

Efficient structure-preserving model reduction for nonlinear mechanical systems with application to structural dynamics

Kevin Carlberg,^{*} Ray Tuminaro,[†] and Paul Boggs[‡]
Sandia National Laboratories, Livermore, CA 94551, USA[§]

This work proposes a model-reduction methodology that both preserves Lagrangian structure and leads to computationally inexpensive models, even in the presence of high-order nonlinearities. We focus on parameterized simple mechanical systems under Rayleigh damping and external forces, as structural-dynamics models often fit this description. The proposed model-reduction methodology directly approximates the quantities that define the problem’s Lagrangian structure: the Riemannian metric, the potential-energy function, the dissipation function, and the external force. These approximations preserve salient properties (e.g., positive definiteness), behave similarly to the functions they approximate, and ensure computational efficiency. Results applied to a simple parameterized truss-structure problem demonstrate the importance of preserving Lagrangian structure and illustrate the method’s ability to generate speedups while maintaining observed stability, in contrast with other model-reduction techniques that do not preserve structure.

I. Introduction

Computational structural dynamics (CSD) tools have become indispensable in many industries due to their ability to enhance the understanding of complex structural systems, reduce design costs, and improve reliability. However, the large computational cost of high-fidelity structural-dynamics simulations can result in simulation times on the order of weeks, even when high-performance computing resources are available. As a result, high-fidelity CSD tools can be impractical for time-critical applications that demand the accuracy provided by high-fidelity models. In particular, applications such as nondestructive evaluation for structural health monitoring, embedded control, design optimization, and uncertainty quantification require highly accurate results to be obtained in minutes or hours.

Model reduction methods present a promising approach for realizing this goal. These methods approximate the high-fidelity model by reducing the number of equations and unknowns describing it. To do so, they employ a projection process: they compute fast ‘online’ solutions by searching in a low-dimensional space that was computed *a priori* by expensive ‘offline’ computations. Thus, the reduced-order model used for online computations is characterized by a low-dimensional dynamical system that is formed by a projection process on the equations characterizing the high-fidelity model.

However, generating a reduced-order model that preserves the Lagrangian structure intrinsic to structural-dynamics models is not a trivial task. Such structure is critical to preserve, as it leads to fundamental properties such as energy conservation (in the absence of non-conservative forces), conservation of quantities associated with symmetries in the system, and symplectic time-evolution maps. In fact, the class of structure-preserving integrators (e.g., geometric integrators,¹ variational integrators²) has been developed to ensure that the solution to the full-order computational model associates with the time-evolution map of a (modified) Lagrangian system.

^{*}Harry S. Truman Fellow, Quantitative Modeling & Analysis Department ktcarlb@sandia.gov, AIAA member

[†]Principal Member of the Technical Staff, Numerical Analysis and Applications Department, rstumin@sandia.gov

[‡]Principal Member of the Technical Staff, Quantitative Modeling & Analysis Department, ptboggs@sandia.gov

[§]Sandia National Laboratories, P.O. Box 969, MS 9159, Livermore, CA 94551, USA. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94-AL85000.

Lall et al.³ showed that performing a Galerkin projection on the Euler–Lagrange equation (as opposed to the first-order state-space form) leads to a reduced-order model that preserves Lagrangian structure. However, the computational cost of assembling the associated low-dimensional equations scales with the dimension of the high-fidelity model. For this reason, this approach is efficient primarily for problems where the low-dimensional operators can be assembled *a priori*; this occurs only in the (very limited) case where the operators 1) exhibit a linear or quadratic dependence on the state, and 2) can be decomposed a sum of products of parameter-independent operators and parameter-dependent functions.

Several methods have been developed in the context of nonlinear-ODE model reduction that can reduce the computational cost of assembling the low-dimensional equations. However, these methods destroy the problem’s Lagrangian structure. Collocation approaches^{4,5} perform a Galerkin projection on only a small subset of the full-order equations, thereby ‘throwing away’ most full-order equations. Although this method works well for some nonlinear ODEs, it destroys Lagrangian structure as will be shown in Section III.B.1. Empirical interpolation^{6–9} and least-squares-reconstruction methods^{10–12} compute a few entries of the nonlinear functions, then approximate the uncomputed entries by interpolation or least-squares regression with an empirically derived basis. Galerkin projection is then performed with the approximated nonlinear function. Even though this approach has led to promising results for nonlinear ODEs without special structure, this technique also destroys Lagrangian structure as explained in Section III.B.2.

The goal of this work is to devise a reduced-order model that is both efficient and preserves Lagrangian structure. We focus particularly on structural-dynamics models described as simple mechanical systems under Rayleigh damping and external forces. The methodology we propose constructs a reduced-order model by directly approximating the quantities defining the problem’s Lagrangian structure:

- *Configuration space.* The configuration space of reduced dimension is derived in the usual fashion, e.g., proper orthogonal decomposition, modal decomposition.
- *Riemannian metric.* The Riemannian metric is defined by a low-dimensional symmetric-positive-definite matrix. We approximate this low-dimensional matrix with a sampling-based method that preserves positive definiteness (cf. Section IV.B).
- *Potential-energy function.* The potential energy function is approximated by employing the original potential-energy function, but with the reduced basis replaced with a sparse matrix with many zero rows. This sparse matrix is computed offline by minimizing the discrepancy in the potential-energy gradient over training data.
- *Dissipation function.* Similar to the Riemannian metric, the dissipation function is defined by a low-dimensional symmetric-positive-semidefinite matrix. We approximate this low-dimensional matrix with the aforementioned sampling-based method; however, in this case we ensure that positive-semidefiniteness is preserved (cf. Section IV.B).
- *External force.* The external force is derived from the Lagrange–D’Alembert principle applied with variations in the configuration space. We approximate this by applying empirical interpolation/least-squares reconstruction to the external force expressed in the original coordinates. As a result, the external force appearing in the equations of motion for the reduced-order model can be derived by the Lagrange–D’Alembert principle applied to the (modified) external force with variations restricted to the reduced-order configuration space.

The remainder of the paper is organized as follows. Section II introduces the Lagrangian-mechanics formulation. Section III outlines existing model-reduction techniques and highlights the need for an efficient, structure-preserving method. Section IV describes the proposed method. Section V presents numerical experiments applied to a conservative truss-structure system. Finally, Section VI concludes the paper.

II. Problem formulation

This work considers nonlinear mechanical systems, with a particular focus on parameterized structural-dynamics models derived by a finite-element formulation. Such structural-dynamics models represent parameterized *simple mechanical systems*, which are defined by a (parameterized) triple (Q, g, V) :

- a differentiable configuration manifold Q ,
- a parameterized Riemannian metric $g : TQ \times TQ \times \mathcal{D} \rightarrow \mathbb{R}_+$, where TQ denotes the tangent bundle of Q and $\mathcal{D} \subset \mathbb{R}^p$ denotes the parameter domain,

- a parameterized potential-energy function $V : Q \times \mathcal{D} \rightarrow \mathbb{R}$.

The structural-dynamics models considered here are characterized by a Euclidean configuration space $Q = \mathbb{R}^N$ (the degrees of freedom in a finite-element model), a Riemannian metric defined as $g : (v, w; \mu) \mapsto v^T M(\mu)w$, and a potential-energy function that describes the strain energy in the model. Here, $M(\mu) \in \text{SPD}(N)$ denotes the mass matrix and $\text{SPD}(N)$ denotes the set of $N \times N$ symmetric positive-definite matrices.

