

Supervised learning approach for recognizing magnetic skyrmion phases

I. A. Iakovlev, O. M. Sotnikov, and V. V. Mazurenko

Theoretical Physics and Applied Mathematics Department, Ural Federal University, Mira Street 19, Ekaterinburg 620002, Russia

 (Received 18 March 2018; revised manuscript received 16 October 2018; published 7 November 2018)

We propose and apply simple machine learning approaches for recognition and classification of complex noncollinear magnetic structures in two-dimensional materials. The first approach is based on the implementation of the single-hidden-layer neural network that only relies on the z projections of the spins. In this setup, one needs a limited set of magnetic configurations to distinguish ferromagnetic, skyrmion, and spin spiral phases, as well as their different combinations in transitional areas of the phase diagram. The network trained on the configurations for the square-lattice Heisenberg model with Dzyaloshinskii-Moriya interaction can classify the magnetic structures obtained from Monte Carlo calculations for a triangular lattice and vice versa. The second approach we apply, a minimum distance method, performs a fast and cheap classification in cases when a particular configuration is to be assigned to only one magnetic phase. The methods we propose are also easy to use for analysis of the numerous experimental data collected with spin-polarized scanning tunneling microscopy and Lorentz transmission electron microscopy experiments.

DOI: [10.1103/PhysRevB.98.174411](https://doi.org/10.1103/PhysRevB.98.174411)

I. INTRODUCTION

Fascinating progress in the development of neural-network-based approaches in condensed-matter theory allows one to advance the methods for studying the physical properties of materials. For instance, a neural network representation of the quantum Hamiltonian wave function proposed by Carleo and Troyer [1] has revolutionized the simulation of complex many-body systems [2,3]. Within such an approach, it becomes possible to model frustrated systems for which existing quantum Monte Carlo methods fail due to the sign problem. Another remarkable example of the innovations in artificial neural network learning is the identification of the magnetic phases of the spin Hamiltonians widely used to describe strongly correlated materials [4–10]. For instance, in the case of the two-dimensional Ising model, the ferromagnetic and paramagnetic phases can be successfully recognized with a single-hidden-layer network [11]. Importantly, topological phases obtained with a more complex XY Hamiltonian [12,13] can also be classified with machine learning, however in this case one needs to design a deep convolutional network and use a system of filters, which makes such an approach similar to image recognition [14].

Therefore, an important question arises. Is it possible to use the machine learning approach in its simplest and transparent formulation with a single hidden layer [11] to explore complex noncollinear magnetic phases of technological importance? In this respect, topologically protected magnetic skyrmions [15–19] and spin spiral states are the first candidates for such a consideration, since they can be used to create novel magnetic memory devices [20]. Numerous experimental studies revealed the skyrmion state in metallic ferromagnets with Dzyaloshinskii-Moriya interaction, such as FeGe [21,22], Fe monolayer on Ir(111) [23], MnGe [24], and $\text{Fe}_x\text{Co}_{1-x}\text{Si}$ [25] in a narrow range of external parameters, magnetic fields, and temperatures. The experimental phase diagrams of these materials [25] contain significant transitional

areas between different phases, which raises the problem of the precise definition of the skyrmion and spin spiral phase boundaries.

Here we show that a standard feed-forward network (FFN) can be used efficiently for supervised learning on topologically protected magnetic skyrmion states and spin spirals originating from the spin-orbit coupling. Figure 1 illustrates the idea of our approach. A noncollinear magnetic configuration obtained from Monte Carlo simulations describing a two-dimensional ferromagnet with Dzyaloshinskii-Moriya interaction [Fig. 1(a)] is projected on the z axis [Fig. 1(b)]. This z -projected magnetic structure is considered as input for the single-hidden-layer network [Fig. 1(c)]. Having trained such a network on a limited set of configurations belonging to pure ferromagnetic, skyrmion, and spiral states on the square lattice, we were able to recognize the states from completely different parts of the phase diagram, including transitional areas between different phases. Moreover, we found that the trained network can classify the data collected for a triangular lattice, which demonstrates the universality of our approach. Another important result of our work is the demonstration of a high classification performance achieved with a nearest centroid method. Although it is one of the simplest machine learning techniques, the centroid classifier nevertheless shows very accurate results in the case of unseen data on skyrmion and spin spiral configurations.

II. MODEL AND METHOD

In our study, to simulate the topological magnetic excitations we used the following spin Hamiltonian on the 48×48 square lattice:

$$H = - \sum_{i < j} J_{ij} \mathbf{S}_i \mathbf{S}_j - \sum_{i < j} \mathbf{D}_{ij} [\mathbf{S}_i \times \mathbf{S}_j] - \sum_i B S_i^z, \quad (1)$$

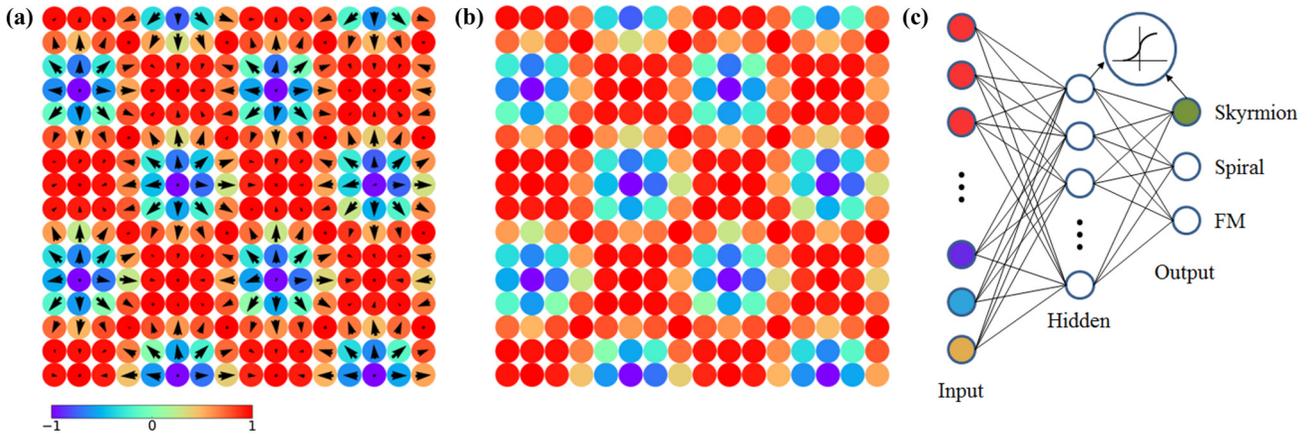


FIG. 1. Schematic representation of the machine learning process. (a) The skyrmion magnetic structure as obtained from the classical Monte Carlo simulations for a two-dimensional ferromagnet with Dzyaloshinskii-Moriya interaction at finite temperature and magnetic field. Black arrows indicate the in-plane xy spin components. (b) The matrix containing the z projection of the spin structure to be classified. (c) Neural network having a single hidden layer of sigmoid neurons. The values of the input neurons are equal to z components of the spins of the magnetic configuration.

