

A numerical comparison of simplified Galerkin and machine learning reduced order models for vaginal deformations

William Snyder^a, Alex Santiago Anaya^a, Justin Krometis^b, Traian Iliescu^c, Raffaella De Vita^{a,*}

^a STRETCH Lab, Department of Biomedical Engineering and Mechanics, Virginia Tech, 330A Kelly Hall, 325 Stanger Street, Blacksburg, 24061, VA, USA

^b National Security Institute, Virginia Tech, 1311 Research Center Drive, RM 2016, Blacksburg, 24061, VA, USA

^c Department of Mathematics, Virginia Tech, 428 McBryde Hall, 225 Stanger Street, Blacksburg, 24061, VA, USA

ARTICLE INFO

Keywords:

Galerkin reduced order modeling
Machine learning
Finite elements
Vaginal tearing
Maternal trauma

ABSTRACT

High-fidelity computer simulations of childbirth remain prohibitively expensive and time consuming, making them impractical for guiding decision-making during obstetric emergencies. Cheap computer simulations that preserve the accuracy of high-fidelity models can be developed using surrogate modeling. Two common approaches to surrogate modeling are physics-based reduced order modeling (ROM) and machine learning (ML), with the latter gaining popularity as the scientific computing community seeks to leverage advances from other, mostly non-physics-based, computational strategies. Although ROM and ML have been compared for various problems, to our knowledge, such a comparison for simulations of vaginal deformations is currently missing. This study provides a baseline numerical comparison between methods from these two fundamentally different approaches. Since there are many methods falling into each modeling approach, to provide a fair and natural comparison, we select a basic model from each category, with each allowing (i) a straightforward implementation in commercial software packages, and (ii) use by practitioners with limited experience in the field. As a benchmark for the numerical comparison of the ROM and ML approaches, we use the finite element (FE) modeling of the *ex vivo* deformations of rat vaginal tissue subjected to inflation testing to study the effect of a pre-imposed tear. From the ROM strategies, we consider a simplified Galerkin ROM (G-ROM) that is based on the linearization of the underlying nonlinear equations. From the ML strategies, we select a feed-forward neural network to create mappings from constitutive model parameters and luminal pressure values to either the FE displacement history (in which case we denote the resulting model ML) or the proper orthogonal decomposition (POD) coefficients of the displacement history (in which case we denote the resulting model POD-ML). The numerical investigation of G-ROM, ML, and POD-ML takes place in the reconstructive regime. The numerical results show that the G-ROM outperforms the ML model in terms of offline central processing unit (CPU) time for model training, online CPU time required to generate approximations, and relative error with respect to the FE models. The G-ROM achieves superior error performance to the best ML model with 11 POD basis functions. With higher-dimensional POD bases, the G-ROM achieves a relative error 3 orders of magnitude lower than that of the best ML model with an online CPU time still on the same order of magnitude as the best ML model. The POD-ML model improves on the speed performance of the ML, having online CPU times comparable to those of the G-ROM given the same size of POD bases. However, the POD-ML model does not improve on the error performance of the ML and is still outperformed by the G-ROM for POD bases of size greater than 11. This baseline numerical investigation serves as a starting point for future computer simulations that consider state-of-the-art G-ROM and ML strategies, and the *in vivo* geometry, boundary conditions, and material properties of the human vagina, as well as their changes during labor.

1. Introduction

During childbirth, vaginal tearing is a common occurrence with about 80% of vaginal deliveries resulting in some degree of tissue lac-

eration [1]. The severity of these injuries varies from small tears that cause little or no harm to large tears that propagate to the muscles of the pelvic floor resulting in long-term complications such as fecal incontinence, urinary incontinence, sexual dysfunction, and prolapse [2].

* Corresponding author.

E-mail address: devita@vt.edu (R. De Vita).

<https://doi.org/10.1016/j.camwa.2023.10.018>

Known risk factors for vaginal tears include birth weight, forceps delivery, and prolonged second stage of labor [3]. However, no clinical technique in obstetrics exists to accurately anticipate the occurrence and severity of vaginal tears as well as their propagation to other pelvic tissues. Real-time non-invasive prenatal methods are needed to predict vaginal tearing during childbirth and establish preventative measures and reduce maternal trauma and morbidity. *In silico* methods that predict deformations and tears experienced by the vagina during childbirth in real time have the potential to become viable non-invasive prognostic models in obstetrics.

The finite element (FE) method is one of the most popular approaches used to simulate childbirth. The main characteristics and findings of current FE models that investigate the process of childbirth, maternal injuries, fetal injuries, and protective clinical measures have been recently reviewed [4]. In particular, the tears in the pelvic floor muscles have been analyzed using FE methods by Oliveira et al. [5–7] with the goal of modeling episiotomy during childbirth. However, deformations of the vagina (and surrounding tissues) with tears have never been described by FE models. Unfortunately, FE models of the childbirth are computationally expensive due to the complex geometries and boundary conditions and the nonlinear constitutive models that are required to characterize the mechanical behavior of the tissues of the pelvic floor. Thus, the FE method remains impractical for real-time predictions of the clinical outcomes of vaginal delivery.

The computational cost of full order models (FOMs) such as FE models can be reduced by adopting reduced order modeling (ROM) strategies [8–11]. ROM techniques have been applied to simulate the deformations of a variety of soft tissue during surgical procedures [12]. Proper orthogonal decomposition (POD) and proper generalized decomposition methods have been implemented for real-time simulations of the liver and cornea [13–16]. Sophisticated simulations of cardiac tissues have been approximated via reduced basis approaches [17–19]. Deformations of the inferior turbinate have been modeled with various ROM techniques as well [20,21]. Recently, we adopted Galerkin ROM (G-ROM) methods to simulate the experimentally-observed deformations of vaginal tissue [22]. However, ROMs have not been used to simulate how the presence of vaginal tears affects the deformations of the vagina.

Alternative methods used to speed up FE models are machine learning (ML) techniques. These techniques have been applied, in conjunction with data generated by FE simulations, to model the mechanical behavior of soft tissues in real time. Simple regression and decision tree-based ML techniques have been used to predict deformations of breast tissue under compression [23] and the liver during breathing [24]. Support vector regression and artificial neural networks (NNs) have been employed to simulate deformations of the brain afflicted by a tumor [25]. Additionally, the stresses of atherosclerotic arterial walls have been described by deep neural networks [26]. To reduce output dimensionality and the computational cost of modeling complex systems, ML models can incorporate POD used in G-ROM. This POD-ML approach has been applied to a variety of problem cases, such as modeling cardiac electrophysiology [27] or electrostatics and fluid dynamics [28]. To the authors' knowledge, neither ML nor POD-ML techniques have been implemented to simulate the deformations of reproductive tissues or the mechanics of childbirth.

