A Scalable Parallel Eigensolver for Large-scale Simulations on Next-generation Computing Environments

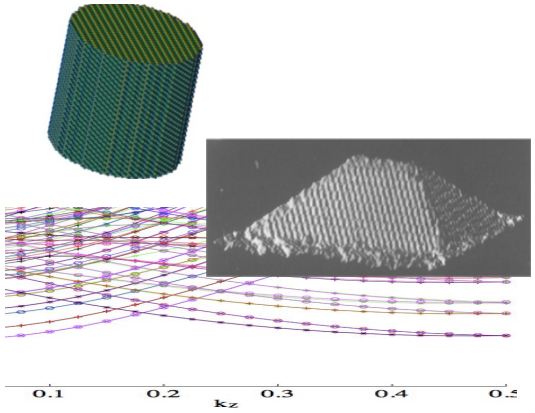
Tetsuya Sakurai (University of Tsukuba, JST CREST)

Joint work with
Akira Imakura, Yasunori Futamura, Xiucai Ye,
Yasuyuki Maeda, Takahiro Yano, Yuto Inoue

Contents

Large-Scale Eigenvalue Problems

- Partial Eigenvalue Problems in Computer Simulations
  - Find eigenvalues in a given interval
    e.g. around HOMO-LUMO band gap
  - Large-scale problems
  - Matrices are given explicitly / implicitly



- Highly Parallel Computing Environment
  - Large-scale supercomputers
  - Distributed computing nodes
    - Reduce communications between nodes

Partial Eigenvalue Problems

- We consider an eigenvalue problem

$$T(\lambda)\boldsymbol{x} = \mathbf{0},$$

where $T(\lambda)$ is a matrix valued function, and $\boldsymbol{x}$ is a nonzero vector of dimension n.

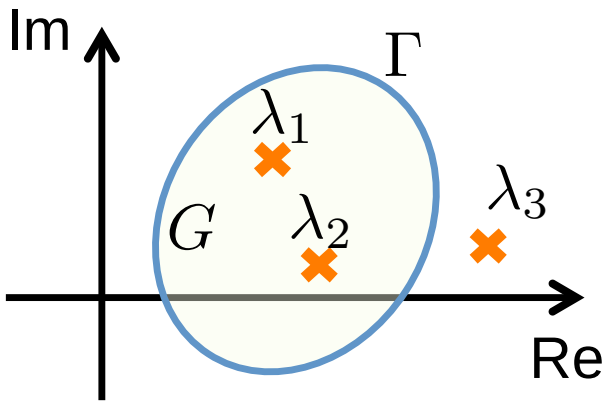
Ex. Standard EP (SEP): $T(\lambda) = A - \lambda I$

Generalized EP (GEP): $T(\lambda) = A - \lambda B$

Nonlinear EP (NEP): $T(\lambda) = A_0 + \lambda A_1 + \lambda^2 A_2$

$T(\lambda) = A_0 + \lambda A_1 + e^{\lambda} A_2$

- Find all eigenvalues located inside Γ
  - Interior eigenvalue problems
  - Large-scale sparse matrices

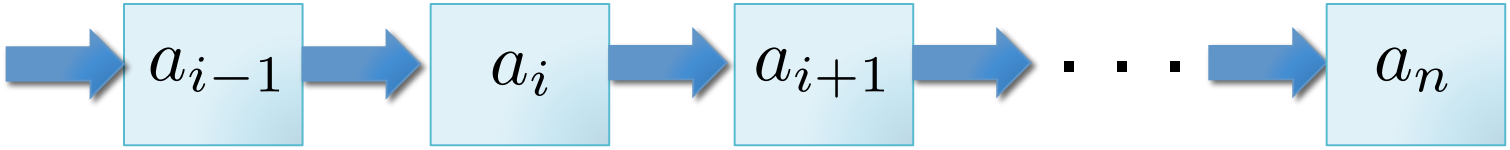


In this talk, we mainly explain GEP: $A\boldsymbol{x} = \lambda B\boldsymbol{x}$

Our Approach for Parallel Scalability

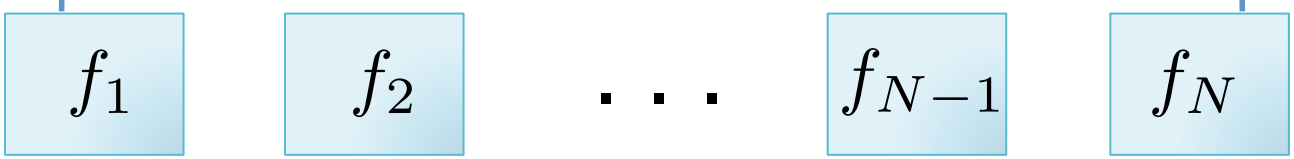
- Avoid recurrence calculations in eigenvalue computation
  - Algorithms described by recurrence relations:
    (ex. Krylov methods):

$$a_{i+1} = b_i\, a_i + c_i\, a_{i-1}$$



  - Algorithms without recurrence calculations:
    (ex. Numerical quadrature):

$$I_N = \sum_{j=1}^{N} w_j f_j$$



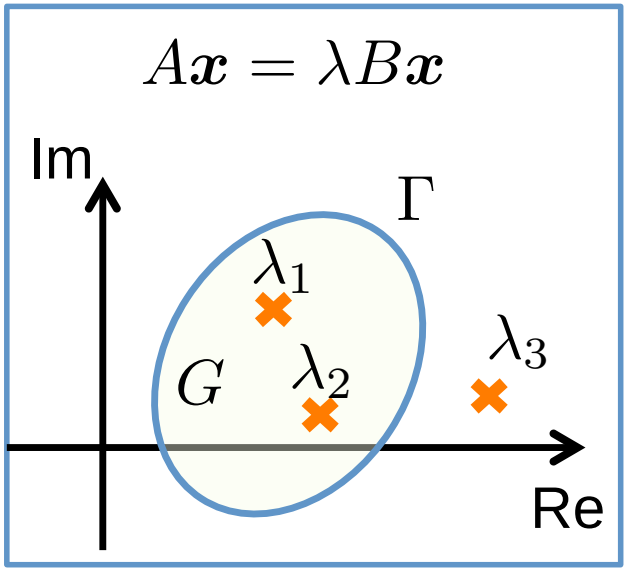
Quadrature-type Eigensolver

Basis of Our Approach

- Spectral decomposition of $(zB - A)^{-1}B$ is given by

$$(zB - A)^{-1}B = \sum_{i=1}^{n} \frac{P_i}{z - \lambda_i},$$

where λ_i is an eigenvalue and P_i is a projection with respect to λ_i.



(for simplicity, we consider the case that λ_i is simple)

By using the residue theorem, we obtain the following relation:

$$P_\Gamma = \frac{1}{2\pi \mathrm{i}} \oint_\Gamma (zB - A)^{-1} B \mathrm{d}z = \sum_{\lambda_i \in G} P_i$$

Approximation by *N*-point Numerical Quadrature

- Contour integral is approximated by numerical quadrature

$$P_\Gamma = \sum_{\lambda_i \in G} P_i = \frac{1}{2\pi \mathrm{i}} \oint_\Gamma (zB - A)^{-1} B \mathrm{d}z$$

$$\approx \sum_{j=1}^{N} w_j (z_j B - A)^{-1} B$$

z_j : quadrature point
w_j : quadrature weight

- Apply for vectors $V = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_L]$:

$$P_\Gamma V \approx \sum_{j=1}^{N} w_j (z_j B - A)^{-1} BV$$

Systems of linear equations at shift points $z_1, \ldots, z_N$

$$(z_j B - A) Y_j = BV, \ j = 1, \ldots, N$$

Extension of Subspace with Higher Order Complex Moments

- Obtain various linear combinations of projections using complex moments:

$$P_\Gamma = \sum_{\lambda_i \in G} P_i = \frac{1}{2\pi \mathrm{i}} \oint_\Gamma (zB - A)^{-1} B \mathrm{d}z$$

$$P_\Gamma^{(k)} = \sum_{\lambda_i \in G} \lambda_i^k P_i = \frac{1}{2\pi \mathrm{i}} \oint_\Gamma z^k (zB - A)^{-1} B \mathrm{d}z, \ k = 0, 1, \ldots$$

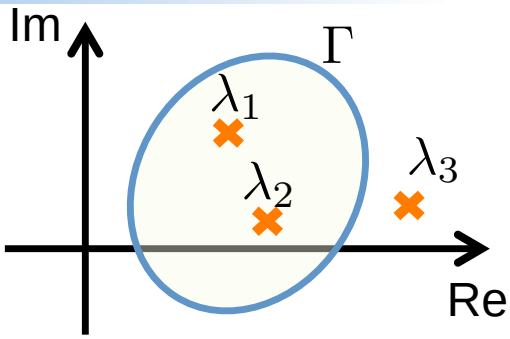
- Apply for vectors $V = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_L]$:

$$P_\Gamma^{(k)} V \approx S_k = \sum_{j=1}^{N} w_j z_j^k (z_j B - A)^{-1} BV, \ k = 0, 1, \ldots, M-1$$

- Eigenpairs are extracted from $S = [S_0, S_1, \ldots, S_{M-1}]$ ($L \times M$)
  - Hankel type, Rayleigh-Ritz type, etc.

