



Estimating theoretical uncertainties of the two-nucleon observables by using backpropagation

K. Topolnicki^a, R. Skibiński, J. Golak

M. Smoluchowski Institute of Physics, Jagiellonian University, Profesora Stanisława Łojasiewicza 11, 30348 Kraków, Poland

Received: 9 December 2024 / Accepted: 30 April 2025

© The Author(s) 2025

Communicated by Vittorio Somà

Abstract We present a novel approach to calculating theoretical uncertainties in few-nucleon calculations, making use of automatic differentiation via backpropagation, which is particularly efficient when there are many input variables but only a few outputs. The methods described in this paper constitute tools that can be used to investigate the properties of scalar functions used to define nuclear potentials and quantify their contribution to the uncertainty of few nucleon calculations. We demonstrate these methods in deuteron bound state and nucleon–nucleon scattering calculations. Backpropagation, implemented in the **Python pytorch** library, is used to calculate the gradients with respect to model parameters and propagate errors from these parameters to the deuteron binding energy and selected phase-shift parameters. The uncertainty values obtained using this approach are validated by directly sampling from the potential parameters. We find very good agreement between two ways of estimating that uncertainty.

1 Introduction

The progress of research on the nuclear potential, and in particular the increasing precision of scattering experiments, has drawn researchers' attention to estimating the uncertainty of the theoretical models used [1]. In the first stage, the influence of experimental uncertainties on the determination of free potential parameters was examined. An important step in that direction was the preparation of a self-consistent database for nucleon-nucleon scattering data by the group from Granada [2] and using it to determine the uncertainty of the parameters of the OPE-Gaussian [3] and other potentials along with the corresponding correlation coefficients. Also the introduction of new generations of few-nucleon forces from the Chiral Effective Field Theory, see e.g. reviews [4–

6], raises the important question of determining the errors associated with few-nucleon quantum mechanical calculations. The concept of determining the correlations between potential parameters was taken over by the Bochum group, which delivered uncertainties in the potential parameters and the resulting uncertainties in nucleon-nucleon phase shifts for the chiral SMS potential [7]. Using statistical analysis methods in [8,9] it was examined how these uncertainties are transferred to observables in elastic nucleon-deuteron scattering and in the deuteron breakup reaction. These methods involve calculating observables for various sets of potential parameters and then examining the obtained probability distributions for individual observables. These distributions depend on both the reaction energy and kinematic variables, e.g. scattering angles. It turned out that the uncertainties of the observables related to the uncertainties of the potential parameters are relatively small and usually remain below 1% for the differential cross section. Similar study but based on approximated solution of three-nucleon problem has been presented in [10], where also other types of uncertainties are discussed. In [11] some uncertainty related to 3NF parameters have been estimated for the nucleon-deuteron scattering by using the approximated solutions of the Faddeev equations. The rapidly increasing computational complexity with the number of nucleons makes investigations based on statistical methods very challenging for many-nucleon systems. Recently new technique based on reduced-basis method has been developed to create and train emulators for many-nucleon systems, see [12,13].

It should be remembered that among other types of theoretical uncertainties related to the chiral potential two play a dominant role. The first is the uncertainty related to the value of the cut-off parameter used in the potential regularization. The second source of uncertainty is the neglect of higher terms of chiral expansion in the nuclear potential, which leads to the so-called truncation errors. Estimati-

^a e-mail: kacper.topolnicki@uj.edu.pl (corresponding author)

ing the uncertainties associated with different cut-off values involves repeating the calculations, in practice for a few values (four in case of the SMS chiral potential [7]). It should be remembered that the regulator value also affects the values of the free potential parameters, so in fact we are dealing with several versions of the potential. The uncertainty associated with using different cut-off values is typically of the order of 1–3% for nucleon-deuteron scattering and for deuteron and ^3He photodisintegration [14]. The estimation of truncation errors was proposed in [15, 16], which treated observables as quantities subject to chiral expansion. This recipe was then extended to many-nucleon processes in [17]. The truncation errors remain typically below 10% depending strongly on the order of chiral expansion and reaction energy for nucleon-deuteron scattering and for ^2H or ^3He photodisintegration [14]. It was shown that this prescription is consistent with the estimation of truncation errors resulting from Bayesian analysis methods with a particular choice of priors [18, 19]. The role of various uncertainties in the nuclear structure calculations as well as various aspects of uncertainty quantification are currently the subject of many efforts, see e.g. [20–22] and references therein.

Despite the fact, that both the cut-off dependence and truncation errors of two- and many-nucleon observables turned out to be usually bigger than the above mentioned dependence on potential parameters, in this work we focus on the uncertainty of two-nucleon observables arising from uncertainty of potential parameters. We demonstrate that software libraries designed for machine learning can be used to investigate these uncertainties. Using these tools to propagate errors from model parameters to observables is, to the best of our knowledge, a novel alternative to using standard statistical methods.

Few nucleon potentials are typically expressed as a linear combination of isospin–spin–momentum operators and scalar functions. The form of the operators is fixed and the scalar functions effectively define the force. In practice, the possibility to investigate the properties of scalar functions that define nuclear interactions and quantify their contribution to the uncertainty of few nucleon calculations using the tools described in this paper is valuable in the development and debugging of a numerical implementation of these potentials.

We use the **Python** library **pytorch** [23] to implement the numerical calculation of the deuteron bound state energy and nucleon–nucleon scattering observables. This library allows for an efficient implementation of numerical calculations. The numerical operations within this library are vectorized making it possible to automatically parallelize their execution. Most importantly, **pytorch** offers the possibility to use backpropagation [24] to calculate the gradients of the results obtained from computations. This will be used to estimate errors of the bound state energy and nucleon–nucleon phase

shifts that result from uncertainties in the two-nucleon (2N) potential.

The theoretical introduction to the calculations is formulated in the language of “tensors”. This term is not used to refer to the mathematical construct but to multidimensional arrays. Numerical operations mapped over multidimensional arrays are relatively straightforward to parallelize over multiple CPU threads on graphics processing units making “tensors” a popular fundamental data type in many python libraries designed for machine learning and numerical calculations. Formulating the calculations in this language makes them easier to translate to other libraries, for example **tensorflow** [25], or if automatic differentiation is not required **numpy** [26].

The input to our calculation is a set of scalar functions that define the two nucleon potential. For our purposes we will be using the well known Bonn B [27] potential but by simply changing the scalar functions, other more modern potentials can be used. Additionally, we will assume that the values of these scalar functions are normally distributed to simulate errors associated with the functions. The parameters of these distributions are an additional part of the input to our algorithms. Since the scalar functions are represented as a discrete lattice of values, the derivatives of the calculated observables with respect to the, hundreds of thousands, of these values needs to be calculated in order to investigate the influence of errors on the input on the calculated quantities. For this reason the calculations in this paper rely heavily on efficient algorithms to perform automatic differentiation. Many current libraries including **pytorch** rely on, so called, backward mode automatic differentiation or backpropagation. In this approach the algorithm keeps track of the numerical operations used to calculate a given result in the form of a graph. When the calculation is complete, the graph can be followed backwards from the output to the inputs and the chain rule can be used to calculate the relevant gradients. Nodes of the graph may point to intermediate results of the calculation making this process more efficient. In another approach to automatic differentiation, forward mode automatic differentiation, the calculation is performed in only one direction. This type of procedure is also widely used, for example in the study [28]. In a very basic form of forward mode automatic differentiation the numerical types used in the calculation can be replaced by appropriate dual numbers [29]. The additional information contained in these numbers makes it possible to track a given derivative throughout the computation.

