Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Dimensionality Effects on the Markov Property in Shape Memory Alloy Hysteretic Environment

Kenton Kirkpatrick and John Valasek, *Senior Member, IEEE*
Aerospace Engineering Department
Texas A&M University
College Station, TX, USA
kentonkirk@gmail.com and valasek@tamu.edu

Abstract— **Shape Memory Alloy actuators can be used for morphing, or shape change, by controlling their temperature, which is effectively done by applying a voltage difference across their length. Control of these actuators requires determination of the relationship between voltage and strain so that an input-output map can be developed. To determine this policy and map the hysteretic region, a Reinforcement Learning algorithm called Sarsa was used. Proper use of Reinforcement Learning requires that the learning environment have the Markov Property. However, hysteresis spaces are commonly referenced as non-Markovian due to the fact that state history is needed to properly predict future states and rewards. This paper reveals that this formerly non-Markovian learning environment of Shape Memory Alloy hysteresis can become Markovian by means of increasing the dimensionality of the measured states. The paper compares learning attempts in both versions of the environment and will show that Reinforcement Learning is successful in the modified learning environment by learning a near-optimal policy for controlling the length of a Shape Memory Alloy wire. This is then validated by using the modified Reinforcement Learning agent to learn a near-optimal control policy in an experimental setting.**

***Keywords*—Markov Property, reinforcement learning, hysteresis, Shape Memory Alloy, morphing**

I. INTRODUCTION

Advancement of aerospace structures has led to an era where researchers now look to nature for ideas that will increase performance in aerospace vehicles, particularly by advancing the research and development of bio- and nano-technology [1]. Birds have the natural ability to move their wings to adjust to different configurations of optimal performance. The ability for an aircraft to change its shape during flight for the purpose of optimizing its performance under different flight conditions and maneuvers would be revolutionary to the aerospace industry. To achieve the ability to morph an aircraft, exploration in the materials field has led to the idea of using Shape Memory Alloys (SMAs) as actuators to drive the shape change of a wing. The idea of using active materials for nonlinear structural morphing is being explored in a variety of ways with different types of smart materials are used, and SMAs are one field that shows promise [2],[3],[4],[5],[6]. The field of SMA research has already begun branching into conceptualized morphing aircraft, with considerations to structure and aeroelasticity being considered [7],[8],[9] There are many types of SMAs which have different compositions, but the most commonly used SMAs are either a composition of nickel and titanium or the combination of nickel, titanium, and copper.

SMAs have a unique ability known as the Shape Memory Effect [10],[11]. This material can be put under a stress that leads to a plastic deformation and then fully recover to its original shape after heating it to a high temperature. This would make SMAs useful for structures that undergo large deformations, such as morphing aircraft [12]. At room temperature, SMAs begin in a crystalline structure of martensite and undergo a phase change to austenite as the alloy is heated. This phase transformation realigns the molecules so that the alloy returns to its original austenitic shape. The original martensitic shape is re-obtained when the SMA is cooled back to a martensitic state, recovering the SMA from the strain that it had endured. This occurs because the phase transformation from martensite to austenite begins and ends at different temperatures than the reverse process, and the relationship is highly nonlinear, as shown in Fig. 1.

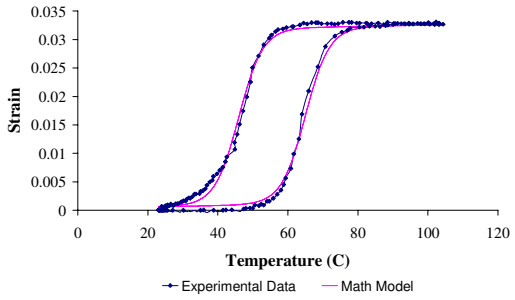

Figure 1. SMA Hysteresis

The hysteresis behavior of SMAs in temperature-strain space is most often characterized through the use of constitutive models that are based on material parameters or by models resulting from system identification [13]. This is a time and labor intensive process that requires external supervision and does not actively discover the hysteresis in real-time, both of which are considerations that are undesirable for online

learning of a control policy. Other methods that characterize this behavior are phenomenological models [14],[15],[16], micromechanical models [17],[18], and empirical models based on system identification [19],[20]. These models are quite accurate, but some only work for particular types of SMAs and most require complex computations. Many of them are also unable to be used in dynamic loading conditions, making them unusable in the case of morphing. A drawback to using any of these methods is that the minor hysteresis loops within an SMA that is not fully actuated are not characterized and must be determined within analytical models.

