Physics – Uspekhi **66** (11) 1164 – 1173 (2023)

METHODOLOGICAL NOTES PACS numbers: 07.05.Mh, **75.10. – b, 75.30. – m**, 75.40.Cx

Machine learning for the search for topological spin textures

G V Paradezhenko, A A Pervishko, D I Yudin

DOI: https://doi.org/10.3367/UFNe.2022.12.039303

Contents

1. Introduction **1164**
2. Overview of machine learning **1165**
3. Neural network algorithm **1166**
3.1 Localized spins; 3.2 Itinerant magnet
4. Two-dimensional magnetic systems **1168**
4.1 Skyrmions; 4.2 Biquadratic interaction
5. Three-dimensional magnetic systems **1169**
5.1 Skyrmion tubes; 5.2 Magnetic hedgehog lattice
6. Conclusions **1170**
References **1172**

Abstract. We present an alternative method for numerical modeling of topological magnetic textures using a neural network algorithm. We discuss a model of localized spins where topological magnetic textures are stabilized due to a delicate interplay between the symmetric exchange interaction, and the antisymmetric interaction caused by exchange–relativistic effects, as well as a model of an itinerant magnet where noncoplanar spin configurations emerge when taking multispin interactions into account. The viability of the proposed method is illustrated with the formation of lattices of skyrmions and antiskyrmions, magnetic hedgehogs, and skyrmion tubes in two-dimensional and three-dimensional magnetic systems.

Keywords: magnets, machine learning, spin–orbit coupling, Dzyaloshinskii–Moriya interaction, multispin interaction, skyrmions, antiskyrmions, magnetic hedgehogs

1. Introduction

The magnetic order of 3d–4f metals and their compounds depends on the electron subsystem state, can be captured by either the model of localized magnetic moments or the sd-exchange model [1]. In the first case, the subject of study is magnetic nonmetals, all of whose electrons are localized on atoms, with overlapping electron shells of neighboring atoms. The magnetic moments interact via direct exchange, and the magnetic properties of a material can be described in the framework of the classical Heisenberg model [2, 3]. In magnetic metals and alloys, besides localized electrons, there are electrons that from an itinerant subsystem. The exchange interaction between localized and itinerant electrons is assumed in the sd-exchange model [4–6].

When the radius of the incomplete electron shell is less than the crystal lattice parameter, the wave functions of neighboring sites have practically no overlap, which makes the exchange integral negligible. The exchange interaction that stabilizes the magnetic order in the system, the Ruderman–Kittel–Kasuya–Yoshida (RKKY) interaction, is then indirect, and is mediated via itinerant electrons [7–9]. We note that a similar situation arises in highly dilute magnetic alloys where the distance between spatially separated d or f ions is sufficiently large [10–12].

In cubic crystals, taking only the direct exchange interaction between localized spins into account stabilizes the collinear magnetic ordering. But the situation changes if the crystal structure does not have an inversion center; in such systems, an antisymmetric Dzyaloshinskii–Moriya interaction (DMI) arises [13, 14]. The existence of spatially modulated magnetic configurations with an incommensurate period and a fixed sense of rotation directly follows from the competition between the spin-bilinear exchange interaction and the DMI. We also note that surface DMI can occur at the interfaces of layered magnetic structures, which typically contain layers of a ferromagnetic material and a metal with strong spin-orbit coupling; such interfaces lead to mirror symmetry breaking [15–21].

In two-dimensional magnetic materials placed in a uniform external magnetic field, the spin helix becomes unstable. A lattice of skyrmions — localized vortex-like spin states — then forms [22, 23]. The existence of such magnetic textures and their nucleation mechanism can be explained using continuum models [24–28], including Lifshitz invariants corresponding to the terms in free energy that are linear in the magnetization vector gradients [29, 30]. Theoretical studies of two-dimensional classical spin models, including

G V Paradezhenko [1, a], **A A Pervishko** [1, 2, b], **D I Yudin** [1, 2, c]
[1] Skolkovo Institute of Science and Technology,
Bol'shoi bul'var 30, 121205 Moscow, Russian Federation
[2] Far Eastern Federal University,
Russkii Island, Ajax Bay 10, 690922 Vladivostok, Russian Federation
E-mail: [a] g.paradezhenko@skoltech.ru, [b] a.pervishko@skoltech.ru,
[c] d.yudin@skoltech.ru

Received 19 October 2022, revised 24 November 2022
Uspekhi Fizicheskikh Nauk **193** (11) 1237 – 1247 (2023)
Translated by S Alekseev

Monte Carlo simulations, clearly indicate the formation of a skyrmion crystal in chiral magnets [31–39].

Experimental observations of a skyrmion lattice in three-dimensional MnSi [40] and $Fe_{1-x}Co_xSi_2$ [41] chiral magnets, and in $FeGe_3$ thin films [42], have attracted considerable attention to systematic studies of noncoplanar magnetic states and the associated magnetoelectric phenomena [43–48]. In addition to ferromagnetic materials, skyrmions can be found in antiferromagnetic [49–51] and ferrimagnetic structures [52], and the existence of an antiskyrmion lattice has also been noted in Heusler alloys [53]. All of the listed spin configurations are two-dimensional topologically protected magnetic textures, whose stability is ensured by the presence of an energy barrier separating them from magnetic structures with a different topology [54–56]. The possibility of stabilizing three-dimensional topological spin configurations was noted recently; among them, the lattice of magnetic hedgehogs [57] and the magnetic Hopfion [58] are singled out. Importantly, the microscopic mechanism responsible for the formation of a magnetic hedgehog lattice in noncentrosymmetric metals is associated with competition between the antisymmetric DMI due to spin–orbit coupling and multispin interactions [59]. In contrast to the model of localized magnetic moments, the interaction between atoms is long-range in this case, and the effective spin Hamiltonian can be obtained from the sd-exchange model by integrating out the degrees of freedom associated with itinerant electrons [60–63].

In this paper, we show how the methods of machine learning can be adapted to search for noncoplanar magnetic textures. In particular, the use of neural networks to find the minimum energy of interacting classical spins is discussed in detail. Before proceeding to a detailed description of the results obtained, we give a general introduction to machine learning methods, focusing on the structure of a multilayer perceptron, which represents a class of feed-forward neural networks. Next, we discuss the neural network algorithm that we use for the study of the formation of two-dimensional lattices of magnetic skyrmions and antiskyrmions in the model of localized magnetic moments. For two-dimensional magnetic systems, we also study the ground states of the effective spin Hamiltonian with bilinear and biquadratic interactions, obtained from the sd-exchange model by integrating out the degrees of freedom of the itinerant subsystem. For three-dimensional magnetic materials, similarly, we obtain spin configurations in the form of skyrmion tubes in the model of localized magnetic moments and a magnetic hedgehog lattice for the effective spin Hamiltonian with biquadratic interaction.

2. Overview of machine learning

The renewed interest in data mining methods in recent years, after the so-called winter of artificial intelligence, is associated with the widespread use of high-performance computing systems, on the one hand, and the high volume of accumulated data, on the other [64]. This naturally makes machine learning, which allows finding general patterns from individual posterior data, the main avenue in the development of artificial intelligence systems [65]. In machine learning, it is common to distinguish between supervised and unsupervised learning [66]. The difference between them is shown in Fig. 1. In supervised learning, the task is posed such that, given a set of input data, the program learns to predict output data that it has not encountered before; this relates to a classification problem. In the case of unsupervised learning, the program is assumed to automatically find patterns and dependences based on a set of input data; this relates to solving the clustering problem or creating generative models capable of producing new output data similar to what the program saw before. The most natural way to implement machine learning is via artificial neural networks [67].

An artificial neural network is a set of simple computation units, so-called neurons, that perform certain arithmetic operations with data received from other neurons and provide the result. Essentially, a neural network can be represented by a directed graph whose vertices are neurons and the connecting edges ensure directed data exchange between them, which means that the output of some neurons is used as input to others. In each individual neuron, the input data received from other neurons are summed with some specific weights, and then a nonlinear activation function is applied to the resultant sum (Fig. 1a). We note that the choice of weights and the activation function can vary from neuron to neuron. Figure 1b shows one of the most important types of artificial neural networks: a multilayer perceptron, whose neurons are organized into layers such that the output of each layer is used as the input to the next [68]. The first, or input, layer of the network serves to input data into the neural network, and the last, or output, layer delivers the result of calculations of the entire network. Depending on the architecture of the neural network, there may also be hidden layers between the input and output layers, those output is not directly observable when using the network. The number of layers of a multilayer perceptron is called the network depth, and the number of neurons in a layer is called the width of that layer.

