INSTITUT DE FRANCE
Académie des sciences

Comptes Rendus
Mécanique

Rodayna Hmede, Frédéric Chapelle and Yuri Lapusta

Modeling the butterfly behavior of SMA actuators using neural networks

Volume 350 (2022), p. 143-157

Published online: 28 April 2022

https://doi.org/10.5802/crmeca.108

Les Comptes Rendus. Mécanique *sont membres du*
Centre Mersenne pour l'édition scientifique ouverte
www.centre-mersenne.org
e-ISSN : 1873-7234

Comptes Rendus
Mécanique
2022, 350, p. 143-157
https://doi.org/10.5802/crmeca.108

Short paper / *Note*

Modeling the butterfly behavior of SMA actuators using neural networks

Rodayna Hmede [*, a], **Frédéric Chapelle** [a] **and Yuri Lapusta** [a]

[a] Université Clermont Auvergne, Clermont Auvergne INP, CNRS, Institut Pascal, F-63000 Clermont-Ferrand, France

E-mails: rodayna.HMEDE@uca.fr (R. Hmede), frederic.chapelle@sigma-clermont.fr (F. Chapelle), yuri.lapusta@sigma-clermont.fr (Y. Lapusta)

Abstract. Shape memory alloy (SMA) actuators are an important application of smart materials for robotics. However, the nonlinear behavior of SMA leads to difficulties in real-time simulations using numerical methods. Artificial Intelligence can be used to bypass this problem. In this paper, we study several neural networks (NNs) to model the superelastic or pseudo-elasticity effect (SEE) as well as the shape memory effect (SME) used in SMA. Focusing on antagonistic actuating, we first model a single wire to train the best NN with the proper characteristics that fit the behavior of SEE. Then, we model the SME of two linear antagonistic SMA wires used as an actuator. In both systems, single and antagonistic wires, we train the networks to obtain the stress–strain diagrams representing the behavior. The network type and training algorithm are key factors and are evaluated depending on the RMSE values. As a result, we find that the long short-term memory NN, used with a regression layer on standardized data sets, models the butterfly-shaped behavior of the actuator system with less RMSE value.

Keywords. Neural networks, Shape memory alloys, Shape memory effect, Superelastic effect, Hysteresis behavior, Antagonistic systems.

Manuscript received 29th January 2022, revised 20th February 2022, accepted 5th April 2022.

1. Introduction

Shape memory alloy (SMA) actuators are active materials used in numerous areas such as medical sciences, aerospace, civil, and robotics engineering. They can be integrated into various types of sensors [1], actuators [2], self-centering, and damping devices [3]. The type of application influences the searched form of the SMA and its related properties.

In robotic structures, the SMA actuators can be used for crawling, jumping, flowering, swimming, walking, and for medical and bio-mimetic hands. For example, SMA actuators have been extensively studied in soft robots for surgery applications such as endoscopy [4]. The success of SMA actuators is due to their ability to undergo complex motions depending on their specific characteristics, such as large deformation and memory. The last two specificities characterize

* Corresponding author.

SMAs: SEE [5] and SME [6]. To describe them, we need to understand the two main phases of reversible transformation of SMA, i.e., martensitic and austenitic. These phases depend on the temperature and stress factors.

SEE is a reversible interaction between the strain and stress or, in other words, the deformation of SMA upon loading and unloading. The specialty of SMA deformation during loading at constant temperature is that the SMA transfers elastically from austenitic to martensitic at critical stress. On the other hand, upon unloading, SMA reverses to the austenitic phase in which it recovers from the earlier large deformations, forming its parent phase. Many studies focused on the properties of the SEE: for a thin sheet of SMA [7], for CuAlNi using traditional methods with thermodynamic arguments, for polycrystalline specimen using the mathematical "Preisach Model" [8]. The studies that focused on the simulation of the stress–strain diagram used numeric models [9], such as the self-consistent model [10] and technical or machining response.

SMA can furthermore undergo hysteretic transformation occurring one-way and two-way due to heating or cooling upon other physical properties such as stress. Thermal transformation occurs between the two phases, martensitic and austenitic, at specific temperatures related to the type of SMA. Upon heating the SMA over the austenite temperature, it recovers its parent or memory phase. Upon cooling, it begins to transform to the martensitic phase at the martensitic start temperature.

An SMA can be actuated using SME and spring bias or other antagonistic systems for the return deformation. Antagonistic systems made up of two SMAs got close attention in the robotic field. Some studies tried to model their properties as a stress–strain diagram or temperature–stress–strain diagram and their control as mathematical formalisms. To our knowledge, there is no explicit simulation of stress–strain diagrams using artificial intelligence tools, especially neural networks (NNs). However, with differences from NNs, the neuro-fuzzy algorithm showed the possibility to briefly and accurately NN-model SMA's seismic performances [11].

Our objective in this paper is to build a sufficiently fast and accurate predictive model based on NNs that could be used in future works in the preliminary design, optimization, and control of SMA actuators, taking into account the actuating system's internal state of load without the need for sensors. For that, we need to know the conditions on the best NNs, learning parameters, nonlinear learning base construction, and the precision level expected for the antagonistic actuation system. In the following sections, we give the protocol of our experimentations on the modeling by NNs, before applying it on a single wire firstly, then on an antagonistic actuator case.

2. Methodological background

Neural networks, as part of the machine learning domain, achieve high performance in engineering applications, including robotics [12–17] and material science [18].

In our paper, we apply them for modeling SMA actuators. Different characteristics specify different types of NNs: weights of inputs showing the importance of each input neuron, number of hidden layers (one for a shallow NN), activation and transformation functions (such as sigmoid, linear, step, rectified functions), that decide the threshold of the neurons and the calculation of the output response.

NN tasks can be divided into four categories depending on the type of the studied problem: fitting, clustering, time series, and pattern recognition. Each category has its specific network architectures that depend on the organization of the layers, the used function of activation and transformation, and the training algorithm applied on a learning base.

The fitting and pattern categories usually rely on shallow feedforward NN with two layers and different activation functions. The clustering category usually implies self-organizing mapping

with one-hidden-layer NNs. The time-series category for nonlinear prediction, especially for real-time simulation, can use the NNs depending on the source of the non-linearity.

