

# LA-UR-23-28515

Approved for public release; distribution is unlimited.

**Title:** Generating MCNP Input Files for Unstructured Mesh Geometries

**Author(s):** Armstrong, Jerawan Chudoung  
Kelley, Karen Corzine

**Intended for:** Report

**Issued:** 2023-07-26



Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC for the National Nuclear Security Administration of U.S. Department of Energy under contract 89233218CNA000001. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

# Generating MCNP Input Files for Unstructured Mesh Geometries

Jerawan Armstrong and Karen Kelley

## 1 Introduction

The Los Alamos National Laboratory's (LANL) Monte Carlo N-Particle (MCNP)<sup>1</sup> transport code version 6.3 (also known as MCNP6.3) has the capability for tracking particles on unstructured mesh (UM) geometry models embedded into constructive solid geometry (CSG) cells [1, 2]. This feature has been developed for performing calculations of complex geometry models because manually creating CSG models is time-consuming and error-prone as the complexities of geometries increase. A UM geometry model is a collection of finite elements representing a solid geometry. The first step of the MCNP UM calculation is using other software packages to create a finite element mesh representation of a solid 3D geometry because the MCNP code cannot be used to generate a UM model. Computer-aided design (CAD) software is typically used to create a solid geometry model, which is later imported into mesh generation software to create a UM model. Some mesh generation software packages may also be used to create solid geometries and thus CAD files are not needed.

The MCNP UM feature was originally designed for models generated by the Abaqus/CAE software suite [3]. The MCNP code version 6.0 and later can process UM models formatted as Abaqus input files. Starting with a 6.3 version, the MCNP code can process HDF5 mesh input files. We only focus on the UM models formatted as Abaqus input files in this report since currently no external software can be used to generate HDF5 mesh input files for MCNP UM calculations. The MCNP code version 6.3 can be used to convert the Abaqus mesh input files into the HDF5 mesh input files, but this option is typically used by the MCNP code development team to test the HDF5 mesh input file feature. Several software packages (such as Abaqus, Attila4MC, or Cubit) can be used to create the Abaqus input files for MCNP UM calculations [4, 5].

An MCNP UM calculation using an Abaqus model requires two input file types: MCNP and Abaqus input files. The Abaqus input files needed for MCNP UM calculations must have the correct Abaqus syntax and meet the additional requirements by the MCNP code. The MCNP code can process only Abaqus input files that make use of part and assembly definitions, where elements in each part must be grouped into one or more element sets (i.e., elset) using \*Elset keyword lines with specified naming

---

<sup>1</sup>MCNP<sup>®</sup> and Monte Carlo N-Particle<sup>®</sup> are registered trademarks owned by Triad National Security, LLC, manager and operator of Los Alamos National Laboratory for the U.S. Department of Energy. Any third party use of such registered marks should be properly attributed to Triad National Security, LLC, including the use of the ® designation as appropriate. Any questions regarding licensing, proper use, and/or proper attribution of Triad National Security, LLC marks should be directed to [trademark@lanl.gov](mailto:trademark@lanl.gov). For the purposes of visual clarity, the registered trademark symbol is assumed for all references to MCNP within the remainder of this report.

formats. The MCNP and Abaqus input files required for MCNP UM simulations must be related; pseudo-cells in an MCNP input file must be constructed from mesh model data from an Abaqus input file.

For large complex UM models, it is tedious to manually create MCNP UM input files. The *um\_pre\_op* (unstructured mesh pre operations) program with the `-m` option can be used to create a skeleton MCNP input file from an Abaqus input file [6]. Since the *um\_pre\_op* program was written in Fortran and was not written for optimized performance, this program is a deprecated feature in the MCNP code version 6.3 and may be removed in the next release of the code. To improve calculation flow of multiphysics calculations, a Python3 code called *write\_mcnp\_um\_input* has been developed to generate an MCNP input file instead of using the *um\_pre\_op -m* option. This Python code was initially released to the public in 2020 [7]. We have updated this Python code for MCNP6.3 and it was used to generate the MCNP input files used to verify the MCNP6.3 code [8, 9]. The *write\_mcnp\_um\_input* code is included with the MCNP6.3 code package which will be released to the public through the Radiation Safety Information Computational Center (RSICC) at Oak Ridge National Laboratory. This report is a revision of LA-UR-20-27139 report [7].

## 2 Abaqus Input File

An Abaqus input file is an ASCII file defining a finite element model. Three line types are used in an Abaqus input file: comment, keyword, and data lines. This section provides a brief description of the Abaqus keyword and data lines required to construct UM models for MCNP simulations. We highly recommend that the users should be familiar with the Abaqus input file format described in the MCNP manual [2] before creating the Abaqus input files for MCNP UM calculations. An example of an Abaqus input file that the MCNP code can process is shown in Appendix A.

