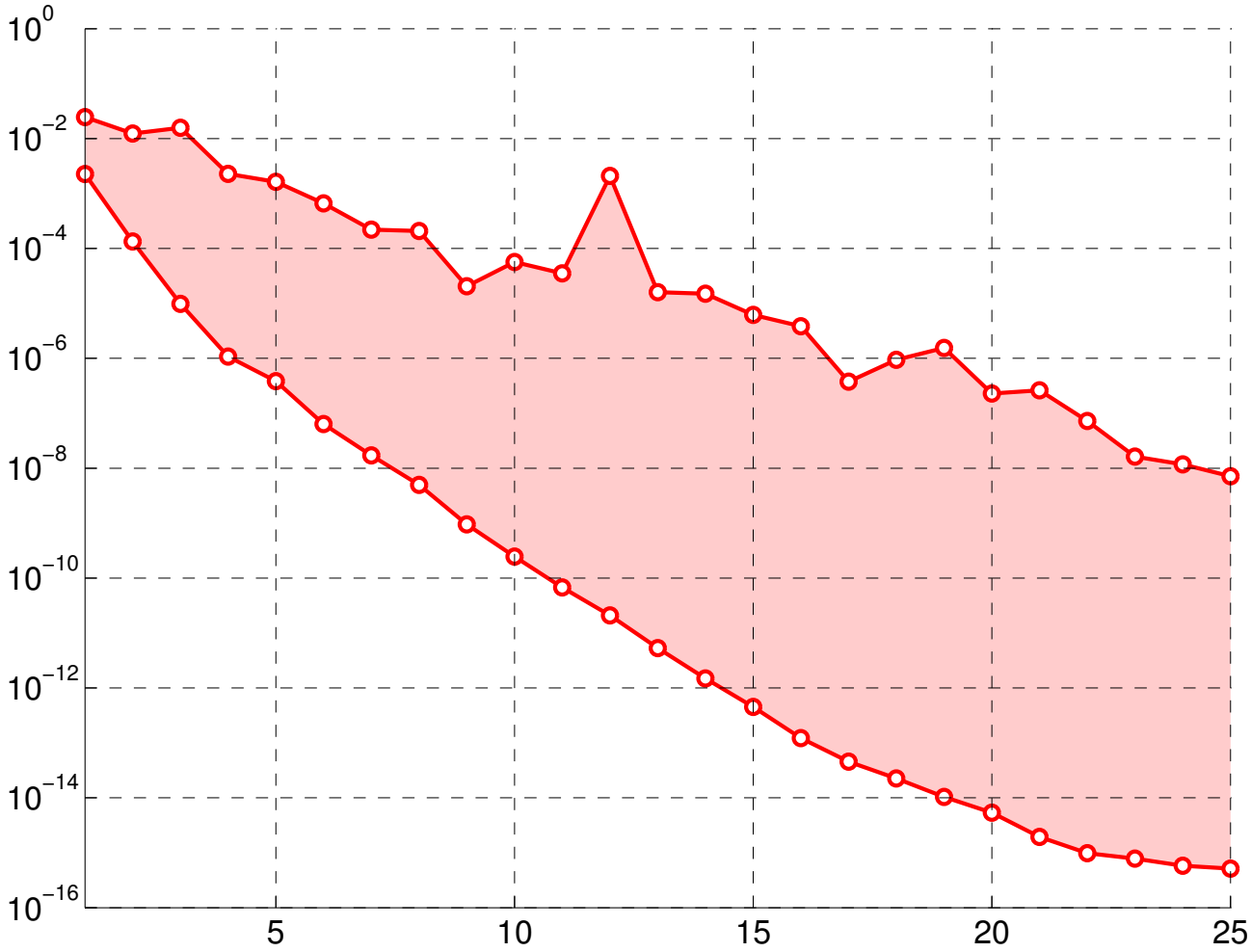
Results: Counting and residuals

- Matrix size $n \approx 1.2M$, ratio $\frac{\text{nnz}}{n^2} \approx 0.001\%$ (12 nonzero entries per row).
- Graphene II with gap around zero.

Matrix	est.	found	iters.
Graphene I	535	535	7
Graphene II (ℓ)	492	492	4
Graphene II (r)	546	546	8
Graphene III	574	572	25