Quadrature-type Methods

- Using Numerical Quadrature
  - GEPs (Generalized Eigenvalue Problems)
    - SS-H [S/Sugiura'2003]
      - Grid RPC(remote procedure call) (2004)
        - Coarse-grained communication, Hierarchical structure
    - SS-RR [S/Tadano'2007]
    - Block SS [Ikegami/S/Nagashima'2008]
    - FEAST [Polizzi'2009]
    - SS-Arnoldi [Imakura/Du/S'2014]
  - NEPs (Nonlinear Eigenvalue Problems)
    - SS-H [Asakura/S et al.'2009]
    - Beyn's method [Beyn'2012]
    - SS-RR [Yokota/S'2013]
    - SS-Arnoldi [Imakura/Du/S'2015]



SS method with Rayleigh-Ritz procedure

- Algorithm of SS-RR

Appropriate path and parameters

0: Set contour path and parameters

1: Solve linear systems

Most time consuming part

$$\text{Solve } (z_j B - A)Y_j = BV \text{ for } Y_j,\ j = 1, 2, \ldots, N$$

2: Construct a subspace

$$\text{Compute } \hat{S}_k = \sum_{j=1}^{N} w_j z_j^k Y_j,\ k = 0, 1, \ldots, M-1$$

$$\text{Set } \hat{S} = [\hat{S}_0, \hat{S}_1, \ldots, \hat{S}_{M-1}]$$

3: Extract eigenpairs

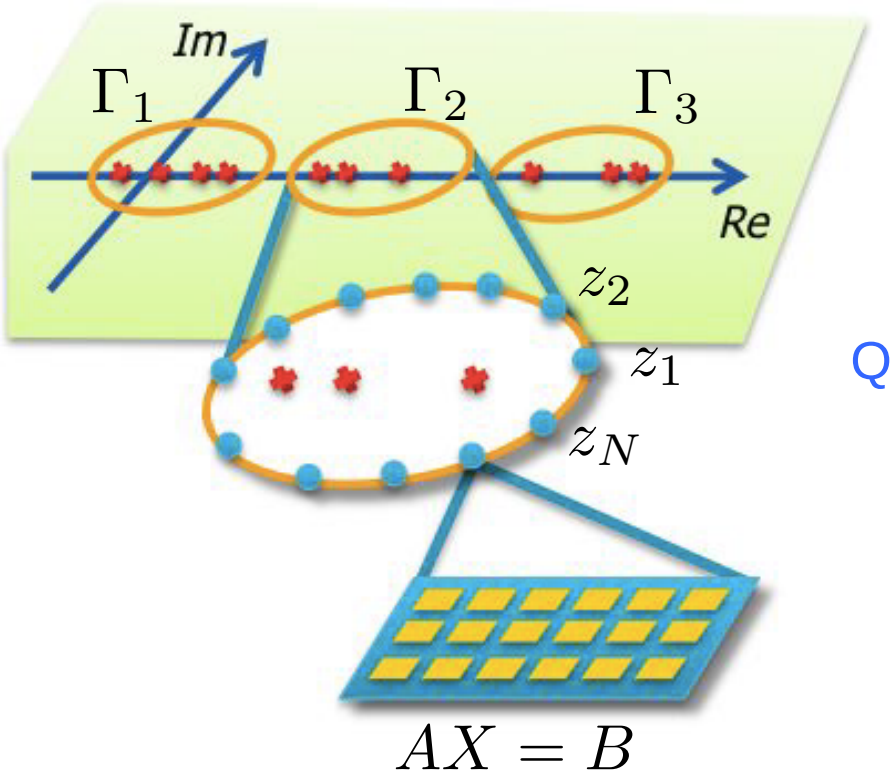
$$\text{Compute low-rank approx. s.t. } \hat{S} = U\Sigma W^T \approx U_1 \Sigma_1 W_1^T$$

$$\text{Compute Ritz paris with } \hat{A} = U_1^H A U_1 \text{ and } \hat{B} = U_1^H B U_1$$

Hierarchical Parallel Structure

- Computing resources are assigned according to a hierarchical structure of the algorithm

