Improvements to ELPA Eigensolvers

ELPA-AEO

http://elpa-aeo.mpcdf.mpg.de

GEFÖRDERT VOM

Bundesministerium
für Bildung
und Forschung

BMBF Projekt 01IH15001
Feb 2016 - Jan 2019

ELPA-AEO

Project Partners

MPCDF Max Planck Computing & Data Facility

BERGISCHE UNIVERSITÄT WUPPERTAL

Lehrstuhl für Angewandte Informatik
Prof. B. Lang (BUW)

Fritz-Haber-Institut der Max-Planck-Gesellschaft

Prof. M. Scheffler, Dr. Ch. Carbogno (FHI)

MPCDF Max Planck Computing & Data Facility

Dr. H. Lederer, Dr. A. Marek (MPCDF)

TUM

Lehrstuhl für Informatik mit Schwerpunkt Wissenschaftliches Rechnen (TUM-SCCS)
Prof. H.-J. Bungartz, Prof. Th. Huckle

Lehrstuhl für Theoretische Chemie
Prof. K. Reuter, Dr. Ch. Scheurer (TUM-CH)

Stepwise Approach for the Eigenproblem

Solution of the generalized matrix eigenproblem generally proceeds in five steps:

(I) Transformation to a dense standard eigenproblem (e.g., by Cholesky decomposition of S),
$H_{KS}c_l = \varepsilon_l S c_l \;\rightarrow\; Aq_A = \lambda q_A, \quad \lambda = \varepsilon_l$

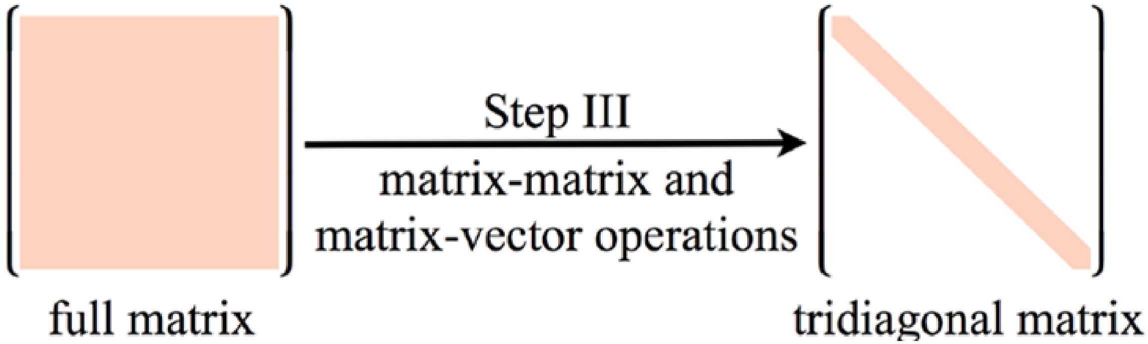
(II) Reduction to tridiagonal form, A -> T;

(III) Solution of the tridiagonal problem for k eigenvalues and -vectors, $Tq_T = \lambda q_T$;

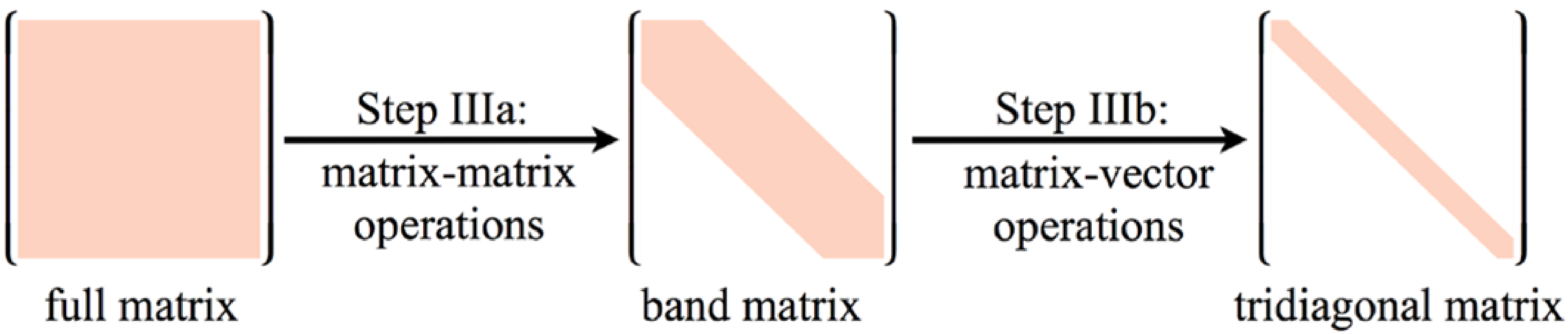
(IV) Back transformation of k eigenvectors to dense orthonormal form, $q_T \rightarrow q_A$;

(V) Back transformation to the original, non-orthonormal basis, $q_A \rightarrow c_l$.