The paper is organized as follows. In Sect. 2 we outline our approach to calculating the deuteron bound state. Section 2.1 contains a description of the implementation of the calculation in the **pytorch** library and Sect. 2.2 contains a discussion of the numerical results and obtained error estimates. Next, in Sect. 3 we outline the theoretical formalism used in nucleon–nucleon scattering calculations. Section 3.1

contains details related to the implementation of these calculations in the **pytorch** library and Sect. 3.2 contains a discussion of the numerical results and obtained error estimates. Finally, Sect. 4 contains the summary and outlook.

2 The deuteron bound state

We will use the Schrödinger equation in integral form to calculate the deuteron bound state:

$$(E_d - H_0)^{-1} V |\phi_d\rangle \equiv K(E_d) |\phi_d\rangle = |\phi_d\rangle. \quad (1)$$

In this equation $|\phi_d\rangle$ is the deuteron bound state, E_d is the deuteron binding energy, $H_0 = \frac{\mathbf{p}^2}{m}$ is the kinetic energy operator with $\mathbf{p}$ being the relative 2N momentum operator, V is the 2N potential operator and m is the nucleon mass. For practical calculations the deuteron binding energy E_d in (1) is replaced by E and a slightly different equation is solved:

$$K(E) |\phi\rangle = \lambda |\phi\rangle. \quad (2)$$

The bound state is found by searching for a value of E such that there exists an eigenvalue λ in (2) equal to 1 up to assumed precision. This value of E , if found, is taken as the deuteron bound state energy and the corresponding eigenstate $|\phi\rangle$ as the deuteron bound state.

The fundamental element in this calculation is the 2N potential V . Following [30] we will assume that the potential satisfies:

$$\langle t' m'_t | V | t m_t \rangle = \delta_{t't} \delta_{m'_t m_t} V^{t m_t} \quad (3)$$

where t (t') is the total 2N isospin in the initial (final) state, m_t (m'_t) is the projection of the isospin in the initial (final) state and $V^{t m_t}$ is the potential operator in the joined spin and momentum space of the 2N system.

The 2N potential can be expanded into a linear combination of scalar functions and operators [31]:

$$\langle \mathbf{p}' | V^{t m_t} | \mathbf{p} \rangle = \sum_{i=1}^6 v_i^{t m_t} (p', p, \hat{\mathbf{p}}' \cdot \hat{\mathbf{p}}) w_i(\mathbf{p}', \mathbf{p}) \quad (4)$$

where $v_i^{t m_t} (p', p, \hat{\mathbf{p}}' \cdot \hat{\mathbf{p}})$ are scalar functions of the relative 2N momentum magnitudes $|\mathbf{p}| = p$ ($|\mathbf{p}'| = p'$) in the initial (final) state and $w_i(\mathbf{p}', \mathbf{p})$ are operators in the spin space of the 2N system [30]. Since in this section we are concentrating on deuteron bound state calculations, we will be using only the V^{00} part of the potential and drop the upper index 00 in V^{00} and v_i^{00} for brevity.

Our choice for the set of the w_i operators is the same as in [30]:

$$\begin{aligned} w_1(\mathbf{p}', \mathbf{p}) &= 1, \\ w_2(\mathbf{p}', \mathbf{p}) &= \sigma(1) \cdot \sigma(2), \\ w_3(\mathbf{p}', \mathbf{p}) &= i(\sigma(1) + \sigma(2)) \cdot (\mathbf{p} \times \mathbf{p}'), \\ w_4(\mathbf{p}', \mathbf{p}) &= \sigma(1) \cdot (\mathbf{p} \times \mathbf{p}') \sigma(2) \cdot (\mathbf{p} \times \mathbf{p}'), \\ w_5(\mathbf{p}', \mathbf{p}) &= \sigma(1) \cdot (\mathbf{p}' + \mathbf{p}) \sigma(2) \cdot (\mathbf{p}' + \mathbf{p}), \\ w_6(\mathbf{p}', \mathbf{p}) &= \sigma(1) \cdot (\mathbf{p}' - \mathbf{p}) \sigma(2) \cdot (\mathbf{p}' - \mathbf{p}), \end{aligned} \quad (5)$$

where $\sigma(i)$ is a vector of Pauli matrices acting in the space of particle i . The 2N potential is determined by the scalar functions v_i . These scalar functions will be represented numerically as discrete values on a lattice. More specifically, if the momenta p' , p share the same discrete values:

$$p_1, p_2 \dots p_N \quad (6)$$

and the cosine of the angle between the momenta, $\hat{\mathbf{p}}' \cdot \hat{\mathbf{p}}$, is also discretized and can take on values:

$$x_1, x_2 \dots x_M, \quad (7)$$

then the scalar functions will be represented as an array $\mathcal{V}$ whose elements are:

$$\begin{aligned} \mathcal{V}_{ijkl} &= v_i(p_j, p_k, x_l), \\ i &= 1 \dots 6, \quad j = 1 \dots N, \\ k &= 1 \dots N, \quad l = 1 \dots M. \end{aligned} \quad (8)$$

When integration over momentum or angle is required, integration weights $\mathcal{W}_{i=1 \dots N}^p$ or $\mathcal{W}_{i=1 \dots M}^x$ will be used for p , p' or x respectively. Our calculations make use of Gaussian quadrature points and weights.

Our numerical calculation will use the array (8) as input. In the first step, the machine learning library **pytorch** [23] is used to calculate the partial wave decomposition of the 2N potential which in turn will be used to solve the eigenproblem (2). Finally, we will use the backpropagation algorithm implemented in **pytorch** to calculate gradients of the resulting eigenvalues λ from (2) with respect to the elements of $\mathcal{V}$. This will allow us to propagate the theoretical uncertainties from the $\mathcal{V}$ array to the eigenvalues λ that are closest to 1 for a given energy. Since the dependence of the eigenvalue closest to 1 on the energy $\lambda(E)$ is used to estimate the deuteron binding energy $\lambda(E_d) = 1$, knowing the uncertainty of $\lambda(E)$ will also determine the uncertainty of the deuteron binding energy estimate.

2.1 Implementation using **pytorch**

The partial wave decomposition of the 2N potential was performed using the procedure from [30]. Since our goal is to use backpropagation, the implementation was written entirely using methods and functions from the **pytorch** library.