Shallow NNs can use feedforward or cascade forward propagation. Their one or two hidden layers specify them. There is no huge difference between feedforward and cascade forward NNs. The cascade NN is specified by connecting each layer to all the next layers. Shallow NNs typical arguments are the number of inputs and hidden layers, bias (constant added to the weighted inputs), weights of connections between layers, activation functions of neurons, and output layer. The number of input connectors is the number-of-layers-by-number-of-inputs represented as a Boolean matrix. For layer connectors, it is the number of-layers-by-number-of-layers. A 1-by-number of layers Boolean vector represents the output connectors. On the other hand, real-time simulations with a high sequence of data and non-linear behavior provided us with other categories of NNs, especially the long short-term memory (LSTM) NN. LSTM functionality relies on a network with the ability to learn long-term dependencies between time steps in time series and sequence data. It can predict long patterns with unidentified lengths and thus is known for its memory capabilities.

By these categories, NNs can provide interesting solutions to modeling and predicting the characteristics of smart materials. Concerning SMA, NN was used in various applications to model the hysteresis behavior of SMA [12], to train NNs' parameters using particle swarm optimization technique (PSO) [19], and to predict the reduction factor of SMA as a function of the reinforcement ratio and the reinforcement modulus of elasticity of SMA [20], using shallow NNs with constraints [21], and feed-forward and back-propagation NNs [22].

However, the usage of SMA as raw material referring to its central "superelastic" and "shape memory" stress–strain characteristics have not been modeled by NN till date, to the best of our knowledge. However, the real-time simulation of the SMA stress–strain diagrams is necessary to understand the behavior with rapid and precise responses.

We evaluate in the following the possibility for a NN to model shape memory systems and their complexity. We start modeling with a single SMA wire to choose the best NN among the four stated NNs: LSTM, fitting NN (shallow NN with one hidden layer), feedforward NN, cascade forward NN, depending on the experimental stress–strain data provided by an Ansys platform during the preparation stage (Section 4.1). After that, we model an SMA actuator of higher complexity from its experimental stress–strain diagram. Here, the difficulty is that the modeling can no longer be considered as an assembly of lines with given slopes (Section 5).

3. Single SMA wire

3.1. *Methodology*

3.1.1. *Generating the learning base*

We use the finite element method (Ansys) to model SMA's behavior [23]. Solid–solid phase transformations take place and form the stress–strain diagram (Figure 1) [24]. The superelastic behavior is presented in the figure, while the loading cycle (increasing stress) and unloading cycle (decreasing stress) involve a high strain variation. Therefore, SMA undergoes large deformations without showing residual strains. There are solid-phase transformations between austenitic and martensitic phases during each cycle.

After preliminary training tests of the NNs with a whole non-regular data set, we found it necessary to discretize the data set by regular variation. The data set is split into two cycles: upper part and lower part. Each cycle is split into three continuous parts: the two increasing or two decreasing parts and the constant part (continuous lines presented by numbers in Figure 1).

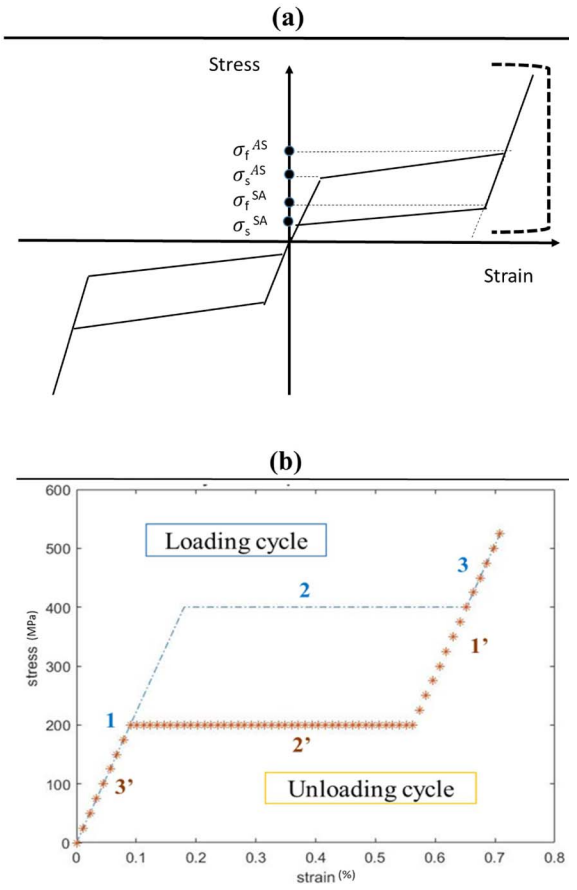

Figure 1. Stress–strain diagrams: (a) of the learning base model and (b) of the learning base implemented in Matlab. (a) σ_f^{AS} is the martensite finish stress, σ_s^{AS} the martensite start stress, σ_f^{SA} the austenite finish stress, and σ_s^{SA} the austenite start stress. (b) The curve shows the two separate loading (blue dashed lines) and unloading (red stars) cycles.

Therefore, as a first step, we study each cycle separately. Therefore, in each algorithm, the data is separated into three referring lists. We created algorithms for the four different NNs that use the input data (stress–strain data to plot the stress–strain diagram).

3.1.2. *Training the neural networks*

We used Matlab to study the performance on NNs on the superelastic behavior of SMA (Figure 1). Treatment of data set, neurons' weights, algorithm, and network type are the critical factors of training NNs to perform excellently. The tested and trained NNs are feedforward NN, cascade forward, fitting NN, and LSTM NN.

The data set (learning base) is prepared with 64 values of stress vs. 64 of strain, with a step variation of 0.01 for strain and 25 MPa for stress. We studied the feedforward NN, cascade forward, fitting NN, and LSTM NN training on the whole data set as an input to see if they could succeed in testing the whole data one at a time. Afterwards, we found the need to study the data set of the lower and upper parts separately. Moreover, we have studied the data set by splitting

the continuous parts of each cycle. For performance, we also introduced standardizing method using an equation of sigmoid and mean values of the data set for the NNs having the best results. For that, we produced four algorithms: standardizing each regular data of upper part, standardizing data of lower part, non-standardizing regular data of upper part, and non-standardizing regular data of lower part.