where J_{ij} and \mathbf{D}_{ij} are the isotropic exchange interaction and Dzyaloshinskii-Moriya vector, respectively. \mathbf{S}_i is a unit vector along the direction of the i th spin, and B denotes the z -oriented magnetic field. We take into account only the interactions between nearest neighbors. The isotropic exchange interaction is positive in our simulations, which corresponds to the ferromagnetic case. The symmetry of the Dzyaloshinskii-Moriya vectors is of C_{4v} type. The Dzyaloshinskii-Moriya interaction (DMI) has an in-plane orientation and is perpendicular to the corresponding intersite radius vector. The Hamiltonian was solved by using the classical Monte Carlo approach. The spin update scheme is based on the METROPOLIS algorithm. The systems in question are gradually (200 temperature steps) cooled down from high temperatures ($T \sim 3J$) to the required temperature. Each temperature step run consists of 1.5×10^6 Monte Carlo steps.

To identify the different magnetic phases of the spin Hamiltonian, Eq. (1), we calculated spin-spin correlation functions [26] and topological charges [27] (the corresponding expressions are presented in Appendix A), and visualized a number of spin configurations from each simulation. By using such information, the neural network was trained as described below.

III. NEURAL NETWORK

In our study, we employ a standard network architecture that is a one-layer feed forward network [presented in Fig. 1(c)]. It consists of one hidden layer of sigmoid activation neurons and three output sigmoid neurons that activate depending on the particular magnetic phase. For the training set, we generated 1000 configurations for each of the ferromagnetic, skyrmion, and spiral states corresponding to the areas marked in Fig. 2. In these simulations, we fixed $J = 1$ and used a uniform distribution for the magnetic field and Dzyaloshinskii-Moriya interaction. The simulation temperatures were taken in the range $T \in [0.02, 0.1]$ in units of isotropic exchange interaction. Moreover, we generated 1000 configurations belonging to the paramagnetic phase at high

temperatures ($T \sim 10J$) and added them to the training set. For these paramagnetic configurations, the ground-truth labels of all the output neurons were set to zero.

The main challenge in machine learning for classification of magnetic phases is how to relate the states of the input neurons of the network to the particular magnetic configuration. As was shown in Ref. [11], in the case of the Ising model with $S^z = \pm 1$ there is a one-to-one correspondence between the neuron values in the input layer and the spins of the particular configurations. As another example, for the XY model solutions characterized by in-plane noncollinear magnetic states, the authors of Ref. [12] used angles magnitudes determining the in-plane orientations of the spins.

In the case of noncollinear magnetic configurations, the situation is more complicated because the orientation of a spin cannot be described by a single angle value. However,

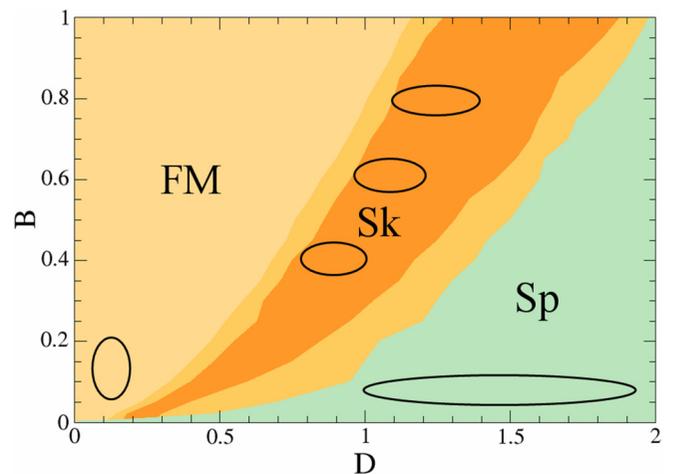


FIG. 2. Phase diagram in terms of Dzyaloshinskii-Moriya interaction and magnetic field. The abbreviations Sk, FM, and Sp denote skyrmion lattice state, ferromagnetic, and spin spiral state, respectively. The phase diagram was obtained at $T = 0.02$. All the parameters are given in units of J . Black ovals denote the phase areas used for supervised learning.

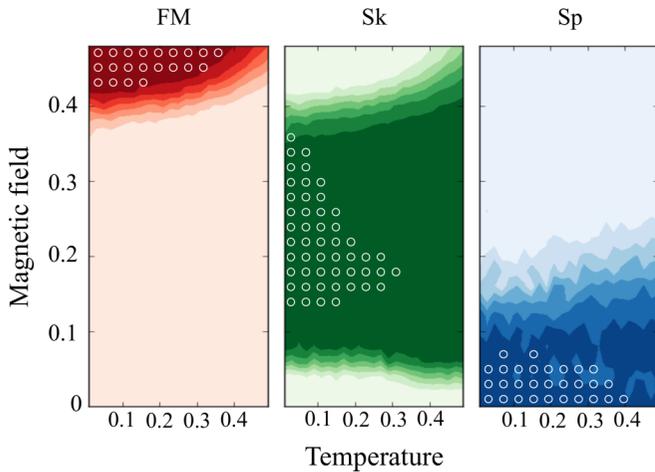


FIG. 3. Phase triptych obtained by using the neural network with 64 hidden neurons. Color intensities indicate the values of the output neurons for different phases; dark and light colors correspond to 1 and 0, respectively. The Dzyaloshinskii-Moriya interaction was chosen to be $D = 0.72$. All the parameters are given in units of J . White circles denote the phase boundaries defined with the spin structure factors.

one can make use of the fact that skyrmions are characterized by a typical profile, i.e., the core and background spins of a skyrmion aligned antiparallel and parallel to the applied magnetic field [Fig. 1(b)], respectively. This means that the skyrmion excitation can be detected by analyzing the only z components of the spins [28]. We use this fact to realize our neural network approach, in which the values of the input neurons are equal to the z components of the spins obtained from Monte Carlo simulations of Eq. (1). As we will show below, such an approach also works well in the case of spin spiral and ferromagnetic phases.