Due to the rotational invariance of the 2N potential, its partial-wave matrix element is determined in [30] by a simpler function H :

$$\langle p'(l's)jm_j | V | p(ls)jm_j \rangle = H(p', p; l', l, s, j). \quad (9)$$

Here

$$|p(ls)jm_j\rangle \quad (10)$$

are states with the magnitude of the relative momentum p . The states carry information about the orbital angular momentum of the two-particle system, l , which is coupled with the spin s to form a state with the total angular momentum j with projection m_j . Practical numerical calculations are carried out by considering only a finite subset of partial wave states $|p(ls)jm_j\rangle$. For deuteron calculations it is sufficient to consider only two partial wave states, however in our calculations we consider all partial wave states with $j \leq 5$ for bench-marking purposes.

The explicit form of H , for arguments where it has non zero values, is [30]:

$$\begin{aligned} H(p', p; l', l, s, j) = & 8\pi^2 \int_{-1}^1 d(\cos \theta') \frac{1}{2j+1} \sum_{m_j=-j}^j \\ & \times \sum_{m'_l=-l'}^{l'} c(l', s, j; m'_l, m_j - m'_l, m_j) \\ & \times \sum_{m_l=-l}^l c(l, s, j; m_l, m_j - m_l, m_j) \\ & \times Y_{l'm'_l}^*(\theta', 0) Y_{lm_l}(0, 0) \\ & \times \langle \mathbf{p}'; s, m_j - m'_l | V | \mathbf{p}; s, m_j - m_l \rangle \end{aligned} \quad (11)$$

where $\mathbf{p} = (0, 0, p)$, $\mathbf{p}' = p'(\sin \theta', 0, \cos \theta')$, $c(j_1, j_2, j; m_1, m_2, m)$ are Clebsch–Gordan coefficients, $Y_{lm}(\theta, \phi)$ are spherical harmonics and $|\mathbf{k}; s, m_s\rangle$ are states with relative 2N momentum $\mathbf{k}$ and spin s with projection m_s . The numerical calculation of (9) can be obtained by substituting (4) in (11). Assuming that the momentum magnitudes and cosines of angles are discretized, (9) can be approximated by:

$$\langle p_j \alpha | V | p_k \beta \rangle \approx \sum_{i=1}^6 \sum_{l=1}^M \mathcal{H}_{i\alpha k \beta l} \mathcal{W}_l^x \mathcal{V}_{ijkl} \equiv \mathcal{V}_{\alpha k \beta}^{pwd}, \quad (12)$$

where $\mathcal{W}_l^x$ are integration weights for (7), and α (β) are the complete set of discrete quantum numbers in the final (initial) state from (9). Finally, the values of $\mathcal{H}_{i\alpha k \beta l}$ are calculated by taking the integrand from (11) with $p' = p_j$, $p = p_k$ and substituting the w_i operator from (5) for the potential V .

Each of the three arrays $\mathcal{H}$, $\mathcal{W}^x$ and $\mathcal{V}$ is represented by **pytorch** “tensors”. The first “tensor” $\mathcal{H}$ was calculated, for a given set of discrete quantum numbers, using symbolic programming in *Mathematica* [32]. The resulting expressions were translated into **Fortran** and then used to create a **Python** module with **f2py**, a part of the **numpy** library [26]. The second “tensor” $\mathcal{W}^x$ contains Gaussian integration weights. Only the last “tensor” $\mathcal{V}$ requires the gradient since values of its elements are assumed to have uncertainties. The values of $\mathcal{V}$ are calculated in a separate program for the Bonn B [27] potential.

The sum in (12) is calculated using the **einsum** method from the **pytorch** library. The created $\mathcal{V}^{pwd}$ “tensor” together with the integration weights for (6), $\mathcal{W}_l^p$, can be used to construct a discrete representation of the $K(E)$ operator from (2). The action of this operator on a 2N state $|\phi\rangle$ given in partial wave representation:

$$\langle p_j \alpha | K(E) | \phi \rangle \approx \sum_{k=1}^N \sum_{\beta} \mathcal{K}(E)_{j\alpha k \beta} \langle p_k \beta | \phi \rangle, \quad (13)$$

where the second sum is over a finite subset of all possible discrete quantum numbers β , is given in terms of the $\mathcal{K}(E)_{j\alpha k \beta}$ array:

$$\begin{aligned} \mathcal{K}(E)_{j\alpha k \beta} &= \left(E - \frac{p_j^2}{m} \right)^{-1} \langle p_j \alpha | V | p_k \beta \rangle p_k^2 \mathcal{W}_k^p \\ &\equiv \left(E - \frac{p_j^2}{m} \right)^{-1} \mathcal{V}_{j\alpha k \beta}^{pwd} p_k^2 \mathcal{W}_k^p. \end{aligned} \quad (14)$$

The $\mathcal{K}(E)_{j\alpha k \beta}$ “tensor” can be constructed using **pytorch** methods and then reshaped into a two dimensional matrix. This matrix can be used to solve the eigenproblem (2) with the **eigvals** function. Because all operations, including the calculation of the eigenvalue, are performed within the **pytorch** library it is possible to calculate the gradient of a specific eigenvalue with respect to elements of $\mathcal{V}$:

$$\frac{\partial \lambda}{\partial \mathcal{V}_{ijkl}} \quad (15)$$

using backpropagation. In practice this is done in two steps. First the **backward()** method is called on a given eigenvalue. Next the gradient can be read from the **grad** field of the **pytorch** “tensor” corresponding to $\mathcal{V}$. In practice we will

calculate gradients of eigenvalues that are closest to 1 for a given energy E .

Finally, we introduce uncertainties to the elements of $\mathcal{V}$ assuming their standard deviations arise from the uncertainties of the scalar functions v . For the purposes of this paper we use the following choice:

$$\sigma(\mathcal{V}_{ijkl}) = \alpha(i) |\mathcal{V}_{ijkl}|, \quad (16)$$

in which the standard deviation σ is not constant across the scalar function values but proportional to their absolute value. We allow the proportionality factor $\alpha(i)$ to have different values for different operators w_i from (4). The values of $\alpha(i)$ are an important part of the input to our calculations, changing them allows the investigation of the influence of uncertainties associated with scalar functions on the uncertainty associated with the calculated observables. Additionally we assume that there are no correlations, so that the errors can be propagated using:

$$\sigma(\lambda) = \sqrt{\sum_{ijkl} \sigma(\mathcal{V}_{ijkl})^2 \left(\frac{\partial \lambda}{\partial \mathcal{V}_{ijkl}} \right)^2}. \quad (17)$$

Adding correlations is possible but it would significantly increase the memory requirements of the calculation since we need to treat the hundreds of thousands of values in $\mathcal{V}$ as variables for automatic differentiation; we plan to investigate correlated errors in future work. Note that we assign errors to individual elements of $\mathcal{V}$. The scalar functions are continuous but we associate uncertainties only with the values of the scalar functions at specific points.

The next section contains numerical results and describes the procedure to obtain the deuteron binding energy and its uncertainty.

