NuDEX

a Nuclear DE-eXcitation code

User's Manual

Release 1.0

E. Mendoza, D. Cano-Ott

Contents

1 Introduction

NuDEX is a computer toolkit which allows to generate de-excitation cascades of atomic nuclei. It is written in C++ programming language and can be used to create different applications, adapted to specific needs. At the moment, NuDEX is distributed together with two specific applications: one to generate cascades emitted after neutron capture, and the other to generate cascades starting from a certain set of excited levels.

To model the de-excitation of a nucleus, NuDEX generates the full level scheme, branching ratios and internal conversion coefficients. To do this, it takes the known values from a database based mainly in RIPL-3 [1] and ENSDF [2], and fills in the missing information with statistical models. The database allows to generate automatically the de-excitation cascades of ~3000 nuclei. The reliability of the cascades produced by NuDEX depends on many factors, such as the information available in the databases, the specific de-excitation, and the observable that is to be simulated. For example, simulating the response of a calorimetric detector is not the same as the response of HPGe to specific γ-ray lines. NuDEX is not expected to work very well for light nuclei, with few excitation levels; with the exception of cascades from neutron capture in the thermal of well-known nuclei, where the experimental information is complete enough to generate realistic de-excitation cascades.

2 Performance of the code

2.1 Overview

NuDEX operates in a similar way as DICEBOX [3] and DEGEN [4]. To perform the γ-ray decay of a nucleus from a certain excited level, NuDEX generates the full level scheme and associated branching ratios, together with the internal conversion coefficients.

The known part of the level scheme, including branching ratios and internal conversion coefficients, is obtained from the RIPL-3 database, which takes the data from ENSDF but provides (unlike ENSDF) a unique spin and parity assignment for each level. In addition, the database provides a cut-off energy E_{crit} , up to which a level scheme is supposed to be complete. NuDEX uses this information up to E_{crit}.

Above E_{crit}, the level scheme is generated randomly according to level density formulas. The distances between consecutive levels are sampled according to a Wigner distribution. Some of the level density models implemented in the code are the Back-Shifted Fermi Gas Model and the Gilbert-Cameron Model, both as they are defined in the RIPL-3 manual and using the parameters provided by the RIPL-3 database (files *level-densities-bfmeff.dat* and *level-densities-ctmeff.dat*). Known levels above E_{crit} may be inserted in the scheme, whenever possible, replacing some of the levels generated randomly with the same spin and parity, and with a similar excitation energy.

In order to reduce the computation memory and time, NuDEX has the possibility of grouping the levels into *bands*. The levels that make up each of these bands all have the same spin and parity.

Each of these bands is then characterized by their lower and upper excitation energy, their spin and parity, and the number of levels which contains. The effect of making bands does not have an important impact in the resulting electromagnetic cascades for most practical cases.

Branching ratios not present in the databases are generated according to the so called *extreme statistical model* [3]. The branching ratio of the transition from a level *a* to a level *b*, $BR_{a\to b}$, is computed according to:

$$BR_{a\to b} \propto \sum_{X,L} \xi^2_{a\to b} E_\gamma^{2L+1} \overleftarrow{f^{XL}}(E_\gamma, E_a) \qquad (1)$$

where X is the type of transition (electric or magnetic); *L* is the multipolarity; $\xi_{a\to b}$ is a random variable drawn independently from a normal distribution with zero mean and unit variance, which introduce the Porter-Thomas fluctuations; E_γ is the difference between the energy of the level a, E_a, and the energy of the level b, E_b; and $\overleftarrow{f^{XL}}(E_\gamma, E_a)$ is the Photon Strength Function (PSF). The branching ratios are normalized so that the sum of all of them starting from the same level equals 1, $\sum_{i=0}^{n-1} BR_{n\to i} = 1$. For the moment, only E1, M1 and E2 transitions are considered.

In addition to the branching ratios from the known part of the level scheme, the NuDEX database contains also lists of primary γ-rays from thermal neutron capture, together with the observed intensities, taken from ENSDF. These data are particularly important for simulating thermal neutron capture γ-rays of light nuclei.

Internal conversion coefficients are obtained from the same RIPL-3 database as the known levels, if present. Otherwise they are obtained from the data tables provided by Band et al. in [5].

For the generation of the statistical nuclei (levels and branching ratios) and to perform the decays, NuDEX needs to use a random number generator. NuDEX takes the random number generators from ROOT [6], and for this reason NuDEX programs must be compiled along with ROOT. No other features of ROOT different than the random number generators are used.

