Extrapolation to infinite model space of no-core shell model results using machine learning

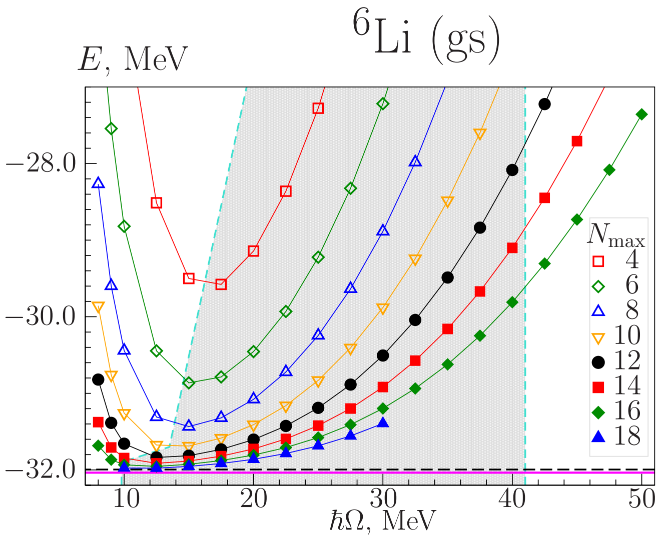
Khabarovsk

A. Mazur, **R. Sharypov** *Pacific National University, Khabarovsk, Russia*
A. Shirokov *SINP Moscow State University, Moscow, Russia*
I. J. Shin *Institute for Basic Science, Daejeon, Korea*
H. Li *Institute of Modern Physics CAS, Lanzhou, China*
P. Yin *Henan University of Science and Technology, Luoyang, China*
J. P. Vary *Iowa State University, Ames, IA, USA*

Nucleus-2025
Saint-Petersburg University
July 1–6, 2025

Problem

^{6}Li, NCSM with Daejeon16



The number of basis functions grows exponentially with N_{max}

^{14}F, N_{max}=8: 1 990 061 078

Now machine learning methods have been widely used in nuclear physics.

In my talk, I will consider only one problem: the extrapolating results of NCSM (variational) calculations to large model spaces.

As input, we have a set of NCSM energies (or other observable) in different model spaces N_{max} as a functions of the $\hbar\Omega$.

We need to extrapolate this data to the large N^f_{max} (we use $N^f_{max} = 300$) in order to estimate the binding energy (rms radii, etc.).

Outlook

Artificial Neural Networks (ANNs)

- Topology
- Training

Extrapolation algorithm

- Selection of input data
- Array of predictions
- Selection of ANNs

Applications:

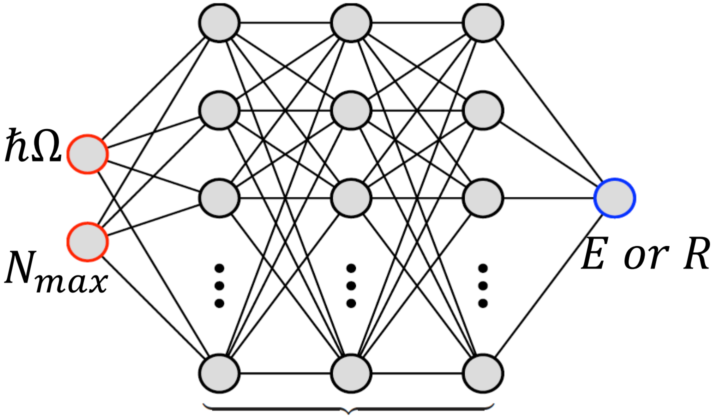
- ^{6}Li: ground and excited (3^+0), (0^+1) states (energy, rms-radii)
- ^{6}He, ^{6}Be: ground states (energy, rms-radii)
- ^{10}Be, ^{10}C: quadrupole moments and probabilities of $E2$ transitions

Artificial Neural Networks

We modified approach

G. A. Negoita et al. PRC 99, 054308 (2019).

More complicated topology:
3 hidden layers with 10 neurons in each



1. We do not have any problems with ANNs **overfitting.** We use **all the data to train the networks.**
2. More stable predictions

Input layer: $N_{\max}$, $\hbar\Omega$
Output layer: Energy, rms-radii, ets.
The **neurons** collects all input signals x_q^j and calculates a net signal x_p^i as the weighted sum of them:

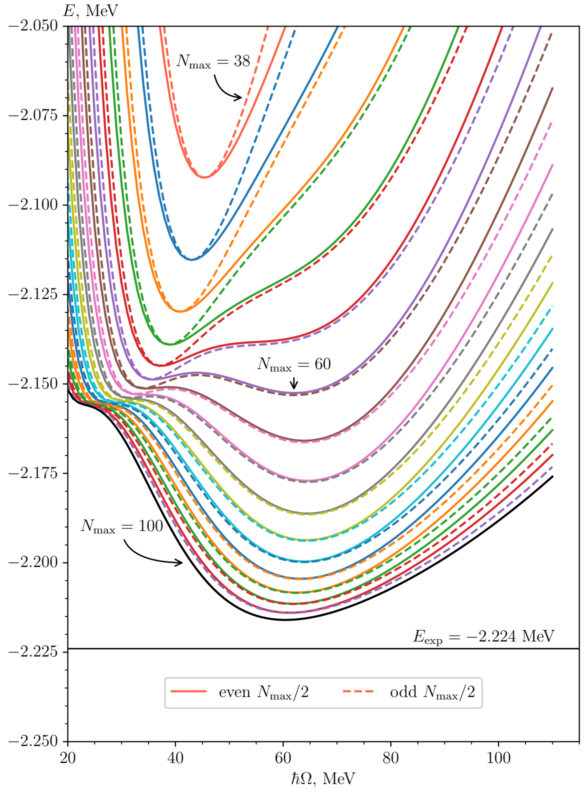
$$x_p^i = \sum_j \omega_p^{ij} \cdot x_{p-1}^j + b_p^i$$

Then results transforms by activating function:
linear $\quad f_p(x) = x$
or sigmoid $\quad f_p(x) = 1/[1 + \exp(-x)])$
and transmit to next layer.

