



Brief paper

Optimal compression of generalized Prandtl–Ishlinskii hysteresis models[☆]Jun Zhang, Emmanuelle Merced, Nelson Sepúlveda, Xiaobo Tan¹

Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA

ARTICLE INFO

Article history:

Received 22 November 2013

Received in revised form

18 January 2015

Accepted 2 April 2015

Keywords:

Hysteresis

Prandtl–Ishlinskii model

Dynamic programming

Optimal compression

Entropy

ABSTRACT

Prandtl–Ishlinskii (PI) hysteresis models have been used widely in magnetic and smart material-based systems. A generalized PI model, consisting of a weighted superposition of generalized play operators, is capable of characterizing saturated and asymmetric hysteresis. The fidelity of the model hinges on accurate representation of the envelope functions, play operator radii, and corresponding weights. Existing work has typically adopted some predefined play radii, the performance of which could be far from optimal. In this paper, novel schemes are proposed for optimally compressing generalized PI models, subject to a complexity constraint characterized by the number of play operators. An information-theoretic tool, entropy, is adopted to capture the information loss in replacing a group of weighted play operators with a single play operator. The optimal compression algorithm is reformulated as an optimal control problem and solved with dynamic programming, the computational complexity of which is shown to be much lower than that of exhaustive search. Simulation results are first presented to examine the performance of the proposed approach in approximating a PI model consisting of a large number of play operators, where cases with different types of weight distributions are explored. The approach is further applied to compress an experimentally identified generalized PI model for the complex hysteresis behavior between the resistance and temperature of vanadium dioxide, a promising multifunctional material. Both simulation and experimental results demonstrate that the proposed algorithms in general yield far more superior performance than a typically adopted scheme where the play radii are assigned uniformly.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Hysteresis is a nonlinear phenomenon that has been found in a wide class of magnetic and smart material systems. There has been considerable amount of recent work dealing with modeling and control of systems with hysteresis (Krejci & Kuhnen, 2001; Song, Zhao, Zhou, & de Abreu-Garcia, 2005; Tan & Baras, 2004; Zhang, Merced, Sepúlveda, & Tan, 2014b). Hysteresis models can be classified into physics-based and phenomenology-based models.

The physics-based models (Harrison, 2003) are usually derived on the basis of particular material properties or physical principles. In contrast, phenomenology-based models, such as Preisach operator (Song et al., 2005; Tan & Baras, 2004; Zhang et al., 2014b), Prandtl–Ishlinskii (PI) model (Esbrook, Tan, & Khalil, 2013; Krejci & Kuhnen, 2001; Wang & Su, 2006; Zhang, Merced, Sepúlveda, & Tan, 2014a), Duhem model (Oh & Bernstein, 2005), Bouc–Wen model (Ikhouane & Rodellar, 2006), and Maxwell model (Vo-Minh, Tjahjowidodo, Ramon, & Brussel, 2011), are often applicable to a broader class of systems with hysteresis, and thus have been adopted more extensively to model hysteresis behavior. Among them, the PI model is popular and has been proven effective in hysteresis modeling and control.

A PI model consists of a weighted superposition of elementary play (or stop) operators. The classical PI model (Krejci & Kuhnen, 2001) is limited to modeling symmetric and non-saturated hysteresis. Kuhnen (2003) formulated a modified PI model that combines play operators with deadzone operators to enable the modeling of asymmetric hysteresis. Generalized play operators (Brokate & Sprekels, 1996; Visintin, 1994), where the envelope

[☆] This work was supported in part by the National Science Foundation (ECCS 0547131, CMMI 0824830, CMMI 1301243). Emmanuelle Merced was supported by the National Science Foundation under Grant No. DGE-0802267 (Graduate Research Fellowship Program). The material in this paper was partially presented at the 2013 ASME Dynamic Systems and Control Conference, October 21–23, 2013, Palo Alto, CA, USA. This paper was recommended for publication in revised form by Associate Editor Alessandro Chiuso under the direction of Editor Torsten Söderström.

E-mail addresses: zhangj78@egr.msu.edu (J. Zhang), mercedem@hp.com (E. Merced), nelsons@egr.msu.edu (N. Sepúlveda), xbtan@egr.msu.edu (X. Tan).

¹ Tel.: +1 517 432 5671; fax: +1 517 353 1980.

function takes a nonlinear form, can be used to construct generalized PI models capable of modeling complex asymmetric hysteresis with output saturation (Al Janaideh, Rakheja, & Su, 2009). Al Janaideh et al. further developed an analytical inversion algorithm for a generalized PI model (Al Janaideh, Rakheja, & Su, 2011).

Existing work on PI models has typically adopted some predefined play radii (Esbrook et al., 2013; Krejci & Kuhnen, 2001; Kuhnen, 2003), the modeling performance of which could be far from optimal. While it is generally true that the modeling performance improves with an increasing number of play operators, computational and data storage cost will also increase for the model and the corresponding model-based inverse compensation schemes. Obtaining an accurate model while maintaining relatively low calculation and storage complexity, most dominantly reflected by the number of play operators, is thus an issue of practical interest. In the recent work of the authors (Zhang, Merced, Sepúlveda, & Tan, 2013a), the Kullback–Leibler (KL) divergence was explored in optimally compressing the density function of a Preisach operator. The optimization, however, was realized by an exhaustive search due to the particular setting of the Preisach plane.

In this paper we present, to our best knowledge, the first study on the optimal compression of a generalized PI model subject to a given number of play operators. An information-theoretic tool, entropy, is adopted to measure the information loss when representing a group of weighted play operators with a single play. Prior to compression, a scaling operation on the original weights is further introduced to accommodate the fact that, given the same weight value, a generalized play with a larger radius has less impact on the total output. The compression problem is then reformulated as an optimal control problem and subsequently solved with dynamic programming (DP). The computational complexity of the DP algorithm is shown to be much lower than that of exhaustive search.

Extensive simulation results are presented to examine the performance of the proposed compression algorithms, where a generalized PI model with a large number of play operators is treated as “original”. Weight distributions of different forms are explored. Simulation results show that, in general, the entropy-based approaches deliver far better performance than a typically adopted scheme where the play radii are assigned uniformly. The effectiveness of the proposed approach is further verified in the compression of an experimentally identified generalized PI model for the complex hysteretic relationship between temperature and resistance of a promising multifunctional material, vanadium dioxide (VO₂) (Coy, Cabrera, Sepúlveda, & Fernández, 2010; Merced, Cabrera, Davila, Fernández, & Sepúlveda, 2012; Zhang et al., 2014b). Note that the proposed optimal compression approach also works for the classical PI model since it is a special case of the generalized PI model.

The remainder of the paper is organized as follows. In Section 2, the classical and generalized PI models are briefly reviewed. Section 3 presents the optimal compression problem formulation, and the entropy-based optimal compression algorithms. In Section 4, the performance of the compression algorithms is tested in simulation for different forms of weight distributions for the generalized PI model. In Section 5, the compression schemes are further verified using experimental data on a VO₂ film. Finally, concluding remarks are presented in Section 6.

A preliminary version of this paper was presented at the 2013 ASME Dynamic Systems and Control Conference (Zhang, Merced, Sepúlveda, & Tan, 2013b). The enhancements of this paper over (Zhang et al., 2013b) include (1) an improved measure for information loss that incorporates both the entropy of the weight distribution and the effect of spatial distribution of the radii. (2) a proposed method for scaling the weight distribution of a

generalized PI operator prior to compression, and demonstration of its advantage over compression without this scaling step, (3) extensive simulation studies on evaluating the performance of the proposed algorithms for weight distributions of different characteristics, and (4) improved structuring and presentation throughout the paper.

2. Review of PI models

In this section, a brief overview of the classical and generalized PI models are provided. Readers are referred to Al Janaideh et al. (2009), Brokate and Sprekels (1996), Kuhnen (2003) and Visintin (1994) for more details.

