

ARIADNE – a program estimating covariances in detail for neutron experiments

Denise Neudecker*

Los Alamos National Laboratory, Los Alamos, NM, USA

Received: 16 November 2017 / Received in final form: 5 February 2018 / Accepted: 4 May 2018

Abstract. The python program ARIADNE is a tool developed for evaluators to estimate detailed uncertainties and covariances for experimental data in a consistent and efficient manner. Currently, it is designed to aid in the uncertainty quantification of prompt fission neutron spectra, and was employed to estimate experimental covariances for CIELO and ENDF/B-VIII.0 evaluations. It provides a streamlined way to estimate detailed covariances by (1) implementing uncertainty quantification algorithms specific to the observables, (2) defining input quantities for typically encountered uncertainty sources and correlation shapes, and (3) automatically generating plots of data, uncertainties and correlations, GND formatted XML and plain text output files. Covariances of the same and between different datasets can be estimated, and tools are provided to assemble a database of experimental data and covariances for an evaluation based on ARIADNE outputs. The underlying IPython notebook files can be easily stored, including all assumptions on uncertainties, leading to more reproducible inputs for nuclear data evaluations. Here, the key inputs and outputs are shown along with a representative example for the current version of ARIADNE to illustrate its usability and to open a discussion on how it could address further needs of the nuclear data evaluation community.

1 Introduction

At the “CW2017” workshop, progress on covariance estimation for and evaluation of prompt fission neutron spectra (PFNS) was shown. One key point was that estimating experimental covariances in detail versus extracting total uncertainties from EXFOR [1,2] and using simplifying assumptions for the total correlations may result in significantly different evaluated PFNS, benchmark calculations and associated uncertainties. For instance, the effective multiplication factor, k_{eff} , of the Jezebel critical assembly changes by nearly 200 pcm if a detailed versus a simplified experimental covariance matrix is used as input for an evaluation of the ^{239}Pu PFNS induced by neutrons of 500 keV [3]. This change is significant, given that a difference of 270 pcm in k_{eff} of a Plutonium system can lead from a controlled to an uncontrolled reaction.

Given that a detailed uncertainty estimate for experimental data may significantly impact evaluations and application calculations, the python program ARIADNE was developed to estimate experimental covariances for evaluation purposes in detail in a consistent and streamlined manner. This program was developed concurrently and

partially in support of work for the IAEA coordinated research project on PFNS [4], and was subsequently used to provide input for CIELO [5] and ENDF/B-VIII.0 [6] ^{235}U PFNS evaluations [7]. ARIADNE was designed such that documenting the experimental covariances is straightforward. All assumptions in the estimation process are explicitly stored in ARIADNE input decks and the output is given in GND format [8] (As an example, the documentation for experimental ^{235}U PFNS covariances is given in Ref. [9]). This feature should support guaranteeing the reproducibility of (evaluated) data – a fundamental principle of science which is often violated as rarely all input used to generate evaluated data is readily available for the whole community. Therefore, ARIADNE could provide input for the WPEC Subgroup 44 on “Investigation of Covariance Data in General Purpose Nuclear Data Libraries” [10] on guidelines how evaluators can document their evaluations.

ARIADNE can be currently used to estimate experimental covariances for PFNS. An extension to estimate covariances for measured neutron-induced fission cross-sections is in progress. Section 2 shows the input necessary for estimating the total covariances of PFNS with ARIADNE, the automatically generated output files and how it can be used to estimate covariances between different experimental datasets and assemble databases from multiple experimental datasets for evaluation purposes. The example in Section 3 will illustrate how

* e-mail: dneudecker@lanl.gov

ARIADNE can be used to estimate a total covariance matrix for one particular dataset. Section 4 summarizes and provides an outlook to future developments.

2 General Description of ARIADNE

2.1 Input

The ARIADNE input parameters, assumptions and data can be transparently stored in an IPython or Jupyter notebook. From this notebook, ARIADNE will be run and the output plots and data files will be automatically generated. These notebooks can be easily exchanged between evaluators and can be converted to LaTeX or pdf format and thus facilitate the documentation of the input assumptions for estimating experimental covariances.

For estimating total covariances for measured PFNS with ARIADNE, it needs to be specified whether the data are “shape”, “shape ratio” or “shape ratio calibration” data. In a shape measurement, the detector efficiency is determined explicitly, while in a ratio experiment the detector efficiency cancels as the measured ratio data were measured with the same detector. In a ratio calibration measurement, the detector efficiency is obtained by a measurement of the monitor PFNS and using a numerical representation for this measured PFNS. It is important to specify which measurement type PFNS data belong to as different uncertainty sources need to be considered for specific measurement types and the actual algorithms to estimate total covariances differ as outlined in Section III.M of reference [4]. For instance, if a time resolution of 1 ns for a time-of-flight (TOF) length of 1 m is converted to an uncertainty relative to the PFNS using a Maxwellian temperature of 1.33 and 1.42 MeV for the investigated and monitor isotope, the resulting uncertainties are distinctly different in Figure 1 depending on what data type is used. The time resolution uncertainty relative to the PFNS substantially reduces for shape ratio data compared to shape data while the time resolutions of the investigated and monitor isotope measurement need to be considered for shape ratio calibration data leading to increased uncertainties compared to shape data.

The covariance estimation algorithms presented in Section III.M of reference [4] were implemented in ARIADNE to estimate total covariances for PFNS experimental data. The algorithms are summarized in the Appendix of this paper for the sake of completeness and to link the ARIADNE input quantities defined below with the algorithms in reference [4].