This study presents the first comparison of two fundamentally different techniques for reduced order modeling, the Galerkin projection-based ROM and the data-driven NN, to approximate FE-based simulations of torn vaginal tissue. Building on a FOM framework [22] that captures the *ex vivo* micro-structural and mechanical behavior of the rat vagina under inflation, we produce new FE simulations of the organ having a pre-imposed tear along the axial direction by changing the microstructure of the organ (i.e., mean preferred fiber orientations) and applied luminal pressure (Section 2.1). Both ROM and ML techniques are used to approximately reconstruct the linearized FOM solutions produced by the final Newton-Raphson iterations of our FE solver (Sec-

tions 2.2, 2.3, 2.4), and each technique's performance is measured in a Pareto space composed of relative error with respect to the FOM and central processing unit (CPU) time required to produce approximations (Section 3). We then compare the two techniques, assessing the advantages and limitations of each (Section 4), and, finally, summarize our preliminary investigation (Section 5) evaluating the potential of G-ROM and ML strategies for the development of real-time computational tools to predict vaginal tissue tearing during childbirth.

We emphasize that G-ROM and ML methods used in our numerical investigation are not state-of-the-art in their respective classes. Instead, we utilize simple G-ROM and ML strategies that allow an easy implementation in available software packages such as Abaqus and TensorFlow, which facilitates their use by practitioners with limited experience in ROM and ML. We also stress that the conclusions of our numerical investigation are valid only for the two simple G-ROM and ML methods used, and cannot be extrapolated to the entire class of G-ROM and ML strategies. We believe, however, that this preliminary numerical investigation can serve as a stepping stone toward more realistic settings (e.g., complex geometry, boundary conditions, loading conditions, and constitutive descriptions) which could provide insight into the use of ROMs in computer simulations of childbirth.

2. Methods

2.1. Full order model

In this section, we present the FE models that are used to capture the deformations of rat vaginal canals with tears subjected to increasing luminal pressure. We first describe the selection of geometry, boundary conditions, and constitutive parameters of these models and then the solution methods used to obtain training data to create G-ROM, ML, and POD-ML models.

2.1.1. Geometry, boundary conditions, and constitutive parameters

In this study, we construct an FE model that describes deformations of rat vaginas that are induced by increasing luminal pressure. All FE simulations were carried out using Abaqus/Standard (Abaqus 2020, Dassault Systèmes Americas Corp., Waltham, MA) on a 24-core Intel® Xeon® Gold6248R CPU @ 3.00 GHz with 191 GB of usable RAM. Each simulation used a linear ramp function to increase luminal pressure from 0 kPa to a given target pressure. Abaqus outputs for displacement vectors were recorded at the final step of each ramp function to create a quasi-static snapshot corresponding to the given target pressure on the lumen.

The geometry, boundary conditions, constitutive model, and associated material parameters of our model were presented in detail in our previous study [22]. However, in the current FE model, we considered the effect of a pre-imposed tear in the form of an elliptical hole that had a length of 5 mm and a width at mid-span of 1.08 mm. The major axis of the tear was aligned along the axial direction of the vagina and the minor axis was aligned along the hoop direction and centered at mid-height. Fig. 1 displays the tear in the vagina, the other dimensions of the hollow prolate spheroid which approximates the vaginal geometry, the boundary conditions, and the coordinate system of the FE model. The compressible Holzapfel-Gasser-Ogden (HGO) model for anisotropic hyperelastic materials with two families of fibers was selected to describe the mechanical response of the vagina [29]. The material parameters of this constitutive model were constant ($c = 6$ kPa, $k_1 = 15$ MPa, $k_2 = 15$, $D = 0.05$, $\kappa = 0.25$, $N = 2$) with the exception of the mean preferred fiber orientations.

The vagina was divided in three anatomical regions (distal, mid, and proximal) as shown in Fig. 1(a), each with two different mean preferred fiber directions. The mean preferred fiber directions of the two families of fibers, $\mathbf{a}_1 = (0, \cos \beta, \sin \beta)$ and $\mathbf{a}_2 = (0, \cos \beta, -\sin \beta)$, and the fiber dispersion parameter, κ , were chosen to approximately represent

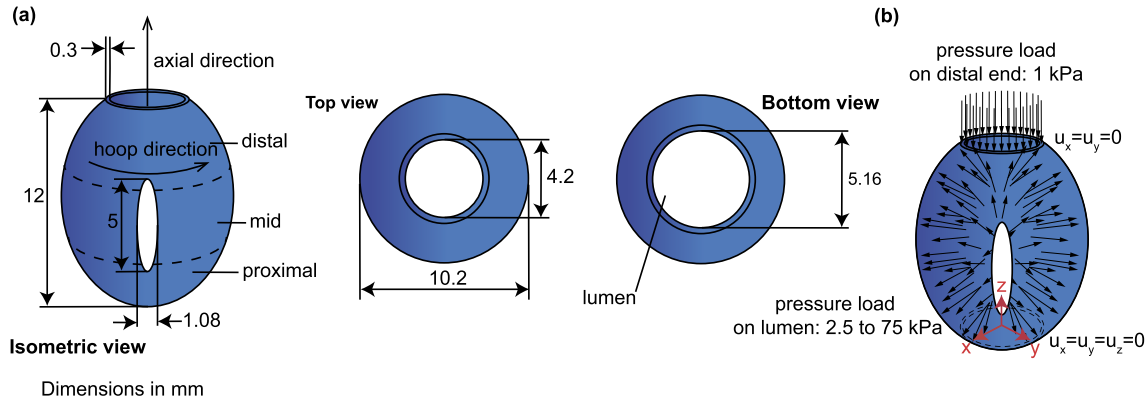


Fig. 1. (a) Isometric, top, and bottom views of the rat vaginal specimen with dimensions and anatomical (proximal, mid, and distal) regions. (b) Boundary conditions for the rat vaginal specimen used to simulate inflation testing. The annotations u_x , u_y , and u_z denote the translational displacements of nodes at the boundaries. The uppermost distal surface was fixed in the hoop and radial directions and subjected to a constant pressure in the axial direction, while the lowermost proximal surface was fixed.

Table 1

Mean preferred fiber orientations, β_d , β_m , and β_p , for the distal, mid, and proximal regions of the vagina, respectively, used in the FE simulations. The orientations were defined relative to the hoop direction of the vagina.

Parameter Set	β_d	β_m	β_p
μ_1	35°	35°	55°
μ_2	35°	35°	65°
μ_3	25°	25°	65°
μ_4	35°	25°	55°
μ_5	25°	35°	55°
μ_6	35°	25°	65°
μ_7	25°	35°	65°
μ_8	25°	25°	55°

the experimentally measured collagen fiber organization in the tangential (hoop-axial) plane of the vagina (refer to Figure 10 in [30]). The vectors $\mathbf{a}_1$ and $\mathbf{a}_2$ are defined using a pseudo-cylindrical local Cartesian coordinate system described in detail in our previous study [22]. The angle β was defined relative to the hoop direction of the vagina, i.e., the hoop direction was at a $\beta = 0^\circ$ angle. In each of the three anatomical regions of the vagina, the two families of fibers had mean preferred fiber orientations defined by $\pm\beta$. The mean preferred fiber orientations in the distal, mid, and proximal regions were assumed to differ based on experimental data [30] and be defined by the angles $\pm\beta_d$, $\pm\beta_m$, and $\pm\beta_p$, respectively.