3.2. *Results and analysis*

In the beginning, we use the whole data set for the training (the two cycles together). Then, we split the cycles and separate the continuous parts. Finally, we standardize the data set for the training of the successful NNs. In each interval, we vary the number of hidden layers between 2 and 10, and the percentage of data used for training is 70% and 30% for testing for feedforward NN, cascade forward NN, and fitting NN. For LSTM NN, we vary the epoch number (complete iterations) between 1000 and 300 and the hidden units between 200 and 800. The previously specified parameters and ways of treating the training data depend on the linearity of the continuous part and the step of discretization in the data. They are chosen empirically; the lowest RMSE value (root mean square error) proves the best parameters and data treatment.

For cascade forward NN, feedforward NN, and fitting NN, splitting the two cycles is a condition of training success, with five hidden layers and Levenberg–Marquardt backpropagation as training function. However, splitting the continuous parts enhances the RMSE value (Figures 2–4). A highlight of the result here is that the loading cycle training has higher values of RMSE than that of non-loading. This observation results from a few outside points in the increasing/decreasing parts, because when the line begins from a high level (non-zero), the training is not entirely stable.

For LSTM NN, splitting the cycles and splitting the continuous parts are conditions for training success. The best training occurs with 500 hidden units and 500 epoch numbers for constant parts, 1000 hidden units, and 300 epoch numbers for increasing and decreasing parts.

The fitting NN successfully predicts the non-loading and loading cycles using separated cycles with and without splitting them into three parts. Here, the feedforward NN succeeds in the split case. The fitting NN has generally better RMSE values than the feedforward NN. The RMSE values of fitting NN are around 0.03 and 0.2, and that of feedforward NN is around 0.22 and 1.9. However, the cascade forward NNs' training shows success in the split case only, predicting the unloading and loading cycles, with separation of each of their three parts. Therefore, fitting NN has the best results among the feedforward and cascade forward NNs.

In addition, the LSTM NN succeeds in predicting the stress–strain values by splitting the continuous parts of the separated loading and unloading cycles with low RMSE values around 0.1 and 0.5. Therefore, LSTM has comparable results to that of the fitting NN.

To enhance the results after comparing the different NNs, we use the standardization of the data set to train again the LSTM and fitting NNs (see Figures 4 and 5). Thus, LSTM NN, by standardizing the training data (and with splitting the continuous parts) achieves the lowest RMSE of around 0.06 and 0.08, while that of fitting NN are around 0.2 and 3. The highlight here is that the results of fitting NN with standardizing data are higher than those without standardizing the data. As we can see in Figure 5, this is because of the lower ability to follow the non-constant continuous parts of the unloading cycle.

Finally, LSTM can model the nonlinear SMA in both cycles, in which standardization has better results than non-standardization. However, the computational time of training the LSTM NN is much longer than training the other three NNs. Its computational time is 10 min and for others NNs, around 1 min.

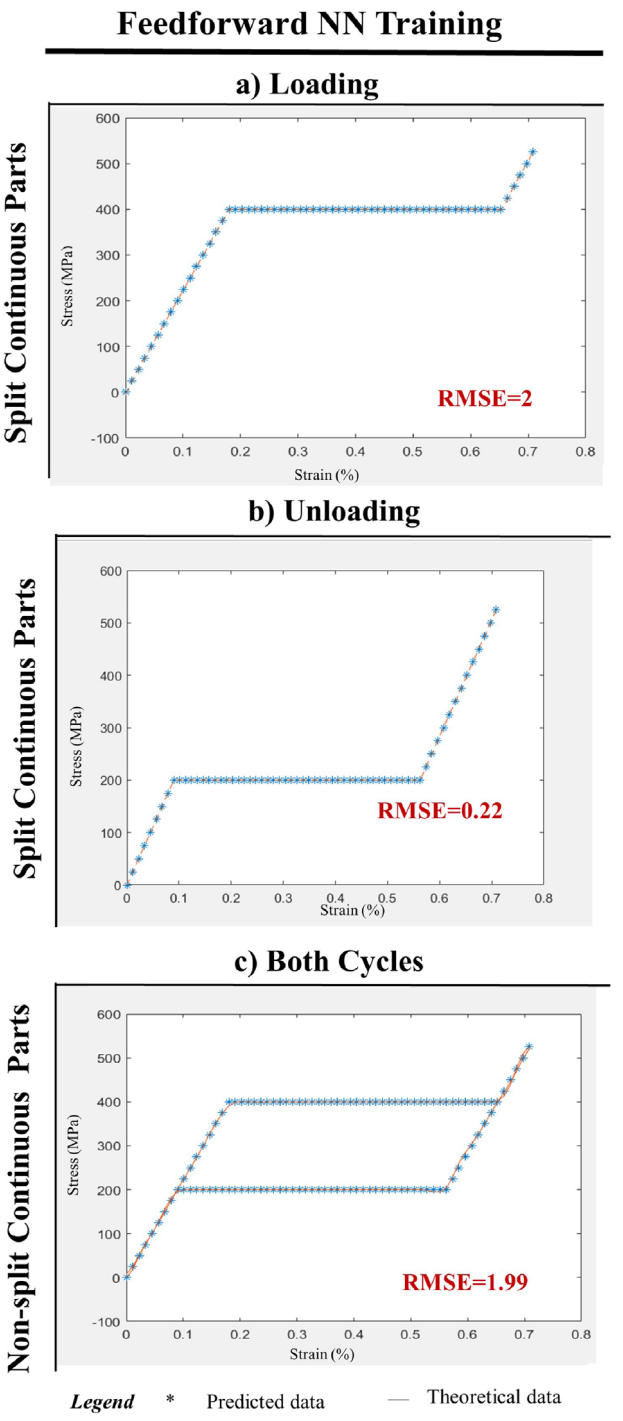

Figure 2. Results of the feedforward NN for a single wire. (a) Training the loading cycle with splitting the continuous parts. (b) Training the unloading cycle with splitting the continuous parts. (c) Training the separated loading and unloading cycles without splitting the continuous parts.