The commands to invoke the total covariance estimation are “PFNS_shape(*general_info*, *data*, *trsl*, *tof_length*, *maxw_t_iso*, *unc_iso*)”, “PFNS_ratio(*general_info*, *data*, *trsl*, *tof_length*, *reference*, *maxw_t_iso*, *maxw_t_ref*, *unc_iso*, *unc_ref*)” and “PFNS_ratiocalibration(*general_info*, *data*, *trsl*, *tof_length*, *reference*, *maxw_t_iso*, *maxw_t_ref*, *unc_iso*, *unc_ref*)” with the following input quantities:

general_info: This dictionary contains general information specifying the observable, naming the dataset, the output files/folders and some documentation. For the following keys, values need to be provided:

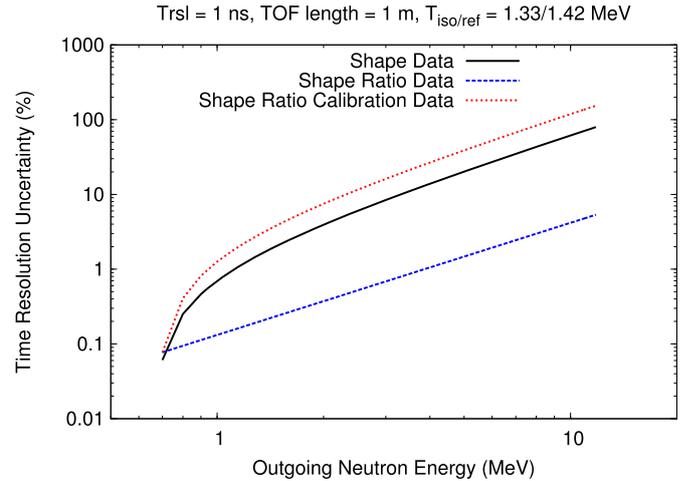


Fig. 1. The total time resolution uncertainty relative to the PFNS is compared for shape, shape ratio and shape ratio calibration data for a time resolution of 1 ns, a TOF length of 1 m for all measurements and Maxwellian temperatures of 1.33 and 1.42 MeV for the investigated and the monitor isotope, respectively.

- name: a string naming the experiment,
- isotope: a string specifying the isotope under investigation,
- quantity: for the current class this is the string “PFNS”,
- reaction: a string, either “n,f” for neutron-induced PFNS, or “sf” for spontaneous fission,
- output_file: a string specifying the XML file name and path,
- output_folder: a string specifying the path to the folder where the output plots should be stored,
- documentation: a string including documenting information for the measurement, such as the EXFOR-entry number, literature references and general comments on the uncertainty estimate and assumptions made in it.

data: This dictionary contains the data of the dataset with the following keys:

- no_einc: integer specifying the number of different incident neutron energies for which data are given,
- einc: float array of the length of the data containing the incident neutron energies,
- einc_unit: string specifying the unit of einc, e.g., “eV”,
- eout: float array of the length of the data containing the outgoing neutron energies,
- eout_unit: string specifying the unit of eout, e.g., “MeV”,
- eout_unc: float array of the length of the data containing uncertainties associated with eout,
- eout_unc_unit: string specifying the unit of eout_unc, e.g., “%”,
- eout_unc_type: string containing the type of correlations between eout_unc with options described below,
- eout_unc_type_arg: a dictionary containing information necessary to invoke specific types of eout_unc_type as specified below,
- values: float array of the data,
- values_unit: string of units of values.

Currently, five different pre-defined shapes can be used to assign correlation matrices for partial uncertainties with the keys “eout_unc_type” in the dictionaries data and unc_ref or “type” in the dictionaries unc_iso and unc_ref as defined in this section. All pre-defined correlation matrices satisfy mathematical criteria imposed on correlation matrices. Their diagonal entries are one and their off-diagonal entries assume values between -1 and 1 . The matrices are all symmetric and positive semi-definite. The value “Diagonal” returns a correlation matrix with all off-diagonal entries zero, while for “Positive_fully” all off-diagonal entries are one. “Constant” results in off-diagonal entries all assuming the same value within the interval $[0,1]$. This value is specified within the dictionary eout_unc_type_arg in the dictionaries data and unc_ref or type_arg in the dictionaries unc_iso and unc_ref with the key “damp_term”. For damp_term, a one-dimensional array of floats with length n is provided with each entry corresponding to the n different partial uncertainties provided. The value “Gaussian” provides a correlation matrix with shape:

$$\text{Cor}_{i,j} = \exp \left\{ - \left[\frac{c(E_{\text{out}}^i - E_{\text{out}}^j)}{\max(E_{\text{out}}^i, E_{\text{out}}^j)} \right]^2 \right\},$$

where the constant damping factor c with values between 0 and 1 is defined with the key damp_term in the dictionaries eout_unc_type_arg or type_arg. In addition to c , the energies E_{out} of the PFNS need to be provided with the keyword “eout” with the same dictionaries. The correlation matrix “Gaussian-Anticorrelated” is very similar to the one termed Gaussian,

$$\text{Cor}_{i,j}^a = \text{Cor}_{i,j} A_{i,j}(E_t),$$

except for the matrix $A_{i,j}$ with matrix entries of 1 if the energies E_{out}^i and E_{out}^j are both either larger or smaller than E_t and -1 otherwise. The values of c and E_{out}^i are provided as described above. In addition, E_t is provided with the key “eout-turningpoint” with the same dictionaries.

trsl: This dictionary contains all information necessary to estimate uncertainties due to the finite time resolution for a PFNS measurement with keys:

- value: a float with the time resolution, can be a float array of the same length as the data if multiple time resolutions are given for one dataset,
- unit: string of unit of value,
- ref_value [only ratio and calibration measurements]: a float with the time resolution of the monitor measurement, can be a float array of the same length as data if multiple time resolutions are given for one dataset,
- ref_unit [only ratio and calibration measurements]: string of unit for ref_value.

