Mathematical Libraries and Application Software

JSC Training Course

May 2021 | I.Gutheil, D.Alvarez | JSC

JÜLICH Forschungszentrum | JÜLICH SUPERCOMPUTING CENTRE

Outline

- Navigating modules
- Compiling for the JURECA Booster
- Sequential Libraries
- Parallel Libraries and Application Systems:
  - Threaded Libraries
  - MPI parallel Libraries
  - Application System
  - GPU Library
- Software for Materials Science
- Software for Computational Engineering
- Further Information

Modules environment

Hierarchical modules

- `GCCcore/.9.3.0` is preloaded, which enables a lot of base software
- For HPC software you have to load a compiler, to expand the module tree, for example
  `ml GCC` (Default is `9.3.0`)
  `ml Intel` (Default is `2020.2.254`)
- Then you load an MPI version, for example
  `ml ParaStationMPI`
  Default version is `5.4.7-1` (`MPI_THREAD_MULTIPLE` is not supported, to get it you need `5.4.7-1-mt`)
- Then you can load any other packages, for example
  `ml QuantumESPRESSO/6.6`

Modules environment

Available MPI versions in Stages/2020

- `ParaStationMPI/5.4.7-1` (`5.4.8-1`)
- `ParaStationMPI/5.4.7-1-mt` (with `MPI_THREAD_MULTIPLE`)
  currently only with Intel compiler,
  can be installed on request
- `OpenMPI/4.1.0rc1`
- `IntelMPI/2019.8.254` (not recommended on AMD hardware)

Modules environment

Available Compilers in Stages/2020

- `GCC/9.3.0`
- `Intel/2020.2.254`
- `NVHPC/20.11`, (`NVHPC/21.1`)

Available Compiler/MPI Combinations in Stages/2020

Compiler	MPI	Cuda available
GCC 9.3.0	ParaStationMPI 5.4.7	yes
GCC 9.3.0	OpenMPI 4.1.0rc1	yes
NVHPC 20.11	ParaStationMPI 5.4.7	yes
NVHPC 21.1	ParaStationMPI 5.4.8-1	yes
Intel 2020.2.254	ParaStationMPI 5.4.7[-mt]	yes
Intel 2020.2.254	IntelMPI 2019.8.254	no
Intel 2020.2.254	OpenMPI 4.1.0rc1	yes

Modules environment

Available Combinations of Compilers/MPI/Math libraries in Stages/2020

Compiler	MPI	Math	Cuda	arch
GCC	ParaStationMPI	MKL 2020.4.304	yes	both
GCC	OpenMPI	MKL 2020.4.304	yes	both
Intel	ParaStationMPI[-mt]	MKL 2020.4.304	yes	both
Intel	IntelMPI	MKL 2020.4.304	no	Intel
Intel	OpenMPI	MKL 2020.4.304	yes	both
NVHPC	ParaStationMPI	MKL 2020.4.304	yes	GPU

Modules environment

- After loading compiler and MPI `ml avail` shows the software available with that combination
- `ml avail` *name* and `ml help` *name* will show you details about the *name* package
- Many libraries are available for more than one combination/toolchain

- Write e-mail to `sc@fz-juelich.de` if you want special versions or new software
  - No guarantee the software will be installed
- $EBROOT*NAME* is the root directory where the library is installed

Modules environment

- `ml spider` *name* shows whether a library is available in the current stage and in which versions
- `ml spider` *name*/*version* shows which environment you have to load before you can load that version

- Many packages are hidden. To see them use
  `ml --show-hidden spider` *name*

Modules environment

- For R, Python and Perl we use bundles
  - You might be looking for a software package that is part of a bigger module
- Use `ml key software`
  `ml key numpy` will suggest `SciPy-Stack`
- You can use then `ml spider` to find out how to load the module

Modules environment

Stages

- The whole software stack in JURECA-DC, JURECA Booster, JUWELS cluster, and JUWELS Booster will be updated regularly
- Current stage is 2020
- Old stages are still accessible on JUWELS cluster and JURECA Booster, but no guarantee
- To check availability in other stages first type
  `ml use $OTHERSTAGES`
- MKL version 2020.4.304 is now Default, but for installed software needing MKL it will be switched back to 2020.2.254
- New GCC version will soon be available in the default stage

Compiling for the JURECA Booster

- Cross-compilation on login nodes is not recommended but technically possible
- `ml Architecture/KNL`
  - This should be the first module you load
  - Then you have the JURECA booster software stack available.
  - This doesn't work on the cluster compute nodes!