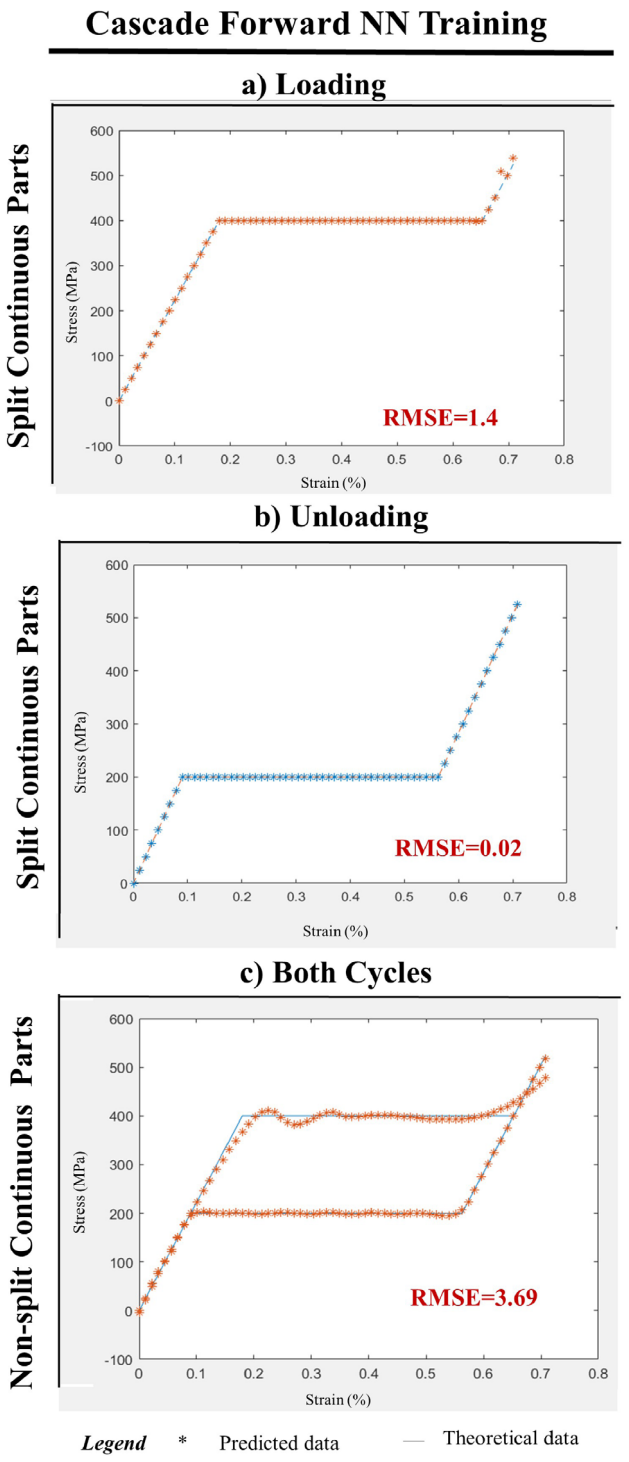

Figure 3. Results of the cascade forward NN for a single wire. (a) Training the loading cycle with splitting the continuous parts. (b) Training the unloading cycle with splitting the continuous parts. (c) Training the separated loading and unloading cycles without splitting the continuous parts.

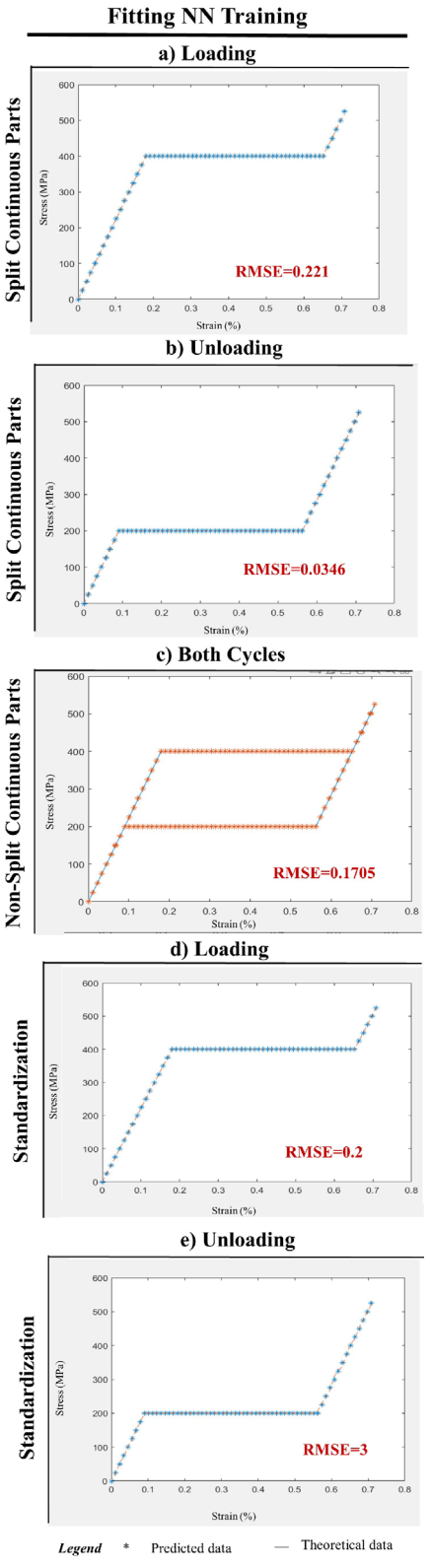

Figure 4. Caption continued on next page.

Figure 4 (cont.). Results of the fitting NN for a single wire. (a) Training the loading cycle with splitting the continuous parts. (b) Training the unloading cycle with splitting the continuous parts. (c) Training the separated loading and unloading cycles without splitting the continuous parts. (d–e) Training the standardized data of the separated loading and unloading cycles with splitting the continuous parts.