It can easily be understood that the output of a multilayer perceptron produces a function that is a composition of the nonlinear functions of each layer. It turns out that any continuous function of many variables can be approximated as accurately as desired by a multilayer perceptron. This statement corresponds to the so-called universal approximation theorem, often also called the Tsybenko theorem [69]. In practice, the function we want to determine using a neural network is known only indirectly through some sampling, and the task of machine learning is to find a sufficiently good approximation for this function by adjusting the weights. The data set used to train a neural network is called the training data, and it consists of a sample distribution of the function we want to reconstruct. In other words, the data consists of sets $\{x_j, y_j\}_{j=1}^{N}$, where x and y denote the input data and the target value of the function corresponding to it, for a sample of N elements. The difference between the prediction of the neural network and the exact one is quantified by the so-called objective function. Typically, the objective function is chosen in the form of the mean squared error between the exact output value and the network prediction, but there are many other more specialized loss functions, such as cross entropy.

The process of training a neural network, schematically presented in Fig. 1c, consists of minimization of the objective function averaged over the training data set with respect to its weights using some version of gradient descent. The version used most commonly is stochastic gradient descent: the gradient is then calculated not for each element of the sample but for a randomly selected subset of the training data set at each training step, which simplifies the calculations, although

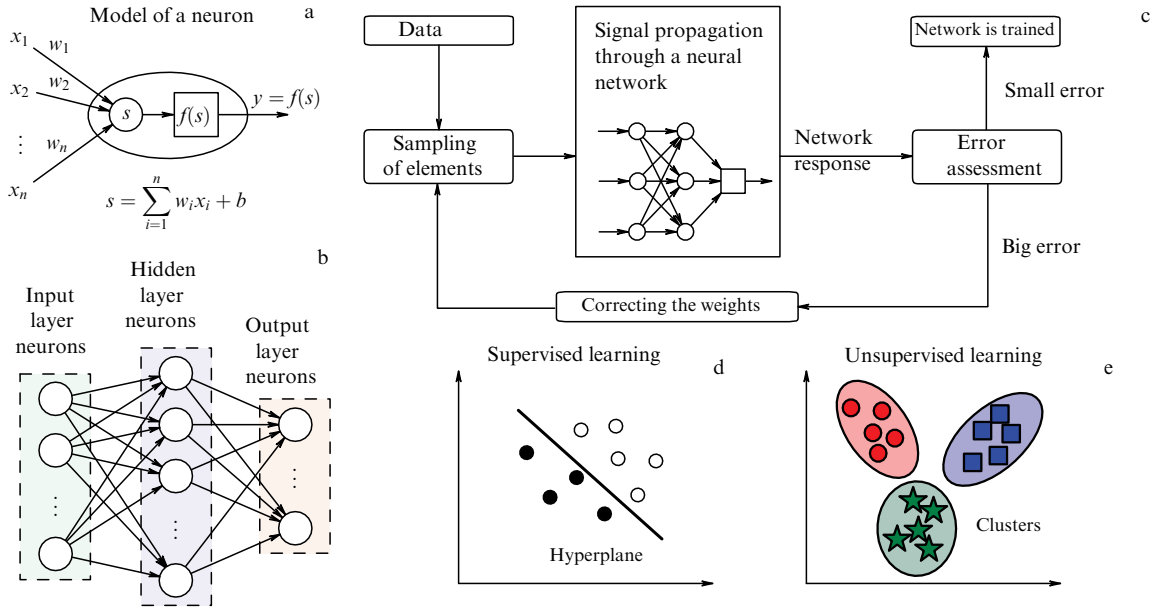

Figure 1. (a) Mathematical model of a neuron is given by a map $\mathbb{R}^n \to \mathbb{R}$ that takes input signals $x_1, x_2, \ldots, x_n$ to produce output y. Incoming signals x_i are summed with some weights (more precisely, synaptic weights) w_i, which express the degree of suppression or amplification of the incoming signal. Threshold nature of signal generation by a neuron is usually implemented by introducing so-called nonlinear activation function f. In its simplest form, this is a step function: neuron then calculates weighted sum s and generates an output signal if the value of s exceeds a threshold. (a) Artificial neural network is a software implementation of a mathematical model and can be regarded as a sequence of neurons connected by synapses. (b) Multilayer perceptron is structured as input and output layers, with one or more hidden layers in between. (c) Typical scheme for training an artificial neural network: elements from the data set are fed to neural network input, and initially the neural network weights are initialized randomly. Depending on the magnitude of error between the neural network prediction and the exact value expected at neural network output, the network is considered trained or the cycle repeats with a new sample and new values of the neural network weights adjusted so as to minimize the error. Neural network algorithms are widely used in machine learning problems, including both (d) supervised learning and (e) unsupervised learning.

in general, requires more training steps to cover the entire data set [70]. The number of training steps required to cover the entire training data set is called an epoch and is conventionally used as a measure of the learning rate of a neural network. Due to the multilayer perceptron design method, gradients can be efficiently calculated using the backpropagation method [71]. For this, we start at the end of the network and calculate the gradient relative to the weights of the last layer. Using the rule for the derivative of a composite function, such gradients can then be used to compute the gradient with respect to the weights of the preceding layer, which can be used to compute the next layer in the direction of the input layer, and so on.

3. Neural network algorithm

Machine learning methods, which have proven their efficiency in image recognition and natural speech systems [66, 72–76], have recently become a working tool in theoretical physics [77, 78]. Research in recent years has convincingly demonstrated that neural networks can be used to determine macroscopic phases of matter and associated phase transitions [79, 80], and also for effective representation of quantum states [81] and in quantum tomography problems [82, 83]. The effectiveness of machine learning methods to describe many-particle states lies in the use of effective approaches to reducing the dimension of the Hilbert space formed by all possible states of a quantum system. For example, for a wide class of magnetic compounds, the microscopic description of can be performed by the Heisenberg exchange model; neural network algorithms can then be successfully used to classify magnetic states, including magnetic skyrmions [84], and determine phase transitions in real materials [85]. An equally interesting direction is the use of the functional of neural networks directly to determine the ground state of magnetic Hamiltonians. In what follows, we describe the operation of a corresponding algorithm using the model of localized spins and a itinerant magnet as an example, and discuss the advantages of the method compared with standard approaches based, in particular, on the use of a simulated annealing algorithm.

3.1 Localized spins

We start with the Heisenberg magnet, whose Hamiltonian is given by

$$\mathcal{H}_{\text{exch}} = -\sum_{\langle i,j\rangle} J_{ij}\,\mathbf{S}_i\mathbf{S}_j - \sum_{\langle i,j\rangle} \mathbf{D}_{ij}\,(\mathbf{S}_i \times \mathbf{S}_j)$$

$$- A\sum_i (S_i^z)^2 - \sum_i \mathbf{h}\,\mathbf{S}_i\,, \qquad (1)$$

where the summation over nearest neighbors is understood in the first line. The first term in (1) corresponds to the symmetric exchange interaction, whose strength is determined by the exchange integral J_{ij}; the second term is the antisymmetric DMI, whose structure is determined by the Dzyaloshinskii vector $\mathbf{D}_{ij}$. The third term determines the uniaxial magnetic anisotropy of a magnitude $A > 0$, and the last term is the Zeeman energy in a uniform external magnetic field $\mathbf{h}$. It is assumed that localized spins are described by a unit-length classical vector field, $||\mathbf{S}_i|| = 1$.

