PREPRINT · NOT PEER REVIEWED

An in-browser calculator for the nuclear-data uncertainty, similarity and GLS bias of k: description and code-to-code comparisons

What the S/U calculator computes, how it does it, and what its results were compared with

Claude and Ian Hill

Claude is an AI model developed by Anthropic, which has not reviewed or endorsed this article. Ian Hill directed the work.

Posted 29 October 2026 · Comparisons run 28–29 October 2026

ABSTRACT

The S/U calculator is a web page that runs first-order sensitivity and uncertainty calculations in the user's browser, from TSUNAMI sensitivity data files or Serpent sensitivity output. It computes the nuclear-data uncertainty of k by the sandwich rule, its breakdown by nuclide and reaction, and the similarity index c_k between systems. It also computes a generalized-linear-least-squares (GLS) adjustment, as in SCALE's TSURFER, that gives the bias of k for applications from benchmark experiments, the uncertainty that remains after the adjustment, and χ^2 of the experiments. The covariances are ENDF/B-VIII.1 in SCALE's 56 groups (AMPX and NJOY routes), ENDF/B-VII.1 from SAUNA, or a COVERX file of the user's own, and the user's files are never uploaded. This article describes the methods and reports seven sets of code-to-code comparisons. For 120 ICSBEP configurations with each of the three libraries, the calculator gives the uncertainties of the companion ICSBEP study to within 1.5×10^{-5} percentage points. The COVERX reader and the Serpent reader agree with their references to the precision of the stored or printed data, as does the uncertainty breakdown against SAUNA's (2.6×10^{-6} of the variance per pair). The GLS adjustment agrees with a separate Python implementation to 8.5×10^{-14} (relative) in every output. SANDY's parameter-space update (`gls_update`), written by other authors, was run with the dense covariance matrix of up to 8,400 parameters; the calculator's bias and posterior uncertainty agree with it to 1.4×10^{-14}. These comparisons show that the implementations agree with each other; they do not test the method against measurements. The calculator has not yet been compared with TSURFER, TSUNAMI-IP or NDaST directly.

Keywords: sensitivity and uncertainty analysis; sandwich rule; similarity index c_k; generalized linear least squares; TSURFER; data adjustment; criticality safety

validation; ENDF/B-VIII.1; TSUNAMI; Serpent; web application

CITE THIS PAPER

Claude and Ian Hill (2026). An in-browser calculator for the nuclear-data uncertainty, similarity and GLS bias of k: description and code-to-code comparisons. Preprint, not peer reviewed; posted 29 October 2026. Modern Nuclear Data Processing. https://doi.org/10.5281/zenodo.23765908 (web version: https://hardtohit10.gitlab.io/su-calculator-paper.html)

This article is available under the Creative Commons Attribution 4.0 International licence (CC BY 4.0): it may be shared and adapted, with credit to the authors. Claude is an AI model developed by Anthropic.

1. Introduction

Sensitivity and uncertainty (S/U) methods propagate nuclear-data covariances to an integral response, here the multiplication factor k, through its first-order sensitivities. Three calculations are standard in criticality safety [1]. The sandwich rule gives the nuclear-data uncertainty of k of one system. The similarity index c_k gives the correlation between the nuclear-data errors of two systems, which is used to judge whether a benchmark experiment is relevant to an application. A generalized-linear-least-squares adjustment combines measured benchmarks with the covariances to estimate the bias of an application's calculated k and the uncertainty that remains [2][3]. In SCALE these are TSUNAMI-IP and TSURFER [4]. The NEA's NDaST [5] and the open Python packages SAUNA [6] and SANDY [7] do some or all of them as well.

The S/U calculator [8] does all three in a web page. It needs no installation and no licence, and it reads the user's sensitivity files in the browser without sending them anywhere, which matters where sensitivity data describe proprietary or sensitive designs. It uses the ENDF/B-VIII.1 covariance libraries in SCALE's 56 groups of a companion project [9], and its arithmetic is the one used for the uncertainties of 2,843 ICSBEP configurations [10] and for a CANDU-6 similarity study [11]. This article describes what the calculator computes and how (Section 2), and reports the comparisons made with other implementations (Section 3), with their limits (Section 4). The equations of the GLS adjustment are derived in more detail, with worked examples, in the calculator's theory manual [12].

2. The calculator

2.1 Structure and privacy

The calculator is three files: the page (`su-tool.html`), its script (`su-tool.js`, which handles reading files, choosing libraries and showing results) and `su-core.js`, an ES module with the file readers and all the arithmetic, which also runs in Node.js and is what the tests call (Figure 1). Sensitivity files, COVERX files and the CSV tables of the GLS section are read with the browser's `File` interface into the tab's memory. The page fetches only the site's covariance files, one per nuclide and only for the nuclides in the loaded files, and

its Content Security Policy blocks connections to any other server. Results are shown on the page, and CSV results are shown as text to copy rather than as a download. A separate page describes how the files stay private, with the network requests the page makes and how to check them [13].