Weights ω_p^{ij} and **biases b_p^i** are **trainable parameters (261)**

ANN: training

deuteron, Nijmegen II



- Training a neural network consists of searching such weights ω_p^{ij} and biases b_p^i that an acceptable level of error between the desired and predicted results is achieved. The difference between desired and predicted outputs is measured by the loss function L.
  We use mean-square error (MSE).

- After each pass of signals through the ANN (epoch) and obtaining predictions, the **L** is calculated.
  Then using the **Adam** algorithm, the weights and biases are changed to minimize the **L**.

- The training process is controlled by **hyperparameters**, a set of which we identified and tested in the extrapolation of variational calculations with the realistic Nijmegen II NN potential of the deuteron ground state energy.

Extrapolation algorithm

- We use an **ensemble** of ANNs with **identical hyperparameters**, but **different initial values of the weight** coefficients. We train **1024 neural networks**.

- Before the training, the weights ω_p^{ij} are randomly initialized in the interval

$$\left[-\left(\frac{6}{n_i + n_{i-1}}\right)^{\frac{1}{2}}, \left(\frac{6}{n_i + n_{i-1}}\right)^{\frac{1}{2}}\right],$$

n_i, n_{i-1} are the numbers of neurons in neighboring layers.

Initial biases b_p^i are set to zero.

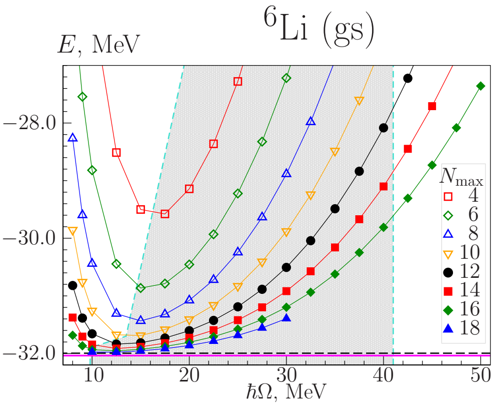
Extraplation:

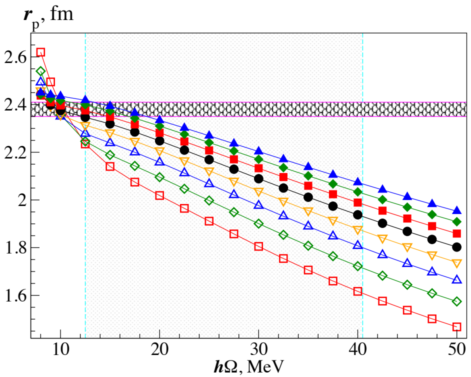
The **preliminary selection of input data** plays an **important** role.

NCSM results

Energy

rms-radii





All selected input data scale on interval [0,1]

selection of input data

We use:

for energy: from minimum $E_{gs}(\hbar\Omega)$ up to 40 MeV

for rms-radii: from 12.5 up to 40 MeV

ISU /PRC 99(2019)/

ISU-a: all data

ISU-b: from 10 up to 30 MeV

TUDa /PLB 839(2023)/

from 10 up to 20 MeV

array of predictions

Obtaining an array of predictions by generation results for the model spaces $N^u_{max}+2$, $N^u_{max}+4$, … N^f_{max} (N^f_{max}=300) for all values of $\hbar\Omega$

Extrapolation: selection of trained ANNs

Not all trained neural networks will provide reliable predictions.
So, it is important to have criteria to select obviously incorrectly trained networks.

Criteria what we use :

- **"Soft" variational principle** (*for energy*).
  Predictions can a little violate variational principle.
  Total violation at N^f_{max}=300 not exceed~ **5 keV**.
- **No dependence on $\hbar\Omega$** of predictions at large N^f_{max}=300.
- **"Convergence" principle** (*for energy*)
  For each NN we define such N^c_{max} ($N^u_{max} < N^c_{max} < N^f_{max}$) starting from which the following conditions are satisfied:
  1) for fixed model space the difference between the maximum and minimum of $E(\hbar\Omega)$ not exceed **0.02 MeV;**
  2) the difference between the minimum energy values in neighboring spaces N^c_{max} , $N^c_{max}+2$ not exceed **0.001 MeV**.

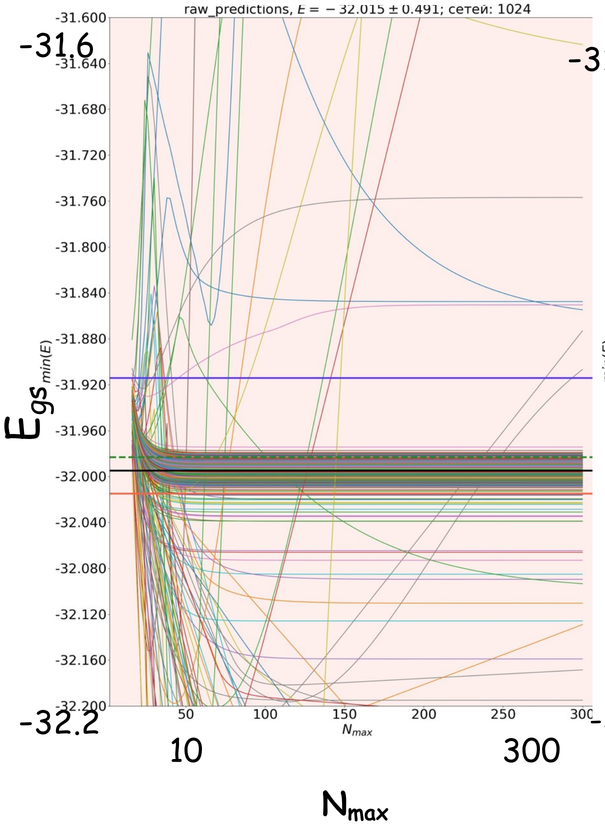
  Then from the obtained distribution ANNs we select 80 % networks with minimal values of N^c_{max} .
- **Additionally** we exclude 5% of networks with the highest loss function values to eliminate obvious outliers in training quality.