This research investigates a technique for determining input-output policies for controlling SMA wires. Since there is not a parametric model available to use for this policy, this research uses a machine learning algorithm known as Reinforcement Learning (RL) to discover the black-box control policy for an SMA wire [21]. RL is a form of machine learning that utilizes the interaction with multiple situations many times in order to discover the optimal path that must be taken to reach the pre-determined goal. By learning the behavior in real-time, both major and minor hysteresis loops can be experimentally determined through this method, while simultaneously learning the policy required for control.

The policy that is learned by RL and simultaneously used for learning consists of a discrete table of values representing those actions that have the highest probabilities of providing the maximum reward at each given current state. Since this research uses strain as the goal, and each goal strain is attainable from each current strain in a single action, the action that maximizes reward is the action that achieves the goal strain immediately. Since the resulting policy is a direct input-output map from current state and goal state to action, the environment being explored must be Markovian, meaning that it has the Markov Property. Hysteresis spaces are classic examples of non-Markovian environments, making it difficult to apply a RL approach.

In this paper, the hysteretic state-space will be expanded to include temperature as another dimension in order to obtain a Markovian learning environment. In Section IIA, the RL algorithm used in this research will be explained. Following this, Section IIB will provide a comparison between the two learning environments as well as an explanation for why the modified environment is Markovian. Section III provides simulation results including comparisons between attempted applications of RL in each of the two state-spaces. Finally, experimental verification of a learned policy will be demonstrated in Section IV, followed by conclusions in Section V.

II. REINFORCEMENT LEARNING

A. Sarsa

Reinforcement Learning is a process of learning through interaction in which a program uses previous knowledge of the results of its actions in each situation to make an informed decision when it later returns to the same situation. It is a method that has be used for many diverse situations ranging from board games to behavior-based robotics [22],[23],[24]. The purpose of the learning agent used in RL is to maximize the long-term cumulative reward, not just the immediate reward [22]. However, in this research there is only one dimension that yields any reward: strain. Since any goal strain is attainable within a certain error range based on knowledge of both current strain and current temperature, this implies that the agent maximizes rewards by minimizing the actions required to reach the goal strain, making the action associated with the maximum immediate reward also the action associated with the maximum cumulative award. The agent uses the knowledge gained by reward maximization to update a control policy that is a function of the states and actions. This control policy is essentially a large matrix that is composed of every possible state for the rows, and every possible action for the columns. In this research, a third dimension is included in the control policy that is composed of every possible goal state.

The three most commonly used classes of RL algorithms are Dynamic Programming, Monte Carlo, and Temporal Difference [22]. The majority of Dynamic Programming methods require an environmental model, making the use of them impractical in problems with complex models. Monte Carlo only allows learning to occur at the end of each episode, causing problems that have long episodes to have a slow learning rate. Temporal Difference methods have the advantage of being able to learn at every time step without requiring the input of an environmental model. This research utilizes a method of Temporal Difference known as Sarsa. Sarsa is an on-policy form of Temporal Difference, meaning that at every time interval the control policy is evaluated and improved. An on-policy method is preferred here because the learning will occur in real-time, and the Q matrix needs to be updated in real-time as this learning progresses. Sarsa updates the control policy by using the current state, current action, future reward, future state, and future action to dictate the transition from one state/action pair to the next [22]. The action value function used to update this control policy is:

$$Q_q(s,a) \leftarrow Q_q(s,a) + \eta[r_{q+1}(s',a') + \gamma Q_{q+1}(s',a') - Q_q(s,a)] \quad (1)$$

In (1), Q is the control policy, s is the current state, a is the current action, s' is the future state, a' is the future action, r is the reward, η is a repetition penalty, γ is a future policy weight, and the subscript q represents the time step. In this research, the control policy represented in (1) was modified to include the goal state as a third dimension in order to have one control policy that represents all goal states, rather than a different policy for each goal.