2.1. Classical PI model

The classical PI model consists of a weighted superposition of basic play (or stop) operators. As illustrated in Fig. 1(a), the play operator is characterized by its radius r . For a given input function $v(t)$, the output $w(t)$ of a play operator with radius r and initial condition $w(t^-)$ is defined as

$$w(t) = F_r[v](t) = f_r(v(t), F_r[v](t^-)), \quad (1)$$

where

$$f_r(v(t), w(t^-)) = \begin{cases} \max(v(t) - r, w(t^-)), & \text{if } v(t) > v(t^-) \\ \min(v(t) + r, w(t^-)), & \text{if } v(t) < v(t^-) \\ w(t^-), & \text{if } v(t) = v(t^-), \end{cases} \quad (2)$$

and $t^- = \lim_{\epsilon \rightarrow 0, \epsilon > 0} t - \epsilon$.

The output of a classical PI model is expressed as an integral in the following form:

$$y_P(t) = \int_0^R p(r) F_r[v](t) dr, \quad (3)$$

where $p(r)$ is the weighting function of the PI model, which is usually chosen to be non-negative, and R represents the maximum play radius.

For practical implementation, the classical PI model is represented as a weighted summation of a finite number of play operators as follows:

$$y_P(t) = p(r_0)v + \sum_{j=1}^N p(r_j) F_{r_j}[v](t), \quad (4)$$

where $r_j > 0$ is the play radius of the j th play operator, $p(r_j)$ is the corresponding weight, and N denotes the number of play operators.

2.2. Generalized PI model

The generalized PI hysteresis model can capture complex hysteresis loops with both asymmetry and output saturation (Al Janaideh et al., 2009, 2011).

Following a similar treatment as in Al Janaideh et al. (2009, 2011), a generalized play operator with radius r is defined by (see Fig. 1(b))

$$w(t) = F_r^\gamma[v](t) = f_r^\gamma(v(t), F_r^\gamma[v](t^-)), \quad (5)$$

where $f_r^\gamma(t, w(t))$ is defined as

$$f_r^\gamma(v(t), w(t^-)) = \begin{cases} \max(\gamma_L(v(t)) - r, w(t^-)), & \text{if } v(t) > v(t^-) \\ \min(\gamma_R(v(t)) + r, w(t^-)), & \text{if } v(t) < v(t^-) \\ w(t^-), & \text{if } v(t) = v(t^-), \end{cases} \quad (6)$$

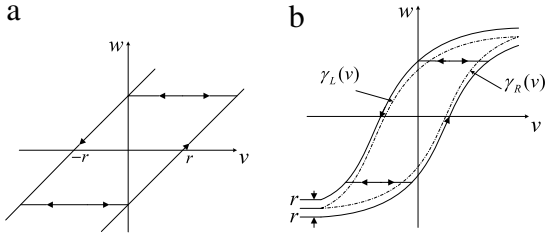


Fig. 1. Input–output relationships of (a) a classical play operator with radius r ; (b) a generalized play operator with radius r (shown as solid curves) and a generalized play operator with radius $r = 0$ (shown as dashed curves).

where $\gamma_L(\cdot)$ and $\gamma_R(\cdot)$ are two envelope functions that are strictly increasing. The envelope functions describe the properties of the play operators. For any radius $r \geq 0$ and input $v(t)$, the condition $\gamma_L(v(t)) + r \geq \gamma_R(v(t)) - r$ needs to be satisfied in order to meet the order preservation property of hysteresis behavior (Iyer & Tan, 2009).

The output of a generalized PI model can be expressed in the integral form as

$$y_P(t) = \int_0^R p(r) F_r^\gamma[v](t) dr. \quad (7)$$

Similar to the classical PI case, a discrete-version of the generalized PI model can be written as

$$y_{P_\gamma}(t) = D(v(t)) + \sum_{j=1}^N p(r_j) F_{r_j}^\gamma[v](t), \quad (8)$$

where $D(\cdot)$ is a non-decreasing Lipschitz continuous function that represents the non-hysteretic component.

When $\gamma_L(v(t)) = \gamma_R(v(t))$, the generalized PI model can be utilized to model symmetric hysteresis; when $D(\cdot)$ is linear and $\gamma_L(v(t)) = \gamma_R(v(t)) = v(t)$, the generalized PI model degenerates to a classical PI model. For this reason, the remainder of the paper deals exclusively with the generalized PI model.

3. Optimal compression: problem statement and algorithms

Note that, for a PI hysteresis operator, its output is a weighted superposition of outputs (states) of individual play operators and hence is linear with respect to the weight parameters. As a result, one can identify the weight parameters offline or online by minimizing the difference between the actual output and the output of the estimated model. On the other hand, the play radii of a PI operator determines the states of individual play operators, but the relationship between the radii and the states are complex and involves the past history of the input, and one cannot express the output of a PI operator directly in terms of its play radii. Consequently, determining the play radii based on the output difference between the original model and the estimated model is difficult if not impossible. Therefore, in this paper we seek to minimize the difference in “weight distribution” (including both play radii and their weights) between the original and reduced PI operators, which would imply that the output of the reduced model will be close to that of the original model, under all input functions.

The number of play operators in the generalized PI model directly determines the computational and storage cost in hysteresis modeling, parameter identification, and inverse compensation. Therefore, it is taken as the measure of complexity for a generalized PI model in this paper. Consequently, the compression of a high-fidelity PI model with a large number (N) of play operators deals with finding a smaller number (M , $M < N$) of play operators and

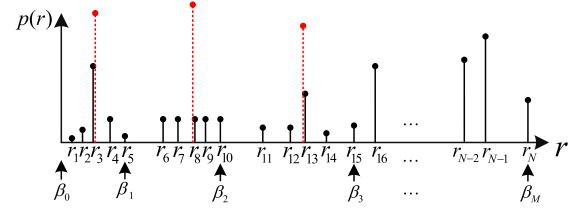


Fig. 2. Schematic illustrating the compression of a weighting function. The solid-line segments are the original weighting function, and the dotted-line segments are the new weighting function. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the corresponding weights to best represent the original PI model. This problem is closely tied to optimal compression of the weight vector $\{p(r_j)\}_{j=1}^N$ in the discrete case, which is formulated precisely and solved in this section.

3.1. Formulation of the weight compression problem

Fig. 2 shows a nonnegative weighting function $p(r)$ with N elements, $p_j = p(r_j)$, $0 \leq r_1 < r_2 < \dots < r_N < \infty$. The compression of the original weighting function is to use a new weighting function with $(M, M < N)$ elements: $\hat{p}_j = \hat{p}(\hat{r}_j)$, $0 \leq \hat{r}_1 \leq \hat{r}_2 \leq \dots \leq \hat{r}_M < \infty$ to approximate the original weighting function.

In order to compress the weighting function, the notion of partition is firstly introduced. Denote $\mathcal{D} = \{\beta_k\}_{k=0}^M$ as the set of partition indices ($0 = \beta_0 < \beta_1 < \dots < \beta_{M-1} < \beta_M = N$), that partitions the weighting function into M groups. The original weights with indices $\beta_{k-1} + 1, \beta_{k-1} + 2, \dots, \beta_k$ belong in the k -th group, $k = 1, 2, \dots, M$. For each group, the original weighting function is approximated with only one element (shown in Fig. 2 as red dotted segments), and the new element is characterized as

$$\hat{r}_k = \frac{\sum_{i=\beta_{k-1}+1}^{\beta_k} p(r_i)}{\sum_{j=\beta_{k-1}+1}^{\beta_k} p(r_j)} r_i, \quad (9)$$

$$\hat{p}(\hat{r}_k) = \sum_{i=\beta_{k-1}+1}^{\beta_k} p(r_i).$$

The optimal compression problem is to find the compression strategy $\mathcal{D}^* = \{\beta_k^*\}_{k=0}^M$ that best approximates the given weighting function. To facilitate the formulation of the problem, we consider a function F_k as the information loss measure in approximating the distribution $p(r_i)$, $\beta_{k-1} + 1 \leq i \leq \beta_k$ with $\hat{p}(\hat{r}_k)$. The overall compression cost function can be generally chosen as either $\sum_{k=1}^M F_k$ or $\max_k F_k$.