2.2 Results for the deuteron binding energy

Equation (2) is solved by scanning different values of E and searching for an energy for which $\lambda \approx 1$. This is illustrated in Fig. 1 where we limit the errors to a single scalar function ($\alpha(1) = 0.2$, $\alpha(j) = 0$ if $j \neq 1$). The dots mark the eigenvalue that is closest to 1 for a given energy. The error bars are calculated from Eq. (17) with the gradients obtained from backpropagation. We will approximate this data using a linear function, with the solid line being the result of a least squares regression. The plot title contains an estimate of the deuteron binding energy E_d , calculated as the energy value for which the line crosses 1 and the error associated with this result.

For one value of the energy in Fig. 1, there are additional gray dots. These values were calculated by directly sampling new $\mathcal{V}'$ “tensors” from the normal distribution with standard

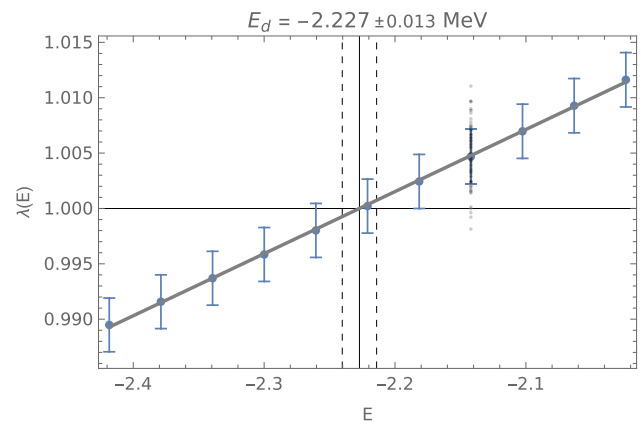


Fig. 1 Eigenvalues closest to 1 for different values of E in MeV from (2). The standard deviation of the scalar function values is taken to be $\sigma(\mathcal{V}_{ijkl}) = 0.2 |\mathcal{V}_{ijkl}|$, other values have no uncertainty. Gaussian quadrature points and weights were used both for momenta and angles. For momenta in (6), 48 points were used in the interval $(0 \text{ fm}^{-1}, 20 \text{ fm}^{-1})$. For x in (7), 48 points were used in the interval $(-1, 1)$. All discrete quantum numbers with total angular momentum less than or equal to 5 were used in the partial wave representation. The error bars show the standard deviation calculated obtained using backpropagation for a given energy

deviation (16) and mean $\mathcal{V}$, solving (2) and using the eigenvalue that is closest to 1. A histogram of these sampled points is compared with the uncertainty calculated using backpropagation in Fig. 2. The blue solid line is the normal distribution probability density function with parameters determined from (17). Although the error $\alpha(1)$ is quite large a comparison with the histogram shows that the sampled distribution is well approximated by a Gaussian. The values of the standard deviations are also in good agreement for backpropagation and sampling. Finally, Fig. 1 contains an estimation of the deuteron binding energy and the uncertainty of this estimate. These values were obtained in two steps. Firstly, the uncertainty for eigenvalues at individual energies, obtained from (17) with $\frac{\partial \lambda}{\partial \mathcal{V}_{ijkl}}$ calculated using backpropagation and assuming (16), is taken into account when calculating the least squares linear fit. Secondly, the deuteron binding energy is calculated as the intersection of the fitted line with 1 and the uncertainty of the deuteron binding energy is calculated from the errors of the least squares regression. Note that the bound state calculations were performed considering all partial wave states with total angular momentum less than or equal to 5. For the deuteron it is sufficient to consider only two partial wave states, so increasing the number of considered states served as a benchmarking test for our method. Indeed, including this large number of partial waves still reproduces the correct binding energy for the Bonn-B [27] potential.

Using gradients, a linear approximation of the λ dependence on the scalar function values can be constructed allowing the errors to be calculated using simple methods. This approximation is justified if the uncertainties of the scalar

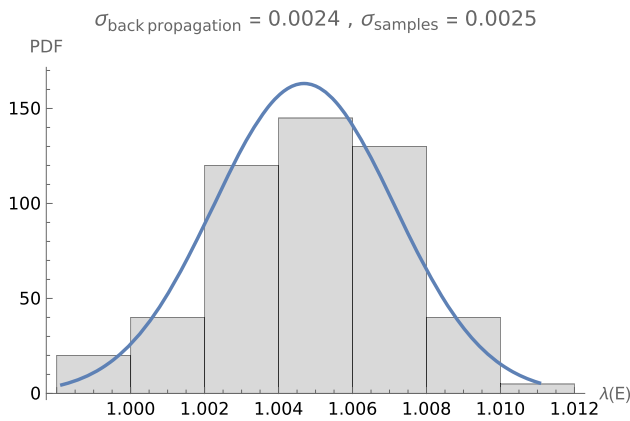


Fig. 2 Histogram of eigenvalues closest to 1 for single energy $E = -2.14206$ MeV from Fig. 1. The blue line is the normal distribution probability density function (PDF) with parameters calculated using (17). Gaussian quadrature points and weights were used both for momenta and angles. For momenta in (6), 48 points were used in the interval $(0 \text{ fm}^{-1}, 10 \text{ fm}^{-1})$. For x in (7), 48 points were used in the interval $(-1, 1)$. All discrete quantum numbers with total angular momentum less than or equal to 5 were used in the partial wave representation. The data was tested for normality using the Kolmogorov-Smirnov test, the p value is 0.955

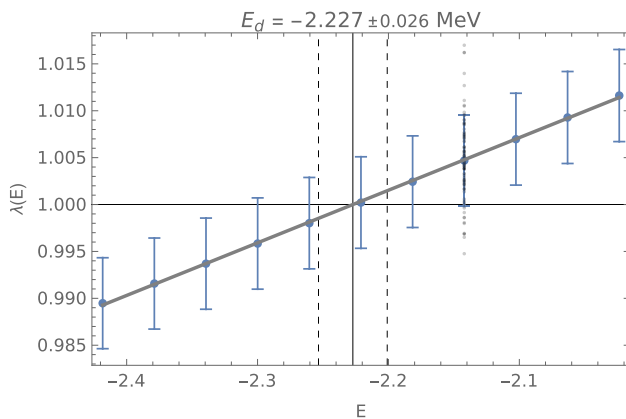


Fig. 3 Similar to Fig. 1, eigenvalues closest to 1 for different values of E in MeV from (2). The standard deviation of the scalar function values is taken to be $\sigma(\mathcal{V}_{ijkl}) = 0.4 |\mathcal{V}_{ijkl}|$, other values have no uncertainty

functions result in small deviations from the mean. It is interesting to investigate a situation where this approximation is no longer valid. Figures 3 and 4 contain results where the relative uncertainty α is increased. This results in an increased uncertainty of determining the deuteron binding energy. Additionally, the values of the standard deviations are in slightly worse agreement as can be seen in Fig. 4.