2.2 Parameters

As mentioned before, NuDEX generates the full level scheme, branching ratios and electron conversion coefficients of a nucleus to compute the de-excitation cascades. There is usually no single way to do it, since different models can be used, with different parameters, etc. A list of options is provided in Table 1. All these options have a default value, which can be changed using an input file which can be usually passed to the NuDEX application via the command line. Another possibility is to place the input file into the *SpecialInputs* directory inside the NuDEX data library. Additionally, some of the default options can be modified in the *GeneralStatNuclParameters.dat* file, also inside the NuDEX data library.

The input is a text file containing new values for the different variables. To modify a variable one has to write in the input file the keyword that appears in the first column of Table 1, followed by

its value. Lines starting with the "#" symbol are ignored. Anything written after the "END" keyword is also ignored.

In addition to the options listed in Table 1, it is also possible to change the parameters that define the level densities and the photon strength functions. The way to change them is through the input file. For this, the input file should contain the keywords `LDPARAMETERS`, to change the level densities, and/or `PSF`, to change the PSF; followed by the required information according to the formats presented in Table 2 and Table 3.

Concerning the level densities, there are three possibilities:

- Back-Shifted-Fermi-Gas model, as defined in [1] (LDtype=1), which depends on four parameters: $\tilde{a}$, γ, δW and Δ.
- Gilbert-Cameron model, as defined in [1] (LDtype=2), which depends on seven parameters: $\tilde{a}$, γ, δW, Δ, E_{match}, E_0 and T.
- Back-shifted: Egidy, as defined in [7] (LDtype=3), which depends on two parameters: a and Δ.

Concerning the PSFs they can be defined using any linear combination of the models listed in Table 4.

Variable name (keyword)	Variable type	Description
LDtype (LEVELDENSITYTYPE)	integer	Level density model to be used: Back-Shifted-Fermi-Gas model, as defined in [1] (LDtype=1); Gilbert-Cameron model, as defined in [1] (LDtype=2); Back-shifted: Egidy, as defined in [7] (LDtype=3). Default: LDtype=1.
MaxSpin (MAXSPIN)	real	Maximum spin to be considered in the statistical nucleus. Default: MaxSpin=10.
MinLevelsPerBand (MINLEVELSPERBAND)	integer	If levels are grouped into bands (BandWidth> 0), then this is the minimum number of levels which will be merged into a band. If the number of levels candidate to form a band in an energy interval is less than MinLevelsPerBand, then the levels are kept separate, without being grouped into a band. Default: MinLevelsPerBand =10.
BandWidth (BANDWIDTH_MEV)	real	If levels are grouped into bands, BandWidth is the width, in MeV, of these bands. If BandWidth≤ 0 then no bands are created. Default: BandWidth=0.05.
MaxExcEnergy (MAXEXCENERGY_MEV)	real	The level scheme is generated up to MaxExcEnergy, in MeV. If MaxExcEnergy<=0, then the level scheme is generated up to the neutron separation energy plus the absolute value of MaxExcEnergy. Default: MaxExcEnergy

		=-1.
E_{crit} (ECRIT_MEV)	real	Cut-off energy, up to which the known part of the level scheme is supposed to be complete. Default: obtained from the RIPL-3 database.
KnownLevelsFlag (KNOWNLEVELSFLAG)	integer	If it is set to 0, NuDEX will use the RIPL-3 known levels only up to E_{crit}. If it is set to 1, NuDEX will try to use also the known levels above E_{crit}, by replacing some of the randomly generated levels with the same spin and parity. Default: KnownLevelsFlag=1.
PSFFlag (PSF_FLAG)	integer	PSF to be used: IAEA PSF models and database [8] (PSF-Flag=0), RIPL-3 PSF models and database (PSFFlag=1). Default: PSFFlag=0.
BrOption (BROPTION)	integer	This is a flag which indicate how the branching ratios are stored in memory. Branching ratios are calculated always when needed. If BrOption=0, then they are not stored in memory. In case they are needed again, they are re-calculated. If BrOption=1, then they are stored in memory after being computed. If BrOption=2, then they are stored only for the first transition of each cascade. With BrOption=0 the code will use less memory, with BrOption=1 the code will be faster, and BrOption=2 is something in betweeen. Default: BrOption=2.
SampleGammaWidths (SAMPLEGAMMAWIDTHS)	integer	This is a flag indicating if Porter-Thomas fluctuations are disabled (SampleGammaWidths=0) or enabled (SampleGammaWidths=1) for the calculation of the branching ratios, as given in Eq. (1). Default: SampleGammaWidths=1.
ElectronConversionFlag (ELECTRONCONVERSIONFLAG)	integer	If it is set to 0, then no electron conversion will be performed, i.e. all the ICC values are set to 0. If it is set to 1, then only ICC values from RIPL-3 are considered. If it is set to 2, NuDEX will use ICC factors from RIPL-3 + data tables from [5]. Default: ElectronConversionFlag=2.
PrimGamNormFactor (PRIMARYTHCAPGAMNORM)	real	Can be used to re-normalize the intensities of the primary γ-rays from thermal neutron capture. Default: PrimGamNormFactor=1.
PrimaryGammasEcut (PRIMARYGAMMASECUT)	real	Primary γ-rays from thermal neutron capture provides information about the largest transition probabilities from the capture level to a set of levels ranging from the fundamental to a certain excitation energy. PrimaryGammasEcut defines what to do with the transition probabilities in the same energy range but

