# Monte Carlo Adjustment:

*an application to $^{239}$Pu from ENDF/B-VII.0*

**D. Rochman and A.J. Koning**

*Nuclear Research and Consultancy Group,*

*NRG, Petten, The Netherlands*

CSWEG Meeting, Montauk, NY, June 2011

# Contents

① Methodology: TALYS + Monte Carlo + benchmarking

② Goal:

⟹ *Can we use Monte Carlo to create/adjust data libraries ?*

③ Example with $^{239}$Pu

⟹ *500 (and more) random nuclear data libraries*

④ Results 1:

⟹ *Search for the optimum entire file/evaluation*

⑤ Results 2:

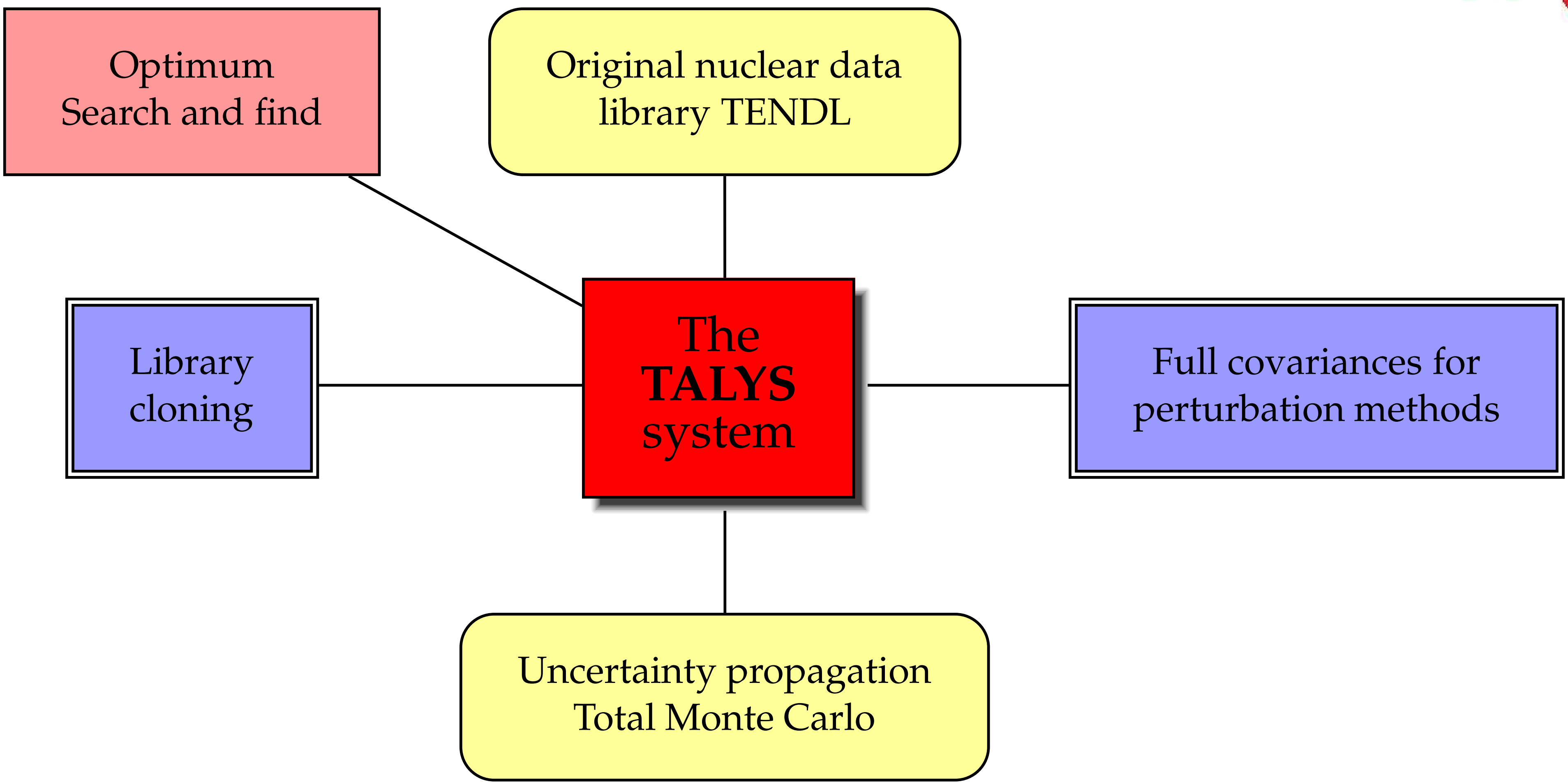
⟹ *Search for the optimum fission neutron spectra*

⑥ Secret plan:

⟹ *to be achieved by 2013*

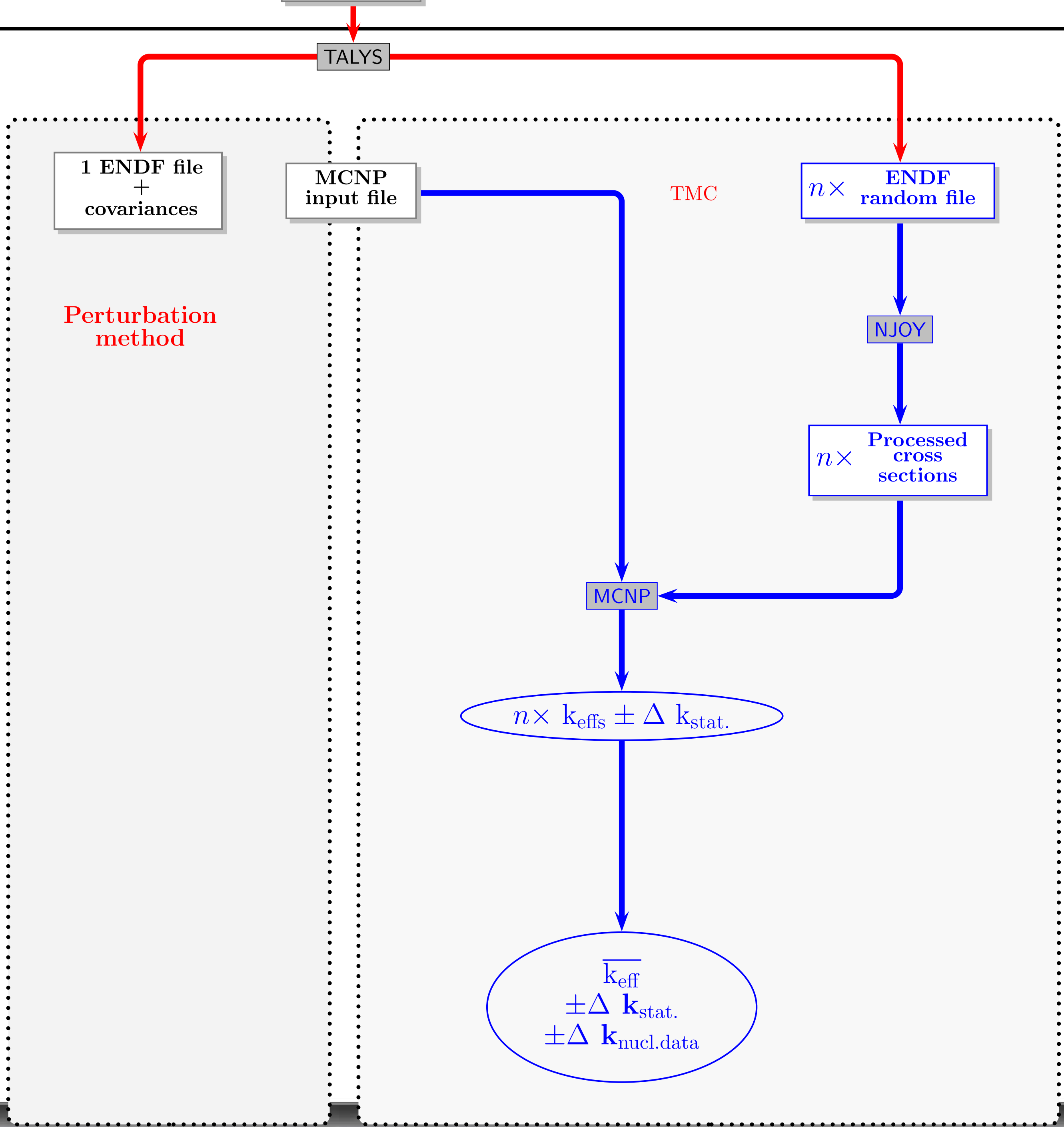
⑦ Conclusions

## The TALYS system and outcomes




Our work is based on the "TALYS system". Completely different applications are possible, all unprecedented.

# Idea: TALYS +( $n$ TALYS input files ) Monte Carlo = Total Monte Carlo

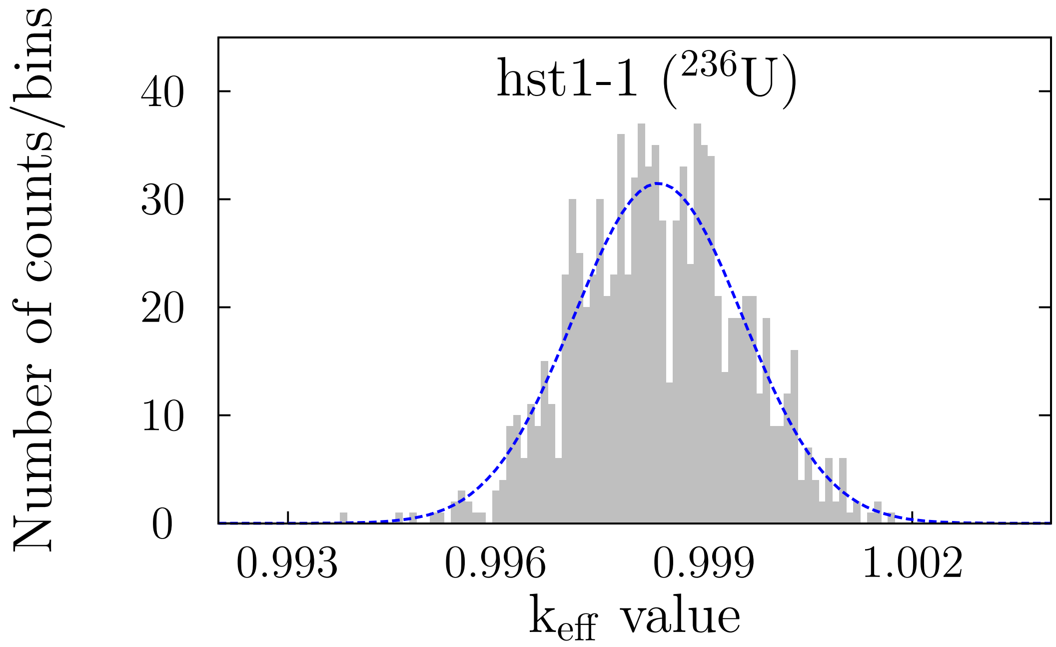


# (1) Total Monte Carlo: examples

*n* ENDF file + *n* MCNP calculations $\Longrightarrow$ *n* different $k_{eff}$

Each random file is completely different: nu-bar, resonance parameters, cross sections...

$$\sigma^2_{total} = \sigma^2_{statistics} + \sigma^2_{nuclear\ data}$$




Basic and original ideas developed and presented by D. Smith in 1991, 2004 and 2007. Test on a unique benchmark by T. Kawano in 2006 (Jezebel $^{239}$Pu fission cross section).

# Example 1 of innovative approach on $^{239}$Pu

Total Monte Carlo + selection
$\Longrightarrow$ adjustment

① Use TALYS to create a single $^{239}$Pu evaluation close or equal to ENDF/B-VII.0 or JEFF-3.1.1

② Randomize all model parameters (resonances, nubar, fission neutron spectrum, TALYS parameters) to create 500 random $^{239}$Pu evaluations

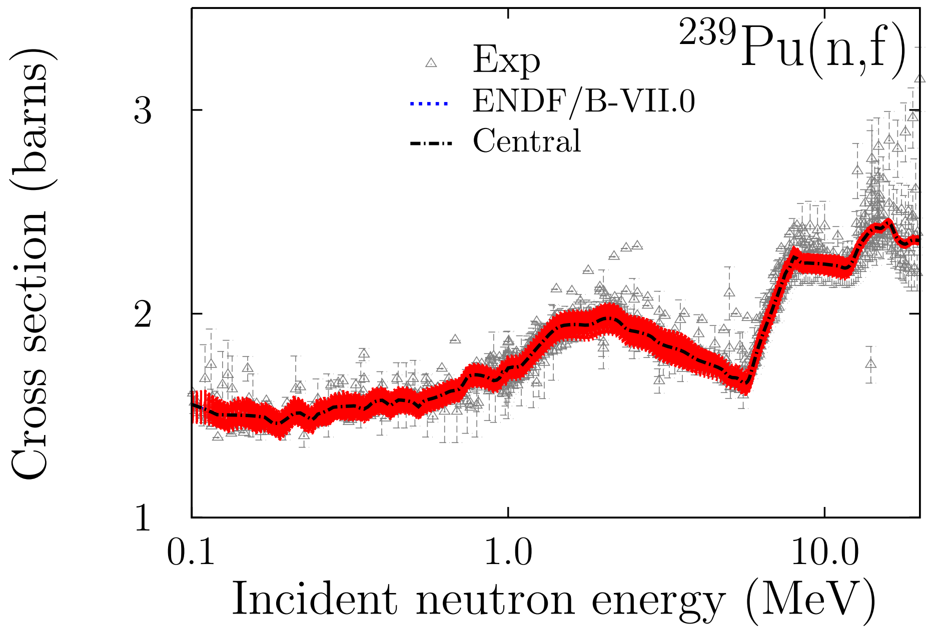
③ Benchmarks the $n \geq 500$ files with the same set of criticality benchmarks