Figures 5 and 6 show results where we are not limiting the uncertainty to a single scalar function. The relative uncertainty $\alpha = 0.05$ is small compared to the previous examples and this results in a small uncertainty for the deuteron binding energy. The sampled eigenvalues are in good agreement

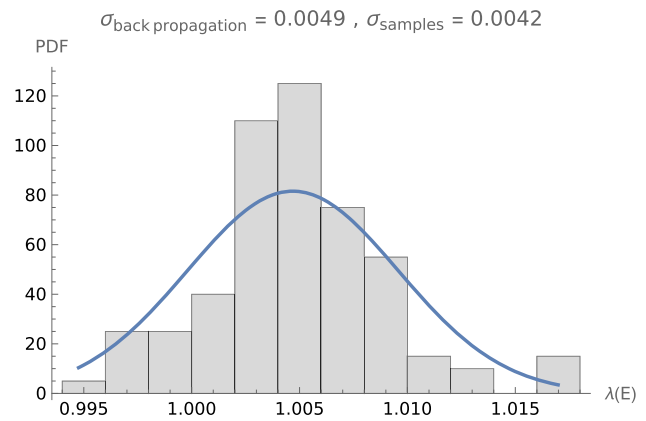


Fig. 4 Similar to Fig. 2, histogram of eigenvalues closest to 1 for single energy $E = -2.14206$ MeV from Fig. 3. The blue line is the normal distribution probability density function (PDF) with parameters calculated using (17). The data was tested for normality using the Kolmogorov-Smirnov test, the p value is 0.382

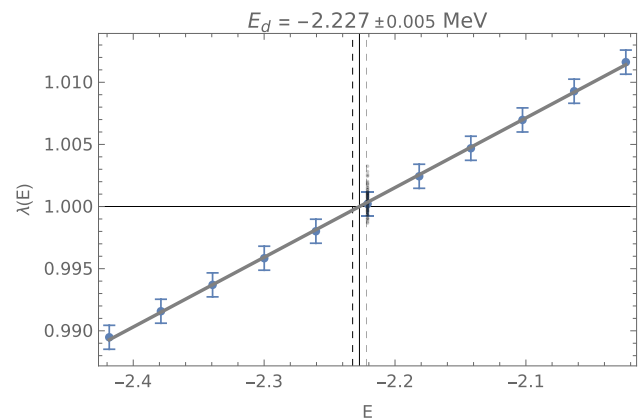


Fig. 5 Similar to Fig. 1, eigenvalues closest to 1 for different values of E in MeV from (2). The standard deviation of the scalar function values is taken to be $\sigma(\mathcal{V}_{ijkl}) = 0.05 |\mathcal{V}_{ijkl}|$, for all i, j, k , and l

with the normal distribution whose parameters were obtained using backpropagation as can be seen in Fig. 6.

Finally, Fig. 7 shows the gradient of the eigenvalue closest to 1 from (2) for selected scalar function values and error estimates for different values of the relative error are gathered in Table 1. The first column of this table contains the relative error α . It was assumed that this value is the same for all scalar functions. The second and third column contain the calculated deuteron binding energy and the error associated with this estimate. For all considered values of the relative error, the calculated deuteron binding energy is similar and the error increases in a roughly linear fashion with α .

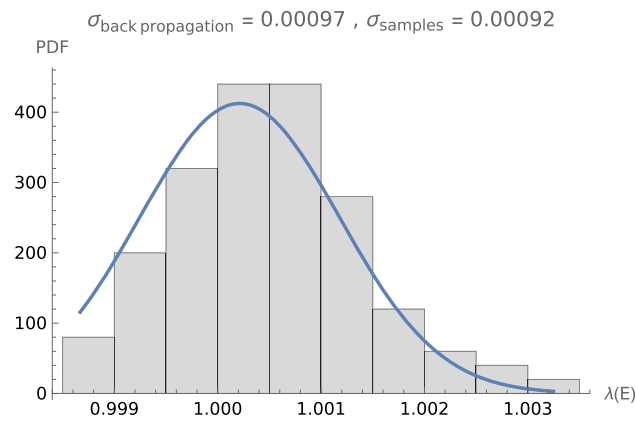


Fig. 6 Similar to Fig. 2, histogram of eigenvalues closest to 1 for single energy $E = -2.22099$ MeV from Fig. 5. The blue line is the normal distribution probability density function (PDF) with parameters calculated using (17). The data was tested for normality using the Kolmogorov–Smirnov test, the p value is 0.424

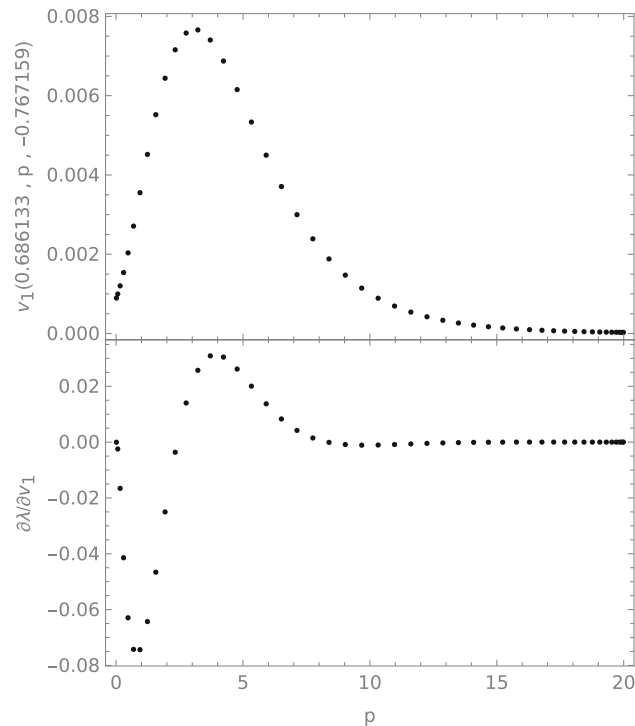


Fig. 7 Upper plot: selected values of $v_1(0.686133, p, -0.767159)$ from (4) are plotted for different values of p in fm^{-1} . Lower plot: the gradient of the eigenvalue λ from (2) that is closest to 1 ($\lambda = 1.000212$) for energy -2.22099 MeV with respect to the corresponding scalar function values from the upper plot

3 Nucleon–nucleon scattering

In order to calculate the nucleon–nucleon (NN) scattering observables we will solve the Lippmann–Schwinger equation:

$$t(z) = V + VG_0(z)t(z). \quad (18)$$

Table 1 The deuteron binding energy and its uncertainty for different values of the relative error α . The second column contains roughly the same value for all relative errors – the deuteron binding energy in MeV. The third column contains errors of the energy estimate in MeV obtained using backpropagation. The relationship between the deuteron binding energy error value and the relative error is roughly linear

$\alpha(i = 1 \dots 6)$	E_d	$\sigma(E_d)$
0.01	-2.2272	0.0010
0.03	-2.2272	0.0031
0.05	-2.2272	0.0052
0.07	-2.2272	0.0073
0.1	-2.2272	0.0104

to obtain the “ t matrix”, $t(z)$, for the complex argument

$$z = E + i\epsilon$$

where $E > 0$ is the energy of the 2N system. The infinitesimal ϵ is real and we take the limit $\epsilon \rightarrow 0$. In equation (18), the free propagator G_0 has the form:

$$G_0(z) \equiv (z - H_0)^{-1}$$

where $H_0 = \frac{\mathbf{p}^2}{m}$ is the free Hamiltonian operator with $\mathbf{p}$ being the relative NN momentum and m being the nucleon mass. With the assumption that the potential satisfies (3) Eq. (18) separates and can be solved independently for each isospin case $t^{tm_i}(z)$.