The kinetic energy of simple mechanical systems is $T(\dot{q}; \mu) = \frac{1}{2}g(\dot{q}, \dot{q}; \mu) = \frac{1}{2}\dot{q}^T M(\mu)\dot{q}$. The Lagrangian can then be expressed as

$$L(q, \dot{q}; \mu) = T(\dot{q}; \mu) - V(q; \mu) \quad (1)$$

$$= \frac{1}{2}\dot{q}^T M(\mu)\dot{q} - V(q; \mu). \quad (2)$$

Given the Lagrangian (2), the equations of motion for a simple mechanical system subject to *non-conservative*^a forces $F_n(t, q, \dot{q}; \mu)$ can be derived from the forced Euler–Lagrange equation

$$\frac{d}{dt}\nabla_{\dot{q}}L(q, \dot{q}; \mu) - \nabla_q L(q, \dot{q}; \mu) = F_n(t, q, \dot{q}; \mu). \quad (3)$$

In structural dynamics, the non-conservative forces often consist of a configuration-independent external force $(t, \mu) \mapsto f^{\text{ext}}(t; \mu)$ with $f^{\text{ext}} : [0, T] \times \mathcal{D} \rightarrow \mathbb{R}^N$ and T denoting the final time, and a dissipative force corresponding to Rayleigh viscous damping. This dissipative force derives from a positive-semidefinite dissipation function^b

$$\mathcal{F}(\dot{q}; \mu) \equiv \frac{1}{2}\dot{q}^T C(\mu)\dot{q}, \quad (4)$$

where $C(\mu) \in \text{SP}(N)$ and $\text{SP}(N)$ denotes the set of $N \times N$ symmetric positive-semidefinite matrices. So, we consider non-conservative forces of the form

$$F_n(t, q, \dot{q}; \mu) = f^{\text{ext}}(t; \mu) - \nabla_{\dot{q}}\mathcal{F}(\dot{q}; \mu). \quad (5)$$

Substituting (2), (4), and (5) into the Euler–Lagrange equation (3) leads to the familiar equations of motion for structural dynamics

$$M(\mu)\ddot{q} + C(\mu)\dot{q} + \nabla_q V(q; \mu) = f^{\text{ext}}(t; \mu). \quad (6)$$

Conservative mechanical systems ($F_n(t, q, \dot{q}; \mu) = 0$) exhibit important properties. For example, these systems conserve energy and quantities associated with symmetry, and their time-evolution maps are symplectic. Because these properties are important characteristics of the mechanical systems, it is desirable for the numerical-integration schemes to preserve these properties when applied to such systems. As previously mentioned, the class of structure-preserving integrators has been developed for this purpose. This class of integrators is derived to ensure intrinsic properties such as energy conservation, momentum conservation, and symplecticity are preserved by in the numerical solution.^{1,2}

For this reason, we aim to develop a reduced-order model that preserves the structure of the mechanical system, yet is computationally inexpensive to simulate. This will ensure that the reduced-order model preserves these characteristic properties. Further, the equations of motion for these models can be solved with a structure-preserving integrator; this will ensure that the numerical solution computed with the reduced-order model will also preserve these properties. In summary, the properties of the model we seek to preserve are:

1. a configuration space,
2. a parameterized Riemannian metric defined on the tangent bundle to the configuration space,
3. a parameterized potential-energy function,
4. a parameterized positive-semidefinite dissipation function, and
5. an external force derived from the Lagrange–D’Alembert principle applied with variations in the configuration space.

Properties 1–3 constitute the parameterized triple that ensures the model describes a simple mechanical system; Properties 4–5 characterize the non-conservative forces.

^aConservative forces can be handled by directly including them in the Lagrangian.

^bNon-viscously damped systems can also often be derived by a positive-semidefinite dissipation function.¹³

III. Existing model-reduction techniques

Model-reduction techniques seek to generate a low-dimensional model that is inexpensive to evaluate, yet captures the essential features of the high-fidelity model. To do so, these methods first perform analyses of the system at a set of training parameters $\mathcal{D}_{\text{sample}} \subset \mathcal{D}$ during a computationally intensive ‘offline’ (i.e., training) stage. These analyses may include the integration of the equations of motion, modal analyses, etc. Then, the data generated during these analyses are employed to define a configuration manifold of reduced dimension $\mathbf{Q}_r \subset \mathbf{Q}$, with $\dim \mathbf{Q}_r = n \ll N$. Once this low-dimensional configuration manifold is defined, it is employed to generate a low-dimensional model that can be used to perform computationally inexpensive analyses for any $\mu^* \in \mathcal{D}$ during the ‘online’ (i.e., deployed) stage.

When the configuration space is Euclidean (as is the case for the models considered herein), the low-dimensional configuration space can be expressed as $\mathbf{Q}_r = q_0(\mu) + \mathcal{Y}$, where $\mathcal{Y} \subset \mathbb{R}^N$, $\dim \mathcal{Y} = n$. The trial subspace is spanned by a basis such that $\mathcal{Y} = \text{range}(\Phi)$ with $\Phi \in \mathbb{R}_*^{N \times n}$. Here, $\mathbb{R}_*^{n \times m}$ denotes the noncompact Stiefel manifold: the set of full-rank $n \times m$ matrices. This leads to the following expression for the generalized coordinates and their derivatives:

$$q = q_0(\mu) + \Phi q_r \quad (7)$$

$$\dot{q} = \Phi \dot{q}_r \quad (8)$$

$$\ddot{q} = \Phi \ddot{q}_r, \quad (9)$$

where $q_r \in \mathbb{R}^n$. Thus, the low-dimensional configuration space can be described in terms of low-dimensional generalized coordinates $q_r \in \mathbf{Q}_r \equiv \mathbb{R}^n$ or in terms of original coordinates by the definition

$$\mathbf{Q}_r \equiv \{q_0(\mu) + \Phi q_r \mid q_r \in \mathbf{Q}_r\}. \quad (10)$$

The basis (in matrix form) Φ can be determined by a variety of techniques, including proper orthogonal decomposition and modal decomposition.

III.A. Galerkin projection

Model-reduction methods based on Galerkin projection preserve the problem’s structure. As was pointed out by Lall et al.,³ the Galerkin projection must be carried out on the Euler–Lagrange equation (3)—not the first-order state-space form $\dot{x}(t) = f(x(t))$ —in order to preserve Lagrangian structure.

Following the approach of Lall et al.,³ Galerkin-projection-based methods substitute Eqs. (7)–(9) directly into Lagrangian (and dissipation function) and derive the equations of motion in using a set of generalized coordinates that has lower dimension. In this way, the resulting model has an identical structure to the original problem.

For structural dynamics, this amounts to defining the Lagrangian as

$$L_r(q_r, \dot{q}_r; \mu) \equiv L(q_0(\mu) + \Phi q_r, \Phi \dot{q}_r; \mu) \quad (11)$$

$$= \frac{1}{2} \dot{q}_r^T \Phi^T M(\mu) \Phi \dot{q}_r - V(q_0(\mu) + \Phi q_r; \mu) \quad (12)$$

and the dissipation function as

$$\mathcal{F}_r(\dot{q}_r; \mu) \equiv \mathcal{F}(\Phi \dot{q}_r; \mu) \quad (13)$$

$$= \frac{1}{2} \dot{q}_r^T \Phi^T C(\mu) \Phi \dot{q}_r. \quad (14)$$

The external force, which is derived based on the Lagrange–D’Alembert variational principle, is transformed by relation (7) into

$$f_r^{\text{ext}} = \Phi^T f^{\text{ext}}. \quad (15)$$

Following Section II, the forced Euler–Lagrange equation applied to the Lagrangian L_r , the dissipation function \mathcal{F}_r , and the external force f_r^{ext} leads to the reduced-order equations of motion

$$\frac{d}{dt} \nabla_{\dot{q}_r} L_r(q_r, \dot{q}_r; \mu) - \nabla_{q_r} L_r(q_r, \dot{q}_r; \mu) + \nabla_{\dot{q}_r} \mathcal{F}_r(\dot{q}_r; \mu) = f_r^{\text{ext}}. \quad (16)$$

This can be rewritten as

$$\Phi^T M(\mu) \Phi \ddot{q}_r + \Phi^T C(\mu) \Phi \dot{q}_r + \Phi^T \nabla_q V(q_0(\mu) + \Phi q_r; \mu) = \Phi^T f^{\text{ext}}(t; \mu). \quad (17)$$

Note that Eq. (17) is equivalent to applying Galerkin projection to the original forced Euler–Lagrange equation (6).

Thus, the reduced-order model preserves the problem structure because it preserves all five properties described in Section II:

1. a configuration space $Q_r = \mathbb{R}^n$, which relates to the original configuration space by Eq. (10),
2. a parameterized Riemannian metric $g_r : (v_r, w_r; \mu) \mapsto v_r^T \Phi^T M(\mu) \Phi w_r$,
3. a parameterized potential-energy function $V_r : (q_r; \mu) \mapsto V(q_0(\mu) + \Phi q_r; \mu)$,
4. a parameterized positive-semidefinite dissipation function $\mathcal{F}_r : (\dot{q}_r; \mu) \mapsto \frac{1}{2} \dot{q}_r^T \Phi^T C(\mu) \Phi \dot{q}_r$, and
5. an external force f_r^{ext} derived from applying the Lagrange–D’Alembert principle to the original external force f^{ext} , but restricted to variations in the configuration space Q_r .

