

Preliminary remarks

This guide describes the steps for running parallel computations performed by ANSYS on the Lichtenberg cluster¹. In this case a computational problem is prepared in advance by means of an application with the graphical user interface (GUI), while computations on the cluster are performed only in batch mode without using the GUI.

If this is the first time you want to run a job on Lichtenberg Cluster, please visit an introductory course!

ANSYS is a licensed software, please respect the license-agreement for the use of ANSYS at TU Darmstadt:
www.hrz.tu-darmstadt.de/software/uebersicht_1/campuslizenzen/ansys

ANSYS Mechanical on a Linux machine - Command Line

To use Ansys MAPDL on the Lichtenberg cluster, a module of a proper version must be found and loaded:

```
module avail ansys  
module load ansys/<VERSION>
```

Attention: It is highly recommended to prepare a problem and to perform computations by using the same version of the ANSYS package (it must be checked before computations).

After loading the module, the Ansys manual with detailed information is available by the command

```
anshelp
```

To run ANSYS MAPDL for performing parallel computations in batch mode on the cluster, the `mapdl` command is applied with the following options:

- `-dis` to enable the distributed solver;
- `-usessh` to switch from `rsh` to `ssh`;
- `-mpi intelmpi` to specify Intel MPI as the type of the used MPI;
- `-machines <machines>` to specify the machines for computing;
- `-j` to specify a name assigned to all files generated by the program for a specific model;
- `-s` to specify that the program reads the `start.ans` file at start-up;
- `-b` to activate the MAPDL program in batch mode;
- `-i <input_file>` to specify the input file of a computational problem;
- `-o <output_file>` to specify the output file of the batch execution of MAPDL.

¹This guide should work for other clusters with the scheduler SLURM, too

The names of the machines, <machines>, are generated automatically (see the batch script at the end). The input file <input_file> is a text file which describes the numerical problem in terms of the APDL language (it can be generated in any ANSYS GUI, e.g. ANSYS Workbench). The output file <output_file> is generated during computations and contains some results. The names of the input and output files are user-defined and must **not** contain any whitespaces.

As an example, the following command is applied to start ANSYS MAPDL for parallel computing in batch mode with the input file `inputfile.inp` and the output file `outputfile.out`:

```
mapdl -dis -usessh -mpi intelmpi \
  -machines $MYMACHINES \
  -j "file" \
  -s read \
  -b \
  -i ./inputfile.inp \
  -o ./outputfile.out
```

The input and output files are given without paths and must be located in the same directory where the `mapdl` command is executed (this directory is called the working directory). Otherwise, these names must be specified together with their paths. How to use this command for computing on the cluster is discussed at the end of this manual.

Submitting a job: Batch script

To run parallel computations on the Lichtenberg cluster, the `mapdl` command with its options must be specified in a so-called batch script, which is a text file. An example of the batch script for submitting the job to the scheduler SLURM is shown in listing 1. The command to submit the batch script to SLURM:

```
sbatch <batch_script>
```

where the name <batch_script> is user-defined (whitespaces are not allowed). The directory where the batch script is submitted is called the working directory. If all input/output file are located in this directory, it is enough to specify only their names without the paths to these files.

The batch script must include the settings carefully specified and described below.

1. The name of the job is specified by the `-J` option (whitespaces are not allowed).
2. The name of the standard output file and the name of the file of errors are specified by the options `-o` and `-e`, respectively (whitespaces are not allowed).
3. The required memory specified by `--mem-per-cpu` is an estimated value. In case of memory issue, the value can be increased until a certain limit (e.g. the maximum of memory available for a processor).
4. The type of computational nodes specified by `-C` is limited by `avx` for phase 1 (Sandy-Bridge nodes) and `avx2` for phase 2 (Haswell nodes).
5. The wall-clock time specified by `-t` should be estimated carefully. It is recommended to perform some tests to estimate the rough time required for the analysis. The corresponding average time can be found in the output file of ANSYS MAPDL. The wall-clock time must be chosen so that computations have performed within the time-window. Note that computations with a shorter wall-clock time can be started by SLURM faster.

6. The number of cores for computing specified by the `-n` option should be chosen carefully. On the one hand, many cores allow to run numerical simulation of large problems and to perform computations faster. On the other hand, there is the maximum number of cores, when further increase does not influence the performance significantly. This maximum depends on a problem and can be determined by performing short tests.
7. The module of the ANSYS package loaded for performing computations on the cluster must correspond to the version of ANSYS used in preparing numerical simulation.

To avoid problems while running computations, the important recommendations must be followed.

1. Names of directories/files of all input/output files used in computations must not contain whitespaces.
2. One directory should be used as the working directory only for one run of computations to avoid rewriting output files. In other words, only one job should be submitted in one directory, where output files are generated.
3. If the input file is used in multiple numerical simulations, it is reasonable to store it in a separate directory and to specify its name together with the absolute path.
4. It is highly recommended to specify all necessary loaded modules directly in the batch script and not to use other files including configuration files (e.g. the `.bashrc` file).

```
#!/bin/bash

#SBATCH -J jobmapdl_test
#SBATCH -e jobmapdl_test.err.%j
#SBATCH -o jobmapdl_test.out.%j
#SBATCH --mem-per-cpu=1650
#SBATCH -C avx
#SBATCH -t 00:10:00
#SBATCH -n 16

module purge
module load ansys/17.2

#Generate line of names of computational nodes.
MYMACHINES=$(srun hostname | sort | uniq -c | \
    awk '{print $2 ":" $1}' | \
    paste -s -d ":" -)

echo $MYMACHINES

#Run Ansys MAPDL.
mapdl -dis -usessh -mpi intelmpi \
    -machines $MYMACHINES \
    -j "file" \
    -s read \
    -b \
    -i ./inputfile.inp \
    -o ./outputfile.out
```

Listing 1: The example of the batch script for running parallel computations on the Lichtenberg cluster performed by ANSYS MAPDL with using 16 cores of nodes of phase 1. The names of the computational nodes (machines) stored in the MYMACHINES variable are generated automatically. The input file, `inputfile.inp`, is prepared in advance. The output file of the batch execution, `outputfile.out`, is generated by ANSYS during computations. Both names are user-defined, must **not** contain whitespaces, and are located in the working directory. The script can be adapted for other parallel computations performed by ANSYS MAPDL.

Remark

The present “How To” is based on the experience obtained in the HKHLR group. Any questions, suggestions and possible improvements are welcome as well as the feedback, if the guide is useful. Please, contact us by email: staff@hpc-hessen.de.