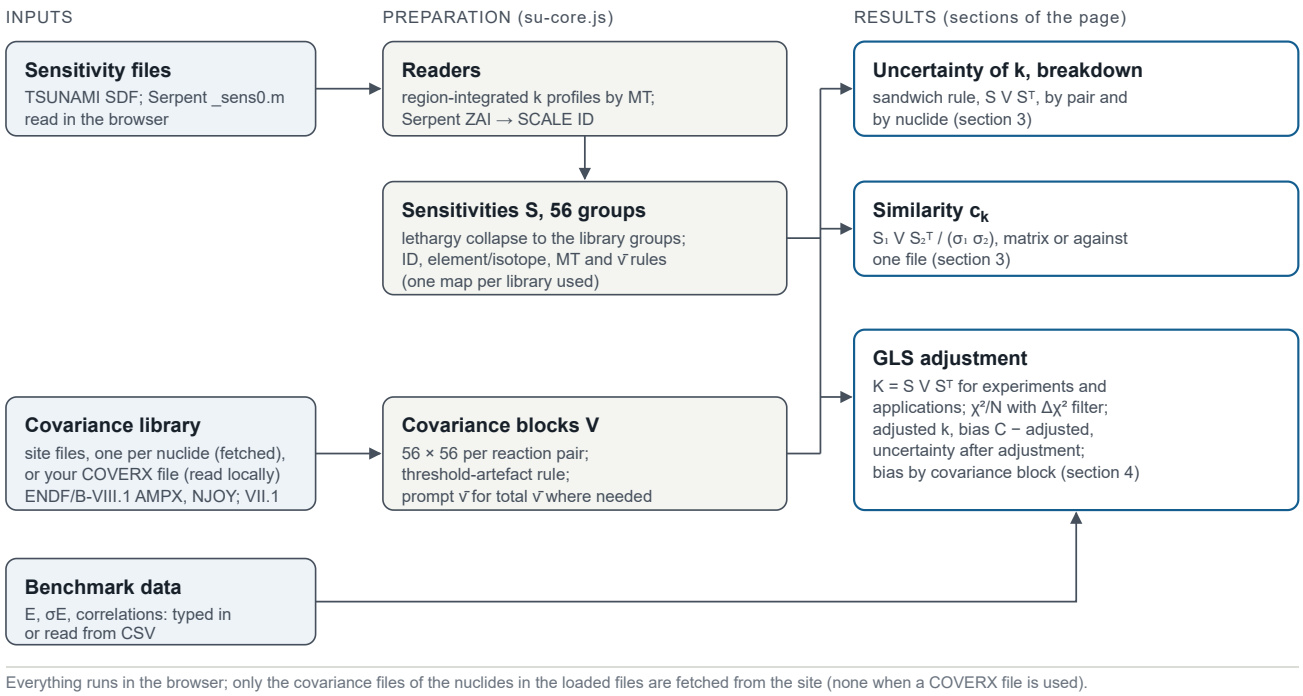

Figure 1. Data flow of the calculator. Everything runs in the browser; only the covariance files of the nuclides in the loaded files are fetched from the site.

2.2 Sensitivity files

TSUNAMI SDF. The reader takes the sensitivity data files of SCALE 6.0–6.3 in any group structure, and uses only their region-integrated profiles. Each profile is identified by its SCALE nuclide ID and MT number, and k and its standard deviation are read from the header.

Serpent. For Serpent 2 [14], the reader takes the k-eff sensitivities of the `_sens0.m` output (`ADJ_PERT_KEFF_SENS`, all latent generations) [15] for the material "total". It follows the Serpent reader of SAUNA [6]. A ZAI (10000 Z + 10 A + I) becomes the SCALE ID 1000 Z + A, plus 10^6 for a metastable state. When a sum and its parts are both listed, the parts are left out, for example the inelastic levels MT 51–91 when MT 4 is listed. Thermal scattering is added to elastic scattering, and total $\bar{\nu}$ and χ are MT 452 and 1018. A file with summed reaction modes (`sens pert xs all`) is mapped to MT 2, 4, 16, 18 and 102, with a note that its capture includes all non-fission absorption. Serpent files hold no k, so the GLS section asks for it.

2.3 Covariance data

Three libraries are on the site: release v1.2 of the ENDF/B-VIII.1 56-group libraries processed by the AMPX route (the default) and by the NJOY route [9], 467 nuclides each, and SAUNA's ENDF/B-VII.1 56-group library [6], converted to the same format. Each nuclide is one JSON file with the relative standard deviations and the correlation matrices of its reaction pairs, the correlations stored to four decimal places. A user's COVERX file, for example SCALE's own covariance library, is read in the browser and replaces or

complements these. A different library can be chosen for individual nuclides. Covariances between nuclides assigned to different libraries are then left out.

2.4 Processing rules

The sensitivities are collapsed to the library's groups, splitting a bin of the file that straddles a library boundary in proportion to lethargy, and profiles that map to the same nuclide and reaction are added. MT 1, 3, 101, 455 and 456 are left out, as in SCALE's sandwich calculations, because they duplicate other reactions. Where a nuclide has a prompt-ν̄ covariance but no total-ν̄ covariance, the prompt-ν̄ data stand for total ν̄. Bound-scatterer IDs take the covariances of their nuclide. An element without covariances takes those of its dominant isotope when that isotope's abundance reaches a threshold (90 % by default, an option of NDaST [5]). An isotope without covariances, of an element that has them, is added to the elemental sensitivity. In a few groups at reaction thresholds the processed libraries give relative standard deviations of 10^{2} to 10^{8}; by default the rows and columns of every group above 100 are set to zero, following Section 2.3 of [11].

