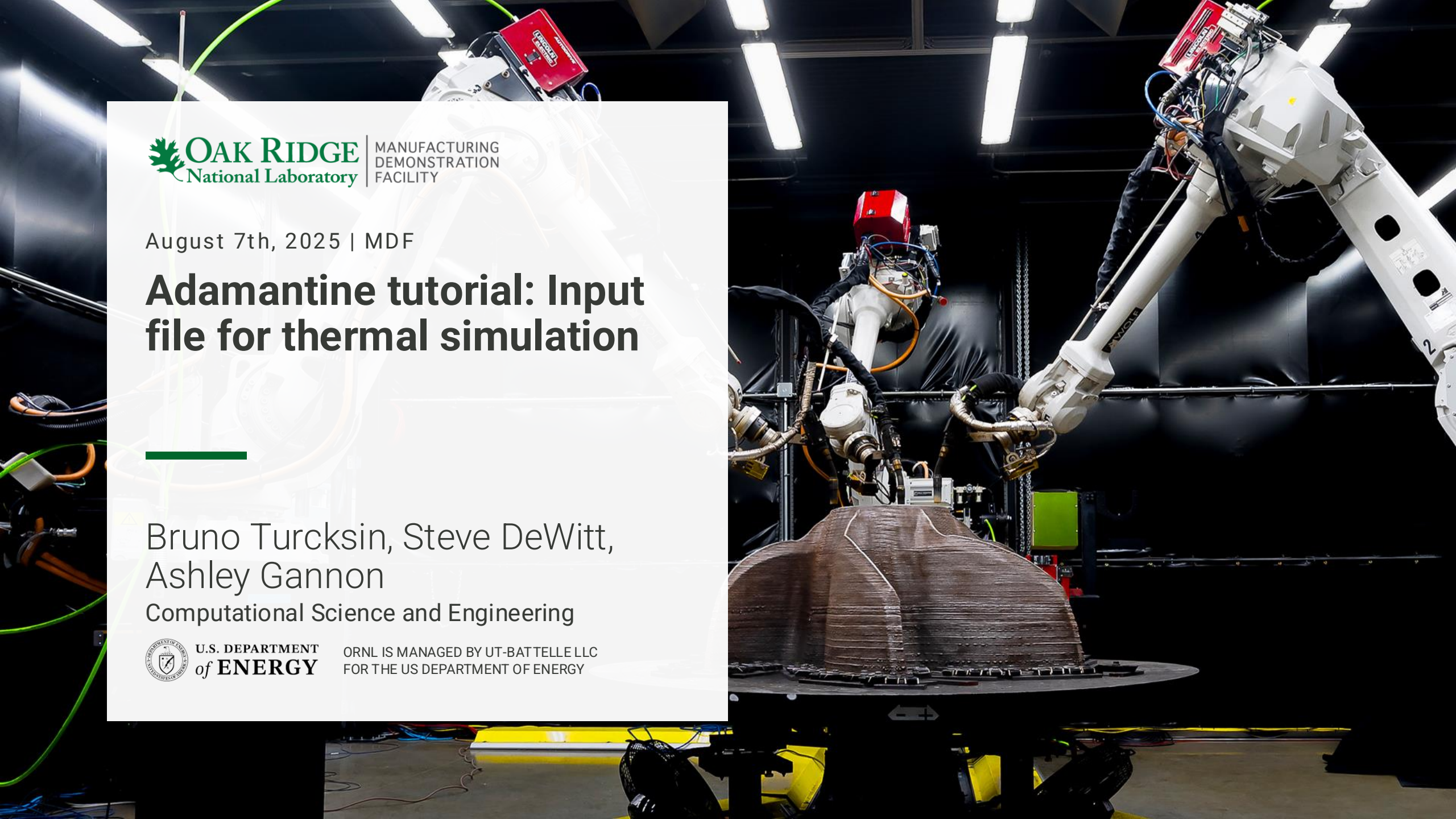

OAK RIDGE National Laboratory | MANUFACTURING DEMONSTRATION FACILITY

August 7th, 2025 | MDF

Adamantine tutorial: Input file for thermal simulation

Bruno Turcksin, Steve DeWitt, Ashley Gannon

Computational Science and Engineering

U.S. DEPARTMENT of ENERGY

ORNL IS MANAGED BY UT-BATTELLE LLC
FOR THE US DEPARTMENT OF ENERGY

Visualization

- Output files produced by a simulation:
  - **.vtu:** output data for a given time step and a given processor
  - **.pvtu:** one file per time step to assemble the .vtu from different processors
  - **.pvd:** one file per simulation to assemble the different .pvtu files
- Inactive cells have a temperature of 0K
- To only visualize the active cells:
  - In VisIt: use Threshold operator with a minimal temperature
  - In ParaView: use the Threshold filter with a minimal temperature