tof_length: This dictionary contains all information needed to include TOF length uncertainties in a total PFNS covariance matrix. Even if the TOF length uncertainty itself is zero, a TOF length needs to be provided to estimate time resolution uncertainties. The keys of this dictionary are:

- value: a float with the TOF length of the measurement, can be a float array of the same length as the data if multiple TOF lengths were used,
- value_unit: string of unit of value,
- unc: float of TOF length uncertainty, can be a float array of the same length as the data if multiple TOF length uncertainties are given,
- unc_unit: string of unit of unc,
- ref_value [only ratio calibration measurements]: a float with the TOF length of the monitor measurement, can be a float array of the same length as data if multiple TOF lengths are given for one dataset,
- ref_value_unit [only ratio calibration measurements]: string of unit for ref_value,
- ref_unc [only ratio calibration measurements]: a float with the TOF length uncertainties of the monitor measurement, can be a float array of the same length as data if multiple TOF length uncertainties are given for one dataset,
- ref_unc_unit [only ratio calibration measurements]: string of unit for ref_unc.

maxw_t_iso: This dictionary contains information on the Maxwellian temperature fitted to the measured PFNS. This information is necessary to convert TOF length uncertainties or a time resolution in length units into uncertainties relative to the PFNS which is an essential step towards generating a total PFNS covariance matrix. The keys are:

- value: a float with the Maxwellian temperature fitted to the measured PFNS, can be a float array of the same length as the data if multiple Maxwellian temperatures are given. Multiple Maxwellian temperatures are usually provided for measurements at different einc,
- unit: string of unit of value, e.g., “MeV”.

maxw_t_ref: [only ratio and calibration measurements] A dictionary containing the same information as maxw_t_iso except for the monitor isotope.

unc_iso: This dictionary contains partial uncertainties for the PFNS measurements as well as information necessary to estimate their correlation matrices. The keys are:

- values: a two-dimensional array of floats with as many columns n as uncertainty sources given and as many rows as data points,
- units: string array of length of columns n in values specifying the unit of each uncertainty source in values. Entry one corresponds to column one in values, and so on,
- type: string array of length of n in values specifying the type of correlation matrix for each uncertainty source described above. Entry one corresponds to column one in values, and so on,
- type_arg: a dictionary containing information necessary to invoke specific types of correlations related to the key type as specified above.

unc_ref: [only ratio and calibration measurements] A dictionary containing all the information given in unc_iso for the monitor isotope instead of the isotope in questions as well as the following keys:

- `eout_unc`: float array of the length of the data containing the energy uncertainties of the monitor measurement,
- `eout_unc_unit`: string of unit of `eout_unc`, e.g., “%”,
- `eout_unc_type`: string containing the type of correlations between `eout_unc` with options described above,
- `eout_unc_type_arg`: a dictionary containing information necessary to invoke specific types of `eout_unc_type` as specified above.

reference: [only ratio and calibration measurements]

This dictionary contains the information necessary to specify the monitor isotope, reaction and the nuclear data which should be used to convert ratio PFNS to PFNS. The keys are:

- `isotope`: string specifying the monitor isotope,
- `quantity`: string specifying the observable measured with the monitor isotope. This key will have the value “PFNS” here,
- `reaction`: a string, either “n,f” for neutron-induced PFNS, or “sf” for spontaneous fission,
- `identifier`: a string identifying the dataset that can be used. Available datasets are listed in the function `Manage_ReferenceData.py`. The data of Mannhart [11,12] with identifier “Mannhart-Pointwise” were frequently used for establishing a database of ^{235}U PFNS [9].

2.2 Output

ARIADNE automatically produces the following output in the folder specified with the variable `output_folder`:

- an XML output file with a filename specified with the variable `output_file` is provided in GND format [8]. This file contains the information provided in the variable documentation, the incident and outgoing neutron energy, the total relative uncertainty, the PFNS and the total correlation matrix. If the original data are given in ratio to a monitor PFNS, the ratio data are converted to PFNS data using the nuclear data specified in “`identifier`” for the monitor PFNS;
- the simple text-file “`Partial_Unc.dat`” contains incident neutron energies, outgoing neutron energies, PFNS, total and all partial uncertainties relative to the PFNS;
- a simple text-file “`TotalCor.dat`” containing the correlation matrix associated with the total relative uncertainties in `Partial_Unc.dat`;
- plots of the PFNS for all incident neutron energies, partial and total relative uncertainties for all incident neutron energies, total and partial uncertainty correlation matrices are automatically generated. If the provided PFNS are ratio or ratio calibration data, also plots showing the interpolated monitor PFNS compared to the original ones are provided.

2.3 Covariances between experiments

The estimation of covariances between PFNS of different experiments uses current ARIADNE capabilities for estimating PFNS covariances for single experiments. In the future, a module will be developed that suggest possible

matches leading to correlations between uncertainty sources of two experiments and the possibility to select a correlation factor. Right now, covariances between different experiments can be estimated with ARIADNE by invoking a total covariance estimation procedure first for both measurements in the same IPython-notebook. Then, correlations between different partial uncertainty sources of two experiments are estimated by accessing the partial covariances of each experiment and cross-correlating them using functions of ARIADNE. A text output file with the covariances between those experiments is produced which can be used as input to assemble an experimental database for an evaluation.