		which are not present in the database. If PrimaryGammasEcut<0, then NuDEX will generate transition probabilities to these levels using the statistical models. If PrimaryGammasEcut>0, then all the branching ratios from the thermal level to levels with energy smaller than PrimaryGammasEcut (in MeV) are set to 0, except those branching ratios in the database. Finally, if PrimaryGammasEcut=0, then PrimaryGammasEcut is set to the energy of the level corresponding to the lowest energy primary gamma ray in the database; in other words, no transitions are allowed from the thermal level to levels close to those in the database. Default: PrimaryGammasEcut=0.
Seed1 (SEED1)	unsigned integer	Seed for the random number generator used to create the statistical part of the level scheme.
Seed2 (SEED2)	unsigned integer	Seed for the random number generator used to calculate the branching ratios, i.e. to simulate the Porter-Thomas fluctuations.
Seed3 (SEED3)	unsigned integer	Seed for the random number generator used to produce the cascades.

Table 1: Variables which define how NuDEX creates statistical nucleus.

```
LDPARAMETERS [LDType=1]
[dW] [gamma] [ainf] [Delta (or pairing)]

or:
LDPARAMETERS [LDType=2]
[dW] [gamma] [ainf] [Delta (or pairing)] [Ex] [E0] [T]

or:
LDPARAMETERS [LDType=3]
[ainf] [Delta (or pairing)]
```

Where:

LDType:	Level density type.
dW:	Shell correction energy used in the Ignatyuk formula in MeV
gamma:	Damping parameter of the Ignatyuk formula

`ainf`:	Asymptotic level density parameter in 1/MeV
`pairing`:	Effective energy shift in MeV
`Ex`:	Energy at which the low and high energy formulae match in MeV
`E0`:	Energy shift for the low-energy approach in MeV
`T`:	Temperature for the low-energy approach in MeV

Example:

```
LDPARAMETERS 1
 -3.42272  0.07039  20.56586  -0.92723
```

Table 2: Format to change the level densities using the input file.

```
PSF
[nR_E1]
[PSFType_E1] [E_E1] [G_E1] [s_E1] (more parameters in some cases)
...
[nR_M1]
[PSFType_M1] [E_M1] [G_M1] [s_M1] (more parameters in some cases)
...
[nR_E2]
[PSFType_E2] [E_E2] [G_E2] [s_E2] (more parameters in some cases)
...
```

Where:

`[nR_E1]`, `[nR_M1]` and `[nR_E2]` are the number of PSF resonances (or expressions) defining, respectively, the E1, M1 and E2 PSFs. After these variables, there should be one line per resonance, with the type of resonance first (`[PSFType_XX]` as defined in the first column of Table 4), followed by the values of the corresponding parameters (last column of Table 4).

Example:

```
PSF
2
  11  13.2  4.283  0.8
  11  14.753  4.734  0.4
3
  0  7.46  4  2.46
  0  2.9  1.5  0.296
  21  0.188  0.8  0
1
  0  10.8  1.92  0.83
```

Table 3: Format to change the PSFs using the input file.