- Remember, cross-compiling is not recommended unless you know what you are doing

Compiling for the JURECA Booster

Recommended way to compile for the JURECA Booster

- Start an interactive session, by
  `salloc --partition=booster-devel -A <your account>`
- After the allocation is successful start a remote shell from within the salloc session and connect it to a pseudo terminal using
  `srun --cpu_bind=none --nodes=1 --pty /bin/bash -i`
  Architecture is switched to KNL automatically then

- In batch scripts add
  `module load Architecture/KNL`
  before all other module commands

Sequential Libraries and Packages

Remember that with intel compilers if you want to link Fortran subroutines with the C linker you need to add -lifcore -lifport

Vendor specific Libraries for Intel architectures

- MKL Intel® Math Kernel Library
  2020.4.304 and 2020.2.254 on JUWELS, JURECA-DC, and JURECA Booster
  For more information see
  https://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Software/SystemDependentLibraries/MKL.html?nn=1035570

Vendor specific Libraries for AMD

- BLIS version 2.2-amd on JURECA-DC and JUWELS Booster,
  for some applications faster than MKL,
  in Stages/Devel-2020 only
- libFLAME version 2.2-amd library containing LAPACK calling BLIS,
  in Stages/Devel-2020 only,
  use with care, there are a few corner cases with wrong results,
  LAPACK and LAPACKe public domain versions will be made available

Sequential Libraries and Packages

Public domain Libraries

- LAPACK (Linear Algebra PACKage)
- ARPACK (Arnoldi PACKage)
- GSL (Gnu Scientific Library)
- GMP (Gnu Multiple Precision Arithmetic Library)
- METIS (Serial Graph Partitioning and Fill-reducing Matrix Ordering)
- SCOTCH (Graph Partitioning)

Commercial library

NAG Fortran and C Library: JURECA-DC only

Contents of Intel® MKL

- BLAS, Sparse BLAS, CBLAS
- LAPACK
- Iterative Sparse Solvers, Trust Region Solver
- Vector Math Library
- Vector Statistical Library
- Fourier Transform Functions
- Trigonometric Transform Functions

Contents of Intel® MKL

- GMP routines
- Poisson Library
- Interface for fftw

For more information see
https://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Software/SystemDependentLibraries/MKL.html?nn=1035570

Contents of AMD BLIS and libFLAME libraries

- BLIS: BLAS, CBLAS
- libFLAME: LAPACK

For information how to use BLIS and libFLAME see
https://developer.amd.com/wp-content/resources/AOCL_User%20Guide_2.2.pdf

Usage of Intel® MKL

- Can be loaded with Intel compiler or GCC
- MPI has to be loaded before imkl, only MPICH, no OpenMPI
- FORTRAN, C, and C++ callable
- Arrays FORTRAN like, i.e. column-first (except cblas)
- Compilation and linking of program name.f calling sequential MKL routines:

```
ifort name.f -o name -lmkl_intel_lp64 -lmkl_sequential
-lmkl_core [-liomp5 -lpthread]
```

Usage of Intel® MKL

To use CBLAS include mkl.h into source code

Compilation and linking of program name.c calling sequential MKL

```
icc name.c -o name -lmkl_intel_lp64 -lmkl_sequential
-lmkl_core [-liomp5 -lpthread -lifcore -lifport]
```

For flags to compile with other compilers or multithreaded versions take a look at:

https://software.intel.com/en-us/articles/intel-mkl-link-line-advisor

LAPACK

- Part of MKL in libmkl_core.a
- Can be loaded with Intel, GCC on JURECA-DC, JURECA Booster, and JUWELS
- On AMD hardware libFLAME/2.2-amd
- LAPACK public domain version to come