2.4 Assembling a database as input for evaluations

Two steps need to be performed when assembling a database of experimental data and covariances for a PFNS evaluations: As a first step, the experimental PFNS have to be scaled with a constant factor for each experimental dataset at one incident neutron energy with respect to either a model curve or general basis function used as the prior for the evaluation [13] as all PFNS data are treated as shape data for evaluation purposes. This step ensures that the χ^2 between experimental data and the prior mean values only reflects differences in the actual shape and not due to the scaling of the data. In the second step, the experimental database is assembled from different datasets and their associated covariances and cast into a format readable by evaluation codes used for evaluations of reference [7].

In ARIADNE, the function “`Assemble_Plot_PFNS-Database (experimental_data, reference_data, output_folder, {}, perform_tasks)`” performs these two steps with the following input observables:

- **experimental_data**: a dictionary with the keys “`paths`” (providing an array of strings specifying the XML output files produced by ARIADNE which should be included in the database), “`Eout_lower`” and “`Eout_upper`” (one-dimensional arrays of floats with lengths of number of datasets specified in `paths` providing the energy range of the data used for scaling. If 0 is given, the full energy range of the data is used for scaling),
- **reference_data**: a dictionary with the keys “`path`” (string identifying the dataset to which experimental data are scaled to), “`Tmaxw`” (a float specifying the temperature of the Maxwellian which is used to plot the PFNS in ratio to it), “`Tmaxwunit`” (a string with the unit of `Tmaxw`), “`Name`” (a string used to name the reference data in the legend of the plots), “`Isotope`” (a string identifying the isotope in the ZZZAAA notation),
- **output_folder**: string with name of the output folder where data files and plots should be stored. If none is given, they will be stored in the `src` folder of ARIADNE,
- **cross_correlations**: a dictionary containing the paths to the cross-correlations. Right now, only an empty dictionary can be given which leads to the assumption of 0 correlation between experiments. Cross-correlations are currently added with another sub-program which is being reworked into `Assemble_Plot_PFNSDatabase`,

```

Documentation = "EXFOR No. = None. Journal article = J.P. Lestone and E.F. Shores, NDS 119, 213 (2014); \
    J. P. Lestone and E.F. Shores, Los Alamos National Laboratory Report LA-UR-14-24087 (2014); \
    J. P. Lestone and E.F. Shores, Los Alamos National Laboratory Report LA-UR-13-21239 (2013); \
    Data and uncertainties were taken directly from LA-UR-14-24087. These data contain a detailed \
    description of uncertainties affecting this data set. J.P. Lestone also gives hints on the \
    correlations and they were chosen accordingly. No definite TOF length or Maxwellian temperature \
    is provided. However, time resolution uncertainties are given in PFNS space, so the latter \
    two quantities are not needed for this uncertainty analysis. Assumptions had to be made on \
    the type of correlations for unc. 2, 3, 4, 5 and 8 of LA-UR-14-24087. All were assumed to have \
    Gaussian shape with damping parameters of [0.2,0.5,0.5,1.0,1.0]."

general_info = {'name': 'Lestone', 'isotope': 'U-235', 'quantity': 'PFNS', 'reaction': 'n,f', \
    'output_file' : '/home/dneudecker/Experiments/PythonProgram/WorkDir/U-235/PFNS_Lestone/\
    PFNS_Lestone.xml', \
    'output_folder' : '/home/dneudecker/Experiments/PythonProgram/WorkDir/U-235/PFNS_Lestone/', \
    'documentation' : Documentation}

trsl = {'value': 0.0, 'unit': 'ns'}
tof_length = {'value': 0.1, 'value_unit': 'cm', 'unc': 0.0, 'unc_unit': 'cm'} # unknown value
# not needed because time resolution uncertainty is given already converted relative to PFNS
maxw_t_iso = {'value': 1.32, 'unit': 'MeV'}
# not needed for uncertainty analysis because time resolution uncertainty is given already converted relative to PFNS

unc_iso = {'values':array([U235_statunc,U235_geometry,U235_nonHcont,U235_fusionpeak,U235_fusionmagnitude,\
    U235_trsl,U235_mcunc,U235_fusionncor]).transpose(), \
    'units':['%', '%', '%', '%', '%', '%', '%', '%', '%'], \
    'type':['Diagonal', 'Gaussian', 'Gaussian', 'Gaussian', 'Positive_fully', 'Diagonal', 'Gaussian'], \
    'type_arg':{'eout':array(Eout), 'damp_term':[1.0,0.2,0.5,0.5,1.0,1.0,1.0,1.0],}}
data = {'no_einc': 1, 'einc': array(Einc), 'einc_unit': 'MeV', 'eout': array(Eout), 'eout_unit': 'MeV', \
    'eout_unc': array(Eunc), \
    'eout_unc_unit': '%', 'eout_unc_type': 'Positive_fully', 'eout_unc_type_arg': {}, \
    'values':array(Data), 'values_unit': "1/MeV" }

LestonePFNS = CP.PFNS_shape(general_info,data,trsl,tof_length,maxw_t_iso,unc_iso)

```

Fig. 2. An ARIADNE input deck for estimating total covariances for Lestone et al. data is shown.

– **perform_tasks**: an array of strings specifying which tasks the program should execute. “Plot_Experimental-Data_vs_Reference” plots experimental data versus reference data in the folder `output_folder`. “Write_ScaledData_ToOneFile” writes incident neutron energies, outgoing neutron energies, the scaled experimental data and relative uncertainties to a text file named “DataBase_Single_Rescaled_Experiments.dat” in the folder `output_folder`. “Write_DatabaseOutputfile” writes database to the file “DataBase_ForEvaluation.dat” into the folder `output_folder` in a format readable by an evaluation program used for reference [7] but can be extended to other formats if needed.

3 Example

As example, it is shown how total covariances for the ^{235}U PFNS analyzed by Lestone et al. [14,15] are estimated using ARIADNE. This particular dataset was chosen as only nine data points are provided leading to a relatively small XML-output file which can be easily shown here. Also, the uncertainty sources listed in the input deck shown in Figure 2 can be easily aligned with partial uncertainties given in reference [15] for Lestone et al. PFNS.