Hierarchical structure of the algorithm

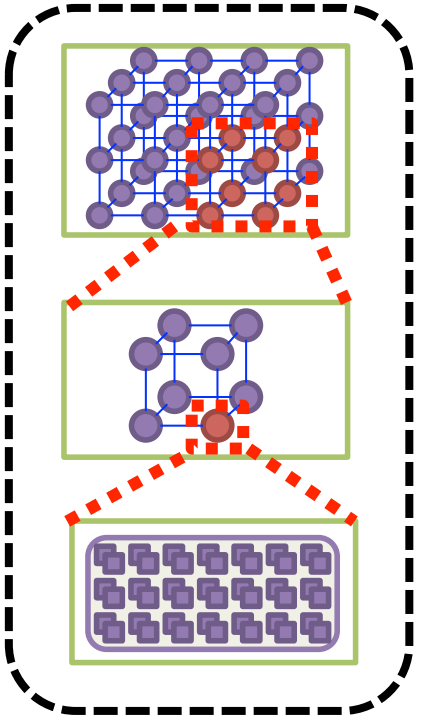


Contour paths

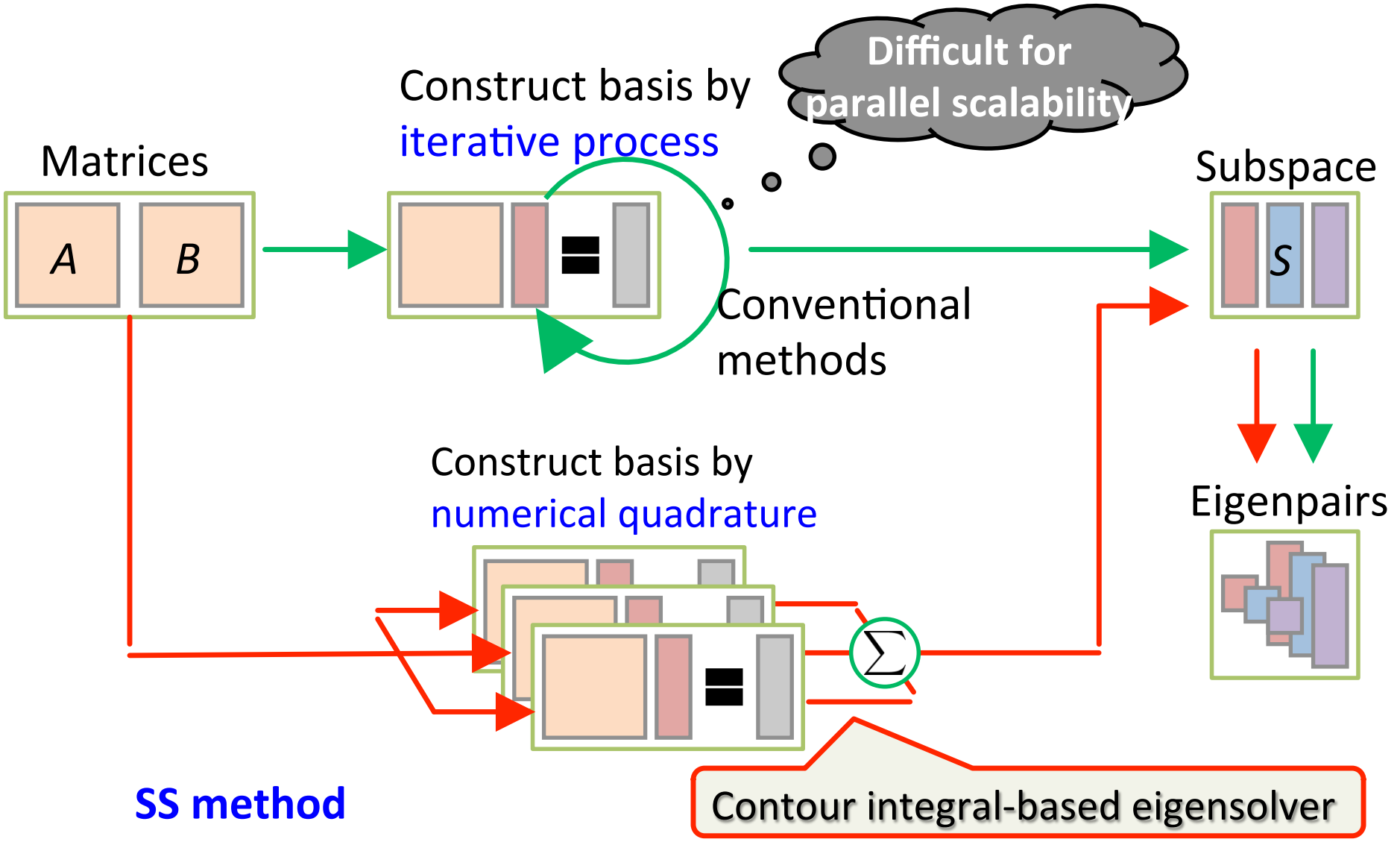
Quadrature points

Linear solvers

Hierarchical structure of a machine



Our Approach for Parallel Scalability



Software

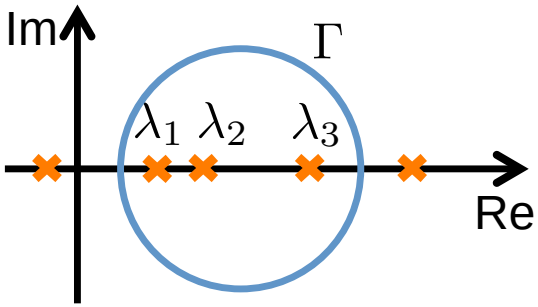
- Published software
  - z-Pares
    - Fortran95, MPI
    - For large-scale distributed parallel computing
  - CISS
    - SLEPc / PETSc
    - For evaluating efficiency of the algorithm in distributed parallel computing
  - SSEIG
    - MATLAB
    - For evaluating efficiency of the algorithm
- Available:
  - **http://zpares.cs.tsukuba.ac.jp/**

Stochastic Estimation of Eigenvalue Distribution

Eigenvalue Count in a Given Domain

- The number of eigenvalues m in Γ is given by

$$m = \frac{1}{2\pi \mathrm{i}} \oint_{\Gamma} \mathrm{tr}((zB - A)^{-1} B) \mathrm{d}z$$



  - Approximate contour integral of a trace of inverse matrix by[1), 2)]

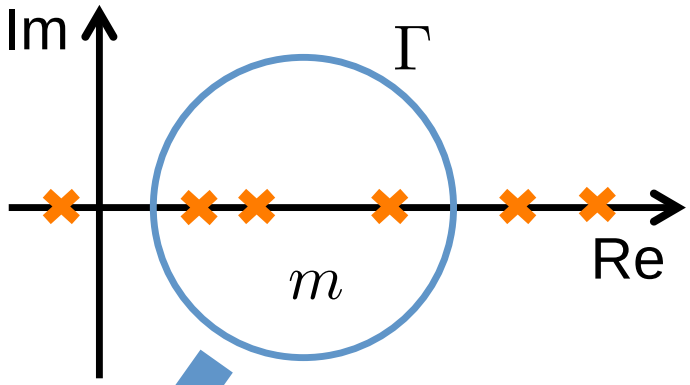
$$m \approx \sum_{j=1}^{N} w_j \left(\frac{1}{L} \sum_{\ell=1}^{L} \boldsymbol{v}_\ell^{\mathrm{T}} (z_j B - A)^{-1} B \boldsymbol{v}_\ell \right)$$

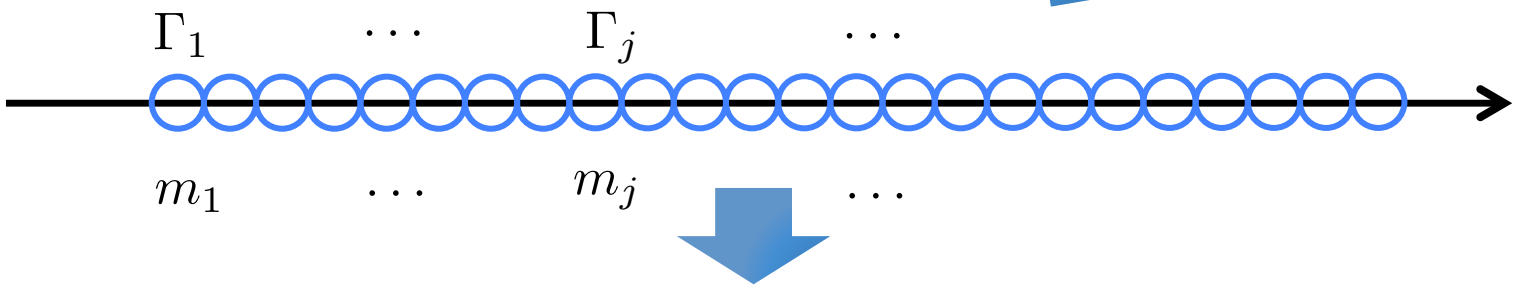
where $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_L$ are *L* sample vectors.

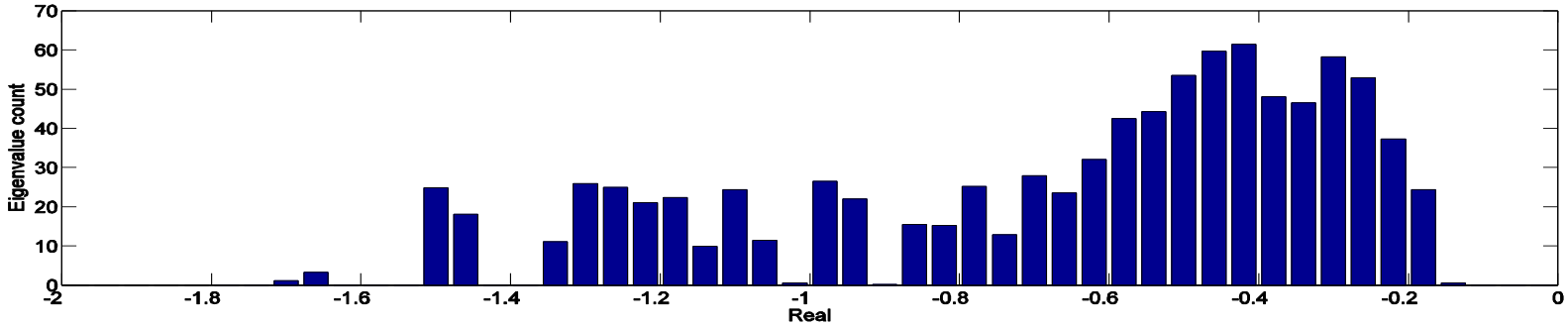
1) Futamura, H. Tadano, and T. Sakurai, JSIAM Letters 2, 127-130 (2010).
2) Y. Maeda, Y. Futamura, A. Imakura and T. Sakurai, JSIAM Letters 7, 53-56 (2015).

Estimation of Eigenvalue Density

- Estimate the eigenvalue density
  - Put sub-intervals on a target interval







Numerical Examples

K-Computer (AICS)

- The K Computer (AICS, RIKEN)
  - Number of nodes : 88,128 (705,024 cores)
  - Performance : 10.51 PFLOPS (Peak 11.28 PFLOPS)
  - CPU: SPARC64 VIIIfx 8 cores 2.0GHz (128GFLOPS)
  - Memory: 16GB/node (total memory: 1.41 PB)
  - Memory Bandwidth: 64GB/s

Supercomputers in Tsukuba

- HA-PACS System @Univ. Tsukuba
  - CPU: Intel E5-2670 2.56GHz x2
  - GPU: NVIDIA M2090 x4
  - Number of nodes : 268
  - Peak performance : 802 TFLOPS
    (CPU: 89 TFLOPS, GPU: 713 TFLOPS)

