ADDER v1.0.1 Release Notes

The Advanced Dimensional Depletion for Engineering of Reactors (ADDER) software performs lifecycle neutronics analysis of a nuclear reactor. This includes performing neutronics calculations with external neutronics solvers, inventory transmutation with an external or internal depletion solver, modifying the neutronics model consistent with fuel management strategies used in a reactor design, and providing a simply interface for critical position/rotation searching.

The ADDER software is written for Python 3.7 or greater [1]. The neutronics interface is currently implemented for MCNP5 version 1.60 [2] and MCNP6 version 6.2 [3]. The current depletion solver is ORIGEN2 (version 2.2) [4] or an internal solver using the Chebyshev Rational Approximation Method (CRAM) [5].

Interested users may contact Argonne National Laboratory for availability and licensing conditions. Please contact Erik Wilson (erikwilson@anl.gov) for information on how to obtain the code.

Modifications in ADDER v1.0.1

ADDER Version 1.0.1 is the second release of ADDER. The following changes have been made since ADDER v1.0.0 was released:

1. The MIT Open Source Software license is now distributed with the ADDER Software.
2. ADDER v1.0.0 may unintentionally deplete materials with incorrect fluxes because of an error in the order that it processed MCNP tallies. ADDER v1.0.1 corrects this error.
3. ADDER v1.0.0 did not provide an error to the user when an element or isotope is present in the initial neutronics model that does not have neutronics cross sections available in the neutronics library file. ADDER v1.0.1 now provides such an error.
4. If an isotope is present in the neutronics model but is not present in the depletion library, then ADDER should mark that isotope as non-depleting. ADDER v1.0.0 only did this for the first isotope encountered in a depleting material and all others were marked as depleting. ADDER v1.0.1 correctly applies the same non-depleting status to all instances of such isotopes in depleting materials.
5. ADDER v1.0.0 crashes when the “all_depleting” value is used as an argument to the “write_depletion_lib” operation. ADDER v1.0.1 has (1) renamed this value to “all_modified” to more accurately reflect that only non-default libraries are written and (2) now no longer crashes when this is used.
6. ADDER v1.0.1 now raises an error when a user provides a material that is both: (1) marked as depleting based on the ADDER input file, and (2) the neutronics inputs file does not have a default neutronics cross section library defined.
7. ADDER v1.0.0 would erroneously not apply the non_depleting_isotopes input to the model. In ADDER v1.0.1, isotopes explicitly marked by the user as ‘’non_depleting_isotopes’’ are now read correctly and these isotopes not depleted.
8. ADDER v1.0.1 provides a user-understandable warning to the log file when user-supplied MCNP inputs contain zero mass and/or atom fractions on material cards.
9. ADDER v1.0.1 now provides a user-facing option where the user can specify if all isotopes explicitly entered by the user in the initial neutronics model are intended to be included in all time steps of the model regardless of their reactivity importance and the value of the

"neutronics_reactivity_threshold" parameter. This "apply_reactivity_threshold_to_initial_inventory" input parameter can be set either globally or on a per-material basis.