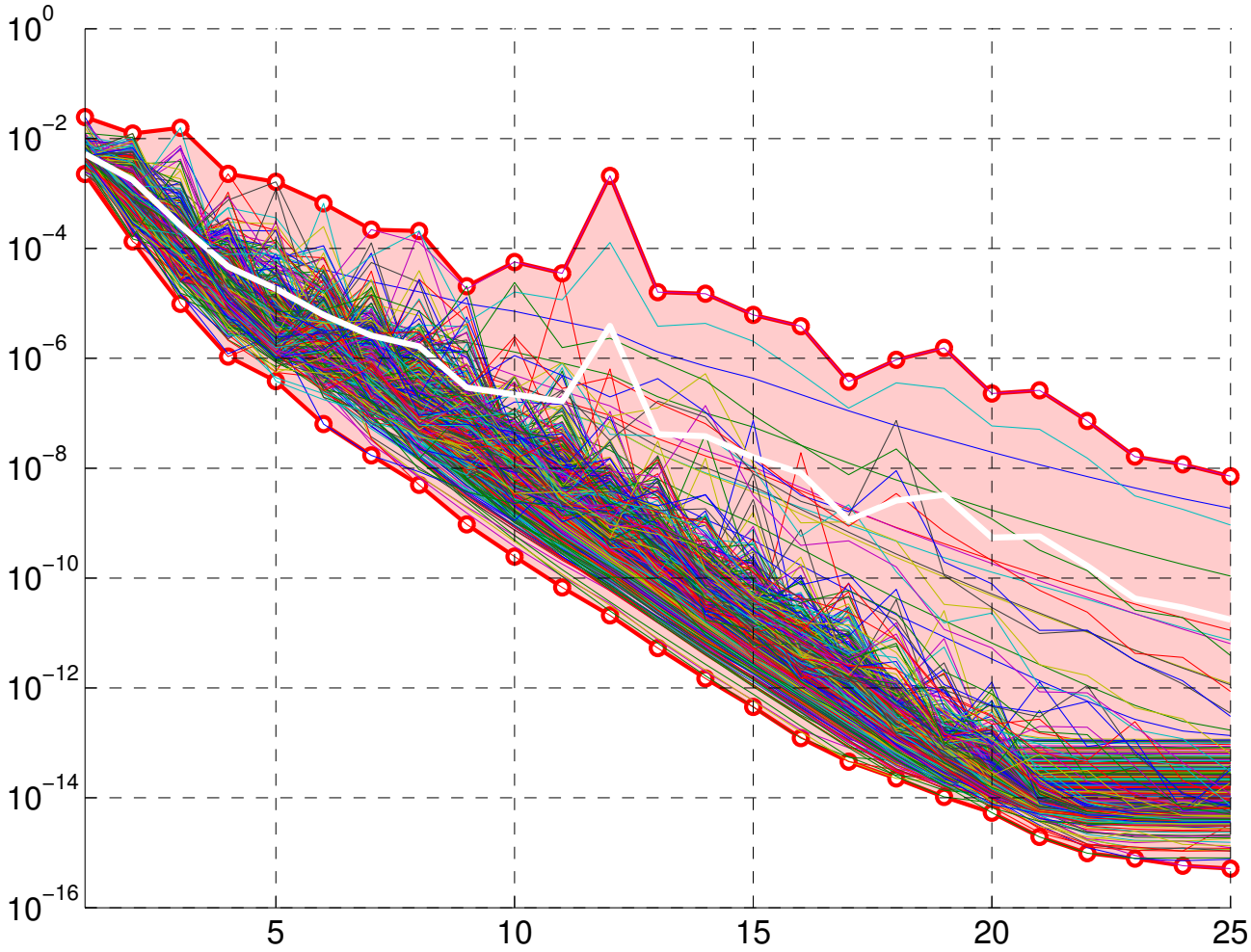
- Eigenpair residuals: $10^{-9} \ldots 10^{-16}$
- Linear system residuals: $10^{-10} \ldots 10^{-16}$

Residuals: history



Range of residuals vs. iteration number for Graphene III

Residuals: history



Individual residuals vs. iteration number for Graphene III.
White line = average.

Results: orthogonality

- measure orth $:= \max_{i \neq j} |x_i^T x_j|$
- Problem: orthogonality across intervals
- Better if eigenvalues are separated
- For Graphene II (gap): orth $= 6 \cdot 10^{-11}$
- Individual intervals: orth $= 10^{-13} \ldots 10^{-15}$ (basically ensured by design of algorithm).

Conclusion

- Reliable eigenvalue counting and
- efficient computation of a reasonable search space is possible

Conclusion

- Reliable eigenvalue counting and
- efficient computation of a reasonable search space is possible
- FEAST applied to eigenvalue problems from graphene yields accurate results.
- For certain problems, orthogonality issues are not very serious

Thank you for listening!

References:

- Krämer, L., Di Napoli, E., Galgon, M., Lang, B. and Bientinesi, P., Dissecting the FEAST algorithm for generalized eigenproblems, J. Comput. Appl. Math. **244**, pp. 1–9, 2013.
- Galgon, M., and Krämer, L., and Lang, B., Counting eigenvalues and improving the integration in the FEAST algorithm, Preprint BUW-IMACM 12/22
- Di Napoli, E., Polizzi, E., and Saad, Y., Efficient estimation of eigenvalue counts in an interval, Preprint 2013.
- Tang, P., and Polizzi, E., Subspace Iteration with Approximate Spectral Projection, Preprint 2013.