The network was trained to minimize the error function that is a standard mean-squared error (MSE) function. Weights of neurons were adjusted by means of the backpropagation method. Details of the learning process are given in Appendix B. The network was trained with different numbers of hidden neurons from 8 to 128. According to our simulations, the network with 64 hidden neurons gives reliable results for the phases recognition. We found that a further increase of hidden neurons for the considered case leads to a decrease in recognition quality. Thus, the total number of adjustable parameters are $64L^2 + 192$, which is much smaller than in previous work [12].

A. Phase diagram

The developed neural network approach was used for construction of the phase diagram of the spin model, Eq. (1), on the square lattice. To do this, we used a grid of 625 points on the temperature–magnetic-field plane. For each point, the values of the neural network output neurons were averaged over 10 Monte Carlo runs. Thus the total number of Monte Carlo calculations was equal to 6250.

From Fig. 3 one can see that the trained network can successfully recognize all the phases of interest at low temperature, which follows from a comparison with the

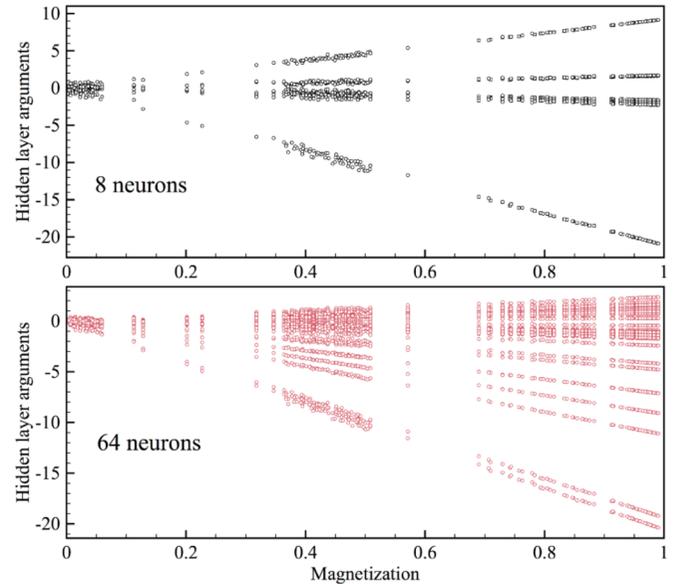


FIG. 4. Hidden-layer arguments as a function of the z -oriented magnetization of the simulated spin configurations for (top) 8- and (bottom) 64-hidden-neuron network.

boundaries obtained by calculating the structure factor (white circles). It is worth mentioning that we obtained a large value of the skyrmion number ($Q > 15$) for the parameters corresponding to the dark green area in Fig. 3. Importantly, it is possible to perform a composition analysis of the transitional areas between different phases. For each point of the phase diagram, one can define the values of the output neurons that indicate the contributions of the phases. This gives us an opportunity to solve the complex problem of defining the phase boundaries, and to quantitatively characterize the transitional areas between different phases [29,30].

B. Analysis of the classification process

The results of the previous neural-network-based studies [1,11–13] raise new fundamental questions on how a network learns different phases of matter. It was shown in Ref. [11] that identification of the Ising model states is related to the difference in total magnetization of the spin configurations belonging to different phases. In our case, such an explanation can also be used since the phases we simulated are characterized by different magnetizations. The magnetization per spin, $m(x) = \frac{1}{N} \sum_i^N S_i^z$, in the training set is in the range [0.91, 0.99], [0.38, 0.53], and [0, 0.03] for ferromagnetic, skyrmion, and spin spiral phases, respectively. At the same time, the test sets include pure spin configurations that are characterized by wider ranges of the average magnetization: [0.84, 0.99] (ferromagnetic), [0.33, 0.69] (skyrmion), and [0, 0.07] (spin spiral). In agreement with Ref. [11], we obtain that the components of the vector Wx (here W is the weight matrix between input and hidden layers) become linear functions of the magnetization $m(x)$ (Fig. 4). However, in our case the increase in the number of hidden neurons leads to a larger number of neuron categories, which may mean that the

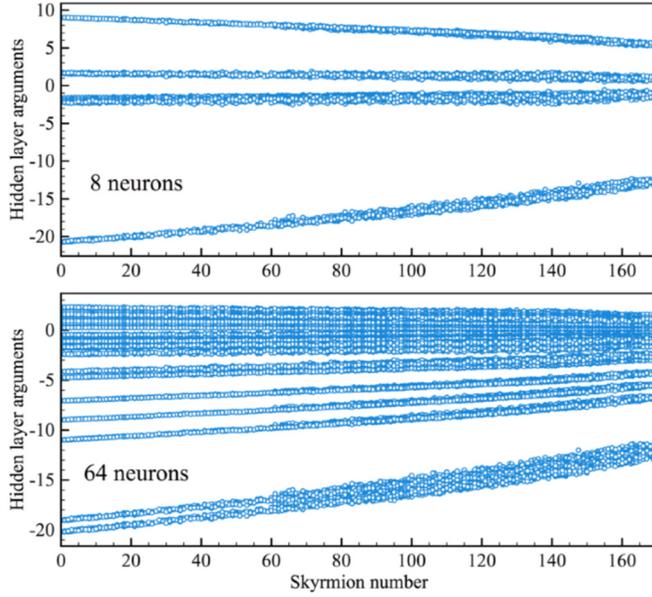


FIG. 5. Hidden-layer arguments as a function of the skyrmion number of the simulated spin configurations for (top) 8- and (bottom) 64-hidden-neuron network. The results have been obtained for the pure DMI model having $J = 0$.

magnetization is not the only parameter the network uses for recognition.

Since the focus of this study is on skyrmion phase recognition, we have also investigated the dependence of the hidden-neurons arguments on the topological charge. For that purpose, the pure DMI model with zero isotropic exchange interaction was simulated at different magnitudes of magnetic field. This gives us the opportunity to produce 2000 magnetic configurations characterized by completely different skyrmion numbers (from 0 to 170) for the same system size. These results are presented in Fig. 5. All the neurons can be divided into two categories. The first one corresponds to neurons with argument values that are close to zero and are not sensitive to topological charge. The second depends on the skyrmion number of the particular magnetic configuration.