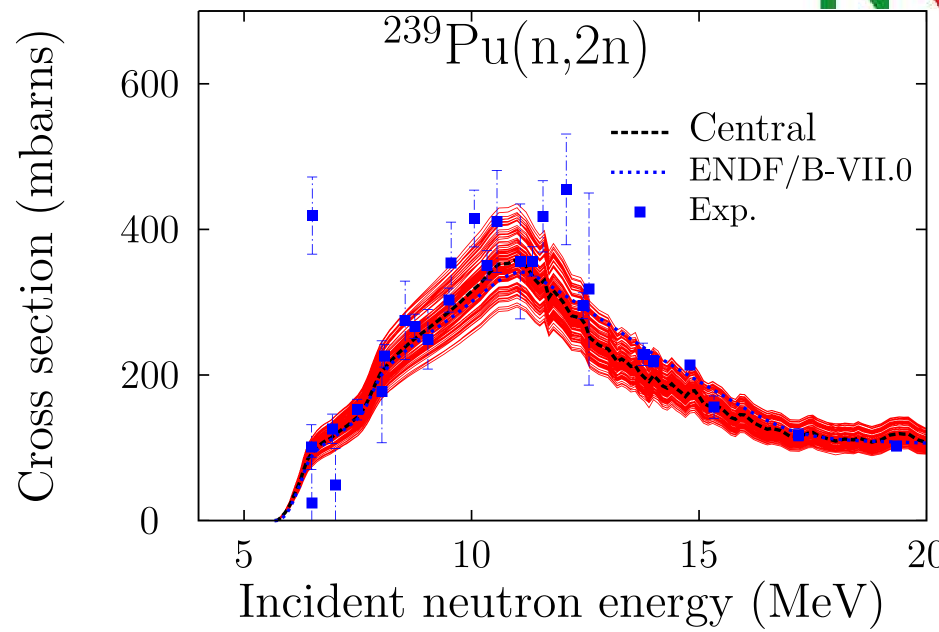
④ Select the best random file

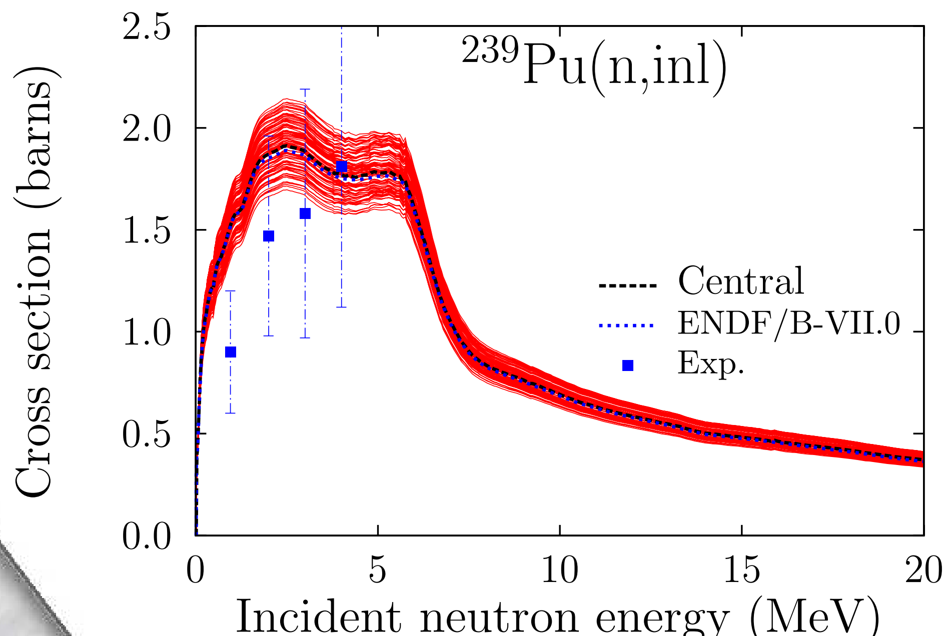
⑤ Create JEFF-3.2β = JEFF-3.1.1 + new $^{239}$Pu

Example: 100 benchmarks, 500 random files $\Longrightarrow$ 500 TALYS + NJOY and $100 \times 500 = 5 \cdot 10^4$ MCNP loops, 1.4 years on a single processor, or 5 days on 100 processors (3 GHz)

