applied sciences

MDPI

Article

Prediction of Critical Currents for a Diluted Square Lattice Using Artificial Neural Networks

Sajjad Ali Haider †, Syed Rameez Naqvi *, Tallha Akram and Muhammad Kamran †

Department of Electrical Engineering, COMSATS Institute of Information Technology, Wah Campus, Wah Cantonment 47040, Pakistan; sajjadali@ciitwah.edu.pk (S.A.H.); tallha@ciitwah.edu.pk (T.A.); muhammad_kamran@comsats.edu.pk (M.K.)

* Correspondence: rameeznaqvi@ciitwah.edu.pk; Tel.: +92-51-931-4382 (ext. 256)

† These authors contributed equally to this work.

Academic Editor: Faris Ali

Received: 30 December 2016; Accepted: 21 February 2017; Published: 2 March 2017

Abstract: Studying critical currents, critical temperatures, and critical fields carries substantial importance in the field of superconductivity. In this work, we study critical currents in the current–voltage characteristics of a diluted-square lattice on an Nb film. Our measurements are based on a commercially available Physical Properties Measurement System, which may prove time consuming and costly for repeated measurements for a wide range of parameters. We therefore propose a technique based on artificial neural networks to facilitate extrapolation of these curves for unforeseen values of temperature and magnetic fields. We demonstrate that our proposed algorithm predicts the curves with an immaculate precision and minimal overhead, which may as well be adopted for prediction in other types of regular and diluted lattices. In addition, we present a detailed comparison between three artificial neural networks architectures with respect to their prediction efficiency, computation time, and number of iterations to converge to an optimal solution.

Keywords: superconducting film; diluted square lattice; Shapiro steps; prediction; artificial neural networks

1. Introduction

Mixed state in superconductors is the sign of existence of vortices, which is the most interesting research area in low temperature physics. The vortices can be in the form of liquid, glassy, or crystalline phases. These vortex phases can be studied in high temperature superconductor systems and type II superconducting thin films with an array of dots/antidots. Over the last few decades, various properties of superconducting thin films with an array of artificial pinning centers have been explored [1–5]. Different geometrical structures [2,3,5–10] have been used in the composition of the array of dots or antidots. Previous works [10] showed that using a diluted array of antidots increases pinning effect along with energy conservation. In this work, an experimental setup based on a diluted square array of antidots is used to measure its current–voltage (IV) behavior.

It has been noted in previous works [11] that repeated transport measurements may become costly and cumbersome to obtain. As a result, there is a dire need to come up with a theoretical or a formal model that could approximate the IV curves of superconducting films, and thereby relieve the researchers from repeatedly measuring these physical properties. Artificial neural networks (ANNs) are among the most widely adopted techniques for modeling complex systems. The concept of ANN is derived from the actual working of a human neuron system [12], in which different neurons connect with each other through some network's coefficients (called weights). The working of ANNs is well addressed in literature [13]. ANNs learn and identify relationships between systems' parameters: this gives them an astonishing approximation or prediction capability. Other advantages associated with the use of ANNs include modeling non-linearity [14–17], fault tolerance, parallelism, robustness of the learning process, and ability to handle fuzzy information [18,19]. The advantage of using ANNs over other statistical methods—such as linear and nonlinear regression techniques—has been advocated on multiple occasions for various applications. For example, [20] made a comparison between the two techniques for the prediction of yarn tensile properties, [21] carried out the same comparison for Iran's annual electricity load, and [22] compared ANN with linear regression models for predicting hourly and daily diffuse fraction. All of them concluded that ANN was the better prediction approach.

Despite their tremendous features, ANNs have had a very limited application in the field of materials science [11,23,24]—let alone the prediction of IV curves. In [11], ANNs were used for the prediction of the current–voltage curves for a square array of nano-engineered periodic antidots. Diluted square arrays—which are formed by removing a quarter of the sites from the original square lattice—offer a larger interstitial area in comparison to the original square lattice. This means that a large number of interstitial vortices may easily be accommodated, leading to increased energy conservation; this is commonly referred to as *caging effect* [3]. In this work, we predict the IV curves for a diluted square array of antidots, and propose a framework based on ANN for extrapolating the IV behavior for a wide range of temperature and magnetic field values. In addition to this, we present a thorough comparison of three different ANN architectures trained with six different training algorithms for the framework. The comparison of training algorithms is based on prediction accuracy in terms of mean squared error (MSE), number of iterations needed to converge (epochs), and training time.

Our findings may be used as a benchmark in any followup work concerning the study of IV characteristics of any regular or diluted lattice, since we pinpoint the pros and cons of several architectures and training algorithms, and conclude on the most suitable options for our specific application. The rest of the paper is organized as follows. Section 2 summarizes the experimental details for acquiring the datasets. The choice of the architectures and training algorithms is given in Section 3. Simulation results and a comparison are given in Section 4, followed by the concluding remarks in Section 5.