2.1.2. Finite element method

In this section, we present the FE method that was used to generate the FOM data. In our numerical investigation, the FOM results served as training data for the ROM (Section 2.2), the ML (Section 2.3), and the POD-ML (Section 2.4) strategies. They were also used as benchmark in the evaluation of the ROM, ML, and POD-ML results.

For each of the eight mean preferred fiber orientation combinations $\mu_1, \dots, \mu_8$ reported in Table 1, FE solutions at thirty luminal pressures, $p_1 = 2.5$ kPa, $p_2 = 5$ kPa, $p_3 = 7.5$ kPa, $\dots$, $p_{30} = 75$ kPa, were obtained. The pressure was incremented from 2.5 kPa to 75 kPa in equal steps of 2.5 kPa. In total, $n = 8 \times 30$ nonlinear systems of equations were solved in Abaqus using the Newton-Raphson method. The linearized FE system of equations resulting from the final Newton-Raphson iterations of the weak form of the equilibrium equations for each set of mean preferred fiber orientations and luminal pressures is the following [31]:

$$\mathbf{K}^{(i)} \mathbf{u}^{(i)} = \mathbf{f}^{(i)} \quad \text{for } i = 1, \dots, n, \quad (1)$$

where $\mathbf{K}^{(i)}$ is the $m \times m$ tangent stiffness matrix, $\mathbf{f}^{(i)}$ is the $m \times 1$ column vector of loads corresponding to a given set of fiber orientations and the discrete luminal pressure, and $\mathbf{u}^{(i)}$ is the $m \times 1$ column vector of displacements. The integer m refers to the number of degrees of freedom of each linearized system. We stored the solutions provided by the final Newton-Raphson iterations for each set of fiber orientations and luminal pressures into n vectors, $\{\mathbf{u}_{FOM}^{(1)}, \mathbf{u}_{FOM}^{(2)}, \dots, \mathbf{u}_{FOM}^{(n)}\}$, each belonging to $\mathbb{R}^m$. These vectors were then used to build and train the G-ROM and ML as described in Sections 2.2, 2.3, and 2.4

The vaginal tissue was meshed in Abaqus using the structured scheme of second-order quadratic hexahedral elements (C3D20) [32]. A mesh convergence study was performed via h -refinement with the maximum element size decreasing from 2 mm to 1 mm in steps of 0.2 mm. The convergence study considered only a single fiber orientation set, denoted by μ_1 in Table 1, for which the mean preferred fiber orientations by region were $\beta_d = \beta_m = 35^\circ$ and $\beta_p = 55^\circ$. The luminal pressure was set to 75 kPa. Meshes of different maximum element size, h , were compared using the mesh energy, E_h , given by

$$E_h = \frac{1}{2} (\mathbf{u}_h^\top \mathbf{K}_h \mathbf{u}_h), \quad (2)$$

where $\mathbf{u}_h$ is the $m \times 1$ displacement column vector, and $\mathbf{K}_h$ is the $m \times m$ tangent stiffness matrix of the linearized FE system for the chosen fiber orientation set and luminal pressure. The matrices $\mathbf{K}_h$ and vectors $\mathbf{u}_h$ are recorded from the solution provided by the last Newton iteration. To assess convergence, we calculate the mesh energy relative error defined as:

$$\mathcal{E}(h) = \frac{|E_h - E_{h_{min}}|}{|E_{h_{min}}|}, \quad (3)$$

where $E_{h_{min}}$ is the mesh energy associated with the smallest maximum element size considered in the convergence study (in this case, $h = 1$ mm). The mesh energy of the most refined mesh is used as the benchmark against which the other mesh energies, E_h , are compared.

The relative mesh energy error values are plotted against the number of degrees of freedom in the reduced linearized FE systems corresponding to each mesh size in Fig. 2. The “reduced system” excludes degrees of freedom at fixed nodes because they have displacement values of 0 and so do not contribute to the mesh energy. Mesh convergence was considered acceptable at $h = 1$ mm as there was indication of diminishing returns with increasing mesh refinement. The relative difference in mesh energy between $h = 1.2$ mm and $h = 1$ mm was only 3.28% compared to a 17.56% difference between $h = 1.4$ mm and $h = 1$ mm, and a 52.68% difference between $h = 1.6$ mm and $h = 1$ mm.

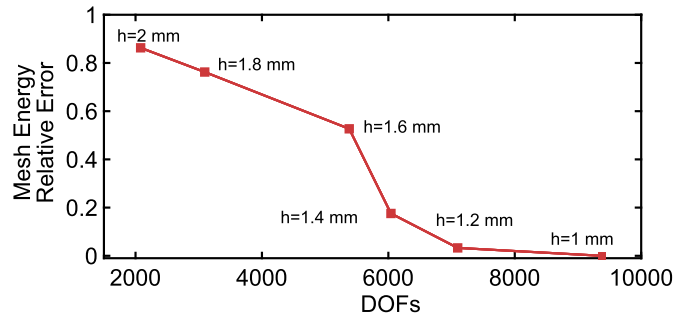


Fig. 2. Mesh convergence study via h -refinement using mean preferred fiber orientations given by $\beta_d = \beta_m = 35^\circ$ and $\beta_p = 55^\circ$ and luminal pressure $p_{30} = 75$ kPa. The mesh energy relative error is plotted against the number of degrees of freedom (DOFs) in the reduced linearized FE system.

The finest mesh contained 576 elements and 3419 nodes. With this mesh, given that there were 3 degrees of freedom for each node, the linear system described in Eqs. (1) had a total of 10257 linear equations.

2.2. Linearized Galerkin-ROM (G-ROM)

In this section, we outline the construction of the G-ROM for the linear systems in Eqs. (1). Moreover, we describe the criteria used to assess the G-ROM performance. Here, we use the POD method [9,11] to determine an orthonormal basis, the POD basis $\{\psi_i\}_{i=1}^l$ of size l , $l \leq \min\{m, n\}$, for the set spanned by n vectors, $\{u_{FOM}^{(1)}, u_{FOM}^{(2)}, \dots, u_{FOM}^{(n)}\}$, belonging to $\mathbb{R}^m$. These vectors represent the so-called “snapshots.” Thus, in our numerical investigation, the snapshots are solutions of the FE systems in Eqs. (1).