# Nuclear data: random $^{239}$Pu in the thermal and fast range



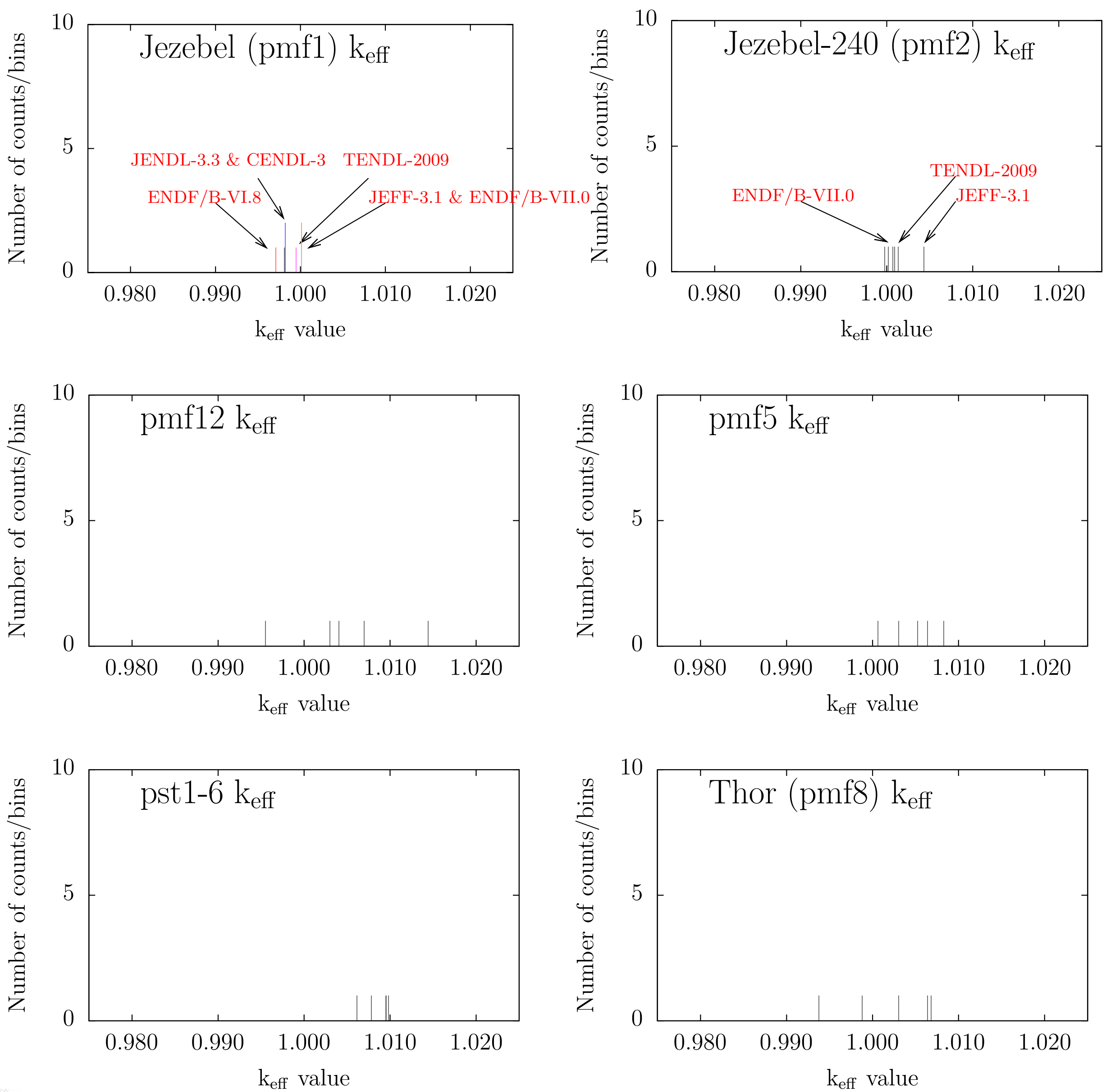




(1) Central cross sections *almost* equal to ENDF/B-VII.0 or JEFF-3.1

(2) Random cross sections obtained from random model parameters

(3) Similar results in MF1, 2, 4, 5 and 6

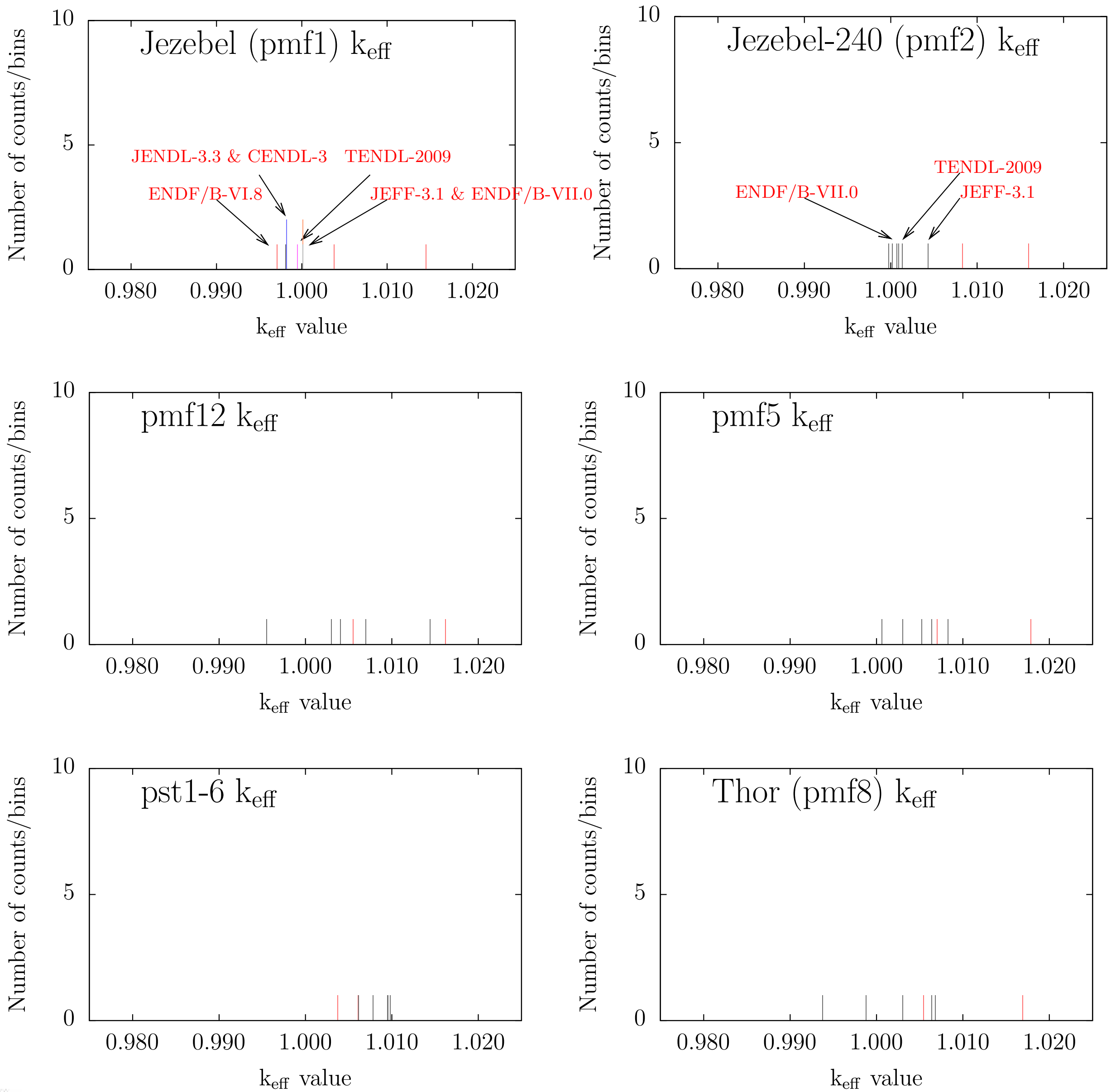
# Benchmarking: simple example with 6 $k_{eff}$ benchmarks




| | $\alpha$ |
|---|---|
| JEFF-3.1.1: | $1.14e^{-4}$ |
| JENDL-3.3: | $1.71e^{-4}$ |
| TENDL-2009: | $3.66e^{-4}$ |
| ENDF/B-VI.8: | $1.72e^{-4}$ |
| ENDF/B-VII.0: | $1.69e^{-4}$ |

$$\alpha = \sum_{i=0}^{n} \frac{(C_i - E_i)^2}{C_i},$$

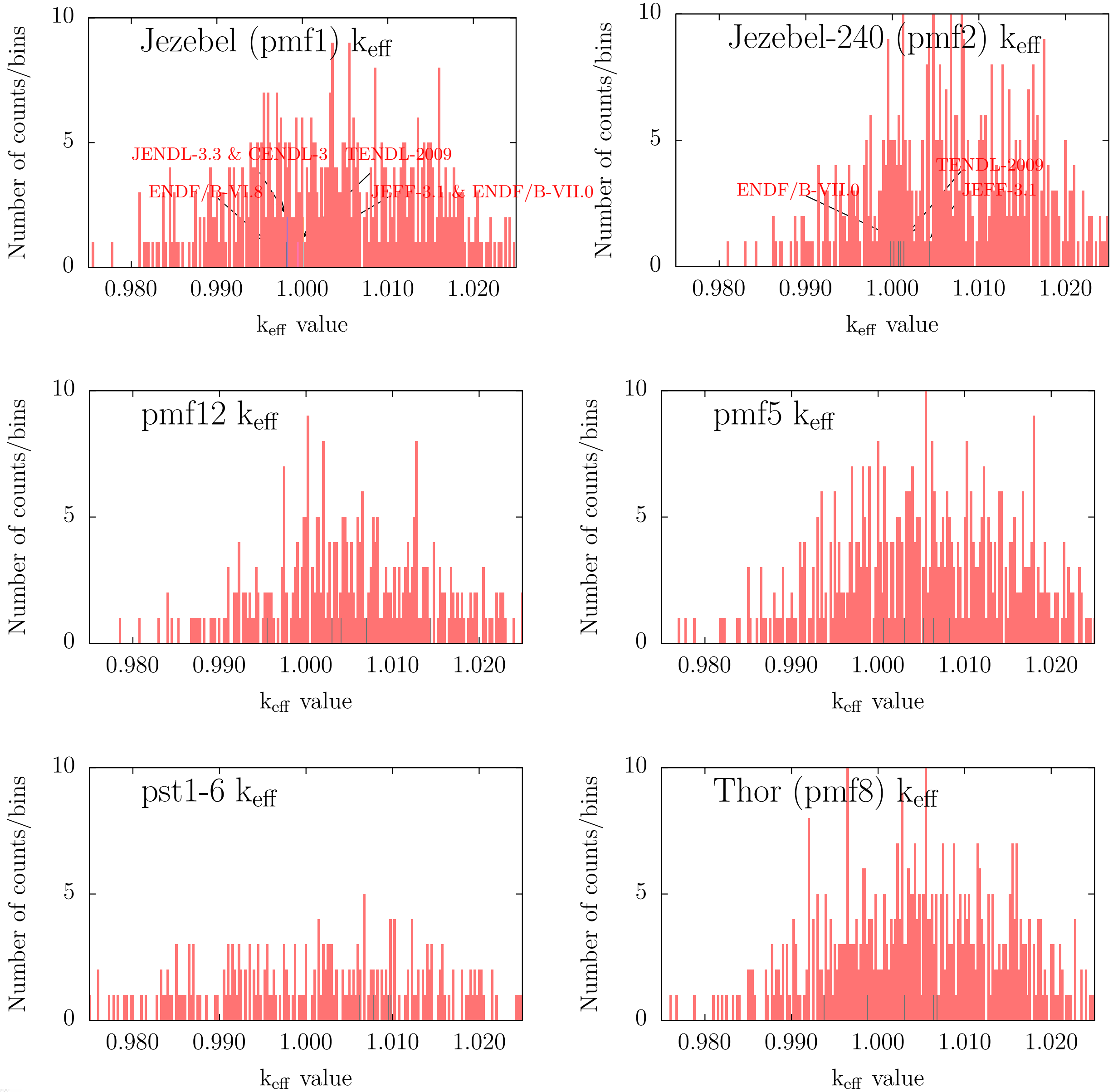
# Benchmarking: simple example with 6 $k_{eff}$ benchmarks