2.5 Uncertainty and similarity

The relative variance of k is the sandwich rule, $(\sigma_k/k)^2 = \mathbf{S}\,\mathbf{V}\,\mathbf{S}^{\mathrm{T}}$, summed over the covariance blocks $\mathbf{V}_{pq}$ between nuclide–reaction pairs p and q, with a block between two different reactions counted in both orderings. Each block's term is reported as a signed square root in % Δk/k and as a share of the variance, and the terms are also added by nuclide (Figure 2). Sensitivities that have no covariance data, and nuclide IDs the library lacks, are listed. The similarity index is $c_k = \mathbf{S}_1\,\mathbf{V}\,\mathbf{S}_2{}^{\mathrm{T}} / (\sigma_1\,\sigma_2)$ [1]. It is computed by forming each system's $\mathbf{V}\,\mathbf{S}^{\mathrm{T}}$ once and taking dot products with the others, and is shown as a matrix for up to 40 files and against one chosen file for any number.


BREAKDOWN candu6-cooled.sdf

0.807 % Δk/k from 215 nuclide–reaction terms; k = 1.12531 ± 0.00012 in the file

CANDU-6 37-element cell, cooled, k-inf, OpenMC CE sensitivities, ENDF/B-VIII.1

BY NUCLIDE–REACTION PAIR

Pair	% Δk/k	Share of variance
U-235 ν̄	0.4265	27.94 %
H-2 n,2n	0.3816	22.37 %
U-238 n,γ	0.3738	21.46 %
Zr-91 n,γ	0.2503	9.62 %
Zr-92 n,γ	0.2293	8.08 %
U-235 fission	0.2233	7.65 %
H-2 elastic	0.0957	1.41 %
U-238 ν̄	0.0617	0.58 %
Nb-93 n,γ	0.0411	0.26 %
U-238 fission	0.0361	0.20 %
U-235 n,γ	0.0306	0.14 %
U-238 elastic × n,γ	-0.0268	-0.11 %
U-235 fission × n,γ	-0.0265	-0.11 %
O-16 n,α	0.0263	0.11 %

Show all 215 terms

adds to the variance reduces it

BY NUCLIDE

Nuclide	% Δk/k	Share of variance
U-235	0.4823	35.72 %
H-2	0.3938	23.82 %
U-238	0.3802	22.20 %
Zr-91	0.2503	9.62 %
Zr-92	0.2293	8.08 %
Nb-93	0.0411	0.26 %
O-16	0.0361	0.20 %
Zr-90	0.0229	0.08 %
U-235 × U-238	-0.0087	-0.01 %
H-1	0.0084	0.01 %
Zr-94	0.0066	0.01 %
B-10	0.0025	0.00 %
Fe-56	0.0024	0.00 %
Zr-96	0.0022	0.00 %

Terms within one nuclide added; terms between two nuclides as pairs.

SENSITIVITIES WITHOUT COVARIANCE DATA

None with |sum| ≥ 0.001.

NUCLIDE IDS NOT IN THE LIBRARY

Every nuclide ID of the file is in the library.

Figure 2. Breakdown of the nuclear-data uncertainty of k-inf for the page's CANDU-6 example, cooled, with the AMPX-route library (the page's section 3). The CANDU-6 study [11] gives 0.81 % for the same cell and library.

2.6 GLS adjustment

Files are marked as experiments, with their measured k (E) and its standard deviation, or as applications. For all of them the page forms the matrix $\mathbf{K}$ with elements $K_{ij} = \mathbf{S}_i\,\mathbf{V}\,\mathbf{S}_j^{\mathrm{T}}$. For the experiments, $d_i = (E_i - C_i)/C_i$, and the combined covariance of the discrepancies is

$$\mathbf{A} = \mathbf{K}_{ee} + \mathbf{V}_d, \quad (1)$$

where $\mathbf{V}_d$ holds $(\sigma_E/C)^2$, plus $(\sigma_C/C)^2$ for the calculation's own (for example Monte Carlo) uncertainty when that option is on (the default), and $\rho_{ij}\,\sigma_{E,i}\,\sigma_{E,j} / (C_i\,C_j)$ for experiments whose uncertainties are correlated. With $\mathbf{w} = \mathbf{A}^{-1}\mathbf{d}$, the consistency of the experiments is

$$\chi^2 = \mathbf{d}^{\mathrm{T}}\,\mathbf{A}^{-1}\,\mathbf{d}, \quad (2)$$

reported as χ^2/N, and for an application a the relative change of k, the bias and the remaining relative variance are

$$\delta_a = \mathbf{K}_{ae}\,\mathbf{A}^{-1}\,\mathbf{d}, \quad \beta_a = C_a - C_a(1+\delta_a) = -C_a\,\delta_a, \quad K_{aa} - \mathbf{K}_{ae}\,\mathbf{A}^{-1}\,\mathbf{K}_{ea}. \quad (3)$$