When approaching the point in the algorithm where the action must be determined from Q, the problem of which method would be best for choosing this action must be solved. The dilemma lies in the fact that the policy does not have any information about the system in the beginning, and must explore so that it can learn the system. The point of using RL

is to learn the system when no prior knowledge of the system is known by the algorithm, so it can not exploit previous knowledge in the beginning stages. However, in future episodes the policy will have more information about the system, and exploitation of knowledge becomes more favorable. The key to optimizing the convergence of the RL module upon the best control policy is to balance the use of exploration and exploitation.

The ε-Greedy method of choosing an action is used in this research, which means that for some percentage of the time that an action is chosen, the RL module will choose to randomly explore rather than choose the action that the action-value function declares is the best [25]. This is because the RL agent might not have already explored every possible option, and a better path may exist than the one that is presently thought to yield the greatest reward. A fully greedy method chooses only the optimal path without ever choosing to explore new paths, which corresponds to an ε-Greedy method where $\varepsilon = 0$.

To converge on the optimal control policy in the shortest amount of time, this research used an episodically changing ε-Greedy method by altering the exploration constant, ε, depending upon the current episode. ε is a number between 0 and 1 that determines the percent chance that exploration will be used instead of exploitation. In the first episodes, little to no information has been learned by the policy, so a greater degree of exploration is required. Conversely, in future episodes less exploration is desired so that the RL module can exploit the knowledge of the system that it has learned.

To achieve an episodically changing ε-Greedy method, a simple algorithm was constructed that determines what value would be used for ε at each individual episode. The values of ε ranged from 70% in the first several episodes to 5% in the final episodes, and were chosen during simulation by experimenting with the values and episode numbers until the best convergence time was found. Even during later episodes, the algorithm still never exhibits a fully greedy method of choosing actions. A small chance of performing exploratory actions is still used because it allows the system to check for better paths in case the path it converged upon is not actually the most optimal choice. The episodes at which the values of ε were modified and the corresponding values are as follows:

TABLE I. EPISODIC Ɛ-GREEDY VALUES

Episode	1	30	60	80	100	140+
ε	0.7	0.6	0.5	0.3	0.2	0.05

Once the RL algorithm learns the optimal voltage required to achieve each goal strain from each initial strain, it can then be used to control the length of a SMA wire in real-time. The learned policy's ability to control the SMA wire's length can then be demonstrated and plotted for validation.

B. The Markov Property

For RL to be used to learn an input-output relationship, the environment needs to have the Markov property. In an environment with the Markov property, learning how to move from one state to another depends only on the current state, and not state history [22],[26]. In a general environment, the probability of achieving a specific goal and thereby obtaining a specific reward depends on the current and past states, actions, and rewards. This is demonstrated in (2) [22].

$$\Pr\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t, r_t, s_{t-1}, a_{t-1}, \ldots, r_1, s_0, a_0\} \qquad (2)$$

In an environment that has the Markov property, the probability distribution described by (2) can be simplified to only depend on the current state and action. The dynamics of the system can be fully described by only using the probability of achieving a certain state and obtaining the associated reward given the current state. The Markov property probability distribution is represented by (3) [22].

$$\Pr\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t\} \qquad (3)$$

Hysteresis is non-Markovian in nature because moving from one state to another requires knowing not only the current state but also the state history. In this research, this would imply that due to the hysteresis, both the current strain and past strains would be needed to know how to reach the goal strain. This is a problem when using RL because the control policy learned by RL is a function of only the current state, action, and goal. However, this problem was overcome by the specific formulation of this learning environment. Hysteresis is non-Markovian in the case of attempting to move from one strain in the hysteretic space to another, as shown in Fig. 3.

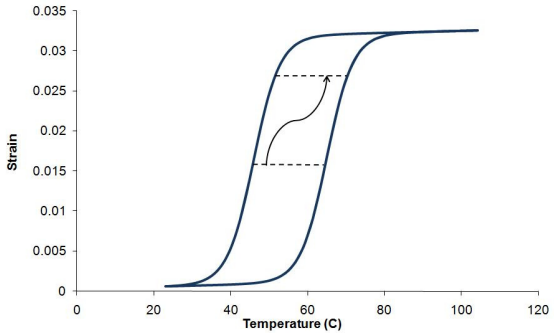

Figure 2. Non-Markovian Travel in SMA Hysteresis

Attempting to learn this motion using RL would be a challenge since moving from one strain to another in a hysteretic environment requires strain time history to be known. However, in this research the current state of the system is not simply the current strain, but both current strain

and temperature. The goal in this research is to move from one specific point in temperature-strain space to any point along the goal strain line in one action, without any restrictions on goal temperature. This type of learning environment is represented in Fig. 3.