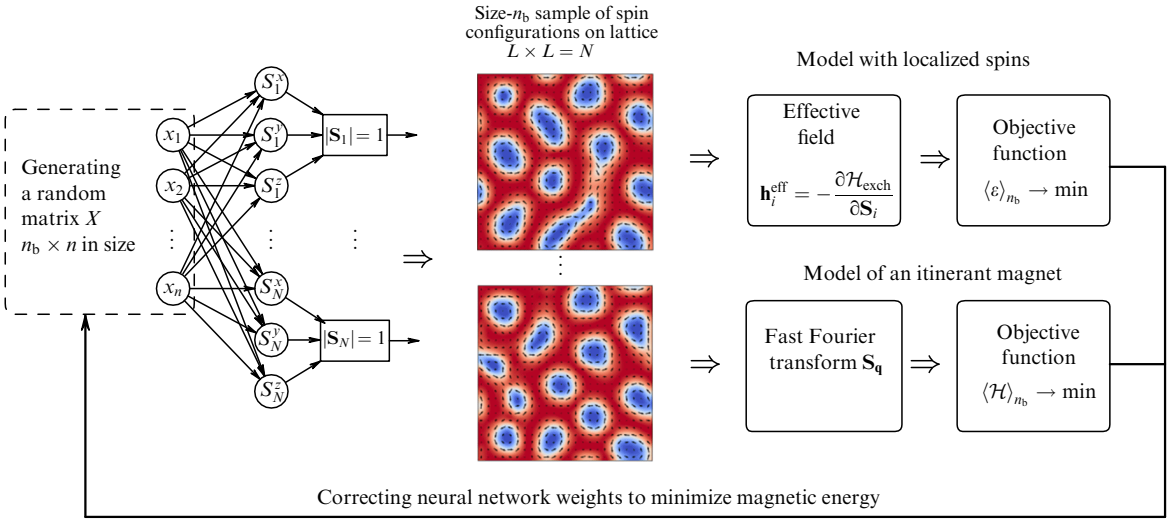

Figure 2. Block diagram of neural network algorithm. Numerical calculations use fully connected neural network without hidden layers: each n neuron of the input layer is connected to the $3N$ neurons of the output layer, where N is the total number of localized magnetic moments in the system, coinciding with the number of lattice sites. Process of neural network training begins with generating a matrix of random numbers $n \times n_b$ in size, which means initializing a set of n_b vectors, each containing n random numbers. Neural network weights, representing a matrix $n \times 3N$ in size, are also chosen randomly at the first stage. In the output layer, because hidden layers are absent, we have a linear combination of input vector components. We regroup neurons of the output layer into N three-dimensional vectors $(\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_N)$ and normalize each $\mathbf{S}_i$ vector to unity. This method of normalizing input data is a realization of the activation function. We consider resulting N unit-length vectors as if they represent a distribution of localized magnetic moments on the lattice. From input matrix X, we obtain n_b such spin configurations. For each of them, in the case of a model with localized spins, we calculate the effective field $\mathbf{h}^{\text{eff}}$ in accordance with Eqn (3) and magnetic energy (2) averaged over n_b realizations, and, for the itinerant magnet model, we move to spins in the momentum representation and directly calculate Hamiltonian (5) averaged over n_b. Expressions $\langle \varepsilon \rangle_{n_b}$ or $\langle \mathcal{H}_{\text{itin}} \rangle_{n_b}$ obtained in this way are implicit functions of synaptic weights; neural network training process consists in determining weights that minimize these functions and is implemented using the gradient descent method. Each iteration ends with updating neural network weights; loop repeats until a stable solution is reached, and at each iteration we use a newly generated random matrix X, which avoids falling into a local minimum.

We use a neural network algorithm based on [86–88] to find the spin configuration of the ground state of model (1), depending on the parameters of magnetic interactions. Recall that an artificial neural network is a mathematical implementation of a model organized in accordance with the principle of functioning of a biological neural network (see Fig. 1a, b); in other words, a sequence of neurons connected to each other by synapses is implemented at the software level. Depending on the architecture of the neural network, there may also be hidden layers between the input and output layers; information is fed to the input layer neurons, the neurons of the hidden layers perform simple computations with it, and the output layer neurons deliver the result of the calculation. In turn, a synapse is characterized by a single parameter, the synaptic weight, which shows how much input information changes when transmitted from one neuron to another. In this paper, we use a fully connected neural network without hidden layers — with the stipulations that each neuron of a fully connected neural network be connected to all other neurons located in the neighboring layers, and all connections be directed strictly from input to output neurons.

In Figure 2, we show a block diagram of the neural network algorithm that we use. The input and output layers respectively contain n and $3N$ neurons, with N being the total number of lattice sites. We feed to the neural network a matrix X with each of its n_b columns representing an n-dimensional vector of normally distributed random numbers. At the output of the neural network, we have a set of n_b configurations of the form $(\mathbf{S}_1, \ldots, \mathbf{S}_N)$, where each three-dimensional vector $\mathbf{S}_i$ is normalized to unity.

In what follows, we regard the n_b resulting configurations $(\mathbf{S}_1, \ldots, \mathbf{S}_N)$ as N spins living on the original lattice and use them to calculate the average over the n_b energy realizations in the effective magnetic field $\mathbf{h}^{\text{eff}}$ created by the localized magnetic moments:

$$\langle \varepsilon \rangle_{n_b} = -\frac{1}{N} \sum_{i=1}^{N} \langle \mathbf{S}_i \mathbf{h}_i^{\text{eff}} \rangle_{n_b} . \tag{2}$$

It is worth noting that finding the minimum energy from Eqn (2) is equivalent to searching for stationary solutions of the Landau–Lifshitz equation [89, 90]. The effective magnetic field for each localized spin is defined as $h_\alpha^{\text{eff}} = -\partial \mathcal{H}_{\text{exch}} / \partial S_\alpha$ $(\alpha = x, y, z)$, and Eqn (2) represents the objective function of the problem under consideration. It is easy to see that the Hilbert space dimension of the original problem defined by Hamiltonian (1) is reduced in the neural network representation, because we go from the parameterization in terms of the vector components of localized magnetic moments to parameterizing the magnetic energy by the neural network weights. The values of the neural network weights that minimize objective function (2) provide the set $(\mathbf{S}_1, \ldots, \mathbf{S}_N)$ at the output that is the best approximation of the ground-state spin configuration.

Recalling the antisymmetry of the Dzyaloshinskii vector $\mathbf{D}_{ji} = -\mathbf{D}_{ij}$, we can write the effective field in the form

$$\mathbf{h}_i^{\text{eff}} = 2 \sum_j W_{j-i} \mathbf{S}_j + 2 A S_i^z \hat{\mathbf{e}}_z + \mathbf{h} , \tag{3}$$

where $\hat{\mathbf{e}}_z$ is the unit vector along the z-axis. The first term on the right-hand side is the convolution of spins $\mathbf{S}_j$ with the

kernel $W_{j-i} = W_{ji}$,

$$W_{ij} = \begin{pmatrix} J_{ij} & -D_{ij}^{z} & D_{ij}^{y} \\ D_{ij}^{z} & J_{ij} & -D_{ij}^{x} \\ -D_{ij}^{y} & D_{ij}^{x} & J_{ij} \end{pmatrix}, \tag{4}$$

whose parameters depend on the distance between the spins at sites i and j and are determined by model Hamiltonian (1).

At each iteration in the neural network training process, effective field (3) is calculated using the vectors $(\mathbf{S}_1, \ldots, \mathbf{S}_N)$ obtained at the neural network output. The gradient descent method is used to minimize objective function (2); upon completion of an iteration, the weights of the neural network take new values, and the cycle is repeated until a stable solution is obtained. We note that, at each new iteration, a newly generated random matrix X is fed to the neural network input. This technique is used to avoid falling into a local minimum.