2. Experimental Setup and Transport Measurements

2.1. Experimental Setup

For this work, we deposited a high-quality 60-nm-thick superconducting Nb film on a S_iO_2 substrate. Ultraviolet photolithography and reactive ion etching techniques were used to fabricate the microbridges for transport measurements, followed by standard lithography on a polymethyl-metacrylate (PMMA) resist layer to obtain the desired arrays. Magnetically enhanced reactive ion etching was used to transfer the patterns to the film. Our measurements were carried out using a commercially available Physical Properties Measurement System (PPMS) from Quantum Design. A scanning electron micrograph (SEM) (HITACHI, Tokyo, Japan) of the diluted square array is shown in Figure 1.

2.2. Measurements Using PPMS

The patterned superconducting film with a diluted square lattice had T_c of 8.646 K, which is smaller than T_c of the unpatterned film. For transport measurements, we placed the sample in liquid helium to help reduce the heating contact. Figure 2 shows the voltage measurements at different temperatures below T_c and zero applied fields. Change in the slope in these IV curves suggest the existence of three regions: one where the voltage is almost zero, which gradually increases in the second region, followed by the region in which the current increases linearly with voltage. While the Shapiro steps may be clearly observed in the second region within this temperature range, they continue to weaken until completely vanishing at higher temperatures; this happens mainly due to thermal fluctuations. Our observations are in agreement with those existing in literature [25].

For seven constant values of temperature (8.0, 8.1, 8.2, 8.3, 8.4, 8.5 and 8.6 K) and four constant values of magnetic fields (0, 100, 200 and 300 Oe), the measuring unit generated a dataset of 7 × 4 × 642 IV values. The next section describes the ANN methodology to be used for prediction.

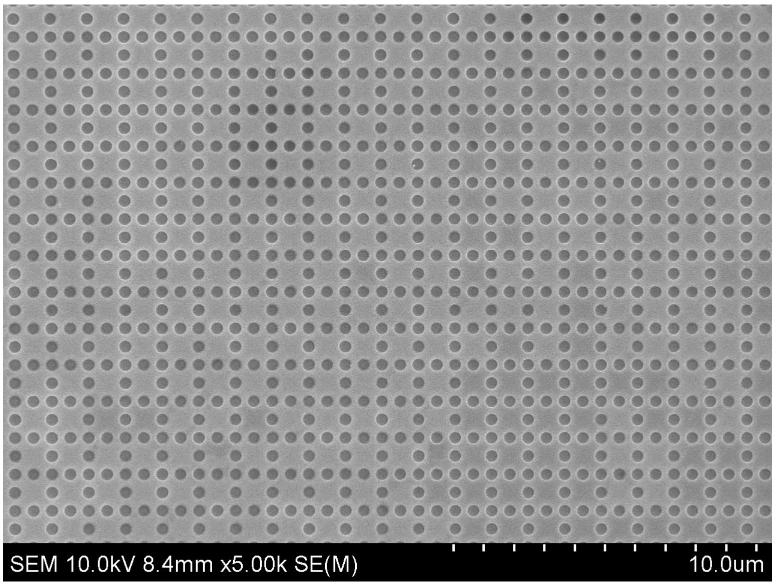


Figure 1. Scanning electron micrograph (SEM) of a superconducting Nb film with a diluted square array of holes.

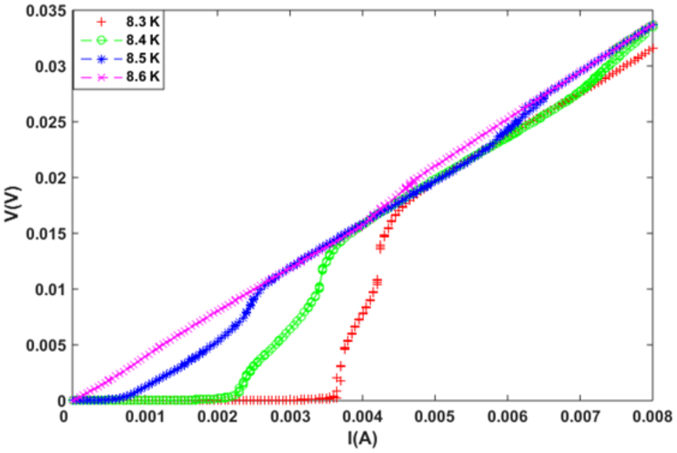


Figure 2. The current–voltage (IV) characteristic of a diluted square array of holes at different temperatures.