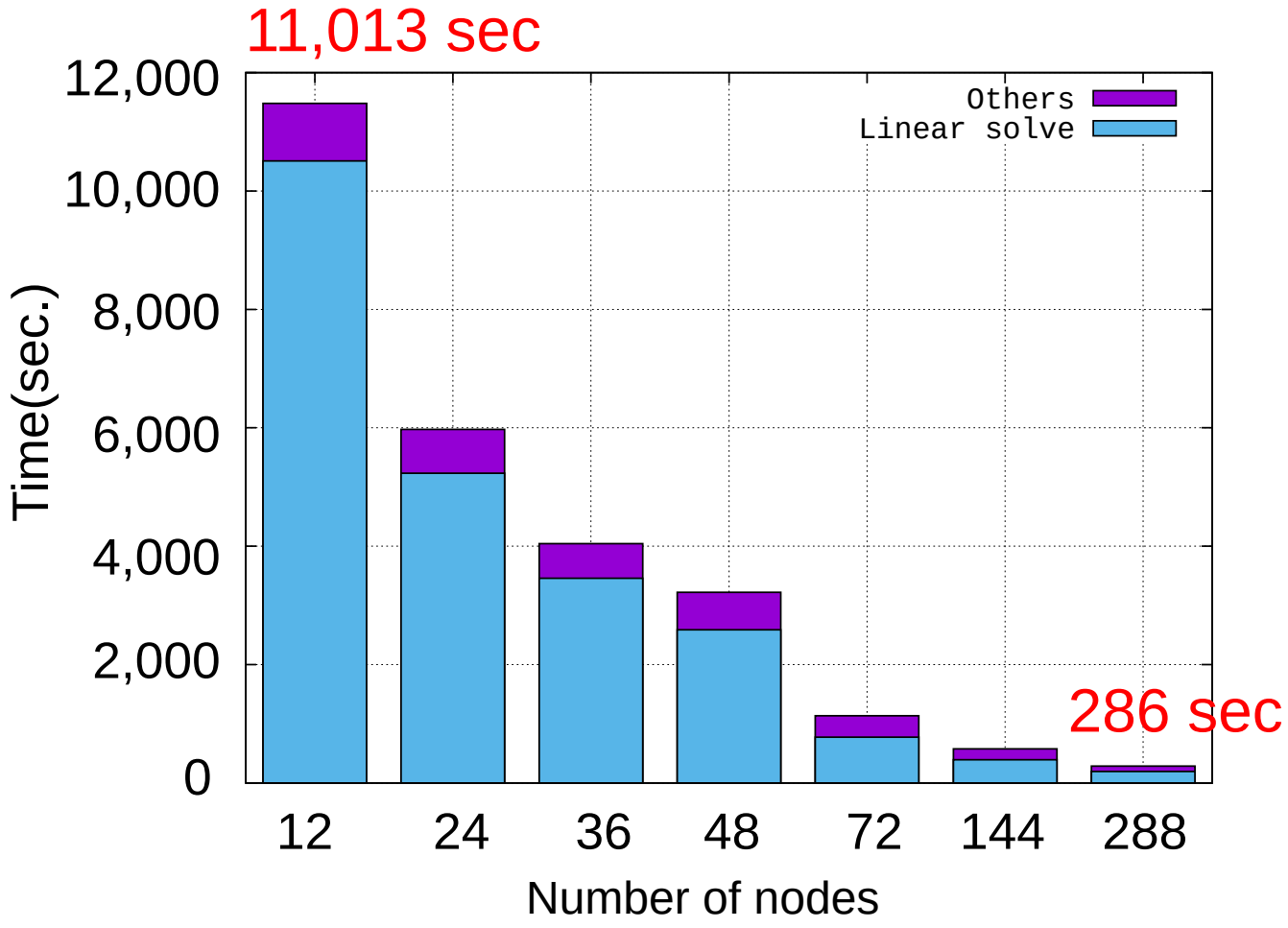
- COMA (PACS-IX) System @Univ. Tsukuba
  - CPU: Intel E5-2670v2 2.5GHz x2
  - MIC: Intel Xeon Phi 7110P 61core x2
  - Number of nodes : 393
  - Peak performance : 1.001 PFLOPS
    (CPU: 157 TFLOPS, MIC: 844 TFLOPS)

Numerical Example: Transmission Design

- Application: Automatic transmission design
  - Matrices are derived by FEM
  - Frequency range: 0 – 6,000 Hz
    - 5,435,438 Nodes
    - 16,747,926 DOF
    - 916 eigenpairs are calculated
  - Test environment: COMA @Univ. of Tsukuba
  - Solvers
    - Eigensolver: z-Pares (SS method)
      - #contour paths: 4
      - #quadrature points: $N = 16$
    - Linear solver: MUMPS (sparse direct solver)

Numerical Example: Transmission Design

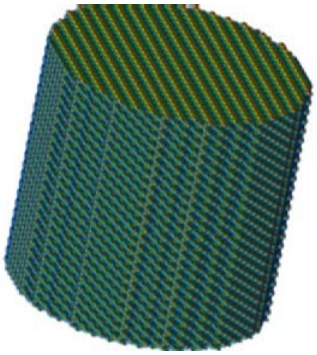
- Matrix dimension: 16,747,926, 916 eigenpairs in [0, 6000] Hz
- #contour paths: 4, #quadrature points: N = 16



Numerical Example: Density Functional Theory

- Band structure calculation with real space density functional theory (RSDFT by Iwata et al.)
  - An interior standard eigenvalue problem (SEP)
    - Eigenvalues around the band gap
    - Matrices with several wave numbers:

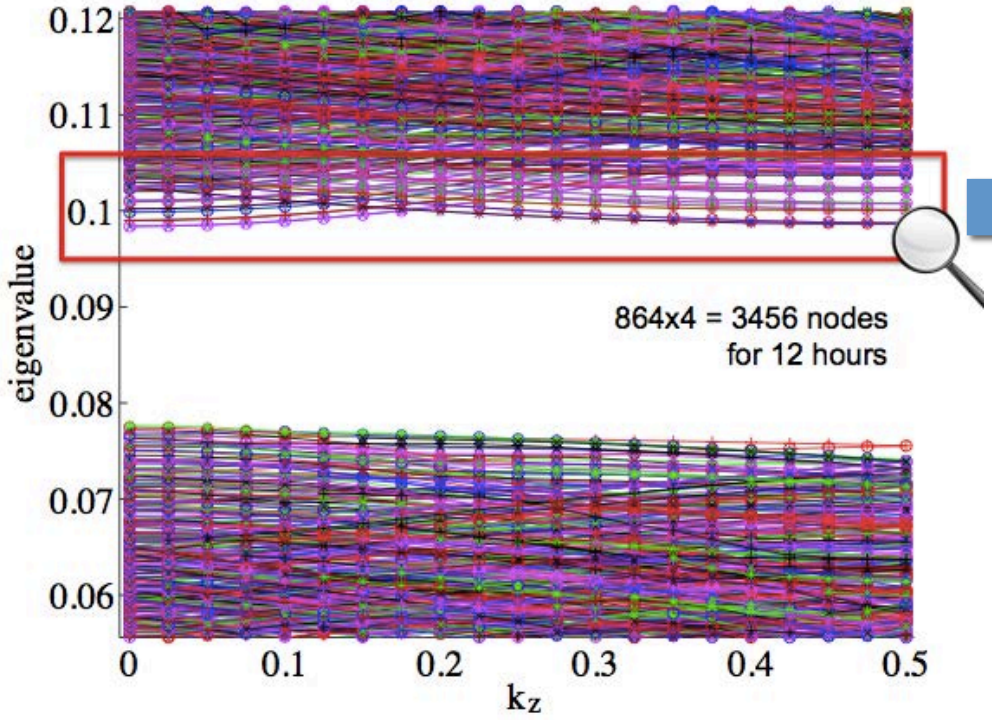
$$A(k)\boldsymbol{u} = \lambda \boldsymbol{u},\ \ k = 0, \Delta k, 2\Delta k, \ldots$$

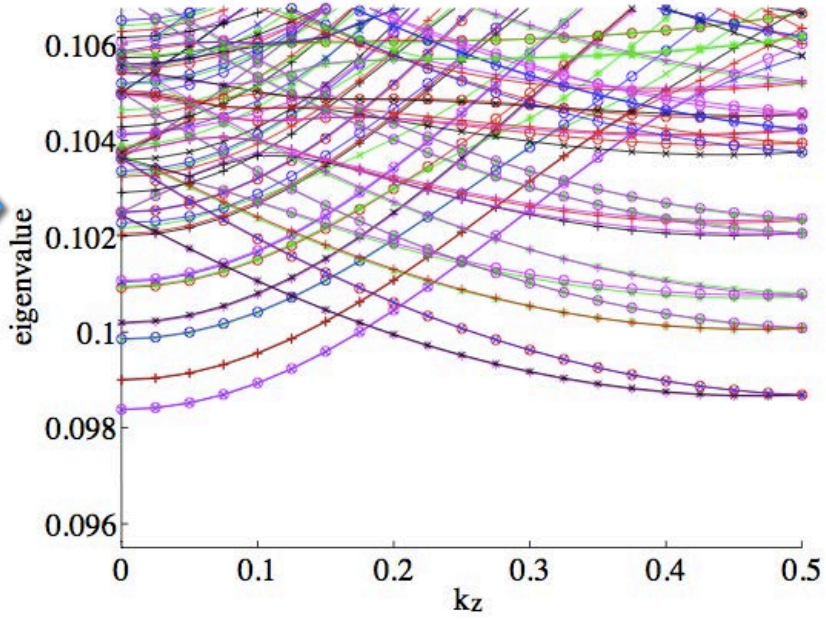


- Material: Silicon nanowire 9,924 atoms
  Matrix dim.: 8,719,488 (only mat-vec operation is given)
- Test environment: the K Computer (AICS, Japan)
  - 3,456 nodes (27,648 cores)
- Linear solver: Shifted Block CGrQ