Extrapolation: selection of trained ANNs

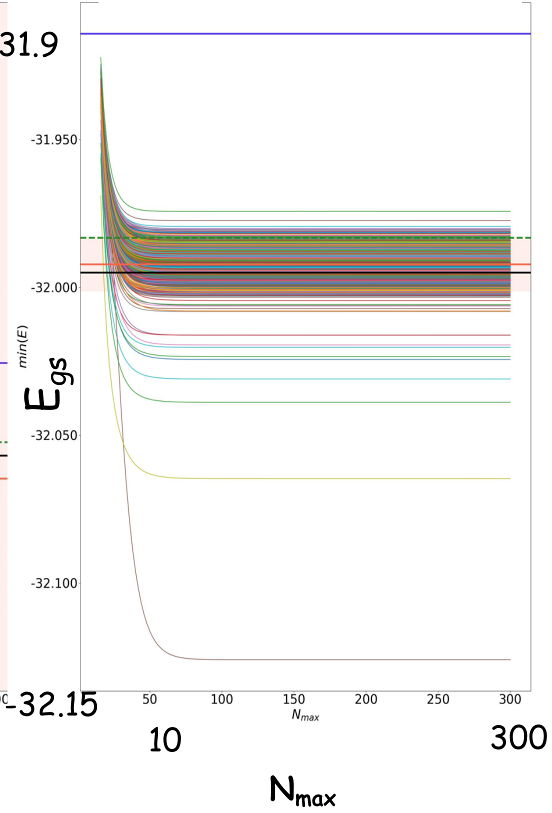
As a **result of selection** from the initial ensemble with **1024** ANNs remains **500-800** networks, what provides a fairly **high statistical significance** of our predictions.

For comparison:
in **PRC99(2019)** from **~400 000** trained ANNs selected only **50** ones.

Initial (1024)



After selection (554)
Egs=-31.992(9)



Extrapolation:

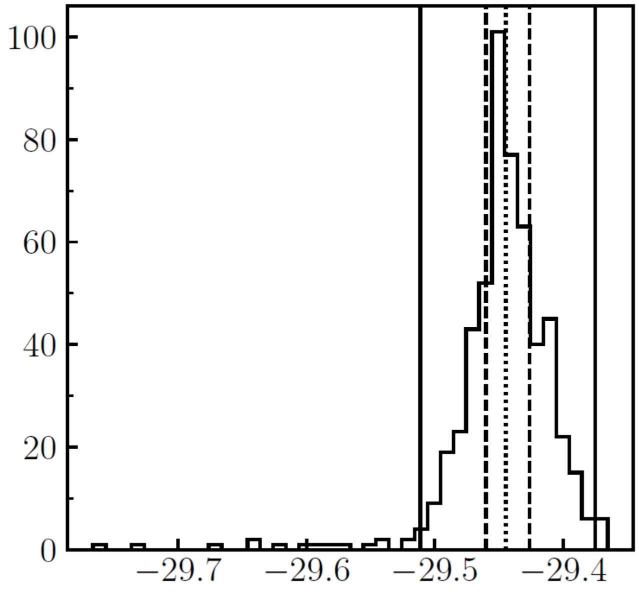
Statistical postprocessing of results

To correctly take into account the asymmetry of the obtained distribution, we divide it into 4 areas, each of which contains 25% selected ANNs (quartiles).

The position of the median (dotted line) is taken as the energy prediction $\bar{E}$, the difference between the median and quartiles (dashed lines) is taken as the corresponding errors $\Delta E_1, \Delta E_2$ (range of results $\Delta E = \Delta E_1 + \Delta E_2$).

We eliminate networks with results out the interval $[\bar{E} - \Delta E_2 - 1.5\Delta E, \bar{E} + \Delta E_1 + 1.5\Delta E]$ and repeat the procedure again.

The updated values $\bar{E}, \Delta E_1, \Delta E_2$ are the final predictions.



Results: ^{6}Li(gs 1+0). NCSM, Daejeon16

The ^{6}Li nucleus is of particular interest to us, since this nucleus investigated in other approaches with using ANN's: ISU, TUDa.