3. ANN Architectures and Training Algorithms

The topology of the ANN can be described in terms of a directed graph of nodes with a transfer function $\delta(\sum w_{ji}x_i + b_i)$, where x_i is called a state variable for each node i, w_{ji} is called a weight carrying some real value between two nodes i and j, b_i is a real-valued bias term, and δ is typically chosen to be a step or a linear function, and is called an *activation* function. This transfer function also represents the expected output of a system comprising just the input and output layers. It is well known that such a system is not capable of implementing many functions; it is usually necessary to incorporate a few hidden layers. Setti and Rao [26] showed that two hidden layers are usually an optimum choice capable of representing most of the desired functions. In this work, we also fix the number of hidden layers to two; however, we will vary the number of neurons in the hidden layers to have comparable prediction efficiencies.

The l^{th} output, y_l, of a network having two hidden layers is given as in Equation (1):

$$y_l = \delta_o(\sum_{k=0}^{P} w_{lk}^{y}(\delta_{H2}(\sum_{j=0}^{M} w_{kj}^{H2}(\delta_{H1}(\sum_{i=0}^{N} w_{ji}^{H1} x_i + b_i)) + b_j)) + b_k), \tag{1}$$

where:

- w_{lk}^{y} represents the weights from neuron k in the second hidden layer to the l^{th} output neurons.
- w_{kj}^{y} represents the weights from neuron j in the first hidden layer to neuron k in the second layer.
- w_{ji}^{H} represents the weights from neuron i in the input layer to the neuron j in the first hidden layer.
- x_i represents the i^{th} element in the input layer.
- b_i, b_j, and b_k represent the bias values for the hidden and output layers.
- δ_{H1}, δ_{H2} and δ_o are the activation functions: $H1$, $H2$, and o stand for the first and second hidden and the output layers, respectively.

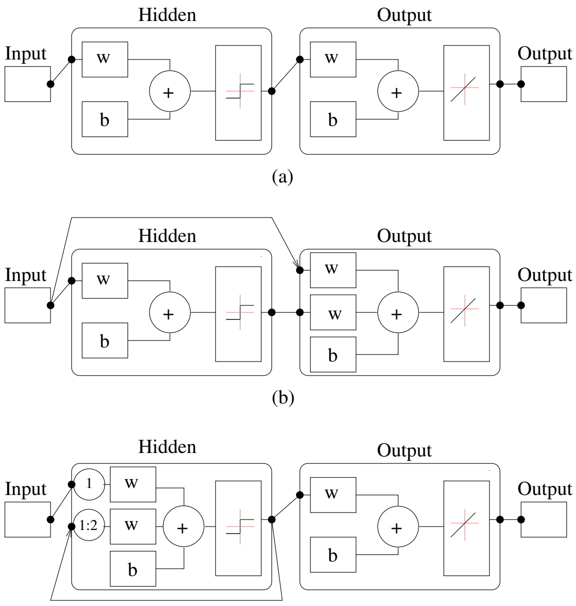
The primary objective of this analysis is to minimize a cost function given by Equation (2):

$$\phi = \frac{1}{2}\sum_{l=1}^{M}(y_l - d_l) = \frac{1}{2}\sum_{l=1}^{M} e_l^2 \tag{2}$$

The layers in the directed graphs are usually arranged in one of the three possible options, giving rise to three different ANN architectures. More specifically, the manner in which the layers are interconnected describes a particular architecture. In what follows, we briefly describe each architecture; comparing them in terms of prediction accuracy, epochs, and training time is the contribution of this work—this is covered in Section 4.

3.1. Feedforward Networks

Figure 3a shows an example of a feedforward network; for simplicity, the system is shown for one hidden layer. In such networks, each layer is only connected to its immediate neighbors: it takes input from the preceding layer, and generates output to the subsequent. In this way, mapping between input and output is achieved.



(a)

(b)

(c)

Figure 3. Various artificial neural network (ANN) architectures: (**a**) Feedforward; (**b**) Cascaded; (**c**) Layer-Recurrent.

3.2. Cascade-Forward Networks

Cascade-forward networks are a variant of feedforward networks. In this case, each layer is not just connected to the preceding node, but has a connection with the input as well; this is shown in Figure 3b. This will slightly modify Equation (1) to have $w_{kj}^{H2}x_i$ and $w_{lk}^{y}x_i$ in the second hidden and output layers.

3.3. Layer-Recurrent Networks

In layer-recurrent networks, each hidden layer has a recurrent connection with additional tap delays. The circles in the hidden layer of Figure 3c depict these delays. This feature allows such networks to have a dynamic response to time series data. In our work, we have used a tap delay of two in each hidden layer.

In all of our simulations, we have made use of *tan-sigmoid* (a hyperbolic tangent sigmoid function in the hidden layers) and *purelin* (a linear function in the output layer) as the activation functions. As far as training the network is concerned, several algorithms have been proposed in literature. The most widely adopted among those—which we also consider in this work—include Levenberg–Marquardt (LM) [27], Bayesian Regularization (BR) [28], Resilient Backpropogation (NR), Conjugate Gradient (CGF) [29–33], Quasi-Newton Backpropagation (BFGS) [34], and Variable Learning Rate Back Propagation with Momentum (GDX) [35].