Numerical Example : Density Functional Theory

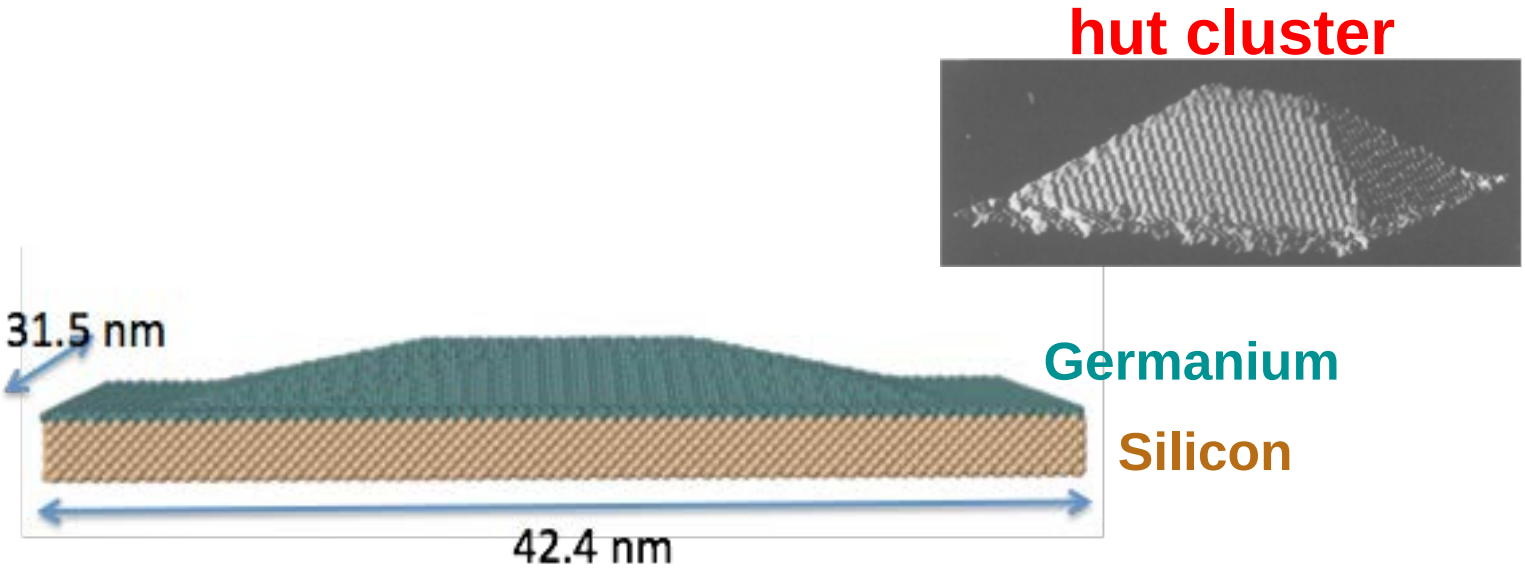
- Band structure calculation for Silicon Nanowire 9,924 atoms
  - 21 problems are solved
  - Four intervals are set around the band gap
  - 3,456 nodes of the K Computer (about 12 hours)



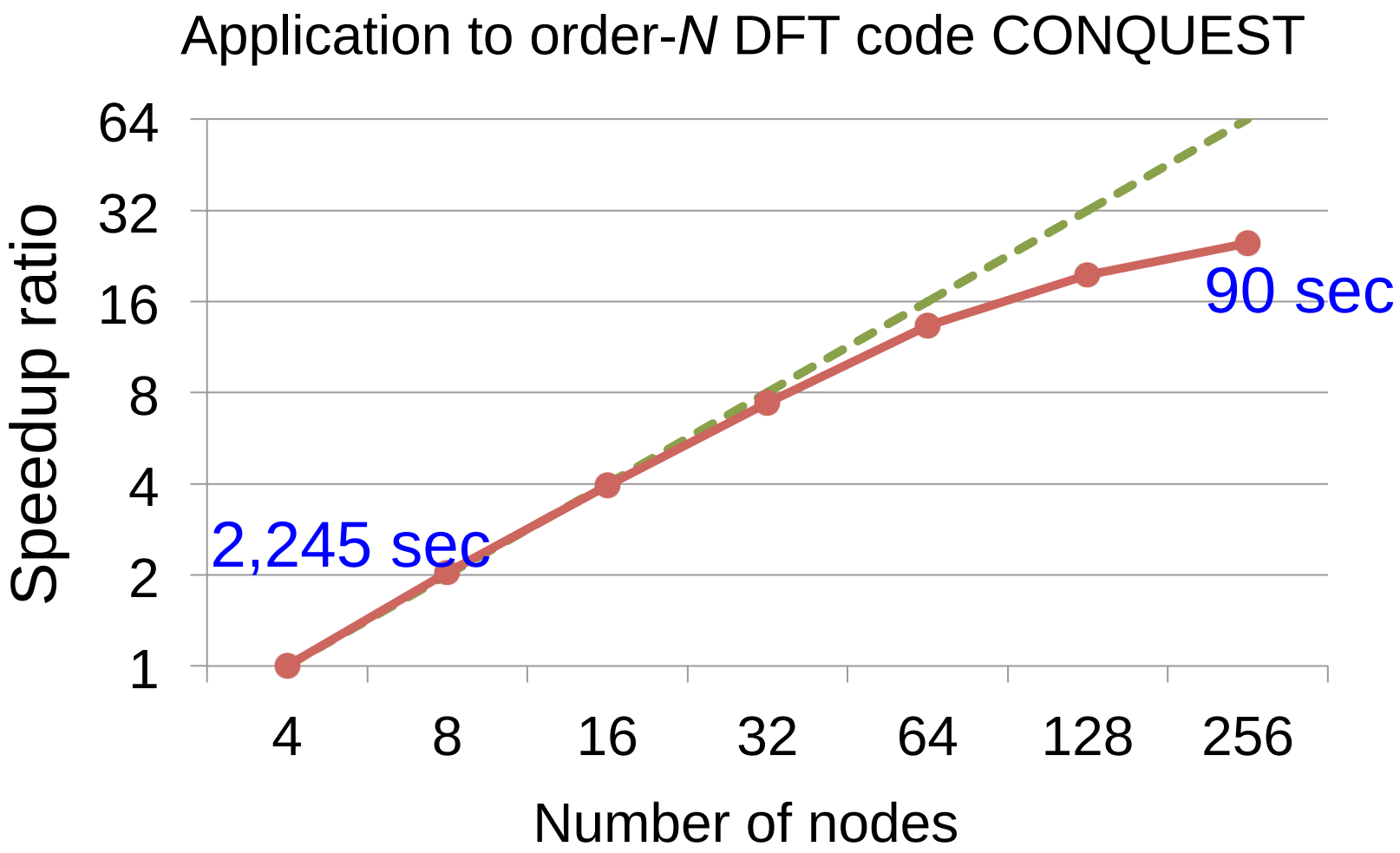


Numerical Example: Order-*N* DFT

- Order-*N* DFT code CONQUEST
  - SiGe hut cluster with 200,000 atoms
    by Nakata and Miyazaki
    - Matrix size : 778,292, NNZ : 13,247,248
    - GEP: 223 eigenpairs around HOMO-LUMO are computed
  - Test environment : COMA@Univ. of Tsukuba
  - Linear solver : MUMPS (sparse direct solver)

