

JOINT INSTITUTE FOR NUCLEAR RESEARCH  
Dzhelepov Laboratory of Nuclear Problems

# FINAL REPORT ON THE START PROGRAMME

*Development of a utility for reading ENDF-6  
format data and calculating neutron spectra  
using double-differential cross sections  
for the  $(\alpha, n)$  reactions from JENDL-5 files*

**Supervisor:**

Ph.D. Gromov Maxim Borisovich

**Student:**

Goncharenko Ivan, Russia  
Lomonosov Moscow State University  
Faculty of Physics

**Participation period:**

July 10 - August 19,  
Summer Session 2023

Dubna, 2023

# Abstract

As a consequence of the high interest in  $(\alpha, n)$  reactions, there are more and more tools for analytical calculation and evaluation of nuclear data. They allow making predictions for modern ultra-low background neutrino and dark matter detectors relative to the background level caused by these reactions. One of these tools is called NeuCBOT. This work is intended to contribute to the expansion of the functionality of the NeuCBOT program by calculating neutron spectrum data using double-differential cross sections from the JENDL-5 database in the ENDF-6 format.

# Contents

<b>Abstract</b>	<b>1</b>
<b>Introduction</b>	<b>3</b>
<b>1 Alpha-particle library structure and data storage format</b>	<b>3</b>
1.1 Structure of the ENDF-6 data storage format . . . . .	4
1.2 JENDL-5 Alpha-particle sublibrary structure . . . . .	5
1.3 Data storing format in tables for <b>MF</b> = 6 . . . . .	5
1.4 Data storing format in tables for <b>MT</b> = 50 – 91 . . . . .	6
<b>2 Transformation of the angle distribution into the neutron energy ditribution</b>	<b>7</b>
<b>3 Software implementation of the neutron spectrum calculation</b>	<b>8</b>
3.1 Auxiliary functionality of the program . . . . .	9
3.2 The main functionality of the program . . . . .	11
<b>4 Analysis of the obtained results</b>	<b>13</b>
<b>Conclusion</b>	<b>17</b>
<b>Bibliography</b>	<b>18</b>

# Introduction

The importance of taking into account  $(\alpha, n)$  reactions in low-background experiments increases with improving accuracy of measurements in modern neutrino detectors and in detectors for direct search of dark matter particles. Installations of this type are used for conducting scientific research on a wide range of physical tasks: searching for dark matter particles, investigation of the neutrino mass hierarchy issue, measurement of the geoneutrino flux, studying neutrino oscillation, the structure of the nucleus and the processes during supernova explosions, the formation of neutron stars and nucleosynthesis in the universe. As a consequence of the high interest in  $(\alpha, n)$  reactions, there are more and more tools for analytical calculation and evaluation of nuclear data that allow for modern ultra-low background neutrino and dark matter detectors to make predictions about the background level caused by these reactions, for example, NeuCBOT [1,2], SOURCES4 [3,4], NEDIS [5–8], USD [9], SaG4n [10].

Last year, the NeuCBOT program received an update, which, among other things, included the expansion of the functionality for calculating  $(\alpha, n)$  yields by adding new data from the Japanese Evaluated Nuclear Data Library database (hereinafter referred to as JENDL) [11]. However, this update did not allow obtaining neutron spectra, due to the lack of relevant data. This work is aimed at eliminating this shortcoming.

## 1 Alpha-particle library structure and data storage format

To improve the NeuCBOT program, it was decided to use the data recorded in [JENDL-5 Alpha-particle sublibrary](https://www.ndc.jaea.go.jp/ftpnd/jendl/jendl-5-a.html)<sup>1</sup>.

The data in the database is presented in the ENDF-6 format (Evaluated Nuclear Data File) [12]. Having not found ready-made tools for reading and processing data of this format on the Internet, it was decided to create a utility that will automatically read, convert and write JENDL-5 data in the desired format. It received the working name “ENDF6-reader”.

---

<sup>1</sup><https://www.ndc.jaea.go.jp/ftpnd/jendl/jendl-5-a.html>

The work on the program was divided into two stages:

- Understand the ENDF-6 format structure and read the JENDL-5 data.
- Get distribution of neutrons by alpha particle energies and by neutron energies.

In addition, a plotting block was added for the convenience of monitoring the retrieved data.

## 1.1 Structure of the ENDF-6 data storage format

The concept of the ENDF format was developed specifically for recording, storing and reading nuclear research data. The structure of files largely corresponds to the logic of writing the FORTRAN language, and initially it was the language that was used to work with the ENDF format. The hierarchy of ENDF files is very developed and has 9 steps, which allows them to be used in a wide range of tasks. Since the purpose of the work is to develop a utility that allows you to read JENDL-5 data recorded in the ENDF-6 format, then all of the following will apply only to this format.

First of all, the records are divided by the name of the target material, which is converted into a natural number **MAT** according to the rules described in ENDF-6 Formats Manual, p. 335, Appendix C [12]. It can be a single nuclide, a chemical element containing several isotopes, or a mixture/alloy of several elements. Different files are used for different values of **MAT**.