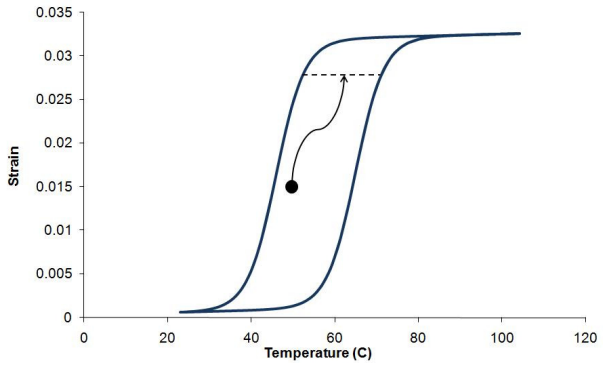

Figure 3. Markovian Travel in SMA Hysteresis

The learning environment described by Fig. 3 allows travel from any one point in the hysteresis space to anywhere along the horizontal goal line in the hysteresis space without knowing state history, indicating that it is Markovian. The only reason history of strain would be needed would be in the event that temperature was not measured, in which case the agent would need to know where along the horizontal line of current strain it lies. In the learning environment used in this research, temperature is directly measured by means of a thermocouple. The need to know strain state history is eliminated by the inclusion of a temperature dimension, indicating that the environment is Markovian. Using this construct of the learning environment, RL can be used for learning the optimal control policy.

III. SIMULATION

To validate the need to increase the dimensionality of the state-space before using RL to learn an SMA control policy, a comparison of the learning behavior using each of the respective state-spaces is necessary. In this section, SMA hysteresis is simulated and exploited by the learning agent under each of the scenarios.

A. SMA Simulation Model

For simulation of the learning environment to be properly achieved, the SMA temperature-strain space must be accurately modeled with real-time feedback. In this research, a hyperbolic tangent model was used to provide the simulated SMA dynamics. The hyperbolic tangent model is based on the curves given by (4) and (5).

$$M_l = \frac{H}{2}\tanh\left(\left(T - ct_l\right)a_h\right) + s_h\left(T - \frac{(ct_l + ct_r)}{2}\right) + \frac{H}{2} + c_s \quad (4)$$

$$M_r = \frac{H}{2}\tanh\left(\left(T - ct_r\right)a_h\right) + s_h\left(T - \frac{(ct_l + ct_r)}{2}\right) + \frac{H}{2} + c_s \quad (5)$$

In these equations, H, ct_r, a_h, s_h, ct_l, and c_s are constants that determine the shape of the hyperbolic tangent model. M_r and M_l are the strain values that correspond to the temperature input into the equations. The constants were selected by creating a curve that best fit an experimentally determined hysteresis behavior for a SMA wire. The minor hysteresis loops are approximated by compressing and translating the major hysteresis curves according to which maximum and minimum minor strains are involved in the process. Using this approximation of the SMA hysteresis behavior, the RL agent can simulate interaction with an SMA wire.

B. 1-D State-Space Simulation

This section will include RL agent learning results in the simulation of an SMA environment involving only a 1-D state-space consisting of strain. When these results are available, the section will be updated.

For comparison reasons, this section shows simulation results of using the Sarsa learning agent when no temperature information is available for input. The state-space includes only strain for the current state, and desired strain is the goal state.

This simulation was run for the same amount of time as the simulation in Section IIIC, and the resultant testing of the control policy yielded the time history in Figure 4.