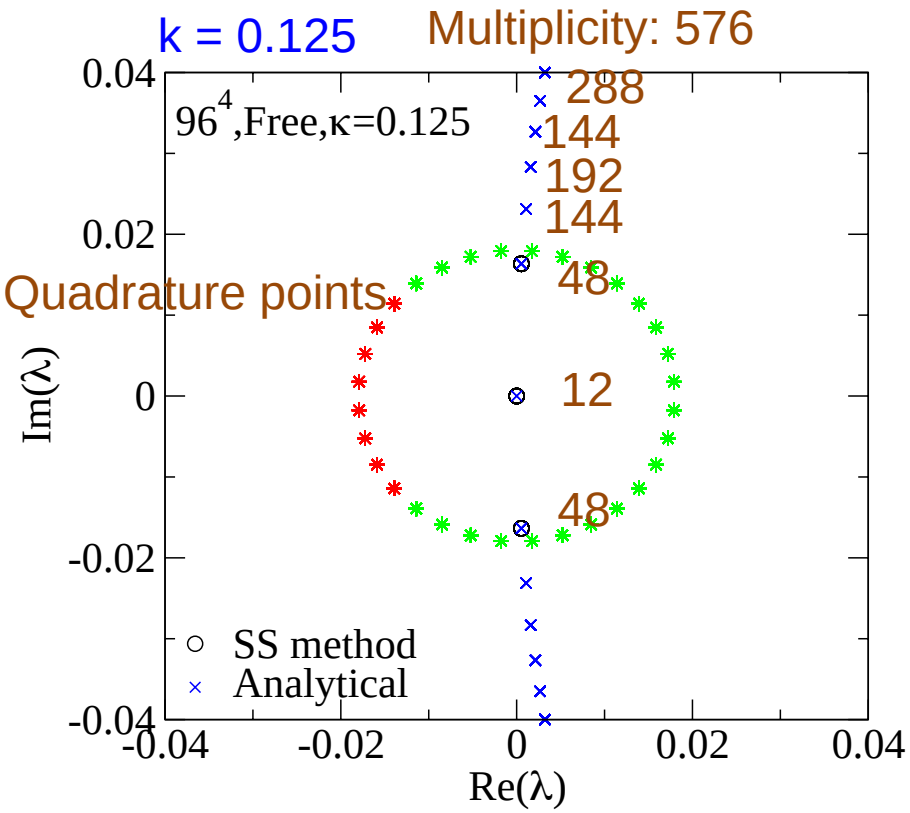


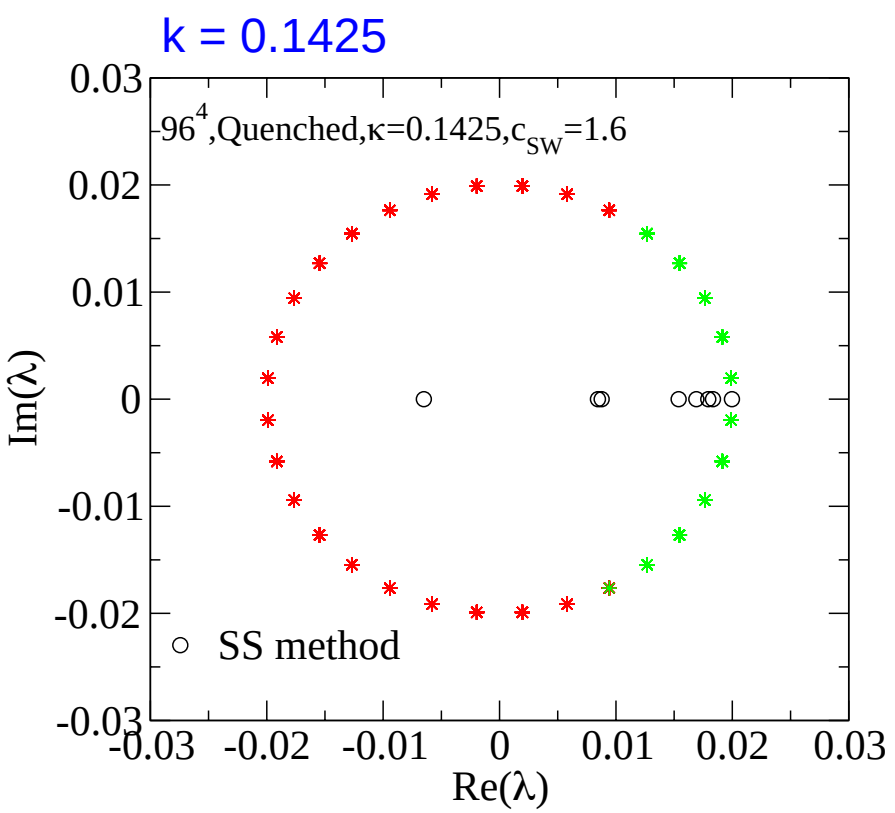
Numerical Example: Order-*N* DFT



Numerical Example: Lattice QCD

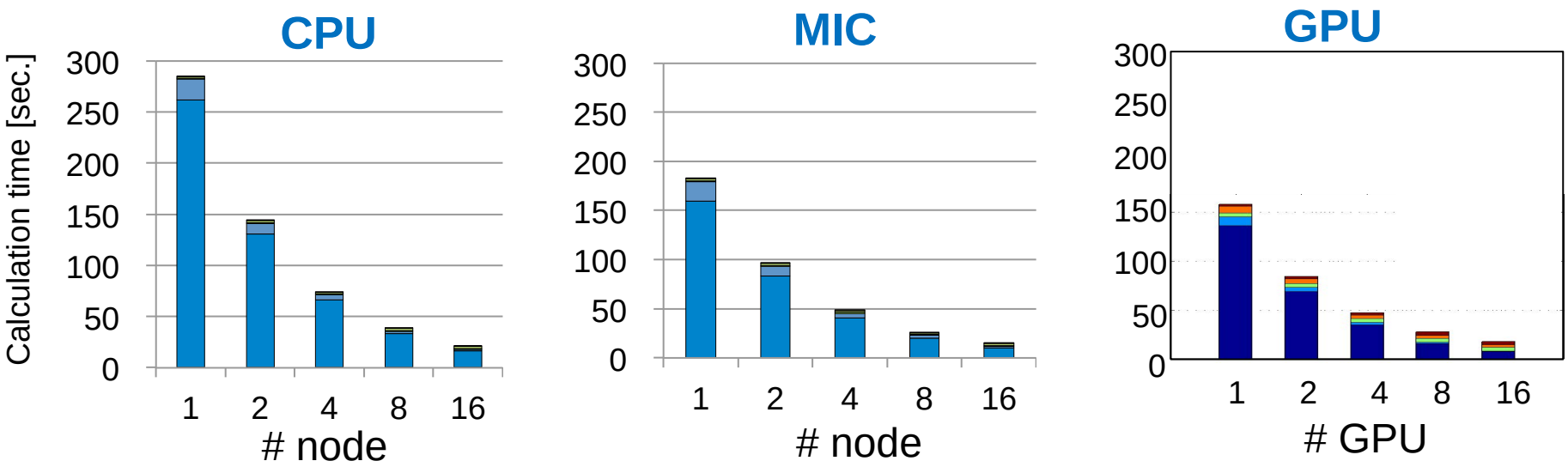
- O(a)-improved Wilson-Dirac operator (Suno, Kuramashi, S, et al.)
  Matrix dim.: 1,019,215,872 (non-Hermitian)
  Test environment: 16,384 nodes of the K-Computer
  Linear solver: BiCGStab





Numerical Example: MIC/GPU

- Linear solver is replaced for MIC/GPU
  - Test matrix : GEP, real symmetric, dense (n=10,000)
    - 5% of eigenpairs are calculated
  - CPU: COMA@Univ. Tsukuba
  - MIC: COMA@Univ. Tsukuba
  - GPU: HA-PACS@Univ. Tsukuba



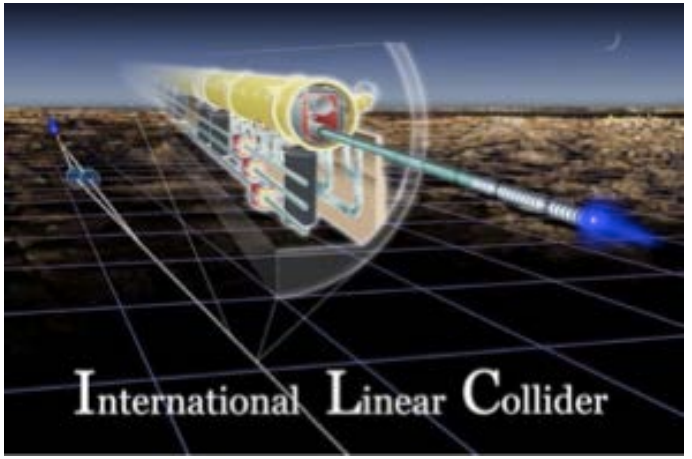
Numerical Example: Nonlinear Eigenvalue Problem

- Test problem:
  Simulation of the international linear collider

$$T(\lambda)\boldsymbol{x} = \mathbf{0}$$

$$T(\lambda) = K - \lambda^2 M + \mathrm{i}\sum_{j=1}^{t}\sqrt{\lambda^2 - \sigma_j^2}W_j,$$

where $t = 1$, $\sigma_1 = 0$.