Further, each file is divided into sections having a single data format - **MF**, which is given as a natural number according to the table from ENDF-6 Formats Manual, p. 12, section 0.4.2 [12]. The **MF** formats are designed for various data types. In particular, they allow storing data of resonance parameters, reaction cross sections, angular and energy distributions of particles, data of scattering laws, data of outputs, etc. in the specified format.

Inside sections with the same format, there are partitions into tables corresponding to their type of reaction. They are set as a natural number **MT** according to the rules from ENDF-6 Formats Manual, p. 13, section 0.4.3.2 [12]. For convenience, the entire list of possible reactions for recording is listed in the table in ENDF-6 Formats Manual, p. 321, Appendix B [12].

Further it can be divided into sub-tables or other structural elements. This will be discussed in more detail on specific examples in the next section.

## 1.2 JENDL-5 Alpha-particle sublibrary structure

JENDL - is a database containing data on nuclear reactions, decays, neutron reactions, etc. [13] It consists of subsections (sublibraries) that contain experimental data on either one type of reaction or one type of particle. In the case of the NeuCBOT program, it is the subsection with  $(\alpha, n)$  reactions that are of interest.

Alpha-particle sublibrary contains data on reactions initiated by  $\alpha$ -particles with energy up to 15 MeV, for 18 nuclides from  ${}^6\text{Li}$  to  ${}^{30}\text{Si}$ . The data for each nuclide is stored in a separate file.

In a specific version of JENDL-5 Alpha-particle sublibrary, each file contains data from the section **MF** = 3 - cross-sections, and **MF** = 6 - energy-angle distributions of the reaction product, i.e. the neutron. Since the cross-section data has already been added to NeuCBOT, the section with distributions is of interest, so we consider only it further.

The set of reaction types **MT** differs from file to file, depending on whether the data of the desired type for the selected nuclide is available in this section. The  $(\alpha, n)$  reactions we need have **MT** = 50 – 54, 91. Where **MT** = 50 - means a reaction with a final core in the ground state, **MT** = 51 - in the first excited state, and so on. **MT** = 91 corresponds to a final nucleus in the state with the lowest excitation energy corresponding to a continuous spectrum.

## 1.3 Data storing format in tables for **MF** = 6

The ENDF-6 Formats Manual provides a chapter for each format **MF** with a description of the structure of stored data and the rules for writing them. In our case, with the distribution of energy and angle, the record is kept in the form of double-differential sections, determined by the formula:

$$\frac{d^2\sigma_i(\mu, E, E')}{d\mu dE'} = \sigma(E) y_i(E) f_i(\mu, E, E')/2\pi, \quad (1)$$

where  $i$  corresponds to the resulting particle,  $E$  is the energy of the resulting particle,  $E'$  is the energy of the particle flying out at an angle whose cosine equals  $\mu$ ,  $\sigma(E)$  is the total cross-section of the reaction,  $y_i(E)$  - in our case, the output of the reaction product particles,  $f_i(\mu, E, E')$  is a normalized distribution for which it is true:

$$\int dE' \int d\mu f_i(\mu, E, E') = 1 \quad (2)$$

You can learn more about the data format **MF** = 6 in the ENDF-6 Formats Manual, p. 130 [12].

## 1.4 Data storing format in tables for **MT** = 50 – 91

Each section **MF** starts with a line containing information about the data stored in this section. This is the nuclide code **ZA**, its mass in the masses of free neutrons **AWR**, the marker of the reference system **LCT** (in the files under consideration it is always laboratory), the number of ejecting reaction particles in addition to the nucleus **NK** and others. See ENDF-6 Formats Manual, pp. 131-133 [12].

Further, according to the number of outgoing particles **NK** (in our case, this is one neutron), subsections are created. They have a small service table. It stores the code of the reaction product particle **ZAP**, its mass in the masses of free neutrons **AWP**, the marker of the distribution law of the magnitude  $f_i(\mu, E, E')$  **LAW** and some other flags concerning data interpolation carried out when creating the file. See ENDF-6 Formats Manual, pp. 133, 33 [12].

Depending on the distribution law **LAW**, further tables may have a different appearance. In our case, all the reaction types of interest **MT** have **LAW** = 2, which means the type of distribution corresponding to the 2-body problem.

After the service table there is a row in which the number of subtables is indicated, each of which corresponds to its energy of the incoming particle, and in which the final data is recorded in the form of Legendre coefficients. It

is proposed to extract the distribution law using the following formula:

$$p_i(\mu, E) = \int dE' f_i(\mu, E, E') = \frac{1}{2} + \sum_{l=1}^{NL} \frac{2l+1}{2} A_l(E) P_l(\mu), \quad (3)$$

where  $NL$  is the highest degree of the Legendre polynomial present in the subtable,  $A_l(E)$  is the Legendre coefficient written in the subtable with a given energy of the incoming particle  $E$ ,  $P_l(\mu)$  is the value of the Legendre polynomial of degree  $l$  at the point  $\mu$ . Moreover, the value of  $p_i(\mu, E)$  is normalized by one. For more information, see ENDF-6 Formats Manual, p. 140 [12].

Having processed the data from the table correctly, we will get the value  $p_i(\mu, E)$ , which reflects the distribution of the number of neutrons flying out at a given angle at a given energy of the incoming  $\alpha$  particle. Now these data must be reduced to some dependence on  $E$  and  $E'$ , that is, to the type of neutron spectrum.