To understand neural network internal functioning, one can also visualize hidden-layer neurons. By the example of the configuration with big skyrmions presented in Fig. 6,

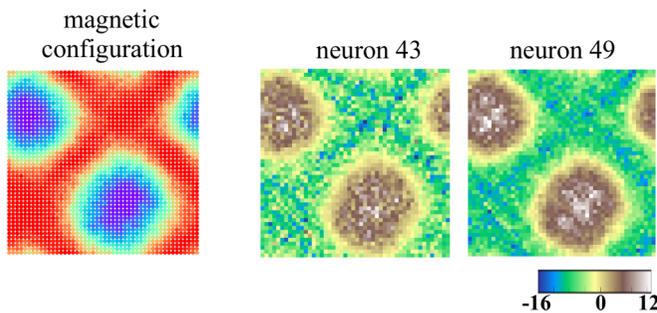


FIG. 6. (Left) z -projection of the skyrmion magnetic configuration obtained with the parameters $J = 1$, $D = 0.2$, $T = 0.02$, and $B = 0.02$. (Right) Visualization of the arguments of the specific hidden layer neurons.

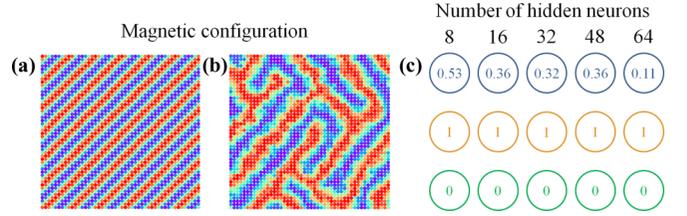


FIG. 7. (a) Example of a spiral state ($D = 1.4$, $B = 0.02$, $T = 0.05$, and $J = 1$) used for training the network. (b) Example of a complex spiral configuration from a test set obtained with $D = 0.72$, $B = 0.03$, $T = 0.22$, and $J = 1$. (c) The output neurons values in the case of configuration (b) depending on the number of hidden neurons. Numbers in blue, orange, and green circles correspond to values of skyrmion, spiral, and FM outputs, respectively.

we performed such an analysis. Importantly, the size of the skyrmions in the training dataset does not exceed $10a$, where a is the lattice constant, but we found that the trained neural network correctly classifies the configurations having skyrmions of much larger diameter. Indeed, the diameter of the skyrmion in Fig. 6 is about $35a$, and such a skyrmion state is uniquely recognized by the neural network even with eight hidden neurons.

Figure 6 gives a two-dimensional representation of two hidden-neuron arguments that are the weight matrix multiplied by spin z components corresponding to the magnetic configuration. The maximal and minimal intensities of the core and background areas of the skyrmions are different for these neurons. Nevertheless, one can easily recognize the original skyrmion structure. The visualization of the neural network weights by themselves does not give any useful information about network functioning.

As a hard test for our neural-network approach, we generated 300 high-temperature spiral configurations ($T \in [0.18; 0.26]$, $D = 0.72$, $B = 0.03$). A typical example of such configurations is presented in Fig. 7(b). It is of labyrinth type and consists of broken spin spirals that are distorted due to temperature effects. Importantly, the training set contains only ideal spin spirals presented in Fig. 7(a). One can see that an increase in the number of hidden neurons leads to a decrease in the value of the output neuron corresponding to the skyrmion phase that provides a more accurate phase separation. Having analyzed this test set, we found that the total number of clearly recognized configurations increased from 40% to 75% using 8 and 64 hidden neurons, respectively.

C. Variation of the lattice structure

The next step of our investigation is to examine the network trained on the square-lattice magnetic configurations for recognizing the phases of the spin Hamiltonian on the triangular lattice. For that we solved Eq. (1) with DMI of C_{3v} symmetry and generated magnetic configurations belonging to skyrmion, spin spiral, and ferromagnetic phases as well as their mixtures. Figure 8 gives the corresponding examples. For preparation of the test configurations, we have solved the spin Hamiltonian, Eq. (1), on a 48×48 triangular lattice with periodic boundary conditions. The supercell of the rhombic shape was replicated. A square area of 48×48 spins cropped

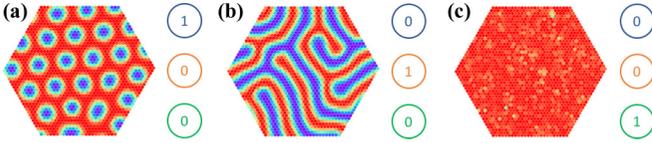


FIG. 8. Examples of (a) skyrmion, (b) spin spiral, and (c) ferromagnetic configurations stabilized on the triangular lattice and recognized with the neural network trained on square-lattice data. Numbers in blue, orange, and green circles correspond to values of skyrmion, spiral, and ferromagnetic outputs, respectively.

from the replicated lattice was used to define the values of the neural network input neurons.

It was found that the trained network classifies the skyrmion and ferromagnetic triangular-lattice configurations with high precision. In the case of the spin spiral states, the classification accuracy is low since such magnetic configurations [a typical example is presented in Fig. 8(b)] strongly differ from those we used in the training set [Fig. 7(a)].

IV. MINIMUM DISTANCE (NEAREST CENTROID) CLASSIFICATION

As was shown in the previous section, the neural network approach paves the way to exploring the magnetic phase diagram of noncollinear magnets including mixed states such as spin-spiral-skyrmion and skyrmion-ferromagnetic states. At the same time, the problem of when a particular state should be assigned to only one magnetic phase can be solved with a much simpler method.

