

LA-UR-22-30831

Approved for public release; distribution is unlimited.

Title: Easy_PERT: a Python tool for writing PERT cards and parsing PERT card results

Author(s): Clark, Alexander Rich

Intended for: 2022 MCNP User Symposium, 2022-10-17/2022-10-21 (Los Alamos, New Mexico, United States)

Issued: 2022-10-14



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.



Easy_PERT: a Python tool for writing PERT cards and parsing PERT card results

Alexander R. Clark

Monte Carlo Codes Group (XCP-3)

2022 MCNP® User Symposium
October 17-21, 2022

LA-UR-22-xxxxx

PERT card overview

- Uses differential operator method¹ to compute first- and second-order tally variations due to density, composition, and reaction cross-sections
- Can have multiple PERT cards in one MCNP input deck to study tally variations for several sets of nuclides, reactions, and energy ranges
- METHOD option tells MCNP to calculate either the perturbed tally (METHOD=-1, -2, -3) or the change in the unperturbed tally (METHOD=1, 2, 3)

1. J. A. Kulesza et al., "MCNP® Code Version 6.3.0 Theory & User Manual," LA-UR-22-30006, Rev. 1, Section 2.12.1, (2022)



Sensitivity analysis via the PERT card

- The PERT card can be used¹ to compute tally sensitivities with respect to nuclear data
 - For each nuclide, reaction, and energy range, write one PERT card with METHOD=2
 - Parse the MCTAL file for the unperturbed tally (c_0) and tally variation (Δc_1)
 - Calculate the first-order sensitivity as $S_{c,\sigma_x} = \frac{\Delta c_1}{p \cdot c_0}$
 - p is an arbitrary perturbation size (typically choose $p = 1$)
- PERT cards can also be used¹ to calculate the statistical uncertainty of the sensitivities
 - Treating c_0 and Δc_1 as independent, only METHOD=2 is needed
 - To account for correlation between c_0 and Δc_1 , need METHOD=2, -2

1. J. Favorite, "Using the MCNP Taylor series perturbation feature (efficiently) for shielding problems," EPJ Web of Conferences 153, 06030 (2017)



Difficult to use PERT card capability by hand

- User must add the following to their MCNP input deck
 - Create a “fictitious” material that is copied from an existing material
 - Multiply original nuclide fraction by $1 + p$
 - Multiply original material density by $\frac{\sum_i \omega_i'}{\sum_i \omega_i^0}$
 - Write PERT cards for all nuclides, reactions, energy ranges, and methods
- User must parse the output MCTAL file for tally results and their corresponding uncertainties to calculate
 - Tally sensitivities to nuclear data
 - Uncertainty in sensitivities due to Monte Carlo statistics



Easy_PERT simplifies the process

- Python tool that is run from the command line
 - Uses Faust¹ to modify the MCNP input deck and write outputs in a JSON format
 - Uses MCNPTools² to parse the output MCTAL file
- User provides an existing MCNP input deck and specifies nuclides, reactions, energy ranges, and methods
- Tool handles
 - Calculating modified fractions and material densities
 - Writing a modified MCNP input deck
 - Parsing the output MCTAL file into a JSON format

1. W. Haeck, A. R. Clark, and M. W. Herman, "Calculating the Impact of Nuclear Data Changes with Crater," Trans. Am. Nucl. Soc., Vol. 123, No. 1, p. 723-726 (Nov. 2020)