3.2 Itinerant magnet

In an itinerant magnet, in contrast to the model of localized spins with short-range magnetic interactions, the indirect RKKY interaction between localized magnetic moments is long range. The RKKY interaction stabilizes the spin helix with an incommensurate period, while taking multispin interactions into account leads to a rearrangement of this state into a noncoplanar magnetic texture. The effective spin Hamiltonian is then given by [59, 62]

$$\mathcal{H}_{\text{itin}} = -2\sum_{\eta}\left[\tilde{J}\,\mathbf{S}_{\mathbf{Q}_\eta}\mathbf{S}_{-\mathbf{Q}_\eta} - \frac{K}{N}\left(\mathbf{S}_{\mathbf{Q}_\eta}\mathbf{S}_{-\mathbf{Q}_\eta}\right)^2 + i\tilde{\mathbf{D}}_\eta\left(\mathbf{S}_{\mathbf{Q}_\eta}\times\mathbf{S}_{-\mathbf{Q}_\eta}\right)\right] - \sum_i \mathbf{h}\,\mathbf{S}_i\,, \tag{5}$$

where $\mathbf{S}_{\mathbf{q}} = (1/\sqrt{N})\sum_i \mathbf{S}_i \exp(-i\mathbf{q}\,\mathbf{r}_i)$ is the Fourier transform of a localized spin $\mathbf{S}_i$. We note that Eqn (5) arises as an effective spin Hamiltonian when integrating out conduction electrons in the sd-exchange model on a cubic lattice and subsequently expanding in powers of the sd-exchange interaction [59–63]. The first nonvanishing term represents the RKKY interaction of strength $\tilde{J}$. Among various multispin interactions that arise in the subsequent expansion, we keep only the biquadratic exchange coupling of magnitude K [91], determined by the second term. In a cubic noncentrosymmetric crystal, the presence of spin-orbit coupling gives rise to the DMI, determined in this case by the vectors $\tilde{\mathbf{D}}_\eta$. All exchange interactions in (5) are long-range in real space, but the form of writing (5) corresponds to an effective spin Hamiltonian where the wave vectors $\mathbf{Q}_\eta$ are determined by the maxima of the spin susceptibility of the itinerant subsystem [62].

We show how a neural network algorithm can be adapted to analyze models with long-range action like (5). The main part of the algorithm remains the same, but instead of calculating the effective field, we apply the fast Fourier transform to obtain a set of n_b spin configurations in momentum space $(\mathbf{S}_{\mathbf{q}_1}, \ldots, \mathbf{S}_{\mathbf{q}_N})$, as shown in Fig. 2. Next, we extract the Fourier components $\mathbf{S}_{\mathbf{Q}_\eta}$ determined by the set of wave vectors $\mathbf{Q}_\eta$, calculate the conjugate $\mathbf{S}_{-\mathbf{Q}_\eta} = \tilde{\mathbf{S}}_{\mathbf{Q}_\eta}$, and substitute the result directly into Eqn (5). Note that all calculations are performed for two-dimensional and three-dimensional lattices with periodic boundary conditions. The numerical results presented below were obtained with the hyperparameters $n_\text{b} = 1024$ and $n = 64$, by implementing 10^4 steps of the Adam optimizer [92] with a learning rate of 10^{-3}. The algorithm for finding the ground-state spin configuration for the Hamiltonian was implemented using the TensorFlow machine learning library, and the calculations were performed on a GPU. It is worth noting that, in machine learning, hyperparameters are usually understood as parameters whose values are set prior the start of the learning process. In our case, hyperparameters include the number n of neurons in the input layer and the number n_b of vectors at the neural network input. They should not be confused with the parameters calculated during the training: synaptic weights and the value of the minimized function. These parameters are optimized during the training of a neural network.

4. Two-dimensional magnetic systems

We consider the results obtained using a neural network algorithm in the case where localized spins interacting in accordance with exchange Hamiltonian (1) are located at the sites of a square lattice. We focus on the formation of a lattice of magnetic skyrmions and antiskyrmions. Here and hereafter, we use magnetization, the structure factor, and scalar spin chirality to quantify magnetic textures. For each of the obtained spin configurations, the magnetization

$$M = \frac{1}{N}\left\langle\left|\sum_i \mathbf{S}_i\right|\right\rangle_{n_\text{b}} \tag{6}$$

is obtained by averaging over the set n_b; the structure factor

$$\mathcal{S}(\mathbf{Q}) = \frac{1}{N^2}\sum_{ij}\langle\mathbf{S}_i\,\mathbf{S}_j\rangle_{n_\text{b}}\exp\left[-i\mathbf{Q}\,(\mathbf{r}_i - \mathbf{r}_j)\right] \tag{7}$$

is found from the Fourier transform of the spin-spin correlation function; and the scalar spin chirality

$$\chi = \frac{1}{4\pi N}\sum_{\Delta}\left\langle\mathbf{S}_i\,(\mathbf{S}_j\times\mathbf{S}_k)\right\rangle_{n_\text{b}} \tag{8}$$

is calculated by summing the mixed spin products over triangular plaquettes. Importantly, the structure factor allows detecting long-range magnetic order in the system, and a nonzero scalar spin chirality guarantees the noncoplanarity of the spin configuration.

4.1 Skyrmions

We start with the Hamiltonian of localized spins $\mathcal{H}_\text{exch}$ on a 24×24 lattice. We choose the Dzyaloshinskii vector $\mathbf{D}_{ij} = D\mathbf{r}_{ij}$ in accordance with the symmetry of the system under consideration, where D determines the DMI strength and the unit vector $\mathbf{r}_{ij}$ connects the ith and jth lattice sites; we also ignore the contribution from uniaxial magnetic anisotropy (assuming $A = 0$). We further assume that the exchange interaction $J = 1$ determines the energy scale, fix $D/J = 1.4$, and take the magnetic field to be directed along the $\hat{\mathbf{e}}_z$-axis. The phase diagram depending on the DMI parameter, D, and the magnetic field strength h are presented, for example, in [93].

Magnetic configurations in real space determined by the machine learning method are shown in Fig. 3. Competition between the symmetric exchange interaction and the antisymmetric DMI leads to the stabilization of the spin helix, as shown in Fig. 3a. According to the conventional classifica-

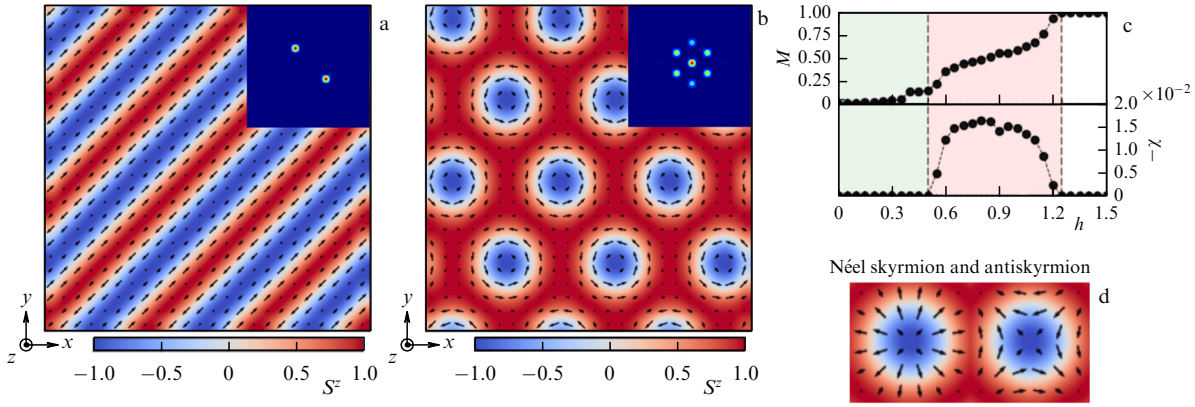

Figure 3. Ground-state spin configurations in real space, obtained for the Hamiltonian of localized magnetic moments (1) using a neural network algorithm on a 24×24 square lattice at a fixed $D = 1.4J$, corresponding to (a) a spin helix ($h = 0$) and (b) a lattice of Bloch magnetic skyrmions ($h = 0.75J$). Components (S_i^x, S_i^y) of localized magnetic moments are shown with black arrows in the xy plane, S_i^z projection is shown in color. Insets to (a) and (b) also show structure factor $\mathcal{S}(\mathbf{Q})$. (c) Magnetization M per lattice site and scalar spin chirality χ depending on the external magnetic field strength h in the [001] direction. Green, red, and white areas in the figure mark the respective spin helix, magnetic skyrmion lattice, and spin-polarized state. (d) Néel skyrmion and antiskyrmion, also studied in the framework of model (1).

tion, a multi-Q state is understood as a spatially modulated magnetic structure with several coexisting Q vectors [30]. Notably, the spin helix in Fig. 3a corresponds to a $1Q$ state; its structure factor (7) contains two symmetric peaks (twists in opposite directions) and is shown separately in the inset to Fig. 3a. In a uniform magnetic field, the helical order is unstable with respect to the formation of a hexagonal lattice of skyrmions, and $\mathcal{S}(\mathbf{Q})$ exhibits a distinct structure of six peaks (Fig. 3b) representing a superposition of three spin helices (the so-called $3Q$ state). We note that the trivial peak at $\mathbf{Q} = 0$ is due to the appearance of uniform magnetization along the field direction.