Lestone et al. PFNS are shape data and, thus the ARIADNE function `PFNS.shape()` is called in the input deck in Figure 2 with function arguments as described in Section 2. For instance, it is specified that Lestone et al. data are ^{235}U PFNS and a documentation is provided with references to the literature used for the uncertainty estimate and assumptions made for the estimation process. The partial uncertainty sources and their correlation coefficients are specified with the dictionary “`unc_iso`”. Eight different uncertainty sources are provided as ARIADNE input following closely reference [15]. For each partial uncertainty source, a vector of partial uncertainties and information on the shape (e.g., most of them have Gaussian shape, except the fully correlated time resolution uncertainties and diagonal statistical uncertainties). The dictionary “`trsl`” contains a zero time resolution as Lestone et al. provided already time resolution uncertainties relative to the PFNS which are considered within the dictionary “`unc_iso`”. The dictionary `trsl` must still be provided, even if the time resolution itself is zero, to remind the user that such an uncertainty would need to be accounted for when estimating uncertainties of measured PFNS. The same reasoning applies why a dictionary “`tof_length`” with zero TOF length uncertainty needs to be provided although this uncertainty is given as partial

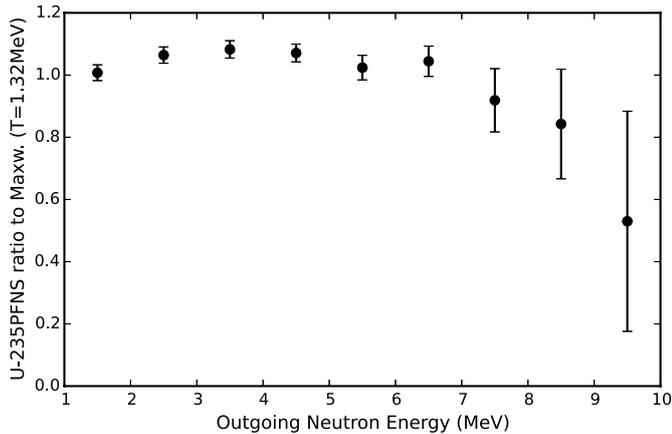


Fig. 3. A plot of Lestone et al. PFNS data with total 1-sigma uncertainties estimated by ARIADNE using the input deck in Figure 2 is shown as automatically generated by the program.

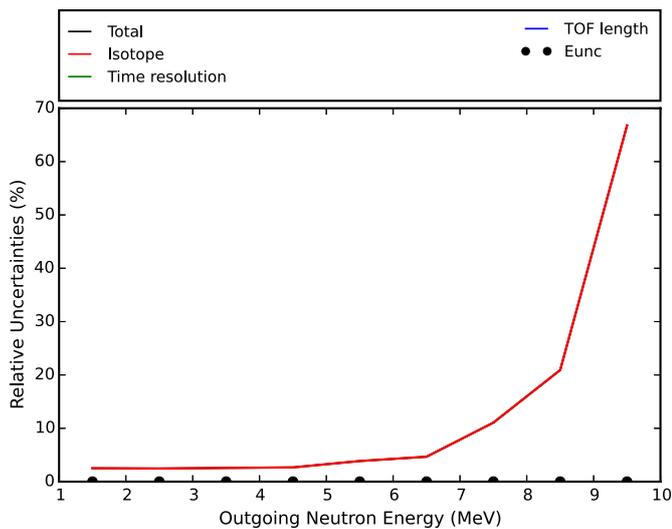


Fig. 4. A plot of relative uncertainties given and estimated for the data of Lestone et al. is shown as automatically generated by ARIADNE.

uncertainty relative to the PFNS for Lestone et al. data. ARIADNE was programed that way as most PFNS measurements provide non-zero time resolution and TOF length uncertainty values, and by having to make the deliberate choice that they are zero, it is less likely that these uncertainty sources are forgotten during the estimation process.

Figures 3–5 show plots automatically produced by ARIADNE to visualize the estimated covariances. The PFNS values shown in Figure 3 are given in ratio to a Maxwellian with a temperature specified in `maxw_t_iso` along with their total 1-sigma uncertainties estimated using ARIADNE. Figure 4 shows relative uncertainties for four different—partially combined—uncertainty sources compared to the total uncertainties and, thus, helps in roughly identifying the major uncertainty sources of this measurement. All uncertainty sources for this particular

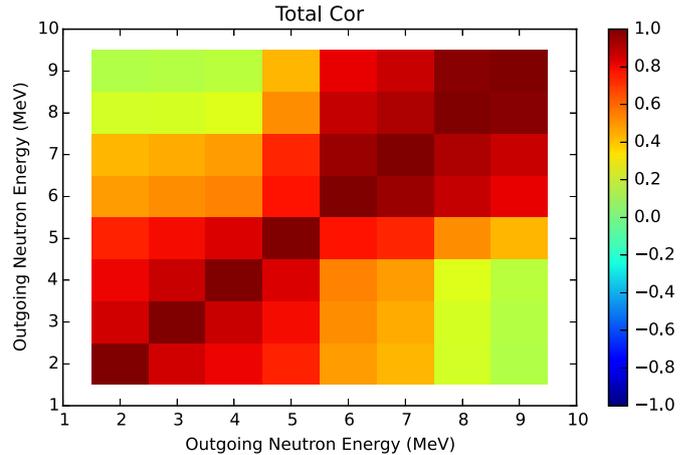


Fig. 5. A plot of the total correlation matrix estimated for Lestone et al. PFNS using ARIADNE with the input deck in Figure 2 is shown as automatically generated by the program.

