Continue

Orca 4.1.0 manual

ORCA 4.0.1.2 and 4.1.0 executables have been installed on Owens and Pitzer clusters for users to access. For more information, please refer to the software page of ORCA. ORCA is a powerful quantum chemistry tool used primarily for studying spectroscopic properties in open-shell molecules. It offers a wide range of computational methods, including semiempirical, DFT, ab initio, and correlated methods. The installed versions are listed below: - Version Owens Pitzer Cardinal 4.0.1.2 X X openmpi/2.1.6-hpcx 4.1.0 X X openmpi/3.1.4-hpcx 4.1.1 X X openmpi/3.1.4-hpcx 4.1.2 X X openmpi/3.1.4-hpcx 4.2.1 X* X* openmpi/3.1.6-hpcx 5.0.0 X X openmpi/5.0.2-hpcx 5.0.2 X X openmpi/5.0.2-hpcx 5.0.3 X X openmpi/5.0.2-hpcx 5.0.4 X X For users to access ORCA, you need to sign up for the ORCA Forum and receive a registration confirmation email. You can then contact OSC Help with this email to gain access to the tool. ORCA modules are controlled through various methods. You can load a specific ORCA version by using the command "module load orca/{version}" at the command line, in your shell initialization script, or within your batch scripts. For instance, use "module load orca/4.2.1" to load ORCA version 4.2.1. Batch Usage Upon logging into owens.osc.edu or pitzer.osc.edu, you are essentially accessing a Linux box referred to as the login node. To leverage the multiple processors in the computing environment, you must submit your job to the batch system for execution. Batch jobs can request multiple nodes/cores and compute time up to the limits of the OSC systems, as outlined in Queues and Reservations and Batch Limit Rules. These jobs execute on the compute nodes rather than the login node. Interactive Batch Session To initiate an interactive batch session, run the command: "sinteractive -A -n 1 -t 00:20:00". You can adjust the numbers according to your needs. Non-Interactive Batch Job A batch script can be created and submitted for a serial or parallel run. You can create the batch script using any text editor you prefer in a working directory on the system of your choice. Below is an example batch script for a parallel run: #!/bin/bash #SBATCH --job-name=orca_mpi_test #SBATCH --time=0:10:0 #SBATCH --nodes=2 --ntasks-per-node= #SBATCH --account= module reset module load openmpi/3.1.6-hpcx module load orca/4.2.1 module list sbcast -p h2o_b3lyp_mpi.inp $TMPDIR/h2o_b3lyp_mpi.inp cd $TMPDIR $ORCA/orca h2o_b3lyp_mpi.inp > $SLURM_SUBMIT_DIR/h2o_b3lyp_mpi.out Please note that the should not exceed the maximum cores per node. You can refer to Cluster Computing for more information on each cluster's maximum number of cores. Best practices It is generally recommended to utilize 3000, which represents 75% of the usable memory per core on each cluster. However, you may need to increase %maxcore due to the methods and modular system used. In this case, you can decrease the number of cores for the same job. For example, if you have a script that runs an 80-core ORCA job on two Pitzer 40-core nodes: #!/bin/bash #SBATCH --nodes=2 --ntasks-per-node=40 module reset module load To run ORCA/4.2.1, load the module and list available sbcast options. Run the job with mpi input file and output redirection. For increased core usage, modify the command line to specify 30 cores per node. Be aware of potential multi-node hang issues due to heavy I/O in certain modules. A solution is to upgrade ORCA to version 5.0.x and OpenMPI to 5.0.x. Default CPU binding for ORCA jobs may fail; add "overload-allowed" option to bypass protection. Known workarounds include disabling CPU binding, using a non-hpcx MPI module, or adjusting SLURM ntasks relative to ORCA nprocs. These solutions may have performance side effects and direct charging consequences. Please remove all extra parameters for mpirun from the command line. For example, $ORCA/orca h2o_b3lyp_mpi.inp "--machinefile $PBS_NODEFILE" > h2o_b3lyp_mpi.out should be changed to $ORCA/orca h2o_b3lyp_mpi.inp > h2o_b3lyp_mpi.out We found a bug in OpenMPI following the recent SLURM update, causing multi-node MPI jobs to fail immediately when using mpirun. To fix this issue, we replaced mpirun with srun in ORCA. ORCA 4.1.0 issue with scratch filesystem: When running an MPI job that requires multiple nodes, it can be run from a globally accessible working directory such as home or scratch directories to have more space for temporary files. However, ORCA 4.1.0 cannot run jobs on our scratch filesystem. This issue has been reported and resolved in ORCA 4.1.2. For further information, please refer to the Storage Documentation and scratch storage information available from Chemical Engineering & Chemistry. Hujo, Mihály Kállay, Holger Kruse, Jiri Pittner, Philipp Pracht, Marcel Müller, Liviu Ungur, Edward Valeev, Lukas Wittmann and also Wolfgang Schneider contributed to the manual. The manual is available as a PDF file named orca.pdf.

Orca 4.2.1. Orca manual.