4. Simulation Results

Our experimental setup generated a data set comprising several values of current and voltage for a wide range of magnetic field and temperature readings, a few entries of which are given in Table 1. Our methodology used 70 percent of these values as training data, while the rest was used to serve the validation and prediction purposes. We made use of MATLAB's toolbox called Neural Network to perform these simulations. As mentioned already, we implemented three architectures; each was trained using six algorithms, where each algorithm was run ten times with a different number of neurons in the hidden layers. This generated a total of 180 different models (6 algorithms × 10 configurations × 3 architectures). Table 2 summarizes the six algorithms and ten sets of number of neurons for each iteration of the algorithms, leading to sixty entries in total. Note that the entry No. of neurons $[x\ y]$ indicates x neurons in the first hidden layer, and y in the second.

Each one of the 180 models was trained five times, and the best results in terms of minimum MSE, epochs, and training time were saved. The obtained results showed MSE in the range of 6.5×10^{-6} to 2.7×10^{-8}. For the purpose of cross-validation, results from all of the models were compared with the actual data generated by the PPMS. Figures 4 and 5 show the actual and ANN-predicted IV curves for the diluted square array of antidots. The measurements—specifically for testing—were taken with the following parameters: temperature = 8.5 K, magnetic field = 300 Oe, and temperature = 8.4 K, magnetic field = 100 Oe, respectively, for the two figures, while current was varied from 0 to 8 mA in each case. Note that these values were deliberately not included in the training process of the ANN—they have been explicitly used for validation.

Figure 6 shows MSE in IV curves predicted by 180 models. Note that the horizontal axis corresponds to the sixty entries of Table 2. It may be observed in the figure that the feedforward network with [12 10] neurons in the hidden layers for *BR* as the training algorithm achieves the lowest MSE (i.e., 4.55×10^{-8}). However there are other results that have MSE in the same range, mostly trained with *BR*.

Table 1. Data set used for ANN training: (**a**) T = 8.0 K; (**b**) T = 8.1 K.

I	V	I	V	I	V	I	V
2.0×10^{-5}	5.0×10^{-7}	2.0×10^{-5}	1.5×10^{-6}	2.0×10^{-5}	-6.7×10^{-7}	2.0×10^{-5}	-6.3×10^{-7}
5.0×10^{-5}	5.1×10^{-7}	5.0×10^{-5}	4.0×10^{-7}	5.0×10^{-5}	3.5×10^{-7}	5.0×10^{-5}	3.8×10^{-7}
1.0×10^{-4}	5.2×10^{-7}	1.0×10^{-4}	4.0×10^{-7}	1.0×10^{-4}	3.5×10^{-7}	1.0×10^{-4}	3.7×10^{-7}
1.5×10^{-4}	5.1×10^{-7}	1.5×10^{-4}	4.1×10^{-7}	1.5×10^{-4}	3.4×10^{-7}	1.5×10^{-4}	3.4×10^{-7}
2.0×10^{-4}	5.1×10^{-7}	2.0×10^{-7}	3.9×10^{-7}	2.0×10^{-4}	3.6×10^{-7}	2.0×10^{-4}	3.5×10^{-7}
2.5×10^{-4}	5.1×10^{-7}	2.5×10^{-4}	4.1×10^{-7}	2.5×10^{-4}	3.5×10^{-7}	2.5×10^{-4}	3.8×10^{-7}
0H (Oe)		100H (Oe)		200H (Oe)		300H (Oe)	
(**a**)							
5.0×10^{-5}	5.1×10^{-7}	5.0×10^{-5}	4.0×10^{-7}	5.0×10^{-5}	3.8×10^{-7}	5.0×10^{-5}	4.4×10^{-7}
1.0×10^{-4}	5.0×10^{-7}	1.0×10^{-4}	3.9×10^{-7}	1.0×10^{-4}	3.6×10^{-7}	1.0×10^{-4}	4.4×10^{-7}
1.5×10^{-4}	5.0×10^{-7}	1.5×10^{-4}	3.9×10^{-7}	1.5×10^{-4}	3.7×10^{-7}	1.5×10^{-4}	4.3×10^{-7}
2.0×10^{-4}	5.0×10^{-7}	2.0×10^{-4}	3.9×10^{-7}	2.0×10^{-4}	3.7×10^{-7}	2.0×10^{-4}	4.2×10^{-7}
2.5×10^{-4}	5.1×10^{-7}	2.5×10^{-4}	4.1×10^{-7}	2.5×10^{-4}	3.6×10^{-7}	2.5×10^{-4}	4.1×10^{-7}
0H (Oe)		100H (Oe)		200H (Oe)		300H (Oe)	
(**b**)							

Table 2. Sixty different ANN configurations for each architecture. BFGS: Quasi-Newton Backpropagation; BR: Bayesian Regularization; CGF: Conjugate Gradient; GDX: Variable Learning Rate Back Propagation with Momentum; LM: Levenberg–Marquardt; NR: Resilient Backpropogation.