## 2 Transformation of the angle distribution into the neutron energy ditribution

For each nucleus, reaction type and energy of the  $\alpha$ -particle, calculations will be carried out independently according to the same algorithm. Since we are dealing with the problem of 2 bodies in a laboratory reference frame, then from kinematics, knowing the masses of reaction particles and their initial energies, we can obtain the dependence of the kinetic energy of the scattering particle on the cosine of its scattering angle. The desired formula is obtained from the laws of conservation of energy, momentum and the cosine theorem for the reaction  $A(\alpha, n)B$ .

In our case, the target  $A$  is stationary, the energy of the incoming  $\alpha$ -particle is determined, all participating particles are known, therefore their mass can be determined from the laws of conservation of energy:  $m_A + \overset{0}{\cancel{T_A}} + m_\alpha + T_\alpha = m_n + T_n + m_B + T_B$ , and the law of conservation of momentum:  $\overset{0}{\cancel{p_A}} + \vec{p}_\alpha = \vec{p}_n + \vec{p}_B$ .

For the convenience of writing and in order to reflect the nature of



calculations in the code, we will use auxiliary algebraic constructions that are often found in the final expression for the neutron energy:

$$S = 2 m_\alpha T_\alpha \cdot m_n \cdot \cos^2 \theta_n, \quad (4)$$

$$L = (m_n + m_B) (m_B (Q + T_\alpha) - m_\alpha T_\alpha) \quad (5)$$

where  $Q = (m_A + m_\alpha) - (m_B + m_n)$  - reaction energy.

So, we get:

$$T_n(\cos \theta_n) = \frac{(S + L) \pm \sqrt{S^2 + 2 \cdot SL}}{(m_n + m_B)^2} \quad (6)$$

With a positive value of  $\cos \theta_n$  it is needed to add the root. If  $\cos \theta_n$  is negative, subtract.

Now, having dependencies  $p_i(\mu, E)$  and  $T_n(\mu)$  (let me remind you that  $\mu \equiv \cos \theta_n$ , and  $E \equiv T_\alpha$ ), we have the opportunity to get the desired distribution  $p'_i(T_\alpha, T_n)$ . This can be done by inverting the function  $T_n(\mu)$  to  $\mu(T_n)$ . There is no need to get it in the form of another expression. Here is used the fact that computer calculations are carried out discretely for each value of  $\mu$  on the segment  $[-1; 1]$  and express this function explicitly at each point by writing the values  $\mu, T_n(\mu)$  to the file by columns and  $p_i(\mu, E)$ , corresponding to each other in each row.

### 3 Software implementation of the neutron spectrum calculation

The code is written in Python version 3.10.6 and is available for download on [my github page](#)<sup>2</sup>.

The project consists of:

- **plotter.py, processor.py, converter.py** - main executable files,
- **script.sh, polynomials.py, constants.py, chemistry.py** - auxiliary files,

---

<sup>2</sup><https://github.com/iv-gonch/ENDF6-reader>

- `/downloaded`, `/converted`, `/reshaped`, `/spectra`, `/graphs2D`, `/graphs3D` - folders with data. (see fig. 1)

The program is launched for each nuclide, MF and MT values separately by reading keywords from the console “**-graph**”, “**-reshape**”, “**-convert**” and “**-spectra**”, which indicate the desired outcome of the program, and auxiliary keywords that convey information about the name of the particle, the type of data, the type of reaction, etc. Keywords allow you to perform any stage of the program separately. If there is not enough data calculated at the previous stages at the start of the program, then this data is created automatically. If the data is available, it will not be re-created.

Logically, the program execution is divided into three stages: reading the JENDL-5 file and writing data in the NeuCBOT format, analyzing and converting data from the file into distribution of neutrons by alpha particle energies and by neutron energies and plotting.

### 3.1 Auxiliary functionality of the program

`polynomial.py` has only one function `getLegendre(points, lmax)`, where **points** is the number of points along the  $Ox$  axis, and **lmax** is the largest value of the order of the Legendre polynomial. Given the number of points *points* and the maximum order *lmax*, a two-dimensional array of dimension [**points** × **lmax**] is created, filled with the values  $P_l(x)$  for **points** points on the axis  $Ox$  for each of the **lmax** first orders of the Legendre polynomial.

Calculations are performed according to the recurrent formula:

$$P_0(x) = 1, \quad P_1(x) = x,$$

$$P_{l+1}(x) \cdot (l + 1) = xP_l(x) \cdot (2l + 1) - P_{l-1}(x) \cdot l. \quad (7)$$

To check the calculations in the same file there is a block for plotting Legendre polynomials of various orders.