PSF model	Abbreviation	Description	Parameters
PSFType=0	SLO	Standard Lorentzian model [1], given by Eq.(2).	E_r, Γ_r, σ_r
PSFType=1	EGLO	Enhanced Generalized Lorentzian Model [9], given by Eqs. (5) and (6) with k_1=k_2=k, and k is obtained from Eq. (143) of [1].	E_r, Γ_r, σ_r
PSFType=2	SMLO1	Simplified Modified Lorentzian model, as defined in [1], given by Eqs. (9) and (10).	E_r, Γ_r, σ_r
PSFType=3	GLO	Generalized Lorentzian Model [9], given by Eqs. (5) and (6) with k_1=k_2=1.	E_r, Γ_r, σ_r
PSFType=4	MGLO	Modified Generalized Lorentzian Model [10], given by Eqs. (5) and (6) with k_1 obtained from Eq. (143) of [1] and k_2=1.	E_r, Γ_r, σ_r
PSFType=5	KMF	Kadmenskij-Markushev-Furman model [11], given by Eqs. (3) and (4).	E_r, Γ_r, σ_r
PSFType=6	GH	Hybrid model [12], given by Eqs. (7) and (8).	E_r, Γ_r, σ_r
PSFType=7	MEGLO	Modified version of the Enhanced Generalized Lorentzian model, given by Eqs. (5) and (6) with k_1=k_2=k.	E_r, Γ_r, σ_r, *k*
PSFType=8	MEGLO	Modified version of the Enhanced Generalized Lorentzian model, given by Eqs. (5) and (6).	E_r, Γ_r, σ_r, k_1, k_2
PSFType=9	MEGLO	Modified version of the Enhanced Generalized Lorentzian model, given by Eqs. (5) and (6) with k_1=k_2=k and a constant temperature of the nucleus is provided.	E_r, Γ_r, σ_r, *k, T*
PSFType=10	MEGLO	Modified version of the Enhanced Generalized Lorentzian model, given by Eqs. (5) and (6) plus a constant temperature of the nucleus is provided.	E_r, Γ_r, σ_r, k_1, k_2, *T*
PSFType=11	SMLO2	Simplified Modified Lorentzian model, as defined in [8][13][14], given by Eqs. (11), (12) and (13).	E_r, Γ_r, s_r
PSFType=20	Gauss	Gaussian function, given by Eq. (14).	E_r, σ, *Area*
PSFType=21	Expo	Exponential function, given by Eq. (15).	*A*, η(*)

Table 4: Photon Strength Function models available in NuDEX. (*)Although there are only two parameters, a third "dummy" parameter, which is not used, has to be provided in the input file.

The expressions for the different PSF models are:

$$\overleftarrow{f}_{SLO}(\epsilon_\gamma) = C \times \sigma_r \Gamma_r \frac{\epsilon_\gamma \Gamma_r}{\left(\epsilon_\gamma^2 - E_r^2\right)^2 + \epsilon_\gamma^2 \Gamma_\gamma^2} \tag{2}$$

$$\overleftarrow{f}_{KMF}(\epsilon_\gamma, T) = C \times \sigma_r \Gamma_r \times 0.7 \frac{E_r \Gamma_c(\epsilon_\gamma, T)}{\left(\epsilon_\gamma^2 - E_r^2\right)^2} \tag{3}$$

$$\Gamma_c(\epsilon_\gamma, T) = \frac{\Gamma_r}{E_r^2}\left(\epsilon_\gamma^2 + 4\pi T^2\right) \tag{4}$$

$$\overleftarrow{f}_{MEGLO}(\epsilon_\gamma, T) = C \times \sigma_r \Gamma_r \times \left[\frac{\epsilon_\gamma \Gamma_{k_1}(\epsilon_\gamma, T)}{\left(\epsilon_\gamma^2 - E_r^2\right)^2 + \epsilon_\gamma^2 \Gamma_{k_1}^2(\epsilon_\gamma, T)} + \frac{0.7\Gamma_{k_2}(\epsilon_\gamma = 0, T)}{E_r^3}\right] \tag{5}$$

$$\Gamma_k(\epsilon_\gamma, T) = \frac{\Gamma_r}{E_r^2}\left(\epsilon_\gamma^2 + 4\pi T^2\right)\left[k + (1-k)\frac{\epsilon_\gamma - \varepsilon_0}{E_r - \varepsilon_0}\right] \tag{6}$$

$$\overleftarrow{f}_{GH}(\epsilon_\gamma, T) = C \times \sigma_r \Gamma_r \frac{\epsilon_\gamma \Gamma_h(\epsilon_\gamma, T)}{\left(\epsilon_\gamma^2 - E_r^2\right)^2 + \epsilon_\gamma^2 \Gamma_h(\epsilon_\gamma, T)\Gamma_r} \tag{7}$$