measurement, including the time resolution and TOF length uncertainty, are given relative to the PFNS and are combined into the uncertainties titled “Isotope”. Therefore, the total uncertainties are the same as the “Isotope” uncertainties for the case of Lestone et al. PFNS. The total correlation matrix in Figure 5 is strongly positively correlated for neighboring energy bins while energy bins far apart are less strongly correlated. This behavior of the correlations was expected given that most uncertainty sources listed in the input deck were assumed to have a Gaussian type correlation. More plots were produced (e.g., PFNS not in ratio to a Maxwellian, PFNS and relative uncertainties on a logarithmic scale and correlation matrices for each uncertainty source listed in the legend of Fig. 4), but are not shown here as they would not add more information on this dataset. For instance, the correlation matrix for the uncertainty source termed “Isotope” within ARIADNE is the same as that one of Figure 5 given that all uncertainty sources are given relative to the PFNS for Lestone et al. data.

Apart from the figures, a GND formatted XML output file as shown in Figure 6 is produced. It not only provides incident and outgoing neutron energies, PFNS, total relative uncertainties and correlations, but also unambiguously specifies the data as experimental ^{235}U PFNS and provides a documentation.

4 Summary and future developments

The program ARIADNE to estimate covariances in detail for neutron experiments was presented. This program was designed for evaluators to consistently estimate covariances of single datasets, between different experimental datasets and assemble a database from these estimated covariances for evaluation purposes.

Right now, ARIADNE was developed to estimate covariances for PFNS experiments. Currently, no openly available tool exists specifically designed for detailed covariance estimates for experimental PFNS data while IAEA tools connected to EXFOR exist which help in

```

<reactionSuite formatVersion="GND 1.7" library="Estimated experimental data and covariances" projectile="n" projectileFrame="lab" target="U-235"
version="0.1">
  <styles>
    <exp_unc date="2016-02-09" name="estimated experimental uncertainty"/>
  </styles>
  <documentations>
    <documentation name="Lestone U-235 PFNS">
      EXFOR No. = None. Journal article = J.P. Lestone and E.F. Shores, NDS 119, 213 (2014); J. P. Lestone and E.F. Shores, Los Alamos National
      Laboratory Report LA-UR-14-24087 (2014); J. P. Lestone and E.F. Shores, Los Alamos National Laboratory Report LA-UR-13-21239 (2013); Data and
      uncertainties were taken directly from LA-UR-14-24087. These data contain a detailed description of uncertainties affecting this data set. J.P.
      Lestone also gives hints on the correlations and they were chosen accordingly. No definite TOF length or Maxwellian temperature is provided.
      However, time resolution uncertainties are given in PFNS space, so the latter two quantities are not needed for this uncertainty analysis.
      Assumptions had to be made on the type of correlations for unc. 2, 3, 4, 5 and 8 of LA-UR-14-24087. All were assumed to have Gaussian shape
      with damping parameters of [0.2,0.5,0.5,1.0,1.0].
    </documentation>
  </documentations>
  <reaction ENDF_MT="18" fissionGenre="total" label="45" outputChannel="n[emissionMode:'prompt']">
    <outputChannel genre="NBody">
      <product emissionMode="prompt" label="n" name="n">
        <distributions>
          <uncorrelated productFrame="lab" style="exp_unc">
            <multiD_XYs dimension="2" label="energy">
              <axes>
                <axis index="2" label="energy_in" unit="MeV"/>
                <axis index="1" label="energy_out" unit="MeV"/>
                <axis index="0" label="P(energy_out|energy_in)" unit="1/MeV"/>
              </axes>
              <XYs index="0" value="1.5">
                <values length="18">
                  1.5 0.2947 2.5 0.1884 3.5 0.1063 4.5 0.0559 5.5 0.0277 6.5 0.0144 7.5 0.00638 8.5 0.00292 9.5 0.00091
                </values>
                <uncertainties>
                  <uncertainty relation="relative" type="variance">
                    <values length="18">
                      1.5 0.0251594912508 2.5 0.0246576560119 3.5 0.0257487863792 4.5 0.0267394839142 5.5 0.0386393581727 6.5 0.0467332857822 7.5
                      0.110661646472 8.5 0.209176958578 9.5 0.667791883748
                    </values>
                  </uncertainty>
                </uncertainties>
              </XYs>
              <covarianceMatrix type="cor">
                <matrix columns="9" form="symmetric" rows="9">
                  1.0 0.854939982397 0.8121131927 0.745165160248 0.489521929486 0.430134591248 0.204050245055 0.125372345817 0.0452245456228 1.0
                  0.871826193402 0.796091978258 0.517759675766 0.455995607388 0.217323044927 0.133978559357 0.0483299793472 1.0 0.838613453973
                  0.544459022595 0.491343608227 0.241847897355 0.153385616144 0.0565450364901 1.0 0.77514728829 0.735246808501 0.52227291232
                  0.434347096483 0.336543786797 1.0 0.947684111579 0.875616614449 0.820027881687 0.757006613742 1.0 0.91839453219 0.87081020741
                  0.801987240852 1.0 0.981316421784 0.952427713379 1.0 0.981654935114 1.0
                </matrix>
              </covarianceMatrix>
            </multiD_XYs>
          </uncorrelated>
        </distributions>
      </product>
    </outputChannel>
  </reaction>
</reactionSuite>

```

Fig. 6. The GND formatted XML output file automatically generated by ARIADNE with the input deck of Figure 2 is shown. It contains a documentation, the incident and outgoing neutron energies, the data, total relative uncertainties and associated correlations.

estimating covariances for neutron induced cross-sections [16]. ARIADNE streamlines the detailed uncertainty estimate – usually a time and labor-intensive procedure – by providing algorithms for estimating PFNS uncertainties for typical PFNS measurement types [4], converting data given as ratio to a monitor PFNS into PFNS data, providing shapes of typical correlation matrices for partial uncertainties and automatically producing from this information total covariances, plots and GND formatted XML and plain text output files. The program relies on input for partial uncertainty sources, information on their correlation as well as important parameters of a PFNS measurement (e.g., time resolution, TOF length). Hence, while ARIADNE facilitates and fastens the uncertainty process, the evaluator still has to extract and identify uncertainties from EXFOR entries and the literature of a dataset.