`constants.py` created to increase the readability of the code. It stores the

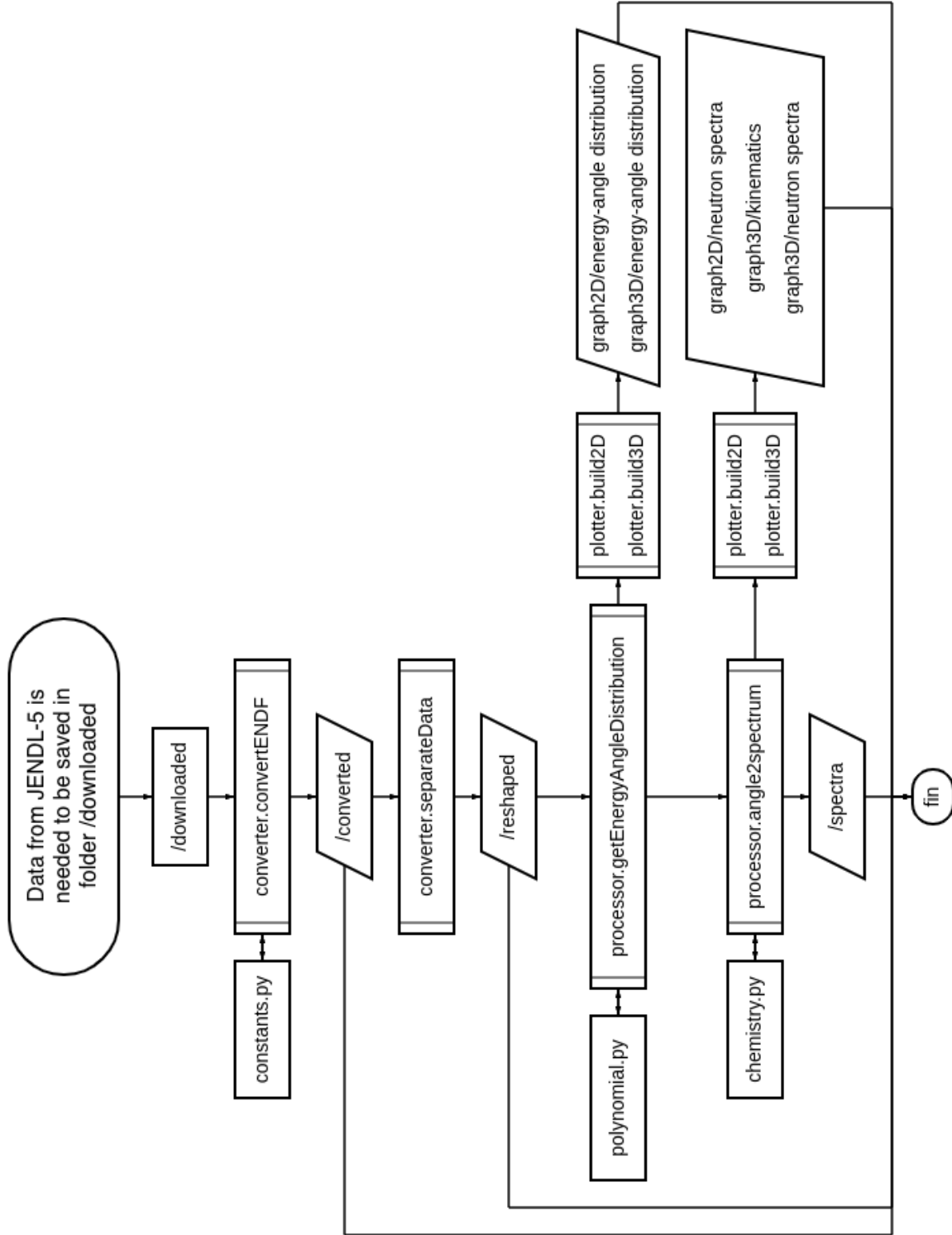


Figure 1: ENDF6-reader project outline.

constants used when reading ENDF-6. It also has short descriptions of these constants.

**chemistry.py** contains several functions that allow you to determine the name of the element by the charge number and vice versa. In addition, it is possible to determine the mass of the isotope in relative units of neutron masses by the mass and charge number.

**script.sh** allows you to initiate simultaneous execution of code with different input parameters.

## 3.2 The main functionality of the program

Below **fname** means the name of the nuclide in the format “the name of the element as in the periodic table \_ mass number of the nuclide”, for example, “C\_13”,

**MF**, **MT** - section format and reaction type, respectively,

**points** sets the number of points for plotting over the range of values of  $\cos \theta_n$  and by the range of values  $p'_i(T_{n,\alpha} = const)$ .

**converter.py** works with input data, treating them as strings, without performing any calculations, and has two main and one auxiliary functions:

- `convertENDF(fname)`, which for a given nuclide **fname** adds separating characters “|” between all elements of each row and converts an exponential number record of the form  $\pm 0.0 \pm 0$  into a record of the form  $\pm 0.0e \pm 0$  acceptable for reading `float()`.  
Then the received files are written to `/converted/fname`.
- `separateData(fname, MF, MT)`, which for a given nuclide **fname** reads strings with the corresponding input values **MF** and **MT**.  
Then the received files are written to `/reshaped/fname/MF_ MT_/NK_ NE_`. Where the values **MF**, **MT**, etc. are specified in place of the “\_” symbol.
- `convertLine(line)` is an auxiliary function called during `separateData` execution, which reads the string **line** from the converted file and

converts it into an array of numbers, and determining their type (integer or float).

After splitting the data into separate files, it becomes algorithmically easier to write data into fixed-length arrays. This opens up the possibility of using the numpy library, which allows you to perform calculations more accurately and quickly.