III.A.1. Computational bottleneck

Although the equations of motion (17) are of small dimension $n \ll N$, it is computationally expensive to solve them with a numerical-integration method. The reason is simple: computing the low-dimensional components of (17) incur large-scale operations.

For example, consider computing $\Phi^T M(\mu) \Phi$ for some $\mu = \mu^*$ during the online stage. If the mass matrix can be expressed as an affine function of the parameters $M(\mu) = \sum_i \alpha_i(\mu) M_i$ with $\alpha_i : \mathcal{D} \rightarrow \mathbb{R}$ and $M_i \in \mathbb{R}^{N \times N}$, then the products $\Phi^T M_i \Phi$ can be assembled during the offline stage, and $\Phi^T M(\mu) \Phi = \sum_i \alpha_i(\mu) [\Phi^T M_i \Phi]$ can be computed in $\mathcal{O}(n^2)$ floating-point operations (flops) during the online stage.^{14,15} However, this scenario is quite limiting and is not generally applicable.

In the general setting, the steps required to compute $\Phi^T M(\mu) \Phi$ for each $\mu^* \in \mathcal{D}$ are:

- i compute $M(\mu^*)$, which incurs $\mathcal{O}(N\omega)$ floating-point operations (flops), where ω denotes the average number of nonzeros per row of the sparse matrix $M(\mu^*)$. Note that $\omega \ll N$ for finite-element models constructed using basis functions with compact support,
- ii compute the product $M(\mu^*) \Phi$, which incurs $\mathcal{O}(N\omega n)$ flops, and
- iii compute the product $\Phi^T (M(\mu^*) \Phi)$, which incurs $\mathcal{O}(Nn^2)$ flops.

Thus, the cost scales with N : the large dimension of the original configuration manifold. The same analysis holds for computing the product $\Phi^T C(\mu^*) \Phi$.

If the potential energy V exhibits a nonlinear dependence on coordinates q_r of polynomial degree greater than two, the situation worsens. In this case, the product $\Phi^T V(q_0(\mu) + \Phi q_r; \mu^*)$ must be recomputed whenever q_r changes; this must be done for each Newton step (within each time step) if an implicit numerical integrator is employed to solve (17). Similarly, $f^{\text{ext}}(t; \mu^*)$ and subsequently $\Phi^T f^{\text{ext}}(t; \mu^*)$ must be computed at each time step.

Thus, for systems with nonaffine parameter dependence or high-order nonlinearities, a dimension reduction is insufficient to generate models with computational complexity independent of N . In fact, such models are often *more expensive* to simulate than the original model for this reason.

III.B. Galerkin projection with function sampling

Several techniques have been developed to mitigate the computational bottleneck described in Section III.A.1. These ‘function sampling’ methods select only a few entries of the vector-valued functions exhibiting non-affine parameter dependence or nonlinear dependence on coordinates q ; other entries are ignored or are effectively set to zero. Such methods have been successfully applied to ODEs without special structure. However, when applied to mechanical systems described by Lagrangian mechanics, these techniques destroy the problem’s Lagrangian structure.

III.B.1. Collocation

Collocation approaches^{4,5} compute only a subset of the full-order equations (6) before performing the Galerkin projection. That is, the reduced-order equations of motion (17) are approximated by

$$\Phi^T Z^T Z M(\mu) \Phi \ddot{q}_r + \Phi^T Z^T Z C(\mu) \Phi \dot{q}_r + \Phi^T Z^T Z \nabla_q V(q_0(\mu) + \Phi q_r; \mu) = \Phi^T Z^T Z f^{\text{ext}}(t; \mu). \quad (18)$$

Here, the ‘sampling matrix’ $Z \in \mathbb{R}^{n_Z \times N}$ consists of $n_Z \ll N$ selected rows of the identity matrix, where $N \geq n_Z \geq n$.

Computing each component of (18) is computationally inexpensive when the matrices $M(\mu)$, $C(\mu)$, and $\nabla_{qq} V(q_0(\mu) + \Phi q_r; \mu)$ are sparse (i.e., $\omega \ll N$). For example, the steps required to compute the first term in (18) are:

- i compute $ZM(\mu^*)$, which incurs $\mathcal{O}(n_Z \omega)$ floating-point operations (flops),
- ii compute the product $(ZM(\mu^*)) \Phi$, which incurs $\mathcal{O}(n_Z \omega n)$ flops, and
- iii compute the product $\Phi^T Z^T (ZM(\mu^*) \Phi)$, which incurs $\mathcal{O}(n_Z n^2)$ flops.

Thus, the cost scales with $n_Z \ll N$: the small number of rows in the sampling matrix Z . The same operation-count analysis holds for computing the other terms. However, this cost-reduction approach destroys the problem’s structure, as it does not preserve the following properties described in Section II:

2. The reduced mass matrix $\Phi^T Z^T Z M(\mu) \Phi$ is not symmetric, so it does not define a metric.
3. The term $\Phi^T Z^T Z \nabla_{qq} V(q_0(\mu) + \Phi q_r; \mu) \Phi$ is not symmetric, so it does not derive from a potential-energy function.
4. The reduced damping matrix $\Phi^T Z^T Z C(\mu) \Phi$ is not symmetric, so it does not derive from a dissipation function.

Note that Property 1 is trivially satisfied, as the configuration space can be described as $Q_r = \mathbb{R}^n$ and relates to the original configuration space by Eq. (10). Further, Property 5 is satisfied, because the non-conservative forces can be derived by applying the Lagrange–D’Alembert variational principle to a (modified) external force $Z^T Z f^{\text{ext}}(t; \mu)$, but restricted to variations in the (true) configuration space Q_r .

III.B.2. Empirical interpolation/least-squares reconstruction

Methods based on empirical interpolation^{6–9} or least-squares reconstruction^{10,11} approximate nonlinear functions $f(q; \mu)$ that appear as $\Phi^T f(q; \mu)$ in the reduced-order equations of motion. In the current case, four nonlinear functions of the form $f(q; \mu)$ are approximated: $M(\mu) \Phi \ddot{q}_r$, $C(\mu) \Phi \dot{q}_r$, $\nabla_q V(q_0(\mu) + \Phi q_r; \mu)$, and $f^{\text{ext}}(t; \mu)$.

During the offline stage, these methods construct a basis $\Phi_f \in \mathbb{R}^{N \times n_f}$ with $n_f \leq n_Z$ to approximate the function f . The basis Φ_f can be computed empirically via proper orthogonal decomposition (POD). This consists of two steps: 1) collect a set of ‘snapshots’ $\mathcal{X}_f = \{f(q; \mu) \mid \mu \in \mathcal{D}_{\text{sample}}, t \in \mathbb{T}_{\text{sample}}(\mu)\}$, where $\mathbb{T}_{\text{sample}}(\mu) \subset [0, T]$ are the time steps taken by the numerical-integration method for the training simulation with parameters μ , and 2) compute Φ_f by Algorithm 1 of Appendix VI.A using \mathcal{X}_f as the input. During the online stage, these methods compute an approximation $\tilde{f}(q; \mu)$ by computing \hat{f} as the solution to the linear least-squares problem (19)

$$\underset{x \in \mathbb{R}^{n_f}}{\text{minimize}} \|Zf(q; \mu) - Z\Phi_f x\|_2 \quad (19)$$

and then computing $\tilde{f}(q; \mu)$ via

$$\tilde{f}(q; \mu) = \Phi_f \hat{f} = \Phi_f [Z\Phi_f]^+ Zf(q; \mu). \quad (20)$$

Here, a superscript $+$ denotes the Moore–Penrose pseudoinverse. Notice that when $n_f = n_Z$, the least-squares residual is zero (assuming the $Z\Phi_f$ has full column rank) and so the above procedure corresponds to interpolation in this case.