Comments in an Abaqus input file begin with an asterisk in columns 1 and 2 (\*\*) and they are ignored by the MCNP code. Keyword lines must begin with an asterisk (\*) in column 1 and may have parameters which appear as words or phrases separated by commas. The keyword must be followed by a comma (,) if it has parameters. Parameters in a keyword line can stand alone or have values. If the parameters have values, their values should not contain a comma. Most keyword lines require one or more data lines. If the data lines are required, they must immediately follow the keyword line. The data lines are used to provide numeric or alphanumeric entries for the associated keyword options. For the example file in Appendix A, “\*Part, name=BLOCK-B” is a keyword line that has a name parameter but does not have a data line; “\*Node” is a keyword line that does not have a parameter but has data lines; “\*Element, type=C3D8R” is a keyword line that has a type parameter and data lines.

An Abaqus input file often begins with the \*Heading keyword line which is used to define a title for the analysis. After the heading, the input file contains geometry model data described by elements and their nodes and can be defined by organizing into parts. The MCNP code recognizes only an Abaqus model whose parts are instanced to form an assembly. The keywords and associated data lines required for MCNP UM calculations include: \*Part, \*Node, \*Element, \*Elset, \*End Part, \*Assembly, \*Instance, \*End Instance, \*End Assembly, and \*Material. The \*Density keywords associated with \*Material keywords are not required by the MCNP code, but the *write\_mcnp\_um\_input* code uses the densities in an Abaqus input to write the

densities on pseudo-cell cards in an MCNP input file. If material densities are not in an Abaqus input file, the *write\_mcnp\_um\_input* code will put zero as densities on pseudo-cell cards and the user must manually update material densities.

The MCNP code makes use of the Abaqus definition of a set of elements to define an MCNP pseudo-cell. A keyword line starts with *\*Elset* is used to group elements into an element set. The *\*Elset* keyword lines can be at the Part level and Instance level, but only the *\*Elset* keyword lines at Part level are used by the MCNP code. One or more *\*Elset* keyword lines must be used to describe element sets in each part in an Abaqus input file. Restrictions are imposed on Elset and Material names in UM models generated for MCNP UM calculations. An elset name must contain **material**, **statistic**, or **material-statistic** and end with **-n** or **\_n** where n is a number for a statistic elset or a material number for a material elset. Each part must has **material** and **statistic** elset lines or a **material-statistic** elset line. A material name on *\*Material* keyword line should match with a material number in a material elset. The MCNP code uses the information in *\*Elset*, *\*Instance*, and *\*Material* lines to create MCNP pseudo-cell cards.

### 3 MCNP Unstructured Mesh Input Cards

An MCNP input file consists of a series of entries grouped into cell, surface, and data blocks. An entry in an MCNP input file is referred to as a card. An example of MCNP UM input file is shown in Appendix B.

An MCNP geometry is defined in the cell block, and one or more cell cards used to describe a UM model are called pseudo-cells. A pseudo-cell card is defined with a cell number followed by a material number, density, a null surface, and a universe assignment. A large complex UM geometry model is typically defined by many pseudo-cells grouped into a single universe. It is required that a universe containing a UM geometry model must be infinite in extent. Thus, a background cell is needed to create an infinite region of a specified material external to a UM geometry model defined in an Abaqus input file. A background cell card format is the same as the pseudo-cell card format, and void is typically used as a material in a background cell. In order to form a mesh universe that can be filled into a CSG cell using a fill card, the background cell and pseudo-cells of a UM model must be placed in the same universe. Note that other standard MCNP input values allowed in a cell card can also be entered in the pseudo-cell and background cell cards.

A mesh universe defined in an MCNP input file must match with a UM model defined in an Abaqus input file. The MCNP code uses an EMBED card to connect the UM data defined in the MCNP and Abaqus input files. An MCNP UM input file must contain an EMBED card in the data block. This card is mainly used to provide (i) the mesh geometry file type and name, (ii) correlation of mesh-cells from an Abaqus input file to pseudo-cells in an MCNP input file, (iii) designation of the background cell, and (iv) designating which output files will be produced. Each EMBED card contains several required keyword parameters that must have values. For the MCNP code version 6.3, the required keywords of EMBED cards include MESHGEO, MGEOIN, BACKGROUND, and MATCELL. A UM file type and name is specified by using the MESHGEO and MGEOIN keywords, respectively. The MATCELL keyword is for matching mesh-cells in an Abaqus input file to pseudo-cells in an MCNP input file. A background cell is defined by the BACKGROUND keyword and is truncated when a mesh universe is placed into a CSG fill cell. The MCNP code requires that a CSG