Equations (1)–(3) are the GLS update of the nuclear data, $\mathbf{x}^{\text{post}} = \mathbf{V}\,\mathbf{S}_e^{\mathrm{T}}\,\mathbf{A}^{-1}\,\mathbf{d}$ and $\mathbf{V}^{\text{post}} = \mathbf{V} - \mathbf{V}\,\mathbf{S}_e^{\mathrm{T}}\,\mathbf{A}^{-1}\,\mathbf{S}_e\,\mathbf{V}$, projected onto the applications: $\mathbf{S}_a\,\mathbf{x}^{\text{post}} = \delta_a$, and $\mathbf{S}_a\,\mathbf{V}^{\text{post}}\,\mathbf{S}_a^{\mathrm{T}}$ is the remaining variance [2][3]. The page never forms the data adjustment itself. Its matrices are $N \times N$ in the number of systems rather than in the number of parameters, which is (reactions × groups), up to tens of thousands. $\mathbf{A}$ is factored by Cholesky decomposition. For each experiment the page shows the individual χ, $|d_i|/\sqrt{A_{ii}}$, the fall in χ^2 if it is left out, $\Delta\chi^2_i = w_i^2/(\mathbf{A}^{-1})_{ii}$, and its C/E before and after adjustment. An optional filter removes, one at a time, the experiment with the largest $\Delta\chi^2$ until χ^2/N reaches a target (1.2 by default). Experiments marked *keep* are never removed, by this filter or by an optional minimum c_k. Because $\delta_a = \mathbf{S}_a\,\mathbf{V}\,\mathbf{u}^{\mathrm{T}}$ with $\mathbf{u} = \Sigma_e\, w_e\, \mathbf{S}_e$, the bias splits exactly over the covariance blocks, and the page lists these parts, which add up to the bias (Figure 3). The derivations, the numerical method and worked examples, including a two-experiment case that can be followed by hand, are in the theory manual [12].

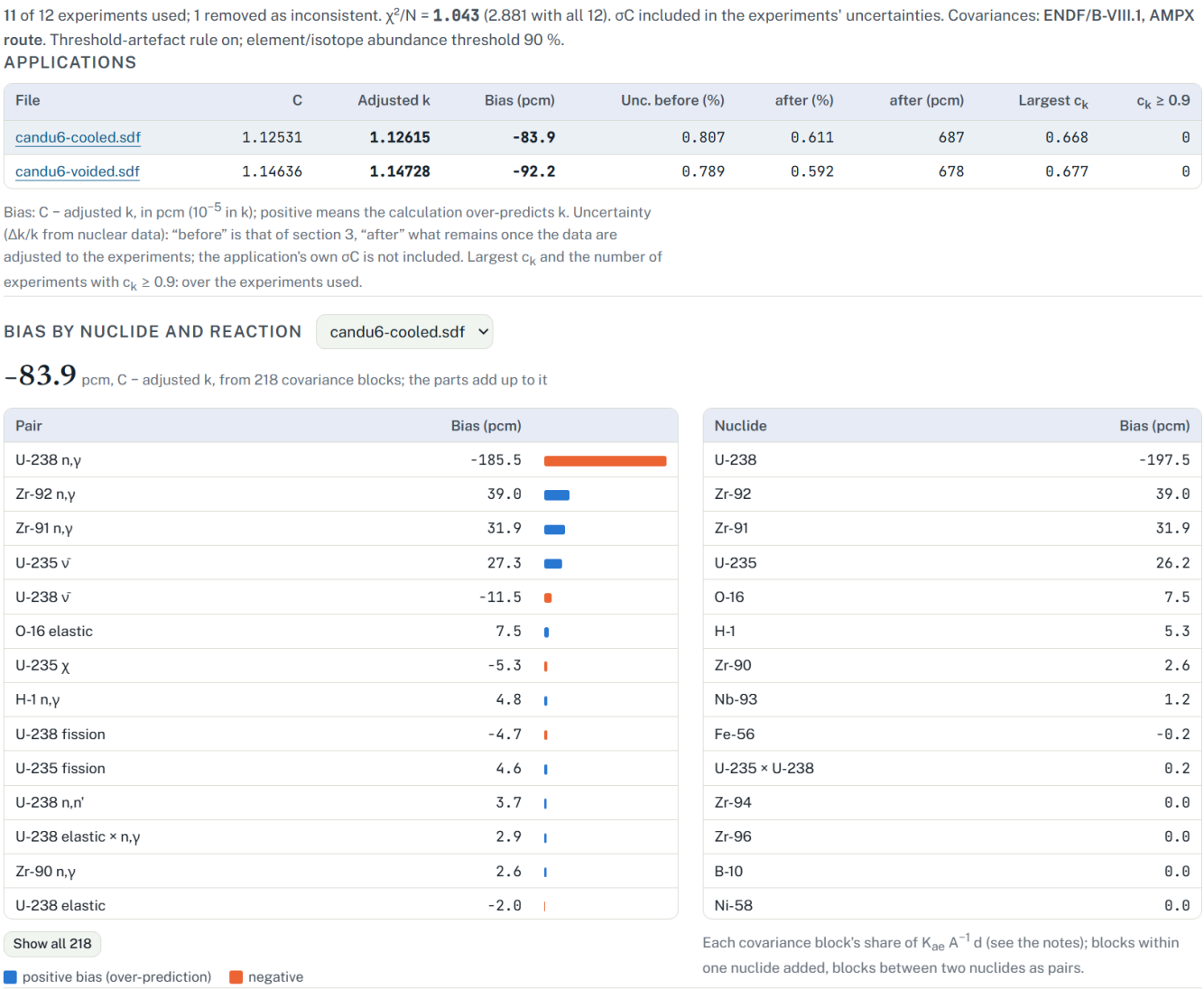

Figure 3. The GLS section of the page with 12 LEU-COMP-THERM configurations as experiments and the two CANDU-6 cells as applications (AMPX route). The calculated k values come from the configurations' sensitivity files, but **the measured values are made up** (E = 1 + 0.004 sin 3*i*, σ_E = 0.0020); the numbers illustrate the output and are not a validation result. The table of experiments below these on the page is not shown.