Arpack

- ARPACK-NG/3.7.0
- Iterative solver for sparse eigenvalue problems
- Reverse communication interface
- FORTRAN 77
- Calls LAPACK and BLAS routines, MKL or BLIS and libFLAME loaded automatically

GSL – GNU Scientific Library

- `module load Intel GSL/2.6`
  for icc version
- `module load GCC GSL/2.6`
  for gcc version
- Provides a wide range of mathematical routines
- Not recommended for performance reasons
- Often used by configure scripts

GMP- GNU Multiple Precision Library

- version 6.2.0

METIS

- version 5.1.0

SCOTCH

- version 6.1.0

NAG Libraries

- NAG Mark 27 combined C and Fortran library on JURECA-DC available with Intel compiler

Parallel Libraries

Threaded Parallelism

- MKL
  is multi-threaded when linked with `-lmkl_[intel,gnu]_thread`
  Usage:
  ```
  ifort name.f -o name -lmkl_intel_lp64
  -lmkl_intel_thread -lmkl_core -liomp5 -lpthread
  ```
- BLIS-amd is multi-threaded when linked with `libblis-mt.a`
- FFTW 3.3.8 (Fastest Fourier Transform of the West)
  MPI, OpenMP and threads version
  `http://www.fftw.org`
  if OMP_NUM_THREADS is not set,
  256 threads used on JURECA-DC
  68 threads on JURECA Booster
  96 threads on JUWELS and JUWELS Booster
  (80 in accelerated JUWELS cluster nodes)

Parallel Libraries

MPI Parallelism

- ScaLAPACK (Scalable Linear Algebra PACKage)
- ELPA (Eigenvalue SoLvers for Petaflop-Applications)
- Elemental, C++ framework for parallel dense linear algebra
- FFTW (Fastest Fourier Transform of the West)
- MUMPS (MUltifrontal Massively Parallel sparse direct Solver)
- ParMETIS (Parallel Graph Partitioning)
- PT-SCOTCH (Parallel Graph Partitioning)
- Hypre (high performance preconditioners)

Parallel Libraries

MPI Parallelism

- PARPACK (Parallel ARPACK), Eigensolver
- SPRNG (Scalable Parallel Random Number Generator)
- SUNDIALS (SUite of Nonlinear and DIfferential/ALgebraic equation Solvers)

Parallel Systems, MPI Parallelism

- PETSc, toolkit for partial differential equations
- SLEPc Scalable Library for Eigenvalue Problem Computations Extension to PETSc for the computation of eigenvalues and eigenvectors

GPU Library

- MAGMA, Matrix Algebra on GPU and Multicore Architectures

ScaLAPACK

- part of MKL, only for IntelMPI and ParaStationMPI
  recommended on Intel Architecture
- ScaLAPACK/2.2-amd with GCC
  recommended on AMD hardware,
  calls BLIS and libFLAME
- Parallel BLAS 1-3, PBLAS Version 2
- Dense linear system solvers
- Banded linear system solvers
- Solvers for Linear Least Squares Problem
- Singular value decomposition
- Eigenvalues and eigenvectors of dense symmetric/hermitian matrices
- `http://www.netlib.org/scalapack/index.html`

Usage of ScaLAPACK

- Linking a program name.f calling routines from MKL ScaLAPACK, Intel compiler:

```
mpif77 name.f -lmkl_scalapack_lp64
-lmkl_blacs_intelmpi_lp64 -lmkl_intel_lp64
-lmkl_intel_thread[-lmkl_sequential]
-lmkl_core -liomp5 -lpthread
```

- For information how to use ScaLAPACK-amd see
  https://developer.amd.com/wp-content/resources/AOCL_User%20Guide_2.2.pdf

ELPA

Eigenvalue SoLvers for Petaflop-Applications

ELPA uses ScaLAPACK, must be linked together with ScaLAPACK

- FORTRAN 2003, same data-distribution as ScaLAPACK
- http://elpa.rzg.mpg.de/elpa-english?set_language=en
- Pure MPI and hybrid version 2020.05.001
- Version with GPU acceleration only pure MPI