In Figure 3c, we plot magnetization (6) and scalar spin chirality (8) as functions of the external magnetic field strength h. For small external fields $h \leqslant 0.5J$, the chirality χ is zero, which indicates a coplanar ordering of spins corresponding to a spin helix. As the magnetic field increases, a noncoplanar spin texture ($\chi \neq 0$) appears in the form of a skyrmion lattice, which exists up to $h \approx 1.3J$. Finally, in high fields $h \geqslant 1.3J$, the magnetization reaches saturation: the magnetic system enters a spin-polarized state. We note that choosing the directions of the Dzyaloshinskii vectors $\mathbf{D}_{ij}$ allows stabilizing not only the Bloch skyrmions shown in Fig. 3b but also the Néel skyrmions and antiskyrmions (see, e.g., [94]), as shown in Fig. 3d.

4.2 Biquadratic interaction

We consider a two-dimensional spin model with bilinear and biquadratic spin interactions, defined by the Hamiltonian $\mathcal{H}_{\text{itin}}$ under the assumption that the antisymmetric DMI is absent ($\tilde{\mathbf{D}}_\eta = 0$). In real space, multispin interaction in the form of biquadratic exchange corresponds to the $(\mathbf{S}_i\mathbf{S}_j)^2$ coupling between magnetic moments localized at the ith and jth sites. In Eqn (5), following [63], we fix a pair of wave vectors $\mathbf{Q}_1 = (\pi/3, \pi/3)$ and $\mathbf{Q}_2 = (\pi/3, -\pi/3)$, which corresponds to a square lattice, a 24×24 one as previously, and $\tilde{J} = 1$.

Using a neural network algorithm in the form shown in Fig. 2, we can easily verify that the RKKY interaction stabilizes the spin helix. Taking the multispin interaction $K > 0$ into account gives rise to a $2Q'$ spatially modulated magnetic structure. It is worth noting that Q' means that the components of $\mathbf{Q}_1$ and $\mathbf{Q}_2$ in the structure factor $\mathcal{S}(\mathbf{Q})$ correspond to peaks of different intensities (Fig. 4a). In the presence of an external magnetic field, competition between K and h allows stabilizing four different states [63]: $1Q$, a pair of $2Q'$, and $2Q$ spatially modulated magnetic structures. The last is of greatest interest, because it can be represented as a periodic array of vortices and antivortices (Fig. 4b). The structure factor $\mathcal{S}(\mathbf{Q})$ contains two pairs of equal-intensity peaks at $\pm\mathbf{Q}_1$ and $\pm\mathbf{Q}_2$, and a peak at $\mathbf{Q} = 0$ coming from the magnetization induced by the external magnetic field.

5. Three-dimensional magnetic systems

We now proceed to the analysis of numerical results obtained using a neural network algorithm in the case of three-dimensional magnetic systems, focusing on the formation of skyrmion tubes in the model of localized magnetic moments and a magnetic hedgehog lattice for an effective spin Hamiltonian with bilinear and biquadratic couplings. Determining the microscopic mechanism responsible for the stability of the magnetic hedgehog lattice is of independent academic interest in light of recent experimental data on the existence of two different types of lattices in MnSi chiral magnets doped with germanium, depending on the concentration of the dopant [57]. In the calculations in what follows, we use a simple cubic lattice with the total number of sites $16 \times 16 \times 16$ and periodic boundary conditions.

5.1 Skyrmion tubes

As an example of a three-dimensional topological spin texture, we consider a lattice of skyrmion tubes. As in the two-dimensional case, we start from the Hamiltonian of localized magnetic moments, Eqn (1), on a simple cubic lattice. We note that the presence of uniaxial anisotropy is an essential factor that allows stabilizing this magnetic structure [95]. We again choose $J = 1$ as the unit of measurement of energy in the system; similarly to the two-dimensional case, we parameterize the antisymmetric DMI by the vectors $\mathbf{D}_{ij} = D\mathbf{r}_{ij}$. We fix $D = J$ and $K = 0.16J$ and demonstrate the formation of skyrmion tubes in an external uniform magnetic field along the [001] direction of the crystal.

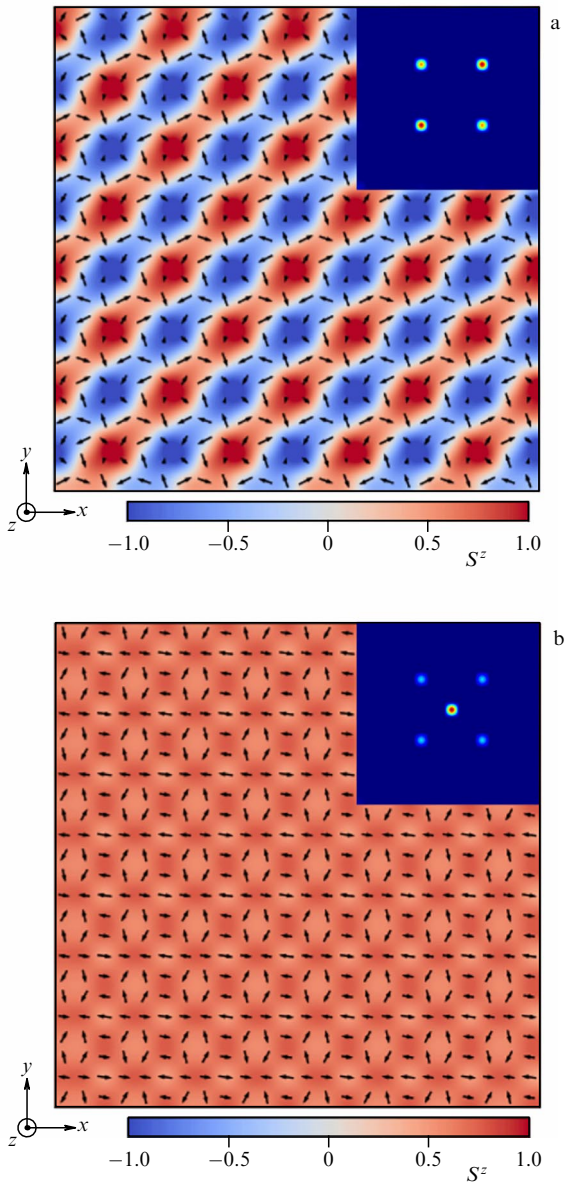

Figure 4. Spin structures in real space obtained for the effective spin model (5) using a neural network algorithm on a 24×24 square lattice at fixed $K = 0.6\tilde{J}$, corresponding to (a) $2Q'$ for $h = 0$ and (b) $2Q$ for $h = \tilde{J}$ spatially modulated magnetic textures. Insets also show structure factor $\mathcal{S}(\mathbf{Q})$.

To quantify the noncoplanarity of spin configurations, we use the scalar spin chirality, which is defined in the three-dimensional case as follows. First, we define the spin chirality at site i as the sum of mixed products of spins located on triangular plaquettes covering $\mathbf{S}_i$,

$$\chi_i^{\gamma} = \frac{1}{2} \sum_{\alpha,\beta,\nu_\alpha,\nu_\beta} \epsilon^{\alpha\beta\gamma} \nu_\alpha \nu_\beta \, \mathbf{S}_i (\mathbf{S}_{i+\nu_\alpha \hat{\mathbf{e}}_\alpha} \times \mathbf{S}_{i+\nu_\beta \hat{\mathbf{e}}_\beta}) \,, \tag{9}$$

where $\epsilon^{\alpha\beta\gamma}$ is the totally antisymmetric tensor, $\alpha, \beta, \gamma = x, y, z$ and $\nu_{\alpha,\beta} = \pm 1$. Summing over all sites and γ and averaging over the n_b, we obtain the spin chirality of the system $\chi = N^{-1} \sum_{\gamma,i} \langle \chi_i^{\gamma} \rangle_{n_\mathrm{b}}$.