In this section, we utilize the nearest centroid classification method as implemented in the SCIKIT-LEARN PYTHON package [31]. Figure 9 shows the overall process of the classifier training. As in the case of FFN, we perform data preprocessing by projecting local magnetization vectors on the z axis. The next step is to calculate mean data values for each class α (magnetic phase) in the training dataset. These mean values are called centroids and are given by

$$\langle \mathbf{X} \rangle_{\alpha} = \frac{1}{M_{\alpha}} \sum_{k=1}^{M_{\alpha}} \mathbf{X}_{\alpha}^{(k)}, \quad (2)$$

where $\mathbf{X}_{\alpha}^{(k)} = \{S_{z1}^{(k)}, S_{z2}^{(k)}, \dots, S_{zN}^{(k)}\}$ is a vector formed from z components of local magnetization $S_{zi}^{(k)}$ for the k th magnetic configuration, $\alpha = \text{FM, PM, Sk, Sp}$ denotes a phase and M_{α} is the number of the magnetic configurations belonging to the phase α . Thus, one can identify the phase α_{test} of a magnetic configuration \mathbf{X}_{test} by determining the minimum distance from it to the centroid of each class (magnetic phase):

$$d = \min_{\alpha} \{ \|\langle \mathbf{X} \rangle_{\alpha} - \mathbf{X}_{\text{test}}\| \}, \quad (3)$$

where $\|\dots\|$ indicates the norm of a high-dimensional vector.

As in the case of the neural network for training of the nearest centroids classifier, we used the same set comprising 4000 square-lattice magnetic configurations. Figure 9 (right panels) gives two-dimensional visualizations of the centroids calculated for different magnetic phases. As one can expect, the maximal and minimal centroid intensities are connected to the average magnetization per spin for each phase. However,

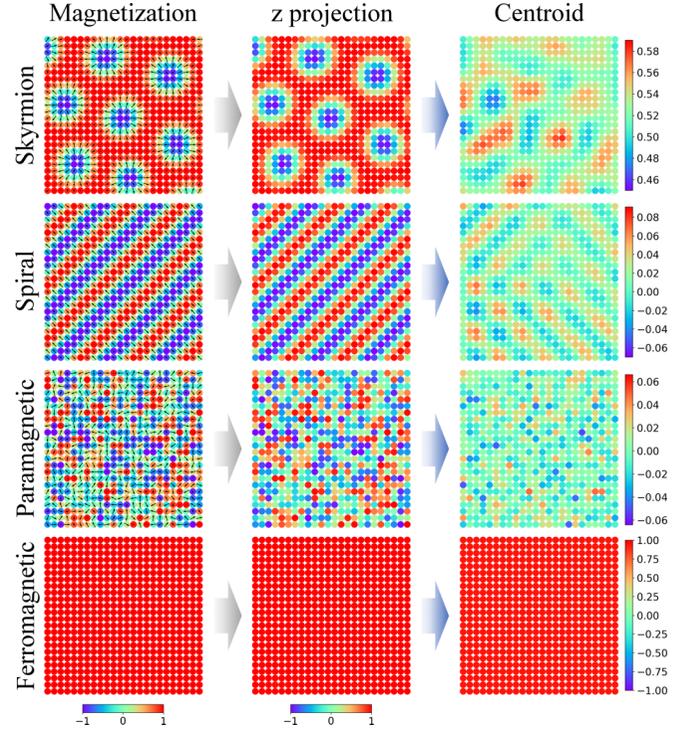


FIG. 9. Comparison of centroids calculated for different phases. Left panels are the examples of skyrmion, spin-spiral, paramagnetic, and ferromagnetic configurations. The arrows denote in-plane orientations of the magnetic moments. Middle panels are the corresponding z projections of the example magnetic configurations. Right panels represent two-dimensional visualizations of the centroids calculated with Eq. (2) for training datasets.

each centroid has a distinct magnetic pattern inherent to the corresponding phase. For example, the average magnetization values (per spin) of spin spiral and paramagnetic phases are close to zero, but the centroid of the spiral phase preserves the ordering, whereas the mean of the paramagnetic phase configurations remains disordered. This feature allows the method to distinguish PM and spin spiral phases.

The next important step is to estimate the performance of the algorithm on the unseen data, such as big skyrmions (Fig. 6), high-temperature spin spirals (Fig. 7), and triangular lattice configurations (Fig. 8). The results of the classification are presented in Table I. It was found that both the neural

TABLE I. Comparison of different ML classifiers trained with the same dataset. Testing sets include 100 big skyrmions (Big Sk), 300 high-temperature spin spirals (HT Sp), and a dataset for the triangular lattice (Δ).

Dataset	FFN (%)	Mean (%)	k -NN (%)
Big Sk	94	100	0
HT Sp	75	78	9
Δ FM	100	100	100
Δ Sp	40	54	25
Δ Sk	91	100	48
Δ PM	37	90	100

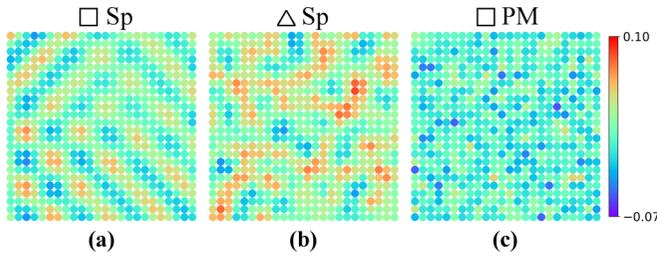


FIG. 10. Comparison of centroids calculated with (b) test set configurations on the triangular lattice, and (a,c) training set on the square lattice.

network and the nearest centroids classifier show comparable recognition accuracy for big skyrmions (94% and 100%), high-temperature spirals (75% and 78%), ferromagnetic configurations (both 100%), and skyrmion configurations on a triangular lattice (91% and 100%). Finally, the nearest centroid algorithm demonstrates excellent performance for the classification of the paramagnetic phase (90%). The centroid classifier shows slightly better classification of 880 spin spiral configurations stabilized on a triangular lattice (54%) than the neural network one, but recognition accuracy is still low. This can be due to the different topology of the underlying spiral structure for triangular and square lattices. Indeed, Fig. 10 shows that the centroid patterns for spin spirals stabilized on triangular and square lattices are completely different. One can also note that the calculated distances $\| \langle X \rangle_{\alpha} - \langle X \rangle_{\text{test}} \|$ from the triangular lattice spiral set centroid to the square lattice spiral and paramagnetic set centroids are comparable (1.7 and 1.8, respectively), thus explaining the part of the triangular spiral configurations recognized as paramagnetic.

V. k -NEAREST-NEIGHBOR CLASSIFIER

The choice of the best classifier algorithm is a nontrivial task and is highly dependent on the nature of the classified data and the purpose of classification. Often one needs to test a number of approaches to find the appropriate one. Here, we present results obtained for the k -nearest-neighbor method, which is widely used for classification tasks.