3.2. Optimal compression algorithm

While a number of methods, such as evolutionary algorithms (Goldberg, 1989) and simulated annealing (Kirkpatrick, Gelatt, & Vecchi, 1983), could be used to solve nonlinear optimization problems, these approaches are typically computation-intensive and cannot guarantee globally optimal solutions. In this paper, we exploit the structure of the compression problem and reformulate it as an optimal control problem. The reformulation allows us to use dynamic programming to obtain the (globally) optimal solution, as well as analyze the complexity of the algorithm. Denote $x_k = \beta_k$ as the state, and $u_k = \beta_k - \beta_{k-1}$ as the control input, $k = 1, \dots, M$. The optimization problem is then reformulated as: finding inputs $\mathbf{u} = (u_1, u_2, \dots, u_{M-1})$, such that the total cost is minimized. Note that since $\beta_0 = 0$ and $\beta_M = N$ are

fixed, u_M will be determined automatically by \mathbf{u} and thus is not a decision variable. The compression cost F_k for each group is clearly determined by β_{k-1} and u_k , or $F_k = F_k(x_{k-1}, u_k)$. The dynamic programming algorithm to be presented next considers the overall cost function J_1 with the form of $\sum_{i=1}^M F_i$. The algorithm is similar for the case when the cost function is in the form of $\max_i F_i$. Specifically, we have

$$x_k = x_{k-1} + u_k, \\ J_1(x_0, \mathbf{u}) = \sum_{i=1}^{M-1} F_i(x_{i-1}, u_i) + f(x_{M-1}), \quad (10)$$

where $f(x_{M-1})$ represents the “terminal cost”—the information loss for the last group. The optimal control $\mathbf{u}^* = (u_1^*, u_2^*, \dots, u_{M-1}^*)$ is defined as

$$\mathbf{u}^* = \arg \min_{\mathbf{u}} J_1(x_0, \mathbf{u}). \quad (11)$$

Once the optimal control \mathbf{u}^* is found, the optimal compression strategy is obtained as: $\beta_0^* = 0, \beta_1^* = \beta_0^* + u_1^*, \dots, \beta_k^* = \beta_{k-1}^* + u_k^*, \dots, \beta_M^* = N$. The following proposition provides the algorithm for finding \mathbf{u}^* , the proof of which follows the standard dynamic programming principle (Bertsekas, 2005).

Proposition 1. Consider a sequence of $M-1$ optimization problems, with the cost functions defined as

$$J_k(x_{k-1}, \{u_i\}_{i=k}^{M-1}) = \sum_{i=k}^{M-1} F_i(x_{i-1}, u_i) + f(x_{M-1}), \quad (12)$$

$k = 1, \dots, M-1$, and the corresponding value function as

$$V_k(x_{k-1}) = \min_{\{u_i\}_{i=k}^{M-1}} J_k(x_{k-1}, \{u_i\}_{i=k}^{M-1}). \quad (13)$$

Then the value functions along with the optimal control sequence $\{u_i^*\}$ can be obtained recursively as follows:

$$V_{M-1}(x_{M-2}) = \min_{u_{M-1}} F_{M-1}(x_{M-2}, u_{M-1}) + f(x_{M-2} + u_{M-1}), \quad (14)$$

...

$$V_k(x_{k-1}) = \min_{u_k} F_k(x_{k-1}, u_k) + V_{k+1}(x_{k-1} + u_k), \quad (15)$$

$k = M-2, \dots, 2, 1$, and u_k^* is obtained as the minimizing u_k in the computation of $V_k(x_{k-1})$, $k = 1, \dots, M-1$.

Remark 1. Note that the procedure in Proposition 1 will generate $\{u_k^*\}$ as a state-dependent policy. The original optimization problem has a fixed initial state of $x_0 = 0$, which results in a specific optimal control sequence when applied to the feedback policy.

The dynamic programming approach has a significant advantage over the exhaustive search in terms of computational complexity. Take the number of evaluations of information loss in partitioned groups required by each algorithm as the metric of computational complexity. For the dynamic programming approach, the terminal cost function $f(x_{M-1})$ requires N evaluations since x_{M-1} could take any values of $\{0, 1, \dots, N-1\}$, (14) requires $N-1$ evaluations, and (15) requires $N-M+k$ evaluations, $1 \leq k \leq M-2$, resulting in a total of $\frac{M(2N-M+1)}{2}$ evaluations. For the exhaustive search, on the other hand, there are $\frac{(N-1)!}{(M-1)!(N-M)!}$ possible partitions for the weights, and each partition requires M evaluations, resulting in a total of $\frac{(N-1)!M}{(M-1)!(N-M)!}$ for the number of evaluations, which is significantly larger than the complexity of the dynamic program algorithm, as will be further demonstrated in Section 4.5.

3.3. Information loss metrics: entropy-based measure

The discussions so far have assumed a generic function F_k that represents the information loss in replacing the weight distribution of the k th group, $\{p(r_i)\}_{i=\beta_{k-1}+1}^{\beta_k}$, with a single weight $\hat{p}_k(\hat{r}_k)$. In this subsection, an information-theoretic tool, entropy, is exploited for defining the information loss in compression. Entropy (Shannon, 1948) is a measure of the uncertainty in a random variable, which have been used extensively in statistics (Balakrishnan & Toubia, 2007) and signal processing (Wu & Hsu, 2000).

For a discrete random variable R with pmf $\bar{p}(r_i)$, $i = 1, 2, \dots, L$, the entropy is defined as

$$H(R) = - \sum_{i=1}^L \bar{p}(r_i) \log(\bar{p}(r_i)). \quad (16)$$

The convention $0 \log 0 = 0$ is adopted. For a given L , the entropy of R is lowest when there exists a $k \leq L$, such that $\bar{p}(r_k) = 1$. On the other hand, the uniform distribution, where $\bar{p}(r_i) = 1/L$, $i = 1, 2, \dots, L$, has the largest entropy.

Intuitively, if the weight distribution of (multiple) play operators, when properly normalized, is close to a uniform distribution, the compression loss is high; Conversely, if the group has a single operator with weight dominantly larger than those of the rest operators, the compression loss is expected to be small. These considerations make the entropy a natural candidate for measuring the information loss. In addition, if the dominant play operators are located far away from \hat{r}_k , the compression loss is also high, motivating the incorporation of the distances between the play radii and the “centroid” \hat{r}_k into the cost function. Specifically, the following procedure is proposed to compute an entropy-based measure for the information loss in approximating a discrete distribution group $p(r_i)$, $i = \beta_{k-1} + 1, \beta_{k-1} + 2, \dots, \beta_k$:

(1) Calculate the total weight in the group:

$$T_k = \sum_{i=\beta_{k-1}+1}^{\beta_k} p(r_i).$$

(2) Get the normalized pmf for the group:

$$\bar{p}_i = p(r_i)/T_k, \\ i = \beta_{k-1} + 1, \beta_{k-1} + 2, \dots, \beta_k.$$

(3) Obtain the entropy for the normalized pmf:

$$H_k = - \sum_{i=\beta_{k-1}+1}^{\beta_k} \bar{p}_i \log \bar{p}_i.$$

(4) The effect of the distances between play radii and the centroid needs to be incorporated; one way to do this is to define the cost function for the k -th group as $\mathcal{E}_k = T_k \cdot \sqrt{\sum_{i=\beta_{k-1}+1}^{\beta_k} (\bar{p}_i (r_i - \hat{r}_k))^2} \cdot H_k$,

where T_k is included to reflect the impact of the total weight for the group. Note that while there might be other alternatives, we will show later in the paper that the proposed scheme is adequately effective. Finally, for a partition strategy \mathcal{D} , the entropy-based overall cost functions can be chosen as:

$$J_{\mathcal{E}_{\text{SUM}}}(\mathcal{D}) = \sum_{k=1}^M \mathcal{E}_k, \quad (17)$$

$$J_{\mathcal{E}_{\text{MAX}}}(\mathcal{D}) = \max_k \mathcal{E}_k. \quad (18)$$

The optimization algorithms based on the cost functions (17) and (18) are denoted as Entropy Sum and Entropy Max, respectively, for the remainder of this paper.