4. Antagonistic SMA wires

After analyzing the best NN characteristics in the previous section, we describe the modeling of an antagonistic SMA actuator with two SMAs wires.

4.1. *Methodology*

The experimental data of stress and strain is developed in [25], using two SMA wires with a one-way memory effect placed in series (Figure 6). The aim is to analyze the evolution of the force passing through the wires and the position of the bonding point between these wires during heating/returning cycles at room temperature.

This system was first identified experimentally using traction experiments on a wire at two speeds: 0.5 mm/min for test 1 and 125 mm/min for test 2. The tests correspond to the austenite to martensite states. The parameters of the SMA do not make it return to the austenitic state upon unloading (it keeps the deformed shape). This part is treated in this section. Each wire was then heated individually and sequentially by passing a current of 3 A with a 2 V power generator. Heating SMA 1 stretches SMA 2, leading to the martensitic transformation. Then, heating the SMA 2 leads to return to its initial position and to stretch the SMA 1. By that, the cyclic movement of coupling the two agonistic wires of SMA is produced, measured by a force sensor and an LVDT displacement sensor. This part will be considered in Section 6. The strain–stress data set is prepared by regulating the rate of variation and separating the continuous lines of each test.

4.2. *Results*

As a result, the LSTM has low RMSE values in modeling the hysteresis behavior of the SMA wire (Figures 7a, 8a). Moreover, the results are highly enhanced by standardizing the data set of the NN (Figures 7b, 8b), except for the constant phase, since there is no need for complexity.

RMSE has high values without standardizing the data, especially with higher non-linearity continuous parts as the third continuous part of test 1 (3: increasing part) and the second continuous part of test 2 (2: increasing part). Moreover, the sharp decreasing parts of both tests have a high RMSE value, beacuse, as the starting point is far from zero, the error of predicting the results increases.

In test 1, there is more negligible non-linearity than in test 2, so the RMSE is less in test 1 (0.36 for test 1 and 3.6 for test 2). It is good to know here that the error in prediction of the decreasing part in test 2 causes the high RMSE value of the system at all levels.

By standardizing data, the RMSE values are clearly decreased, which are for the continuous parts alone between 0.02 and 1 for test 1 and 0.03 and 2.7 for test 2. Moreover, the standardization solved the sharp decreasing part in test 1 and test 2. The RMSE values vary from one continuous part to another depending on the rate of variation between two points and the global linearity of the part. The RMSE values in the figures present the average value of the RMSE of continuous parts.

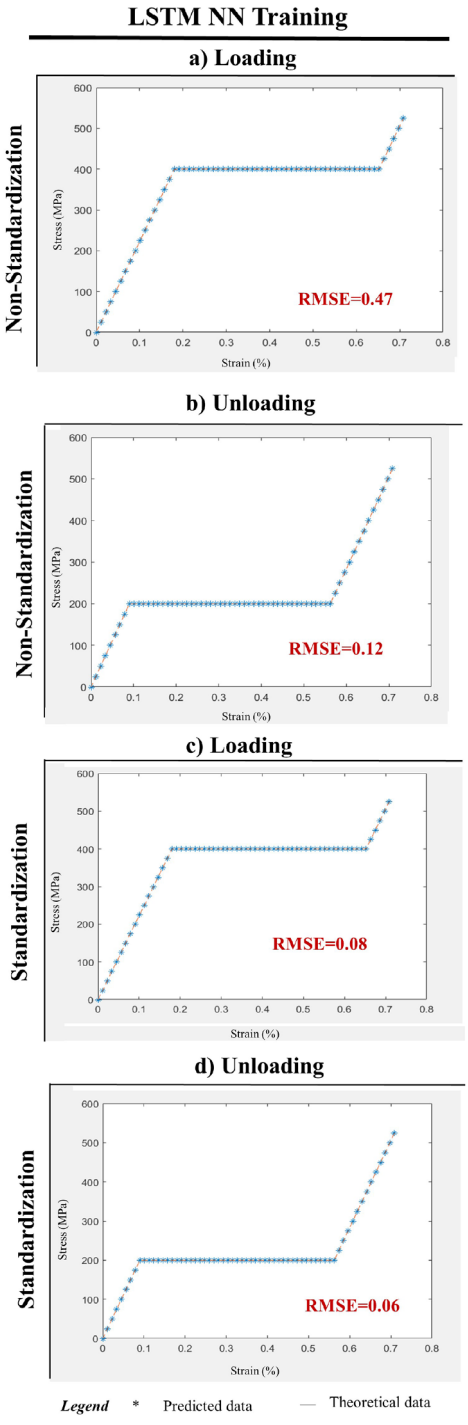

Figure 5. Results of the LSTM NN for a single wire. (a) Training the non-standardized loading cycle data with splitting the continuous parts. (b) Training the non-standardized unloading cycle data with splitting the continuous parts. (c) Training the standardized loading cycle data with splitting the continuous parts. (d) Training the standardized unloading cycle data with splitting the continuous parts.

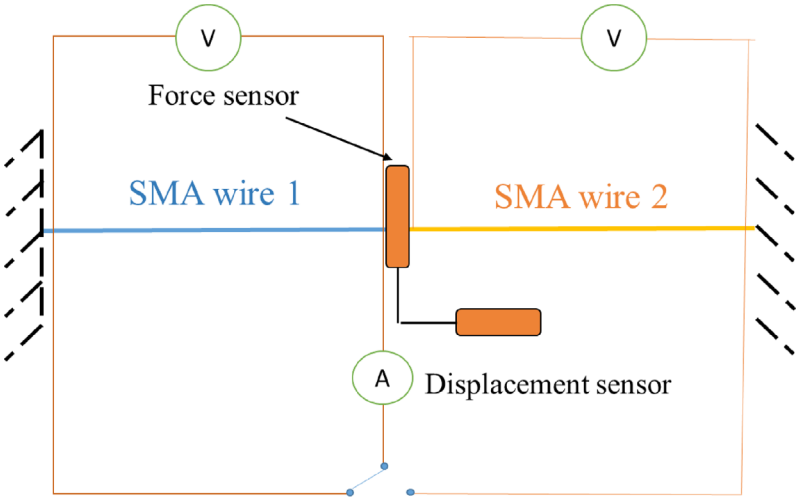

Figure 6. Set-up diagram of an antagonistic SMA actuator with two voltmeters and an ammeter.