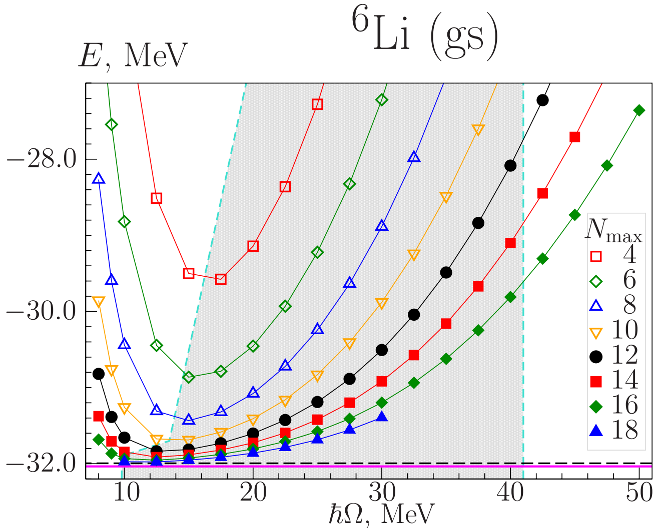
Moreover, all approaches use training sets based on the results of calculations in the NCSM with NN interaction Daejeon16 in model spaces up to $N^u_{max} = 18$

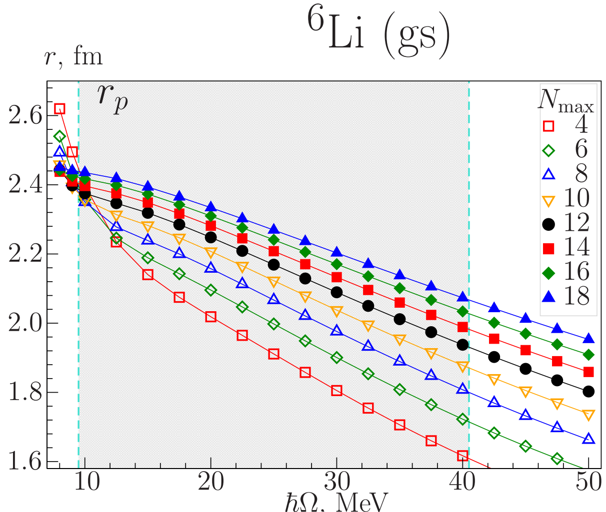
But the topology of the neural network, the selection of input data and some individual aspects of this approaches are different.

ISU-a: ħΩ [8, 50] MeV
ISU-b: ħΩ [10, 30] MeV

TUDa: ħΩ [10, 20] MeV

Extrap. B: $N_{max} = 18$
TTE: $N_{max} = 14$





Results: ^{6}Li(gs, 1+0). ANN

Convergense

Energies (N^u_{max} = 18)

Our results	Experiment [4]	Other extrapolations	
^{6}**Li** (g. s.)			
−32.036(3)	−31.995	−32.007(9)	Extrap. B [3]
		−32.061(4)	ISU-a [1]
		$-32.011^{+0.006}_{-0.014}$	TUDa [2]

rms-radii (N^u_{max} = 18)

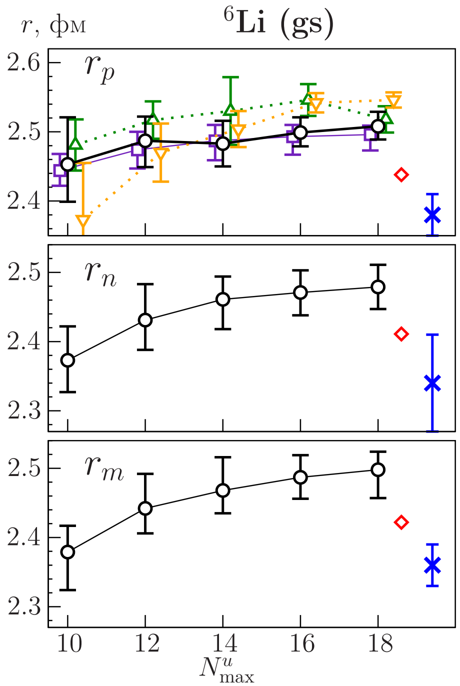
	Our results	Experim. [5]	Other extrapolations	
^{6}**Li** (g. s.)				
r_p	2.51(2)	2.38(3)	2.411	TTE [7]
			2.518(19)	ISU-a
			$2.496^{+0.013}_{-0.023}$	TUDa
r_n	2.48(3)	2.34(7)	2.438	TTE [7]
r_m	$2.50^{+0.03}_{-0.04}$	2.36(3)	2.422	TTE [7]