This approximation technique leads to computational-cost savings during the online stage, as the computational cost of evaluating the terms $\Phi^T \tilde{f}_i$, $i = 1, \dots, 4$ that compose the equations of motion scales with $n_Z \ll N$. Unfortunately, this approximation method also destroys the problem’s structure, as it violates the following properties:

2. The reduced mass matrix $\Phi^T \Phi_{f_1} [Z\Phi_{f_1}]^+ ZM(\mu)\Phi$ is not symmetric, so it does not define a metric.
4. The reduced damping matrix $\Phi^T \Phi_{f_2} [Z\Phi_{f_2}]^+ ZC(\mu)\Phi$ is not symmetric, so it does not derive from a dissipation function.
3. The term $\Phi^T \Phi_{f_3} [Z\Phi_{f_3}]^+ Z\nabla_{qq}V(q_0(\mu) + \Phi q_r; \mu)\Phi$ is not symmetric, so it does not derive from a potential-energy function.

Again, Property 1 is satisfied. Property 5 is also satisfied, because the non-conservative external force can be derived by the Lagrange–D’Alembert principle applied to the (modified) external force $\tilde{f}^{\text{ext}}(t; \mu) = \Phi_{f_4} [Z\Phi_{f_4}]^+ Zf^{\text{ext}}(t; \mu)$ with variations restricted to the (true) configuration space \mathcal{Q}_r .

IV. Efficient, structure-preserving model reduction

The main idea of the proposed approach is to directly approximate the quantities defining the structure of the Galerkin-projection reduced-order model. For the structural-dynamics models considered herein, these quantities are enumerated at the end of Section III.A: the Riemannian metric g_r , the potential-energy function V_r , the semidefinite dissipation function \mathcal{F}_r , and the external force f_r^{ext} . These approximations should 1) preserve salient properties, 2) behave similarly to the functions they approximate, and 3) lead to computationally inexpensive reduced-order-model simulations. Note that we must approximate only quantities exhibiting non-affine parameter dependence or scalar quantities with nonlinearities in the q of polynomial degree exceeding two.

To this end, we propose a model defined by

1. the configuration space $\mathcal{Q}_r = \mathbb{R}^n$, which relates to the original configuration space by Eq. (10),
2. an approximated Riemannian metric $\tilde{g}_r \approx g_r$,
3. an approximated potential-energy function $\tilde{V}_r \approx V_r$,
4. an approximated positive-semidefinite dissipation function $\tilde{\mathcal{F}}_r \approx \mathcal{F}_r$, and
5. an external force derived from applying the Lagrange–D’Alembert principle to an approximated force $\tilde{f}_r^{\text{ext}} \approx f_r^{\text{ext}}$, but restricted to variations in the configuration space \mathcal{Q}_r . Thus, the form of the approximated external force is

$$\tilde{f}_r^{\text{ext}}(t; \mu) \equiv \Phi^T \tilde{f}^{\text{ext}}(t; \mu). \quad (21)$$

After these approximations are defined, the equations of motion can be derived from the forced Euler–Lagrange equation using these approximated quantities.

The next sections describe the proposed approximations in detail. We note that the proposed approximation methods are not the only possibilities; other techniques that preserve salient properties may be derived.

IV.A. External-force approximation \tilde{f}_r^{ext}

Because the external-force approximation is of the form described by Eq. (21), the task of generating this approximation can be reduced to approximating the function $f^{\text{ext}}(t; \mu)$. One way to accomplish this is the empirical interpolation/least-squares reconstruction approach described in Section III.B.2. Applying this technique to the external force yields

$$\tilde{f}^{\text{ext}} = \Phi_f [Z\Phi_f]^+ Zf^{\text{ext}}(t; \mu). \quad (22)$$

Here, $\Phi_f \in \mathbb{R}^{N \times n_f}$ is a basis for the external force. Substituting (22) into (21) yields the expression for the approximated external force:

$$\tilde{f}_r^{\text{ext}} = \Phi^T \Phi_f [Z\Phi_f]^+ Zf^{\text{ext}}(t; \mu). \quad (23)$$

IV.A.1. Offline/online decomposition

The offline stage requires the following steps to implement this approximation:

1. Collect snapshots of the external force: $\mathcal{X}_f \equiv \{f^{\text{ext}}(t; \mu) \mid \mu \in \mathcal{D}_{\text{sample}}, t \in \mathbb{T}_{\text{sample}}(\mu)\}$
2. Compute POD basis Φ_f using snapshots \mathcal{X}_f .

3. Determine the sampling matrix Z .
4. Compute the low-dimensional matrix $\Phi^T \Phi_f [Z \Phi_f]^+$.

The online stage consists of the following:

1. Compute $n_Z \ll N$ entries of the external force $Z f^{\text{ext}}(t; \mu)$.
2. Compute the low-dimensional matrix–vector product $[\Phi^T \Phi_f [Z \Phi_f]^+] [Z f^{\text{ext}}(t; \mu)]$.

IV.B. Riemannian-metric and dissipation-function approximations \tilde{g}_r and $\tilde{\mathcal{F}}_r$

Each of the terms g_r and \mathcal{F}_r is quadratic, and is therefore defined by a low-dimensional matrix $\Phi^T A(\mu) \Phi \in \mathbb{R}^{n \times n}$, where $A = M$ (positive definite) or $A = C$ (positive semidefinite). To this end, we propose sampling-based approximations that directly approximate these low-dimensional matrices in a manner that preserves symmetry and positive (semi)definiteness:

$$\tilde{g}_r(v_1, v_2; \mu) = v_1^T h_{\text{SPD}}(M(\mu), \underline{M}) v_2 \quad (24)$$

$$\tilde{\mathcal{F}}_r(\dot{q}_r; \mu) = \frac{1}{2} \dot{q}_r^T h_{\text{SP}}(C(\mu), \underline{C}) \dot{q}_r. \quad (25)$$

Here, $h_{\text{SPD}} : \text{SPD}(N) \times (\text{S}(N))^{n_A} \rightarrow \text{SPD}(n)$ and $h_{\text{SP}} : \text{SP}(N) \times (\text{S}(N))^{n_A} \rightarrow \text{SP}(n)$, where $\text{S}(p)$ denotes the set of $p \times p$ symmetric matrices. These functions approximate the low-dimensional matrix $\Phi^T A(\mu) \Phi$ by the same functional form:

$$h_{\text{SPD}}(A, \underline{A}) = \sum_{i=1}^{n_A} \Phi^T \underline{A}^i \Phi \xi_{\text{SPD}}^i(A, \underline{A}) \quad (26)$$

$$h_{\text{SP}}(A, \underline{A}) = \sum_{i=1}^{n_A} \Phi^T \underline{A}^i \Phi \xi_{\text{SP}}^i(A, \underline{A}). \quad (27)$$

Here, $\underline{A} \equiv \{\underline{A}^i\}_{i=1}^{n_A}$ with $\underline{A}^i \in \text{S}(N)$, $i = 1, \dots, n_A$ represents a basis for the matrix A .

The next sections describe how the functions h_{SPD} and h_{SP} are defined, i.e., the method for computing the coefficients $\xi_{\text{SPD}} \equiv (\xi_1, \dots, \xi_{n_A})$, $\xi_{\text{SP}} \equiv (\xi_1, \dots, \xi_{n_A})$, and matrix basis \underline{A} .

IV.B.1. Matrix basis

To obtain the matrix basis \underline{A} , we propose applying a vectorized POD method. This approach allows \underline{A} to be considered a set of ‘principal matrices’ that optimally represent $A(\mu) \in \text{S}(N)$ over a set of training parameters $\mu \in \mathcal{D}_{\text{sample}}$.^c The steps for this method are:

1. Collect ‘snapshots’ of the matrices: $\mathcal{X}_A \equiv \{A(\mu) \mid \mu \in \mathcal{D}_{\text{sample}}\}$.
2. Vectorize the snapshots:

$$\mathcal{X}_{v(A)} \equiv \{v(A) \mid A \in \mathcal{X}_A\}, \quad (28)$$

where the function $v : \mathbb{R}^{N \times N} \rightarrow \mathbb{R}^{N^2}$ vectorizes a matrix.

3. Compute a n_A -dimensional POD basis of the vectorized snapshots $\Phi(n_A, \mathcal{X}_{v(A)})$, where $n_A \leq \text{card}(\mathcal{X}_A)$ (cf. Algorithm 1 in Appendix VI.A).
4. Transform these POD basis vectors into their matrix form:

$$\underline{A} = \{v^{-1}(\phi_A^i)\}_{i=1}^{n_A}, \quad (29)$$

where $\Phi(n_A, \mathcal{X}_{v(A)}) \equiv [\phi_A^1 \ \dots \ \phi_A^{n_A}]$.