fill cell in which a mesh universe is placed must not crop the pseudo-cells. Manually defining the pseudo-cell and EMBED cards in an MCNP input file for a large complex UM model is tedious. The MCNP code can write tally-like results for each element in the mesh into a file called an EEOUT file; EEOUT is short for Elemental Edit OUTPUT. The MCNP code version 6.3 can produce an ASCII and HDF5 EEOUT file. The MEEOUT and HDF5FILE keywords on EMBED cards are respectively for an ASCII and HDF5 EEOUT filename. The ASCII EEOUT feature is a deprecated feature in the MCNP code version 6.3. Note that the MCNP code requires that all filenames on the EMBED cards must be in lowercase letters.

## 4 Generating MCNP Unstructured Mesh Input Files

We have developed a Python code (*write\_mcnp\_input.py*) to create an MCNP input file from an Abaqus input file; this code is distributed with the MCNP6.3 code package. The intent of this Python3 code is to make it easier for users to create MCNP UM input files. This program reads and processes the Abaqus input file to create the appropriate pseudo-cell cards, background cell, minimal CSG world to fill the mesh universe consisting of pseudo-cells and background cell, and the embed control card for the data section. A new algorithm is implemented in the *write\_mcnp\_input* code to create MCNP pseudo-cells. In addition, this Python code performs extensive error checking before generating the MCNP input file since Python is well-suited for file parsing and processing.

### 4.1 Command Line Usage

The command line *python write\_mcnp\_input.py --help* or *python write\_mcnp\_input.py -h* can be used to display the command line options:

```
usage: write_mcnp_um_input.py [-h] -i <file.inp> [-o <file.mcnp>]
                               [-mi <filename.info>] [-e <file.eeout>]
                               [-cc <cellcards.txt>] [-sc <surfacecards.txt>]
                               [-dc <datacards.txt>] [-b <material_number>]
                               [-l <len_conversion>] [-re <radius_extension>]
                               [-du <density_unit>] [-t <title>]
                               [-et <eeout_type>] [-wc] [-wm]
```

**\*\* Write MCNP Unstructured Mesh Input file \*\***

optional arguments:

```
-h, --help                show this help message and exit
-i <file.inp>, --input <file.inp>
                           Abaqus input filename to be processed
-o <file.mcnp>, --output <file.mcnp>
                           MCNP filename to be produced
-mi <filename.info>, --meshinfo <filename.info>
                           mesh information filename to be produced
-e <file.eeout>, --eeout <file.eeout>
                           EEOUT filename on an EMBED card
-cc <cellcards.txt>, --cellcards <cellcards.txt>
                           cell card filename (containing MCNP cell cards to be
                           included in an MCNP input file)
-sc <surfacecards.txt>, --surfacecards <surfacecards.txt>
                           surface card filename (containing MCNP surface cards to be
```

```

        included in an MCNP input file)
-dc <datacards.txt>, --datacards <datacards.txt>
    data filename (containing MCNP data cards file to be
    included in an MCNP input file)
-b <material_number>, --back <material_number>
    background material number
-l <len_conversion>, --length <len_conversion>
    a multiplication conversion factor to centimeter
-re <radius_extension>, --radext <radius_extension>
    a radius extension in centimeter of a sphere surface
    for a fill cell
-du <density_unit>, --densityunit <density_unit>
    a material density unit option: g/cc [default] or
    atoms/barn-cm
-t <title>, --title <title>
    MCNP title line
-et <eeout_type>, --eeouttype <eeout_type>
    EEOU file type: ascii or hdf5 [default]
-wc, --writecomments write pseudo-cell comments in an MCNP input file
-wm, --writemeshinfo write mesh information into a mesh information file

```

The only required option is the `--input` option. The other options are optional and the default values will be used if optional values are not provided.

The `--input` option is used to enter an Abaqus input file to be processed. An Abaqus input filename typically has an “.inp” file extension. We will refer to the basename of an Abaqus input file without a file extension as **basename** in this section. Since the MCNP code requires that an Abaqus input filename must be in lowercase letters, this requirement is checked and an error is thrown if an Abaqus filename is not in lowercase letters. An Abaqus input file must have the correct Abaqus syntax and the requirements imposed by the MCNP code. The `write_mcnp_um_input` code performs extensive checking to ensure that the Abaqus input file is in the correct format.

