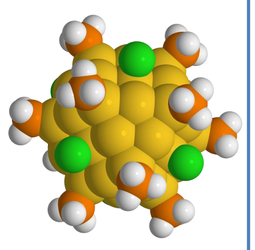
Efficient Parallel Strategies in Computational Modeling of Materials

Anca Berariu, Hugh Chaffey-Millar, Alexei Matveev, Astrid Nikodem, Martin Roderus,Thomas Soini, David Tittle, Arndt Bode, Hans-Joachim Bungartz, Michael Gerndt, Sven Krüger, Notker Rösch

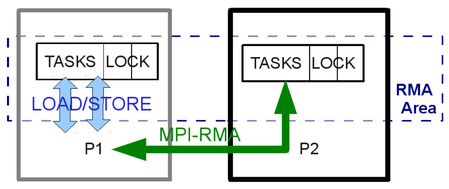
Quantum Chemistry with ParaGauss

- Typical problem: electronic energy and atomic forces of a molecule
- Electronic structure methods (DFT) do not admit a homogeneous parallelization strategy
- About 15 different parallelization algorithms
- About 99 % (real time) of overall task parallelized
- Main computational effort: self-consistent field procedure, forces and electron repulsion integrals
- Project goal – multilevel parallelization
- Modularization and module-specific parallelization
- Performance tuning for large CPU numbers and multicores
- Exact exchange and electron repulsion integrals
- Parallelization over independent electronic structure calculations



Dynamic Load Balancing (DLB)

- Work stealing algorithm re-distributes tasks only if necessary
- Detect global termination: (integer) credit distribution algorithm
- Multithreaded backend:
  - Thread safe MPI
  - POSIX threads
- Remote memory access backend (RMA):
  - Uses MPI RMA for task storage
  - Best with hardware support
  - Not all MPI RMA asynchronous requires source to be in MPI context
  - MPI-RMA lock allows only one MPI-operation on each storage element
  - No read-modify-write with MPI-RMA lock possible
  - Solution: user lock, but yields only try-read-modify-write
  - Access to specific user lock not guaranteed
  - Thus unsafe routines necessary to avoid deadlock (filing up tasks, knowing if own storage is empty)



Electron Repulsion Integrals (ERI)

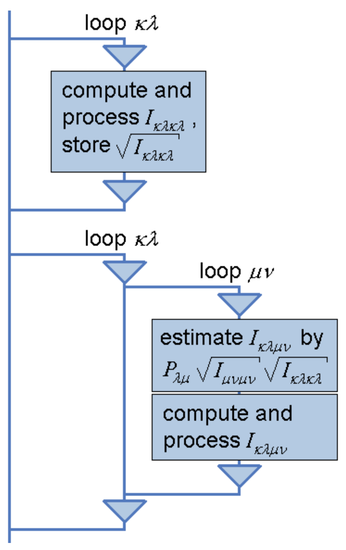
- Hybrid functionals: Promise higher accuracy in some cases but include an exact exchange term (like Hartree-Fock exchange)
- Construction of exact exchange matrix $\mathbf{K}$ requires $\mathcal{O}(N^4)$ four center ERIs over Gaussian basis functions $\phi_\kappa(\vec{r})$

$$K_{\kappa\nu} = \sum_{\lambda\mu} K_{\lambda\mu} I_{\kappa\lambda\mu\nu} \qquad I_{\kappa\lambda\mu\nu} = \iint d\vec{r}_1^{\,3} d\vec{r}_2^{\,3}\ \phi_\kappa(\vec{r}_1)\ \phi_\kappa(\vec{r}_1) \frac{1}{|\vec{r}_1 - \vec{r}_2|} \phi_\mu(\vec{r}_2)\ \phi_\nu(\vec{r}_2)$$

  - Permutational index symmetry
  - Exploit common intermediates – batchwise calculation of ERIs
- Storage of all ERIs impossible: Direct SCF methods
  - ERI batches recalculated every SCF iteration and contributions directly added to $\mathbf{K}$.
  - Formal iterative $\mathcal{O}(N^4)$ scaling
- Integral screening techniques
  - Compute only batches with nonvanishing contribution to $\mathbf{K}$ Selection by Schwarz Inequality:

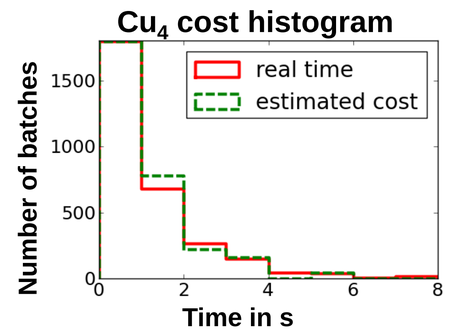
$$P_{\lambda\mu} I_{\kappa\lambda\mu\nu} \le P_{\lambda\mu} \sqrt{I_{\kappa\lambda\kappa\lambda}} \sqrt{I_{\mu\nu\mu\nu}}$$

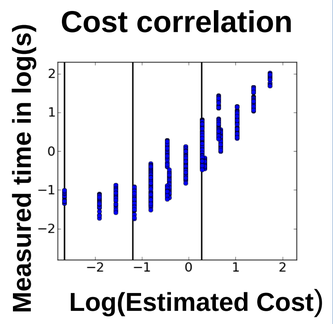
  - Working with incremental matrices i.e. $\Delta\mathbf{K}$ and $\Delta\mathbf{P}$ may reduce scaling

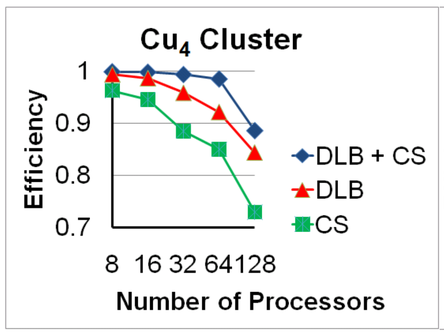


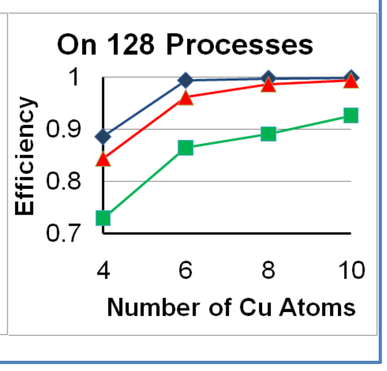
Application of DLB for the calculation of ERIs

- Only rough cost sorting for ERI batches
- Cost sorting (CS) for start distribution
- Wide batch cost spectrum
- Example: ERIs of Cu_4-Cu_{10}
- Many small and few large batches
- $\text{Efficiency} = \dfrac{\sum \text{work time}}{\text{timespan}}$
- DLB better than static CS
- DLB + CS even better due to balancing of smaller batches, reduces idle time at end
- Scheduling overhead: < than 2s / 1h
- Termination overhead: < than 4s / 1h
- Cu_4 too small for 128 processors (3003 batches / 1h), even DLB + CS only 89%
- Larger problems → higher efficiency
- Cu_6 14535 batches / 5h, with DLB + CS at 99.4%