Spin configurations in the form of skyrmion tubes in real space, calculated using the neural network algorithm, are shown in Fig. 5a. Evidently, a uniaxial anisotropy $A > 0$ along the z-axis allows the Bloch-type skyrmion lattice to be stabilized in the xy plane. This state is stable at external magnetic field strengths $0.27J < h < 0.45J$, where the scalar spin chirality χ is nonzero, which indicates the noncoplanar nature of spin ordering (Fig. 5b). In magnetic fields $h \leqslant 0.27J$, the spin configuration of the ground state corresponds to a spin helix, and at $h \geqslant 0.45J$ the system goes into a fully spin-polarized state where the magnetization reaches saturation.

5.2 Magnetic hedgehog lattice

Another example of a three-dimensional topological spin texture is the magnetic hedgehog lattice, which represents a periodic array of spin monopoles and antimonopoles. We consider the effective spin Hamiltonian in Eqn (5) and choose the wave vectors $\mathbf{Q}_\eta$ corresponding to so-called $4Q$ magnetic hedgehogs (the $3Q$ case was considered in [88]). For this, we fix the vectors $\mathbf{Q}_1 = (Q, -Q, -Q)$, $\mathbf{Q}_2 = (-Q, Q, -Q)$, $\mathbf{Q}_3 = (-Q, -Q, Q)$, and $\mathbf{Q}_4 = (Q, Q, Q)$, which form a tetrahedron. Following [59], we choose $Q = \pi/4$ (the period of the structure then corresponds to eight lattice sites) and assume $\tilde{\mathbf{D}}_\eta \parallel \mathbf{Q}_\eta$ (under the condition $||\tilde{\mathbf{D}}_\eta|| = \tilde{D}$). The magnetic field $\mathbf{h}$ is chosen along the [001], [110], and [111] directions in accordance with [59].

As a result of the competition between the antisymmetric DMI and the multispin (in this case, biquadratic) interaction, the magnetic system defined by Hamiltonian (5) for the chosen $\mathbf{Q}_\eta$ has a nontrivial phase diagram [59]. For example, depending on a different relation between K and $\tilde{D}$, a $1Q$ spin helix, a $2Q$ spatially modulated magnetic texture, or a $4Q$ magnetic hedgehog lattice can be stabilized.

Let us discuss the spin configuration corresponding to the magnetic hedgehog lattice in somewhat greater detail. Figure 6a shows a $4Q$ magnetic hedgehog lattice in real space obtained using the neural network algorithm in the absence of an external field. The scalar spin chirality χ depending on the external field strength h along the [001], [110], and [111] directions is shown in Fig. 6b. For these field directions, as can be seen, the magnetic hedgehog lattice is, respectively, stable up to $h_c^{[001]} \simeq 2\tilde{J}$, $h_c^{[110]} \simeq 1.5\tilde{J}$, and $h_c^{[111]} \simeq 1.1\tilde{J}$. We note that the presented results based on a neural network algorithm are consistent with standard calculation methods using a simulated annealing algorithm [59].

6. Conclusions

We have demonstrated the possibility of using neural networks to search for the ground state of magnetic systems, focusing on topological spin textures. In the model of localized magnetic moments, the formation of topologically protected magnetic textures is the result of competition between interactions that are bilinear in spins. We illustrated this statement by demonstrating the formation of a lattice of magnetic skyrmions and antiskyrmions in two-dimensional magnetic systems, and of skyrmion tubes in three-dimensional magnets in the framework of numerical minimization of the classical spin Hamiltonian in the presence of the symmetric exchange interaction and the antisymmetric DMI. On the other hand, in the framework of effective spin models with long-range magnetic interactions, the formation of topological spin textures is caused by competition between the antisymmetric DMI and multispin interactions. Using our neural network algorithm for a spin model with biquadratic exchange, we demonstrated the formation of a $2Q$ spatially modulated magnetic structure in a two-dimensional system,

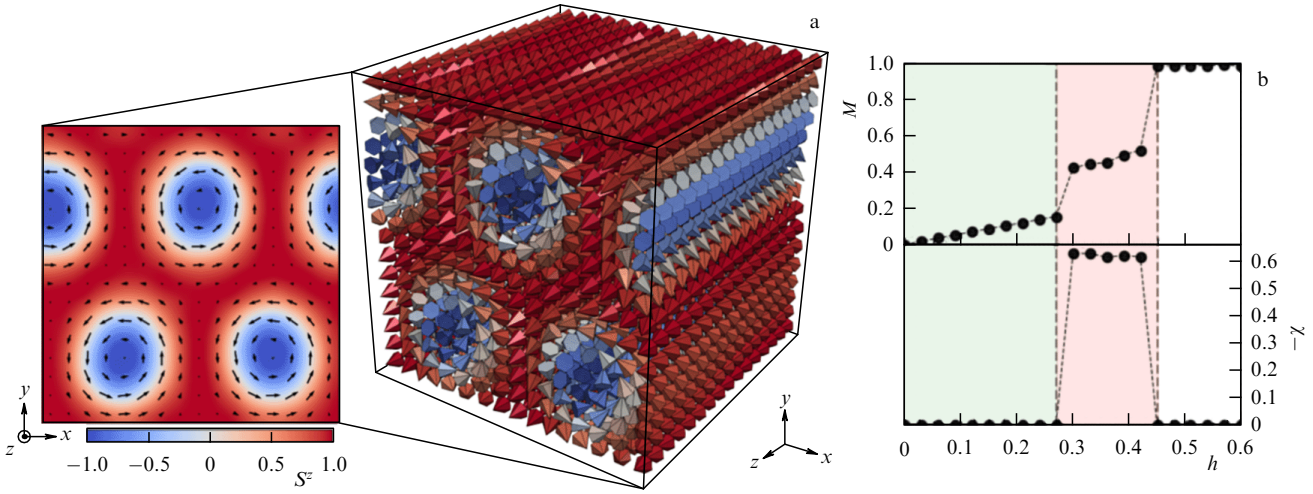

Figure 5. (a) Visualization of a three-dimensional lattice of skyrmion tubes in real space, stabilized at $D = J$, $A = 0.16J$, and $h = 0.4J$ and calculated using a neural network algorithm for exchange Hamiltonian (1) on a $16 \times 16 \times 16$ cubic lattice. Each cone shows the spin orientation, with z-components highlighted in color, and the cross section of the spin structure is drawn perpendicular to the $\hat{\mathbf{e}}_z$ direction. (b) Magnetization M and scalar spin chirality χ depending on the external magnetic field strength directed along [001]. Green, red, and white regions represent magnetic states corresponding to the respective spin helix, skyrmion tube lattice, and spin-polarized state.