ARIADNE is written in python and its input files can be stored as IPython or Jupyter notebooks. These input files can be easily exchanged between evaluators and converted conveniently to LaTeX or pdf format for documentation purposes. Given the straightforward possibilities to document the experimental database entering an evaluation,

ARIADNE can help in making evaluations more reproducible given that experimental covariances entering evaluations are still rarely documented.

Currently, ARIADNE is being updated to estimate total covariances for neutron-induced fission cross-section data. It is planned that an additional layer is added to ARIADNE which requests typically uncertainty sources encountered in PFNS and fission cross-section measurements as listed in references [4,17]. This additional layer should help evaluators in identifying missing uncertainty sources for a particular experiment. A range of typical uncertainties might be provided to help estimating missing uncertainties. Also, a function identifying possible cross-correlations between uncertainty sources of two different experiments will be developed. Once, these parts are finished, steps will be undertaken to release the LANL ARIADNE code as open source program to the nuclear data evaluation community.

This work was carried out under the auspices of the NNSA of the U.S. Department of Energy at LANL under Contract No. DE-AC52-06NA25396.

Appendix. PFNS covariance algorithms implemented in ARIADNE

The algorithm to estimate total covariances $\text{Cov}(N_k, N_j)$ associated with PFNS values N_k and N_j at lattice points k and j are described in detail in Section III.M of reference [4] and are summarized here briefly:

$$\frac{\text{Cov}(N_k, N_j)}{N_k N_j} = \sum_n \delta_k^{n,i} \delta_j^{n,i} \text{Cor}_{k,j}^{n,i} + C_{k,j}^E + C_{k,j}^t + C_{k,j}^l + \delta_k^{ND} \delta_j^{ND} \text{Cor}_{k,j}^{ND} + \sum_m \delta_k^{m,r} \delta_j^{m,r} \text{Cor}_{k,j}^{m,r}. \quad (1)$$

The PFNS values N_k and N_j are provided as input with the key “values” in the dictionary “data”. The variable $\delta_k^{n,i}$ corresponds to the uncertainties associated with the n th uncertainty source for the measurement of the isotope in question (hence superscript i) given relative to the experimental quantity (e.g., N_k for shape data). In ARIADNE, $\delta_k^{n,i}$ is provided as input for dictionary “unc_iso” with key “values”. In the current notation, “units” would be a vector of n strings with “%”. The correlations $\text{Cor}_{k,j}^{n,i}$ associated with $\delta_k^{n,i}$ of uncertainty source n are specified with the keys “type” and “type_arg” in dictionary “unc_iso”. The term $\sum_m \delta_k^{m,r} \delta_j^{m,r} \text{Cor}_{k,j}^{m,r}$ provides covariances for uncertainty sources of and given relative to the reference measurement (superscript r). This term is non-zero for shape ratio and shape ratio calibration measurements and $\delta_k^{m,r}$ and $\text{Cor}_{k,j}^{m,r}$ are provided as input for ARIADNE with the dictionary “unc_ref” analogously to “unc_iso”. The term $\delta_k^{ND} \delta_j^{ND} \text{Cor}_{k,j}^{ND}$ adds nuclear data covariances associated with the nuclear data used to convert shape ratio or shape ratio calibration data to PFNS shape data. This term is zero for shape data. The information to define which nuclear data and covariances should be used for $\delta_k^{ND} \delta_j^{ND} \text{Cor}_{k,j}^{ND}$ is provided in the dictionary “reference”.

The relative covariances $C_{k,j}^E$, $C_{k,j}^t$ and $C_{k,j}^l$ for energy, time resolution and TOF length uncertainties, respectively, for shape data are given by:

$$\begin{aligned} C_{k,j}^E &= \delta E_k^i \delta E_j^i \left(\frac{1}{2} - \frac{E_k}{T^i} \right) \left(\frac{1}{2} - \frac{E_j}{T^i} \right) \text{Cor}_{k,j}^{E,i}, \\ C_{k,j}^t &= 8 \frac{\sqrt{E_k E_j}}{m_n (l^i)^2} \left(\frac{1}{2} - \frac{E_k}{T^i} \right) \left(\frac{1}{2} - \frac{E_j}{T^i} \right) (\Delta t^i)^2, \\ C_{k,j}^l &= \frac{4}{(l^i)^2} \left(\frac{1}{2} - \frac{E_k}{T^i} \right) \left(\frac{1}{2} - \frac{E_j}{T^i} \right) (\Delta l^i)^2. \end{aligned} \quad (2)$$

The energy uncertainties δE_k^i and associated correlations $\text{Cor}_{k,j}^{E,i}$ for the isotope in question are given relative to the outgoing neutron energy E_k as input via the keys “eout_unc” (“eout_unc_unit”=“%” in this notation), “eout_unc_type” and “eout_unc_type_arg” in the dictionary “data”. The outgoing neutron energies E_k are

provided with the same dictionary with the keyword “eout” and “eout_unit”. The Maxwellian temperature T^i of the isotope in question is provided with the dictionary “maxw_t_iso”, while the time resolution Δt^i is provided with the keys “value” and “unit” in the dictionary “trsl”. The TOF length l^i and its uncertainty Δl^i are provided within the dictionary “tof_length” with keys “value”, “value_unit”, “unc” and “unc_unit”. The mass of the neutron m_n is saved as a physical constant in ARIADNE.