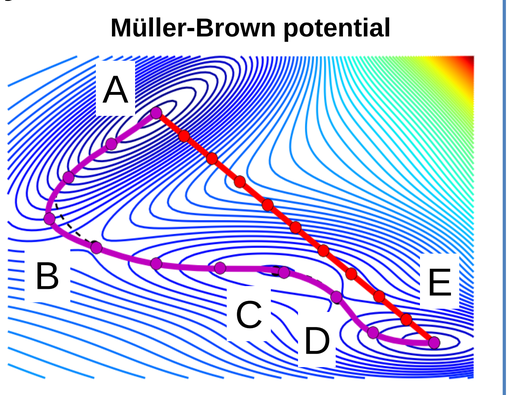


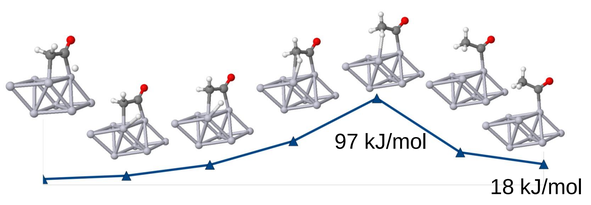




Python framework: exploring potential energy surfaces

- Various tools to explore potential energy surface (PES)
- Several single electronic structure calculations (ESC) per tool
- Vibrational frequencies via numerical differences of gradients
- Reaction path and transition states on PES
- Interfaces to several ESC programs
- Build-in parallelization of the ESC programs
- Parallelization over the different ESCs
- Each ESC should take about the same time
- More efficient than using more processors per ESC
- Simple example: Müller-Brown potential (see picture)
- Periodic example (VASP): dehydrogenation of acetyl on platinum (111) surface
  - Fixed surface, 18 degrees of freedom
  - 7 beads on path
  - Start: linear interpolation between minima
  - 30th iteration (see picture): RMS perpendicular forces < 0.1 eV/Å
- Molecular example (ParaGauss): Neptunyl monoacetat in solution
  - 69 degrees of freedom





Parallel solver for eigenvalue problem

- Symmetry selection rules lead to blocked Hamiltonian matrix $\mathbf{H}$
- Symmetry helps reducing costs of diagonalization to < 5%
- Hamiltonian blocks are of significantly different sizes

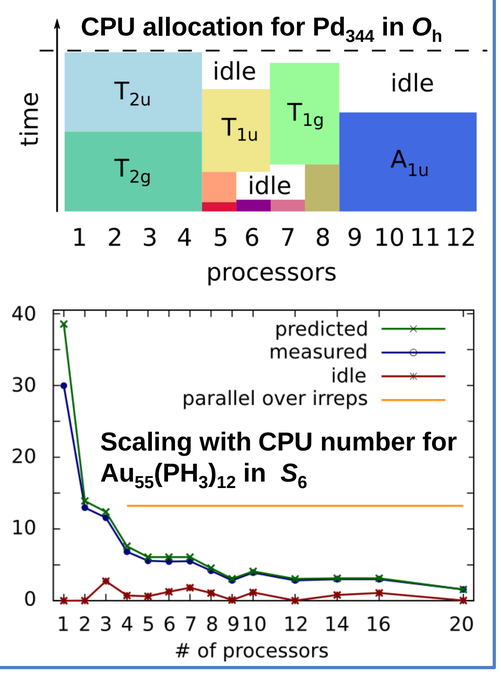
Pd_{344}, O_h symmetry

A_{1g}	A_{2g}	E_g	T_{1g}	T_{2g}	A_{1u}	A_{2u}	E_u	T_{1u}	T_{2g}
199	317	513	838	956	1110	317	471	785	956

$Au_{55}(PH_3)_{12}$, S_6 symmetry

A_g	E_g	A_u	E_u
782	1556	782	1560

- Existing parallel eigensolvers (LA- and ScaLAPACK)
  - inefficient for dim $\mathbf{H}$ << 500
  - turn-over quickly with number of processors
- Combinatorial approach feasible for typical (~10) number of blocks
- Requires accurate cost function for serial and parallel solvers
- Improves timings for medium sized systems
- Scales for large systems



[1] Th. Belling, Th. G rauschopf, S. Krüger, M. Mayer, F. Nörtemann, M. Staufer, C. Zenger, N. Rösch, in Lecture Notes in Computational Science and Engineering, H.-J. Bungartz, F. Durst, C. Zenger (eds.), vol. 8, Springer: Heidelberg 1999, p. 439.
[2] M. Roderus, A. Berariu, H.-J. Bungartz, S. Krüger, A. Matveev, N. Rösch, in Lecture Notes in Computer Science, P. D'Ambra, M. Guarracino, D. Talia (eds.), vol. 6272, Springer: Berlin 2010, p. 113.
[3] H. Chaffey-Millar, A. Nikodem, A. Matveev, S. Krüger, N. Rösch, in preparation.