α

| | |
|---|---|
| JEFF-3.1.1: | $1.14e^{-4}$ |
| JENDL-3.3: | $1.71e^{-4}$ |
| TENDL-2009: | $3.66e^{-4}$ |
| ENDF/B-VI.8: | $1.72e^{-4}$ |
| ENDF/B-VII.0: | $1.69e^{-4}$ |
| random 0: | $2.29e^{-4}$ |
| random 1: | $13.4e^{-4}$ |

# Benchmarking: 6 $k_{eff}$ benchmarks with random $^{239}$Pu



## Real case: 120 $^{239}$Pu benchmarks

Table 1: List of plutonium benchmarks selected for the random search.

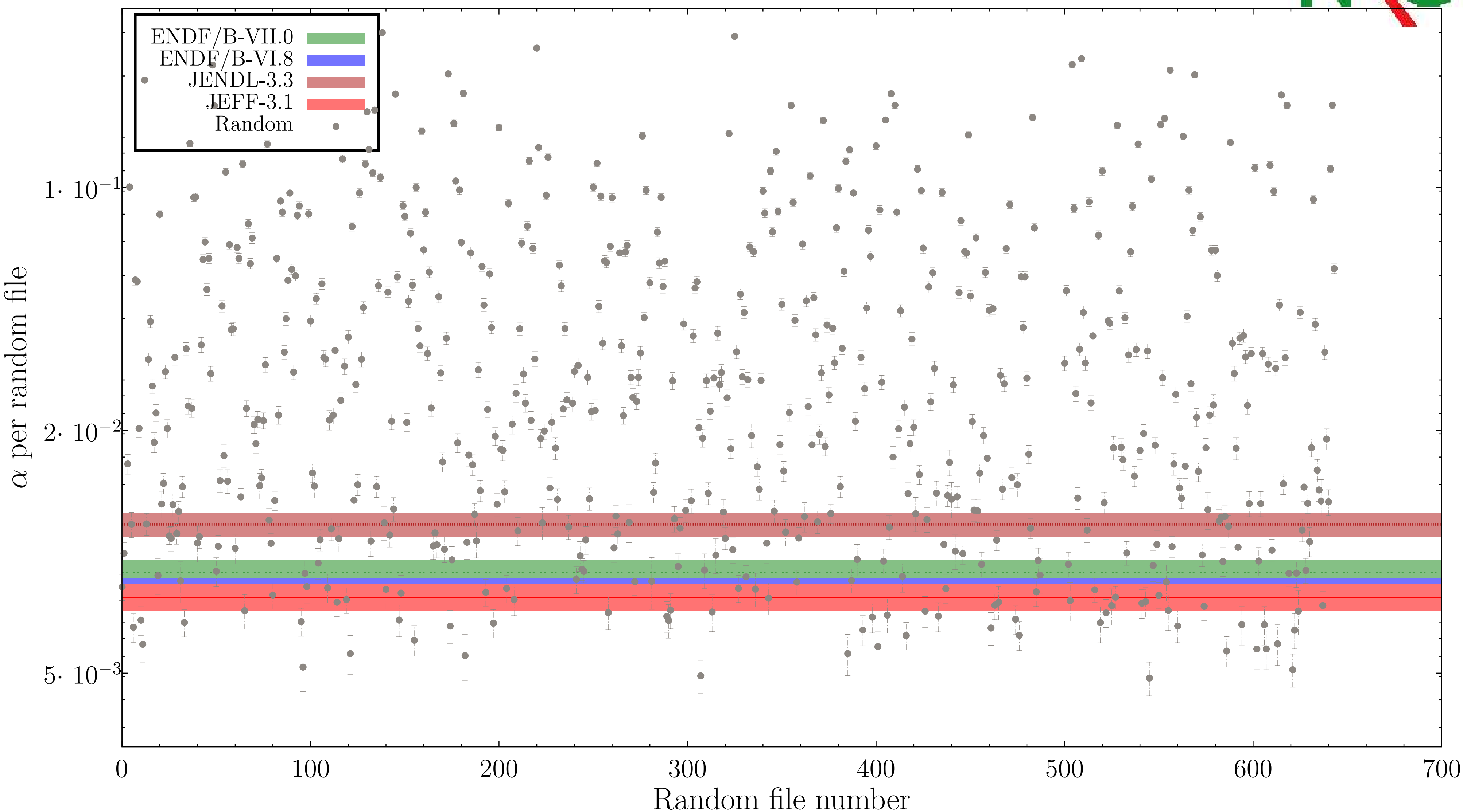
| Name | Cases | Name | Cases | Name | Cases | Name | Cases |
|---|---|---|---|---|---|---|---|
| pmf1 | 1 | pmf2 | 1 | pmf5 | 1 | pmf6 | 1 |
| pmf8 | 1 | pmf12 | 1 | pmf13 | 1 | pci1 | 1 |
| pmi2 | 1 | pst1 | 6 | pst2 | 6 | pst3 | 8 |
| pst4 | 13 | pst5 | 9 | pst6 | 3 | pst7 | 9 |
| pst8 | 29 | pst12 | 22 | pmm1 | 6 | | |