Let U be the $m \times n$ matrix whose columns are the snapshots: $U = [u_{FOM}^{(1)}, \dots, u_{FOM}^{(n)}]$. Let $d \leq \min\{m, n\}$ be the rank of U . To compute the POD basis $\{\psi_i\}_{i=1}^l$ of size l , we employ the method of snapshots [33,11], so we solve the symmetric $n \times n$ eigenvalue problem for ϕ_i :

$$U^T U \phi_i = \lambda_i \phi_i \quad \text{for } i = 1, \dots, d, \quad (4)$$

and then compute the POD basis as follows:

$$\psi_i = \frac{1}{\sqrt{\lambda_i}} U \phi_i \quad \text{for } i = 1, \dots, d. \quad (5)$$

Let Ψ be the $m \times l$ matrix whose columns are the POD basis vectors: $\Psi = [\psi_1, \dots, \psi_l]$. The $m \times 1$ column vector $u^{(i)}$ is approximated by the $m \times 1$ column vector

$$u_{G-ROM}^{(i)} = \Psi \zeta^{(i)}, \quad \text{for } i = 1, \dots, n, \quad (6)$$

where $\zeta^{(i)}$ is an unknown $l \times 1$ column vector that must be determined. Substituting $u_{G-ROM}^{(i)}$ into the linearized system of Eqs. (1) given by the final Newton-Raphson iteration of our FE solver yields the following linear system:

$$K^{(i)} \Psi \zeta^{(i)} = f^{(i)} \quad \text{for } i = 1, \dots, n. \quad (7)$$

Remark 2.1. We note that a classical G-ROM for our setting would be built by inserting the expansion (6) into the nonlinear equations, and then projecting the resulting system onto the ROM space spanned by $\{\psi_1, \dots, \psi_l\}$. This would yield a nonlinear system of equations for the unknown column vector (also known as POD coefficients), $\zeta^{(i)}$. In (7), however, we used a different approach. Specifically, instead of using the full nonlinear equations, we used the linearized FE system of equations in (1), which is the final iteration in the Newton-Raphson method used by Abaqus to solve the underlying nonlinear equations. We chose this linearized G-ROM since its implementation in commercial software packages such as Abaqus is extremely simple. Thus, practitioners with

limited experience in the field can use the linearized G-ROM with minimal effort.

We also note that this *a posteriori* application of POD-based ROM to the linearized equations resulting from FE discretization was demonstrated for modeling palpation of the human cornea with both anisotropic [14] and isotropic [15] hyperelasticity, as well as a model reduction of an extended FEM approach for limbal relaxing incisions on the cornea [34]. In addition, several techniques of model reduction have been applied to linearized FE equations for nonlinear material models of both academic test problems, such as cube compression and thick-walled cylinder compression, and practical biomechanical test problems, such as the inferior turbinate [20,21,35].

We then perform the Galerkin projection of Eqs. (7) onto the space spanned by the POD basis, $\{\psi_i\}_{i=1}^l$ (i.e., we multiply both terms of Eqs. (7) to the left by Ψ^T) to obtain the linearized G-ROM:

$$\Psi^T K^{(i)} \Psi \zeta^{(i)} = \Psi^T f^{(i)} \quad \text{for } i = 1, \dots, n. \quad (8)$$

The left side of Eqs. (8), $\Psi^T K^{(i)} \Psi$, is an $l \times l$ matrix, and the right side, $\Psi^T f^{(i)}$, is an $l \times 1$ column vector. The linear system in Eqs. (8) can be solved to find the $l \times 1$ column vector, $\zeta^{(i)}$. Thus, the m -dimensional FOM system of equations (1) is reduced to the much lower l -dimensional G-ROM system of equations (8) ($l \ll m$), which is computationally cheaper to solve.

We evaluate the G-ROM performance by evaluating the following error, $\mathcal{E}_{G-ROM}$, defined as

$$\mathcal{E}_{G-ROM} = \sqrt{\sum_{i=1}^n \mathcal{E}_{G-ROM}^{(i)}}, \quad (9)$$

with

$$\mathcal{E}_{G-ROM}^{(i)} = \frac{\|u_{G-ROM}^{(i)} - u_{FOM}^{(i)}\|_{\mathbb{R}^m}^2}{\|u_{FOM}^{(i)}\|_{\mathbb{R}^m}^2}, \quad (10)$$

where $\|\cdot\|_{\mathbb{R}^m} = \sqrt{\langle \cdot, \cdot \rangle_{\mathbb{R}^m}}$ denotes the canonical norm in $\mathbb{R}^m$ associated with the inner product $\langle \cdot, \cdot \rangle_{\mathbb{R}^m}$, and $u_{G-ROM}^{(i)}$ is the G-ROM approximation of the FOM displacement snapshot corresponding to the i -th parameter set and determined from Eqs. (6) using $\zeta^{(i)}$ obtained by solving Eqs. (8).

To evaluate the computational cost of the G-ROM, we record the time necessary to run computations on a CPU for both training the G-ROM and approximating the snapshots with the trained G-ROM. For consistency, all the computations are carried out on the same machine used to produce the FE simulations: a 24-core Intel® Xeon® Gold6248R CPU @ 3.00 GHz with 191 GB of usable RAM.

Remark 2.2. In our numerical investigation, we considered only the reconstructive regime for the G-ROM. We note, however, that one could employ our linearized G-ROM approach in the predictive regime via POD interpolation [14,15,36]. This operates under the assumption that the unknown column vectors, $\zeta^{(i)}$, are smooth functions of the load vectors $f^{(i)}$ within a given set of fiber orientations and given luminal pressure. Thus, for a given set of fiber orientations and luminal pressure, one can fit the known POD coefficients and projections of the load vectors onto the POD basis, $\Psi^T f^{(i)}$, using, for example, cubic spline interpolation. The cubic spline polynomial can then be evaluated to solve for coefficients, ζ , corresponding to unseen load vectors $\tilde{f}$ which are determined, for example, via linear interpolation to unseen values of the associated set of fiber orientation and luminal pressure.

2.3. Simplified machine learning (ML) model

This section outlines our implementation of ML models for approximation of the snapshots U . For this, let $x^{(i)}$ be the vector of variable

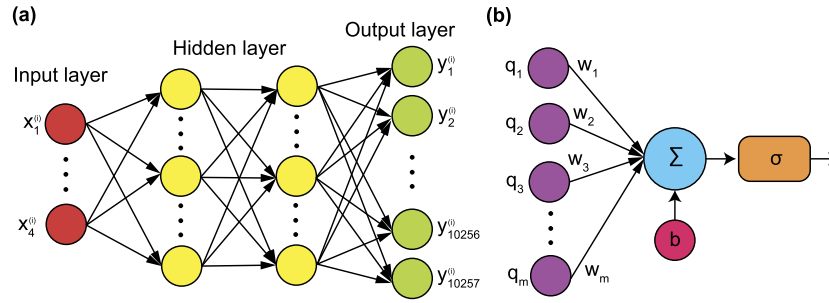


Fig. 3. (a) A flowchart depicting the structure of a typical artificial neural network (NN) with an input layer corresponding to constitutive model parameters and luminal pressure, two hidden layers, and an output layer corresponding to an approximation of a FOM snapshot. (b) A diagram of Eq. (12) which describes how inputs in each layer are used to compute an output.

inputs that contains the values of the mean preferred fiber directions in the three anatomical regions of the vagina ($\mu_1, \mu_2, \dots$, or μ_8) and the value of the luminal pressure (i.e., $p_1 = 2.5$ kPa, $\dots$, or $p_{30} = 75$ kPa) corresponding to the snapshot $\mathbf{u}_{FOM}^{(i)}$, where $i = 1, \dots, n$. These vectors are defined as follows:

$$\begin{aligned} \mathbf{x}^{(1)} &= (\mu_1, p_1) \\ \mathbf{x}^{(2)} &= (\mu_1, p_2) \\ &\vdots \\ \mathbf{x}^{(240)} &= (\mu_8, p_{30}). \end{aligned} \quad (11)$$

Since each set $\mu_1, \dots, \mu_8$ contains three mean preferred fiber orientation parameters (β_d, β_m , and β_p), $\mathbf{x}^{(i)}$ is a 4×1 column vector.