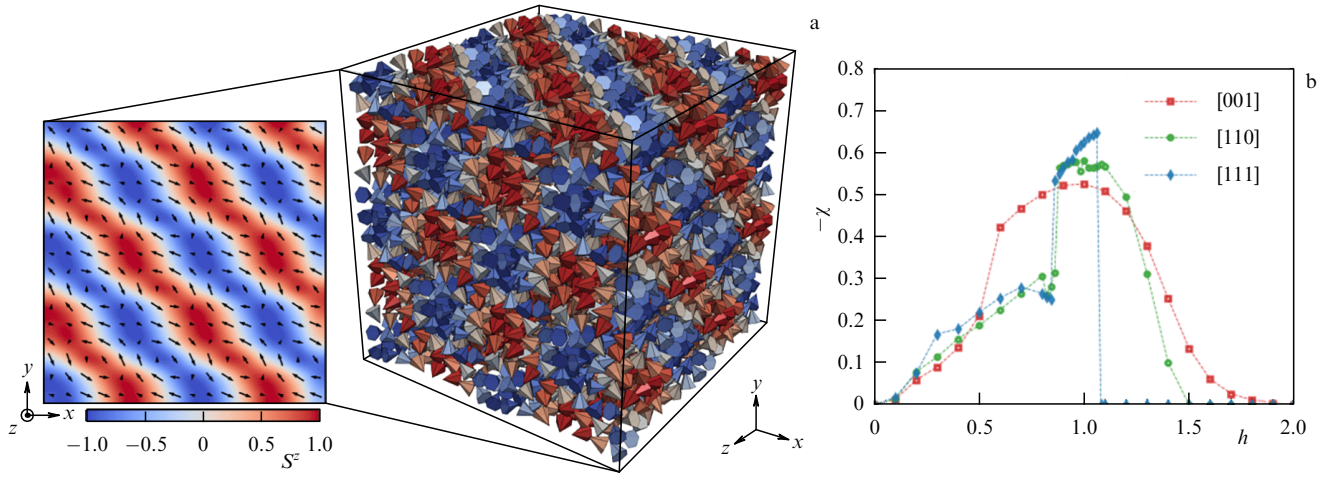

Figure 6. (a) Topological magnetic structure visualizing a $4Q$ lattice of magnetic hedgehogs in real space at $K = 0.6\tilde{J}$ and $\tilde{D} = 0.3\tilde{J}$, calculated using a neural network algorithm in the framework of the effective spin model (5) on a $16 \times 16 \times 16$ simple cubic lattice in the absence of an external field. (b) Scalar spin chirality χ estimated for the $4Q$ magnetic hedgehog lattice as a function of the external magnetic field strength in [001], [110], and [111] directions.

as well as the stabilization of a magnetic hedgehog lattice in three-dimensional magnets.

To conclude, we compare the neural network algorithm with standard methods for finding the ground-state spin configurations of magnetic Hamiltonians. Most popular among the standard methods are various versions of the Monte Carlo method, for example, in the form of a simulated annealing algorithm [96], and so-called greedy algorithms (sometimes also parsimonious search algorithms) [97], well known from graph theory and amounting to the adoption of locally optimal strategies. Following [86], we consider a skyrmion on a square lattice. The parsimonious search algorithm minimizes the classical spin Hamiltonian (1) by locally aligning some selected ith spin along the direction of the effective magnetic field $\mathbf{h}_i^{\text{eff}}$. This process is repeated with updated magnetic moments until the system reaches a stable state. This algorithm has a sufficiently high convergence, but its solution is not accurate; in fact, the system may find itself in a local minimum [86]. This difficulty can be overcome in the framework of simulating the annealing algorithm by temperature fluctuations. In the neural network algorithm, falling into a local minimum state is ruled out by the construction: at each iteration, we use newly generated random matrices X (see Fig. 2). Obviously, the accuracy of the neural network algorithm is strongly dependent on the number of neurons in the input layer. However, even for a sufficiently small number of input neurons, the solution with the neural network algorithm is more accurate in less computation time than with the Monte Carlo method [86]. As an illustration, we consider the spin configurations obtained at the neural network output, depending on the number of iterations (Fig. 7): it is clear that the neural network finds a solution in the form of a lattice of magnetic skyrmions in 10^3 to 10^4 iterations. The simplicity of the chosen neural network architecture (see Fig. 2), which does not involve hidden layers, is explained by the fact that a linear map suffices for obtaining different spin configurations from the random matrix X at each iteration.

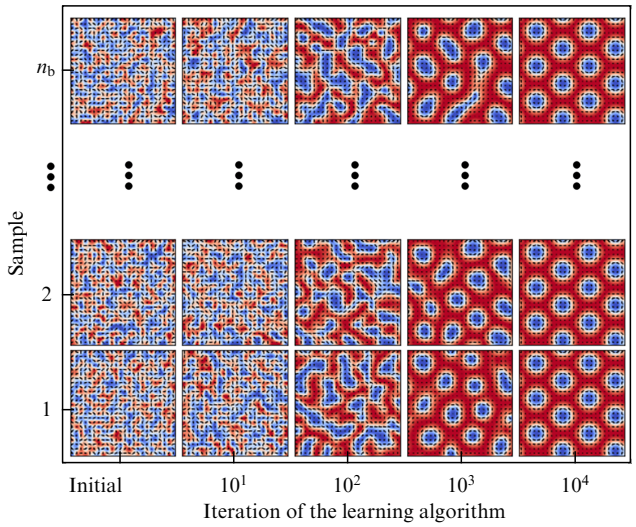

Figure 7. Spin configurations at the output of the neural network shown in Fig. 2, based on the results of 10, 10^2, 10^3, and 10^4 iterations. A stable solution in the form of a hexagonal lattice of magnetic skyrmions minimizing spin Hamiltonian (1) is attained in 10^3 to 10^4 iterations. A linear map (neural network without hidden layers) is sufficient to obtain spin configurations from a random matrix at each iteration.

Acknowledgments. A A Pervishko's work was carried out with financial support from the Russian Science Foundation, grant 22-72-00021 and a scholarship from the President of the Russian Federation, SP-1640.2021.5. G V Paradezhenko's work was supported by the Russian Science Foundation, grant 22-11-00074. D I Yudin thanks the Government of the Russian Federation for the grant for state support of scientific research conducted under the guidance of leading scientists in Russian institutions of higher education, scientific organizations, and state research centers (Megagrants program, project no. 075-15-2021-607, Ferromagnetic spinorbitronics).