ELPA1: one-step direct solver



ELPA2: two-step direct solver



Pros: *allows full use of matrix–matrix products and sparse matrix-vector products*
Cons*: gives rise to one extra back transformation step of the eigenvectors*

ELPA Performance Gain in 2017

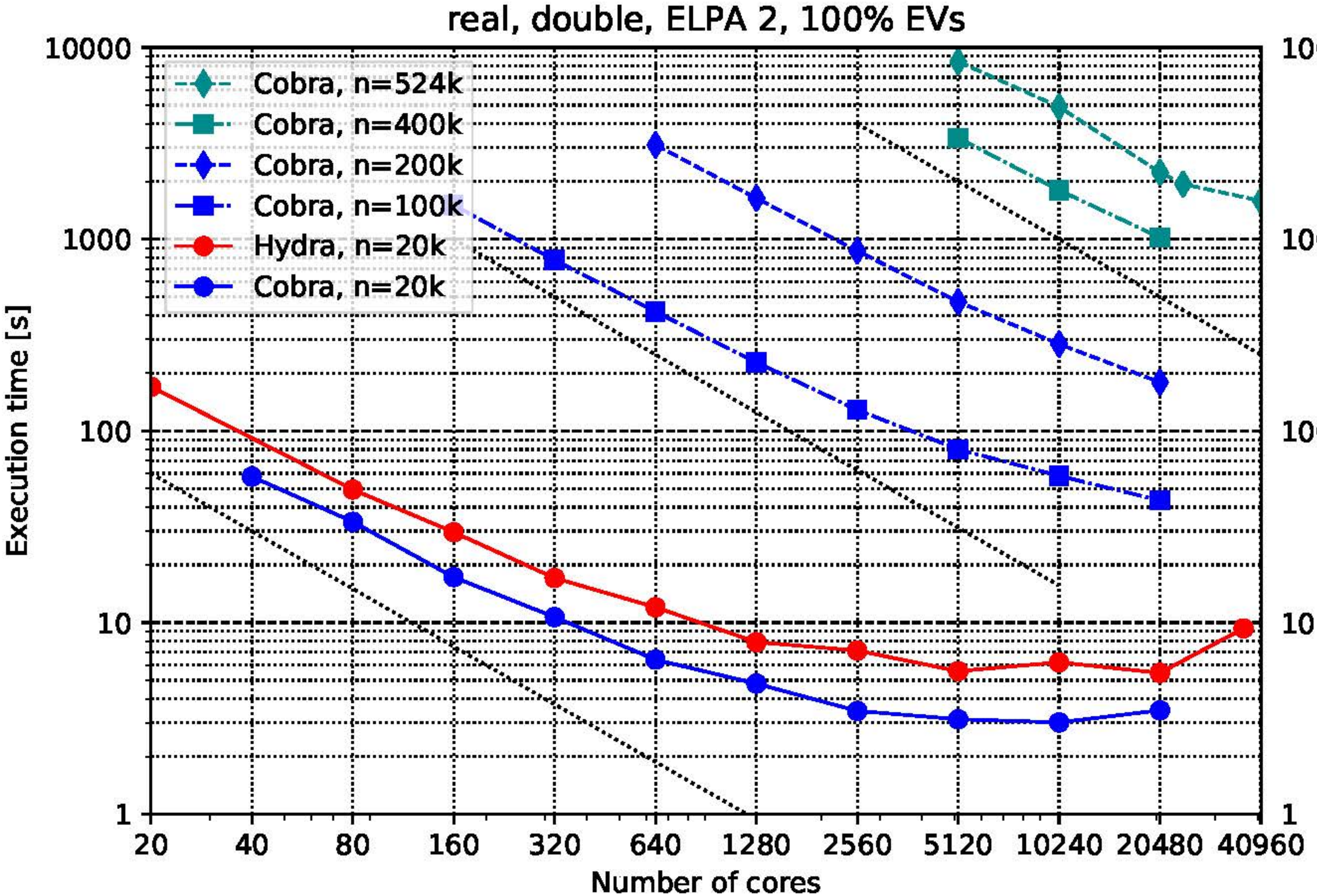
Algorithmic improvements / code optimizations