The artificial neural network (NN) that was used is schematically represented with two hidden layers in Fig. 3(a). Since the goal of this work is to compare the performance of simple G-ROM models with simple ML models, for simplicity we choose a dense (fully-connected) network, wherein the values from each node in a given layer affect each node in the next layer. To be more precise, at each node of the current layer of the network, the m components q_j of the vector $\mathbf{q}$ of node values from the previous layer (e.g., the 4 components of the vector $\mathbf{x}^{(i)}$ for the first hidden layer) are combined with a set of coefficients, or weights w_j , that serve to either amplify or dampen the input components. These input-weight products are then summed and that sum, with some bias b , is passed through the (typically nonlinear) activation function, σ , as shown in Fig. 3(b). The output of each node is given by

$$\sigma \left(\sum_{j=1}^m w_j q_j + b \right). \quad (12)$$

The outputs of each layer are then passed through the next layer of nodes until the output (last) layer is reached. The final layer of the NN yields the vector of model outputs, $\mathbf{y}^{(i)} \approx \mathbf{u}_{FOM}^{(i)}$. For our case, the output is an $m \times 1$ column vector that represents the ML approximation of a snapshot, $\mathbf{u}_{ML}^{(i)}$, corresponding to the set of input parameters $\mathbf{x}^{(i)}$.

To compare our ML model to the G-ROM, the performance of numerous sizes of NN architectures were assessed in terms of relative error between ML approximations and the FOM solution as well as on-line CPU time taken to generate ML approximations. Two architecture parameters, often called “hyperparameters” to differentiate from the weight and bias parameters that make up the NN, were varied. The first NN architecture parameter was the so-called hidden layer size, LS , which represents the number of nodes per hidden layer. A set of LS values starting from 8 and increasing by multiples of 4 up to 2048 was used, yielding a total of five values: $LS = 8, 32, 128, 512, 2048$. The second hyperparameter, the “depth” of the NN, ND , which represents the total number of hidden layers was fixed at $ND = 2$ during the LS sweep. The results of the LS sweep indicated diminishing returns in relative error performance for architectures with $LS > 128$. Thus, a second sweep was performed for a set of alternative depths of $ND = 3, 4, 5$, with LS fixed at 128.

The epochs represent the number of times that the model inputs are fed forward through the NN model and back propagated, adjusting the weights and biases via the adaptive moment estimation optimizer [37] to minimize a loss function. For all of the models, the loss function, J , was selected to be the mean squared error, that is

$$J(\mathbf{w}, \mathbf{b}) = \frac{1}{n_{\text{train}}} \sum_{i=1}^{n_{\text{train}}} \left\| \mathbf{u}_{FOM}^{(i)} - \mathbf{y}^{(i)}(\mathbf{w}, \mathbf{b}) \right\|_{\mathbb{R}^m}^2 \quad (13)$$

where $\mathbf{w}$ and $\mathbf{b}$ are vectors of the weights and biases, respectively, for all nodes at each layer of the network, and n_{train} is the number of snapshots used for training. Since the data size was relatively small, we used the full training dataset at each iteration, so that the number of epochs was equal to the number of optimization steps.

To determine the appropriate number of epochs for training, tests were performed using callbacks to stop training when convergence was observed in the loss of each ML model. Convergence was considered achieved when the minimum recorded loss value had not decreased for more than 25 epochs. We found that when this criterion was met, diminishing returns on loss improvement with additional epochs were consistently observable in fewer than 1024 epochs even for our largest NN with the most trainable parameters ($LS = 2048$, $ND = 2$). Thus, all models across both hyperparameter sweeps were trained for 1024 epochs.

All models were implemented in TensorFlow [38] with 10-fold cross validation; wherein the data are sliced into 10 different “folds” and 10 separate models are trained, each with a different fold extracted from the training data to serve as a validation dataset (so here $n_{\text{train}} = 0.9 \times 240 = 216$). Cross-validation is a common technique for ensuring that ML models are not overfit.

As with the G-ROM, we evaluate ML performance for each trained fold model by defining the following error, $\mathcal{E}_{ML}$:

$$\mathcal{E}_{ML} = \sqrt{\sum_{i=1}^n \mathcal{E}_{ML}^{(i)}} \quad (14)$$

with

$$\mathcal{E}_{ML}^{(i)} = \frac{\left\| \mathbf{u}_{ML}^{(i)} - \mathbf{u}_{FOM}^{(i)} \right\|_{\mathbb{R}^m}^2}{\left\| \mathbf{u}_{FOM}^{(i)} \right\|_{\mathbb{R}^m}^2}. \quad (15)$$

2.4. Simplified proper orthogonal decomposition machine learning (POD-ML) model

Our ML model maps a very low-dimensional input parameter space to a very high-dimensional output space, presenting a challenging structure for surrogate modeling, especially given our small training dataset. This difficulty can potentially be mitigated by choosing an alternative mapping that reduces the size of the output space of our ML models. To that end, we combined aspects of our two techniques to create a mapping from our parameter set (i.e., the set of mean preferred fiber

directions and luminal pressure) to the POD coefficients $\zeta^{(i)}$ of corresponding snapshots via methods similar to those used by Hesthaven et al. [28]. Our existing NN architecture was altered such that the final layer yields an l -dimensional vector of POD coefficients, $\mathbf{y}^{(i)} \approx \zeta_{FOM}^{(i)}$, where $\zeta_{FOM}^{(i)}$ represents the ideal POD coefficients (i.e., the FOM data projected onto the POD basis) rather than the POD coefficients obtained by running the G-ROM. Consequently, the loss function which was minimized to train our POD-ML models took the form:

$$J(\mathbf{w}, \mathbf{b}) = \frac{1}{n_{\text{train}}} \sum_{i=1}^{n_{\text{train}}} \left\| \zeta_{FOM}^{(i)} - \mathbf{y}^{(i)}(\mathbf{w}, \mathbf{b}) \right\|_{\mathbb{R}^m}^2. \quad (16)$$

To determine our training data, $\zeta_{FOM}^{(i)}$, for POD-ML, we performed a singular value decomposition (SVD) on the FOM displacement field data (i.e., the $m \times n$ matrix $\mathbf{U}$ with $m > n$) yielding

$$\mathbf{U} = \mathbf{\Psi} \mathbf{\Sigma} \mathbf{V}^T, \quad (17)$$

where $\mathbf{\Sigma}$ is the $n \times n$ diagonal matrix containing the singular values of $\mathbf{U}$ and $\mathbf{V}^T$ is the $n \times n$ matrix containing the right singular vectors of $\mathbf{U}$. We note that a simple matrix calculation [11] shows that the $m \times n$ matrix of left singular vectors of $\mathbf{U}$ in (17) contains the matrix of POD basis functions of $\mathbf{U}$ found in (5). Thus, for consistency, we denote the matrix of left singular vectors of $\mathbf{U}$ with $\mathbf{\Psi}$.