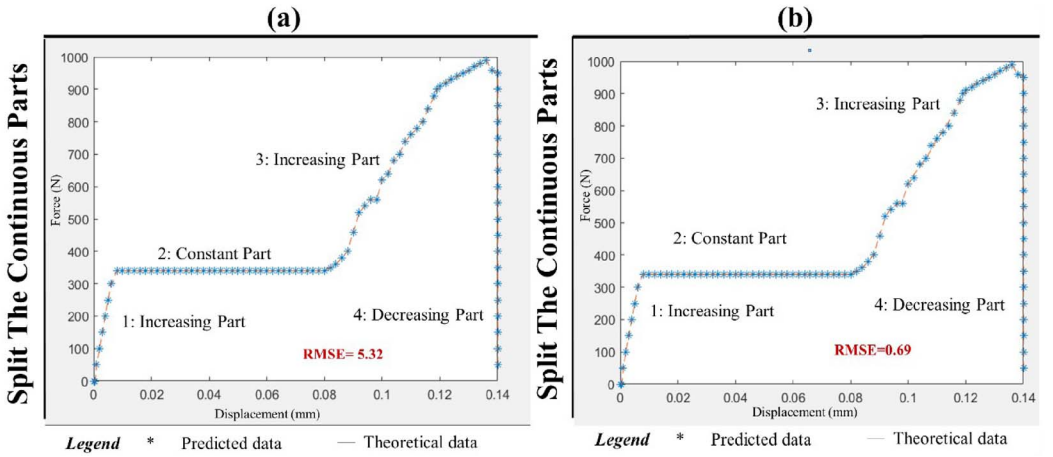

Figure 7. (a) Modeling of non-standardized data of test 1. (b) Modeling of standardized data of test 1. Each continuous part is trained separately.

5. Butterfly shape modeling

5.1. *Experimental background*

During this experiment, we focus on the mid-point of the two SMA wires to model their behavior. For that, we consider they are heated alternatively to create a displacement of this mid-point. The initial state of the experiment is at the austenitic phase of the two SMA wires, and the pre-stretching of the two wires transforms them partially into oriented martensite. Then, the four steps that perform one cycle are applied and repeated cyclically: (1) heat SMA 1, (2) cool down SMA 1 to the room temperature, (3) heat SMA 2, (4) cool down SMA 2 to room temperature. The superposition of the stress–strain curves for a few cycles gives a "butterfly-like" diagram [25]. The data recorded are the mid-point displacement, voltage and current intensity at the wires'

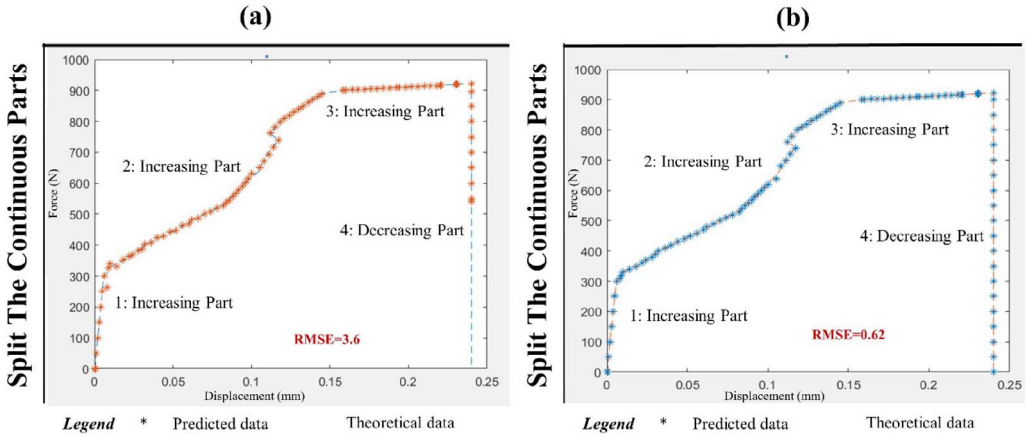

Figure 8. (a) Modeling of non-standardized data of test 2. (b) Modeling of standardized data of test 2. Each continuous part is trained separately.

terminals, and the force passing through the wire. Therefore, the force evolution in the wires as a function of the mid-point displacement is here the tested data to model the butterfly shape.

5.2. *Methodology*

In the previous systems, the LSTM NN with a regression layer proved its ability to model the hysteresis behavior using the standardized input data. Therefore, we check its training for this curvy and complex butterfly shape. We test two cycles, repeating the four steps twice, using the regularization and splitting input data into two parts: left part to zero and right part to zero (negative deformation and positive deformation) for each cycle.

On the Matlab platform, the LSTM with a regression layer is trained by this input data to obtain the experimental output data with low RMSE values and low computational time. For that, we decrease the hidden layer's size to 500 and the epoch value to 500 as suitable values.

5.3. *Results*

Modeling the butterfly shape (Figure 9) is done with RMSE values, between 0.5 and 0.9, using a low computational time of 1 min to 2 min. The tests are decomposed into two series: cycle one and cycle two. Then each series has two parts: the positive deformation and the negative deformation. Therefore, because of the high non-linear behavior of the middle point of the antagonistic actuator, standardizing the data enhances the training of LSTM NN more than for a single SMA wire. Moreover, the RMSE values of cycle one and cycle two again prove the effect of non-linearity on the NN training, which are 0.85 for cycle one and 0.6 for cycle two.

6. Discussion

The LSTM NN, with a regression layer acting as a network for sequence-to-one regression, successfully models the antagonistic SMA actuator.