In order to arrive at a form of (18) that can be used in numerical calculations, the identity operator can be inserted between operators. For partial wave states (10) it is given by:

$$1 = \int_0^\infty dp p^2 \sum_\alpha |p\alpha\rangle \langle p\alpha| \quad (19)$$

with $|p\alpha\rangle$ being partial wave states (10) and α being the complete sets of discrete quantum numbers. In practice the sum over α is limited to a finite number of 2N states and the integral upper limit is replaced with a finite cut-off, turning (19) into an approximation. In the first step, using (4) and inserting (19) between the operators, the Lippmann–Schwinger Eq. (18) can be transformed into:

$$\begin{aligned} \langle p'\alpha'|\phi\rangle - \int_0^\infty dk k^2 \frac{1}{p_0^2 - k^2 + i\epsilon} \\ \sum_\beta m \langle p'\alpha'|V^{tm_i}|k\beta\rangle \langle k\beta|\phi\rangle = \langle p'\alpha'|v\rangle \end{aligned} \quad (20)$$

where

$$|\phi\rangle \equiv t^{tm_t} \left(E + i\epsilon = \frac{p_0^2}{m} + i\epsilon \right) |p\alpha\rangle, \quad (21)$$

and

$$|v\rangle \equiv V^{tm_t} |p\alpha\rangle. \quad (22)$$

Equation (20) can be solved separately for the states $|\phi\rangle$ with given isospin (t, m_t), energy ($E = \frac{p_0^2}{m}$), initial relative momentum (p) and discrete quantum numbers (α). The integral in (20) can be evaluated using the standard approach [30] by replacing infinity with a large finite momentum cut-off $\bar{p}$ and taking the limit $\epsilon \rightarrow 0$ to obtain:

$$\begin{aligned} \langle p'\alpha' | \phi \rangle &= \int_0^{\bar{p}} dk \frac{1}{p_0^2 - k^2} \\ &\times \left(\sum_{\beta} k^2 m \langle p'\alpha' | V^{tm_t} | k\beta \rangle \langle k\beta | \phi \rangle \right. \\ &\left. - \sum_{\beta} p_0^2 m \langle p'\alpha' | V^{tm_t} | p_0\beta \rangle \langle p_0\beta | \phi \rangle \right) - \frac{1}{2} p_0 \sum_{\beta} m \\ &\times \langle p'\alpha' | V^{tm_t} | p_0\beta \rangle \langle p_0\beta | \phi \rangle \left(\ln \frac{\bar{p} + p_0}{\bar{p} - p_0} - i\pi \right) \\ &= \langle p'\alpha' | v \rangle \end{aligned} \quad (23)$$

The implementation of the matrix form of the linear equation (23) for state $|\phi\rangle$ using the **pytorch** library is discussed in Sect. 3.1.

3.1 Implementation using **pytorch**

In order to turn (23) into a matrix equation that can be solved using the **pytorch** library it is useful to extend the set of discrete values of the relative 2N momentum (6):

$$p_1, p_2 \dots p_N, p_{N+1} \equiv p_0 \quad (24)$$

and to set the integration weight $\mathcal{W}_{N+1}^p \equiv 0$. With this extended set of momentum values and the $\mathcal{V}^{pwd}$ “tensor” from (12) containing the partial wave matrix elements of the 2N potential equation (23) becomes:

$$\sum_{k=1}^{N+1} \sum_{\beta} \mathcal{A}^{tm_t}(p_0)_{j\gamma k\beta} \phi_{k\beta} = v_{j\gamma}, \quad (25)$$

where the second sum is over a finite set of considered discrete quantum numbers β . The states (21) and (22) are represented by **pytorch** “tensors” $\phi_{k\beta} \equiv \langle p_k\beta | \phi \rangle$ and $v_{j\gamma} \equiv$

Table 2 1P_1 phase shifts calculated for nucleon–nucleon scattering with $p_0 = 2 \text{ fm}^{-1}$ for different values of the relative error α . The second column contains the same value for all relative errors - the calculated phase shift value in degrees. The third and fourth column contains errors estimated using backpropagation and direct sampling in degrees, respectively. The relationship between the phase shift error value and the relative error is roughly linear

$\alpha(i = 1 \dots 6)$	s	$\sigma^{\text{backpropagation}}(s)$	$\sigma^{\text{samples}}(s)$
0.01	−30.1	0.5	0.5
0.03	−30.1	1.4	1.4
0.05	−30.1	2.4	2.7
0.07	−30.1	3.3	3.2
0.1	−30.1	4.8	4.6

$\langle p_j\gamma | v \rangle$ respectively. The explicit form of the elements of $\mathcal{A}^{tm_t}$ is more complicated than (14) but it can be directly read off from (23). The practical implementation of $\mathcal{A}^{tm_t}(p_0)_{j\gamma k\beta}$ is made easier by utilizing the capabilities of **pytorch** that allow to automatically broadcast functions over “tensors” and by treating the case $k = N + 1$ separately.

In order to calculate observables p is set to p_0 in (21) and (22). Finally, all “tensors” are appropriately reshaped before using the **torch.linalg.solve** function to solve the linear equation for $|\phi\rangle$. Since all operations are performed within the **pytorch** library, gradients of the observables with respect to the scalar functions that define the interaction (4) are available allowing the estimation of errors. Similarly as in the deuteron case, we will assume that the uncertainty of the scalar functions that define the 2N interaction has the form (16) and assume that there are no correlations so that the errors can be calculated using a formula similar to (17) but with λ replaced by the relevant observable. In the next Sect. 3.2 we will present result for phase-shifts calculated for different partial wave channels and energies.

3.2 Results for nucleon–nucleon scattering

To demonstrate our method we computed phase shifts for two uncoupled channels 1S_0 and 1P_1 and for the two coupled cases: $^3S_1 - ^3D_1$ and $^3P_2 - ^3F_2$ using the Bonn-B [27] potential. Calculations at $p_0 = 2 \text{ fm}^{-1}$ and $\alpha(i = 1, \dots, 6) = 0.1$. for 1S_0 resulted in a value of −11.8988 degrees. The error estimated associated with this result was calculated using backpropagation and its value, 3.38991 degrees, is in agreement with the standard deviation calculated using direct sampling of scalar values that define the potential. The sampling procedure is the same as in the deuteron case. The calculation is repeated for each of the 100 samples that contain new scalar function values resulting in 100 phase shift values. The standard deviation of these values is 3.46838.