System: Hydra (node: 2 x Intel Xeon E5-2680v2, 20 cores)
No of cores used: 80
Matrix size: 20 000, real
Solver: ELPA2, 100% Eigenvectors

ELPA 2013 -> ELPA 2017:
Runtime (double precision) on 80 cores: ~ **75 s -> 50 s**
Improvement factor ~ 1.5

Option for mixing precision (both single and double precision supported):
ELPA 2017
Run time (double precision) **50 s -> 30 s (** single precision)
Improvement factor ~ 1.7

ELPA on Intel Xeon Gold 6148



Monitoring

Monitoring tool for autotuning fully implemented

Detailed timing measurements (both exclusive and inclusive) realized for all major routines (elpa1 solver, elpa2 solver, elpa2 optimized kernels, transformation of generalized to standard problem incl. its back transformation

Monitoring for reporting to calling program

Calling program can

- enable/disable timing measurements
- query timing results, number of calls of a subroutine and performance in GigaFlop/s
- print timing results to stdout
- for autotuning

Improvements by Automatic ELPA Autotuning

For a given problem hardware configuration:

Automatic selection of the most efficient procedure for the solution of the eigenproblem

For a given problem (matrix size and number of eigenvalues needed):

Comparing the different available procedures (elpa1 or elpa2) in consecutive SCF cycles, identification and automatic selection of the most efficient one for all further steps.

For elpa2 (more complex than elpa1) :

Optimization of the selection of the block size and the different compute-intenisve kernels.
In case of availability of GPUs, inclusion into selection criteria.

Autotuning

Based on monitoring infrastructure

Main program options: enable/disable (default: disable)

In case of enable:

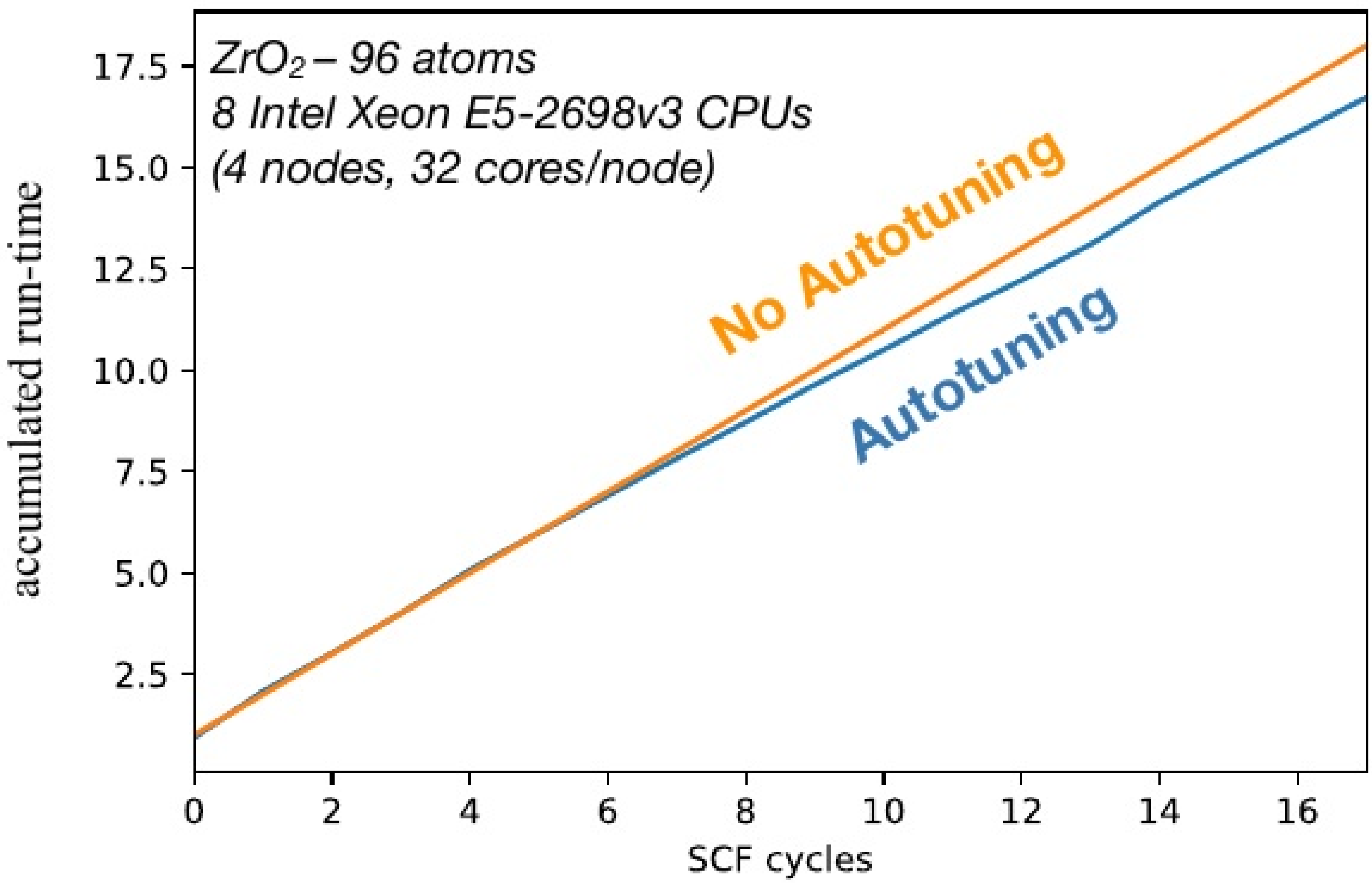
When elpa is called multiple times within an SCF iteration, different elpa solvers (elpa 1 and elpa 2), different tuning options for elpa 2 (kernel) and further run time parameters are tried consecutively and the timings recorded. After measuring all available options, the best performing one is used for all subsequent calls .