The name of the MCNP input to be created can be set with the `--output`. If this option value is not provided, the default filename is `basename.mcnp`. The degree to which a fully functional MCNP input file can be generated depends upon the completeness and correctness of the cell, surface, and data card files provided with the `--cellcards`, `--surfacecards`, and `--datacards` options. If the MCNP card filenames are provided, the MCNP cards contained in these files will be included in the MCNP input file to be produced. The cell cards will be placed after the fill cell in the cell block; the surface cards will be placed after the surface card used to define a fill cell in the surface block; the data cards will be placed after the EMBED card in the data block. The `write_mcnp_um_input` code does not check if the MCNP cards read from the files associated with the `--cellcards`, `--surfacecards`, and `--datacards` options have the correct MCNP card syntax since this checking can be performed by the MCNP code when the Abaqus and MCNP input files are used to run the MCNP code. If the file options for MCNP cards are not used, then the MCNP input file is not functional and it must be manually edited to be a functional input file. Information on the MCNP input setup can be founded in the MCNP manual [2].

The name of the mesh information file to be created can be set by the `--meshinfo` option. If this option value is not provided, the default filename is `basename.info`. When this option is used, the mesh information file contains the corresponding information between mesh information and MCNP pseudo-cells as well as the node, element, elset, instance, and material data read from the Abaqus input file. The

MCNP code does not use the mesh information file, but a user should examine this file to verify a UM model.

The name of the EEOUT file on the EMBED card can be set with the `--eeout` option. If this option value is not provided, the default filename is `basename.eeout.h5`. If this option is used, a filename must be in lowercase letters. The MCNP code can provide both ASCII and HDF5 EEOUT files. By default, the `write_mcnp_um_input` code uses the `hdf5file` option to produce an HDF5 EEOUT file type. The EEOUT file type option can be changed by the `--eeouttype` option, where the `hdf5` option value is for producing an HDF5 EEOUT file and the `ascii` option value is for producing an ASCII EEOUT file. Note that the `HDF5FILE` keyword is a new option in MCNP6.3.

If the `--length` option is not specified to supply a length conversion factor, the `write_mcnp_input` code will make the length conversion to 1.0. The length unit in the MCNP code is in centimeter. If the length values (i.e., node positions, translation and rotation points) in an Abaqus input file are not in centimeters, this option must be used to provide a length conversion factor to centimeters.

The surface of the fill cell is currently fixed to be a sphere with its center and radius computed from nodal data read from the Abaqus input file. The parameter values defined the spherical surface are computed by the following Python code:

```
# x/y/zmin = minimum x/y/z coordinate value in the model
# x/y/zmax = maximum x/y/z coordinate value in the model
x = xmax - xmin
y = ymax - ymin
z = zmax - zmin
diagonal = np.sqrt(x*x+y*y+z*z)
center    = 0.5*np.array([xmin+xmax, ymin+ymax, zmin+zmax])
radius    = 0.5*diagonal + radiusext
```

The `radiusext` value can be specified by the `--radext` option. If this option is not specified to supply a value then the `radiusext` is set to 1.0 centimeter. The `radiusext` value is needed so that the UM model is not clipped by the boundary of the fill cell.

Material densities must be placed in MCNP cell cards where density units can be either “g/cm<sup>3</sup>” or “atoms/barn-cm”. In MCNP input files, negative densities are mass densities (unit in g/cm<sup>3</sup>) while positive densities are atom densities (unit in atoms/barn-cm). The `--densityunit` option is for specifying the density unit. The density values in an Abaqus input file used for an Abaqus calculation must be positive. The `write_mcnp_um_input` code uses the density unit to convert the positive density values to the negative ones when the density unit is “g/cc”. If this option value is not provided, the default value is “g/cc”.

The `--background` option is for providing a background material number. If the `--background` option is not specified with a valid material number for the background cell, the `write_mcnp_input` code will make the background cell void.

The `--writecomments` option is for writing pseudo-cell commands in an MCNP input file; the option is logical (i.e., does not need the value) and by default the pseudo-cell comments are not written into the MCNP input file.

The `--title` option is for providing a title line in the MCNP input file. If the `*Heading` keyword has the value in the Abaqus input file, then the `write_mcnp_um_input` code will use the heading value as the title in the MCNP input file. This option can be used to supply the MCNP title line.