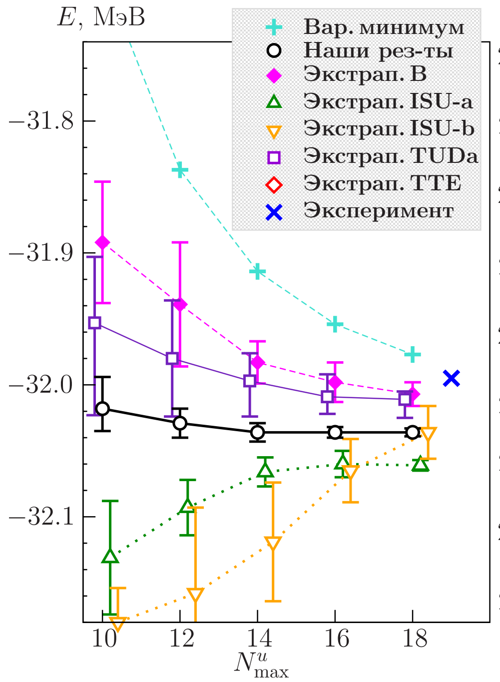
ISU-a: ħΩ [8, 50] МэВ

ISU-b: ħΩ [10, 30] МэВ

TUDa: ħΩ [10, 20] МэВ

Extrap. B: N_{max} = 18



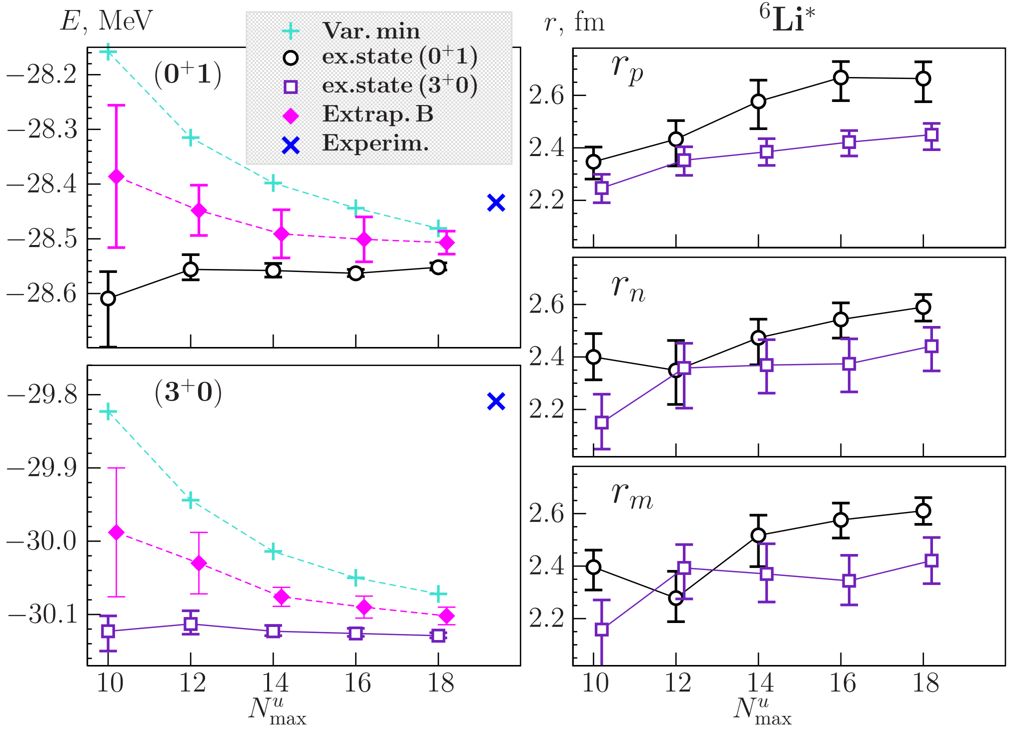


TTE: N_{max} = 14

Results: ANN

^{6}Li* (3+0) E_{exp} = -29.805 MeV

^{6}Li* (0+1) E_{exp} = -28.432 MeV



Energies (N^u_{max} = 18)

Our results	Experiment [4]	Other extrapolations	
^{6}Li $(3^+,0)$			
$-30.129^{+0.004}_{-0.003}$	-29.809	$-30.10(1)$	Extrap. B [3]
^{6}Li $(0^+,1)$			
$-28.552^{+0.008}_{-0.005}$	-28.434	$-28.507(4)$	Extrap. B [3]

^{6}Li* (3+0) → (^{4}He+d), Γ = 20 keV

^{6}Li* (0+1):

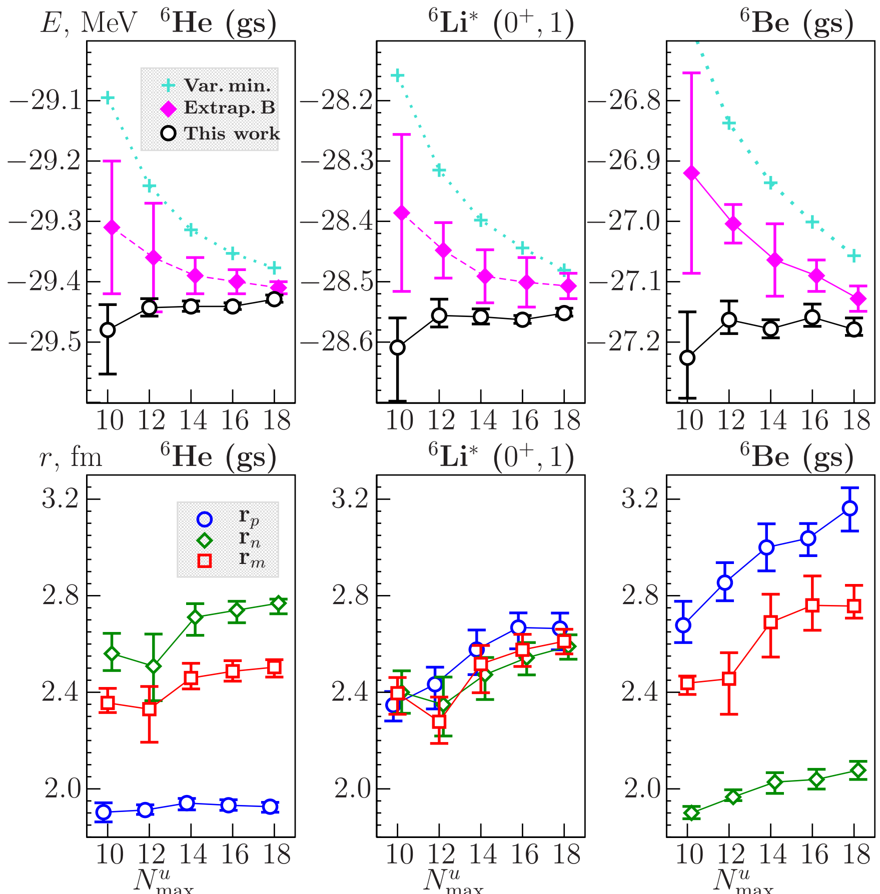
decay (^{4}He+d) forbidden! Γ = 8.2 eV

(isospin + spin-parity)

The energies show good convergence, while the radii tend to increase with increasing N^u_{max}

Results: ANN

^{6}He(gs 0+1) ^{6}Li*(0+1) ^{6}Be (gs 0+1)



Energies (N^u_{max} = 18)