3. Comparisons

Each comparison is a Node.js script in the repository [8] that calls `su-core.js` directly, the same module the page runs. Table 1 lists them and Figure 4 shows how closely each agreed. All were run on 28 and 29 October 2026 with the code of the calculator as published on those days.

Table 1. The comparisons, what the calculator was compared with, who wrote the reference, and the largest difference found.

Quantity	Compared with (written by)	Inputs	Largest difference
Uncertainty of k	Results of the ICSBEP study [10]: Python, full-precision libraries (Claude)	120 DICE configurations, 3 libraries	1.5×10^{-5} %-points 2.9×10^{-5} relative
COVERX reader	The same library in the site's JSON files	Godiva and the two CANDU-6 cells	3.4×10^{-6} relative
Serpent reader	Serpent's energy-integrated sensitivities (Serpent)	600 profiles of SAUNA's MET1000 example	1.4×10^{-5} relative
Uncertainty and pair terms, Serpent and SDF	SAUNA's breakdown [6] (A. Ryzhkov)	MET1000, Serpent and SCALE files, ENDF/B-VII.1	2.6×10^{-6} of the variance
GLS: all outputs, filter	`glls_bias.py`, on the same **K** (Claude)	24 DICE configurations, 2 applications, 9 fits	8.5×10^{-14} relative
K and GLS	SANDY's `gls_update` [7] in parameter space (SANDY's authors), and a dense NumPy matrix product	the same cases; 3,360 and 8,400 parameters	1.4×10^{-14} relative
The page as a whole	`su-core.js` in Node.js with the same inputs	12 DICE configurations, 2 applications, CSV inputs	none in the digits shown

DICE: the sensitivity profiles of the ICSBEP Handbook's configurations computed with SCALE 6.0 [16][17]. In the GLS comparisons the measured k values are made up (Section 3.4).

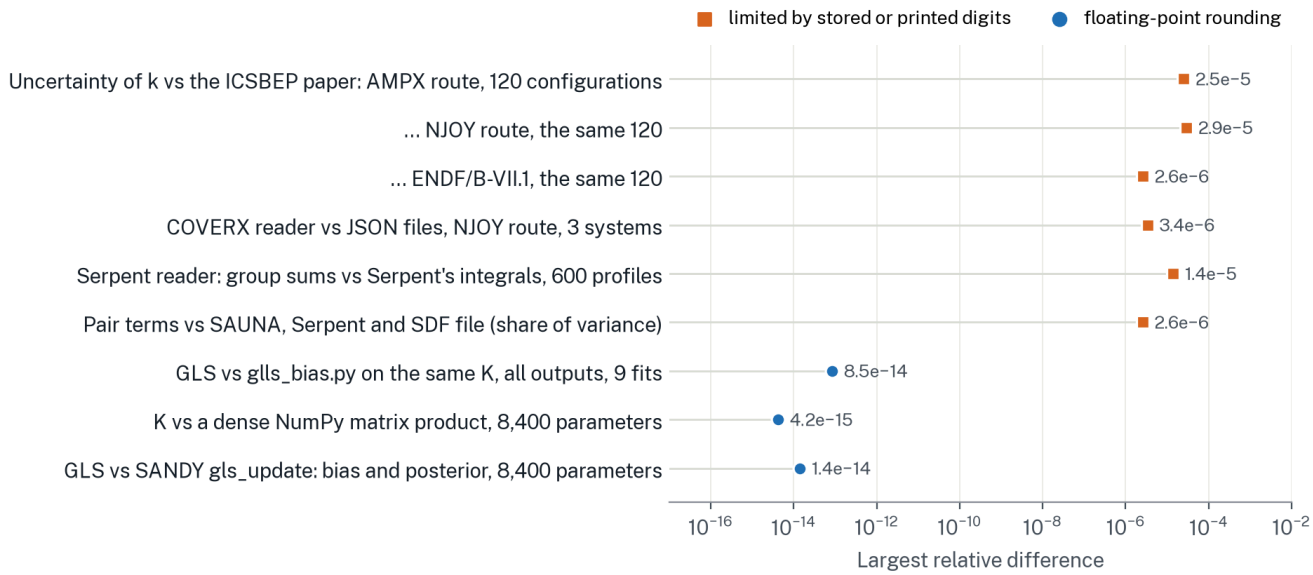

Figure 4. Largest relative difference in each comparison. The first six are limited by the precision of the stored or printed data (correlations stored to four decimals in the site's files; five or six digits in Serpent's output), the last three by floating-point rounding.