## 4.2 Python Function Signature

The following is the signature of the *write\_mcnp\_input* function:

```
def write_mcnp_um_input(filein, fileout=None, meshinfo=None, eeout=None,
                        cellcards=None, surfacecards=None, datacards=None,
                        bgmaterial=0, lenconv=1.0, radiusext=1.0,
                        denunit='g/cc', title='', eeouttype='hdf5',
                        writemeshinfo=False, writecomments=False)
```

where

- **filein** is an Abaqus input file to be processed. To meet an MCNP requirement, checking is performed to ensure that this filename is in lowercase letters. If this requirement is not satisfied, then an error is thrown. An Abaqus input file typically has an “.inp” filename extension. We referred to a basename of filein without a file extension as **basename** in this section.
- **fileout** is an MCNP input file to be produced. If this argument is not provided, the default filename is **basename.mcnp**.
- **meshinfo** is a file to be produced. This file contains mesh information read from an Abaqus input file. This file is not needed by the MCNP code. If this argument is not provided, the default filename is **basename.info**.
- **eeout** is an EEOUT filename placed on an EMBED card. If this argument is not provided, the default filename is **basename.eeout.h5**. If this argument is supplied, then checking is performed to ensure that this filename is in lowercase letters. If a filename entered is not in lowercase letters, an error is thrown.
- **cellcards** is a name of a file containing MCNP CSG cell cards. If the filename is provided, then the lines in this file will be appended into the MCNP cell block in the **fileout** file.
- **surfacecards** is a name of a file containing MCNP surface cards. If this filename is provided, then the lines in this file will be appended into the MCNP surface block in the **fileout** file.
- **datacards** is a name of a file containing MCNP data cards. If this filename is provided, then the lines in this file will be appended into the MCNP data block of the **fileout** file.
- **bgmaterial** is a material number for a background cell. The default material in a background cell is void (which is material number 0 in the MCNP code).
- **lenconv** is a value used to convert a length in an Abaqus input file into a centimeter. The default value is one. This value will be placed into the LENGTH keyword on an EMBED card.
- **radiusext** is used in defining a radius of a spherical surface used by a fill cell. See the `--radiusext` option information in Section 4.1 on how this value is used to define a spherical radius.
- **denunit** is a unit of material densities. The **denunit** option is used to specify the material densities read from the \*Density data lines in an Abaqus input file.

- `title` is used to define the MCNP title line. The default value is the data line of the `*Heading` keyword in an Abaqus input file.
- `eeouttype` is used to define the EEOUT file type. The option can be **hdf5** or **ascii**. The default value is **hdf5**.
- `writmeshinfo` is a logical option for writing the mesh information in the `meshinfo` file. By default the mesh information is not written into the file.
- `writcomments` is a logical option for writing the pseudo-cell comments in the `fileout` file. By default the comments are not written into the file.

Note that the `fileout` and `meshinfo` must be manually removed before the `write_mcnp_um_input` function is called if these files exist.

The `write_mcnp_um_input` function can be imported into a Python script. The following is a Python script example.

```
import os
# write_mcnp_um_input must be in Python path.
from write_mcnp_um_input import write_mcnp_um_input

abq_inp_ready = "godiva_linear_tet_ready.inp" # Abaqus input file
mcnp_inp      = "godiva_linear_tet.i"       # MCNP input file
mesh_info     = "godiva_linear_tet.info"    # Mesh information file
mcnp_datacards = "godiva_datacards.txt"     # MCNP data cards file

# remove existing files to be produced.
if os.path.isfile(mcnp_inp): os.remove(mcnp_inp)
if os.path.isfile(mesh_info): os.remove(mesh_info)

# Create an MCNP UM input file
write_mcnp_um_input(abq_inp_ready,
                    fileout=mcnp_inp,
                    meshinfo=mesh_info,
                    datacards=mcnp_datacards)
```

## Acknowledgement

We would like to thank Joel Kulesza for providing comments to a previous version of this report.

## References

- [1] Los Alamos National Laboratory. *MCNP - Monte Carlo radiation transport code*. <https://mcnp.lanl.gov>.
- [2] Joel A. Kulesza et al. *MCNP<sup>®</sup> Code Version 6.3.0 Theory & User Manual*. Tech. rep. LA-UR-22-30006, Rev. 1. Los Alamos, NM, USA: Los Alamos National Laboratory, Sept. 2022. URL: <https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-22-30006>.