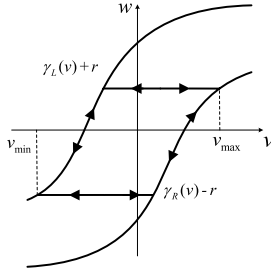


Fig. 3. Illustration of the radius-dependent output range for a generalized play operator.

3.4. Scaling of the weights for a generalized PI model

For a generalized PI model with a certain input range, the constituent play operators will have different ranges of outputs, and thus have different levels of importance to the output of the PI operator even if their weights are equal. In this subsection, proper “scaling” of the weighting function is introduced to accommodate the play radius-dependent importance.

Fig. 3 shows a generalized play operator with radius r , where the input range is $[v_{\min}, v_{\max}]$, and the initial condition $w(0) = \gamma_L(v_{\min}) + r$. It can be easily seen that the output range of the play operator is dependent on r ; specifically, the output $w \in [\gamma_L(v_{\min}) + r, \gamma_R(v_{\max}) - r]$, with a total change of $\gamma_R(v_{\max}) - \gamma_L(v_{\min}) - 2r$. Accordingly, the following scheme is introduced to produce a “scaled” weight distribution for the compression.

Denote the actual weight as p , and the weight after scaling as p' , then for the play operator with radius r_j :

$$p'(r_j) = \frac{\gamma_R(v_{\max}) - \gamma_L(v_{\min}) - 2r_j}{2} * p(r_j). \quad (19)$$

For a generalized play operator whose envelopes are in the form of hyperbolic-tangent functions, when $v_{\min} \rightarrow -\infty$ and $u_{\max} \rightarrow +\infty$, the output $z \in [-1 + r, 1 - r]$. It can be seen that $0 \leq r \leq 1$. The play operator will not produce any output change under a cyclic input when the radius $r > 1$, since the output will never reach both envelopes due to the disjoint ranges of the envelopes. It is for this reason that the radius is always chosen to be no larger than 1 in the remainder of this paper. The advantage of using the scaled weights over the non-scaled weights in compression will be further demonstrated in Section 5.2.

4. Simulation results

In this section, the proposed optimal compression algorithms are tested in simulation for PI models with different characteristics for their scaled weighting functions.

Following AlJanaideh et al. (2009, 2011), the envelope functions for the generalized play operator are chosen to be hyperbolic-tangent functions in the form of

$$\gamma_R(v(t)) = \tanh(a_R v(t) + b_R), \quad (20)$$

$$\gamma_L(v(t)) = \tanh(a_L v(t) + b_L). \quad (21)$$

For simplicity of demonstration, the non-hysteretic component $D(v(t))$ is set to be zero. The parameter values of the generalized play operator are shown in Table 1. The original generalized PI model consists of $N = 30$ play operators, with radii $r_j = j/(N + 1)$, $j = 1, 2, \dots, N$, and input range of $v \in [-1, 1]$.

The compression goal is to use a new generalized PI model with $M = 6$ play operators to approximate the original generalized PI model. Although the unscaled weights are directly related to the output, the scaled weight distribution is considered based on the discussion in Section 3. The output performance of the proposed

Table 1

Parameters of the PI generalized model envelope functions.

a_R	b_R	a_L	b_L
3.5	0	4.5	1

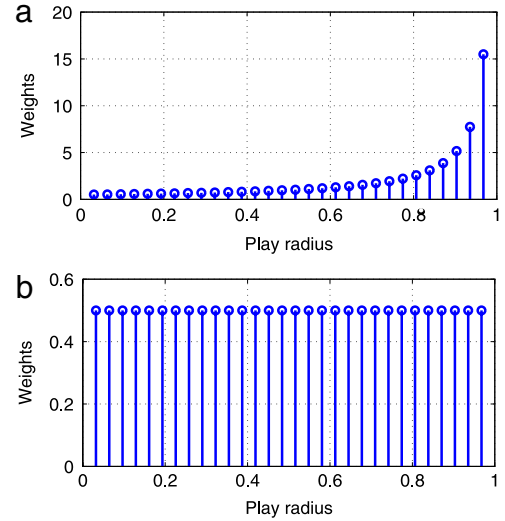


Fig. 4. Weighting function (uniform case) of the generalized PI model: (a) Unscaled. (b) Scaled.

approach will be discussed in Section 4.6. Four cases for the scaled weight distribution of the original model are considered, (1) uniform, (2) one peak, (3) two peaks, and (4) random. In addition to the two compression schemes presented in the previous section (Entropy Sum, Entropy Max), a uniform compression scheme, where every five consecutive play operators are clustered into one group, is considered for comparison purposes.

In order to assess the output prediction performance of the reduced model, throughout the paper, the normalized root-mean-square error (RMSE) is adopted to quantify the modeling performance under different compression strategies. The error is obtained as follows: first, calculate the RMSE between the output of each new generalized PI model and that of the original generalized PI model under the input shown in Fig. 5(a); then divide the RMSE by the output range of the original model. Normalization of the RMSE allows assessing and comparing the algorithms' performance across different weight distributions.

4.1. Case 1: uniform distribution for the scaled weights

First, the following uniform distribution is considered for the scaled weights:

$$p'(r_j) = 0.5, \quad j = 1, 2, \dots, 30. \quad (22)$$

Fig. 4(a) shows the actual weight distribution (unscaled) and Fig. 4(b) shows the corresponding scaled weight distribution obtained based on Eq. (19). Fig. 5(a) shows an input sequence and Fig. 5(b) shows the input–output relationship of the given generalized PI model. Note that the actual weighting functions and the input–output relationships will not be shown for the other simulation examples in the remainder of the paper in the interest of brevity; however, the hysteresis loops in other cases are also verified to be large.

The simulation results are summarized in Table 2. It is shown that, given the uniform distribution, the two entropy-based algorithms are able to compress the distribution uniformly, and generate desirable performance.

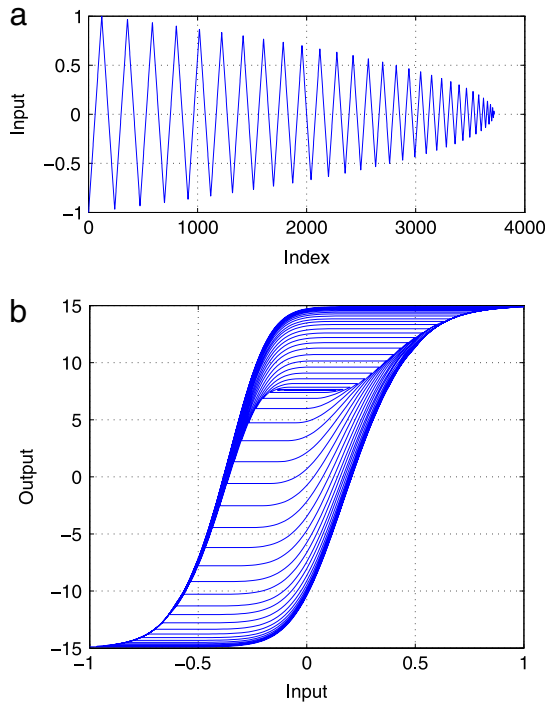


Fig. 5. (a) Input sequence. (b) Input vs. output for generalized PI model with uniform weight function.

Table 2
Compression performance comparison: the uniform case.

Scheme	Partition indices	Error
Uniform	(0, 5, 10, 15, 20, 25, 30)	0.75%
Entropy Sum	(0, 5, 10, 15, 20, 25, 30)	0.75%
Entropy Max	(0, 5, 10, 15, 20, 25, 30)	0.75%

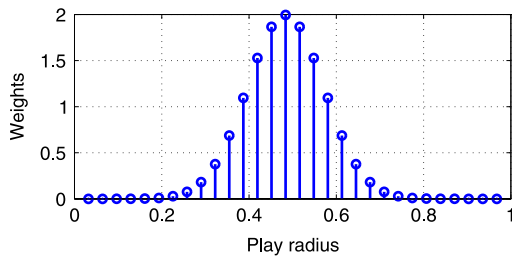


Fig. 6. Scaled weighting function (one-peak case).