As for the nearest centroids classifier, in the k -NN method a magnetic configuration will be assigned to the specific class of magnetic configurations from the test set by using the distance metric, however the classification is now based on the closest k neighbors in the feature space, which is the space of the magnetization vector elements. It is known that the best choice of parameter k depends on the nature of the data. We found that the classification scheme based on three nearest neighbors ($k = 3$) shows the best results for our calculations. Figure 11 gives a one-dimensional representation of the training set within the k -NN method. There is a clear separation of the magnetic configurations belonging to the different phases. Since the training in the k -NN algorithm involves simply storing the magnetization vectors, the neural network approach works more slowly at this stage.

To estimate the accuracy of the k -NN algorithm, we carried out classification for our test datasets presented in Table I with this method. It was found that the k -NN method

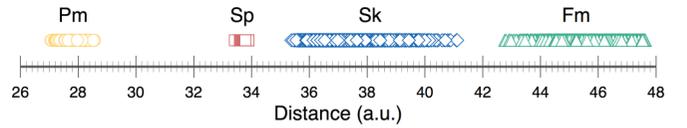


FIG. 11. One-dimensional visualization of a training set comprising 4000 magnetic configurations. Yellow circles, red squares, blue diamonds, and green triangles denote magnetic configurations belonging to paramagnetic, spiral, skyrmion, and ferromagnetic phases, respectively. They are distributed with respect to the distance from the origin in 2304-dimensional space (48×48 spins in total for each configuration).

improperly assigns the big skyrmions to ferromagnetic phase configurations, whereas the neural network correctly classifies 94% of such skyrmions. Then only 9% of 300 high-temperature spin spiral configurations of the labyrinth type were correctly classified. At the same time, the neural network approach demonstrates 75% accuracy for this test. Both the k -NN and neural network methods show 100% classification results in the case of ferromagnetic configurations (880 in total) stabilized on a triangular lattice. In turn, the neural network clearly surpasses k -NN for skyrmion states (880 in total) on the triangular lattice, 91% against 48%. Nevertheless, the k -NN classifier correctly recognizes all paramagnetic configurations.

VI. CONCLUSIONS

We have developed a neural-network-based approach for the recognition of magnetic phases of two-dimensional ferromagnets with Dzyaloshinskii-Moriya interaction in wide ranges of magnetic field and temperatures. One needs to generate only a limited set of magnetic configurations (~ 4000 in total) to train the network. It facilitates the construction of the phase diagram of the system in question during Monte Carlo sampling. Complex and mixed skyrmion-ferromagnetic and skyrmion-spin-spiral configurations can be quantitatively described, which was not possible before. The calculations for spin Hamiltonians on a 128×128 square lattice also demonstrated high accuracy in classification of the magnetic phases. We have shown that the method is not sensitive to the particular lattice structure used for training. By construction, the network approach allows one to recognize skyrmions of different types (Bloch and Néel). It can be used for on-the-fly classification of the skyrmion magnetic configurations observed in experiments. We have also utilized other widely used methods of machine learning classification, and showed that the proposed approach demonstrates a comparable performance with the nearest centroid classifier (except for the paramagnetic phase) and totally surpasses the k -nearest-neighbors method.

ACKNOWLEDGMENTS

We thank Andrey Bagrov, Frédéric Mila, and Jeanne Colbois for fruitful discussions. This work was supported by the Russian Science Foundation Grant No. 18-12-00185.

APPENDIX A: PROBLEM DEMONSTRATION

The aim of this appendix is to demonstrate the complexity of the magnetic phase classification problem using the example of skyrmionic materials. In our previous work [30], we have shown that there are five stable phases in a system described by the spin Hamiltonian, Eq. (1), which can be uniquely identified at low temperature by calculation of the spin structure factors and the skyrmion number (topological charge). The expressions for spin structure factors are given by

$$\chi_{\parallel}(\mathbf{q}) = \frac{1}{N} \left\langle \left| \sum_i S_i^z e^{-i\mathbf{q}\mathbf{r}_i} \right|^2 \right\rangle, \quad (\text{A1})$$

$$\chi_{\perp}(\mathbf{q}) = \frac{1}{N} \left\langle \left| \sum_i S_i^x e^{-i\mathbf{q}\mathbf{r}_i} \right|^2 + \left| \sum_i S_i^y e^{-i\mathbf{q}\mathbf{r}_i} \right|^2 \right\rangle, \quad (\text{A2})$$

where \mathbf{q} is the reciprocal space vector, S_i^{α} [$\alpha = (x, y, z)$] is the projection of the i th spin, and \mathbf{r}_i is the radius vector for the i th site.

In turn, the topological charge is defined in the following way:

$$Q = \frac{1}{4\pi} \sum_l A_l, \quad (\text{A3})$$

where A_l is the surface of spherical triangle l with vertices determined by three unit spin vectors,

$$A_l = 2 \arccos \left(\frac{1 + \mathbf{S}_i \cdot \mathbf{S}_j + \mathbf{S}_j \cdot \mathbf{S}_k + \mathbf{S}_k \cdot \mathbf{S}_i}{\sqrt{2(1 + \mathbf{S}_i \cdot \mathbf{S}_j)(1 + \mathbf{S}_j \cdot \mathbf{S}_k)(1 + \mathbf{S}_k \cdot \mathbf{S}_i)}} \right). \quad (\text{A4})$$

The sign of A_l in Eq. (A3) is determined as $\text{sgn}(A_l) = \text{sgn}(\mathbf{S}_i \cdot [\mathbf{S}_j \times \mathbf{S}_k])$. Importantly, we do not consider the exceptional configurations for which

$$\begin{aligned} \mathbf{S}_i \cdot [\mathbf{S}_j \times \mathbf{S}_k] &= \mathbf{0}, \\ 1 + \mathbf{S}_i \cdot \mathbf{S}_j + \mathbf{S}_j \cdot \mathbf{S}_k + \mathbf{S}_k \cdot \mathbf{S}_i &\leq 0. \end{aligned} \quad (\text{A5})$$

Figure 12 gives examples of the most interesting phases. As can be seen, all of them are recognized correctly by our neural network. However, we are not able to distinguish the first and second configurations by using the skyrmion number. Another problem is connected to the fact that if we rely only on Q and the spin-structure factor, the second state may be associated with the double- q skyrmion state, which is not the case. The developed network approach allows us to overcome this classification problem.