Model No.	No. of Neurons	Train. Algo.	Model No.	No. of Neurons	Train. Algo.	Model No.	No. of Neurons	Train. Algo.
1.	[5 2]	LM	21.	[5 2]	CGF	41.	[5 2]	NR
2.	[10 8]	LM	22.	[10 8]	CGF	42.	[10 8]	NR
3.	[12 6]	LM	23.	[12 6]	CGF	43.	[12 6]	NR
4.	[15 6]	LM	24.	[15 6]	CGF	44.	[15 6]	NR
5.	[18 10]	LM	25.	[18 10]	CGF	45.	[18 10]	NR
6.	[11 5]	LM	26.	[11 5]	CGF	46.	[11 5]	NR
7.	[12 10]	LM	27.	[12 10]	CGF	47.	[12 10]	NR
8.	[14 7]	LM	28.	[14 7]	CGF	48.	[14 7]	NR
9.	[10 5]	LM	29.	[10 5]	CGF	49.	[10 5]	NR
10.	[8 4]	LM	30.	[8 4]	CGF	50.	[8 4]	NR
11.	[5 2]	BR	31.	[5 2]	BFGS	51.	[5 2]	GDX
12.	[10 8]	BR	32.	[10 8]	BFGS	52.	[10 8]	GDX
13.	[12 6]	BR	33.	[12 6]	BFGS	53.	[12 6]	GDX
14.	[15 6]	BR	34.	[15 6]	BFGS	54.	[15 6]	GDX
15.	[18 10]	BR	35.	[18 10]	BFGS	55.	[18 10]	GDX
16.	[11 5]	BR	36.	[11 5]	BFGS	57.	[12 10]	GDX
18.	[14 7]	BR	38.	[14 7]	BFGS	58.	[14 7]	GDX
19.	[10 5]	BR	39.	[10 5]	BFGS	59.	[10 5]	GDX
20.	[8 4]	BR	40.	[8 4]	BFGS	60.	[8 4]	GDX

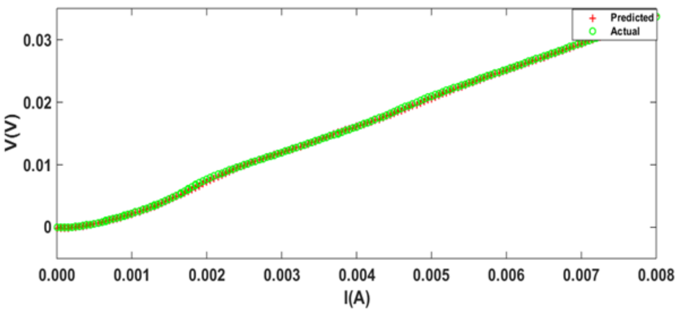


Figure 4. Actual vs. predicted IV curves for temperature = 8.5 K, magnetic field = 300 Oe.

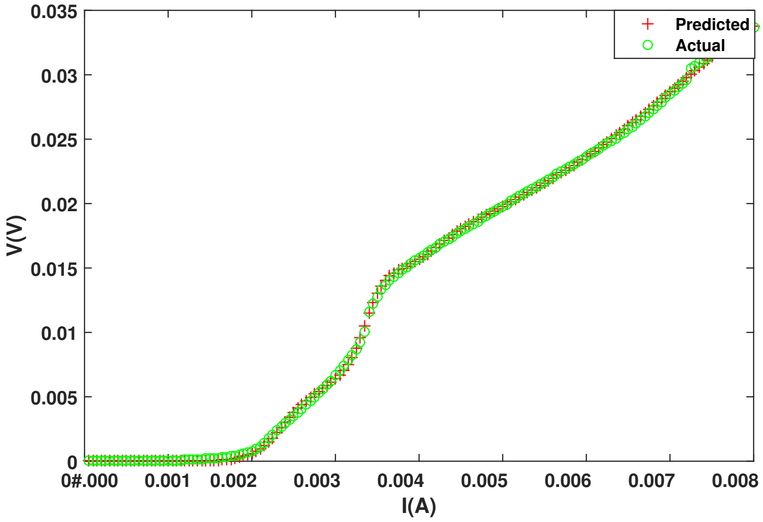


Figure 5. Actual vs. predicted IV curves for temperature = 8.4 K, magnetic field = 100 Oe.