4.2. Case 2: scaled weights with one prominent peak

In the second case, the scaled weight distribution is assumed to have one peak, expressed as:

$$p'(r_j) = \frac{5}{\sqrt{2\pi}} \exp\left(-\frac{(j-15)^2}{15}\right), \quad j = 1, 2, \dots, 30. \quad (23)$$

Fig. 6 shows the scaled weight distribution. Table 3 shows the compression performances based on different approaches. From the simulation results, it is seen that Entropy Sum and Entropy Max approaches are able to generate considerably better performance than the uniform compression scheme. The peak of the original weighting function is in the middle region; the simulation results show that the entropy approaches partition the weights densely in the middle region (with many groups having only one or two elements).

Table 3
Compression performance comparison: the case of one peak.

Scheme	Partition indices	Error
Uniform	(0, 5, 10, 15, 20, 25, 30)	0.34%
Entropy Sum	(0, 11, 13, 14, 16, 18, 30)	0.08%
Entropy Max	(0, 9, 12, 14, 16, 18, 30)	0.14%

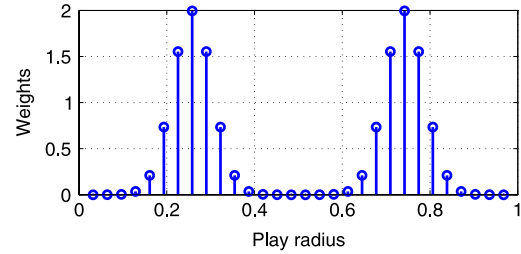


Fig. 7. Scaled weighting function (two-peak case).

Table 4
Compression performance comparison: the case of two peaks.

Scheme	Partition indices	Error
Uniform	(0, 5, 10, 15, 20, 25, 30)	0.42%
Entropy Sum	(0, 7, 8, 14, 22, 23, 30)	0.27%
Entropy Max	(0, 6, 8, 11, 21, 23, 30)	0.20%

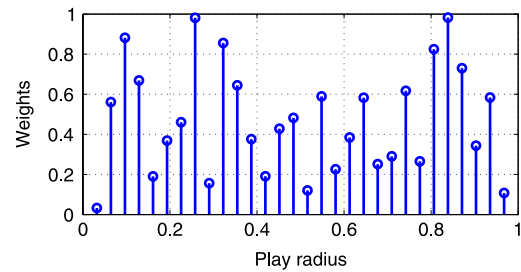


Fig. 8. Scaled weighting function (the random case).

4.3. Case 3: scaled weights with two prominent peaks

In the third case, the scaled weighting function has two peaks, expressed as:

$$p'(r_j) = \begin{cases} \frac{5}{\sqrt{2\pi}} \exp\left(-\frac{(j-8)^2}{8}\right), & j = 1, 2, \dots, 16 \\ \frac{5}{\sqrt{2\pi}} \exp\left(-\frac{(j-23)^2}{8}\right), & j = 17, 18, \dots, 30. \end{cases} \quad (24)$$

Fig. 7 shows the scaled weight distribution. Table 4 shows the compression performances based on the different compression approaches. From the simulation results, both of the proposed approaches show very good performance, with about 40% less error comparing to the uniform compression scheme.

4.4. Case 4: random distribution for the scaled weights

Finally, we consider the case where the scaled weighting function has a random distribution as shown in Fig. 8. Table 5 lists the corresponding simulation results. It can be seen that, under a random distribution, the entropy approaches compress the distribution almost uniformly, with slightly better performance than the uniform compression scheme. A random distribution is similarly difficult as a uniform distribution to compress, since there are usually no particular patterns that facilitate compression.

From the simulation results, overall, both of the proposed approaches show good compression performance. When the pattern

Table 5
Compression performance comparison: the case of random distribution.

Scheme	Partition indices	Error
Uniform	(0, 5, 10, 15, 20, 25, 30)	0.53%
Entropy Sum	(0, 5, 9, 13, 18, 24, 30)	0.54%
Entropy Max	(0, 6, 10, 15, 20, 25, 30)	0.53%

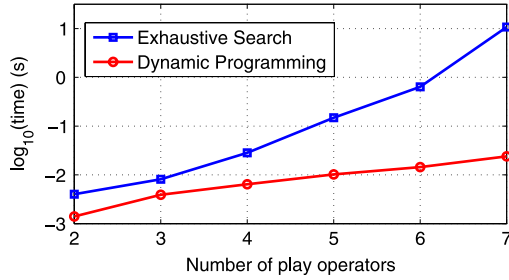


Fig. 9. Comparison of average optimization time. Note the log scale.

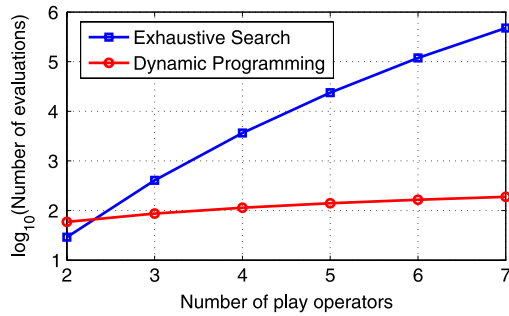


Fig. 10. Comparison of the number of information loss evaluations. Note the log scale.

of the (scaled) weighting function is uniform or random, the optimal compression almost degenerates to uniform compression, and the compression error is larger comparing with other cases that have more features (peaks).

4.5. Computational time for the algorithms

The computational time of the dynamic programming-based optimization is also compared with that using exhaustive search. Due to the similar optimization process under different cost function candidates, only Entropy Sum is considered in this comparison. A generalized PI model with $N = 30$ is used, which has a scaled weight distribution as used in Section 4.2. The computations are run in Matlab on a computer Lenovo Thinkpad T420 with 2.80 GHz CPU and 4.00 GB memory.

In order to compare the computational efficiency, the dynamic programming-based algorithm and the exhaustive search-based algorithm are run 10 times for each setting of M , which is varied from 2 to 7 in this study. The average running times among the 10 times are shown in Fig. 9. It can be seen that, when N is fixed, the time cost under dynamic programming grows much slower than the exhaustive search when M is increased. These results agree well with the complexity analysis in Section 3.2, as shown in Fig. 10, which plots the number of information loss evaluations for the dynamic programming and the exhaustive search methods, respectively. The computational advantage of the dynamic programming approach is evident.

4.6. Comparison with a traditional model identification approach

The effectiveness of the proposed optimal compression approach is further compared with a traditional model identification scheme (referred to as “output optimization” in this paper),

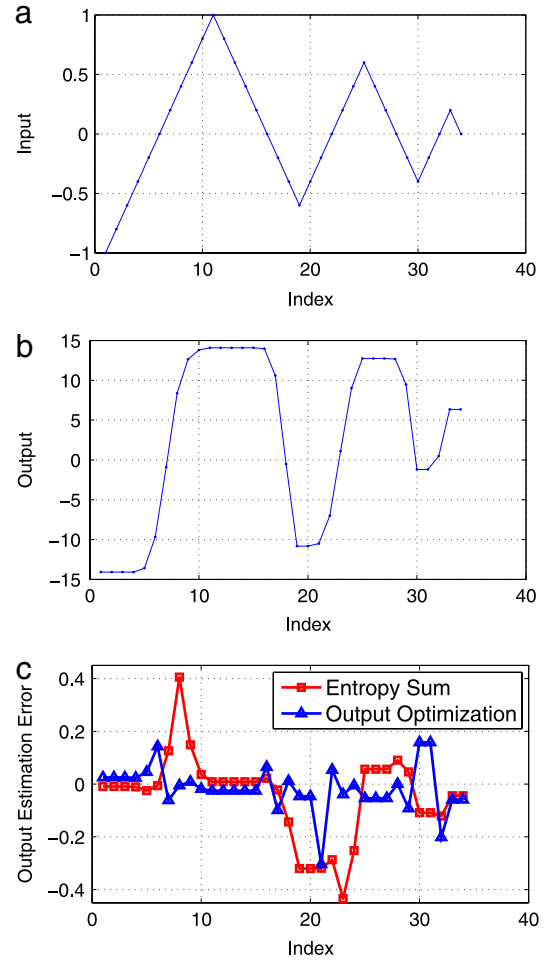


Fig. 11. (a) A third-order reversal input sequence. (b) The corresponding output sequence. (c) The output prediction error between the Entropy Sum approach and output optimization approach.