LSTM NN is composed of a network containing different layers namely: sequence input, LSTM (layer), fully connected, regression, and output. LSTM's used arguments in our tests are one input (corresponding to the strain value array), one output (the stress value array), the

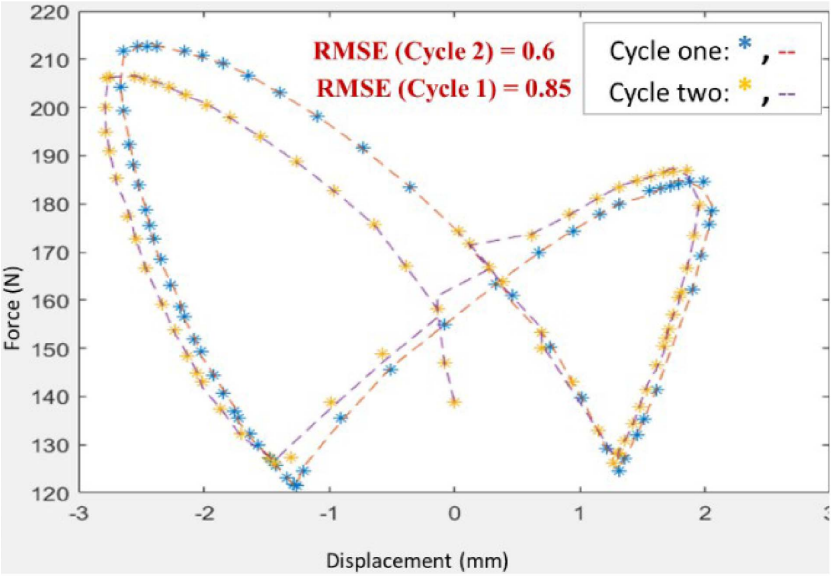

Figure 9. The butterfly shape of the two cycles. Each cycle is trained separately using more or less left and right separated parts.

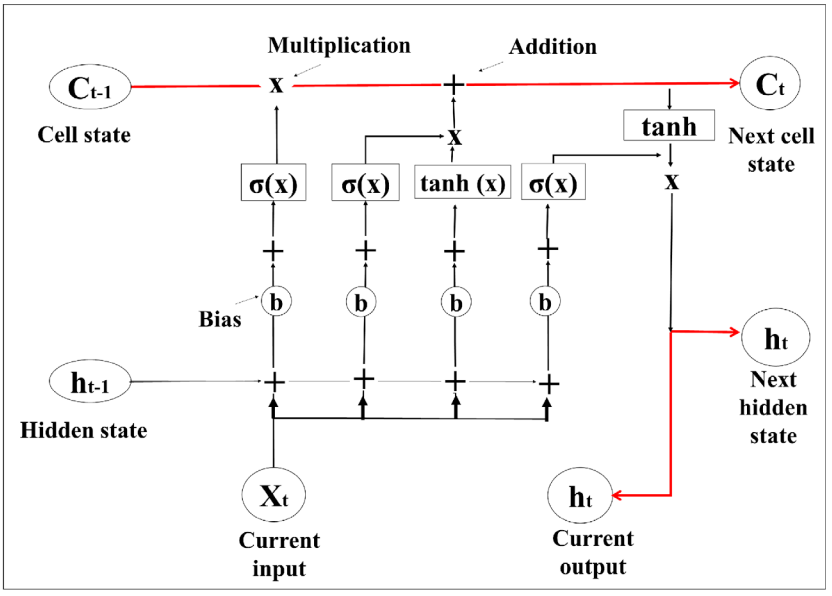

Figure 10. Architecture of the LSTM layer. The additional fully-connected and regression layers are not shown here. The LSTM layer comprises two inner states: cell and hidden.

number of hidden units forming the LSTM layer and corresponding to the amount of information remembered between hidden states, and additional layers as the fully connected layer and the regression layer. The LSTM layer updates its inner cell states and the hidden states using the hyperbolic tangent function (tanh) as the state activation function and the hard sigmoid (σ) function as the gate activation function (see Figure 10).

Moreover, an additional regression layer plays a significant role in the network's work. This linear regression layer is used for many functions, especially prediction, error reduction, and forecasting. It models the relationships using predictor functions whose unknown model parameters are estimated from the data. Moreover, the regression can be single or multiple, but in our case, it remains single. That is because it relates one scalar response to one dependent variable (stress to strain or force to displacement).

Note that there can be other approximation and interpolation methods than NNs, such as polynomial. The polynomial method could be used if the linear regression layer failed to capture the data set points.

The role of the polynomial method is to associate a single or a set of polynomials to a combinatorial studied object. Interestingly, the butterfly shape is not necessarily polynomial, and the NN works well in this case. The results shown in the article constitute a basis for the expected performance of NNs for stress/strain and load/displacement behaviors that could be used in the future for comparison with other advanced interpolation methods such as kriging.

7. Conclusion

The present work demonstrates the potential of artificial NNs to model the hysteresis behavior of SMA wires for robotic applications, starting from a simple form to the complex one of the antagonistic actuator. The significance of studying the smart materials as SMAs using NNs can be proved by the low computing time with high accuracy.

In the antagonistic system, we worked on the system's global behavior (SMA wires) and the behavior of the middle point of the antagonistic system forming the butterfly-shaped diagram. Knowing that the SMA is a spreading material in the robotic field, this work can be helpful in future optimal design or control of the inner load in real-time.

LSTM NN with a regression layer permits the modeling of the hysteresis behavior of SMA with low RMSE value and computational time. The RMSE values and the computational time of the training do not exceed 2% and 3 min on an average, respectively. The prediction times after the training are of the order of several ms.

The best way to train the NNs is to prepare regular input data sets with constant variation rates to provide the best-predicted output. These data regulation heuristics can be easily applied to more extensive databases if needed. At the same time, the computational time for the training can be considered reasonable compared to the construction of other approximation methods or problems. Also, the approximated predicted output data is close to the theoretical and experimental ones.

As a perspective, the proposed procedure can be used as a module predicting the inner load without a sensor in real-time control schemes. This will allow gaining precision without the additional weight of the sensors. Another perspective is to utilize it in material or system design. Fast evaluation schemes enable optimization at correct times concerning behavioral objectives such as the desired stretching or phase changing parameters for some given load actuation requirements. Note that NN in this paper can be further developed by using other training algorithms, as mentioned in the discussion section.