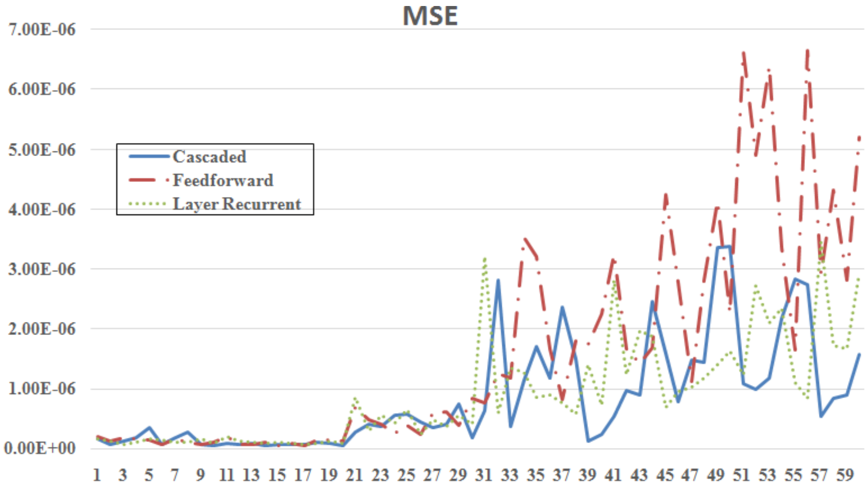


Figure 6. Mean squared error (MSE) in 60 selected models for each architecture.

Figures 7 and 8 show the results with respect to number of iterations (epochs) and training time, respectively. It may be observed that while the cascaded network with [5 2] neurons and *CGF* as the training algorithm gives best results with respect to smallest number of epochs (i.e., 9), the feedforward network with [8 4] neurons and *NR* as the training algorithm gives the best results with respect to minimum training time (i.e., 0.109 s). These two parameters—epochs and training time—are specifically more useful in studies requiring real-time training and prediction than MSE, since smaller delays would yield faster systems. Note that each of the Figures 6–8 corresponds to a temperature of 8.5 K and a magnetic field of 300 Oe.

We have summarized the best results for each measured parameter in Table 3. It can be noted that although the feedforward network with *BR* gives minimum MSE, it takes a large number of iterations and training time to converge. Naturally, it will not be the optimum choice in real-time systems. Similarly, the other results show that the systems that converge faster yield large MSE—making them unsuitable in systems requiring high precision.

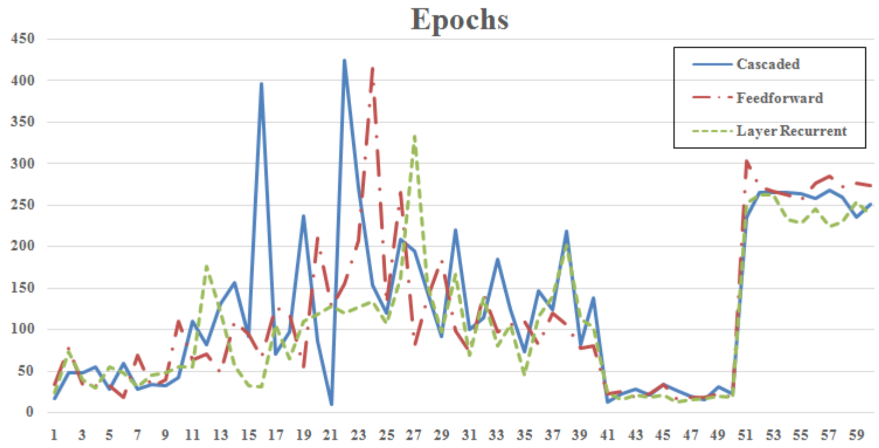


Figure 7. Number of iterations (epochs) taken by each model to converge.

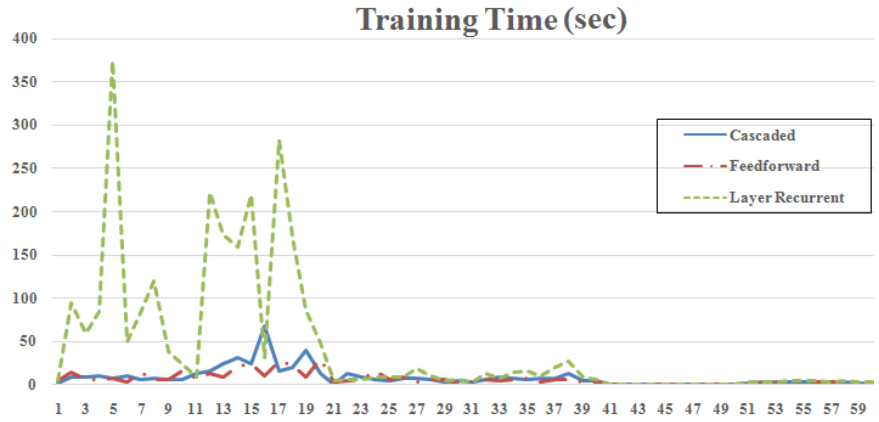


Figure 8. Training time taken by each model to converge.