2. C. R. Bates et al., "The MCNPTools Package: Installation and Use," LA-UR-22-28935, (Aug. 2022)



Easy_PERT demo

- Start with an existing MCNP input deck
- Main Python script is used to perform each of the steps

```
(base) arclark@pp: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ conda activate faust2_
(faust2) arclark@pp: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ clear
(faust2) arclark@pp: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ cat mcnp.FUND-NCERC-PU-HE3-MULT-001-007.r1.inp
very simplified MCNP input deck for FUND-NCERC-PU-HE3-MULT-001-001.keff to compare keff values against SENS3G result.
c cell cards.
100 100 -1.96039000e+01 -100 imp:n=1 $ BeRP ball.
910 0 +100 -910 imp:n=1 $ Gap between BeRP ball and 0.5" nickel reflector.
200 200 -8.84378000e+00 +910 -200 imp:n=1 $ 0.5" nickel reflector.
920 0 +200 -920 imp:n=1 $ Gap between 0.5" and 1.0" nickel reflector.
300 300 -8.79058000e+00 +920 -300 imp:n=1 $ 1.0" nickel reflector.
930 0 +300 -930 imp:n=1 $ Gap between 1.0" and 1.5" nickel reflector.
400 400 -8.74904000e+00 +930 -400 imp:n=1 $ 1.5" nickel reflector.
940 0 +400 -940 imp:n=1 $ Gap between 1.5" and 2.0" nickel reflector.
500 500 -8.71850000e+00 +940 -500 imp:n=1 $ 2.0" nickel reflector.
950 0 +500 -950 imp:n=1 $ Gap between 2.0" and 2.5" nickel reflector.
600 600 -8.72040000e+00 +950 -600 imp:n=1 $ 2.5" nickel reflector.
960 0 +600 -960 imp:n=1 $ Gap between 2.5" and 3.0" nickel reflector.
700 700 -8.79332000e+00 +960 -700 imp:n=1 $ 3.0" nickel reflector.
970 0 +700 -999 imp:n=1 $ void between assembly and problem boundary.
999 0 +999 imp:n=0 $ outside of problem.

c surface cards.
100 so 3.79380000e+00 $ BeRP ball.
910 so 3.85672000e+00 $ Gap between BeRP ball and 0.5" nickel reflector.
200 so 5.11937000e+00 $ 0.5" nickel reflector.
920 so 5.13207000e+00 $ Gap between 0.5" and 1.0" nickel reflector.
300 so 6.38937000e+00 $ 1.0" nickel reflector.
930 so 6.48207000e+00 $ Gap between 1.0" and 1.5" nickel reflector.
400 so 7.65937000e+00 $ 1.5" nickel reflector.
940 so 7.67207000e+00 $ Gap between 1.5" and 2.0" nickel reflector.
500 so 8.92937000e+00 $ 2.0" nickel reflector.
950 so 8.94207000e+00 $ Gap between 2.0" and 2.5" nickel reflector.
600 so 1.01993700e+01 $ 2.5" nickel reflector.
960 so 1.02120700e+01 $ Gap between 2.5" and 3.0" nickel reflector.
700 so 1.14693700e+01 $ 3.0" nickel reflector.
999 so 1.50000000e+01 $ problem boundary.

c data cards.
print 10 40 85 110
c fset1 xs
c iso = 94239
c pxn = -7
c eneg =
c 1.000000000000000e-11 2.96937332818714E-09 1.03641312841130E-08
c 2.99896082485731E-08 4.09909343950883E-08 4.94445050193864E-08
c 7.19413303032538E-08 1.04674017947447E-07 1.52000000000000E-07
c 2.21594897733660E-07 2.51099915574398E-07 2.84533480898340E-07
c 3.22418673725673E-07 3.65348221372105E-07 4.14000000000000E-07
```

```
(faust2) arclark@pp: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ ./easy_pert.py -h
usage: easy_pert.py [-h] {write,parse,combine,sensitivity} ...

"Easy PERT" is a Python tool designed to make the MCNP code PERT card more accessible. The typical workflow is:

 1. Writing all required PERT card entries to an existing set of input decks ("write")
 2. performing the MCNP code calculation
 3. parsing the output MCTAL files ("parse")

Additional utility functions, such as:

 * Combining JSON files ("combine")
 * Computing sensitivities ("sensitivity") are also available.

For more information regarding each command, type "\.easy_pert.py <command> -h."
=====
optional arguments:
  -h, --help            show this help message and exit

commands:
  {write,parse,combine,sensitivity}
    write                Tool for writing PERT cards to an existing MCNP code input deck.
    parse                Tool for parsing PERT card results from MCNP code MCTAL files.
    combine              Tool for combining JSON files.
    sensitivity          Computes sensitivities via the MCNP code PERT card method.
(faust2) arclark@pp: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$
```