3.1 Uncertainty of k against the ICSBEP study

The companion study [10] computed the uncertainty of 2,843 ICSBEP configurations with a Python implementation that reads the libraries at full precision. The test takes every 23rd configuration that the study extracted, 120 configurations from 20 of the Handbook's categories, reads their original SCALE 6.0 sensitivity files with the calculator's reader and computes the uncertainty with each of the three site libraries, without the threshold-artefact rule, which that study did not use. The uncertainties range from 0.42 % to 2.8 % Δk/k. The largest differences are 1.5×10^{-5} percentage points with the AMPX and NJOY routes and 3.1×10^{-6} with ENDF/B-VII.1, and the median relative differences are below 4×10^{-6}. They are consistent with the four-decimal storage of the correlations in the site's files. The same study compared its results with those posted for the NEA's NDaST, an independent implementation, for ENDF/B-VII.1 and SCALE's library. For the 35 per-nuclide mean contributions with enough configurations, the median ratio was 1.00 and 1.03, and in the largest category the two agreed within 2 % (Section 4.6 of [10]). That comparison is of category means over different sets of configurations, not of individual results.

3.2 COVERX reader

The NJOY-route library is published both as a binary COVERX file and as the site's JSON files [9]. Read from the COVERX file, it gives the same 56 group boundaries and 499 materials as the JSON export, and uncertainties of 1.01791 %, 0.80436 % and 0.78875 % for Godiva and the two CANDU-6 cells, which differ from the JSON-based values by at most 3.4×10^{-6} (relative).

3.3 Serpent reader and SAUNA

SAUNA's example model MET1000_FC, a fast metallic core, comes as Serpent output in 56 groups and as a SCALE SDF of the same model [6]. For all 600 profiles of the Serpent file, the group sums of the profiles the calculator reads equal Serpent's own energy-integrated sensitivities (`ADJ_PERT_KEFF_SENS_E_INT`) to

1.4×10^{-5} (relative), the precision of Serpent's printed values. This tests the index arithmetic of the five-dimensional array.

The uncertainty was then compared with SAUNA's own breakdown (`Analysis.get_breakdown`, SAUNA's ENDF/B-VII.1 56-group covariances, regenerated for this article). The totals differ by design: the calculator gives 1.10012 % for the Serpent file and SAUNA 1.09708 %. SAUNA includes MT 251 and the prompt and delayed $\bar{\nu}$ and χ terms, which the calculator leaves out (Section 2.4). Those terms add 0.0347 % in quadrature. The calculator's prompt-for-total $\bar{\nu}$ rule, which applies to seven nuclides here, adds 0.0889 %. Over the same reactions SAUNA gives 1.09653 %, and adding the rule's terms gives 1.10012 %, the calculator's value. For the SCALE SDF the figures are 1.13274 % (calculator) and 1.12924 % (SAUNA), and the difference again comes entirely from the $\bar{\nu}$ rule. Every other nuclide–reaction pair term agrees within 2.6×10^{-6} of the total variance. Five other Serpent files, with 2 to 175 groups and both reaction-mode settings, are read without error; there is no reference for their results.

3.4 GLS adjustment against a separate implementation

`glls_bias.py` is a separate implementation of the same GLS equations in Python and NumPy, written earlier in this project for work outside the browser. It was also generated with Claude, so it is not independent of the calculator's authors. The test takes 24 LEU-COMP-THERM and MIX-COMP-THERM configurations as experiments, with C and σ_C from their files, and the two CANDU-6 cells as applications. The measured values are made up: $E = C(1 + \varepsilon)$, with ε uniform within ±0.6 %, except for every seventh configuration, which gets +1.2 %, and σ_E from 0.1 to 0.4 %. The test computes $\mathbf{K}$ once with the calculator and gives the same matrix to both codes. It then runs nine fits: with and without σ_C, and with a correlation of 0.6 between two experiments, each with all experiments, after the χ^2 filter, and after the filter with one experiment marked *keep*. χ^2/N ranges from 3.51 to 3.75 with all experiments and from 1.09 to 1.18 after the filter, which removes 8 or 10 experiments. In every fit the two codes use the same experiments, the filter removes the same experiments in the same order, and every output agrees within 8.5×10^{-14} (relative): d, σ, the individual χ, $\Delta\chi^2$, C/E before and after, adjusted k, prior and posterior uncertainty, and c_k. The bias split by covariance block adds up to the bias of each application to within 10^{-9} of the bias.

3.5 GLS adjustment against SANDY in parameter space

SANDY [7] is a nuclear-data sampling and processing package by other authors. Its module `sandy.gls` updates parameters $\mathbf{x}$ and their covariance with the GLS formulas of Section 2.6, using a dense covariance matrix and a sensitivity matrix. That is the parameter-space form of the calculation, which the calculator never forms, so this comparison also tests the equivalence that the calculator relies on. The 26 systems of Section 3.4 have sensitivities to 500 nuclide–reaction pairs that have covariance data. A dense matrix of all 28,000 parameters would need about 6 GB, so the test keeps the 60 or the 150 pairs with the largest summed sensitivities (3,360 or 8,400 parameters) and drops the rest from both calculations. The test exports the covariance blocks and sensitivities of the calculator. A short Python script assembles the dense matrix $\mathbf{V}$, forms $\mathbf{K} = \mathbf{S}\,\mathbf{V}\,\mathbf{S}^{\mathrm{T}}$ with NumPy and calls `gls_update`, with σ_C included and a correlation of 0.6 between two experiments. The page's $\mathbf{K}$ agrees with the dense product within 4.2×10^{-15} of $\sqrt{(K_{ii}K_{jj})}$. χ^2/N (3.5147 and 3.5099) agrees with the sum of SANDY's `chi_square` terms. The experiments' adjusted C, $\mathbf{S}_e\,\mathbf{x}^{\mathrm{post}}$, agrees with the page's δ_e within 3.1×10^{-16}. The applications' bias, −978.6 and −991.7 pcm with 8,400 parameters, agrees within 1.4×10^{-14}, and their posterior uncertainty within 4.0×10^{-15} (relative). SANDY's other χ