$$\Gamma_h(\epsilon_\gamma, T) = 0.63\Gamma_r \frac{\epsilon_\gamma^2 + 4\pi T^2}{\epsilon_\gamma E_r} \tag{8}$$

$$\overleftarrow{f}_{SMLO1}(\epsilon_\gamma, T) = C \times \frac{\sigma_r \Gamma_r}{1 - exp(-\epsilon_\gamma/T)} \times \frac{\epsilon_\gamma \Gamma_{SM1}(U_i)}{\left(\epsilon_\gamma^2 - E_r^2\right)^2 + \epsilon_\gamma^2 \Gamma_{SM1}^2(U_i)} \tag{9}$$

$$\Gamma_{SM1}(U_i) = \frac{\Gamma_r}{E_r} U_i \tag{10}$$

$$\overleftarrow{f}_{SMLO2}(\epsilon_\gamma, T) = C \times \frac{\sigma_{TRK}}{1 - exp(-\epsilon_\gamma/T)} \times \frac{2}{\pi} s_r \frac{\epsilon_\gamma \Gamma_{SM2}(\epsilon_\gamma, T)}{\left(\epsilon_\gamma^2 - E_r^2\right)^2 + \epsilon_\gamma^2 \Gamma_{SM2}^2(\epsilon_\gamma, T)} \tag{11}$$

$$\Gamma_{SM2}(\epsilon_\gamma, T) = \frac{\Gamma_r}{E_r}\left(\epsilon_\gamma + \frac{4\pi^2}{E_r} T^2\right) \tag{12}$$

$$\sigma_{TRK} = 60 \frac{NZ}{A} (mb \cdot MeV) \tag{13}$$

$$\overleftarrow{f}_{Gauss}(\epsilon_\gamma) = Area \times \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{\left(\epsilon_\gamma - E_r\right)^2}{2\sigma^2}\right) \tag{14}$$

$$\overleftarrow{f}_{Expo}(\epsilon_\gamma) = A \times exp(-\eta \epsilon_\gamma) \tag{15}$$

where in all these expressions, ϵ_γ is the difference of the energies of the initial and final excited levels. A, Z and N are the number of nucleons, protons and neutrons of the nucleus. E_r, Γ_r and σ_r are, respectively, the energy, width and size of the resonances. E_r and Γ_r are usually given in MeV and σ_r in mb. s_r is a weight factor. C = $8.674 \cdot 10^{-8}$ MeV^{-3} for E1 and M1 transitions and C = $5.22 \cdot 10^{-8}$ MeV^{-3} for E2 transitions. U_i (U_f) is the excitation energy of the initial (final) level. T is the

temperature of the nucleus, in the final excited state unless explicitly stated otherwise. It is obtained, in general, from the expression $T = \sqrt{(U_f - \Delta)/a}$, where Δ is the pairing energy and a the energy dependent level density parameter. In the particular case of the SMLO2 model, it is obtained from $T = \sqrt{10U_f/A}$. If the Gilbert-Cameron level density model is used, the constant temperature of the model is used at low excitation energies.

2.3 Outputs

The output of a NuDEX application will usually contain at least two files:

- One file with the extension *.out, containing information about the parameters used by NuDEX to compute the cascades, known levels, level densities, PSFs etc.
- One file with the extension *.cas, with the de-excitation cascades. The file will contain one cascade per line. The format of each line is:

```
[NPAR]  [PARTYPE_1]  [PARENE_1]  [PARTIME_1]    [PARTYPE_2]  [PARENE_2]
[PARTIME_2] ... [PARTYPE_NPAR] [PARENE_NPAR] [PARTIME_NPAR]

[NPAR]: number of particles emitted
[PARTYPE_XXX]: Particle type --> g:gamma, e:electron, p:positron
[PARENE_XXX]: Particle energy, in MeV
[PARTIME_XXX]: Particle time, in s
```

*Table 5: Format of the *.cas output file.*

3 Executables

NuDEX can be seen as a toolkit which can be used to create different applications, adapted to specific needs. For the moment, it is distributed with two specific applications. These can serve as a basis for creating new applications if the user requires it. These applications are:

- NuDEX_NCaptureCascadeGenerator01: generates de-excitation cascades emitted after neutron capture.
- NuDEX_DecayCascadeGenerator01: generates de-excitation cascades starting from a certain set of excited levels.

It is worth mentioning that to generate reliable cascades it is not enough to model the levels and branching ratios, but it is also necessary to determine the initial level from which de-excitation begins. And this is something that is not easy on many occasions.