$$\alpha = \sum_{i=0}^{n} \frac{(C_i - E_i)^2}{C_i}, \tag{1}$$

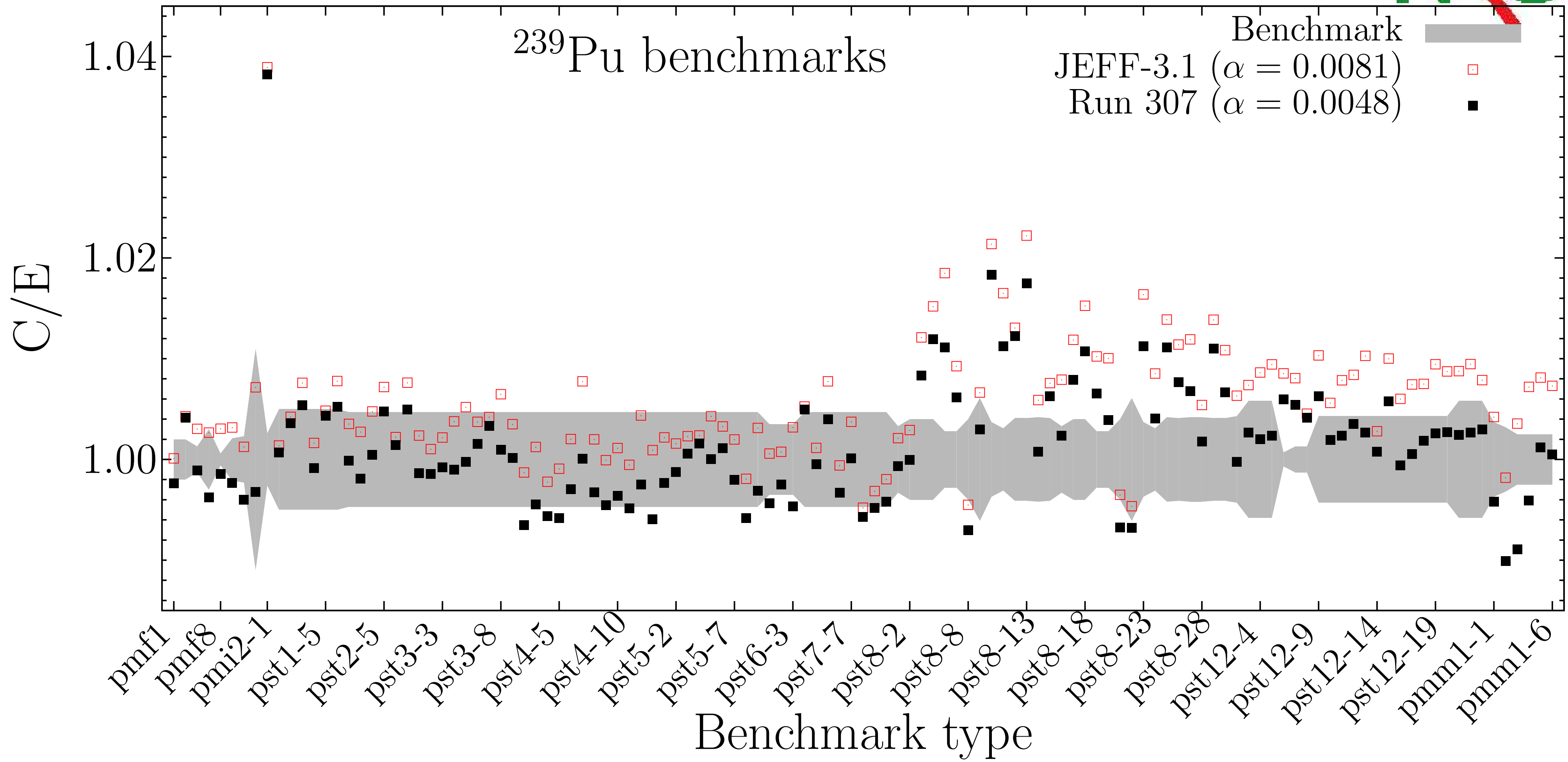
Results independent of the type of factor $\alpha$, $\chi^2$... or

$$F = 1 - 10^{\sqrt{\frac{1}{N}\sum(\log(E_i)-\log(C_i))^2}} \tag{2}$$

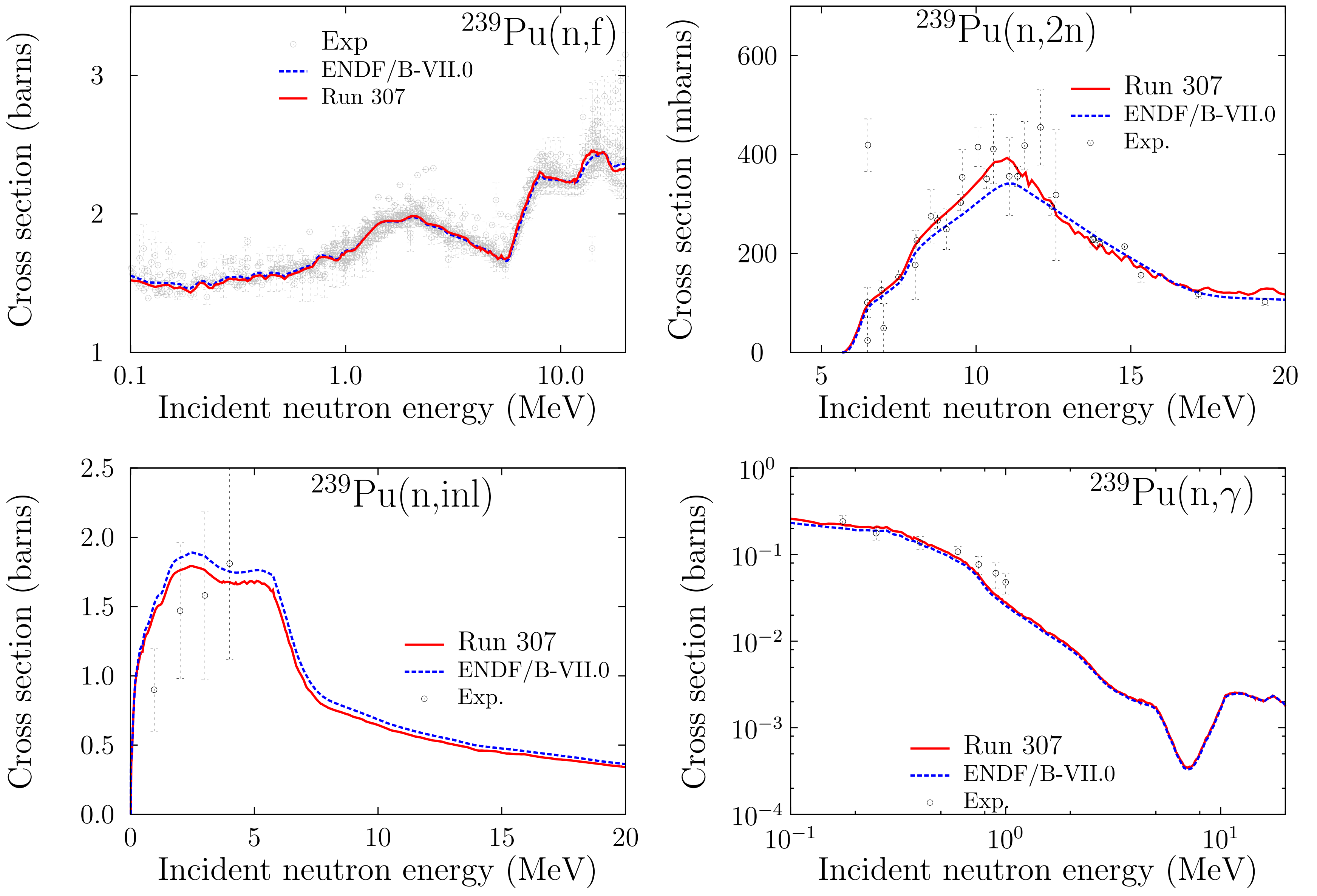
# α values for random $^{239}Pu$ evaluations