Our results	Experiment [4]	Other extrapolations	
^{6}**He** (g. s.)			
$-29.429^{+0.007}_{-0.005}$	−29.269	−29.41(1)	Extrap. B [3]
^{6}**Li** (0$^+$,1)			
$-28.552^{+0.008}_{-0.005}$	−28.434	−28.507(4)	Extrap. B [3]
^{6}**Be** (g. s.)			
$-27.18^{+0.02}_{-0.01}$	−26.924	-27.13(2)	Extrap. B [3]

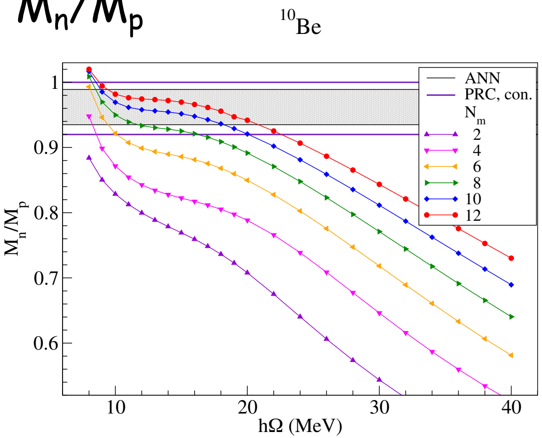
rms-radii (N^u_{max} = 18)

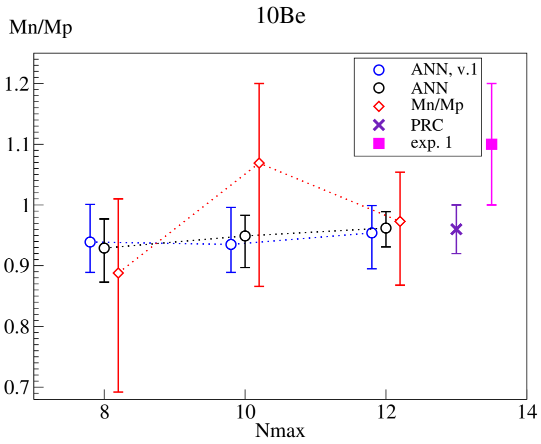
	Our results	Experim. [5]	Other extrapolations
^{6}**He** (g. s.)			
r_p	1.93(2)	1.925(12)	1.871(6) TTE [6] (^{6}He)
r_n	$2.77^{+0.02}_{-0.04}$	2.74(7)	2.663(3) TTE [6] (^{6}He)
r_m	$2.50^{+0.03}_{-0.04}$	2.50(5)	2.430(6) TTE [6] (^{6}He)

agreement with experiment

Results: ^{10}Be ($2^+ \rightarrow 0^+$)

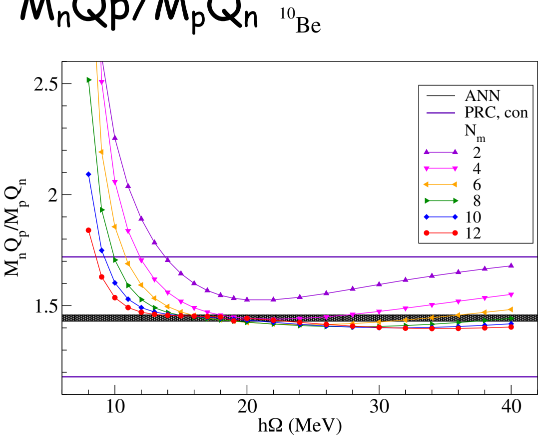
M_n/M_p

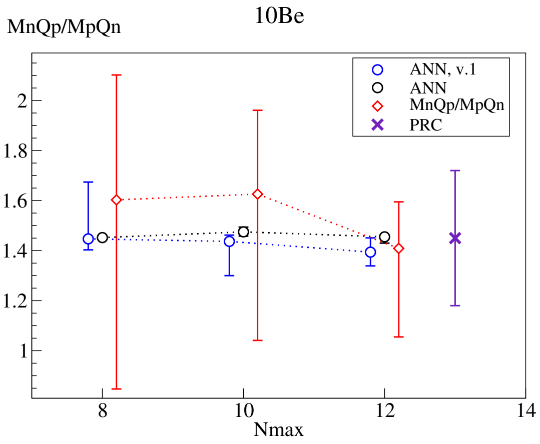




^{10}Be	ANN, our results	PRC 110 (2024)
M_n/M_p	$0.96^{+0.03}_{-0.03}$	0.96(4)
M_nQ_p/M_pQ_n	$1.455^{+0.003}_{-0.025}$	1.45(27)
$B(E2)$	$9.63^{+0.33}_{-0.64}$	9.48(2)
$B(E2)/Q_p^2$	$0.274^{+0.009}_{-0.013}$	0.29(1)

M_nQp/M_pQ_n



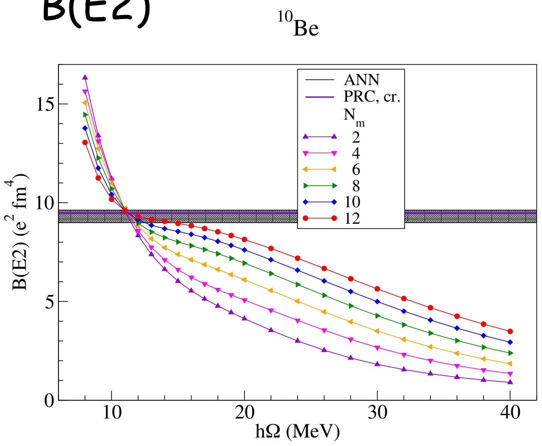


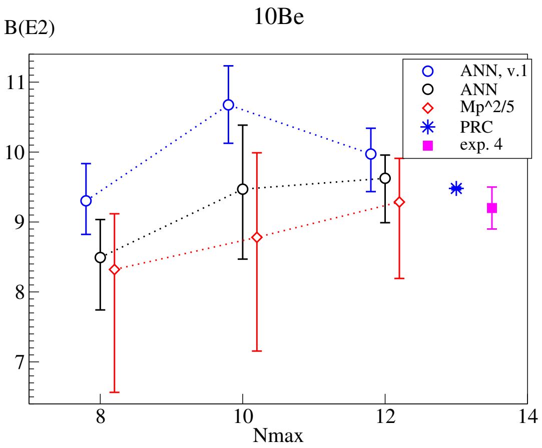
blue – all data

black – [12.5,40] MeV

Results: ^{10}Be ($2^+ \rightarrow 0^+$)