3.1 Compilation

Both applications are inside the "exe" folder. They have to be compiled along with ROOT, i.e. doing something similar to:

```
g++ -std=c++11 ../src/*.cc NuDEX_NCaptureCascadeGenerator01.cc `root-config --libs --cflags` -I../include/ -o NuDEX_NCaptureCascadeGenerator01
```

```
g++ -std=c++11 ../src/*.cc NuDEX_DecayCascadeGenerator01.cc `root-config --libs --cflags` -I../include/ -o NuDEX_DecayCascadeGenerator01
```

Alternatively:

```
./Compile NuDEX_NCaptureCascadeGenerator01
```

```
./Compile NuDEX_DecayCascadeGenerator01
```

3.2 NuDEX_NCaptureCascadeGenerator01

`NuDEX_NCaptureCascadeGenerator01` can be used to generate nuclear de-excitation cascades emitted after neutron capture. It can be executed in the following two ways:

1) `NuDEX_NCaptureCascadeGenerator01 outputfilebase LIBDIR ZA [keyname1] [val1] [keyname2] [val2] ...`
2) `NuDEX_NCaptureCascadeGenerator01 outputfilebase inputfile [keyname1] [val1] [keyname2] [val2] ...`

where `outputfilebase` is the path+base name for the output files (i.e. without the *.out, *.cas … extension); `LIBDIR` is the path to the NuDEX database; `ZA` is used to define the target nucleus (not the compound nucleus) as ZA=1000·Z+A, with Z the number of protons and A the number of nucleons; `inputfile` is the name of the input file; and `[keyname1] [val1] [keyname2] [val2]` … are optional entries which can be used to modify parameters set by default or by the input file.

The easiest way to use `NuDEX_NCaptureCascadeGenerator01` is to execute it using the first way first, without input file. The program will generate three outputs: *.out, with general information; *.cas, with the cascades; and *.inp, with an input file containing all the parameters with their default values. This input file can be then used to execute `NuDEX_NCaptureCascadeGenerator01` again, using the second way. If no changes to the input file are performed, the same output files should be obtained. The default options can be then changed in the input file to produce different cascades.

In the input file we will find:

1- Values for the variables listed in Table 1. The variables are defined in the form `[keyword] [value]`, where `[keyword]` is the keyword that defines the variable, shown in the first column of Table 1, and `[value]` is its value.
2- Values defining the level densities and PSFs, according to the format shown in Table 2 and Table 3. These values have been generated according to the values of the variables *LDtype* and *PSFFlag*. Indeed, if the input file does (doesn't) contain the information in in Table 2 and Table 3, then the values of *LDtype* and *PSFFlag* are not (are) used.
3- Additional variables, which are defined in the form `[keyword] [value]`, which are:
    - `LIBDIR`: path to the NuDEX database.
    - `ZA`: ZA=1000·Z+A, with Z the number of protons and A the number of nucleons.
    - `NCASCADES`: number of de-excitation cascades to be simulated.
    - `NEUTRONENERGY_MEV`: Energy of the neutron, in MeV, used to compute the energy of the initial level of the cascades (where the cascade starts).
    - `JSPIN`: spin of the initial level.
    - `PARITY`: parity of the initial level
    - `DOTHERMALCASCADE`: if this variable is positive, then the capture reaction is assumed to be *thermal*, i.e. due to low energy neutrons. In practice, this means that NuDEX will use primary γ-rays and branching ratios from thermal neutron capture, obtained from ENSDF. Variables *PrimGamNormFactor* and *PrimaryGammasEcut* (Table 1) will have an effect for thermal cascades only.
    - `TIMEWINDOW_NS`: time limit, in ns, for particles in the cascade to be written in the *.cas file (Table 5). If a particle is emitted after this time limit, then it is not written in the *.cas file.
    - `SEED4`: seed to compute the branching ratios of the primary transitions (the first transition of the cascade). According to the model being used, changing this seed is similar to change the level from which the de-excitation starts, i.e. the resonance at which the capture occurs (maintaining the same spin and parity).

Example:

Imagine that we would like to get thermal cascades from ^{197}Au(n,γ). To do this we will type, for example:

```
NuDEX_NCaptureCascadeGenerator01 output01 […]/NuDEXlib-1.0 79197
```

where `[…]/NuDEXlib-1.0` is the data path to the NuDEX data library. Then we will get three different output files: `output01.out`, with general information; `output01.cas`, with the cascades; and `output01.inp`, an input file with all the default parameters.