# $C/E$ values for the *best* $^{239}$Pu (run 307)



# Optimal cross sections (random file 307)



## Does it work ?

① We have shown that it is possible to ”*improve*” an existing evaluation

② Many questions need to be addressed such as

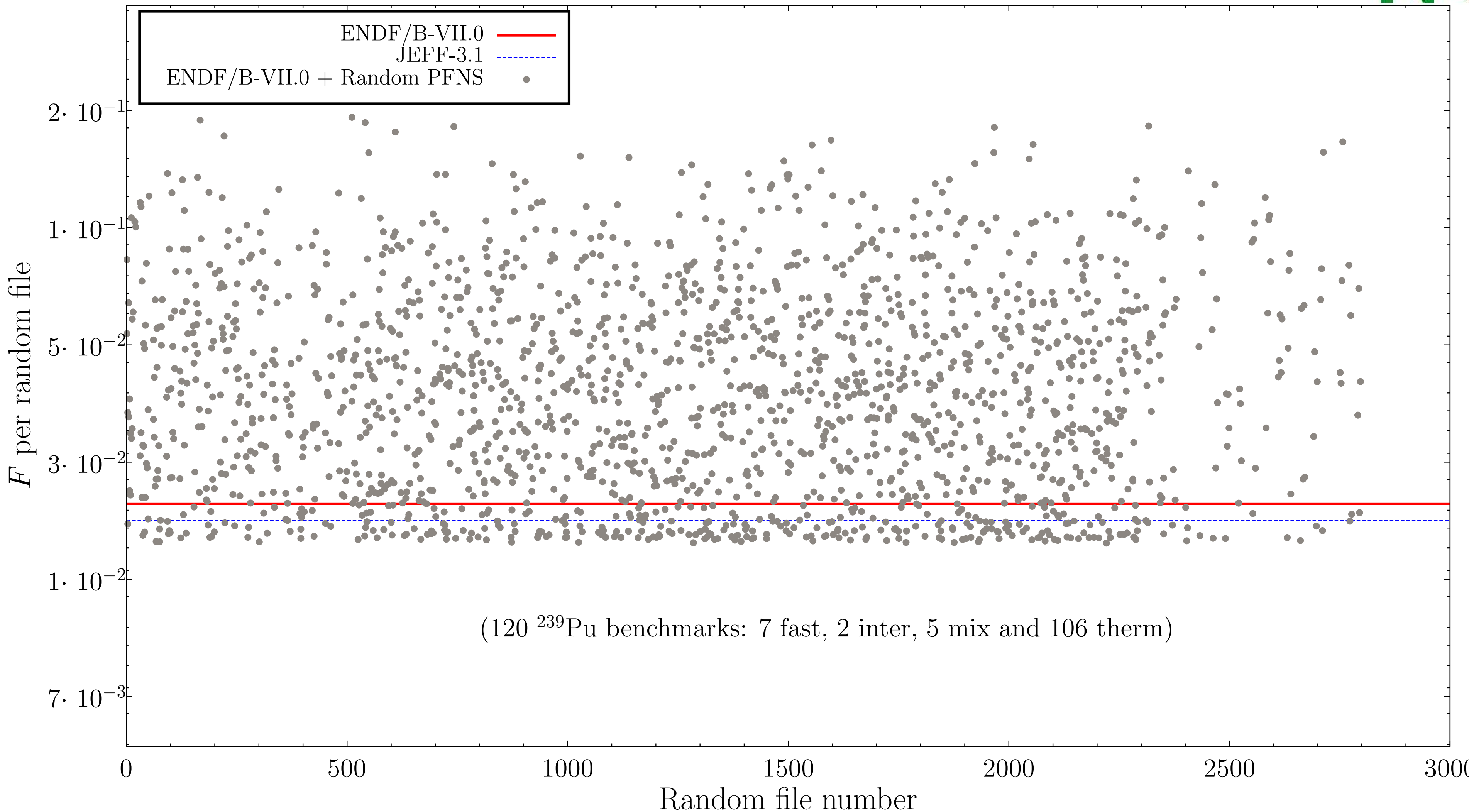
- ➡ Compensation
- ➡ Uncertainty ranges
- ➡ Benchmark list

③ $2^{nd}$ example: fission neutron spectrum for ENDF/B-VII.0 $^{239}$Pu (pfns)

Total Monte Carlo *on pfns*+ selection
$\Longrightarrow$ adjustment *of pfns*

(same benchmarks, keep ENDF/B-VII.0 constant except the pfns, MF5 MT18)