Results for the 1P_1 channel are gathered in Tables 2 and 3. Table 2 contains phase shifts calculated with $p_0 = 2 \text{ fm}^{-1}$

Table 3 1P_1 phase shifts calculated for nucleon–nucleon scattering with the relative error $\alpha(i = 1 \dots 6) = 0.05$ for different values of p_0 in fm^{-1} in the first column. The second column contains the calculated value in degrees. The third and fourth column contains errors estimated using backpropagation and direct sampling in degrees, respectively

p_0	s	$\sigma_{\text{backpropagation}}(s)$	$\sigma_{\text{samples}}(s)$
0.5	−6.3	3225.6	0.4
1.	−14.8	0.4	0.4
2.	−30.1	2.4	2.7
3.	−39.8	1.2	1.1
4.	−41.9	217.8	26.0
5.	−43.0	1.7	1.6
6.	−47.9	2.2	2.0
7.	−55.5	3.5	3.5

for different values of the relative error α . The relative error was assumed to be the same for all scalar functions that define the potential. The second and third column contain estimates of the error associated with the phase shift in column two. These columns were calculated using backpropagation and direct sampling, respectively. The error estimates are in agreement. Table 3 contains phase shifts calculated for different values of p_0 . The second column contains the calculated value in degrees. The third and fourth columns contain errors estimated using backpropagation and direct sampling in degrees. Note the large error and discrepancy between the third and fourth column for $p_0 = 0.5 \text{ fm}^{-1}$ and $p_0 = 4.0 \text{ fm}^{-1}$. This can be explained by the close proximity of these values to momentum values 0.4999 fm^{-1} and 4.0200 fm^{-1} from the Gaussian quadrature points used in calculating integrals and demonstrates the sensitivity of the approach to the details of the numerical implementation. The large uncertainty obtained using backpropagation can be used as a signal indicating the necessity to reconsider the parameters of the numerical calculation. This is especially visible for $p_0 = 4 \text{ fm}^{-1}$ where it is evident that the choice of integration points effects on the uncertainty calculated directly from sampling.

In addition to results for the two uncoupled channels that were calculated using only a single partial wave state, calculations for two coupled channels $^3S_1 - ^3D_1$ and $^3P_2 - ^3F_2$ were carried out at $p_0 = 2 \text{ fm}^{-1}$ and required using two partial waves. The results for $^3S_1 - ^3D_1$ are gathered in Table 4 and results for $^3P_2 - ^3F_2$ are gathered in Table 5. The error estimates obtained using backpropagation are in agreement with estimates obtained using direct sampling with 100 samples. Of course their absolute values depend on the arbitrarily assumed α thus they do not carry any information on quality of the specific potential.

Table 4 Phase shift and mixing parameters for the coupled $^3S_1 - ^3D_1$ channel. The calculations were performed with $p_0 = 2 \text{ fm}^{-1}$ and assumed the relative error $\alpha(i = 1 \dots 6) = 0.1$. The third and fourth column contains error estimates obtained using backpropagation and direct sampling using 100 samples, respectively

Quantity	Value	$\sigma_{\text{backpropagation}}(\text{quantity})$	$\sigma_{\text{samples}}(\text{quantity})$
δ_M	0.2	1.4	1.4
δ_S	−24.3	0.4	0.4
ϵ	−4.3	0.5	0.5

Table 5 Phase shift and mixing parameters for the coupled $^3P_2 - ^3F_2$ channel. The calculations were performed with $p_0 = 2 \text{ fm}^{-1}$ and assumed the relative error $\alpha(i = 1 \dots 6) = 0.05$. The third and fourth column contains error estimates obtained using backpropagation and direct sampling using 100 samples, respectively

Quantity	Value	$\sigma_{\text{backpropagation}}(\text{quantity})$	$\sigma_{\text{samples}}(\text{quantity})$
δ_M	17.23	0.49	0.50
δ_S	0.15	0.24	0.25
ϵ	1.72	0.07	0.07

4 Summary and outlook

In this paper we show that software libraries designed for machine learning can be used in practical few-nucleon quantum mechanical calculations. One benefit of using these libraries, is the possibility to use backpropagation to estimate how the errors propagate from the model parameters, in our case the values of the scalar functions that determine the two-nucleon potential, to observables. In our calculations we use the popular **Python pytorch** library [23]. It has a straightforward interface that allows to easily extend its use beyond machine learning applications and utilize the built in implementation of backpropagation.

To demonstrate this approach we associate an uncertainty with the scalar function values and investigate the effect of this uncertainty on the error of determining the deuteron binding energy and nucleon–nucleon scattering phase shifts. The assumed uncertainties were chosen in order to demonstrate the validity of the approach, for more practical results the numerical errors and truncation errors from higher orders of the chiral expansion could be considered. The actual parameter uncertainty values for the SMS chiral potential [7] are typically or very small ($\ll 1\%$) or up to 3%, and only in few cases are above 10%. For some of them the correlation is non-negligible.

Currently we assume that the errors of the scalar function values are uncorrelated. In future work it will be interesting to see the effect of correlated errors on the deuteron binding energy estimate or other observables. This would open up the possibility to use more realistic error estimates in the calculation. Furthermore, we plan to extend our calculations to the

three-nucleon system to investigate the effects of theoretical and numerical errors on observables obtained using the new generation of few-nucleon forces. We believe that because the method presented here is suitable for calculations based on solving an eigenequation or a system of linear equations, it can be also used in nuclear structure calculations. Even if using the **Python pytorch** library directly may be impractical in that case, the analogous code can be developed in standard programming languages.

Acknowledgements This work was supported by the National Science Centre, Poland under Grant IMPRESS-U 2024/06/Y/ST2/00135. It was also supported in part by the Excellence Initiative–Research University Program at the Jagiellonian University in Kraków. The numerical calculations were partly performed on the supercomputers of the JSC, Jülich, Germany.

Data Availability Statement This manuscript has associated data in a data repository. [Author's comment: The datasets generated during and/or analysed during the current study are available in the GitHub repository, https://github.com/kacpertosol/uncertainties_bkpropagation.]

Code Availability Statement This manuscript has associated code/software in a data repository. [Author's comment: The Code/software generated during and/or analysed during the current study is available in the GitHub repository, https://github.com/kacpertosol/uncertainties_bkpropagation.]