Next, we note that, for any POD basis of size l with $l \leq n$, the training data (i.e., the ideal POD coefficients), $\zeta_{FOM}^{(i)}$, are defined as the projection of the FOM data, $\mathbf{u}_{FOM}^{(i)}$, onto the POD basis:

$$\zeta_{FOM}^{(i)} = \mathbf{\Psi}_l^T \mathbf{u}_{FOM}^{(i)}, \quad i = 1, \dots, n, \quad (18)$$

where $\mathbf{\Psi}_l$ is the $l \times m$ matrix containing the first l columns of $\mathbf{\Psi}$. Finally, since the columns of $\mathbf{\Psi}$ are orthonormal, the SVD (17) yields the following formula for the training data:

$$\zeta_{FOM}^{(i)} = \mathbf{\Sigma}_l \mathbf{v}^{(i)T} \quad i = 1, \dots, n, \quad (19)$$

where $\mathbf{\Sigma}_l$ is the $l \times l$ matrix of $\mathbf{\Sigma}$ containing the first l singular values of $\mathbf{U}$, and $\mathbf{v}^{(i)T}$ are the $l \times 1$ right singular vectors of $\mathbf{U}$ contained in $\mathbf{V}^T$.

The NN architecture for POD-ML had a network depth of $ND = 2$ with $LS = 128$ nodes per hidden layer. The dimension of the output layer varied across trials from $l = 1, \dots, 100$ corresponding to the size of the POD basis and associated POD coefficients for which the NN was trained. Thus, an aptly modified version of Fig. 3(a) would show nodes for $y_1^{(i)}, \dots, y_l^{(i)}$. All other aspects of the NN training process remained the same for the POD-ML as for the ML, such as use of a sigmoid activation function, 1024 training epochs, use of the adaptive moment estimation optimizer, and 10-fold cross-validation.

The relative error of our prediction with respect to the FOM was determined by using values of the POD coefficients predicted by our NN, $\mathbf{y}^{(i)} = \zeta_{ML}^{(i)}$, in conjunction with the known POD basis, $\mathbf{\Psi}$, to produce approximations of the FOM displacement field. Thus, the performance for each trained cross-validation fold model, $\mathcal{E}_{POD-ML}$, was defined as follows:

$$\mathcal{E}_{POD-ML} = \sqrt{\sum_{i=1}^n \mathcal{E}_{POD-ML}^{(i)}} \quad (20)$$

with

$$\mathcal{E}_{POD-ML}^{(i)} = \frac{\left\| \mathbf{\Psi} \zeta_{ML}^{(i)} - \mathbf{u}_{FOM}^{(i)} \right\|_{\mathbb{R}^m}^2}{\left\| \mathbf{u}_{FOM}^{(i)} \right\|_{\mathbb{R}^m}^2}. \quad (21)$$

To ensure fairness of comparison between methods, all computations for training and approximation with the various ML models are carried out on the same machine used for both the FE simulations and the G-ROM: a 24-core Intel® Xeon® Gold6248R CPU @ 3.00 GHz with 191 GB of usable RAM. As with the G-ROM, we evaluate the computational cost of the ML models by recording both the time required to

run computations for training and the time required to approximate the FOM snapshots with a trained model.

3. Results

In this section, we present the results for our numerical investigation. Specifically, we present the FOM results (Section 3.1), which are then used as snapshot data in the G-ROM and ML numerical comparison (Section 3.2).

3.1. FOM

Increase in luminal pressure. The FE simulations of the torn vagina capture a variety of deformations in response to the applied luminal pressure. These deformations changed with increasing luminal pressure, from $p_1 = 2.5$ kPa to $p_{30} = 75$ kPa, and differences in mean preferred fiber directions, as defined by β_p , β_m , and β_d in the proximal, mid, and distal vagina, respectively. The magnitude of the displacement field increased with increasing luminal pressure for every combination of fiber orientations considered, away from the fixed end of the vagina (Fig. 4). The mid region deformed significantly more since most of the tear spanned this region. As pressure increased, the tear predominantly widened in the hoop direction taking a more circular and less elliptical shape.

Variations in fiber organization. Localized variations in the magnitude of the displacement field of FOM solutions obtained with different combinations of fiber orientations at the same luminal pressure were also observed (Fig. 5). Changes in mean preferred fiber orientation from $\beta_d = 25^\circ$ to $\beta_d = 35^\circ$ in the distal region, from $\beta_m = 25^\circ$ to $\beta_m = 35^\circ$ in the mid region, and from $\beta_p = 55^\circ$ to $\beta_p = 65^\circ$ effectively made the mechanical response of these regions more compliant in the hoop direction, resulting in a deformed vagina with a more prolate-like shape.

The model depicted in Fig. 5(b) differed from the model of Fig. 5(a) only due to the mean preferred fiber direction of $\beta_m = 25^\circ$. This decrease in β_m resulted in a narrower shape of the deformed organ through the mid region. The magnitude of the displacement field in the mid region did not change but it was greater in the distal region owing to increased axial displacement as one can appreciate when comparing Fig. 5(a) and Fig. 5(b). Interestingly, when the mean preferred fiber direction was closer to the hoop direction in the mid region of the model, effectively making the vaginal tissue stiffer in the hoop direction through that region, a wider opening of the tear was observed (Fig. 5(a)-(b)).

Both the distal and proximal mean preferred fiber orientations of the model in Fig. 5(c) differed from those of Fig. 5(a) with $\beta_d = 25^\circ$ and $\beta_p = 65^\circ$. These differences resulted in a relative decrease in the magnitude of the displacement field in the distal region and a relative increase in the magnitude of the displacement field in the proximal region for the model of Fig. 5(c) compared to the model in Fig. 5(a) at the same luminal pressure. Consequently, the model in Fig. 5(c) is more prolate-like on the distal end and comparatively spherical on the proximal end. The tear behaved similarly between the models depicted in 5(c) and 5(a). Lastly, Fig. 5(d) depicts a FOM solution with mean preferred fiber directions that are entirely different from the solution in Fig. 5(a), with $\beta_d = \beta_m = 25^\circ$ and $\beta_p = 65^\circ$. The model in Fig. 5(d) had relatively less deformation in the hoop direction in the mid and distal regions compared to the model in Fig. 5(a). However, the displacement field of the distal region for the model in Fig. 5(d) had a greater magnitude compared to the model in Fig. 5(a), owing to greater axial displacement. The proximal region of the model in Fig. 5(d) showed greater deformation in the hoop direction resulting in a more spherical shape than the proximal region of the model in Fig. 5(a). The increased stiffness in the hoop direction due to a decrease in β_m in the mid region of the model in Fig. 5(d) appeared to create a wider opening of the tear compared to the model in Fig. 5(a).