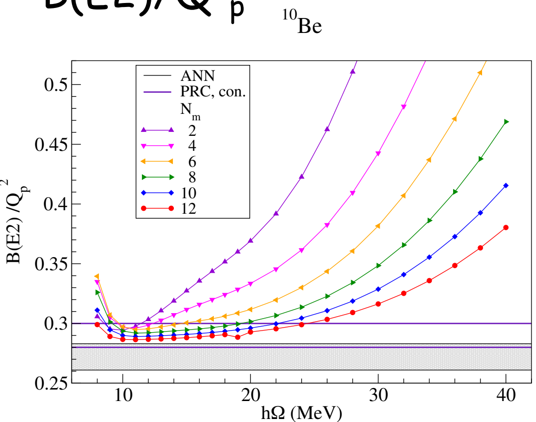
B(E2)

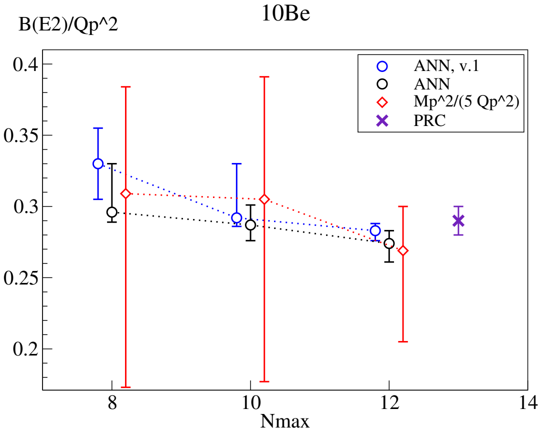




^{10}Be	ANN, our results	PRC 110 (2024)
M_n/M_p	$0.96^{+0.03}_{-0.03}$	0.96(4)
M_nQ_p/M_pQ_n	$1.455^{+0.003}_{-0.025}$	1.45(27)
$B(E2)$	$9.63^{+0.33}_{-0.64}$	9.48(2)
$B(E2)/Q_p^2$	$0.274^{+0.009}_{-0.013}$	0.29(1)

$B(E2)/Q^2_p$



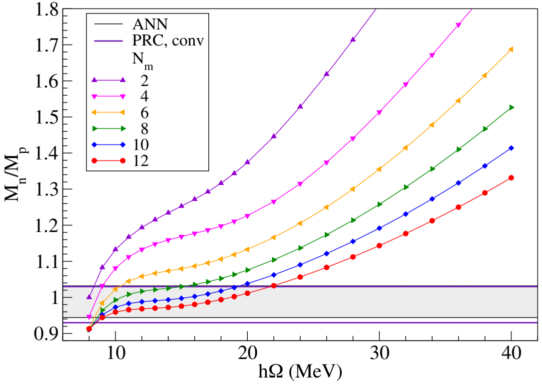


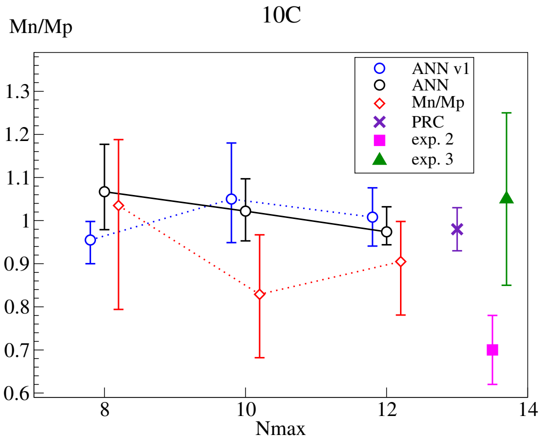
blue – all data

black – [12.5,40] MeV

Results: ^{10}C ($2^+ \rightarrow 0^+$)

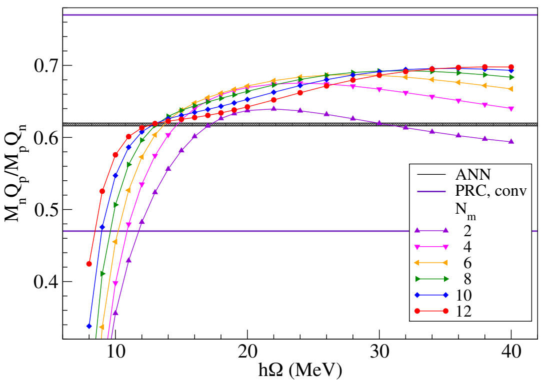
M_n/M_p

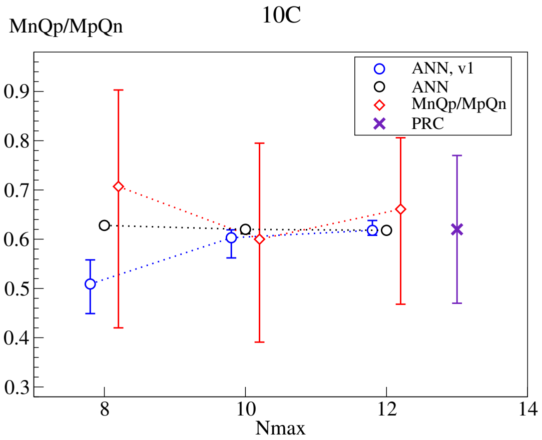




^{10}C	ANN, our results	PRC 110 (2024)
M_n/M_p	$0.97^{+0.06}_{-0.03}$	0.98(4)
M_nQ_p/M_pQ_n	$0.618^{+0.002}_{-0.002}$	0.62(15)
$B(E2)$	$11.91^{+0.77}_{-0.56}$	12.7(8)
$B(E2)/Q_p^2$	$0.73^{+0.01}_{-0.09}$	0.76(6)