For shape ratio data, $C_{k,j}^E$, $C_{k,j}^t$ and $C_{k,j}^l$ read:

$$\begin{aligned} C_{k,j}^E &= (1/T^r - 1/T^i)^2 \delta E_k^i \delta E_j^i \text{Cor}_{k,j}^{E,i}, \\ C_{k,j}^t &= \frac{8(1/T^r - 1/T^i)^2 (\Delta t^i)^2 (E_k E_j)^{3/2}}{m_n (l^i)^2}, \\ C_{k,j}^l &= \frac{4(1/T^r - 1/T^i)^2 E_k E_j (\Delta l^i)^2}{(l^i)^2}, \end{aligned} \quad (3)$$

with the Maxwellian temperature T^r of the reference PFNS provided with the dictionary “maxw_t_ref”.

For shape ratio calibration data, energy uncertainties, time resolutions and TOF length uncertainties need to be provided for the measurement of the isotope in question and the reference measurement. Hence, $C_{k,j}^E$, $C_{k,j}^t$ and $C_{k,j}^l$ are computed by:

$$\begin{aligned} C_{k,j}^E &= \delta E_k^i \delta E_j^i \left(\frac{1}{2} - \frac{E_k}{T^i} \right) \left(\frac{1}{2} - \frac{E_j}{T^i} \right) \text{Cor}_{k,j}^{E,i} \\ &\quad + \delta E_k^r \delta E_j^r \left(\frac{1}{2} - \frac{E_k}{T^r} \right) \left(\frac{1}{2} - \frac{E_j}{T^r} \right) \text{Cor}_{k,j}^{E,r}, \\ C_{k,j}^t &= 8 \frac{\sqrt{E_k E_j}}{m_n} \left[\left(\frac{1}{2} - \frac{E_k}{T^i} \right) \left(\frac{1}{2} - \frac{E_j}{T^i} \right) \frac{(\Delta t^i)^2}{(l^i)^2} \right. \\ &\quad \left. + \left(\frac{1}{2} - \frac{E_k}{T^r} \right) \left(\frac{1}{2} - \frac{E_j}{T^r} \right) \frac{(\Delta t^r)^2}{(l^r)^2} \right], \\ C_{k,j}^l &= \frac{4}{(l^i)^2} \left(\frac{1}{2} - \frac{E_k}{T^i} \right) \left(\frac{1}{2} - \frac{E_j}{T^i} \right) (\Delta l^i)^2 \\ &\quad + \frac{4}{(l^r)^2} \left(\frac{1}{2} - \frac{E_k}{T^r} \right) \left(\frac{1}{2} - \frac{E_j}{T^r} \right) (\Delta l^r)^2. \end{aligned} \quad (4)$$

The energy uncertainties δE_k^r and associated correlations $\text{Cor}_{k,j}^{E,r}$ for the reference isotope are provided relative to the outgoing neutron energy E_k as input via the keys “eout_unc” (“eout_unc_unit”=“%” in this notation), “eout_unc_type” and “eout_unc_type_arg” in the dictionary “unc_ref”. The time resolution of the reference Δt^r is provided with the keys “ref_value” and “ref_unit” in the dictionary “trsl”, while the TOF length l^r and its uncertainty Δl^r of the reference are provided within the dictionary “tof_length” with keys “ref_value”, “ref_value_unit”, “ref_unc” and “ref_unc_unit”.

References

1. Experimental Nuclear Reaction Data Library (EXFOR), IAEA Nuclear Data Section. See <https://www-nds.iaea.org/exfor> (accessed on 10/16/2017)
2. N. Otuka et al., Nucl. Data Sheets **120**, 272 (2014)
3. D. Neudecker et al., Nucl. Data Sheets **131**, 289 (2016)
4. R. Capote et al., Nucl. Data Sheets **131**, 1 (2016)
5. M.B. Chadwick et al., Nucl. Data Sheets **148**, 189 (2018)
6. D. Brown et al., Nucl. Data Sheets **148**, 1 (2018)
7. D. Neudecker et al., Nucl. Data Sheets **148**, 293 (2018)
8. C.M. Mattoon et al., Nucl. Data Sheets **113**, 3145 (2012)
9. D. Neudecker, Los Alamos National Laboratory Report LA-UR-17-28970, 2017
10. Investigation of Covariance Data in General Purpose Nuclear Data Libraries – WPEC subgroup 44 (SG44), <https://www.oecd-nea.org/science/wpec/sg44/> (accessed 10/26/2017)
11. W. Mannhart, International Atomic Energy Agency Report IAEA-TECDOC-410, 1987 pp. 158–171
12. W. Mannhart, International Atomic Energy Agency Report INDC(NDS)-220, 1989 pp. 305–336
13. D.L. Smith et al., International Atomic Energy Agency Report INDC(NDS)-0678, 2015
14. J.P. Lestone et al., Nucl. Data Sheets **119**, 213 (2014)
15. J.P. Lestone et al., Los Alamos National Laboratory Report LA-UR-14-24087, 2014
16. V. Zerkin, EPJ Web Conf. **27**, 00009 (2012)
17. D. Neudecker et al., EPJ Nuclear Sci. Technol. **4**, 21 (2018)

Cite this article as: Denise Neudecker, ARIADNE – a program estimating covariances in detail for neutron experiments, EPJ Nuclear Sci. Technol. **4**, 34 (2018)