Easy_PERT demo

- Use the “write” command to add required PERT card entries to your existing input deck

```
arclark@pc: /mnt/c/Users/ /staff_xcp3/projects/easy_pert
(faust2) arclark@pc: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ ./easy_pert.py write -h
usage: easy_pert.py write [-h] -e E -mcn MCN -ccn CCN [CCN ...] -n N -rxn RXN [-u {eV,keV,MeV}] [-d {n,r,, }] [-fmcn FMCN]
                        [-p P] [-methods {1,-1,2,-2,3,-3} [{1,-1,2,-2,3,-3} ...]] -i I [-r R]

Tool for writing PERT cards to an existing MCNP code input deck.

optional arguments:
  -h, --help                show this help message and exit
  -u {eV,keV,MeV}          Group structure units.
  -d {n,r,, }              Delimiter for group structure file. The `n` and `r` options will be escaped, as expected.
  -fmcn FMCN              Material card number for the fictitious material defining the PERT card perturbation. Must not match an
                          existing material card number.
  -p P                    Relative perturbation size. The default size is recommended but another value can be chosen at the
                          user's discretion.
  -methods {1,-1,2,-2,3,-3} [{1,-1,2,-2,3,-3} ...]
                          Desired PERT card method(s). For example, "-m 1 -1" will perform the first- and second-order
                          perturbations and return the difference in the unperturbed tally and the perturbed tally, respectively,
                          in the MCTAL file. Please see the MCNP manual for more detail.
  -r R                    Random seed used in the MCNP calculation. If no seed is provided, MCNP will use a default value.

Required arguments:
  -e E                    Group structure path. Must be a text file with values separated by delimiters allowed by the `d` flag.
  -mcn MCN                Material card number of the material for which the PERT cards apply.
  -ccn CCN [CCN ...]     Cell card number(s) of the cell(s) for which the PERT cards apply. All cells must contain the same
                          material.
  -n N                    Complete ZAID and library extension for which the PERT cards apply (e.g. "94239.00c").
  -rxn RXN                MT number for which the PERT cards apply. Please see the MCNP manual for a listing of valid entries.
  -i I                    MCNP input deck path.

(faust2) arclark@pc: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ ./driver.sh write
+ input_deck=mcnp.FUND-NCERC-PU-HE3-MULT-001-007.r1.inp
+ group_structure=group_structure_51_wim_MeV.inp
+ command=write
+ [[ write == \w|\r|\i|\t|\e ]]
+ ./easy_pert.py write -i mcnp.FUND-NCERC-PU-HE3-MULT-001-007.r1.inp -e group_structure_51_wim_MeV.inp -mcn 100 -ccn 100 -n 94239.
00c -rxn 18 -u MeV -d n -fmcn 9979 -p 1. -methods 2
Writing PERT cards to MCNP input decks.
(faust2) arclark@pc: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$
```



Easy_PERT demo

- The tool takes care of making the fictitious material, calculating the fictitious density, and writing all of the relevant PERT cards

```
160 c
161 c materials cards.
162 c
163 c BeRP ball.
164 m100 6012.00c 4.54853000e-03
165 11023.00c 2.58307000e-04
166 31069.00c 6.85800000e-04
167 31071.00c 4.55256000e-04
168 73181.00c 2.33361000e-03
169 74182.00c 6.08751000e-04
170 74183.00c 3.28717000e-04
171 74184.00c 7.03705000e-04
172 74186.00c 6.53009000e-04
173 92235.00c 8.51567000e-04
174 92236.00c 1.97532000e-04
175 94238.00c 1.54199000e-04
176 94239.00c 9.26002000e-01
177 94240.00c 5.83805000e-02
178 94241.00c 5.61676000e-04
179 94242.00c 2.73395000e-04
180 95241.00c 2.52674000e-03
181 c 0.5" nickel reflector.
182 m200 5011.00c 1.30276000e-04
```

```
272 c fictitious material for use with the PERT card
273 c using p=1.0, nuclide=94239.00c
274 m9979
275 6012.00c 0.00454853
276 11023.00c 0.000258307
277 31069.00c 0.0006858
278 31071.00c 0.000455256
279 73181.00c 0.00233361
280 74182.00c 0.000608751
281 74183.00c 0.000328717
282 74184.00c 0.000703705
283 74186.00c 0.000653009
284 92235.00c 0.000851567
285 92236.00c 0.000197532
286 94238.00c 0.000154199
287 94239.00c 1.852004
288 94240.00c 0.0583805
289 94241.00c 0.000561676
290 94242.00c 0.000273395
291 95241.00c 0.00252674
292 pert1:n cell=100 mat=9979 rho=-37.76580849855271 rxn=18 erg=1.000000e-11, 2.969370e-09 method=2
293 pert2:n cell=100 mat=9979 rho=-37.76580849855271 rxn=18 erg=2.969370e-09, 1.036410e-08 method=2
294 pert3:n cell=100 mat=9979 rho=-37.76580849855271 rxn=18 erg=1.036410e-08, 2.998960e-08 method=2
295 pert4:n cell=100 mat=9979 rho=-37.76580849855271 rxn=18 erg=2.998960e-08, 4.099090e-08 method=2
296 pert5:n cell=100 mat=9979 rho=-37.76580849855271 rxn=18 erg=4.099090e-08, 4.944450e-08 method=2
297 pert6:n cell=100 mat=9979 rho=-37.76580849855271 rxn=18 erg=4.944450e-08, 7.194130e-08 method=2
```