- [3] Dassault Systemes. *Abaqus/CAE - finite element analysis software*. <https://www.3ds.com/products-services/simulia/products/abaqus/abaquscae/>.
- [4] Jerawan C. Armstrong et al. *MCNP Unstructured Mesh Overview, Improvement, and Verificaiton & Valdiation Testing*. Tech. rep. LA-UR-21-26436. Los Alamos, NM, USA: Los Alamos National Laboratory, July 2021. URL: <https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-21-26436>.
- [5] Jerawan C. Armstrong. *Using CUBIT to Create Unstructured Mesh Models for MCNP Simulations*. Tech. rep. LA-UR-23-21715. Los Alamos, NM, USA: Los Alamos National Laboratory, Feb. 2023. URL: <https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-23-21715>.
- [6] Roger L. Martz. *Unstructured Mesh User's Startup Guide*. Tech. rep. LA-UR-12-00795. Los Alamos, NM, USA: Los Alamos National Laboratory, Feb. 2012. URL: <http://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-12-00795>.
- [7] Jerawan C. Armstrong and Karen C. Kelley. *Generating MCNP Input Files for Unstructured Mesh Geometries*. Tech. rep. LA-UR-20-27139. Los Alamos, NM, USA: Los Alamos National Laboratory, Sept. 2020. URL: <https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-20-27139>.
- [8] Micky R. Dzur, Jerawan C. Armstrong, and Chelsea A. D'Angelo. *Cubit for MCNP Unstructured Mesh Analysis of Oktavian Benchmarks*. Tech. rep. LA-UR-22-29621. Los Alamos, NM, USA: Los Alamos National Laboratory, Sept. 2022. URL: <https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-22-29621>.
- [9] Esteban Gonzalez, Jerawan C. Armstrong, and James C. Tutt. *MCNP6.3 Unstructured Mesh Verification: GodivR and CANDU Models*. Tech. rep. LA-UR-22-33091. Los Alamos, NM, USA: Los Alamos National Laboratory, Dec. 2022. URL: <https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-22-33091>.

## Appendix A An Example of Abaqus Input File

```

*Heading
** Job name: block_hemi_v3_simplified_v5 Model name: block_hemi_v3_simplified_v4
** Generated by: Abaqus/CAE 2019
*Preprint, echo=NO, model=NO, history=NO, contact=NO
**
** PARTS
**
*Part, name=BLOCK-B
*Node
  1,      -5.,      0.,      10.
  4,     -20.,      0.,      10.
  9,      -5.,     10.,      10.
 10,     -5.,     10.,       0.
 11,     -20.,     10.,       0.
 12,     -20.,     10.,      10.
 13,     -20.,      0.,     20.
 18,      -5.,     10.,     20.
 19,     -5.,      0.,       5.
 23,    -15.,      0.,     10.
 24,    -10.,      0.,     10.
 35,     -5.,     10.,       5.
 36,    -10.,     10.,       0.
 37,    -15.,     10.,       0.
 38,    -20.,     10.,       5.
 39,    -15.,     10.,      10.
 40,    -10.,     10.,      10.
 41,    -20.,      5.,     10.
 42,     -5.,      5.,     10.
 43,    -20.,      5.,       0.
 44,     -5.,      5.,       0.
 45,    -20.,      0.,     15.