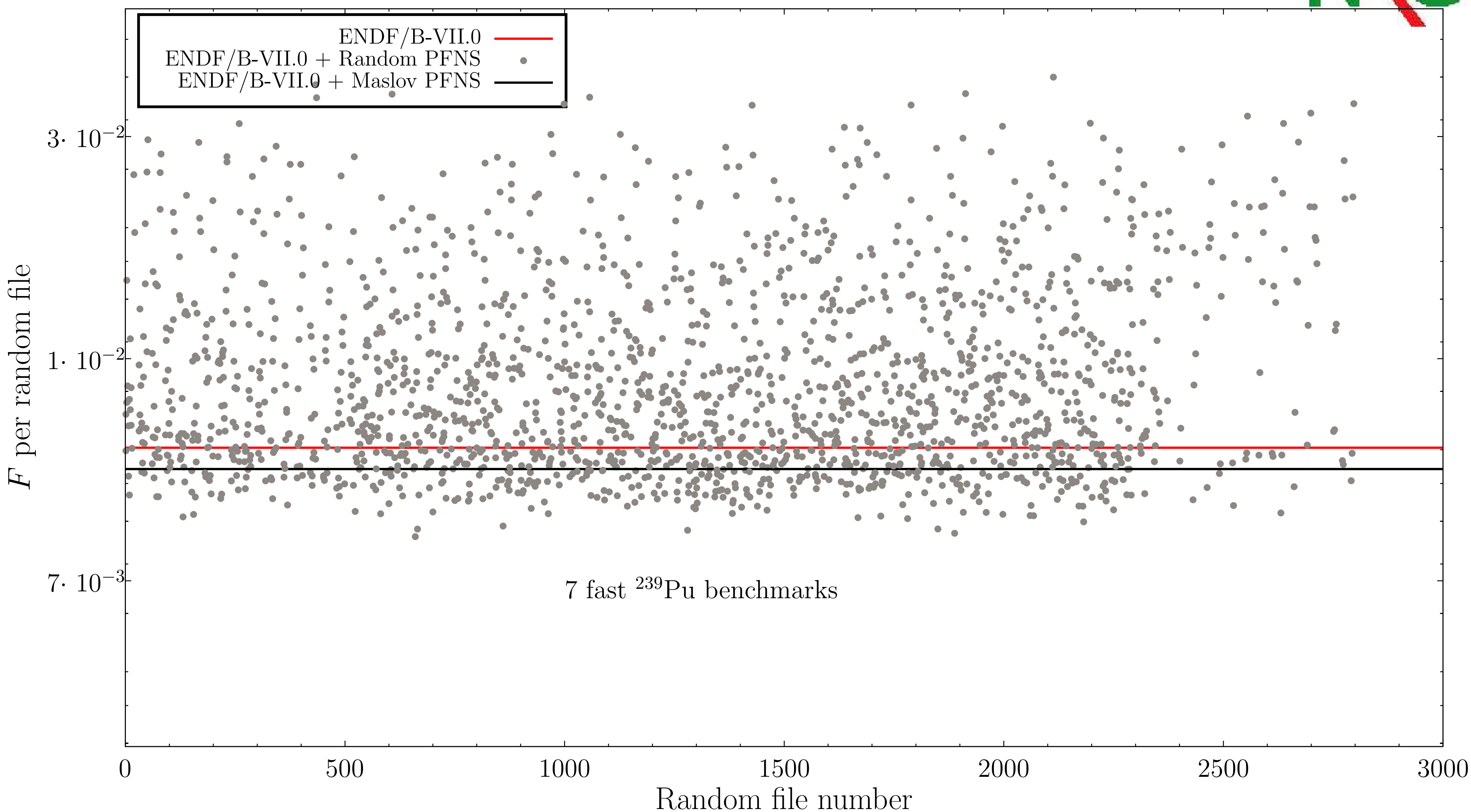
# Example 2 on pfns of $^{239}$Pu



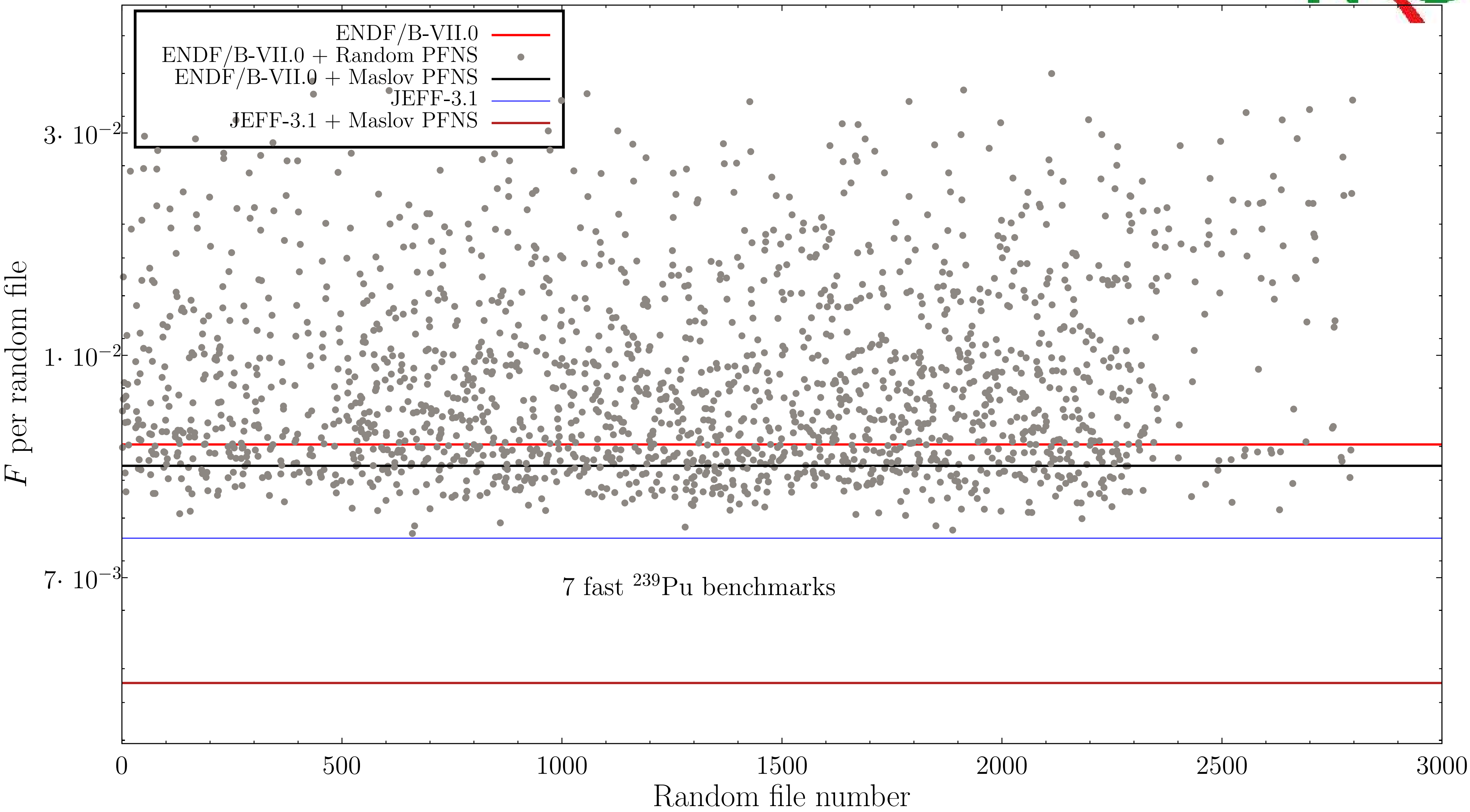

Difference with the previous example (where the whole $^{239}$Pu was random): there is a limit in low $F$

$\Longrightarrow$ like hitting a wall !

# $F$ values for random PFNS $^{239}$Pu evaluations



## $F$ values for random PFNS $^{239}$Pu evaluations



## Generalization

Based on this simple example:

- More random files can be used
- More benchmarks can be included $\Longrightarrow$ JEFF-3.2$\beta$ = JEFF-3.1.1 + New $^{239}$Pu
- Add other actinides (such as $^{235}$U) $\Longrightarrow$ JEFF-3.n$\beta$ = JEFF-3.(n-1)$\beta$ + *i* new isotope
- Apply this method to many (virtually all) isotopes, while improving the agreement with differential data (TALYS) : TENDL

We believe that this approach is condemned to succeed !

# Generalization and (no longer) secret plan

For the next Nuclear Data conference (ND2013) in Manhattan, we want to apply this methodology to produce the first 21$^{st}$ century library (which is different from the first library of the 21$^{st}$ century),

*competing (at least) with any existing library !*

## Generalization and (no longer) secret plan

For the next Nuclear Data conference (ND2013) in Manhattan, we want to apply this methodology to produce the first 21st century library (which is different from the first library of the 21st century),
*competing (at least) with any existing library !*

2010-2013, New library $\Longrightarrow$ A Manhattan Project

## Conclusions

- Clever random search for optimal nuclear data improvement
- What is presented is not about $^{239}$Pu (neither other isotopes)
- It is about
  - propagating current and basic knowledge to applications,
  - using reliable and stable codes,
  - applied for (1) library consistency, robustness
  - and alternatively (2) library optimization without incremental approach
- Not perfect yet (need of more flexible theory, more knowledge...)

None of this would be possible without dependable codes, computer power, systematic approach and a bit of "*brain power*".