helpers were not used. With these made-up E values the biases are large, about −0.9 % of k; they illustrate the arithmetic, not a real validation.

3.6 The page itself

The scripts above test `su-core.js`. To test the page that drives it, it was loaded in headless Microsoft Edge from a local copy of the site and driven by a script. The script loaded the three examples and 12 DICE configurations, read a C/E table with made-up E values and one experimental correlation from CSV, and ran the fit. The adjusted k (1.1267686), bias (−146.16 pcm) and prior and posterior uncertainty (0.806909 % and 0.609543 %) of the cooled CANDU-6 cell, and the experiment the filter removed, were the same as from `su-core.js` run directly in Node.js with the same inputs, in every digit shown. The same run checked that the CSV text is produced and that a changed input marks the results as out of date. This check was done once and is not part of the repository's scripts.

Other tests in the repository check the processing rules of Section 2.4 one by one, for example the element–isotope mapping with each library, the split by library for mixed libraries, and the list of sensitivities without covariance data. The uncertainty of the CANDU-6 cell (0.8069 % with the AMPX route) and its c_k with the voided cell (0.99919) are the values of the CANDU-6 study [11], 0.81 % and 0.999.

4. Discussion

What the comparisons show. The calculator's readers, its collapse and rules, and its sandwich, c_k and GLS arithmetic give the same numbers as the other implementations, to the precision of the data they share. For the uncertainty, the four-decimal storage of the correlations limits the agreement to a few parts in 10^5, far below the uncertainty of the covariances themselves. For the GLS, the agreement is at the level of floating-point rounding.

What they do not show. Most of the references were written in the same project, and two of them, the ICSBEP study's code and `glls_bias.py`, were generated by the same AI model as the calculator. An error of understanding common to both, rather than of coding, would not show. The references by other authors cover different parts. Serpent covers the reading of its own files. SAUNA covers the sandwich rule and the Serpent reader, but only for one model and the ENDF/B-VII.1 library. SANDY covers the GLS algebra given the same **V** and **S**, but not the processing rules that produce them. The NDaST comparison of the ICSBEP study is of category means. No comparison has yet been made with TSURFER or TSUNAMI-IP, which would test the rules and the GLS together against the reference tools of the method; SCALE 6.3 was not available for this work. Nor do code-to-code comparisons test the method itself. The GLS adjustment is first-order, takes the sensitivities as exact, and is only as good as the covariance data and the benchmark values, uncertainties and correlations entered. The theory manual [12] discusses these limits with examples, among them how strongly the bias depends on which experiments are used and on their similarity to the application.

Practical aspects. In Node.js, the 26 × 26 **K** of Section 3.4 takes 0.1 s, and the uncertainties of 120 configurations, including reading their files, take about 9 s per library. In the browser the time is dominated

by downloading the covariance files of the nuclides needed, about 0.9 MB for a system of uranium, water and zirconium (U-235, U-238, H-1, O-16 and three zirconium isotopes with the AMPX route).

5. Conclusions

- The S/U calculator computes the sandwich-rule uncertainty of k, its breakdown, c_k and a GLS (TSURFER-style) adjustment in the browser, from TSUNAMI or Serpent sensitivity files, with ENDF/B-VIII.1, ENDF/B-VII.1 or the user's own COVERX covariances. The files stay on the user's computer.
- Its uncertainties agree with the ICSBEP study's for 120 configurations and three libraries within 1.5×10^{-5} percentage points. Its COVERX and Serpent readers, and its breakdown against SAUNA's, agree to the precision of the stored or printed data.
- Its GLS adjustment agrees with a separate Python implementation within 8.5×10^{-14} in every output, including the χ^2 filter. It also agrees with SANDY's parameter-space `gls_update`, with dense covariance matrices of up to 8,400 parameters, within 1.4×10^{-14}.
- These are code-to-code comparisons of the same equations. A comparison with TSURFER and TSUNAMI-IP on common inputs is the next step.

Acknowledgements

The ENDF/B-VII.1 covariance library and the model files used in Section 3.3 are those distributed with SAUNA by A. Ryzhkov, whose Serpent reader the calculator follows. The parameter-space comparison uses SANDY's `gls` module. The DICE sensitivity profiles are those of the companion study [10].

About the authors and the status of this work

> Claude is an AI model developed by Anthropic. Ian Hill proposed the calculator and its GLS section, created the DICE sensitivity profiles used and provided the original SCALE 6.0 calculations they come from, and directed the work. Claude wrote the calculator, the comparison scripts and the separate Python GLS implementation, ran the comparisons, made the figures and tables, and wrote this article. The article has not been peer reviewed, and the results have not been independently reviewed or verified. Anthropic has not reviewed or endorsed this work.