Note that every element of \underline{A} is guaranteed to be symmetric, as it is a linear combination of symmetric matrices $A(\mu)$, $\mu \in \mathcal{D}_{\text{sample}}$.

^cThese matrices are optimal in the sense that they minimize the average projection error (as measured in the Frobenius norm) of the matrix snapshots.

IV.B.2. Coefficients

The scalar coefficients ξ_{SPD} (resp. ξ_{SP}) are determined by seeking to match the matrix A and the linear combination $\sum_{i=1}^{n_A} \underline{A}^i \xi_{\text{SPD}}^i$ (resp. $\sum_{i=1}^{n_A} \underline{A}^i \xi_{\text{SP}}^i$) at a set of ‘sample elements’ $\mathcal{E} \equiv \{(i_m, j_m)\}_{m=1}^{n_e}$ with $n_e \ll N$. That is, ξ_{SPD} is the solution to (30) with ϵ a numerical threshold for defining a full-rank matrix; ξ_{SP} is the solution to (30) with $\epsilon = 0$:

$$\begin{aligned} & \underset{x \in \mathbb{R}^{n_A}}{\text{minimize}} && \sum_{m=1}^{n_e} \left(A_{i_m, j_m} - \sum_{k=1}^{n_A} \underline{A}_{i_m, j_m}^k x_k \right)^2 \\ & \text{subject to} && \sum_{k=1}^{n_A} \Phi^T \underline{A}^k \Phi x_k \geq \epsilon. \end{aligned} \quad (30)$$

This approach amounts to the gappy POD method¹⁰ applied to matrix data with the addition of a linear-matrix-inequality constraint. The constraint ensures that the image of $h_{\text{SPD}}(A, \underline{A})$ is $\text{SPD}(n)$ (resp. image of h_{SP} is $\text{SP}(n)$).

Problem (30) is characterized a quadratic objective and nonlinear constraints (the linear-matrix-inequality constraint is effectively nonlinear in x). Appendix VI.B describes a method for solving it. Note that this optimization problem must be solved *online* using the online-sampled data A_{i_m, j_m} , $(i_m, j_m) \in \mathcal{E}$.

IV.B.3. Offline/online decomposition

The offline stage consists of the following operations. These steps should be carried out both for $A = M$ and $A = C$.

1. Compute the basis \underline{A} using the vectorized POD approach described in Section IV.B.1.
2. Determine sample elements \mathcal{E} .
3. Compute low-dimensional matrices $\Phi^T \underline{A}^i \Phi$, $i = 1, \dots, n_A$.

During the online stage, the following steps are required for $A = M$ and $A = C$:

1. Compute A_{i_m, j_m} with $(i_m, j_m) \in \mathcal{E}$.
2. Solve the small-scale optimization problem (30) for ξ_{SPD} or ξ_{SP} .
3. Assemble the low-dimensional matrix $h_{\text{SPD}}(A, \underline{A})$ or $h_{\text{SP}}(A, \underline{A})$ by Eq. (26) or (27).

IV.C. Potential-energy-function approximation \tilde{V}_r

When the potential-energy function V_r is quadratic in coordinates q_r , the approximation technique described in the previous section can be used. However, when it exhibits a higher-order nonlinearity, another method is required. When the potential energy is defined by the integral over a domain (i.e., $V(q; \mu) = \int_{\Omega} \mathcal{V}(X, q; \mu) d\Omega_X$), a sparse cubature method¹⁶ can be used to achieve computational efficiency. In the absence of such structure, a different method is needed.

In order to approximate such a potential-energy function in a way that 1) employs the original potential-energy function V , and 2) leads to computationally inexpensive terms in the equations of motion, we propose ‘injecting sparseness’ into the trial basis. To accomplish this, we replace Φ with a *sparse matrix* $\Psi \in \mathbb{R}^{N \times n}$ with only $n_Z \ll N$ rows containing nonzero entries:

$$\tilde{V}_r(q_r; \mu) = V(q_0(\mu) + \Psi q_r; \mu). \quad (31)$$

Note that this sparse matrix may be defined as $\Psi \equiv Z^T \underline{\Psi}$, with $Z \in \mathbb{R}^{n_Z \times N}$ the sampling matrix.

To compute the sparse basis Ψ , we first note that only the gradient $\nabla_{q_r} V_r(q_r; \mu) \equiv \Phi^T \nabla_q V(q_0(\mu) + \Phi q_r; \mu)$ appears in the equations of motion (17). Thus, we aim to compute Ψ such that $\nabla_{q_r} \tilde{V}_r \approx \nabla_{q_r} V_r$. To this end, we propose computing $\underline{\Psi}$ as the solution to (32):

$$\underset{X \in \mathbb{R}_*^{n_Z \times n}}{\text{minimize}} \quad \sum_{\mu \in \mathcal{D}_{\text{sample}}} \sum_{j=1}^J \left\| X^T Z \nabla_q V(q_0(\mu) + Z^T X q_r^j; \mu) - \Phi^T \nabla_q V(q_0(\mu) + \Phi q_r^j; \mu) \right\|_2^2. \quad (32)$$

The variables q_r^j , $j = 1, \dots, J$ denote different values of q_r , which may be encountered at different time steps and Newton iterations during training simulations.

Because the noncompact Stiefel manifold is an open set, problem (32) can be solved by first computing a solution in $\mathbb{R}^{n_z \times n}$ and subsequently projecting this solution onto $\mathbb{R}_*^{n_z \times n}$.

IV.C.1. Offline/online decomposition

To implement this approximation, the following steps are required offline:

1. Collect snapshots of the generalized state and potential-energy gradient:

$$\mathcal{X}_V \equiv \{(q_r^j(\mu), \Phi^T \nabla_q V(q_0(\mu) + \Phi q_r^j; \mu)) \mid \mu \in \mathcal{D}_{\text{sample}}, j = 1, \dots, J\}. \quad (33)$$

2. Determine the sampling matrix Z .
3. Solve (32) for $\underline{\Psi}$ using \mathcal{X}_V , and set $\Psi = Z^T \underline{\Psi}$.

The online stage simply requires employing \tilde{V}_r in lieu of V_r .

IV.D. Equations of motion and computational efficiency

The equations of motion can be derived by applying the forced Euler–Lagrange equation to the approximated quantities:

$$\frac{d}{dt} \nabla_{\dot{q}_r} \tilde{L}_r(q_r, \dot{q}_r; \mu) - \nabla_{q_r} \tilde{L}_r(q_r, \dot{q}_r; \mu) + \nabla_{\dot{q}_r} \tilde{\mathcal{F}}_r(\dot{q}_r; \mu) = \tilde{f}_r^{\text{ext}}, \quad (34)$$

where the approximated Lagrangian is defined as

$$\tilde{L}_r(q_r, \dot{q}_r; \mu) \equiv \frac{1}{2} \tilde{g}_r(\dot{q}_r, \dot{q}_r; \mu) - \tilde{V}_r(q_r; \mu). \quad (35)$$

Eq. (34) can be written as

$$h_{\text{SPD}}(M(\mu), \underline{M}) \ddot{q}_r + h_{\text{SP}}(C(\mu), \underline{C}) \dot{q}_r + \Psi^T \nabla_q V(q_0(\mu) + \Psi q_r; \mu) = \tilde{f}_r^{\text{ext}}. \quad (36)$$

The cost of computing the components of the equations of motion for the proposed reduced-order model is independent of N . For this reason, we expect the reduced-order model to be computationally inexpensive to use in the online stage. To summarize, the online computations needed to compute each term of the proposed reduced-order model are as follows:

- The forcing term requires 1) computing $Z f^{\text{ext}}(t; \mu) \in \mathbb{R}^{n_z}$, which necessitates computing only $n_z \ll N$ entries of the forcing vector, and 2) computing the low-dimensional matrix–vector product

$$\left[\Phi^T \Phi_f [Z \Phi_f]^+ \right] Z f^{\text{ext}}(t; \mu),$$

where the matrix $\Phi^T \Phi_f [Z \Phi_f]^+ \in \mathbb{R}^{n \times n_z}$ is computed in the offline stage.