```

```

54,      -15.,      0.,      20.
55,      -20.,      5.,      20.
56,      -20.,      10.,     15.
57,       -5.,      5.,      20.
58,       -5.,      10.,     15.
59,      -10.,      10.,     20.
60,      -15.,      10.,     20.
61,      -10.,      0.,      5.
62,      -15.,      0.,      5.
71,      -10.,      10.,     5.
72,      -15.,      10.,     5.
73,      -10.,      5.,     10.
74,      -15.,      5.,     10.
75,      -10.,      5.,      0.
76,      -15.,      5.,      0.
77,       -5.,      5.,      5.
78,      -20.,      5.,      5.
83,      -15.,      0.,     15.
84,      -10.,      0.,     15.
87,      -20.,      5.,     15.
88,       -5.,      5.,     15.
89,      -15.,      5.,     20.
90,      -10.,      5.,     20.
91,      -15.,      10.,     15.
92,      -10.,      10.,     15.
95,      -10.,      5.,      5.
96,      -15.,      5.,      5.
99,      -15.,      5.,     15.
100,     -10.,      5.,     15.
*Element, type=C3D8R
13,  1, 19, 61, 24, 42, 77, 95, 73
15, 24, 61, 62, 23, 73, 95, 96, 74
20, 77, 44, 75, 95, 35, 10, 36, 71
21, 73, 95, 96, 74, 40, 71, 72, 39
24, 96, 76, 43, 78, 72, 37, 11, 38
37, 23, 83, 99, 74, 4, 45, 87, 41
38, 83, 54, 89, 99, 45, 13, 55, 87
39, 74, 99, 91, 39, 41, 87, 56, 12
41, 24, 84, 100, 73, 23, 83, 99, 74
44, 100, 90, 59, 92, 99, 89, 60, 91
47, 42, 88, 58, 9, 73, 100, 92, 40
48, 88, 57, 18, 58, 100, 90, 59, 92
*Elset, elset=SET_STATISTIC_004
13, 15, 20, 21, 24
*Elset, elset=SET_STATISTIC_001
37, 38, 39, 41, 44, 47, 48
*Elset, elset=SET_MATERIAL_001
13, 15, 20, 21, 24, 37, 38, 39, 41, 44, 47, 48
** Section: Section-1-SET_MATERIAL_001
*Solid Section, elset=SET_MATERIAL_001, material=ALUMINUM_001
,
*End Part
**
*Part, name=HEMI-A
*Node
 9,      -15.,      0.,      0.
10,      -20.,      0.,      0.
65, -14.2658482,      0.,  4.63525486
66, -12.1352549,      0.,  8.81677914
67, -8.81677914,      0., 12.1352549
69,      -17.5,      0.,      0.
70, -19.0211296,      0.,  6.18033981
71, -16.1803398,      0., 11.7557049
78, -14.4888878,  3.88228559,      0.
79, -12.9903812,      7.5,      0.
83, -19.3185158,  5.17638111,      0.
191, -16.651104,      0.,  5.41421127
192, -14.1600876,      0., 10.2832737
193, -10.2832737,      0., 14.1600876
218, -9.44908714,  4.87010145, 10.5828571
220, -11.0603456,  7.77711439,  6.49501753
221, -12.1539965,  4.35719395,  7.63513088
222, -13.8446913,  4.12033272,  4.0431881
223, -12.4165096,  7.64765692,  3.51334953
227, -16.9026642,  4.53362846,      0.
228, -15.1577482,  8.74622917,      0.
232, -15.9588022,  5.7864871, 10.5751219
233, -18.3967533,  5.52291918,  5.57286406
234, -14.4478245, 10.4146786,  9.09916687
235, -16.4513702, 10.2669563,  4.89305735
239, -12.3548412,  6.48197842, 14.3297539
241, -11.2937136, 11.4542313, 11.8849745
294, -13.3110504,  4.72851801,  8.81317425
295, -12.15802,  8.63381863,  7.4945159
301, -9.3094511,  9.32660103,  9.97525501
302, -10.0414009,  5.15912437, 12.0751715
303, -15.4972162,  4.57867861,  4.65281343
304, -13.8886414,  8.59171391,  4.04228067
*Element, type=C3D8R
147, 302, 193, 192, 294, 218, 67, 66, 221
149, 191, 69, 227, 303, 65, 9, 78, 222
150, 294, 303, 304, 295, 221, 222, 223, 220
151, 303, 227, 228, 304, 222, 78, 79, 223
160, 232, 233, 303, 294, 71, 70, 191, 192
161, 233, 83, 227, 303, 70, 10, 69, 191
162, 234, 235, 304, 295, 232, 233, 303, 294
171, 301, 241, 234, 295, 302, 239, 232, 294
*Elset, elset=SET_STATISTIC_012
147, 149
*Elset, elset=SET_STATISTIC_023
150, 171
*Elset, elset=SET_STATISTIC_034