Data and code availability

The calculator runs at hardtohit10.gitlab.io/su-tool.html. Its code, the comparison scripts (`tools/test_su_core.mjs`, `test_su_tool.mjs`, `test_su_serpent.mjs`, `test_su_gls.mjs`, `test_su_gls_sandy.mjs`), `glls_bias.py` and the script that writes SAUNA's breakdown are in the repository [8], under the MIT licence. The theory manual of the GLS adjustment is the repository's wiki [12].

The comparisons need data that are not in the repository: the DICE sensitivity files (Sections 3.1, 3.4, 3.5), the NJOY-route COVERX file (Section 3.2, from the covariance releases [9]), SAUNA (Section 3.3) and SANDY (Section 3.5). No benchmark values from the ICSBEP Handbook are used or published; the measured values in the GLS comparisons and figures are made up.

References

[1] B.L. Broadhead, B.T. Rearden, C.M. Hopper, J.J. Wagschal, C.V. Parks, "Sensitivity- and Uncertainty-Based Criticality Safety Validation Techniques", *Nuclear Science and Engineering* 146 (2004) 340–366.

[2] M.L. Williams, B.L. Broadhead, M.A. Jessee, J.J. Wagschal, R.A. Lefebvre, "TSURFER: An adjustment code to determine biases and uncertainties in nuclear system responses by consolidating differential data and benchmark integral experiments", in *SCALE Code System*, ORNL/TM-2005/39, Oak Ridge National Laboratory.

[3] OECD/NEA WPEC Subgroup 33, "Methods and issues for the combined use of integral experiments and covariance data", NEA/NSC/WPEC/DOC(2013)445, OECD Nuclear Energy Agency (2013).

[4] *SCALE Code System*, ORNL/TM-2005/39, Oak Ridge National Laboratory: the TSUNAMI-IP and TSURFER chapters.

[5] Nuclear Data Sensitivity Tool (NDaST), OECD Nuclear Energy Agency, https://oecd-nea.org/jcms/pl_32450/nuclear-data-sensitivity-tool-ndast.

[6] SAUNA, a Python package for sensitivity and uncertainty analysis, A. Ryzhkov, https://github.com/AARyzhkov/sauna.

[7] L. Fiorito, G. Žerovnik, A. Stankovskiy, G. Van den Eynde, P.E. Labeau, "Nuclear data uncertainty propagation to integral responses using SANDY", *Annals of Nuclear Energy* 101 (2017) 359–366; SANDY 1.2, https://github.com/luca-fiorito-11/sandy.

[8] *S/U calculator*, source code and tests, https://gitlab.com/hardtohit10/su-calculator; the calculator: https://hardtohit10.gitlab.io/su-tool.html.

[9] Claude and Ian Hill, "ENDF/B-VIII.1 neutron covariances in the SCALE 56-group structure: two processing routes, NJOY and AMPX, and their effect on criticality uncertainties", preprint, https://hardtohit10.gitlab.io/cov56-paper.html (2026), https://doi.org/10.5281/zenodo.23245168; libraries: https://gitlab.com/hardtohit10/endfb81-cov56.

[10] Claude and Ian Hill, "Nuclear-data uncertainty of k_{eff} for 2,843 ICSBEP configurations with ENDF/B-VIII.1 covariances", preprint, https://hardtohit10.gitlab.io/icsbep-uncertainty-paper.html (2026), https://doi.org/10.5281/zenodo.23245174.

[11] Claude and Ian Hill, "Similarity of a CANDU-6 lattice cell to criticality benchmarks for k-infinity and coolant void reactivity", preprint, https://hardtohit10.gitlab.io/candu-similarity-

paper.html (2026), https://doi.org/10.5281/zenodo.23245176.

[12] *GLS adjustment in the S/U calculator: theory manual*, wiki of the su-calculator repository, https://gitlab.com/hardtohit10/su-calculator/-/wikis/GLS-theory-manual.

[13] *S/U calculator: how your files stay private*, https://hardtohit10.gitlab.io/su-privacy.html.

[14] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, T. Kaltiaisenaho, "The Serpent Monte Carlo code: Status, development and applications in 2013", *Annals of Nuclear Energy* 82 (2015) 142–150.

[15] M. Aufiero, A. Bidaud, M. Hursin, J. Leppänen, G. Palmiotti, S. Pelloni, P. Rubiolo, "A collision history-based approach to sensitivity/perturbation calculations in the continuous energy Monte Carlo code SERPENT", *Annals of Nuclear Energy* 85 (2015) 245–258.

[16] *International Handbook of Evaluated Criticality Safety Benchmark Experiments*, NEA/NSC/DOC(95)03, OECD Nuclear Energy Agency.

[17] *DICE: Database for the International Criticality Safety Benchmark Evaluation Project Handbook*, User's Manual, OECD Nuclear Energy Agency, https://www.oecd-nea.org/science/wpncs/icsbep/documents/DICE_users_manual.pdf.

By Claude (Anthropic AI model) and Ian Hill · not peer reviewed

Visits are counted with GoatCounter, without cookies or personal data Posted 29 October 2026