where a model with the same complexity (six generalized play operators) is determined by minimizing the output error under a given training input. While there are infinite number of possible choices for the training input, a third-order reversal input sequence (shown in Fig. 11(a)) is adopted as a representative example. An extensive search within all possible parameterizations of the 6 play operators are conducted in Matlab using the function *fmincon*, to match the output of the original model with 30 plays. The scaled weighting function for the original model has the same random case as shown in Fig. 8 and the corresponding output sequence is shown in Fig. 11(b). Fig. 11(c) shows the corresponding output prediction error under the Entropy Sum approach and the output optimization approach. The RMSE errors of the Entropy Sum approach and the output optimization approach are 0.165 and 0.088, respectively. While that latter indicates the output optimization approach could deliver better performance for a given input sequence, Fig. 12 shows that the proposed approach is more robust in output prediction with respect to input variability. In particular, simulation is run 50 times with different, randomly generated input sequences and the corresponding output prediction performance is recorded. Fig. 12(a) and (b) shows one example for the 50 cases, while Fig. 12(c) summarizes the error statistics over the 50 runs.

5. Experimental results

VO₂ is an interesting class of smart materials with a myriad of microactuation, optical, and memory applications. It undergoes an

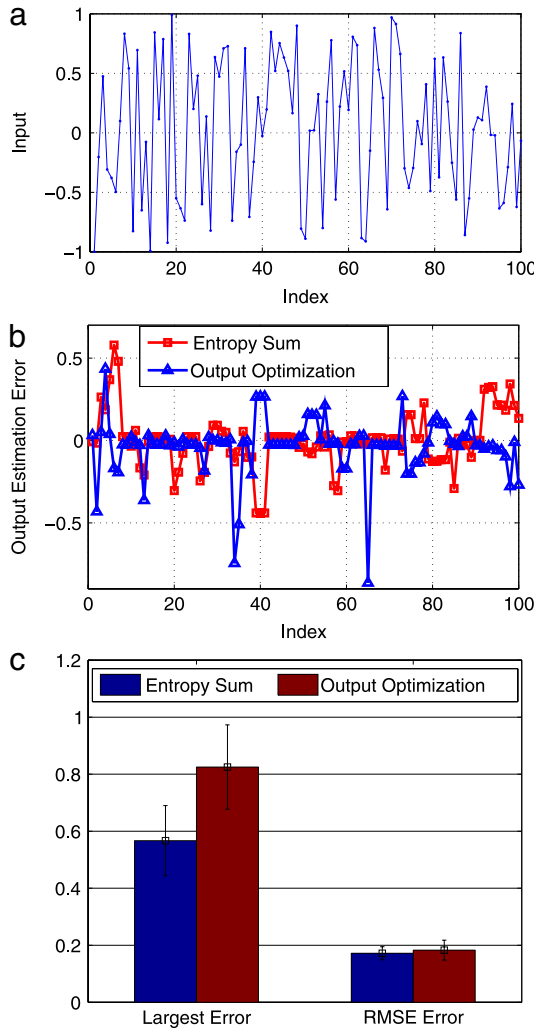


Fig. 12. (a) A random input sequence, and (b) the corresponding output prediction error performance. (c) The output prediction performance based on 50 random input sequences.

insulator-to-metal transition (IMT) at around 68 °C, during which multiple properties of VO₂ (including resistance Merced et al., 2012, induced mechanical stress Zhang et al., 2014b, and optical transmittance Coy et al., 2010) demonstrate pronounced hysteresis with respect to temperature. The hysteresis between the resistance and the temperature in a VO₂ film is used as an example to illustrate the effectiveness of the proposed compression schemes for a generalized PI model.

A VO₂ layer was deposited by pulsed laser deposition following similar treatment as in Zhang et al. (2014b). The film was glued with a highly thermal conductive silver paint, and in close contact with a Peltier heater. The Peltier heater was regulated with a temperature controller with 0.1 °C precision. The experimental setup in this work is similar to the one used in Zhang et al. (2013a) (shown in Fig. 13). The resistance of the film was measured through two aluminum contacts patterned on the VO₂ film.

Since the measured resistance (R) changes by approximately two orders of magnitude during the phase transition, $-\log_{10} R$ is taken as the output, where the negative sign is introduced so that the resulting play operators of the PI model have non-negative weights. The waiting time at each temperature before the resistance measurement is taken to be 8 s to ensure that the thermal steady state has been reached (Zhang et al., 2014b).

In order to get desirable modeling performance, the original generalized PI model has $N = 30$ play operators. Similarly, their

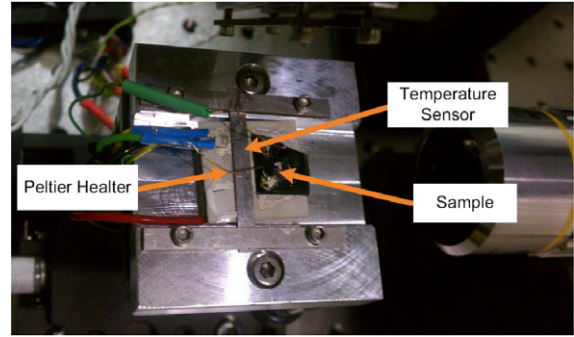


Fig. 13. Experimental setup for resistance versus temperature measurement of a VO₂ film (Zhang et al., 2013a).

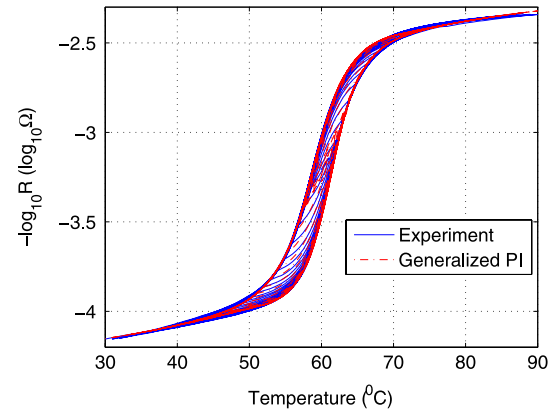


Fig. 14. The performance of a generalized PI model (30 plays) in modeling of the resistance-temperature hysteresis in VO₂.

Table 6

Identified parameters of the envelope functions.

a_R	b_R	a_L	b_L	a_D	b_D	d
0.201	-11.58	0.16	-10.26	0.03	-1.61	-3.26

play radii are $r_j = j/N$, $j = 1, 2, \dots, N$, respectively, and the envelope functions for the generalized play operator are chosen to be hyperbolic-tangent functions in the same form of (21) and (22).

The non-hysteretic component $D(v(t))$ is chosen to be

$$D(v(t)) = \tanh(a_D v(t) + b_D) + d. \quad (25)$$

The full range of temperature input is [30, 90]°C. The hysteresis behavior shown in Fig. 14 is asymmetric and partially saturated. The generalized PI model is identified based on the approach in Al Janaideh et al. (2009). The effectiveness of the generalized PI model is verified in Fig. 14. Table 6 and Fig. 15 show the identified parameters for the envelope functions and the weights of the generalized play operators, respectively. Fig. 16 shows the weight after scaling based on the actual weight. The weights present a non-uniform distribution.