M_nQp/M_pQ_n



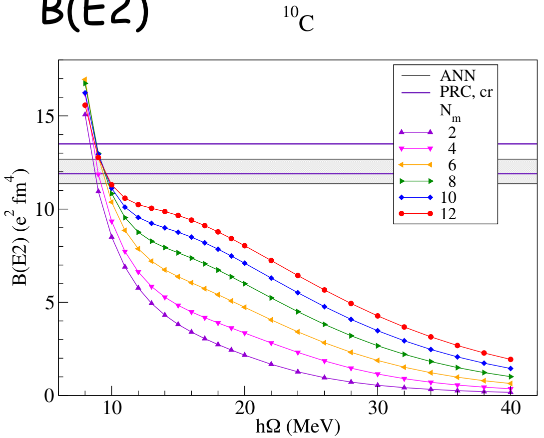


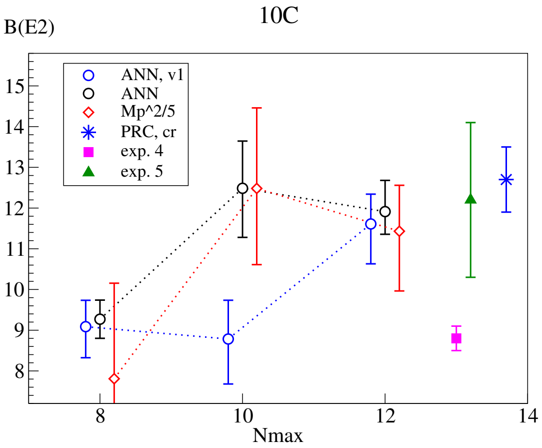
blue – all data

black – [12.5,40] MeV

Results: ^{10}C ($2^+ \rightarrow 0^+$)

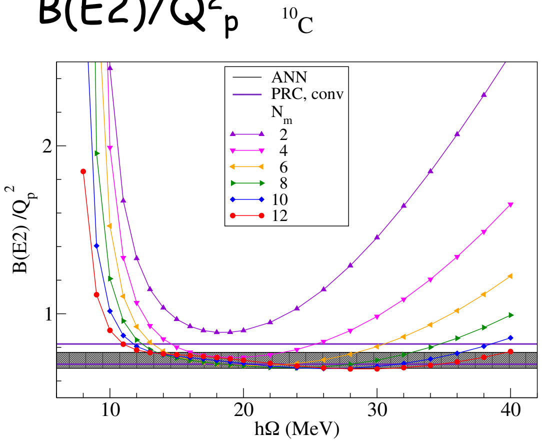
B(E2)

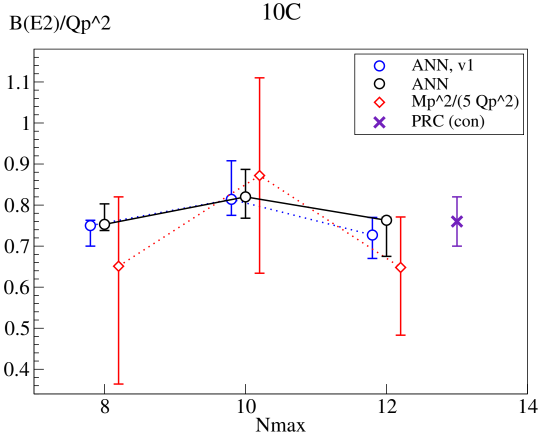




^{10}C	ANN, our results	PRC 110 (2024)
M_n/M_p	$0.97^{+0.06}_{-0.03}$	0.98(4)
M_nQ_p/M_pQ_n	$0.618^{+0.002}_{-0.002}$	0.62(15)
$B(E2)$	$11.91^{+0.77}_{-0.56}$	12.7(8)
$B(E2)/Q_p^2$	$0.73^{+0.01}_{-0.09}$	0.76(6)

B(E2)/Q^2_p





blue – all data

black – [12.5,40] MeV

Conclusion

A method for extrapolating variational calculations to large model spaces based on machine learning of an ensemble of neural networks is discussed.

We used a new neural network topology, different from the paper **Negoita e.a. PRC 99 (2019)**, and formulated criteria for selecting trained networks.

Predictions of the ground state energies of ^{6}Li, ^{6}He, ^{6}Be nuclei and excited states 3^+0 and 0^+1 of ^{6}Li nucleus show good convergence with high statistical significance.
The prediction errors are comparable to the accuracy of the NCSM calculations (1 keV) and do not exceed the errors of extrapolations **PRC 99 (2019)**.

The error in predicting the root-mean-square point radii r_{p}, r_{m}, r_{n} is somewhat worse, about one percent, but, in general, the convergence of the results is also quite good.

The suggested method is universal and can be applied to other observables, such as the quadrupole moment, the probabilities of electromagnetic transitions, and so on.

THANK YOU

for YOUR ATTENTION