Finally, a very interesting perspective is to continue this work as applied to other smart materials such as ionic polymer metal composite (IPMC) which has high significance in robotic and engineering fields. We can study the unique electric load (the needed voltage) and the global deformation characteristics of the IPMC.

Conflicts of interest

Authors have no conflict of interest to declare.

Acknowledgments

This research is funded by the ANR project AIM (ANR-20-THIA-0001) and the European Union under the European Regional Development Fund (ERDF).

References

[1] H. S. Tzou, H.-J. Lee, S. M. Arnold, "Smart materials, precision sensors/actuators, smart structures, and structronic systems", *Mech. Adv. Mater. Struct.* **11** (2004), no. 4–5, p. 367-393.
[2] A. Bellini, M. Colli, E. Dragoni, "Mechatronic design of a shape memory alloy actuator for automotive tumble flaps: a case study", *IEEE Trans. Ind. Electron.* **56** (2009), no. 7, p. 2644-2656.
[3] E. Choi, H. D. Nguyen, J.-S. Jeon, J.-W. Kang, "Self-centering and damping devices using SMA dual rings", *Smart Mater. Struct.* **28** (2019), no. 8, article no. 085005.
[4] M. W. Gifari, H. Naghibi, S. Stramigioli, M. Abayazid, "A review on recent advances in soft surgical robots for endoscopic applications", *Int. J. Med. Robot.* **15** (2019), no. 5, article no. e2010.
[5] C. Fang, Y. Zheng, J. Chen, M. C. H. Yam, W. Wang, "Superelastic NiTi SMA cables: Thermal-mechanical behavior, hysteretic modelling and seismic application", *Eng. Struct.* **183** (2019), p. 533-549.
[6] C. Naresh, P. S. C. Bose, C. S. P. Rao, "Shape memory alloys: a state of art review", *IOP Conf. Ser. Mater. Sci. Eng.* **149** (2016), article no. 012054.
[7] Y. Liu, "The superelastic anisotropy in a NiTi shape memory alloy thin sheet", *Acta Mater.* **95** (2015), p. 411-427.
[8] Y. Huo, "A mathematical model for the hysteresis in shape memory alloys", *Contin. Mech. Thermodyn.* **1** (1989), no. 4, p. 283-303.
[9] T. E. Buchheit, J. A. Wert, "Predicting the orientation-dependent stress-induced transformation and detwinning response of shape memory alloy single crystals", *Metall. Mater. Trans. A* **27** (1996), no. 2, p. 269-279.
[10] S. Nemat-Nasser, W.-G. Guo, "Superelastic and cyclic response of NiTi SMA at various strain rates and temperatures", *Mech. Mater.* **38** (2006), no. 5–6, p. 463-474.
[11] O. E. Ozbulut, S. Hurlebaus, "Evaluation of the performance of a sliding-type base isolation system with a NiTi shape memory alloy device considering temperature effects", *Eng. Struct.* **32** (2010), no. 1, p. 238-249.
[12] T. Fukuda, T. Shibata, M. Tokita, T. Mitsuoka, "Neural network application for robotic motion control-adaptation and learning", in *1990 IJCNN International Joint Conference on Neural Networks*, vol. 2, 1990, p. 447-451.
[13] Josin, Charney, White, "Robot control using neural networks", in *IEEE 1988 International Conference on Neural Networks*, vol. 2, Neural Systems, Inc., Vancouver, BC, Canada, 1988, p. 625-631.
[14] A. Karakasoglu, S. I. Sudharsanan, M. K. Sundareshan, "Identification and decentralized adaptive control using dynamical neural networks with application to robotic manipulators", *IEEE Trans. Neural Netw.* **4** (1993), no. 6, p. 919-930.
[15] Y. H. Kim, F. L. Lewis, "Optimal design of CMAC neural-network controller for robot manipulators", *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **30** (2000), no. 1, p. 22-31.
[16] S.-Y. King, J.-N. Hwang, "Neural network architectures for robotic applications", *IEEE Trans. Robot. Autom.* **5** (1989), no. 5, p. 641-657.
[17] F. Sun, Z. Sun, P.-Y. Woo, "Neural network-based adaptive controller design of robotic manipulators with an observer", *IEEE Trans. Neural Netw.* **12** (2001), no. 1, p. 54-67.
[18] H. S. Tzou, H.-J. Lee, S. M. Arnold, "Smart materials, precision sensors/actuators, smart structures, and structronic systems", *Mech. Adv. Mater. Struct.* **11** (2004), no. 4–5, p. 367-393.
[19] N. T. Tai, K. K. Ahn, "A hysteresis functional link artificial neural network for identification and model predictive control of SMA actuator", *J. Process Control* **22** (2012), no. 4, p. 766-777.
[20] Y. I. E. Elbahy, M. N. Nehdi, M. A. Y. Youssef, "Artificial neural network model for deflection analysis of superelastic shape memory alloy reinforced concrete beams", *Can. J. Civ. Eng.* **37** (2010), no. 6, p. 855-865.
[21] S. Judd, "On the complexity of loading shallow neural networks", *J. Complex.* **4** (1988), no. 3, p. 177-192.
[22] A. Gómez-Espinosa, R. Castro Sundin, I. Loidi Eguren, E. Cuan-Urquizo, C. D. Treviño Quintanilla, "Neural network direct control with online learning for shape memory alloy manipulators", *Sensors* **19** (2019), no. 11, article no. 2576.
[23] R. Boufayed, F. Chapelle, J. F. Destrebecq, X. Balandraud, "Finite element analysis of a prestressed mechanism with multi-antagonistic and hysteretic SMA actuation", *Meccanica* **55** (2020), no. 5, p. 1007-1024.
[24] K. Divringi, C. Ozcan, "Advanced shape memory alloy material models for ANSYS", 2016, https://www.ozeninc.com/wp-content/uploads/2016/01/Advanced-Shape-Memory-Alloy-Material-Models-for-ANSYS.pdf.
[25] A. Waibaye, "Création de structures actives à l'aide d'alliages à mémoire de forme", PhD Thesis, Blaise Pascal University, Clermont-Ferrand, France, 2016.