**processor.py** it is used for calculations and consists of two main and one auxiliary functions:

- `getenergyangledistribution(fname, MF, MT, points, normcheck)` alternately opens files corresponding to different energies from the folder `/reshaped/fname/MF□_MT□` and reads data from files into a two-dimensional array **Coeff**. This array contains Legendre coefficients for each energy of an incoming  $\alpha$  particle. Then passes this array to the `legendre2angle` function, which returns a two-dimensional array of distribution values  $p_i(\cos \theta_n, T_\alpha)$ . If the distribution data is in the database and the flag **normcheck** = True, an additional check is performed for the normalization of the values obtained according to the indication that all values of  $p_i$  are normalized. See ENDF-6 Formats Manual, pp. 140 [12].

- `legendre2angle(Coeff, points, NE)` first gets a two-dimensional array of values of Legendre polynomials of different degrees at points from `polynomials.getLegendre(points,maxNW+1)`. Where **maxNW** is the maximum Legendre coefficient number  $A_l$  in the file. It is important to note that in the formula (3), summation begins with  $l = 1$ , as does the recording of coefficients  $A_l$ , and the calculation of Legendre polynomials with  $l = 0$ , so we pass the value one more.

After obtaining an array with the values of Legendre polynomials using the formula (3), the values of a two-dimensional array **S** of dimension **[NE × points]** are calculated, which corresponds to the distribution  $p_i(\cos \theta_n, T_\alpha)$ .

**NE** - the number of different energies of  $\alpha$ -particles in the array **Coeff**.

- `angle2spectrum(fname, MF, MT, points, NK, NE, E_in, S, isData)`, if

there is data (**isData** = True), converts the distribution  $p_i(\cos \theta_n, T_\alpha)$  stored in **S**[**NE** × **points**] in the final distribution of  $p'_i(T_\alpha, T_n)$ , writing it to the array **E\_n** that the same dimensions. The **E\_in** array stores the energies of incoming  $\alpha$  particles. From **chemistry.getMass(ZA)** we find the masses of  $\alpha$ -particles, initial and final nuclei in relative neutron masses. Next, we apply the previously derived reaction kinematics formula (6). In the case of an excited nucleus, mass recalculation is not provided, since the change is  $\sim 10^{-3}\%$ .

## 4 Analysis of the obtained results

The result of the work, in addition to the calculated distributions of neutron yields by energy for the first four excited states of  $(\alpha, n')$  reactions of 13 light nuclides, is the ENDF6-reader utility, available for download to any Internet user.

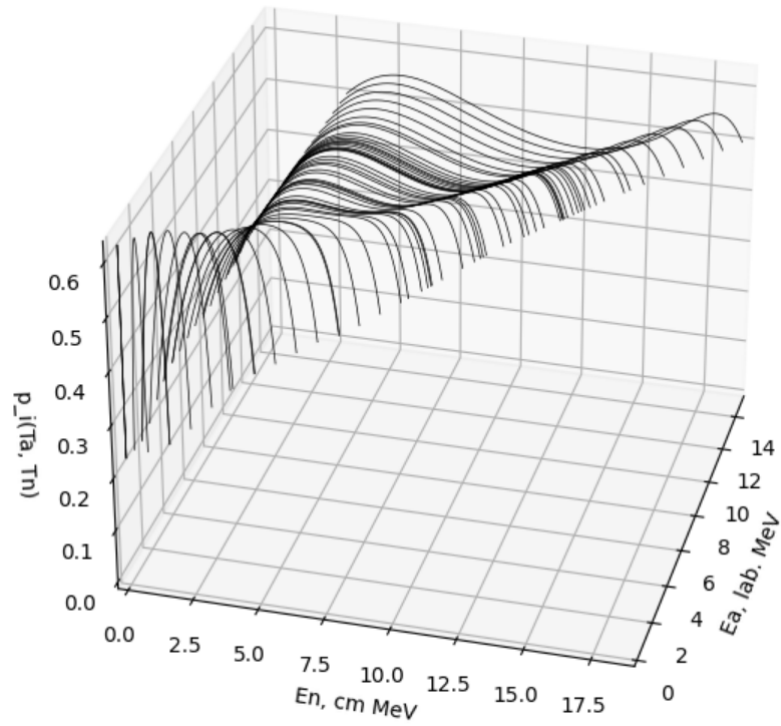
When analyzing the calculated data, some questions arose: below are several obtained graphs of neutron distributions, the appearance of which left doubts about their correctness (see fig. 2). Questions arise because of the atypical shape of the distribution - the maximum is reached at the boundary of the range of values. To identify a possible error, an analysis of all stages of calculations was carried out (see fig. 3, 4, 5), but it was not detected.