By default `NuDEX_NCaptureCascadeGenerator01` generates 100 cascades. Imagine that we would like to generate more cascades (10^6, for example), without changing the rest of the input options. This can be done in two ways. The first one is by changing the input file

output01.inp. In particular, the number 100 placed after NCASCADES should be replaced by 1000000. Then we will type:

```
NuDEX_NCaptureCascadeGenerator01 output02 output01.inp
```

Another possibility is by typing:

```
NuDEX_NCaptureCascadeGenerator01 output02 […]/NuDEXlib-1.0 79197 NCASCADES 1000000
```

In both cases we will obtain a cascades output file, output02.cas, with 10^6 cascades, whose first 100 cascades will be the same as the ones in output01.cas.

3.3 NuDEX_DecayCascadeGenerator01

NuDEX_DecayCascadeGenerator01 can be used to generate nuclear de-excitation cascades starting from one or several excited levels. It can be executed in the following two ways:

1) `NuDEX_DecayCascadeGenerator01 1 outputfilebase LIBDIR ZA EXCENE_MEV [keyname1] [val1] [keyname2] [val2] ...`
2) `NuDEX_DecayCascadeGenerator01 2 outputfilebase inputfile [keyname1] [val1] [keyname2] [val2] ...`

where outputfilebase is the path+base name for the output files (i.e. without the *.out, *.cas … extension); LIBDIR is the path to the NuDEX database; ZA is used to define the nucleus as ZA=1000·Z+A, with Z the number of protons and A the number of nucleons; EXCENE_MEV is the excitation energy of the nucleus, in MeV; inputfile is the name of the input file; and [keyname1] [val1] [keyname2] [val2] … are optional entries which can be used to modify parameters set by default or by the input file.

As in NuDEX_NCaptureCascadeGenerator01, the easiest way to use NuDEX_DecayCascadeGenerator01 is to execute it first using the first way. Then the program will create the same three output files as before: *.out, with general information; *.cas, with the cascades; and *.inp, with an input file containing all the parameters with their default values. This input file can be then modified according to the required needs, and used to execute NuDEX_DecayCascadeGenerator01 again, using the second way of execution.

In the input file we will find:

1- Values for the variables listed in Table 1. The variables are defined in the form [keyword] [value], where [keyword] is the keyword that defines the variable, shown in the first column of Table 1, and [value] is its value.
2- Values defining the level densities and PSFs, according to the format shown in Table 2 and Table 3. These values have been generated according to the values of the variables *LDtype* and *PSFFlag*. Indeed, if the input file does (doesn't) contain the information in in Table 2 and Table 3, then the values of *LDtype* and *PSFFlag* are not (are) used.
3- Additional variables, which are defined in the form [keyword] [value], which are:

- LIBDIR: path to the NuDEX database.
- ZA: ZA=1000·Z+A, with Z the number of protons and A the number of nucleons.
- NCASCADES: number of de-excitation cascades to be simulated.
- TIMEWINDOW_NS: time limit, in ns, for particles in the cascade to be written in the *.cas file (Table 5). If a particle is emitted after this time limit, then it is not written in the *.cas file.
- STARTINGLEVEL: this keyword must be followed by four entries instead of one. It is used to define the excited levels where the cascades start, and it can appear several times in the input file, each of them corresponding to a different starting level. The four entries following the keyword STARTINGLEVEL are the excitation energy (in MeV), the spin, the parity, and the *feeding probability*. This last variable defines the probability that the cascade starts in one or other excited level (they don't need to be normalized to 1). These excited levels will correspond to:
    i. If the spin value provided is positive, then NuDEX will take level with the closest excitation energy as the one provided in the input, among those with the same spin and parity as the ones provided in the input.
    ii. If the spin value provided is positive, then NuDEX will take the excited level with the closest excitation energy as the one provided in the input, regardless of their spin and parity.

Example:

Imagine that we would like to compute cascades from an ^{88}Y radioactive source (^{88}Y decays via β^+ into ^{88}Sr). Then we can start with:

```
NuDEX_DecayCascadeGenerator01  1  output01  […]/NuDEXlib-1.0  38088
2.734
```

where […]/NuDEXlib-1.0 is the data path to the NuDEX data library, and 38088 represents the daughter nucleus ^{88}Sr. Then we will get three different output files: output01.out, with general information; output01.cas, with the cascades; and output01.inp, an input file with all the default parameters. In this case all the decays will start from the same level of ^{88}Sr at 2.734 MeV. In reality, according to ENSDF there are about 94.4% probability of starting from that level. The rest of the decays will start from the excited levels at 1.836 (5.5%), 3.219 (0.028%) and 3.585 (0.066%).