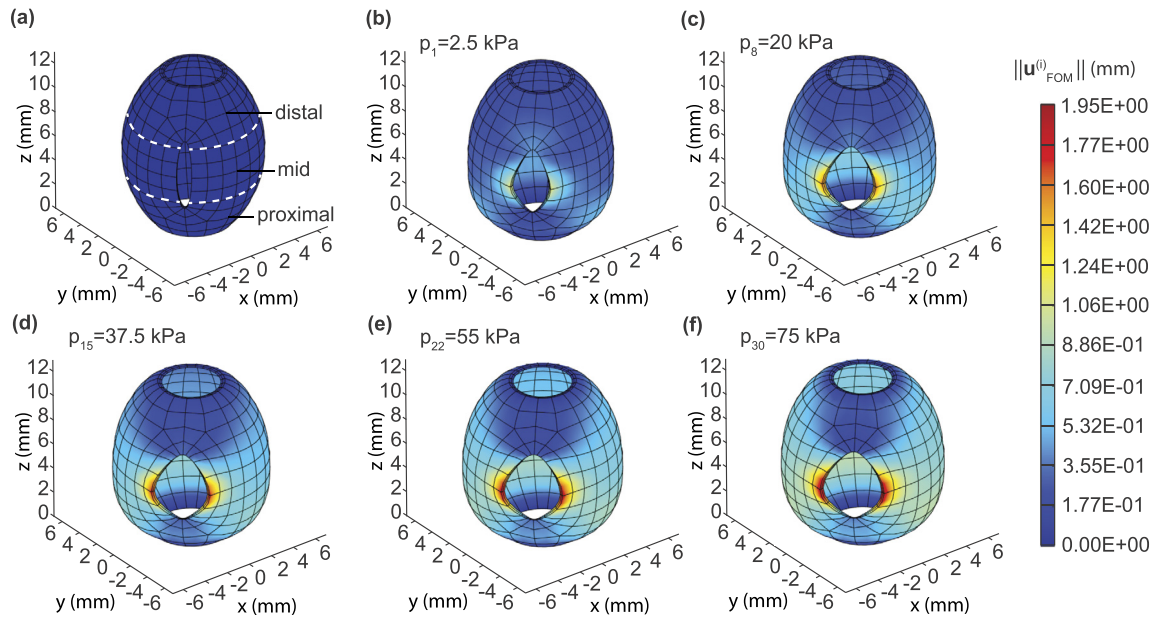


Fig. 4. Nodal displacement magnitude of the FE model of torn vagina for (a) undeformed mesh and (b)–(f) deformed meshes for various values of the applied luminal pressure p_i , with $\beta_d = \beta_m = 35^\circ$ and $\beta_p = 55^\circ$. The upper, middle, and lower thirds of each FE model correspond to the distal, mid, and proximal regions of the vagina, respectively. Note that $\|\cdot\|$ denotes the Euclidean norm in $\mathbb{R}^3$.

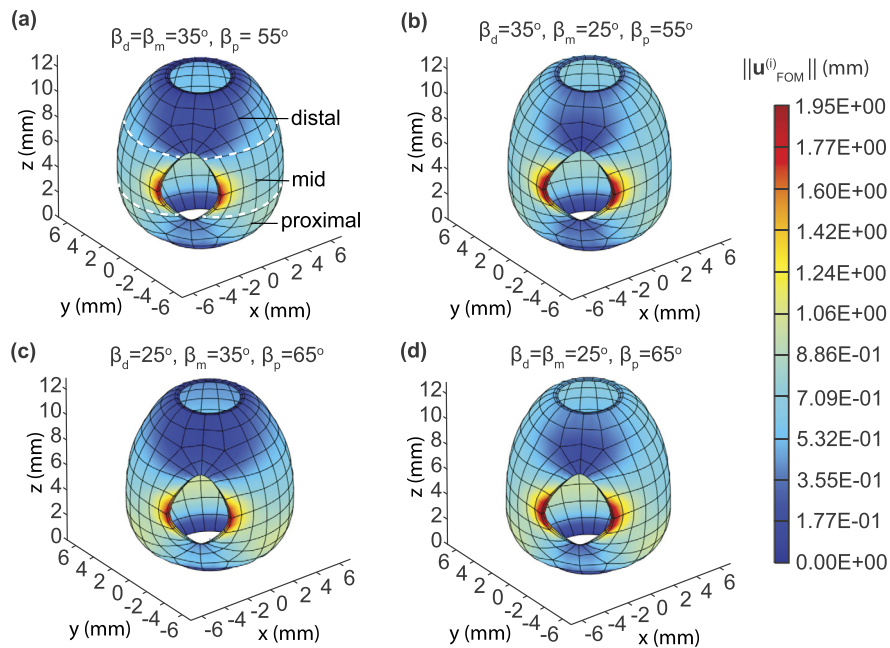


Fig. 5. (a)–(b)–(c)–(d) Nodal displacement magnitude of the FE model of torn vagina for various values of the mean preferred fiber directions, β_d , β_m , and β_p , at an applied luminal pressure $p_{22} = 55$ kPa. The upper, middle, and lower thirds of each FE model correspond to the distal, mid, and proximal anatomical regions of the vagina, respectively. Note that $\|\cdot\|$ denotes the Euclidean norm in $\mathbb{R}^3$.

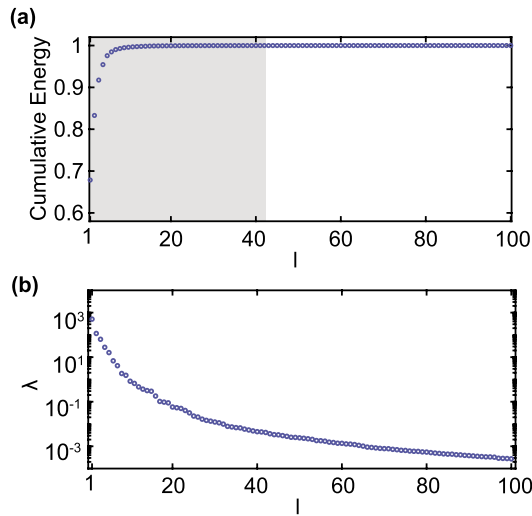


Fig. 6. (a) Cumulative energy of the eigenvalues $\lambda_1, \dots, \lambda_{100}$ corresponding to the POD basis functions $\psi_1, \dots, \psi_{100}$, respectively. The gray shading marks the basis functions that lead to an RIC index of 0.9999. (b) Decay of the eigenvalues $\lambda_1, \dots, \lambda_{100}$ corresponding to the POD basis functions $\psi_1, \dots, \psi_{100}$, respectively.