- Computing $h_{\text{SPD}}(M(\mu), \underline{M})$ and $h_{\text{SP}}(C(\mu), \underline{C})$ require 1) computing only $n_e \ll N$ entries of the large-scale matrices $M(\mu)$ and $C(\mu)$, respectively, and 2) solving the small-scale optimization problem (30).
- The potential-energy term requires 1) computing $Z \nabla_q V(q_0(\mu) + \Psi q_r; \mu)$, which necessitates computing only $n_z \ll N$ entries of potential-energy gradient, and 2) the low-dimensional matrix–vector product $\underline{\Psi}^T (Z \nabla_q V(q_0(\mu) + \Psi q_r; \mu))$. Note that the first of these will be inexpensive if $\nabla_q V$ is sparse; this is the case for finite-element models characterized by basis functions with compact support.

So, the proposed reduced-order model preserves the problem’s structure, and is computationally inexpensive to employ. The next section applies the method to a simple structural-dynamics example.

Remark. This work does not specify a method for selecting the sampling matrix Z or sample elements \mathcal{E} . In fact a different sampling matrix Z could be used for the external force and potential-energy function. Future work will address this topic. The numerical experiments in the next section use the GNAT model-reduction method’s approach for selecting the sample matrix.¹¹

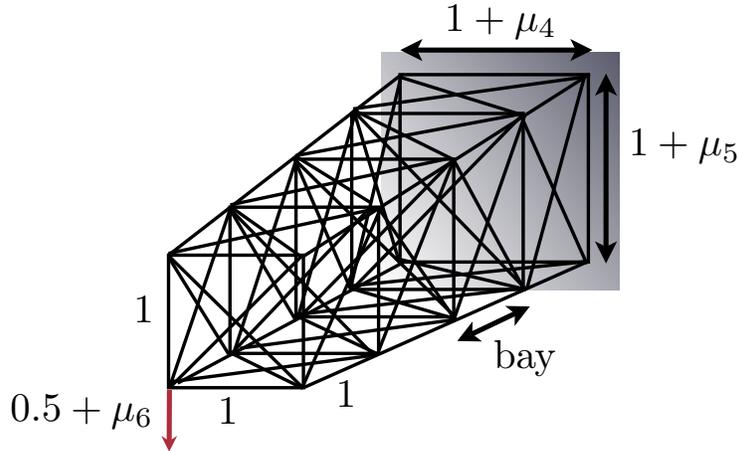


Figure 1. Clamped-free parameterized truss structure problem

V. Numerical experiments

This example considers a conservative structural-dynamics system. That is, damping and external forces are omitted ($F_n(t, q, \dot{q}; \mu) = 0$). As a result, the system is energy-conserving and has a symplectic time-evolution map. Because our proposed reduced-order model preserves such structure, the reduced-order model will also describe a conservative mechanical system. As previously discussed, structure-preserving integrators can be applied to such problems to ensure these properties are conserved. To this end, we apply the midpoint rule (a symplectic integrator) to solve the equations of motion.

Figure 1 depicts the parameterized clamped-free truss structure, where the arrow indicates the initial vertical displacement. We consider a problem with 10 bays. The high-fidelity model is constructed by the finite-element method. It consists of sixteen three-dimensional bar elements per bay with three degrees of freedom per node; this results in 12 degrees of freedom per bay, which leads to 1.2×10^2 degrees of freedom in the high-fidelity model. The bar elements model geometric nonlinearity, which results in a high-order nonlinearity in the strain energy. Each bay has (unitless) depth of 1, with the cross-sectional area determined by the parameterized geometry. Table 1 summarizes the effect of the $p = 6$ parameters on the model; here, $\mu_i \in [-1, 1]$, $i = 1, \dots, 6$.

density	bar cross-sectional area	modulus of elasticity	base width	base height	initial tip displacement
$1 + \mu_1$	$1 + \mu_2$	$1 + \mu_3$	$1 + \mu_4$	$1 + \mu_5$	$0.5 + \mu_6$

Table 1. Parameterization of the truss structure, $\mu_i \in [-1, 1]$, $i = 1, \dots, 6$. All attributes are dimensionless.

The total time is set to $T = 20$. At each time step, a system of nonlinear equations arises due to the use of the midpoint rule, which is implicit. To solve these systems, Newton's method is applied. Each linearized system is solved directly using the Cholesky factorization (the Jacobian of the residual is symmetric), and convergence is declared when the residual is less than or equal to $10^{-5}\delta$, where δ is the maximum of 1.0 and the norm of the state at the previous time step. A time-step size of $\Delta t = 1/3$ is employed; this value was determined by a convergence study on the nominal configuration defined by $\mu_i = 0$, $i = 1, \dots, 6$.

To construct the reduced-order models, we collect snapshots of the required quantities for $\mu \in \mathcal{D}_{\text{sample}}$ and $t \in [0, T_{\text{sample}}]$, where $T_{\text{sample}} = 10$. The basis Φ is determined via POD. Snapshots $\{q(t + n\Delta t + \frac{\Delta t}{2}; \mu) - q(0; \mu) \mid \mu \in \mathcal{D}_{\text{sample}}, n = 1, \dots, n_t\}$, where n_t denotes the total number of time steps, are employed as inputs to Algorithm 1. The experiments compare four reduced-order models: Galerkin projection (Section III.A), Galerkin projection with collocation (Section III.B.1), Galerkin projection with least-squares reconstruction (Section III.B.2), and the proposed efficient, structure-preserving method (Section IV). The same sampling matrix Z is used for the all sampling-based models; it is determined using the GNAT model-reduction method's approach for selecting the sample matrix.¹¹ The sample elements are defined such that $\mathcal{E} = \{(i_m, j_m) \mid i_m, j_m \in \{i_1, \dots, i_{n_Z}\}\}$, where $Z = [e_{i_1} \ \dots \ e_{i_{n_Z}}]^T$.

In practice, we found the constraints to always be inactive at the unconstrained solution to (43); therefore, this problem reduces to a linear least-squares problem. To solve optimization problem (32), we use the Poblano toolbox;¹⁷ the initial guess for Ψ is chosen as $Z^T Z \Phi$.

The output of interest is the downward tip displacement (denoted by d) for all time steps. To quantify the error in solutions generated by reduced-order models, the following metrics are used:

$$e_{\text{HFM}}^1 = \frac{\frac{1}{n_t} \sum_{n=0}^{n_t} |d^n - d_{\text{HFM}}^n|}{\max_n d_{\text{HFM}}^n - \min_n d_{\text{HFM}}^n} \quad (37)$$

$$e_{\text{Gal}}^1 = \frac{\frac{1}{n_t} \sum_{n=0}^{n_t} |d^n - d_{\text{Gal}}^n|}{\max_n d_{\text{Gal}}^n - \min_n d_{\text{Gal}}^n} \quad (38)$$

$$e_{\text{HFM}}^\infty = \frac{\max_n |d^n - d_{\text{HFM}}^n|}{\max_n d_{\text{HFM}}^n - \min_n d_{\text{HFM}}^n}. \quad (39)$$

$$e_{\text{Gal}}^\infty = \frac{\max_n |d^n - d_{\text{Gal}}^n|}{\max_n d_{\text{Gal}}^n - \min_n d_{\text{Gal}}^n}. \quad (40)$$

Here, a subscript ‘HFM’ denotes a quantity computed using the high-fidelity model, and a subscript ‘Gal’ denotes a quantity computed with the Galerkin-projection-based reduced-order model (with no complexity-reduction technique).

V.A. Fixed parameters

We first consider the case of a single training point with online parameters equivalent to the training parameters: $\mathcal{D}_{\text{sample}} = \mu^1$ and $\mu^* = \mu^1$. The parameters are randomly drawn from a uniform distribution on $[-1, 1]$ as

$$\mu^1 = (-0.685, 0.941, 0.914, -0.029, 0.601, 0.502).$$

The dimension of the reduced configuration space is selected using an energy criterion. In particular, $n = n_e(0.99) = 10$ is computed by

$$n_e(\nu) \equiv \arg \min_{i \in \mathcal{V}(\nu)} i \quad (41)$$

$$\mathcal{V}(\nu) \equiv \{n \in \{1, \dots, n_w\} \mid \sum_{i=1}^n \sigma_i^2 / \sum_{i=1}^{n_w} \sigma_i^2 \geq \nu\}. \quad (42)$$

For the Galerkin projection with least-squares reconstruction method (Section III.B.2), we treat the entire residual as the nonlinear function and set $n_f = n = 10$.