User guided autotuning:

User can preselect a subset of settings (e.g. elpa2), then the other options are not considered.

User results with autotuning

Accumulated Runtimes:



Improvements to the generalized eigenvalue problem $AX = BX\mu$

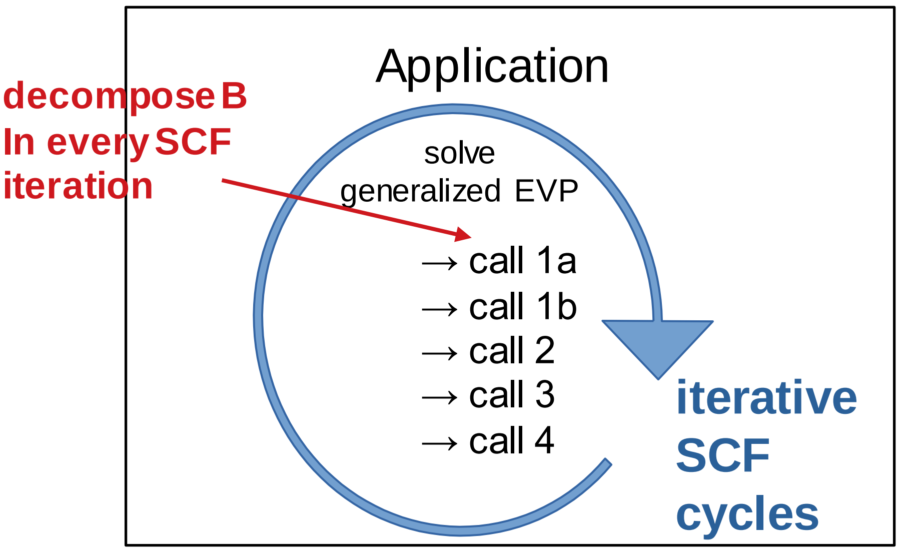
Mathematics	What ELPA provides
1) $B = U^H U$ (Cholesky decomposition)	1a) $B = U^H U$ (Cholesky decomposition) 1b) Explizit construction of inverse U^{-1}
2) $A \rightarrow A^* = U^{-H} A U^{-1}$ (transformation to standard eigenproblem)	2) identical
3) solve standard eigenvalue problem	3) identical
4) Backtransformation of eigenvectors according to 2	4) identical New: **elpa_generalized** combining steps 1a – 4 in single routine