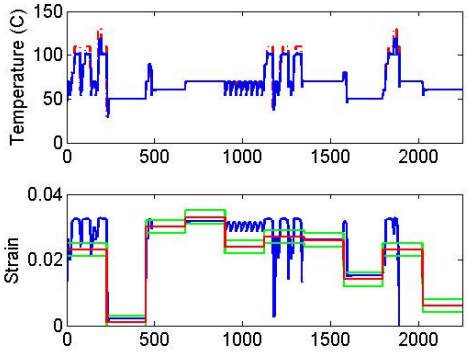

Figure 4. Simulation without Temperature Inputs

C. 2-D State-Space Simulation

For this simulation, the state-space is expanded to include both temperature and strain to define the current state. The goal state is still only representative of strain. Using this learning environment, the RL agent was able to converge on a control policy capable of controlling the simulated SMA wire. Fig. 5 shows the resultant time history response of this learning process.

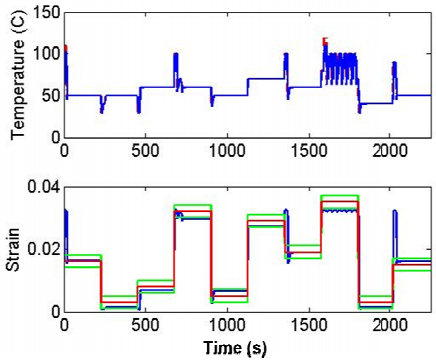

Figure 5. Simulation with Temperature Inputs

As can be seen in Fig. 5, the RL agent is successfully able to converge to a policy capable of controlling the simulated SMA wire for the goals used in simulation. In this figure, the red line represents the goal, the blue represents the RL agent's moves, and the green bars are the goal tolerance. The tolerance allowed for achieving a goal in this simulation was ±0.002 strain. These results show that the RL agent is capable of converging upon a near-optimal control policy, indicating that adding the temperature dimension to the current state does indeed provide a Markovian environment.

IV. EXPERIMENTAL VERIFICATION

To verify that the RL agent demonstrated in simulation is able to learn with an actual SMA wire, the algorithm was modified to allow interaction with an actual SMA wire in an experimental setup. The control policy developed for the particular SMA specimen tested provided the ability to control the length of a NiTi SMA wire for 2 specific goal strains within an error range of ±0.005 strain. The wire used for this experiment had an initial effective length of 13cm, so with a maximum strain possible of 3.3%, the total operating range of motion was 4.29mm. Since the control policy learned was able to reach its goal within a range of ±0.5%, the error range allowed was ±0.65mm. Under these specified conditions, the RL module was executed for 100 episodes using specified alternating goal strains of 2.7% and 0.1%, providing 50 episodes per goal. Each episode in this experiment consists of 450 seconds worth of seeking a single goal, where the RL module is called every 15 seconds. This provides 30 new actions per episode for the learning module.

This goal chosen for experimentation was 2.7% axial strain because it represents a partially actuated state for which the maximum strain of 3.3% falls outside of the allowed tolerance range of ±0.5%. This ensures that it can not achieve the goal by simply applying the maximum voltage available. This goal is also of particular interest since it was previously used for temperature-strain space validation. Under these conditions, the final control policy was tested and the results can be seen in Fig. 6.

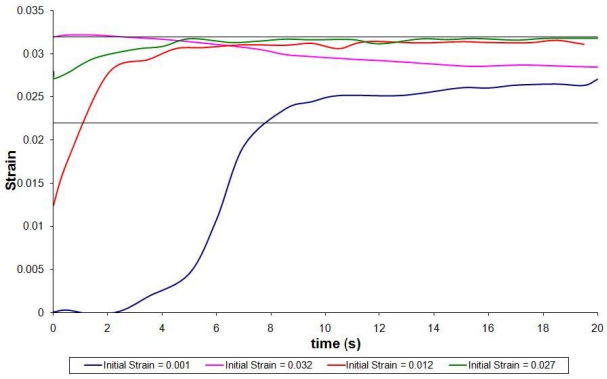

Figure 6. Experimental Time History for Goal = 2.7%

Fig. 6 reveals that the control policy developed by the RL agent is capable of bringing an SMA wire to the desired goal from multiple initial positions. This ability makes the development of morphing actuators possible. In Fig. 6, the initial strains chosen for testing here were 0.1%, 3.2%, 1.2%, and 2.7%. The two horizontal lines represent the goal range of 2.7% ± 0.5% strain. The initial strains of 0.1% and 3.2% were chosen so that the control policy could be tested from initial strains corresponding to fully un-actuated and fully actuated states, respectively. The initial strain of 1.2% was selected in order to test from an initially intermediate strain, and the goal strain of 2.7% was also chosen as an initial strain to show that the agent can learn how to stay within the specified range when the specimen is there initially. As Fig. 6 shows, the control policy was successful in achieving its goal of 2.7% ± 0.5% in all 4 test cases.