http://www.linearcollider.org/

- Test environment:
  Cray-XT4 at NERSC @Berkeley
- Linear solver: SuperLU_DIST

Numerical Example: Nonlinear Eigenvalue Problem

- Strong scalability for nonlinear (NEP) case
  - Matrix size: 2,738,556

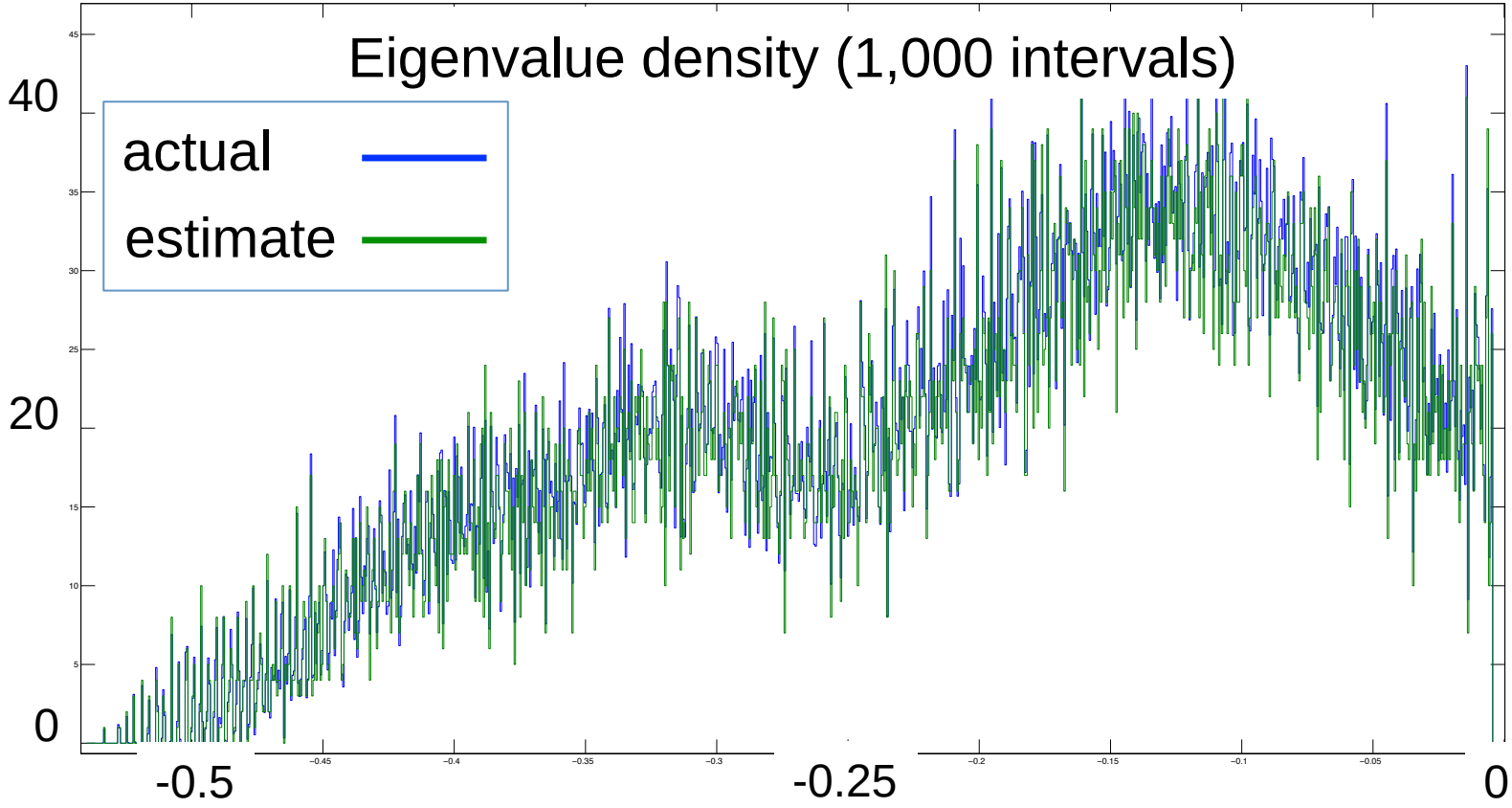
- Two contour paths are located.
  - The number of quadrature points is $N = 32$.
  - 64 linear systems are solved in total.

#cores	256	512	1024	2048
time(sec.)	2513	1273	661	334
speedup	-	1.97	3.80	7.30

[Yamazaki et al. '2013]

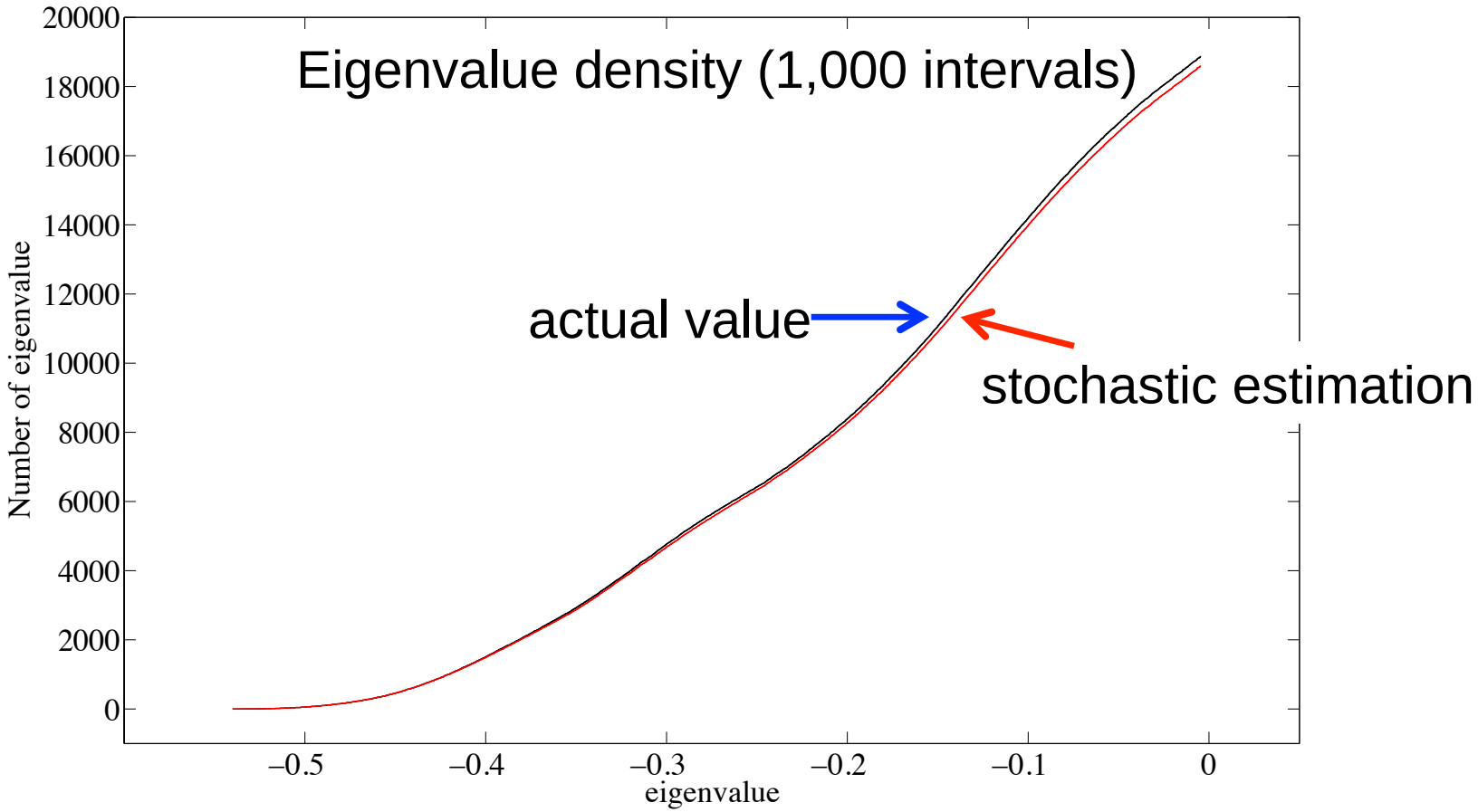
Numerical Example: Eigenvalue Density

- RSDFT SiNW 9,924 atoms
  Eigenvalue density at initial status
  - #quadrature points: $N = 8$
  - Linear solver: Shifted Block CGrQ