Elemental

- C++ framework, two-dimensional data distribution element by element
- https://github.com/elemental/Elemental
- 0.87.7 old with patches
- Hydrogen 1.5.0 will be in Stages/Devel-2020, GPU version of parts of Elemental
- https://github.com/LLNL/Elemental

MUMPS

MUltifrontal Massively Parallel sparse direct Solver

- Solution of linear systems with symmetric positive definite matrices, general symmetric matrices, general unsymmetric matrices
- Real or Complex
- Parallel factorization and solve phase, iterative refinement and backward error analysis
- F90 with MPI and OpenMP since 5.3.4
- `http://graal.ens-lyon.fr/MUMPS/`

ParMETIS

Parallel Graph Partinioning and Fill-reducing Matrix Ordering
developed in Karypis Lab at the University of Minnesota
Version 4.0.3
http://glaros.dtc.umn.edu/gkhome/
Version with double precision real values available

PT-SCOTCH

Software package and libraries for sequential and parallel graph partitioning, static mapping and clustering, sequential mesh and hypergraph partitioning, and sequential and parallel sparse matrix block ordering
Version 6.1.0
https://www.labri.fr/perso/pelegrin/scotch/

Hypre

High performance preconditioners
Version 2.20.0, also version with bigint,
https://github.com/hypre-space/hypre

FFTW

Version 3.3.8 Intel and GCC compiler, all MPI versions
Special -amd version for AMD architectures

PARPACK

- PARPACK MPI-Version, part of ARPACK-NG/3.7.0
- Must be linked with LAPACK and BLAS
- Reverse communication interface, user has to supply parallel matrix-vector multiplication

https://github.com/opencollab/arpack-ng

SPRNG

The Scalable Parallel Random Number
Generators Library for ASCI Monte Carlo Computations
version 5.0-14042019:
various random number generators in one library
Version 1.0 seperate library for each random number generator
http://sprng.cs.fsu.edu/

Sundials (CVODE)

Package for the solution of ordinary differential equations,
Version 5.4.0
https://computation.llnl.gov/casc/sundials/main.html

Application System PETSc

- Portable, Extensible Toolkit for Scientific Computation
- Numerical solution of partial differential equations
- version 3.14
- with several other packages included
- complex version and version with 8-Byte integer
- debug versions in Devel Stages only
- `http://www.mcs.anl.gov/petsc/`
- `ml spider PETSc`

Extension SLEPc

- Scalable Library for Eigenvalue Problem Computations
- linear and nonlinear eigenvalue problems
- partial SVD
- 3.14
- PETSc will be loaded automatically
- works only with the same MPI that was used to install PETSc
- `https://slepc.upv.es`

GPU Library MAGMA

- MAGMA, Matrix Algebra on GPU and Multicore Architectures
- version 2.5.4 with CUDA 11 support
- https://icl.utk.edu/magma/index.html

Software for Materials Science

Package	JURECA-DC	JUWELS
Abinit	yes	yes
ADF	no	no
Amber	no	yes
CP2K	yes	yes
CPMD	yes	no
GPAW	yes	yes
Gromacs	yes	yes
LAMMPS	yes	yes
Molpro	no	no
NAMD	yes	yes
NWChem	yes	no
QuantumEspresso	yes	yes
TURBOMOLE	no	no

Software for Computational Engineering

- JURECA-DC and JUWELS
- CFD Package **OpenFOAM**
  Version 6.0 in Stages/2018a
  and OpenFOAM-Extend 4.0 in Stages/2018a
- Commercial **FEM Software**
  - **ANSYS**, **LS-DYNA** , **COMSOL** are technically maintained on **JURECA-DC** only
  - **Licenses** must be provided by the **User !**

Further information and JSC-people

http://www.fz-juelich.de/ias/jsc/jureca
http://www.fz-juelich.de/ias/jsc/juwels
http://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Software/_node.html

Mailto

Supercomputer support:
sc@fz-juelich.de

I. Gutheil: Parallel mathematical Libraries
i.gutheil@fz-juelich.de

D. Alvarez: Software Manager
d.alvarez@fz-juelich.de

B. Körfgen: Physics and Engineering software
b.koerfgen@fz-juelich.de