Using RL to learn a control policy capable of achieving a strain that rests within the interior of the transformation curve is important because it greatly increases the range of functionality of SMA actuators. If the only values learned by the agent are those that correspond to maximum and minimum strains, a SMA actuator would be limited to only two possible positions. Learning these interior goals is also far more complicated than learning the extreme values because all that would be required for the latter would be to apply the maximum and minimum voltages every time. By showing that this RL approach can learn how to reach 2.7% strain, this research has proven that using a RL agent to learn a SMA control policy makes it possible to create a SMA actuator capable of achieving multiple position changes. It follows from these tests that creating SMA actuators for the purpose of developing morphing aircraft is feasible. These experimental results validate the RL agent's ability to learn in this modified environment.

V. CONCLUSIONS

Based upon the analysis and results presented in this research, the following conclusions are made:

1) Although hysteresis space is classically considered to be non-Markovian, the Shape Memory Alloy's temperature-strain space can be made Markovian by measuring the temperature and using it to increase the dimensionality of the state-space. Measuring strain state history is only needed to know what the current temperature is, so measuring temperature directly eliminates the need to know strain history.

2) The results of the experimental stage established the ability to learn a control policy in an online experiment without human external supervision, and validated the approach experimentally. With the tolerances chosen for goal achievements, the Reinforcement Learning agent was able to converge to a near-optimal policy within 100 episodes.

ACKNOWLEDGMENT

This work was sponsored (in part) by the National Science Foundation Graduate Research Fellowship Program and the Air Force Office of Scientific Research, USAF, under grant/contract number FA9550-08-1-0038. The Technical Monitor is Dr. William M. McEneaney. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the National Science Foundation, Air Force Office of Scientific Research, or the U.S. Government.

REFERENCES

[1] Texas Institute for Intelligent Bio-Nano Materials and Structures for Aerospace Vehicles Home Page. URL: http://tiims.tamu.edu [cited 15 May 2004].

[2] Agarwal, Sandeep, "Structural Morphing using Piezoelectric Modulation of Joint Friction," *Journal of Intelligent Material Systems and Structures,* April 2007, vol. 18, pp. 389-407.

[3] Barbarino, S., Ameduri, S., Lecce, L., and Concilio, A., "Wing Shape Control though an SMA-Based Device," *Journal of Intelligent Material Systems and Structures,* February 2009, vol. 20, no. 3, pp. 283-296.

[4] Kudva, J. N., "Overview of the DARPA Smart Wing Project," *Journal of Intelligent Material Systems and Structures,* April 2004, vol. 15, pp. 261-267.

[5] Yoo, In K. and Desu, Seshu B., "Fatigue and Hysteresis Modeling of Ferroelectric Materials," *Journal of Intelligent Material Systems and Structures,* October 1993, vol. 4, pp. 490-495.

[6] Johnson, Terrence, Frecker, Mary I., Abdalla, Mostafa M., Gurdal, Zafer, and Lindner, Douglas K., "Nonlinear Analysis and Optimization of Diamond Cell Morphing Wings," *Journal of Intelligent Material Systems and Structures,* November 2008, vol. 0, pp. 1045389X08098098v1.

[7] Bae, Jae-Sung, Kyong, Nam-Ho, Seigler, T. Michael, and Inman, Daniel J., "Aeroelastic Considerations on Shape Control of an Adaptive Wing," *Journal of Intelligent Material Systems and Structures,* vol. 16, pp. 1051-1056.

[8] Matsuzaki, Yuji, "Recent Research on Adaptive Structures and Materials: Shape Memory Alloys and Aeroelastic Stability Prediction," *Journal of Intelligent Material Systems and Structures,* December 2005, vol. 16, pp. 907-917.