Figure 13 demonstrates examples of the nonperiodic skyrmion phase at low- and high-temperature pure skyrmion and mixed skyrmion-bimeron phases. As can be seen, all of them have the same smeared spin structure factors and approximately equal skyrmion numbers. This makes it impossible to distinguish them by using common techniques. At the same time, the developed neural network provides

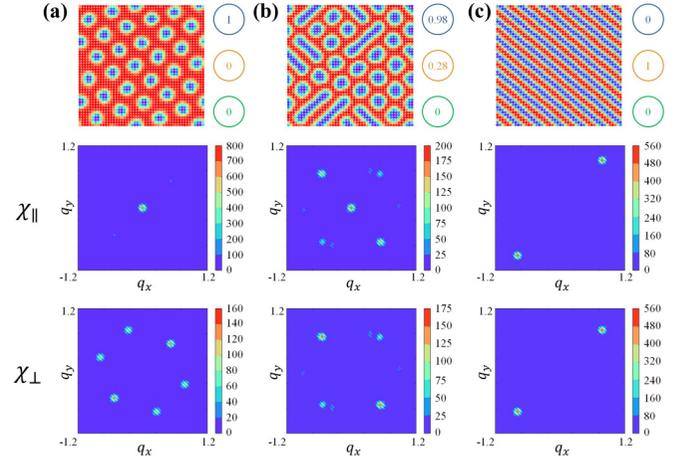


FIG. 12. Examples of pure skyrmion (a), mixed skyrmion-bimeron (b), and pure spiral (c) magnetic configurations obtained at low temperature ($T = 0.02J$) and corresponding spin-structure factors. Numbers in blue, orange, and green circles correspond to values of skyrmion, spiral, and FM outputs, respectively. Skyrmion numbers of these configurations are equal to 32, 28, and 0 from left to right.

excellent classification without requiring significant time for the calculations.

APPENDIX B: MACHINE LEARNING DETAILS

As an input of our FFN, we used the z components of the spins obtained from Monte Carlo simulations, and then the input and output of the hidden-layer neurons (Fig. 14) were

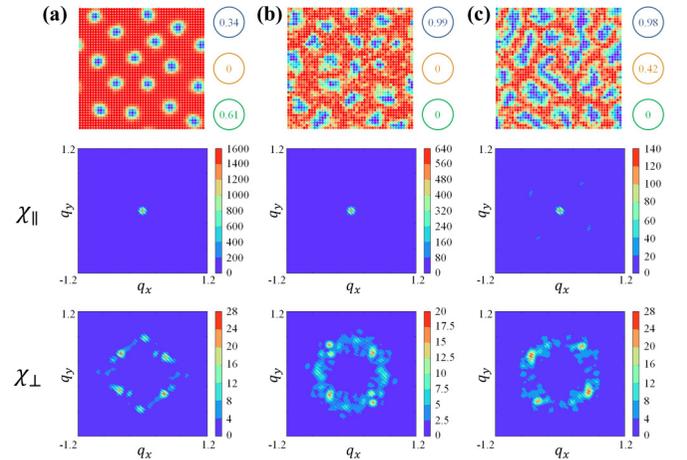


FIG. 13. Examples of nonperiodic skyrmions (a) obtained at low temperature ($T = 0.02J$), pure skyrmions (b), and mixed skyrmion-bimeron (c) magnetic configurations obtained at high temperature ($T = 0.4J$) and corresponding spin-structure factors. Numbers in blue, orange, and green circles correspond to values of skyrmion, spiral, and FM outputs, respectively. Skyrmion numbers of the presented configurations are equal to 15, 19, and 15 from left to right.

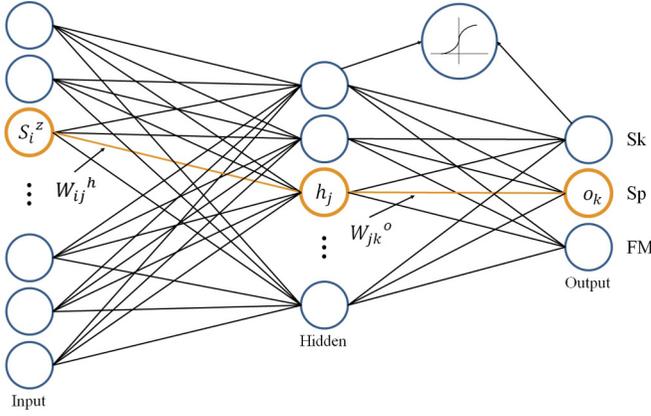


FIG. 14. Schematic representation of a constructed neural network with a single hidden layer. We used sigmoid as an activation function of hidden and output neurons. All the notations are described in the text.

calculated by the following equations:

$$h_j^{\text{inp}} = \frac{1}{\sum_{i=1}^N S_i^z} \sum_{i=1}^N S_i^z W_{ij}^h, \quad (\text{B1})$$

$$h_j^{\text{out}} = \text{sigmoid}(h_j^{\text{inp}}) = \frac{1}{1 + e^{-h_j^{\text{inp}}}}, \quad (\text{B2})$$

where S_i^z is the value of the i th input neuron, W_{ij}^h is the weight between the i th input neuron and the j th hidden neuron, and $N = L \times L$ is the number of input neurons. The normalization factor in the first equation is required in order to shift the input value into the range where $\text{sigmoid}(h_k^{\text{inp}}) \in [0; 1]$. This is very important, especially at the beginning of the learning process when we randomly initialize all the weights in the range $[-1; 1]$. Without normalization, we will obtain h_j^{out} equal to 1 or 0 because of the large number of input units. This will lead to the situation when the weights between the hidden and output neurons become the only parameters that affect the result. The values of the output layer neurons were calculated in a standard way by using the following

equation:

$$o_k = \text{sigmoid} \left(\sum_{j=1}^{N_h} h_j^{\text{out}} W_{jk}^o \right), \quad (\text{B3})$$

where N_h is the number of hidden neurons, and W_{jk}^o is the weight between the j th hidden neuron and the k th output neuron.

During the learning process, we randomly chose 10% of the training set for cross-validation to avoid overfitting, and we defined the stopping point where the error is less than the required value. The error function is given by

$$E(o^{\text{ideal}}, o^{\text{actual}}) = \frac{\sum_{k=1}^{N_o} (o_k^{\text{ideal}} - o_k^{\text{actual}})^2}{N_o}, \quad (\text{B4})$$

where N_o is the number of output neurons, o^{ideal} represents the training labels, and o^{actual} is the calculated values of the output neurons.