5.1. Compression performance

To conduct the compression studies, the identified generalized PI model is taken as the “original” distribution. The modeling performance is analyzed in the same manner as shown in the previous section between a compressed models and the experimental data. The new generalized PI model consists of $M = 5$ play operators. The partition scheme under uniform compression was {0, 6, 12, 18, 24, 30}. Fig. 17 shows the play radii and weights for the M play operators. Uniform compression fails to accommodate

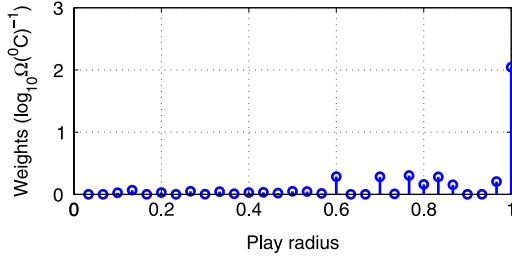


Fig. 15. Identified weights for all the play operators of the generalized PI model.

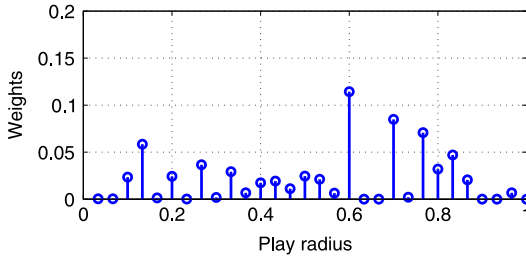


Fig. 16. The scaled weights for the generalized PI operators.

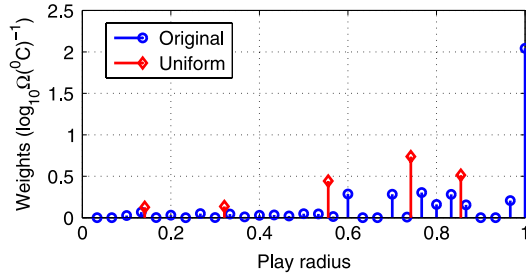


Fig. 17. Parameters of the compressed generalized PI model: uniform compression.

the weighting distribution, with an RMSE of 1.10% for the modeling error (the difference between the model output and the experimental data).

The partition scheme under Entropy Sum was {0, 7, 14, 20, 23, 30}, and that under Entropy Max was {0, 4, 10, 17, 22, 30}. Fig. 18(a), (b) show the compressed play operator radii and weights based on Entropy Sum and Entropy Max, respectively. Both schemes worked much better than the uniform compression case, with RMSE values of 0.73% and 0.76%, respectively, which were about 32% smaller than that in the uniform case.

5.2. Model verification

In order to further validate the proposed approach, a randomly chosen temperature input sequence, shown in Fig. 19(a), was applied to the VO₂ film, and the corresponding resistance output was measured as shown in Fig. 19(b). Predictions of the resistance output were obtained based on the same compressed generalized PI models in Section 5.1 and corresponding compressed generalized PI models without considering the scaling before the compression. The corresponding estimation errors were calculated and shown in Fig. 19(c) and Table 7.

The model verification experiments further demonstrate that the proposed compression schemes outperform the uniform compression. In Table 7, the modeling performance without considering the scaling effect (Zhang et al., 2013b) is also included. It is evident that the performance improves with proposed scaling strategy; for the uniform, Entropy Sum compression schemes, the results have improve about 10%–20% with the scaling.

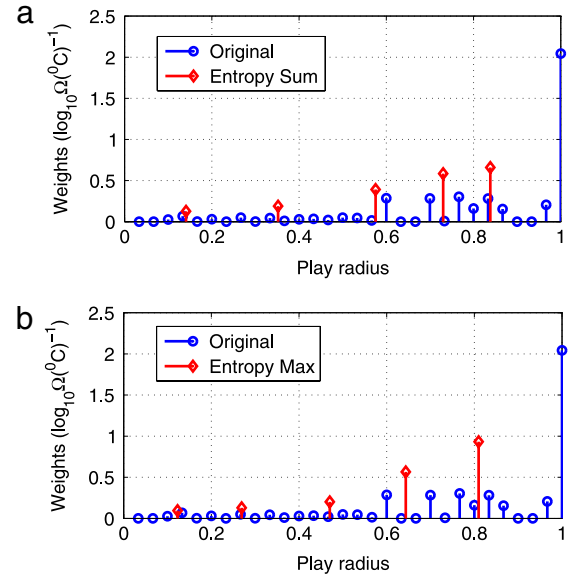


Fig. 18. Parameters of the compressed generalized PI model: (a). Entropy Sum. (b) Entropy Max.

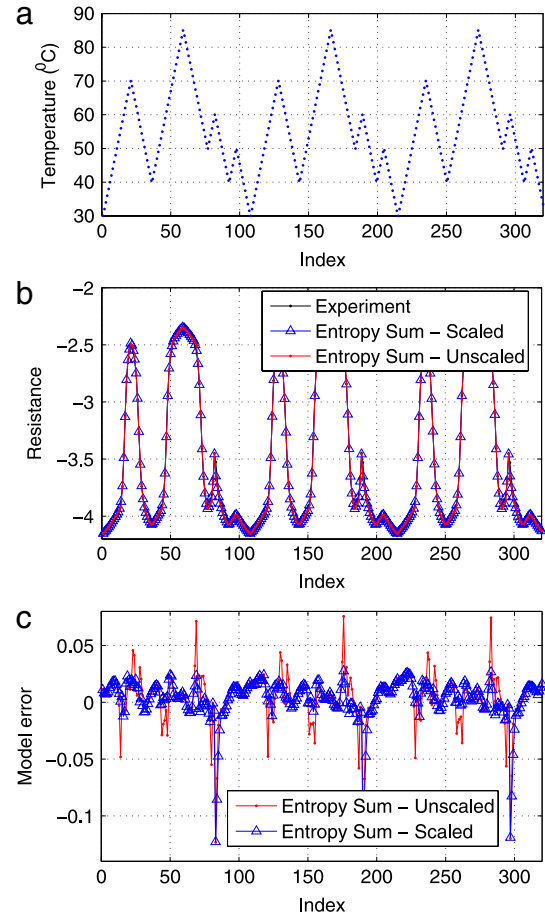


Fig. 19. (a) A new temperature input sequence for model verification. (b) Corresponding output sequence. (c) The output prediction error comparison of Entropy Sum Unscaled approach and the Entropy Sum Scaled approach.

6. Conclusion

In this paper the problem of optimally compressing a generalized PI model was formulated and solved with dynamic

Table 7
Modeling verification error comparison.

Scheme	Non-scaled	Scaled
Uniform	1.76%	1.45%
Entropy Sum	1.21%	1.05%
Entropy Max	0.87%	0.72%

programming, which was shown to have significant computational advantage over the exhaustive search strategy. In particular, two cost functions candidates based on entropy were proposed and examined both using simulation with different types of weighting functions and using experimental data that were collected on a VO₂ film. The results demonstrated that the proposed approach outperforms a traditional uniform scheme, often by a large margin.