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. J. Dobaczewski, W. Nazarewicz, P.-G. Reinhard, Error estimates of theoretical models: a guide. *J. Phys. G Nucl. Part. Phys.* **41**, 074001 (2014)
2. R.N. Pérez, J.E. Amaro, E.R. Arriola, Coarse-grained potential analysis of neutron-proton and proton-proton scattering below the pion production threshold. *Phys. Rev. C* **88**, 064002 (2013)
3. R. Navarro Pérez, J.E. Amaro, E. Ruiz Arriola, Statistical error analysis for phenomenological nucleon-nucleon potentials. *Phys. Rev. C* **89**, 064006 (2014)
4. E. Epelbaum, H.-W. Hammer, U.-G. Meißner, Modern theory of nuclear forces. *Rev. Mod. Phys.* **81**, 1773–1825 (2009)
5. E. Epelbaum, U.-G. Meißner, Chiral dynamics of few- and many-nucleon systems. *Annu. Rev. Nucl. Part. Sci.* **62**(1), 159–185 (2012)
6. H.-W. Hammer, S. König, U. van Kolck, Nuclear effective field theory: status and perspectives. *Rev. Mod. Phys.* **92**, 025004 (2020)
7. P. Reinert, H. Krebs, E. Epelbaum, H. Krebs, Semilocal momentum-space regularized chiral two-nucleon potentials up to fifth order. *Eur. Phys. J. A* **54**, 86 (2018)
8. R. Skibiński, Y. Volkotrub, J. Golak, K. Topolnicki, H. Witała, Theoretical uncertainties of the elastic nucleon-deuteron scattering observables. *Phys. Rev. C* **98**, 014001 (2018)
9. Y. Volkotrub, J. Golak, R. Skibiński, K. Topolnicki, H. Witała, E. Epelbaum, H. Krebs, P. Reinert, Uncertainty of three-nucleon continuum observables arising from uncertainties of two-nucleon potential parameters. *J. Phys. G Nucl. Part. Phys.* **47**, 104001 (2020)
10. S. Miller, A. Ekström, C. Forssén, Posterior predictive distributions of neutron-deuteron cross sections. *Phys. Rev. C* **107**, 014002 (2023)
11. H. Witała, J. Golak, R. Skibiński, Significance of chiral three-nucleon force contact terms for understanding of elastic nucleon-deuteron scattering. *Phys. Rev. C* **105**, 054004 (2022)
12. T. Duguet, A. Ekström, R.J. Furnstahl, S. König, D. Lee, Colloquium: Eigenvector continuation and projection-based emulators. *Rev. Mod. Phys.* **96**, 031002 (2024)
13. X. Zhang, A non-hermitian quantum mechanics approach for extracting and emulating continuum physics based on bound-state-like calculations: technical details (2024). [arXiv:2411.06712](https://arxiv.org/abs/2411.06712)
14. V. Urbanevych, R. Skibiński, H. Witała, J. Golak, K. Topolnicki, A. Grassi, E. Epelbaum, H. Krebs, Application of a momentum-space semi-locally regularized chiral potential to selected disintegration processes. *Phys. Rev. C* **103**, 024003 (2021)
15. E. Epelbaum, H. Krebs, U.-G. Meißner, Improved chiral nucleon-nucleon potential up to next-to-next-to-next-to-leading order. *Eur. Phys. J. A* **51**, 53 (2015)
16. E. Epelbaum, H. Krebs, U.-G. Meißner, Precision nucleon-nucleon potential at fifth order in the chiral expansion. *Phys. Rev. Lett.* **115**, 122301 (2015)
17. E. Epelbaum, J. Golak, K. Hebeler, H. Kamada, H. Krebs, U.-G. Meißner, A. Nogga, P. Reinert, R. Skibiński, K. Topolnicki, Yu. Volkotrub, H. Witała, Towards high-order calculations of three-nucleon scattering in chiral effective field theory. *Eur. Phys. J. A* **56**, 92 (2020)
18. R.J. Furnstahl, N. Klco, D.R. Phillips, S. Wesolowski, Quantifying truncation errors in effective field theory. *Phys. Rev. C* **92**(2), 024005 (2015)
19. J.A. Melendez, R.J. Furnstahl, D.R. Phillips, M.T. Prato, S. Wesolowski, Quantifying correlated truncation errors in effective field theory. *Phys. Rev. C* **100**(4), 044001 (2019)
20. P. Maris, R. Roth, E. Epelbaum, R.J. Furnstahl, J. Golak, K. Hebeler, T. Hüther, H. Kamada, H. Krebs, H. Le, U.-G. Meißner, J.A. Melendez, A. Nogga, P. Reinert, R. Skibiński, J.P. Vary, H. Witała, T. Wolfgruber, Nuclear properties with semilocal momentum-space regularized chiral interactions beyond $n^2\text{LO}$. *Phys. Rev. C* **106**, 064002 (2022)
21. P.J. Millican, R.J. Furnstahl, J.A. Melendez, D.R. Phillips, M.T. Prato, Assessing correlated truncation errors in modern nucleon-nucleon potentials. *Phys. Rev. C* **110**, 044002 (2024)
22. K.S. Becker, K.D. Launey, A. Ekström, T. Dytrych, D. Langr, G.H. Sargsyan, J.P. Draayer, Uncertainty quantification of collective nuclear observables from the chiral potential parametrization (2024). [arXiv:2404.00063](https://arxiv.org/abs/2404.00063) [nucl-th]
23. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimeshain, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: an imperative style, high-performance deep learning library, in *Advances in Neural Information Processing Systems*, vol. 32 (Curran Associates, Inc., 2019), pp. 8024–8035
24. A.G. Baydin, B.A. Pearlmutter, A.A. Radul, J.M. Siskind, Automatic differentiation in machine learning: a survey. *J. Mach. Learn. Res.* **18**(153), 1–43 (2018)
25. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser,

- M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org (2015)
26. C.R. Harris, K.J. Millman, S.J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N.J. Smith, R. Kern, M. Picus, S. Hoyer, M.H. van Kerkwijk, M. Brett, A. Haldane, J.F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, T.E. Oliphant, Array programming with NumPy. *Nature* **585**, 357–362 (2020)
27. R. Machleidt, *The Meson Theory of Nuclear Forces and Nuclear Structure* (Springer US, Boston, 1989), pp.189–376
28. B.D. Carlsson, A. Ekström, C. Forssén, D.F. Strömberg, G.R. Jansen, O. Lilja, M. Lindby, B.A. Mattsson, K.A. Wendt, Uncertainty analysis and order-by-order optimization of chiral nuclear interactions. *Phys. Rev. X* **6**, 011019 (2016)
29. J. Angeles, *The Application of Dual Algebra to Kinematic Analysis* (Springer, Berlin, 1998), pp.3–32
30. J. Golak, D. Rozpędzik, R. Skibiński, K. Topolnicki, H. Witała, W. Glöckle, A. Nogga, E. Epelbaum, H. Kamada, C. Elster, I. Fachrudin, A new way to perform partial-wave decompositions of few-nucleon forces. *Eur. Phys. J. A* **43**, 241–250 (2009)
31. L. Wolfenstein, Possible triple-scattering experiments. *Phys. Rev.* **96**, 1654–1658 (1954)
32. W.R. Inc., Mathematica, Version 14.1, Champaign (2024)