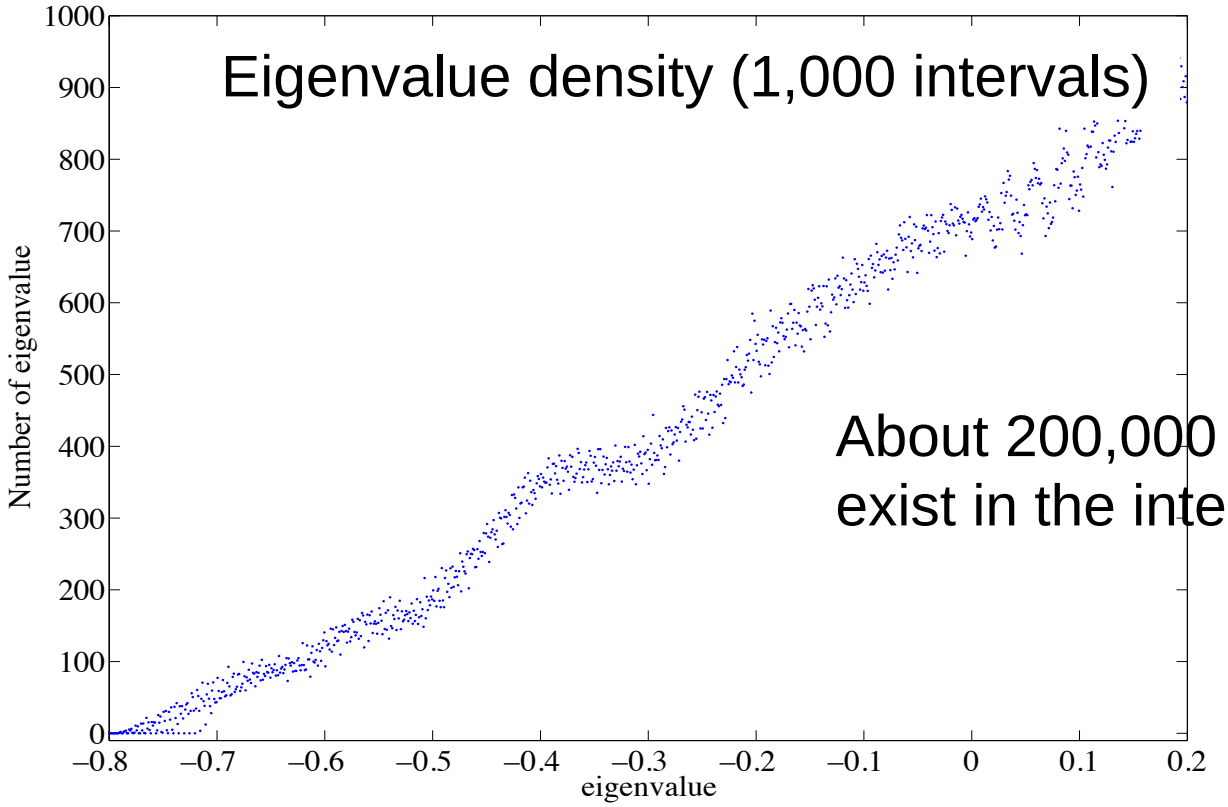
Numerical Example: Eigenvalue Density

- RSDFT SiNW 9,924 atoms
  Accumulated total from left



Numerical Example: Eigenvalue Density

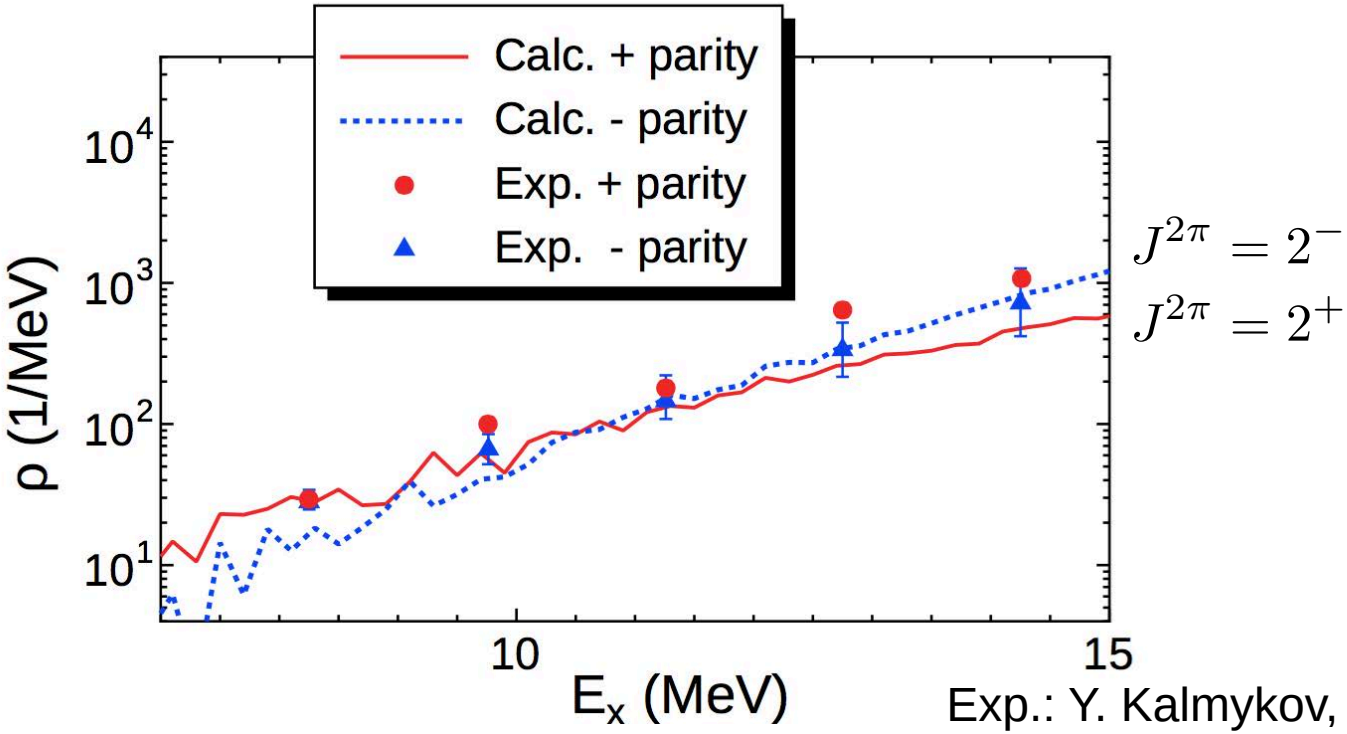
- RSDFT SiNW 107,292 atoms
  Eigenvalue density at initial status
  - Matrix dimension: 64,700,000
  - 10,800 nodes of the K-Computer, 11,890 sec



About 200,000 eigenvalues exist in the interval

Numerical Example: Shell Model Nuclear Level Density

- Shell Model Code: Kshell (Shimizu, et al.)
  - Spin-dependent level density of ^{58}Ni*
    - Matrix dim.: 15 billion
    - 2,304 nodes of the K-Computer, 24 hours



Exp.: Y. Kalmykov, et al. (2007)

*Shimizu, et al., Phys. Lett. B (accepted)

Conclusions

- Scalable Algorithm: SS method
  - Numerical quadrature with higher order moments
- Software
  - z-Pares in Fortran95, CISS in SLEPc, sseig in MATLAB
- Stochastic estimation of eigenvalue density
  - Uses numerical quadrature
- Future work
  - Efficient linear solvers and preconditioners
  - Linear algebra kernels such as matrix-vector product, orthogonalization, SVD, etc.
  - Apply for various applications

References

[1] T. Sakurai, H. Sugiura, A projection method for generalized eigenvalue problems using numerical integration, J. Comput. Appl. Math. 159, 119-128, 2003.

[2] T. Sakurai, H. Tadano, CIRR: a Rayleigh-Ritz type method with contour integral for generalized eigenvalue problems, Hokkaido Math. J. 36, 745-757, 2007.

[3] I. Ikegami, T. Sakurai, U. Nagashima, A filter diagonalization for generalized eigenvalue problems based on the Sakurai-Sugiura projection method,CS-TR-08-13, Tsukuba, 2008.

[4] J. Asakura, T. Sakurai, H. Tadano, T. Ikegami, K. Kimura, A numerical method for nonlinear eigenvalue problems using contour integrals, JSIAM Lett. 1, 52-55, 2009.

[5] T. Sakurai, J. Asakura, H. Tadano, T. Ikegami, Error analysis for a matrix pencil of Hankel matrices with perturbed complex moments, JSIAM Lett. 1, 76-79, 2009.

[6] Y. Futamura, H. Tadano, and T. Sakurai, Parallel stochastic estimation method of eigenvalue distribution, JSIAM Letters 2, 127-130, 2010.

[7] Y. Futamura, T. Sakurai, S. Furuya and J. Iwata, Efficient algorithm for linear systems arising in solutions of eigenproblems and its application to electronic-structure calculations, LNCS 7851, 226-235, 2012.

[8] I. Yamazaki, T. Ikegami, H. Tadano, T. Sakurai, Performance comparison of parallel eigensolvers based on a contour integral method and a Lanczos method, Parallel Computing 39, 280–290, 2013.

[9] T. Yano, Y. Futamura, T. Sakurai, Multi-GPU scalable implementation of a contour-integral-based eigensolver for real symmetric dense generalized eigenvalue problems, Proc. 8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2013), pp. 121-127 (2013).

[10] A. Imakura, L. Du, T. Sakurai, A block Arnoldi-type contour integral spectral projection method for solving generalized eigenvalue problems, Appl. Math. Lett., 32, 22-27, 2014.

[11] A. Imakura, L. Du and T. Sakurai, Error bounds of Rayleigh--Ritz type contour integral-based eigensolver for solving generalized eigenvalue problems, Numerical Algorithms, (accepted).

[12] Y. Maeda, Y. Futamura, A. Imakura and T. Sakurai, Filter analysis for the stochastic estimation of eigenvalue counts, JSIAM Letters 7, 53-56 (2015).