There is reason to believe that the type of dependence of the neutron yield on their energies (spectrum) will change when calculating it using the formula (3) from the article [11]:

$$Y_i(T_n) = \sum_{\alpha} P_{\alpha} \sum_m \frac{N_A C_m}{A_m} \int_0^{T_{\alpha}} \frac{\sigma_m(T'_{\alpha}, T_n)}{\xi(T'_{\alpha})} dT'_{\alpha}, \quad (8)$$

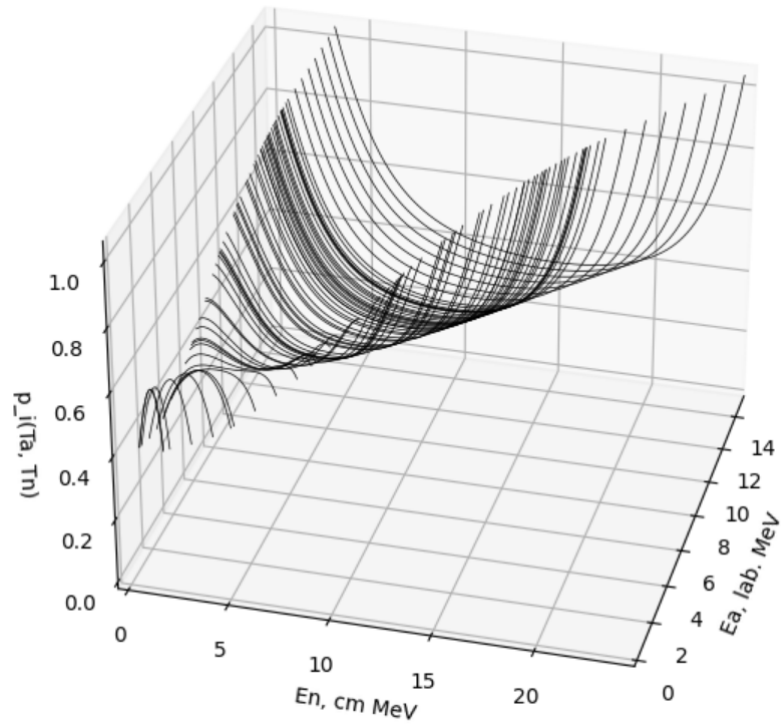
where  $P_{\alpha}$  - the weight specified for  $\alpha$  is the particle with the initial energy  $T_{\alpha}$  or the branching coefficient,  $N_A$  - Avogadro number  $C_m$  is the mass fraction of the nuclide in the target substance,  $A_m$  is its mass number  $\sigma_m(T'_{\alpha}, T_n)$  is the reaction cross section for a given nuclide,  $\xi(T_{\alpha})$  is the stopping power of the material. The first summation is performed for all  $\alpha$  particles, while the second summation is taken for all nuclides in accordance with the composition

B\_10 MF6 MT50



(a)

O\_18 MF6 MT50



(b)

Figure 2: Distribution of neutrons by alpha particle energies and by neutron energies for  $(\alpha, n)$  reaction for a)  $^{10}\text{B}$  and b)  $^{18}\text{O}$ .

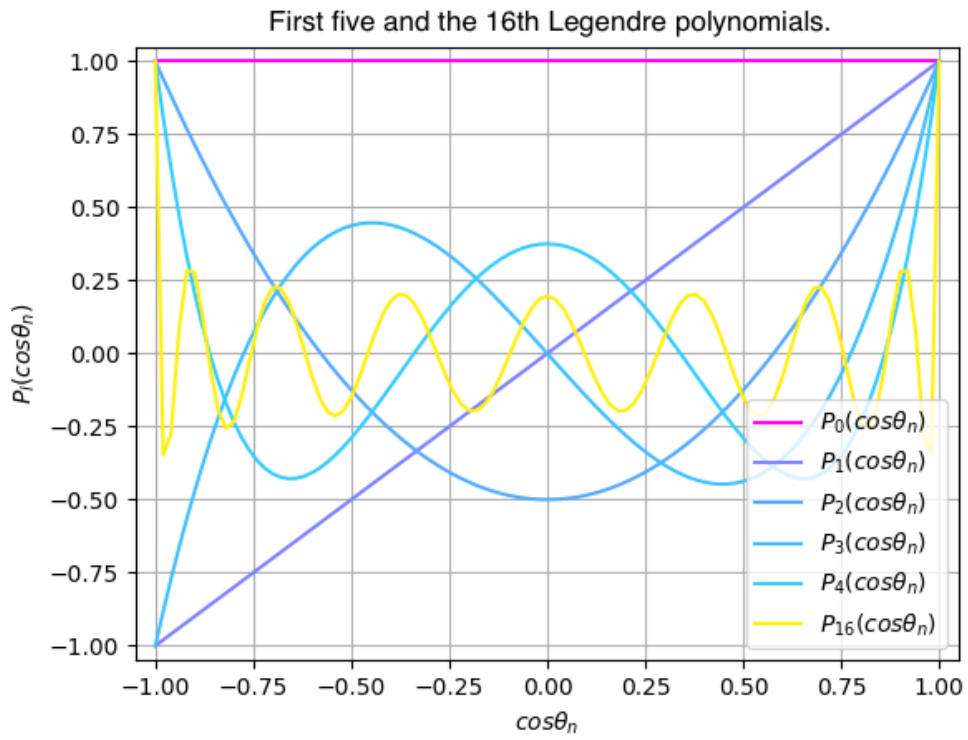


Figure 3: Checking the calculation of Legendre polynomials.