[9] Strelec, Justin K., Lagoudas, Dimitris C., Khan, Mohammed A., and Yen, John, "Design and Implementation of a Shape Memory Alloy Actuated Reconfigurable Airfoil," *Journal of Intelligent Material Systems and Structures*, April 2003, vol. 14, pp. 257-273.

[10] Waram, Tom. *Actuator Design Using Shape Memory Alloys*. Hamilton, Ontario: T.C. Waram, 1993.

[11] Sofla, A.Y.N., Elzey, D.M., and Wadley, H.N.G., "Two-way Antagonistic Shape Actuation Based on the One-way Shape Memory Effect," *Journal of Intelligent Material Systems and Structures,* September 2008, vol. 19, pp. 1017-1027.

[12] Mavroidis, C., Pfeiffer, C. and Mosley, M., "Conventional Actuators, Shape Memory Alloys, and Electrorheological Fluids." *Automation, Miniature Robotics and Sensors for Non-Destructive Testing and Evaluation*, April 1999, p. 10-21.

[13] Lagoudas, D., Mayes, J., Khan, M., "Simplified Shape Memory Alloy (SMA) Material Model for Vibration Isolation," Smart Structures and Materials Conference, Newport Beach, CA, 5-8 March 2001.

[14] Lagoudas, D. C., Bo, Z., and Qidwai, M. A., "A unified thermodynamic constitutive model for SMA and finite element analysis of active metal matrix composites "*Mechanics of Composite Materials and Structures*, vol. 3, 153, 1996.

[15] Bo, Z. and Lagoudas, D. C., "Thermomechanical modeling of polycrystalline SMAs under cyclic loading, Part I-IV" *International Journal of Engineering Science*, vol. 37 1999.

[16] Malovrh, Brendon and Gandhi, Farhan, "Mechanism-Based Phenomenological Models for the Pseudoelastic Hysteresis Behavior of Shape Memory Alloys," *Journal of Intelligent Material Systems and Structures,* January 2001, vol. 12, pp. 21-30.

[17] Patoor, E., Eberhardt, A., and Berveiller, M., "Potential pseudoelastic et plasticite de transformation martensitique dans les mono-et polycristaux metalliques." *Acta Metall* 35(12), 2779, 1987.

[18] Falk, F., "Pseudoelastic stress strain curves of polycrystalline shape memory alloys calculated from single crystal data" *International Journal of Engineering Science*, 27, 277, 1989.

[19] Banks, H., Kurdila, A. and Webb, G. 1997. "Modeling and Identification of Hysteresis in Active Material Actuators, Part (ii): Convergent Approximations," *Journal of Intelligent Material Systems and Structures*, 8(6).

[20] Webb, G., Kurdila, A. and Lagoudas, D. 1998. "Hysteresis Modeling of SMA Actuators for Control Applications," *Journal of Intelligent Material Systems and Structures*, vol. 9, no. 6, pp.432-447.

[21] Kirkpatrick, Kenton and Valasek, John, "Reinforcement Learning for Characterizing Hysteresis Behavior of Shape Memory Alloys," AIAA-2007-2932, Proceedings of the AIAA Infotech@Aerospace Conference, Rohnert Park, CA, 7-10 May 2007.

[22] Sutton, R. and Barto, A., *Reinforcement Learning: An Introduction.* Cambridge, Massachusetts: The MIT Press, 1998.

[23] Konidaris, G. D. and Hayes, G.M., "An Architecture for Behavior-Based Reinforcement Learning," *Adaptive Behavior,* March 2005, vol. 13, pp. 5-32.

[24] Varshavskaya, Paulina, Kaelbling, Leslie Pack, and Rus, Daniela, "Automated Design of Adaptive Controllers for Modular Robots using Reinforcement Learning," *The International Journal of Robotics Research,* March 2008, vol. 27, pp. 505-526.

[25] Whiteson, Shimon, Taylor, Matthew E., and Stone, Peter, "Empirical Studies in Action Selection with Reinforcement Learning," *Adaptive Behavior*, March 2007, vol. 15, pp. 33-50.

[26] Bhatnagar, Shalabh and Abdulla, Mohammed Shahid, "Simulation-Based Optimization Algorithms for Finite-Horizon Markov Decision Processes" *SIMULATION*, December 2008, vol. 84, pp. 577-600.