To obtain more reliable cascades we need to change the line in outpu01.inp:

```
STARTINGLEVEL 2.734  -1  1  1
```

with:

```
STARTINGLEVEL 1.836  -1  1  0.055
STARTINGLEVEL 2.734  -1  1  0.944
STARTINGLEVEL 3.219  -1  1  0.00028
```

```
STARTINGLEVEL 3.585  -1  1  0.00066
```

or:

```
STARTINGLEVEL 1.836  2  +1  0.055
STARTINGLEVEL 2.734  3  -1  0.944
STARTINGLEVEL 3.219  2  +1  0.00028
STARTINGLEVEL 3.585  5  +1  0.00066
```

Then we should type:

```
NuDEX_DecayCascadeGenerator01 2 output02 input01.inp
```

If we would like to get more cascades we can type instead (10^6 cascades):

```
NuDEX_DecayCascadeGenerator01  2  output02  input01.inp  NCASCADES
1000000
```

or, alternatively, we can change in the `input01.inp` file the number following `NCASCADES`.

4 References

[1] R. Capote et al., *RIPL – reference input parameter library for calculation of nuclear reactions and nuclear data evaluations*, Nuclear Data Sheets 110 (12) (2009) 3107 – 3214. https://doi.org/10.1016/j.nds.2009.10.004

[2] https://www.nndc.bnl.gov/ensdf/

[3] F. Bečvář, *Simulation of γ cascades in complex nuclei with emphasis on assessment of uncertainties of cascade-related quantities*, Nucl. Instrum. Methods A 417 (2) (1998) 434 – 449. https://doi.org/10.1016/S0168-9002(98)00787-6

[4] D. Jordan et al., *An event generator for simulations of complex β-decay experiments*, Nucl. Instrum. Methods A 828 (2016) 52 – 57. https://doi.org/10.1016/j.nima.2016.05.034

[5] I. Band, M. Trzhaskovskaya, C. Nestor, P. Tikkanen, S. Raman, *Dirac–fock internal conversion coefficients*, Atomic Data and Nuclear Data Tables 81 (1) (2002) 1 – 334. https://doi.org/10.1006/adnd.2002.0884

[6] R. Brun, F. Rademakers, *Root — an object oriented data analysis framework*, Nucl. Instrum. Methods A 389 (1) (1997) 81–86, new Computing Techniques in Physics Research V. https://doi.org/10.1016/S0168-9002(97)00048-X

[7] T. V. Egidy, H. Schmidt, A. Behkami, *Nuclear level densities and level spacing distributions: Part ii*, Nuclear Physics A 481 (2) (1988) 189 – 206. https://doi.org/10.1016/0375-9474(88)90491-5

[8] S. Goriely et al., *Reference database for photon strength functions*, Eur. Phys. J. A 55 (2019) 172. https://doi.org/10.1140/epja/i2019-12840-1

[9] J. Kopecky, M. Uhl, R. E. Chrien, *Radiative strength in the compound nucleus ^{157}Gd*, Phys. Rev. C 47 (1993) 312–322. https://doi.org/10.1103/PhysRevC.47.312

[10] J. Kroll et al., Strength of the scissors mode in odd-mass Gd isotopes from the radiative capture of resonance neutrons, Phys. Rev. C 88 (2013) 034317. https://doi.org/10.1103/PhysRevC.88.034317

[11] S. G. Kadmenskij, V. P. Markushev, V. I. Furman, Radiative widths of neutron resonances and giant dipole resonances, Yad. Fiz.; (USSR) 37:2.

[12] S. Goriely, *Radiative neutron captures by neutron-rich nuclei and the r-process nucleosynthesis*, Physics Letters B 436 (1) (1998) 10 – 18. https://doi.org/10.1016/S0370-2693(98)00907-1

[13] V. Plujko, O. Gorbachenko, R. Capote, P. Dimitriou, *Giant dipole resonance parameters of ground-state photoabsorption: Experimental values with uncertainties*, Atomic Data and Nuclear Data Tables 123-124 (2018) 1 – 85. https://doi.org/10.1016/j.adt.2018.03.002

[14] S. Goriely, V. Plujko, *Simple empirical E1 and M1 strength functions for practical applications*, Phys. Rev. C 99 (2019) 014303 https://doi.org/10.1103/PhysRevC.99.014303