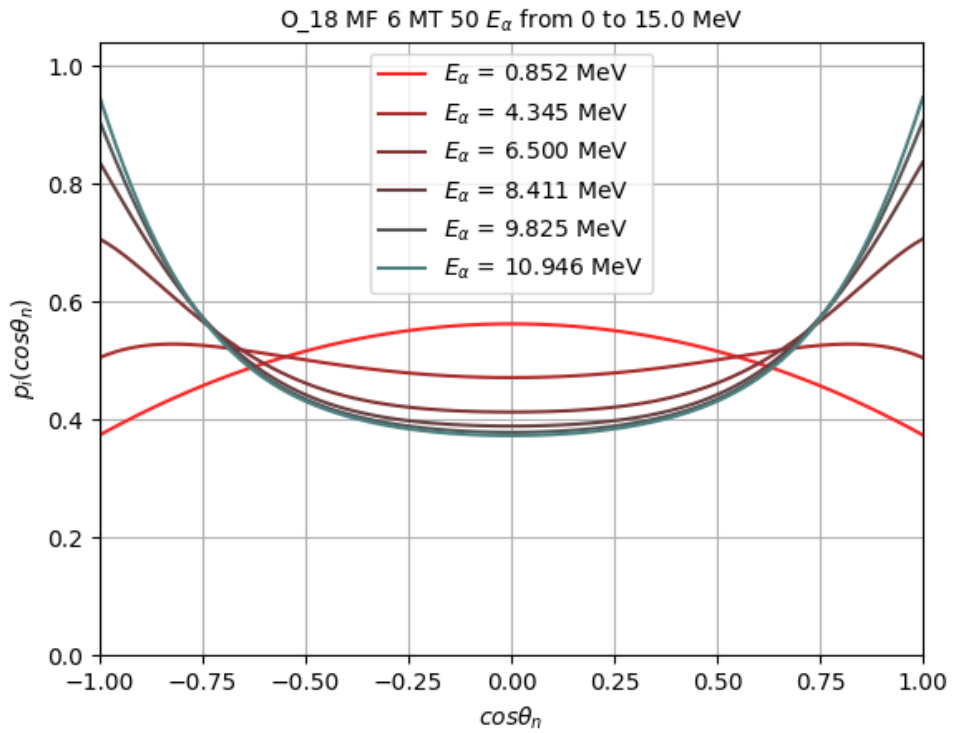


Figure 4: Checking the angle distribution for  $^{18}\text{O}$ .