Table 3. Summary of the best results.

Parameter	Network	No. of Neurons	Algo.	MSE	Epochs	Training Time (s)
Minimum MSE	Feedforward	[12 10]	BR	4.55×10^{-8}	126	26.037
Minimum Epochs	Cascaded	[5 2]	CGF	2.80×10^{-7}	9	0.297
Minimum Time	Feedforward	[8 4]	NR	2.27×10^{-6}	17	0.109

5. Conclusions

We have proposed a method based on ANN for measuring the IV curves in a diluted square array of antidots on an Nb film at different applied fields and temperatures. Because of their exceptional approximation capability, ANNs have recently been recommended for the prediction of IV curves in superconducting films. Their increasing role in this field motivated us to present a thorough analysis of three different architectures—namely, feedforward, cascaded, and layer-recurrent networks—which were trained using six different learning algorithms. Each algorithm was executed for ten different configurations of number of neurons in the hidden layers, resulting in a total of sixty ANN models for each architecture. Our results, based on MATLAB simulations, suggested that feedforward networks trained with *BR* manage to achieve the lowest MSE, but take a lot of time to converge, while those converging faster (in terms of number of iterations and training time) yield larger MSEs.

Since we pinpoint the pros and cons of various architectures with various possible configurations, our proposed framework may be used as a benchmark in all relevant works utilizing ANN in the prediction of IV curves. It is widely known that each geometry of arrays of antidots exhibits different current–voltage curves, leading to vastly varying critical currents and critical temperatures.

We sincerely believe it is prudent to study the effectiveness of our approach for each of those geometries, and this we have left as our prospective followup work.

Author Contributions: Muhammad Kamran performed the transport measurements on the Physical Properties Measurement System. Sajjad Ali Haider was responsible for the research work related to Artifical Neural Networks; he performed all the simulations. Tallha Akram is an expert of algorithms and formal methods; in this work he was responsible to make a choice of the algorithms and architectures, and reviewed the simulation results. Syed Rameez Naqvi conceived the entire concept, analyzed the obtained data from the Physical Properties Measurement System, and did all the write-up.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Baert, M.; Metlushko, V.; Jonckheere, R.; Moshchalkov, V.; Bruynseraede, Y. Composite flux-line lattices stabilized in superconducting films by a regular array of artificial defects. *Phys. Rev. Lett.* **1995**, *74*, 3269.
2. Martin, J.I.; Vélez, M.; Hoffmann, A.; Schuller, I.K.; Vicent, J. Artificially induced reconfiguration of the vortex lattice by arrays of magnetic dots. *Phys. Rev. Lett.* **1999**, *83*, 1022.
3. Latimer, M.; Berdiyorov, G.; Xiao, Z.; Kwok, W.; Peeters, F. Vortex interaction enhanced saturation number and caging effect in a superconducting film with a honeycomb array of nanoscale holes. *Phys. Rev. B* **2012**, *85*, 012505.
4. Villegas, J.; Savel'ev, S.; Nori, F.; Gonzalez, E.; Anguita, J.; Garcia, R.; Vicent, J. A superconducting reversible rectifier that controls the motion of magnetic flux quanta. *Science* **2003**, *302*, 1188–1191.
5. Kamran, M.; Naqvi, S.R.; Kiani, F.; Basit, A.; Wazir, Z.; He, S.K.; Zhao, S.P.; Qiu, X.G. Absence of Reconfiguration for Extreme Periods of Rectangular Array of Holes. *J. Supercond. Novel Magn.* **2015**, *28*, 3311–3315.
6. Martin, J.I.; Vélez, M.; Nogues, J.; Schuller, I.K. Flux pinning in a superconductor by an array of submicrometer magnetic dots. *Phys. Rev. Lett.* **1997**, *79*, 1929.
7. Jaccard, Y.; Martin, J.; Cyrille, M.C.; Vélez, M.; Vicent, J.; Schuller, I.K. Magnetic pinning of the vortex lattice by arrays of submicrometric dots. *Phys. Rev. B* **1998**, *58*, 8232.
8. Cuppens, J.; Ataklti, G.; Gillijns, W.; Van de Vondel, J.; Moshchalkov, V.; Silhanek, A. Vortex dynamics in a superconducting film with a kagomé and a honeycomb pinning landscape. *J. Supercond. Novel Magn.* **2011**, *24*, 7–11.
9. De Lara, D.P.; Alija, A.; Gonzalez, E.; Velez, M.; Martin, J.; Vicent, J.L. Vortex ratchet reversal at fractional matching fields in kagomélike array with symmetric pinning centers. *Phys. Rev. B* **2010**, *82*, 174503.
10. He, S.; Zhang, W.; Liu, H.; Xue, G.; Li, B.; Xiao, H.; Wen, Z.; Han, X.; Zhao, S.; Gu, C.; et al. Wire network behavior in superconducting Nb films with diluted triangular arrays of holes. *J. Phys. Condens. Matter* **2012**, *24*, 155702.
11. Kamran, M.; Haider, S.; Akram, T.; Naqvi, S.; He, S. Prediction of IV curves for a superconducting thin film using artificial neural networks. *Superlattices Microstruct.* **2016**, *95*, 88–94.
12. Guojin, C.; Miaofen, Z.; Honghao, Y.; Yan, L. Application of Neural Networks in Image Definition Recognition. In Proceedings of the IEEE International Conference on Signal Processing and Communications, Dubai, UAE, 24–27 November 2007; pp. 1207–1210.
13. Güçlü, U.; van Gerven, M.A.J. Deep Neural Networks Reveal a Gradient in the Complexity of Neural Representations across the Ventral Stream. *J. Neurosci.* **2015**, *35*, 10005–10014.
14. Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.* **1989**, *2*, 303–314.
15. Hornik, K. Approximation capabilities of multilayer feedforward networks. *Neural Netw.* **1991**, *4*, 251–257.
16. Hornik, K. Some new results on neural network approximation. *Neural Netw.* **1993**, *6*, 1069–1072.
17. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366.
18. Bielecki, A.; Ombach, J. Dynamical properties of a perceptron learning process: Structural stability under numerics and shadowing. *J. Nonlinear Sci.* **2011**, *21*, 579–593.
19. Meireles, M.; Almeida, P.; Simoes, M. A comprehensive review for industrial applicability of artificial neural networks. *IEEE Trans. Ind. Electr.* **2003**, *50*, 585–601.

