Parallel Performance of VASP: CPU and GPU

Alireza Ghasemi

Erlangen National High Performance Computing Center

Monthly HPC Café, Erlangen, Nov. 21, 2023
Kohn-Sham Density Functional Theory

Kohn-Sham energy functional:

\[
E[\{\phi_i\}] = -\frac{1}{2} \sum_{i=1}^{N_{orb}} \int \phi_i^*(r) \nabla^2 \phi_i(r) d^3r + \int V_{\text{ext}}(r) \rho(r) d^3r + \frac{1}{2} \int \int \frac{\rho(r) \rho(r')}{|r - r'|} d^3r d^3r' + E_{xc}[\rho]
\]

\[
\rho(r) = \sum_{i=1}^{N_{orb}} f_i |\phi_i(r)|^2
\]

\[
\frac{\delta E[\rho]}{\delta \phi_i(r)} = 0 \quad \text{yields} \quad \text{Kohn-Sham eigenvalue problem}
\]

Computation cost:

- Basis set: type and size
- Number of KS orbitals
- Exchange-correlation energy \( E_{xc} \): conventional vs. hybrid functional
Vienna Ab initio Simulation Package (VASP)

Computational complexity of major tasks in VASP: plane-wave codes

- Application of Hamiltonian:
  - FFT: $O(N^2 \log(N))$
  - Potential: $O(N^2)$
  - Nonlocal part of the pseudopotential: $O(N^2)$ or $O(N^3)$

- Diagonalization $O(N^3)$
Vienna Ab initio Simulation Package (VASP)

Degrees of freedom within optimization:

- **FFTs:** coefficients, $\text{NCORE}$
- Number of orbitals: **MPI groups**
- **K-points:** $\text{KPAR}$

**VASP makes decisions on top of users’**:

- Number of orbitals, $\rho(r) = \sum_{i=1}^{N_{\text{orb}}} f_i |\phi_i(r)|^2$
- **NCORE**: Parallel efficiency and Memory requirement
- **LREAL=Auto**, **RHOP=1.E-4**, $O(N^2)$ or $O(N^3)$
- **Different versions of VASP!**
Simulation supercells

The systems in this benchmark:

- Three supercells of rocksalt bulk sodium chloride:
  - System(I): #atoms=64 ; #electrons=2*224
  - System(II): #atoms=512 ; #electrons=2*1792
  - System(III): #atoms=1728 ; #electrons=2*6048

- Tiny – Small – Medium – Large - Huge

- System(I) with 2x2x2 k-points with \( \text{ISYM}=0 \) and \( \text{KPAR}=1 \), the rest with no k-point.

- Relaxed geometry with PBE from:
  - https://next-gen.materialsproject.org
HPC clusters & systems at NHR@FAU

HPC clusters used in this benchmark

- Fritz Ice-Lake
  - Two Intel Xeon Platinum 8360Y, base frequency 2.4 GHz
  - 54 MB shared L3 cache per chip
  - 256 GB of DDR4 RAM

- Alex
  - A40: 48 GB DDR6, 696 GB/s, 37.42 TFlop/s in FP32
  - A100: 40 GB HBM2, 1,555 GB/s, 9.7 TFlop/s in FP64 or 19.5 TFlop/s in FP32
  - A100: 80 GB HBM2, 2,039 GB/s, 9.7 TFlop/s in FP64 or 19.5 TFlop/s in FP32
VASP: version, compilation, and libraries

- On Fritz:
  - VASP-6.3.2
  - Intel Compiler, MPI, and MKL

- On Alex:
  - VASP-6.3.0
  - NVHPC Compiler, openmpi CUDA-enabled, Intel MKL
  - NVIDIA NCCL

- All calculations with
  - Standard binary: vasp_std
  - ENCUT=500 eV
**PBE: Impact of NCORE**

System(I) on 1 node (72 cores)

NCORE is **not** the number of CPU cores!