Results

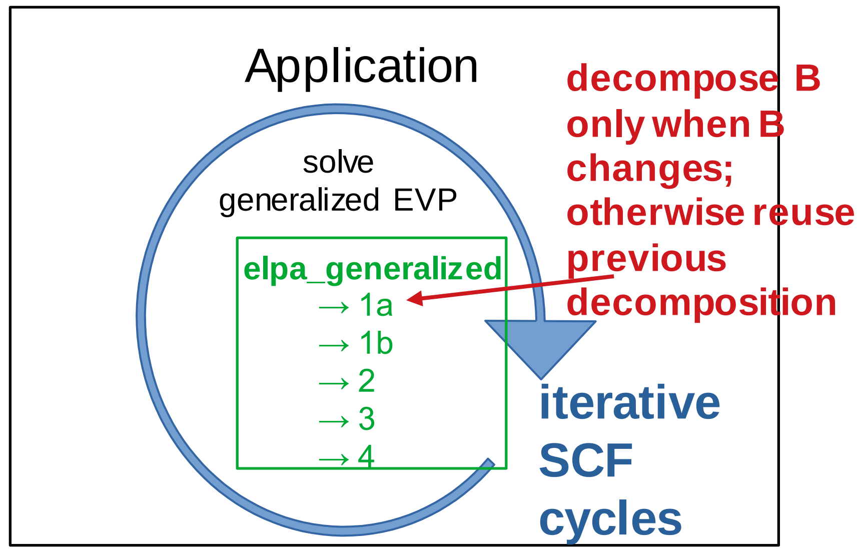
Improvements to the generalized eigenvalue problem

$$AX = BX\mu$$

Up to last ELPA release in 2017:



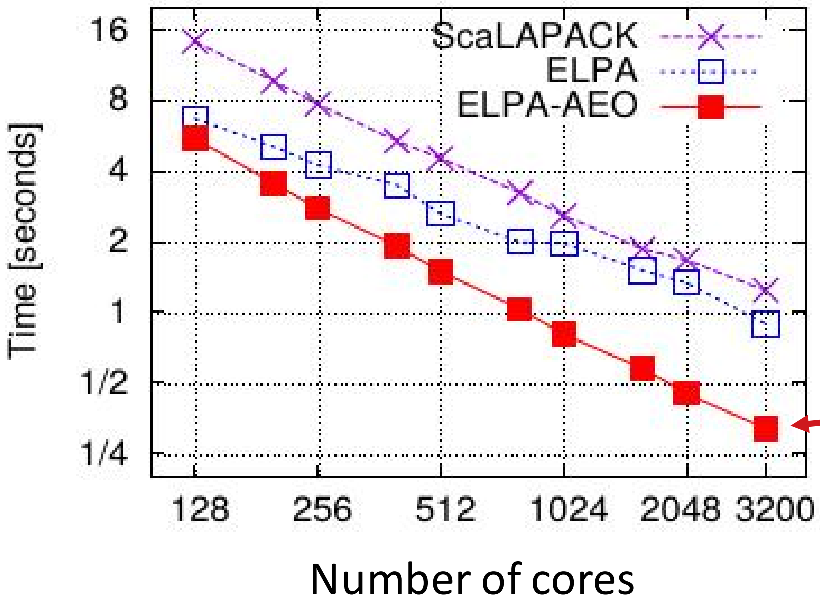
Latest ELPA release:



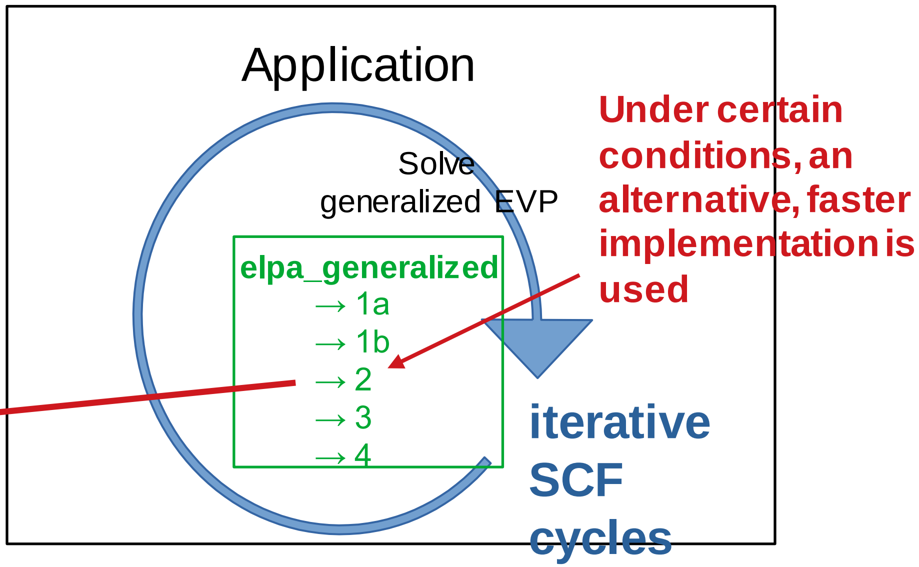
Results

Improvements to the generalized eigenvalue problem

$$AX = BX\mu$$



Latest ELPA release:



Supported Architectures

- **X86 up to Intel SkyLake / AVX512**
  measured up to matrix size 0.5 M up to 40k cores
- **X86 / Intel Xeon KNL**
  measured up to matrix size 1 M on 200k cores
- **IBM Power 8+9**
- **GPU systems**

a) nVIDIA K20X/K40 in X86 host with via PCIe
b) nVIDIA P100 with Power8 hosts via NVLINK1
c) GTX1080 with X86 host via PCIe

- **K Computer** (RIKEN)
- Porting started:
  NEC SX-Aurora vector processors
  OpenPower: NVIDIA V100 GPUs in Power 9 hosts
  NVIDIA V100 GPUs in Intel Xeon CPU hosts

(also running on ARM and on MAC OS systems)

Presentations

P. Kus:
GPU Optimization of Large-Scale Eigenvalue Solver, ENUMATH 2017

B. Lang:
The ESSEX-II and ELPA-AEO Projects. EPASA2018
Int'l Workshop on Eigenvalue Problems: Algorithms; Software and Applications, in Petascale Computing, March 5-6, 2 018, Tsukuba

B. Lang, V. Manin:
Reduction of generalized HPD eigenproblems using Cannon's algorithm.
SIAM Conference on Parallel Processing for Scientic Computing, March 7-10, 2018, Tokyo

B. Lang, V. Manin: Efficient reduction of generalized HPD eigenproblems. PMAA18 { 10th Int'l Workshop on Parallel Matrix Algorithms and Applications, 27.-29. Juni 2018, Zürich

PMAA18: Minisymposium on eigenvalue problems and applications, organised by Th. Huckle (TUM), B. Lang (BUW), and T. Imamura (RIKEN)
Presentations from: BUW, TUM-SCSC, FHI, MPCDF

Conference: Solving or Circumventing Eigenvalue Problems in Electronic Structure Theory, August 15-17, 2018, Richmond, VA
B. Lang (BUW): New algorithmic developments in ELPA-AEO
C. Carbogno (FHI): Recent Advancements in ELPA: Best Practices in Real Applications

Publications

B. Lang and V. Manin:
Cannon-type triangular matrix multiplication for the reduction of generalized HPD eigenproblems to standard form, Parallel Computing 2018

B. Lang:
Efficient reduction of banded hpd generalized eigenvalue problems to standard form

A. Alvermann, A. Basermann, H.-J. Bungartz, Ch. Carbogno, D. Ernst, H. Fehske, Y. Futamura, M. Galgon, G. Hager, S. Huber, Th. Huckle, A. Ida, A. Imakura, S. Köcher, M. Kreutzer, P. Kus, B. Lang, H. Lederer, V. Manin, A. Marek, K. Nakajima, L. Nemec, K. Reuter, M. Rippl, M. Röhrig-Zöllner, T. Sakurai, M. Scheer, Ch. Scheurer, F. Shahzad, D. Simoes Brambila, J. Thies, G. Wellein:
Benefits from using mixed precision computations in the ELPA-AEO and ESSEX-II eigensolver projects

B. Lang
New algorithmic developments in ELPA-AEO
Solving or Circumventing Eigenvalue Problems in Electronic Structure Theory, August 15-17, 2018, Richmond, VA

New ELPA Releases

ELPA-Release from May 2018

- Extended autotuning of performance critical parameters
  (as cache blocking, workload distribution between CPU and GPU
- Driver routines for generalized problems
- Option to skip decomposition of B matrix of generalized eigenproblem

ELPA-Release planned for Nov 2018

- Automatic checkpoint/restart for autotuning
  -> Option to start new simulation with restart file from previous
  autotuning run
- Improved transformation from generalized to standard problem