20. Üreyen, M.E.; Gürkan, P. Comparison of artificial neural network and linear regression models for prediction of ring spun yarn properties. I. Prediction of yarn tensile properties. *Fibers Polym.* **2008**, *9*, 87–91.
21. Ghanbari, A.; Naghavi, A.; Ghaderi, S.; Sabaghian, M. Artificial Neural Networks and regression approaches comparison for forecasting Iran's annual electricity load. In Proceedings of the International Conference on Power Engineering, Energy and Electrical Drives, Lisbon, Portugal, 18–20 March 2009, pp. 675–679.
22. Elminir, H.K.; Azzam, Y.A.; Younes, F.I. Prediction of hourly and daily diffuse fraction using neural network, as compared to linear regression models. *Energy* **2007**, *32*, 1513–1523.
23. Quan, G.z.; Pan, J.; Wang, X. Prediction of the Hot Compressive Deformation Behavior for Superalloy Nimonic 80A by BP-ANN Model. *Appl. Sci.* **2016**, *6*, 66.
24. Zhao, M.; Li, Z.; He, W. Classifying Four Carbon Fiber Fabrics via Machine Learning: A Comparative Study Using ANNs and SVM. *Appl. Sci.* **2016**, *6*, 209.
25. Odagawa, A.; Sakai, M.; Adachi, H.; Setsune, K.; Hirao, T.; Yoshida, K. Observation of intrinsic Josephson junction properties on (Bi, Pb) SrCaCuO thin films. *Jpn. J. Appl. Phys.* **1997**, *36*, L21.
26. Setti, S.G.; Rao, R. Artificial neural network approach for prediction of stress–strain curve of near β titanium alloy. *Rare Met.* **2014**, *33*, 249–257.
27. Marquardt, D.W. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Ind. Appl. Math.* **1963**, *11*, 431–441.
28. MacKay, D.J. A practical Bayesian framework for backpropagation networks. *Neural Comput.* **1992**, *4*, 448–472.
29. Beale, E. A derivation of conjugate gradients. In *Numerical Methods for Nonlinear Optimization*; Academic Press Inc.: Cambridge, MA, USA, 1972; pp. 39–43.
30. Gill, P.E.; Murray, W.; Wright, M.H. *Practical Optimization*; Emerald Group Publishing Limited: Bingley, UK, 1981.
31. Hestenes, M.R. *Conjugate Direction Methods in Optimization*; Springer Science & Business Media: New York, NY, USA, 2012; Volume 12.
32. Johansson, E.M.; Dowla, F.U.; Goodman, D.M. Backpropagation learning for multilayer feed-forward neural networks using the conjugate gradient method. *Int. J. Neural Syst.* **1991**, *2*, 291–301.
33. Powell, M.J.D. Restart procedures for the conjugate gradient method. *Math. Program.* **1977**, *12*, 241–254.
34. Battiti, R.; Masulli, F. BFGS optimization for faster and automated supervised learning. In *International Neural Network Conference*; Springer: Berlin/Heidelberg, Germany, 1990; pp. 757–760.
35. Beale, M.H.; Hagan, M.T.; Demuth, H.B. Neural network toolbox 7. In *Matlab's User Guide*; Mathworks: Natick, MA, USA, 2010.