References

- Al Janaideh, M., Rakheja, S., & Su, C. Y. (2009). A generalized Prandtl–Ishlinskii model for characterizing the hysteresis and saturation nonlinearities of smart actuators. *Smart Materials and Structures*, 18(4).
- Al Janaideh, M., Rakheja, S., & Su, C. Y. (2011). An analytical generalized Prandtl–Ishlinskii model inversion for hysteresis compensation in micropositioning control. *IEEE/ASME Transactions on Mechatronics*, 16(4), 734–744.
- Balakrishnan, K. J., & Touba, N. A. (2007). Relationship between entropy and test data compression. *IEEE Transactions on Computer-Aided Design*, 26(2), 386–395.
- Bertsekas, D. P. (2005). *Athena Scientific optimization and computation series: Vol. 1. Dynamic programming and optimal control: volume I*. Athena Scientific.
- Brokate, M., & Sprekels, J. (1996). *Hysteresis and phase transitions*. Springer.
- Coy, H., Cabrera, R., Sepúlveda, N., & Fernández, F. E. (2010). Optoelectronic and all-optical multiple memory states in vanadium dioxide. *Journal of Applied Physics*, 108(11).
- Esbrouk, A., Tan, Xiaobo, & Khalil, H. K. (2013). Control of systems with hysteresis via servocompensation and its application to nanopositioning. *IEEE Transactions on Control Systems Technology*, 21(3), 725–738.
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.
- Harrison, R. G. (2003). A physical model of spin ferromagnetism. *IEEE Transactions on Magnetics*, 39(2), 950–960.
- Ikhouane, F., & Rodellar, J. (2006). A linear controller for hysteretic systems. *IEEE Transactions on Automatic Control*, 51(2), 340–344.
- Iyer, R. V., & Tan, Xiaobo (2009). Control of hysteretic systems through inverse compensation. *IEEE Control Systems Magazine*, 29(1), 83–99.
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671–680.
- Krejci, P., & Kuhnen, K. (2001). Inverse control of systems with hysteresis and creep. *IEE Proceedings—Control Theory and Applications*, 148(3), 185–192.
- Kuhnen, K. (2003). Modeling, identification and compensation of complex hysteretic nonlinearities: A modified Prandtl–Ishlinskii approach. *European Journal of Control*, 9(4), 407–418.
- Merced, E., Cabrera, R., Davila, N., Fernández, F. E., & Sepúlveda, N. (2012). A micro-mechanical resonator with programmable frequency capability. *Smart Materials and Structures*, 21(3).
- Oh, J. H., & Bernstein, D. S. (2005). Semilinear Duhem model for rate-independent and rate-dependent hysteresis. *IEEE Transactions on Automatic Control*, 50(5), 631–645.
- Shannon, C. E. (1948). A mathematical theory of communication. *AT&T Technical Journal*, 27, 379–423. 623–656.
- Song, G., Zhao, J. Q., Zhou, X. Q., & de Abreu-Garcia, J. A. (2005). Tracking control of a piezoceramic actuator with hysteresis compensation using inverse Preisach model. *IEEE/ASME Transactions on Mechatronics*, 10(2), 198–209.
- Tan, X., & Baras, J. S. (2004). Modeling and control of hysteresis in magnetostrictive actuators. *Automatica*, 40(9), 1469–1480.
- Visintin, A. (1994). *Differential models of hysteresis*. Springer.
- Vo-Minh, T., Tjahjowidodo, T., Ramon, H., & Van Brussel, H. (2011). A new approach to modeling hysteresis in a pneumatic artificial muscle using the Maxwell-slip model. *IEEE/ASME Transactions on Mechatronics*, 16(1), 177–186.
- Wang, Q. Q., & Su, C. Y. (2006). Robust adaptive control of a class of nonlinear systems including actuator hysteresis with Prandtl–Ishlinskii presentations. *Automatica*, 42(5), 859–867.
- Wu, B. F., & Hsu, H. H. (2000). Entropy-constrained scalar quantization and minimum entropy with error bound by discrete wavelet transforms in image compression. *IEEE Transactions on Signal Processing*, 48(4), 1133–1143.
- Zhang, J., Merced, E., Sepúlveda, N., & Tan, X. (2013a). Kullback–Leibler divergence-based optimal compression of Preisach operator in hysteresis modeling. In *Proceedings of the 2013 American control conference*, Washington, DC (pp. 89–94).
- Zhang, J., Merced, E., Sepúlveda, N., & Tan, X. (2013b). Optimal compression of a generalized Prandtl–Ishlinskii operator in hysteresis modeling. In *Proceedings of the 2013 ASME dynamic systems and control conference*, Palo Alto, CA, Paper DSCC2013-3969.
- Zhang, J., Merced, E., Sepúlveda, N., & Tan, X. (2014a). Modeling and inverse compensation of hysteresis in vanadium dioxide using an extended generalized Prandtl–Ishlinskii model. *Smart Materials and Structures*, 23(12), 125017.
- Zhang, J., Merced, E., Sepúlveda, N., & Tan, X. (2014b). Modeling and inverse compensation of non-monotonic hysteresis in VO₂ coated microactuators. *IEEE/ASME Transactions on Mechatronics*, 19(2), 579–588.



Jun Zhang received the B.S. degree in automation from the University of Science and Technology of China, Hefei, China, in 2011. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, USA. His research interests include modeling and control of smart materials.

He received the Student Best Paper Competition Award at the ASME 2012 Conference on Smart Materials, Adaptive Structures and Intelligent Systems, and the Best Conference Paper in Application Award at the ASME 2013 Dynamic Systems and Control Conference, and was named the Electrical Engineering Outstanding Graduate Student for 2014–2015.



Emmanuelle Merced received the B.Sc. and M.Sc. degrees in Electrical Engineering from the University of Puerto Rico, Mayagüez, Puerto Rico, in 2009 and 2011, respectively. He received the Ph.D. degree in Electrical Engineering from Michigan State University in 2014. He is currently a research engineer specialist at Hewlett-Packard Labs. He has participated in numerous research and industry related programs including: Memristor group intern at Hewlett-Packard Laboratories, Palo Alto, CA (2013), independent contractor at Universal Technology Corporation, Dayton, OH (2011), research intern at the

United States Army Corps of Engineers, Vicksburg, MS (2009), and research intern at the Power System and Automation Laboratory at Texas A&M University, College Station, TX (2008), among others.

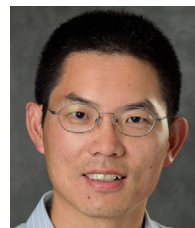
His research interests include memristive memories and systems, design, fabrication, and implementation of microelectromechanical (MEMS) actuators, smart materials-based microtransducers, control systems of hysteretic and nonlinear systems, and tunable microresonators.



Nelson Sepúlveda received the B.S. degree in Electrical and Computer Engineering from the University of Puerto Rico, Mayagüez, Puerto Rico, in 2001, and the M.S. and Ph.D. degrees in Electrical and Computer Engineering from Michigan State University (MSU), East Lansing, MI, USA, in 2002 and 2005, respectively. During the last year of graduate school, he attended the Sandia National Laboratories as part of a fellowship from the Microsystems and Engineering Sciences Applications program.

In January 2006, he joined the faculty of the Department of Electrical and Computer Engineering, University of Puerto Rico. He has been a Visiting Faculty Researcher at the Air Force Research Laboratories (2006, 2007, 2013, and 2014), National Nanotechnology Infrastructure Network (2008), and the Cornell Center for Materials Research (2009), the last two being the National Science Foundation (NSF) funded centers at Cornell University, Ithaca, NY, USA. In Fall 2011, he joined the faculty of the Department of Electrical and Computer Engineering, MSU, where he is currently an Assistant Professor. His current research interests include smart materials and the integration of such in microelectromechanical systems, with particular emphasis on vanadium dioxide (VO₂) thin films and the use of the structural phase transition for the development of smart microactuators.

Dr. Sepúlveda received the NSF CAREER Award in 2010, and the MSU Teacher-Scholar Award in 2015.



Xiaobo Tan received the B.Eng. and M.Eng. degrees in automatic control from Tsinghua University, Beijing, China, in 1995 and 1998, respectively, and the Ph.D. degree in Electrical and Computer Engineering from the University of Maryland, College Park, in 2002.

Dr. Tan is currently an Associate Professor in the Department of Electrical and Computer Engineering at Michigan State University (MSU). His current research interests include modeling and control of systems with hysteresis, electroactive polymer sensors and actuators, and bio-inspired underwater robots and their application to environmental sensing.

Dr. Tan serves as an Associate Editor/Technical Editor for Automatica, IEEE/ASME Transactions on Mechatronics, and International Journal of Advanced Robotic Systems. He served as the Program Chair of the 2011 International Conference on Advanced Robotics, and is currently serving as the Finance Chair of the 2015 American Control Conference. He has coauthored one book (Biomimetic Robotic Artificial Muscles) and over 60 peer-reviewed journal papers, and holds one US patent. He was a recipient of NSF CAREER Award (2006), MSU Teacher-Scholar Award (2010), and several Best Paper Awards.