The number of rows n_Z in the sample matrix is chosen experimentally to be 30.^d Convergence in the solution of (32) is declared when the iterations satisfy a gradient-norm stopping tolerance or relative-function value-change stopping tolerance¹⁷ of 10^{-5} . Figure 2 and Table 2 report the results for this problem.

	Galerkin	Galerkin + collocation	Galerkin + LS recon.	structure- preserving
e_{HFM}^1	6.20%	14.6%	14.1%	4.38%
e_{Gal}^1	-	10.4%	10.4%	5.53%
e_{HFM}^∞	26.6%	62.9%	65.8%	12.4%
e_{Gal}^∞	-	73.7%	76.4%	18.6%
speedup	0.38	1.85	1.90	1.88

Table 2. Performance of reduced-order models for fixed parameters

^dThis leads to an aspect ratio of the least-squares matrix of 3.0 for the GNAT method,¹¹ which has been shown to lead to good results.

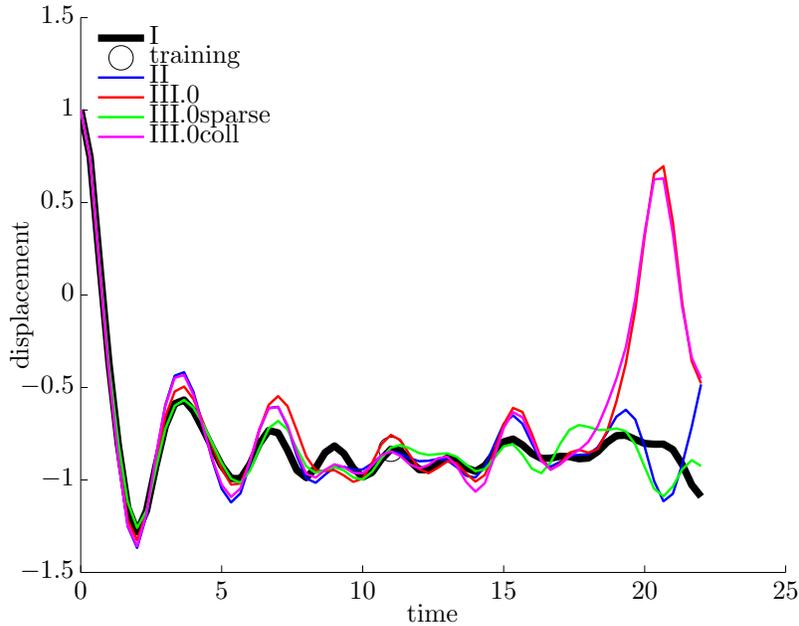


Figure 2. Responses generated by all models for a fixed set of parameters

For this example, the benefits of structure preservation are clear. Both methods that preserve Lagrangian structure—Galerkin projection and the proposed approach—lead to reasonably accurate, stable responses. This can be associated with the fact that the discrete solution behaves as a Lagrangian system (i.e., symplectic flow) when both the model and numerical-integration method (e.g., midpoint rule) preserve properties associated with Lagrangian structure. In particular, note that the approximations introduced by the proposed structure-preserving method introduce very little error to the Galerkin-projection reduced-order model: the proposed method generates a response that differs from the Galerkin-projection solution by less than 3% ($e_{\text{Gal}}^1 = 0.91\%$). The other reduced-order models (which destroy Lagrangian structure) lead to inaccurate responses.

Further, note that Galerkin projection alone is insufficient to generate speedups; in fact, a speedup of $0.38 < 1.0$ indicates that this reduced-order model takes a *longer* time to simulate than the high-fidelity model. This can be attributed to the computational bottleneck discussed in Section III.A.1. On the other hand, the proposed method achieves a speedup of $1.88 > 1$; for larger problems, we expect this speedup to be significantly larger. Thus, the proposed technique is the only method tested that leads to reasonable errors ($e_{\text{HFM}}^1 < 10\%$) and decreased simulation times.

V.B. Variable parameters

This study employs three training points $\mathcal{D}_{\text{sample}} = \{\mu^i, i = 1, \dots, 3\}$ and an ‘online’ point $\mu^* \notin \mathcal{D}_{\text{sample}}$. The points were randomly computed as

$$\begin{aligned}\mu^1 &= (-0.685, 0.941, 0.914, -0.029, 0.601, 0.502) \\ \mu^2 &= (-0.633, -0.277, 0.778, 0.523, -0.928, 0.466) \\ \mu^3 &= (-0.567, 0.603, 0.0989, 0.558, -0.517, 0.249) \\ \mu^* &= (-0.716, -0.156, 0.831, 0.584, 0.919, 0.468).\end{aligned}$$

Again, an energy criterion is used to determine the dimension of the reduced configuration space as $n = n_e(0.99) = 18$. The number of sample indices is set to 31. Once more, the Galerkin projection with least-squares reconstruction treats the entire residual as the nonlinear function and employs $n_f = n = 10$.

Figure 3 and Table 3 report the results for this problem. Again, note that methods that do not preserve structure lead to poor responses. This is particularly pronounced with the least-squares-reconstruction approach; here, the response becomes unstable almost immediately.

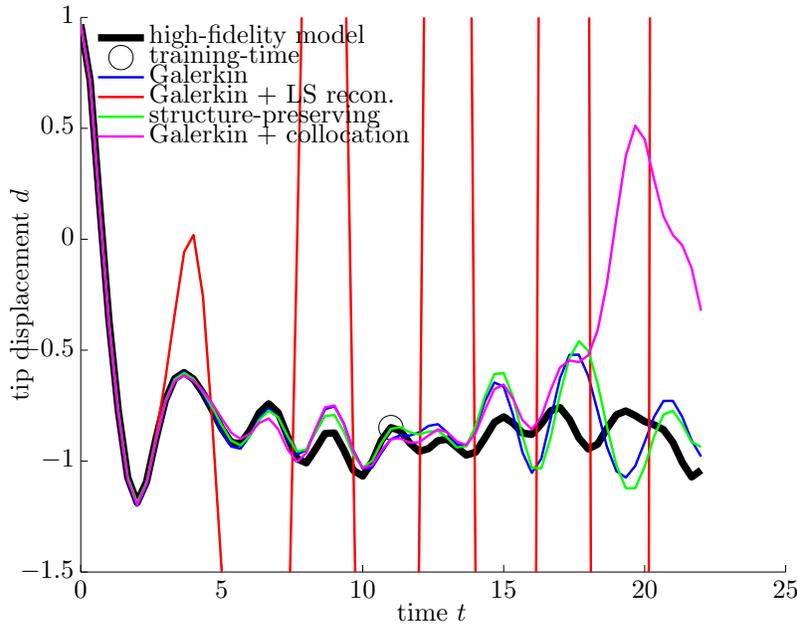


Figure 3. Responses generated by all models for a multiple sets of training parameters in a predictive scenario

	Galerkin	Galerkin + collocation	Galerkin + LS recon.	structure-preserving
e_{HFM}^1	6.85%	18.7%	690%	7.0%
e_{Gal}^1	-	16.2%	679%	2.6%
e_{HFM}^∞	17.4%	60.7%	2324%	20.5%
e_{Gal}^∞	-	70.7%	2299%	5.7%
speedup	0.41	1.77	2.06	1.82

Table 3. Performance of reduced order models for multiple parameters (predictive scenario)

In contrast, both methods that preserve Lagrangian structure lead to stable, reasonably accurate responses ($e_{\text{HFM}}^1 < 10\%$) for this predictive scenario. Again, we observe that the proposed method's approximations introduce very little error to the Galerkin-projection reduced-order model, as $e_{\text{Gal}}^1 = 2.6\%$ for the proposed method.

We again observe that the proposed method is the only model-reduction method that delivers both accuracy and faster simulation times. This highlights the importance of improving computational efficiency while preserving Lagrangian structure.