Li\_7 MF6 MT50

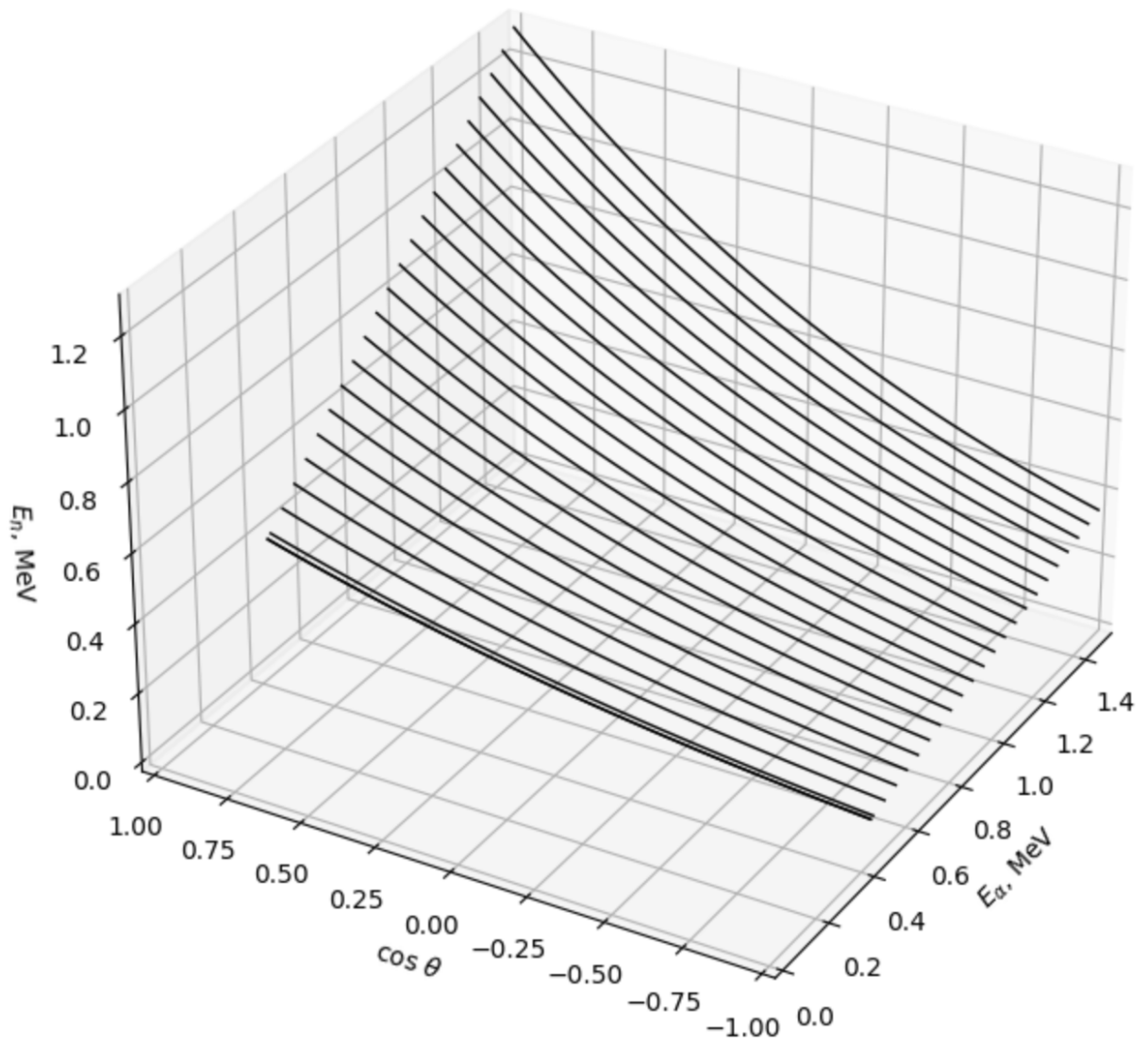


Figure 5: Checking the kinematic relationship of the particle energies and scattering angle for  ${}^7\text{Li}$ .

of the material and the specified isotope content. In this expression, the integral is taken from zero, since it is assumed that the probability of capturing a  $\alpha$  particle is small, so that a particle moving in a substance can lose all its energy.

To link the formula (8) (immediately replace the notation  $\cos \theta_n, T_\alpha, T_n$  by  $\mu, E, E'$ , respectively) with the calculated values of  $p'_i(E, E')$ , we will integrate the formula (1) by  $\mu$ :

$$\int d\mu \frac{d^2\sigma_i(\mu, E, E')}{d\mu dE'} = \frac{\sigma(E) y_i(E)}{2\pi} \int d\mu f_i(\mu, E, E') \Leftrightarrow \quad (9)$$

$$\frac{d^2\sigma_i(E, E')}{d\mu dE'} = \frac{\sigma(E) y_i(E)}{2\pi} p'_i(E, E'). \quad (10)$$

Thus, using the calculated value of  $p'_i(E, E')$ , it is possible to obtain a double-differential cross section  $\frac{d^2\sigma_i(E, E')}{d\mu dE'}$ , which is written in the formula (8) as  $\sigma_m(T_\alpha, T_n)$ .

## Conclusion

The goal was to obtain neutron spectra in NeuCBOT using information from JENDL-5. To achieve this, the data storage format in the database was studied. Based on the knowledge gained, a utility has been developed to read information from this database. A methodology for converting the data from the database into the required calculation data and then for calculating neutron spectra was also developed. Test calculations were performed and analyzed. The values of the distributions  $p'_i(E, E')$  obtained as a result of the work can be used to perform neutron spectrum calculations using the NeuCBOT program. Further development will be carried out in this direction and it will include the adaptation of the NeuCBOT code to work with the obtained distributions  $p'_i(E, E')$  and the calculation of neutron spectra with its help.

## References

1. S. Westerdale, NeuCBOT (Neutron Calculator Based On TALYS). <https://github.com/shawest/neucbot>.
2. S. Westerdale and P. D. Meyers, Nucl. Instrum. Methods Phys. Res., Sect. A **875**, 57 (2017); arXiv: 1702.02465 [physics.ins-det].
3. W. B. Wilson et al., Tech. Rep. LA-13639-MS (Los Alamos Natl. Laboratory, Los Alamos, 1999).
4. W. B. Wilson, R. T. Perry, W. S. Charlton, T. A. Parish, and E. F. Shores, Radiat. Prot. Dosim. **115**, 117 (2005).
5. G. N. Vlaskin, Tech. Rep. VNIINM 06-1 (VNIINM, 2006).
6. G. N. Vlaskin, Y. S. Khomyakov, and V. I. Bulanenko, At. Energy **117**, 357 (2015).
7. G. Vlaskin and Y. Khomiakov, EPJ Web Conf. **153**, 07033 (2017).
8. G. Vlaskin and Y. Khomiakov, At. Energy **130**, 104 (2021).
9. D. M. Mei, C. Zhang, and A. Hime, Nucl. Instrum. Methods Phys. Res. A **606**, 651 (2009); arXiv:0812.4307 [nucl-ex].
10. E. Mendoza, D. Cano-Ott, P. Romojaro, V. Alcayne, P. Garcia Abia, V. Pesudo, L. Romero, and R. Santorelli, Nucl. Instrum. Methods Phys. Res., Sect. A **960**, 163659 (2020); arXiv: 1906.03903 [hep-ph].
11. М.Б. Громов, Ш. Вестердейл, И.А. Гончаренко, А.С. Чепурнов, Ядерная физика **86**(2), 353 (2023).
12. A. Trkov, M. Herman, and D. A. Brown. ENDF-6 Formats Manual. Upton, NY. 11973-5000: 2018.
13. O. Iwamoto et al., Japanese evaluated nuclear data library version 5: JENDL-5, J. Nucl. Sci. Technol. **60**(1), 1 (2023). DOI: 10.1080/00223131.2022.2141903 (Open Access).