References

1. Irkhin V Yu, Irkhin Yu P *Elektronnaya Struktura, Fizicheskie Svoistva i Korrelyatsionnye Effekty v* d- *i* f-*Metallakh i Ikh Soedineniyakh* (Electronic Structure, Physical Properties and Correlation Effects in d- and f-Metals and Their Compounds) (Ekaterinburg: UrO RAN, 2004)
2. Heisenberg W *Z. Phys.* **49** 619 (1928)
3. Eriksson O et al. *Atomistic Spin Dynamics*: *Foundations and Applications* (Oxford: Oxford Univ. Press, 2017)
4. Vonsovskii S V *Zh. Eksp. Teor. Fiz.* **16** 981 (1946)
5. Kondo J *Prog. Theor. Phys.* **32** 37 (1964)
6. Hewson A C *The Kondo Problem to Heavy Fermions* (Cambridge: Cambridge Univ. Press, 2003)
7. Ruderman M A, Kittel C *Phys. Rev.* **96** 99 (1954)
8. Kasuya T *Prog. Theor. Phys.* **16** 45 (1956)
9. Yosida K *Phys. Rev.* **106** 893 (1957)
10. Abrikosov A A *Sov. Phys. Usp.* **12** 168 (1969); *Usp. Fiz. Nauk* **97** 403 (1969)
11. Kondo J *Solid State Phys.* **23** 183 (1970)
12. Tsvelick A M, Wiegmann P B *Adv. Phys.* **32** 453 (1983)
13. Dzyaloshinsky I *J. Phys. Chem. Solids* **4** 241 (1958)
14. Moriya T *Phys. Rev.* **120** 91 (1960)
15. Emori S et al. *Nat. Mater.* **12** 611 (2013)
16. Ryu K-S et al. *Nat. Nanotechnol.* **8** 527 (2013)
17. Yang H et al. *Phys. Rev. Lett.* **115** 267210 (2015)
18. Soumyanarayanan A et al. *Nature* **539** 509 (2016)
19. Caretta L et al. *Nat. Commun.* **11** 1090 (2020)
20. Samardak A S et al. *Phys. Chem. Chem. Phys.* **24** 8225 (2022)
21. Park J et al. *Acta Mater.* **241** 118383 (2022)
22. Bogdanov A N, Yablonskii D A *Sov. Phys. JETP* **68** 101 (1989); *Zh. Eksp. Teor. Fiz.* **95** 178 (1989)
23. Bogdanov A *JETP Lett.* **62** 247 (1995); *Pis'ma Zh. Eksp. Teor. Fiz.* **62** 231 (1995)
24. Bogdanov A, Hubert A *J. Magn. Magn. Mater.* **138** 255 (1994)
25. Bogdanov A N, Rößler U K *Phys. Rev. Lett.* **87** 037203 (2001)
26. Rößler U K, Bogdanov A N, Pfleiderer C *Nature* **442** 797 (2006)
27. Rößler U K, Leonov A A, Bogdanov A N *J. Phys. Conf. Ser.* **200** 022029 (2010)
28. Kiselev N S et al. *J. Phys. D* **44** 392001 (2011)
29. Dzyaloshinskii I E *Sov. Phys. JETP* **19** 960 (1964); *Zh. Eksp. Teor. Fiz.* **46** 1420 (1964)
30. Izyumov Yu A *Sov. Phys. Usp.* **27** 845 (1984); *Usp. Fiz. Nauk* **144** 439 (1984)
31. Yi S D et al. *Phys. Rev. B* **80** 054416 (2009)
32. Han J H et al. *Phys. Rev. B* **82** 094429 (2010)
33. Nagaosa N, Tokura Y *Nat. Nanotechnol.* **8** 899 (2013)
34. Ambrose M C, Stamps R L *New J. Phys.* **15** 053003 (2013)
35. Pereiro M et al. *Nat. Commun.* **5** 4815 (2014)
36. Yudin D, Gulevich D R, Titov M *Phys. Rev. Lett.* **119** 147202 (2017)
37. Böttcher M et al. *New J. Phys.* **20** 103014 (2018)
38. Nishikawa Y, Hukushima K, Krauth W *Phys. Rev. B* **99** 064435 (2019)
39. Mohanta N, Dagotto E, Okamoto S *Phys. Rev. B* **100** 064429 (2019)
40. Mühlbauer S et al. *Science* **323** 915 (2009)
41. Yu X Z et al. *Nature* **465** 901 (2010)
42. Yu X Z et al. *Nat. Mater.* **10** 106 (2011)
43. Lee M et al. *Phys. Rev. Lett.* **102** 186601 (2009)
44. Kanazawa N et al. *Phys. Rev. Lett.* **106** 156603 (2011)
45. Schulz T et al. *Nat. Phys.* **8** 301 (2012)
46. Hayashi Y et al. *Nat. Commun.* **12** 5974 (2021)
47. Bernevig B A, Felser C, Beidenkopf H *Nature* **603** 41 (2022)
48. Wang H et al. *Prog. Mater. Sci.* **130** 100971 (2022)
49. Barker J, Tretiakov O A *Phys. Rev. Lett.* **116** 147203 (2016)
50. Bessarab P F et al. *Phys. Rev. B* **99** 140411 (2019)
51. Legrand W et al. *Nat. Mater.* **19** 34 (2020)
52. Woo S et al. *Nat. Commun.* **9** 959 (2018)
53. Nayak A K et al. *Nature* **548** 561 (2017)
54. Borisov A B *Phys. Usp.* **63** 269 (2020); *Usp. Fiz. Nauk* **190** 291 (2020)
55. Tokura Y, Kanazawa N *Chem. Rev.* **121** 2857 (2021)
56. Göbel B, Mertig I, Tretiakov O A *Phys. Rep.* **895** 1 (2021)
57. Fujishiro Y et al. *Nat. Commun.* **10** 1059 (2019)
58. Kent N et al. *Nat. Commun.* **12** 1562 (2021)
59. Okumura S et al. *Phys. Rev. B* **101** 144416 (2020)
60. Akagi Y, Udagawa M, Motome Y *Phys. Rev. Lett.* **108** 096401 (2012)
61. Hayami S, Motome Y *Phys. Rev. B* **90** 060402 (2014)
62. Hayami S, Ozawa R, Motome Y *Phys. Rev. B* **95** 224424 (2017)
63. Hayami S, Motome Y *Phys. Rev. Lett.* **121** 137202 (2018)
64. Pickover C A *Artificial Intelligence*: *an Illustrated History*: *from Medieval Robots to Neural Networks* (New York: Sterling, 2019); Translated into Russian: *Iskusstvennyi Intellekt. Illyustrirovannaya Istoriya*: *ot Avtomatov do Neirosetei* (Moscow: Sindbad, 2021)
65. Abu-Mostafa Y S, Magdon-Ismail M, Lin H-T *Learning From Data* (Pasadena, CA: AMLBook, 2012)
66. LeCun Y, Bengio Y, Hinton G *Nature* **521** 436 (2015)
67. Bishop C M *Rev. Sci. Instrum.* **65** 1803 (1994)
68. Hastie T, Tibshirani R, Friedman J *The Elements of Statistical Learning*: *Data Mining, Inference, and Prediction* (New York: Springer, 2009)
69. Cybenko G *Math. Control. Signal. Systems* **2** 303 (1989)
70. Spall J C *Introduction to Stochastic Search and Optimization*: *Estimation, Simulation, and Control* (Hoboken, NJ: Wiley-Interscience, 2003)
71. Rumelhart D E, Hinton G E, Williams R J *Nature* **323** 533 (1986)
72. Nikolenko S, Kadurin A, Arkhangel'skaya E *Glubokoe Obuchenie. Pogruzhenie v Mir Neironnykh Setei* (Deep Learning. Dive into the World of Neural Networks) (St. Petersburg: Piter, 2018)
73. Patterson J, Gibson A *Deep Learning*: *a Practitioner's Approach* (Sebastopol, CA: O'Reilly, 2017); Translated into Russian: *Glubokoe Obuchenie s Tochki Zreniya Praktika* (Moscow: DMK Press, 2018)

74. Deng L, Liu Y (Eds) *Deep Learning in Natural Language Processing* (Singapore: Springer, 2018)
75. Bishop C M *Pattern Recognition and Machine Learning* (New York: Springer, 2006); Translated into Russian: *Raspoznavanie Obrazov i Mashinnoe Obuchenie* (Moscow–St. Petersburg: Dialektika, 2020)
76. Tunstall L, von Werra L, Wolf T *Natural Language Processing with Transformers*: *Building Language Applications with Hugging Face* (Sebastopol: O'Reilly Media, 2022)
77. Carleo G et al. *Rev. Mod. Phys.* **91** 045002 (2019)
78. Karniadakis G E et al. *Nat. Rev. Phys.* **3** 422 (2021)
79. Carrasquilla J, Melko R G *Nat. Phys.* **13** 431 (2017)
80. van Nieuwenburg E P L, Liu Y-H, Huber S D *Nat. Phys.* **13** 435 (2017)
81. Carleo G, Troyer M *Science* **355** 602 (2017)
82. Torlai G et al. *Nat. Phys.* **14** 447 (2018)
83. Macarone Palmieri A et al. *npj Quantum Inf.* **6** 20 (2020)
84. Iakovlev I A, Sotnikov O M, Mazurenko V V *Phys. Rev. B* **98** 174411 (2018)
85. Shirinyan A A et al. *Phys. Rev. B* **99** 041108 (2019)
86. Kwon H Y et al. *Phys. Rev. B* **99** 024423 (2019)
87. Swain N et al. *Phys. Rev. B* **104** 235156 (2021)
88. Paradezhenko G V et al. *Phys. Chem. Chem. Phys.* **24** 24317 (2022)
89. Zvezdin A K, Zvezdin K A, Khval'kovskii A V *Phys. Usp.* **51** 412 (2008); *Usp. Fiz. Nauk* **178** 436 (2008)
90. Pervishko A A, Yudin D I *Phys. Usp.* **65** 215 (2022); *Usp. Fiz. Nauk* **192** 233 (2022)
91. Kartsev A et al. *npj Comput. Mater.* **6** 150 (2020)
92. Kingma D P, Ba J, arXiv:1412.6980
93. Kapitan V et al. *AIP Adv.* **11** 015041 (2021)
94. Huang S et al. *Phys. Rev. B* **96** 144412 (2017)
95. Chen J et al. *Sci. Rep.* **7** 7392 (2017)
96. van Laarhoven P J M, Aarts E H L *Simulated Annealing*: *Theory and Applications* (Norwell, MA: Kluwer Acad. Publ., 1987)
97. Cormen T H et al. *Introduction to Algorithms* (Cambridge, MA: MIT Press, 2001); Translated into Russian: *Algoritmy. Postroenie i Analiz* (Algorithms. Construction and Analysis) (Moscow–St. Petersburg: Williams, 2005)