VI. Conclusions

This paper has presented an efficient structure-preserving model-reduction technique for simple mechanical systems. The methodology directly approximates the quantities that define the problem's Lagrangian structure, while ensuring computational efficiency. The method is distinct from typical model-reduction methods for nonlinear ODEs; these methods are typically based on collocation and empirical interpolation/least-squares reconstruction techniques that destroy Lagrangian structure.

Future work includes performing a parameter study on the parameters defining the method, applying the method to a truly large-scale problem, devising methods to select sampling matrix and sample elements, and deriving error bounds and error estimates that rigorously assess the accuracy of the method's predictions.

Appendix

VI.A. Proper orthogonal decomposition

Algorithm 1 describes the method for computing a proper-orthogonal-decomposition (POD) basis given a set of snapshots. The method essentially amounts to computing the singular value decomposition of the snapshot matrix. The left singular vectors define the POD basis.

Algorithm 1 Proper-orthogonal-decomposition basis computation

Input: Set of snapshots $\mathcal{X} \equiv \{w^i\}_{i=1}^{n_w}$ with $w^i \in \mathbb{R}^{N_w}$, $i = 1, \dots, n_w$ and $N_w > n_w$

Output: $\Phi(n, \mathcal{X})$

- 1: Compute thin singular value decomposition $W = V\Sigma V^T$, where $W \equiv [w^1 \ \dots \ w^{n_w}]$
 - 2: Choose dimension of truncated basis $n \in \{1, 2, \dots, n_w\}$
 - 3: $\Phi(n, W) = W \begin{bmatrix} \frac{1}{\sigma_1} v^1 & \dots & \frac{1}{\sigma_n} v^n \end{bmatrix}$, where $V = [v^1 \ \dots \ v^{n_w}]$ and $\Sigma \equiv \text{diag}(\sigma_1, \dots, \sigma_{n_w})$
-

VI.B. Solving the constrained gappy-data-reconstruction optimization problem

This approach reformulates the constraints of (30) in terms of eigenvalues of the reduced matrix: That is, problem (30) is reformulated as

$$\begin{aligned} & \underset{x}{\text{minimize}} && \sum_{m=1}^{n_e} \left(A_{i_m, j_m} - \sum_{k=1}^{n_A} \underline{A}_{i_m, j_m}^k x_k \right)^2 \\ & \text{subject to} && \tilde{\lambda}_j(x) \geq \epsilon, \quad j = 1, \dots, n, \end{aligned} \quad (43)$$

Here, $\tilde{\lambda}_j(x)$, $j = 1, \dots, n$ are the eigenvalues of $\sum_{k=1}^{n_A} \Phi^T \underline{A}^k \Phi x_k$. These problems can be solved using any gradient-based approach.

The gradient of the quadratic objective function is obvious. The gradient of the constraint can be derived by assuming distinct eigenvalues:

$$\frac{\partial \tilde{\lambda}_j}{\partial x_i} = \tilde{y}_j^T \frac{\partial \left(\sum_{k=1}^{n_A} \Phi^T \underline{A}^k \Phi x_k \right)}{\partial x_i} \tilde{y}_j \quad (44)$$

$$= \tilde{y}_j^T (\Phi^T \underline{A}^i \Phi) \tilde{y}_j. \quad (45)$$

Here, \tilde{y}_j is the eigenvector associated with eigenvalue $\tilde{\lambda}_j$. This indicates that computing the gradient $\frac{\partial \tilde{\lambda}_j}{\partial \xi^i}$ is inexpensive and requires the following steps:

1. Compute the eigenvector $\tilde{y}_j \in \mathbb{R}^n$ of the matrix $\sum_{k=1}^{n_A} \Phi^T \underline{A}^k \Phi x_k$.
2. Compute the low-dimensional matrix-vector product $w = (\Phi^T \underline{A}^i \Phi) \tilde{y}_j$.
3. Compute the low-dimensional vector-vector product $\tilde{y}_j^T w$.

We propose using the unconstrained solution to (43) as the initial guess. In practice, this solution is often feasible, so it is typically unnecessary to handle the constraints directly.

Acknowledgments

The authors acknowledge Julien Cortial for both insightful discussions and for providing the original nonlinear-truss code that was modified to generate the numerical results.

This research was supported in part by an appointment to the Sandia National Laboratories Truman Fellowship in National Security Science and Engineering, sponsored by Sandia Corporation (a wholly owned subsidiary of Lockheed Martin Corporation) as Operator of Sandia National Laboratories under its U.S. Department of Energy Contract No. DE-AC04-94AL85000. The authors also acknowledge support by the Department of Energy Office of Advanced Scientific Computing Research under contract 10-014804.

References

- ¹Hairer, E., Lubich, C., and Wanner, G., *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, Vol. 31, Springer Verlag, 2006.
- ²Marsden, J. and West, M., “Discrete mechanics and variational integrators,” *Acta Numerica*, Vol. 10, No. 357, 2001, pp. 514.
- ³Lall, S., Krysl, P., and Marsden, J., “Structure-preserving model reduction for mechanical systems,” *Physica D: Nonlinear Phenomena*, Vol. 184, No. 1-4, 2003, pp. 304–318.
- ⁴Astrid, P., Weiland, S., Willcox, K., and Backx, T., “Missing point estimation in models described by proper orthogonal decomposition,” *IEEE Transactions on Automatic Control*, Vol. 53, No. 10, 2008, pp. 2237–2251.
- ⁵Ryckelynck, D., “A priori hyperreduction method: an adaptive approach,” *Journal of Computational Physics*, Vol. 202, No. 1, 2005, pp. 346–366.
- ⁶Grepl, M. A., Maday, Y., Nguyen, N. C., and Patera, A. T., “Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations,” *ESAIM-Mathematical Modelling and Numerical Analysis (M2AN)*, Vol. 41, No. 3, 2007, pp. 575–605.
- ⁷Nguyen, N. C. and Peraire, J., “An efficient reduced-order modeling approach for non-linear parametrized partial differential equations,” *International Journal for Numerical Methods in Engineering*, Vol. 76, February 2008, pp. 27–55.
- ⁸Chaturantabut, S., Sorensen, D. C., and Steven, J. C., “Nonlinear model reduction via discrete empirical interpolation,” *SIAM Journal on Scientific Computing*, Vol. 32, No. 5, 2010, pp. 2737–2764.
- ⁹Galbally, D., Fidkowski, K., Willcox, K., and Ghattas, O., “Non-linear model reduction for uncertainty quantification in large-scale inverse problems,” *International Journal for Numerical Methods in Engineering*, published online September 2009.
- ¹⁰Everson, R. and Sirovich, L., “Karhunen–Loève procedure for gappy data,” *Journal of the Optical Society of America A*, No. 8, 1995, pp. 1657–1664.
- ¹¹Carlberg, K., Bou-Mosleh, C., and Farhat, C., “Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations,” *International Journal for Numerical Methods in Engineering*, Vol. 86, No. 2, April 2011, pp. 155–181.
- ¹²Carlberg, K., Cortial, J., Amsallem, D., Zahr, M., and Farhat, C., “The GNAT nonlinear model reduction method and its application to fluid dynamics problems,” *AIAA Paper 2011-3112, 6th AIAA Theoretical Fluid Mechanics Conference, Honolulu, HI*, June 27–30, 2011.
- ¹³Adhikari, S., *Damping models for structural vibration*, Ph.D. thesis, Cambridge University, September 2000.
- ¹⁴Ito, K. and Ravindran, S., “A reduced basis method for control problems governed by PDEs,” *Control and estimation of distributed parameter systems*, 1998, pp. 153–168.
- ¹⁵Prud’homme, C., Rovas, D. V., Veroy, K., Machiels, L., Maday, Y., Patera, A. T., and Turinici, G., “Reliable real-time solution of parameterized partial differential equations: Reduced-basis output bound methods,” *Journal of Fluids Engineering*, Vol. 124, No. 1, 2002, pp. 70–80.
- ¹⁶An, S., Kim, T., and James, D., “Optimizing cubature for efficient integration of subspace deformations,” *ACM Transactions on Graphics (TOG)*, Vol. 27, No. 5, 2008, pp. 165.
- ¹⁷Dunlavy, D., Kolda, T., and Acar, E., “Poblano v1. 0: A Matlab toolbox for gradient-based optimization,” *Sandia National Laboratories, Albuquerque, NM and Livermore, CA, Tech. Rep. SAND*, Vol. 1422, 2010.