![Graph showing the impact of NCORE on relative time and number of orbitals.](image-url)
PBE: Impact of NCORE

System(I) on 8 nodes (576 cores)

- $\rho(r) = \sum_{i=1}^{N_{orb}} f_i \left| \phi_i(r) \right|^2$
- Occupied orbitals: 224
  - Default: 1.2*224
- $T_{NCORE=4}/T_{NCORE=36} \approx 2$
PBE: Impact of NCORE

System(II) on 1 node (72 cores)
PBE: Impact of NCORE

System(II) on 8 nodes (576 cores)

The graph shows the relative time (T/T_{NCORE=1}) as a function of NCORE and NBANDS. The data indicates a decrease in relative time with an increase in NCORE, suggesting improved scalability. The number of orbitals also affects the relative time, with a decrease as the number of orbitals increases.
PBE: Impact of NCORE

System(III) on 1 node (72 cores)

- Insufficient memory for NCORE<18
PBE: Impact of NCORE

System(III) on 8 nodes (576 cores)

- Insufficient memory for NCORE=1
Roofline analysis

System(III): MPI-only on one node

- Screenshot from ClusterCockpit:
  https://monitoring.nhr.fau.de

- Average performance
  - 1169 GFLOPS
  - 258 GB/s
  - 4.5 FLOP/Byte
PBE: MPI parallelization

Parallel performance on Fritz with MPI-only

- NCORE=36
- No drastic change in NBANDS
- Fritz nodes are exclusive!
- Parallel efficiency < 0.8:
  - System(I): nodes>= 4
  - System(II): nodes>=10
  - System(III): nodes>=24
PBE: hybrid parallelization

Parallel performance on Fritz with OpenMP

- NCORE=1

  
  OMP_NUM_THREADS=18
  export OMP_PLACES=sockets
  export OMP_PROC_BIND=close

- Drastic change in NBANDS for system(I) with nodes>14

- Improved speedup over MPI-only but not necessarily faster!

- Parallel efficiency < 0.8: Systems (I), (II), and (III): nodes>= 6, 14, and 40, respectively
In the case of OpenMP, drastic change in NBANDS for nodes>14

OpenMP: Shortest run time with nodes=16

Parallel efficiency < 0.8
- MPI: nodes>= 4
- OMP: nodes>=6
OpenMP results in shortest possible run time in both systems (II) & (III)

No drastic change in NBANDS for MPI, but slightly for OpenMP
PBE

Memory use for system(III): MPI vs. OMP

- export OMP_STACKSIZE=?
  - Not used in this case!
- PBE, standard DFT, with 1728 atoms on one node needs almost all memory available!
HSE

System(I): MPI vs. OpenMP

- `export OMP_STACKSIZE=512m`
- Decent parallel performance both with MPI and OMP
- Problem with NBANDS more important in the case of HSE!
NCCL: NVIDIA Collective Communications Library
  - Topology-aware inter-GPU communication
OpenACC: hiding launch latency by asynchronous execution queues
  - Independent kernels
Each A100 GPU: 0.55 €/hour
System(I): MPI on one node ✓
Each Fritz node: 0.45 €/hour
NCCL: no concern with NBANDS
### PBE: run time and cost

**System(II): GPU vs. MPI vs. OMP**

- **OMP**: minor problem with NBANDS for large number of nodes
- **GPU**: shortest possible run time!
PBE: run time and cost

System(III): GPU vs. MPI vs. OMP

- OMP: despite minor problem with NBANDS for large #nodes, shortest possible run time!

- GPU:
  - insufficient device memory, #GPUs<4
  - Superlinearity, 1.03, 1.07, 1.09, 0.98
HSE: run time and cost

System(I): GPU vs. MPI vs. OMP

- A100 (80GB) 10% better than A100 (40GB) while having 30% higher bandwidth
- A40: ✗
- OMP: problem with NBANDS
HSE: run time and cost

System(II): GPU vs. MPI vs. OMP

- **MPI**: better than previous tests
- **OMP**: shortest run time
  - `export OMP_STACKSIZE=2048m`
- **A100**:
  - Due to memory A100 (80GB) needed
  - Parallel efficiency: 0.99
Summary

- **NCORE**: important for memory and performance, check for NBAND
- Shortest run time: either of OpenMP or GPU
  - Depending on type of simulation and system size
- Best way to reduce memory requirement is the use of OpenMP with OMP_NUM_THREADS set to number of cores in NUMA domain
  - Do not forget `export OMP_STACKSIZE=??`
- At shortest run time, i.e. many CPU nodes or multiple GPUs, the latter is cheaper
- Running on GPU with NCCL, no concern over NBAND
- Running VASP on GPUs for Hartree-Fock and HSE calculations is highly recommended and it is a good choice in every respect.
Thanks for support:

- Christoph Kluge
- Thomas Zeiser
- Johannes Veh

Thank you all