10. ADDER v1.0.0 applied incorrect logic for the reactivity threshold implementation in the case of non-fissionable materials. This logic is now corrected in ADDER v1.0.1 by using a method which is only based on the fraction of neutron absorption of each isotope in that material. The method for fissionable materials is unchanged.
11. ADDER v1.0.1 now raises an error whenever an isotope is included multiple times in a material definition, whether they be defined with neutron libraries (e.g., nlibs) or otherwise.
12. ADDER v1.0.0 would incorrectly parse valid real numbers in MCNP input files (e.g., -1+1 to represent –10) and exponential notation used for valid integer input (e.g., 1.0E6 for the integer 1,000,000). ADDER v1.0.1 now correctly processes these valid MCNP input formats.
13. ADDER v1.0.0 would incorrectly not ignore comments in the directory section of an MCNP xsdir file, potentially leading to an error. ADDER v1.0.1 now correctly recognizes and ignores these comments.
14. Certain cards in MCNP input may contain single or multiple *jump* entries. ADDER can already handle such jump entries, however, ADDER v1.0.0 was missing this *jump* handling for MCNP's kcode and print cards. ADDER v1.0.1 will correctly handle such kcode and prdmp cards with these jumps.
15. ADDER v1.0.0 execution would fail when performing a stochastic volume calculation on models initially provided by the user with lost, run, or nps cards. ADDER v1.0.1 now executes such cases successfully by removing these cards and replacing when done with the stochastic volume calculation.
16. ADDER v1.0.1 will shorten the title provided by the user in the initial MCNP input file to make sure that the total length of the final title for each depletion step, including the ADDER-generated information about the step, doesn't exceed 80 characters. ADDER v1.0.0 would not perform such truncations nor provide any warning.
17. ADDER v1.0.0 and its post-processing adder_extract_materials.py script incorrectly extracts the *num_copies* field from material data in ADDER HDF5 files such that a failure was encountered if a material is processed which has been copied more than twice during fuel management operations. ADDER v1.0.1 now correctly processes this parameter and avoids such failure.
18. ADDER automatically creates clones of cells filled with depleting materials that are present in multiple locations in a model. When this is done, ADDER cannot determine how the volume of the divided cell can change and in this case the undivided cell volumes will remain. ADDER v1.0.1 now provides a warning in such a case so the user can be aware that they should check the volumes and potentially run a stochastic volume calculation.
19. MCNP cells can be defined using the complement operator (#n) where n is a previously defined cell number. If cell n is cloned and given a new id such as n+1, ADDER v1.0.0 would erroneously not update the id of such a complement operator to be #n+1 as necessary for creating copies of shuffled universes. In ADDER v1.0.1, if a complement operator is used in a shuffled universe, then a warning will be provided telling the user that they should check that the region definitions are correct.
20. Users do not have to provide material volumes in the ADDER/MCNP input for depleting materials if these volumes are automatically calculated by MCNP. ADDER v1.0.0 will crash if a user does not give a volume for a material and then shuffles that material prior to an MCNP calculation. Such a

situation may be perfectly acceptable and therefore this crash will be corrected and instead a warning relayed explaining that a shuffled material cannot be checked for consistent volumes with items in the chain of shuffled materials.

Software Installation

The generalized process for installing ADDER is documented in section 1.1 of the ADDER User's Manual [6] and any modifications to support the local environments.

Software Verification

The ADDER test suite is fully automated using the widely used testing framework, pytest [7]. This test suite is executed after ADDER installation by running the `pytest` command in the 'tests/' directory of the ADDER distribution. This test suite will automatically execute the tests, compare results, and report the number of passing and failing tests. All tests shall pass.

When using this test suite, the ORIGEN2, MCNP, MPIRUN executables to be used for testing shall be accessible in the environment as the name 'origen2.EXE', 'mcnp.EXE', and 'mpirun'. This may be accomplished through environment variable modifications and symbolic links as needed. Further, the MPI libraries required for 'mpirun' shall also be accessible via the execution environment. This requirement only applies for automated testing and not for typical ADDER usage. For completeness, this test suite should be repeated for every MPI, MCNP, and ORIGEN2 combination to be used with ADDER.

Software Installation and Execution

General installation, user input, and execution instructions are thoroughly documented in the ADDER User Guide [6].

Contact Information

Technical questions and bug reports: Kyle Anderson (kanderson@anl.gov)
All other questions: Joshua Rudolph (jrudolph@anl.gov)

References

1. Python Software Foundation, *Python Language Reference, version 3.7*. Available at https://www.python.org.
2. X-5 Monte Carlo Team, *MCNP- A General Monte Carlo N-Particle Transport Code, Version 5*, *Volume I* (LA-UR-03-1987), *Volume II* (LA-CP-03-0245), *Volume III* (LA-CP-03-0284), Los Alamos National Laboratory, Los Alamos, USA April, 2003 (Revised Feb. 2008).
3. C.J. Werner (editor), *MCNP User's Manual - Code Version 6.2*, LA-UR-17-29981, Los Alamos National Laboratory, Los Alamos, USA (2017).
4. A.G. Croff, *A User's Manual for the ORIGEN2 Computer Code*, ORNL/TM-7175, Oak Ridge National Laboratory, Oak Ridge, USA, July 1980.
5. M. Pusa, "Higher-Order Chebyshev Rational Approximation Method and Application to Burnup Equations", Nucl. Sci. Eng., 182:3, 297-318 (2016).
6. K. Anderson, V. Mascolino, and A. G. Nelson, *User Guide to the Advanced Dimensional Depletion for Engineering of Reactors (ADDER), Release Version 1.0.1*, ANL/RTR/TM-21/8 Rev. 1, Argonne National Laboratory, Lemont, USA, April 2022.
7. Krekel et al., https://github.com/pytest-dev/pytest.