Due to the fact that we optimized our network through the backpropagation method [32] by means of the stochastic gradient descent with momentum, we used the following expression for new weights so as not to get stuck in local minima:

$$W^{(l)} = W^{(l-1)} + \Delta W^{(l)}, \quad (\text{B5})$$

$$\Delta W_{jk}^{o(l)} = \alpha \delta o_k h_j^{\text{out}} + \mu \Delta W_{jk}^{o(l-1)}, \quad (\text{B6})$$

$$\Delta W_{ij}^{h(l)} = \alpha \delta h_j^{\text{out}} S_i^z + \mu \Delta W_{ij}^{h(l-1)}, \quad (\text{B7})$$

where μ is the momentum, α is the learning rate, and l is the optimisation iteration index. These parameters can be chosen by trial and error (in our work, we used $\mu = 0.3$ and $\alpha = 0.8$). δo_k and δh_j^{out} are given by

$$\delta o_k = (o_k^{\text{ideal}} - o_k) o_k (1 - o_k), \quad (\text{B8})$$

$$\delta h_j^{\text{out}} = h_j^{\text{out}} (1 - h_j^{\text{out}}) \sum_{k=1}^{N_o} W_{jk}^o \delta o_k. \quad (\text{B9})$$

-
- [1] G. Carleo and M. Troyer, *Science* **355**, 602 (2017).
[2] Y. Nomura, A. S. Darmawan, Y. Yamaji, and M. Imada, *Phys. Rev. B* **96**, 205152 (2017).
[3] H. Saito and M. Kato, *J. Phys. Soc. Jpn.* **87**, 014001 (2018).
[4] P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst, *Sci. Rep.* **7**, 8823 (2017).
[5] K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, *Phys. Rev. X* **7**, 031038 (2017).
[6] L. Wang, *Phys. Rev. B* **94**, 195105 (2016).
[7] Y. Zhang, R. G. Melko, and E.-A. Kim, *Phys. Rev. B* **96**, 245119 (2017).
[8] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, *Nat. Phys.* **13**, 435 (2017).
[9] Y. Zhang and E.-A. Kim, *Phys. Rev. Lett.* **118**, 216401 (2017).
[10] K. Ch'ng, N. Vazquez, and E. Khatami, *Phys. Rev. E* **97**, 013306 (2018).
[11] J. Carrasquilla and R. G. Melko, *Nat. Phys.* **13**, 431 (2017).
[12] M. J. S. Beach, A. Golubeva, and R. G. Melko, *Phys. Rev. B* **97**, 045207 (2018).
[13] P. Suchsland and S. Wessel, *Phys. Rev. B* **97**, 174435 (2018).
[14] Y. LeCun, Y. Bengio, and G. Hinton, *Nature (London)* **521**, 436 (2015).
[15] A. Bogdanov and A. Hubert, *Phys. Status Solidi B* **186**, 527 (1994).
[16] A. Bogdanov and A. Hubert, *J. Magn. Magn. Mater.* **138**, 255 (1994).
[17] S. Mühlbauer, B. Binz, F. Jonietz, C. Pfleiderer, A. Rosch, A. Neubauer, R. Georgii, and P. Böni, *Science* **323**, 915 (2009).
[18] F. Jonietz, S. Mühlbauer, C. Pfleiderer, A. Neubauer, W. Münzer, A. Bauer, T. Adams, R. Georgii, P. Böni, R. A. Duine, K. Everschor, M. Garst, and A. Rosch, *Science* **330**, 1648 (2010).

- [19] A. Neubauer, C. Pfleiderer, B. Binz, A. Rosch, R. Ritz, P. G. Niklowitz, and P. Böni, *Phys. Rev. Lett.* **102**, 186602 (2009).
- [20] F. Zheng, H. Li, S. Wang, D. Song, C. Jin, W. Wei, A. Kovács, J. Zang, M. Tian, Y. Zhang, H. Du, and R. E. Dunin-Borkowski, *Phys. Rev. Lett.* **119**, 197205 (2017).
- [21] X. Z. Yu, N. Kanazawa, Y. Onose, K. Kimoto, W. Z. Zhang, S. Ishiwata, Y. Matsui, and Y. Tokura, *Nat. Mater.* **10**, 106 (2011).
- [22] M. Nagao, Y.-G. So, H. Yoshida, K. Yamaura, T. Nagai, T. Hara, A. Yamazaki, and K. Kimoto, *Phys. Rev. B* **92**, 140415 (2015).
- [23] S. Heinze, K. von Bergmann, M. Menzel, J. Brede, A. Kubetzka, R. Wiesendanger, G. Bihlmayer, and S. Blügel, *Nat. Phys.* **7**, 713 (2011).
- [24] N. Kanazawa, J.-H. Kim, D. S. Inosov, J. S. White, N. Egetenmeyer, J. L. Gavilano, S. Ishiwata, Y. Onose, T. Arima, B. Keimer, and Y. Tokura, *Phys. Rev. B* **86**, 134425 (2012).
- [25] X. Z. Yu, Y. Onose, N. Kanazawa, J. H. Park, J. H. Han, Y. Matsui, N. Nagaosa, and Y. Tokura, *Nature (London)* **465**, 901 (2010).
- [26] D. I. Badrtdinov, S. A. Nikolaev, M. I. Katsnelson, and V. V. Mazurenko, *Phys. Rev. B* **94**, 224418 (2016).
- [27] B. Berg and M. Lüscher, *Nucl. Phys. B* **190**, 412 (1981).
- [28] M. C. Ambrose and R. L. Stamps, *New J. Phys.* **15**, 053003 (2013).
- [29] S. El Hog, A. Bailly-Reyre, and H. T. Diep, *J. Magn. Magn. Mater.* **455**, 32 (2018).
- [30] I. A. Iakovlev, O. M. Sotnikov, and V. V. Mazurenko, *Phys. Rev. B* **97**, 184415 (2018).
- [31] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, *J. Machine Learning Res.* **12**, 2825 (2011).
- [32] C. MacLeod, *An Introduction to Practical Neural Networks and Genetic Algorithms for Engineers and Scientists* (Robert Gordon University, Aberdeen, Scotland, 2001).