Input file

- Based on Boost Property Tree: arbitrarily nested tree of keys-values
- Supported formats:
    - info: Boost format, supports comments
    - json: file extension must be .json
- By default, use SI units but this can be changed in the input file

Input file

```
geometry
{
  import_mesh false ; Use built-in mesh generator
  dim 2 ; dimension of the domain
  length 2e-2 ; [m]
  height 1e-2 ; [m] In 3D, the third parameters is width
  length_divisions 20 ; Number of cell layers in the length direction
  height_divisions 10 ; Number of cell layers in the height direction
}

physics
{
  thermal    true  ; Thermal simulation
  mechanical false ; Mechanical simulation. If both thermal and mechanical are
                   ; true, solve a coupled thermo-mechanical problem
}

boundary
{
  type adiabatic
}
```

Required inputs

- **geometry:** simulation domain
- **boundary:** boundary conditions
- **time_stepping:** time step size and end time of the simulation
- **sources:** heat sources
- **materials:** material properties
- **discretization:** order the finite element used
- **physics:** thermal and/or mechanical simulation
- **post_processor:** filename and output frequencies
- **refinement:** number of refinement cycles and frequency between refinement

Optional inputs

- **ensemble:** parameters to create an ensemble of simulations with input parameters that follow a normal distribution
- **checkpoint:** filename and frequency of checkpoints
- **data_assimilation:** parameters used by EnKF
- **experiment:** information about the experimental data
- **memory_space:** decide between CPU and GPU simulation
- **microstructure:** filename for temperature gradient and cooling rate
- **profiling:** caliper configuration
- **restart:** restart filename
- **units:** choose non-default units of input entries
- **verbose_output:** turn on/off the verbosity

Geometry

- Describe the mesh used by the simulation
- Mesh can be 2D or 3D
- Box mesh can be generated from input file
- Load a mesh from mesh generator: hexahedra only, no hanging nodes
- Many formats supported: abaqus, gmsh, tecplot, exodus, etc.
- Need at least one active cells at the start of the simulation
- Describe the material deposition: length, width, height, time, etc.

Exercise 1

- Exercise is in the ***Exercises/01*** directory
- The instructions are in the input.info file
- Use https://adamantine-sim.github.io/adamantine/doc/input_file.html#geometry-required
- Run adamantine using: *adamantine –i input.info*
- The solution of the exercise is given in the solution.info file
- You can also try to load the gmsh file: *domain.msh*

Boundary and time stepping

- Boundary:
  - Thermal boundary condition supported are: adiabatic, radiative, and convective
  - A boundary can be both radiative and convective
  - A general boundary can be applied to the entire domain boundary
  - One can overwrite the boundary condition for any boundary ids
- Time stepping:
  - Set the time step size
  - Set the end time of the simulation or until the scan path ends

Exercise 2

- Exercise is in the ***Exercises/02*** directory
- The instructions are in the input.info file
- Use https://adamantine-sim.github.io/adamantine/doc/input_file.html#boundary-required and https://adamantine-sim.github.io/adamantine/doc/input_file.html#time_stepping-required
- Run adamantine using: *adamantine –i input.info*
- The solution of the exercise is given in the solution.info file

Heat Source

- Adamantine supports an arbitrary number of heat sources but the heat sources must be in the same z-plane
- Two types of sources are supported: goldak and electron beam
- Sources must be enumerated consecutively starting at zero
- Two scan path formats are supported: segment format and event format
- Segment format is similar to the format used by AdditiveFoam but the power coefficient multiplies the maximum power set in the input file

Exercise 3

- Exercise is in the ***Exercises/03*** directory
- The instructions are in the input.info file
- Use https://adamantine-sim.github.io/adamantine/doc/input_file.html#sources-required
- Run adamantine using: *adamantine –i input.info*
- The solution is given in the solution.info file

Material Properties

- Materials properties can be set for powder, solid, and liquid state
- Adamantine solves the anisotropic heat equation → the thermal conductivity needs to be given for x, y, and z (follows deposition axis)
- It is not necessary to set all the material properties
- Material properties can be given as a polynomial of the temperature or as a table
- Important:
  - Define a material for each material id in the mesh
  - Material ids are consecutive numbers starting at 0

Exercise 4

- Exercise is in the ***Exercises/04*** directory
- The instructions are in the input.info file
- Use https://adamantine-sim.github.io/adamantine/doc/input_file.html#materials-required
- Run adamantine using: *adamantine –i input.info*
- The solution is given in the solution.info file

Ensemble simulation

- Adamantine can run multiple simulations with different input values described by a normal distribution
- Standard deviations must be set in the ensemble section
- Mean values are set in the regular sections
- This functionality is applicable only to scalar variables
- This does not work for temperature-dependent variables

Exercise 5

- Exercise is in ***Exercises/05*** directory
- The instructions are in the input.info file
- Use https://adamantine-sim.github.io/adamantine/doc/input_file.html#ensemble-optional
- Run adamantine using: *adamantine –i input.info*
- The solution is given in the solution.info file