Easy_PERT demo

- After performing the MCNP calculation to get a MCTAL file, use the “parse” command to parse the tally information into a JSON format

```
arclark@p /mnt/c/Users/raaf_xcp3/projects/easy_pert
(faust2) arclark@p ~: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ ./easy_pert.py parse -h
usage: easy_pert.py parse [-h] -e E -mcn MCN -ccn CCN [CCN ...] -n N -rxn RXN [-u {eV,keV,MeV}] [-d {n,r,,}] [-fmcn FMCN]
                        [-p P] [-methods {1,-1,2,-2,3,-3} [{1,-1,2,-2,3,-3} ...]] [-mctal MCTAL [-t T]] [--pert_only]

Tool for parsing PERT card results from MCNP code MCTAL files.

optional arguments:
  -h, --help            show this help message and exit
  -u {eV,keV,MeV}      Group structure units.
  -d {n,r,,}           Delimiter for group structure file. The 'n' and 'r' options will be escaped, as expected.
  -fmcn FMCN          Material card number for the fictitious material defining the PERT card perturbation. Must not match an
existing material card number.
  -p P                Relative perturbation size. The default size is recommended but another value can be chosen at the
user's discretion.
  -methods {1,-1,2,-2,3,-3} [{1,-1,2,-2,3,-3} ...]
Desired PERT card method(s). For example, "-m 1 -1" will perform the first- and second-order
perturbations and return the difference in the unperturbed tally and the perturbed tally, respectively,
in the MCTAL file. Please see the MCNP manual for more detail.

Required arguments:
  -e E                Group structure path. Must be a text file with values separated by delimiters allowed by the 'd' flag.
  -mcn MCN           Material card number of the material for which the PERT cards apply.
  -ccn CCN [CCN ...] Cell card number(s) of the cell(s) for which the PERT cards apply. All cells must contain the same
material.
  -n N              Complete ZAIID and library extension for which the PERT cards apply (e.g. "94239.00c").
  -rxn RXN         RT number for which the PERT cards apply. Please see the MCNP manual for a listing of valid entries.
  -mctal MCTAL     MCNP MCTAL file containing PERT card results.
  -t T            Tally number for which the PERT cards results are desired.
  --pert_only      Flag for writing only the PERT card results to file. Useful if the unperturbed and perturbed tally
results are calculated in separate runs with different random seeds to avoid correlations in the
statistical uncertainties.

(faust2) arclark@p ~: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$ ./driver.sh parse
+ input_deck=mcnp.FUND-NCERC-PU-HE3-MULT-001-007.r1.inp
+ group_structure=group_structure_51_wim_MeV.inp
+ command=parse
+ [[ parse == \w{r,i}{t,e} ]]
+ [[ parse == \p{a,r}{s}{e} ]]
+ ./easy_pert.py parse -mctal mcnp.FUND-NCERC-PU-HE3-MULT-001-007.r1.pert.m -e group_structure_51_wim_MeV.inp -mcn 100 -ccn 100 -n
94239.00c -rxn 18 -u MeV -d n -fmcn 9979 -p 1. -methods 2 -t 1
Parsing MCTAL file.
(faust2) arclark@p ~: /mnt/c/Users/ /staff_xcp3/projects/easy_pert$
```

```
mcnp.FUND-NCERC-PU-HE3-MULT-001-007.r1.pert.json > ...
1
2
3
4   "attributes": {
5     "comments": "    current tally on the outermost reflector boundary folded wit
6     "tally_number": "1",
7     "perturbed": "False"
8   },
9   "data": {
10    "values": [...
11  ],
12  "uncertainties": [...
13  ],
14  "structure": [...
15  ],
16  "units": {...
17  }
18  },
19  },
20  {
21    "attributes": {
22      "comments": "    current tally on the outermost reflector boundary folded wit
23      "tally_number": "1",
24      "perturbed": "True",
25      "cell": "[100]",
26      "mat": "9979",
27      "rxn": "18",
28      "method": "[2]",
29      "relative_perturbation": "1.0",
30      "nuclide": "94239.00c"
31    },
32    "data": {
33      "values": [...
34    ],
35    "uncertainties": [...
36    ],
37    "structure": [...
38    ],
39    "units": {...
40    }
41  }
42  ]
43  }
```



Summary and future work

- A powerful use-case for the MCNP code PERT card is calculating tally sensitivities to nuclear data¹
- Writing PERT card entries and parsing output MCTAL files is tedious and error-prone
- Easy_PERT makes use of existing tools (Faust² and MCNPTools³) to handle writing PERT card entries and parsing the output MCTAL files
- Early in development process, so upcoming capabilities are
 - Calculating sensitivities
 - Combining MCTAL files from separate runs into one JSON file
- Looking forward to feedback, suggestions, and friendly testers

1. J. Favorite, "Using the MCNP Taylor series perturbation feature (efficiently) for shielding problems," EPJ Web of Conferences 153, 06030 (2017)
2. W. Haeck, A. R. Clark, and M. W. Herman, "Calculating the Impact of Nuclear Data Changes with Crater," Trans. Am. Nucl. Soc., Vol. 123, No. 1, p. 723-726 (Nov. 2020)
3. C. R. Bates et al., "The MCNPTools Package: Installation and Use," LA-UR-22-28935, (Aug. 2022)



Questions?

Contact: arclark@lanl.gov