```

```

151, 162
*Elset, elset=SET_STATISTIC_011
160, 161
*Elset, elset=SET_MATERIAL_001
147, 149
*Elset, elset=SET_MATERIAL_1008
150, 151, 160, 161, 162, 171
** Section: Section-2-SET_MATERIAL_001
*Solid Section, elset=SET_MATERIAL_001, material=ALUMINUM_001
** Section: Section-3-SET_MATERIAL_1008
*Solid Section, elset=SET_MATERIAL_1008, material=WATER_1008
,
*End Part
**
*Part, name=HEX_WEDGE1
*Node
1, 0., 23.8814831, 0.
2, 0., 25.8742371, 0.
3, 1.09208882, 23.5859795, 4.0757308
4, 0.549050272, 23.7657948, 2.04908371
5, 1.09813011, 25.5555992, 4.09827709
6, 0.552914023, 25.7522068, 2.06350327
7, 0.823185027, 23.5859795, 4.1384306
8, 0.413858265, 23.7657948, 2.08060598
9, 0.827738762, 25.5555992, 4.16132355
10, 0.416770637, 25.7522068, 2.09524751
11, 0.550756216, 23.5859795, 4.18340874
12, 0.276894003, 23.7657948, 2.10321879
13, 0.553802907, 25.5555992, 4.2065506
14, 0.278842568, 25.7522068, 2.11801958
15, 0.275968969, 23.5859795, 4.21047306
16, 0.138744071, 23.7657948, 2.11682534
17, 0.277495593, 25.5555992, 4.23376465
18, 0.139720425, 25.7522068, 2.13172174
19, 0., 23.5859795, 4.21950722
20, 0., 23.7657948, 2.12136745
21, 0., 25.5555992, 4.24284887
22, 0., 25.7522068, 2.1362958
*Element, type=C3D8R
1, 7, 8, 10, 9, 3, 4, 6, 5
2, 11, 12, 14, 13, 7, 8, 10, 9
3, 15, 16, 18, 17, 11, 12, 14, 13
4, 19, 20, 22, 21, 15, 16, 18, 17
*Element, type=C3D6
5, 2, 6, 10, 1, 4, 8
6, 2, 10, 14, 1, 8, 12
7, 2, 14, 18, 1, 12, 16
8, 2, 18, 22, 1, 16, 20
*Elset, elset=SET_MATERIAL_001, generate
1, 8, 1
*Elset, elset=SET_STATISTIC_002, generate
1, 8, 1
** Section: Section-3-SET_MATERIAL_001
*Solid Section, elset=SET_MATERIAL_001, material=ALUMINUM_001
,
*End Part
**
*Part, name=TET_ONLY
*Node
1, 0., 5., 10.
2, -3., 5., 10.
3, 0., 5., 5.
4, -3., 5., 5.
5, -6., 5., 5.
6, -6., 2.5, 10.
7, -3., 2.5, 10.
8, 0., 5., 7.5
9, -3., 5., 7.5
10, 0., 2.5, 5.
11, -3., 2.5, 5.
12, -6., 2.5, 5.
13, 1.25195348, 2.3700695, 7.90644789
14, -1.88319409, 1.72202432, 7.78357697
15, -4.05866098, 1.7532028, 7.15756607
*Element, type=C3D4
1, 15, 2, 14, 9
2, 14, 10, 4, 3
3, 14, 11, 4, 10
4, 15, 11, 12, 5
5, 9, 4, 14, 15
6, 8, 13, 1, 14
7, 8, 3, 13, 14
8, 15, 6, 7, 2
*Elset, elset=SET_STATISTIC_001, generate
1, 8, 1
*Elset, elset=SET_MATERIAL_001, generate
1, 8, 1
** Section: Section-4-SET_MATERIAL_001
*Solid Section, elset=SET_MATERIAL_001, material=ALUMINUM_001
,
*End Part
**
**
** ASSEMBLY
**
*Assembly, name=Assembly
**
*Instance, name=HEMI-A-1, part=HEMI-A
*End Instance
**
*Instance, name=BLOCK-B-1, part=BLOCK-B
0., 0., -25.

```

```

*End Instance
**
*Instance, name=TET_ONLY-1, part=TET_ONLY
    10.,      15.,      -15.
    10.,      15.,      -15.,      10.,      15., -13.9999999873119, 89.9999992730282
*End Instance
**
*Instance, name=HEX_WEDGE1-1, part=HEX_WEDGE1
    0.,      0.,      -22.
    0.,      0.,      -22.,      0., 1.00000001268805,      -22., 89.9999992730282
*End Instance
**
*End Assembly
**
** MATERIALS
**
*Material, name=ALUMINUM_001
*Density
2.7,
*Material, name=WATER_1008
*Density
1.0,

```

## Appendix B An Example of MCNP Unstructured Mesh Input File

```

No description
C
C Abaqus Input File: godiva_linear_tet_ready.inp
C
C High-Level Mesh Information:
C   Number of Parts:      1
C   Number of Instances:  1
C   Number of Materials:  1
C
C Details of mesh information are in file: godiva_linear_tet.info
C
C PSEUDO CELLS
1      1  -18.740000      0  u=1  $ HEU-1
2      0                    0  u=1  $ background
C
C LEGACY CELLS
3      0                    -999  fill=1 $ fill cell
4      0                    999
C
C Surface Cards
C
999 sph    2.249000E-03    7.770000E-04    0.000000E+00    1.613624E+01
C
C Data Cards
C
embed1 meshgeo=abacus
      mgeoin=godiva_linear_tet_ready.inp
C      meeout=godiva_linear_tet_ready.eeout
      hdf5file=godiva_linear_tet_ready.eeout.h5
      length=1.000000E+00
      background=2
      matcell=1 1  $ HEU-1
C

```

```
C material
m1 92234.80c 4.9184e-04 92235.80c 4.4994e-02
    92238.80c 2.4984e-03
C
mode n
imp:n 1 1 1 0
kcode 10000 1.0 20 120
sdef erg=d1 rad=d2 pos=0.0 0.0 0.0
sp1 -3
si2 0.0 8.7407
sp2 -21 2
totnu
C
embee4:n embed=1 errors=yes
print
```