3.2. Simplified G-ROM and ML

G-ROM basis construction. There is no universally accepted method to select the number l of POD basis functions for the G-ROM, but the cumulative energy of the eigenvalues, $\mathcal{E} = (\sum_{i=1}^l \lambda_i) / (\sum_{i=1}^d \lambda_i)$, which is also known as the relative information content (RIC) index [39], is often used. The cumulative energy of the eigenvalues exceeded a tolerance of 0.9999 at $l = 42$, as shown in Fig. 6(a). The decay of the eigenvalues $\lambda_1, \dots, \lambda_{100}$ corresponding to the POD basis functions $\psi_1, \dots, \psi_{100}$ from Eqs. (5) did not show the clear plateau that is typically associated with exhaustion of all viable basis functions (Fig. 6(b)). Thus, we concluded that the quality of the G-ROM was dependent on lower energy POD basis functions. This can be seen in Fig. 6(b), where the eigenvalues consistently decayed from values of $O(10^3)$ to $O(10^{-3})$ for $l = 1, \dots, 100$.

Accuracy. The relative error $\mathcal{E}_{G-ROM}$ of the G-ROM approximations with respect to the FOM solutions averaged over all degrees of freedom and all snapshots, shown in Fig. 7, ranged from $O(10^0)$ to $O(10^{-5})$. The error exhibited a typical pattern for G-ROMs, improving rapidly at first and then showing diminishing returns with increasing computational cost as the size of the POD basis increased from $l = 1$ to $l = 100$. Shown for comparison, are the relative errors $\mathcal{E}_{ML}$ of the ML approximations generated using various NN architectures with increasing computational cost, averaged across the 10-fold cross validation models (Fig. 7(a)). For model architectures where $ND = 2$, the error decreased by an order of magnitude from $O(1)$ to $O(10^{-1})$ when increasing from $LS = 8$ to $LS = 128$. However, further increases in LS up to $LS = 2048$ yielded no further error improvements, and even slightly degraded the error performance compared to $LS = 128$. These diminishing returns on error for $LS > 128$ with dramatic increases in online computational cost led us to use $LS = 128$ for our investigation of varying ND . Increasing to $ND = 3$, $ND = 4$, and $ND = 5$ did not yield substantial improvements in error performance over $LS = 128$ and $ND = 2$, with small increases in online computational cost. Also shown are the relative errors, $\mathcal{E}_{POD-ML}$, of the POD-ML models approximating POD coefficients for POD bases of size $l = 1, \dots, 100$. The relative error of the POD-ML approximations decreased with increasing l , but the errors still remained in the same range of magnitudes as did the errors of the ML, $O(1)$ to $O(10^{-1})$. Error improvements for the POD-ML diminished around $l = 12$. The primary advantage of the POD-ML was its generally lower computational cost.

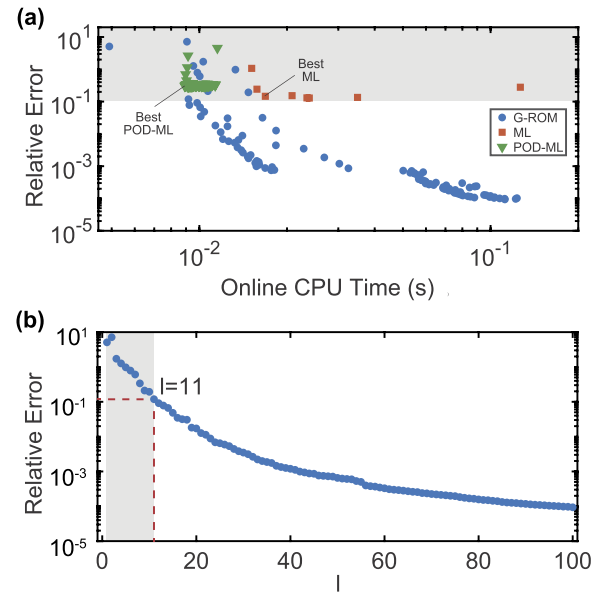


Fig. 7. (a) Pareto plot: Relative errors (Eqs. (9), (14), and (20)) versus the online (CPU) time for G-ROM with POD bases of size l ranging from $l = 1$ to $l = 100$, ML models with various combinations of hyperparameters LS and ND , and the POD-ML models for POD coefficients with l ranging from $l = 1$ to $l = 100$. The gray shading marks the lower bound of the relative error for the ML models. (b) Relative error of the G-ROM versus the number l of the POD basis functions. The gray shading marks the number ($l = 11$) of the POD basis functions with a relative error that is comparable to the ML model with $LS = 128$ and $ND = 2$.

Comparison between G-ROM and ML. The average error of the ML fold models with hyperparameters $LS = 128$ and $ND = 2$ was found to be numerically comparable to the error of the G-ROM with a POD basis of size $l = 11$ (Fig. 7). For this reason, these two models, as well as the POD-ML model corresponding to $l = 11$, were chosen for a more granular comparison of G-ROM and ML error performance at the levels of individual snapshots and mesh nodes. Fig. 8 shows the analysis of the accuracy with which the G-ROM with $l = 11$, the ML with $LS = 128$ and $ND = 2$, and the POD-ML with $l = 11$ approximated individual snapshots at each luminal pressure and for each parameter set describing the fiber organization. The results indicated that all three methods had fairly consistent performance across both pressure values and mean preferred fiber orientations, for snapshots corresponding to pressures ranging from 10 kPa to 75 kPa. However, pressures of 7.5 kPa and below had worse error performance and greater error variation between fiber orientation combinations at the same pressure for all methods, with the POD-ML with $l = 11$ basis functions showing the worst performance. Errors were highest for the snapshots at the lowest pressure value of 2.5 kPa with G-ROM relative error reaching 0.049 for μ_2 , ML error reaching 0.031 for μ_8 , and POD-ML error reaching 0.129 for μ_8 . We note that the G-ROM error performance was generally worse than the ML error performance for pressures lower or equal to 7.5 kPa, but the POD-ML performed worse than either. We also note that, for pressures greater or equal to 10 kPa, the G-ROM generally performs better in terms of error than the ML or POD-ML, with the lowest values of error at 0.001 for the G-ROM, 0.004 for ML, and 0.003 for POD-ML.

Lastly, we calculated nodal errors of the G-ROM and ML approximations relative to the FOM solutions at one low pressure snapshot ($p_1 = 2.5$ kPa) and one high pressure snapshot ($p_{22} = 55$ kPa) for μ_1 , and mapped them over the surface of the deformed mesh to compare localized error performance between our techniques (Fig. 9). At the lower pressure, the G-ROM showed largest errors on the sides of the tear and in the distal region, especially close to the opening (Fig. 9(a)). Smaller errors for the same case were displayed in the proximal region close to the opening and in a band around the mid region, away from the tear.

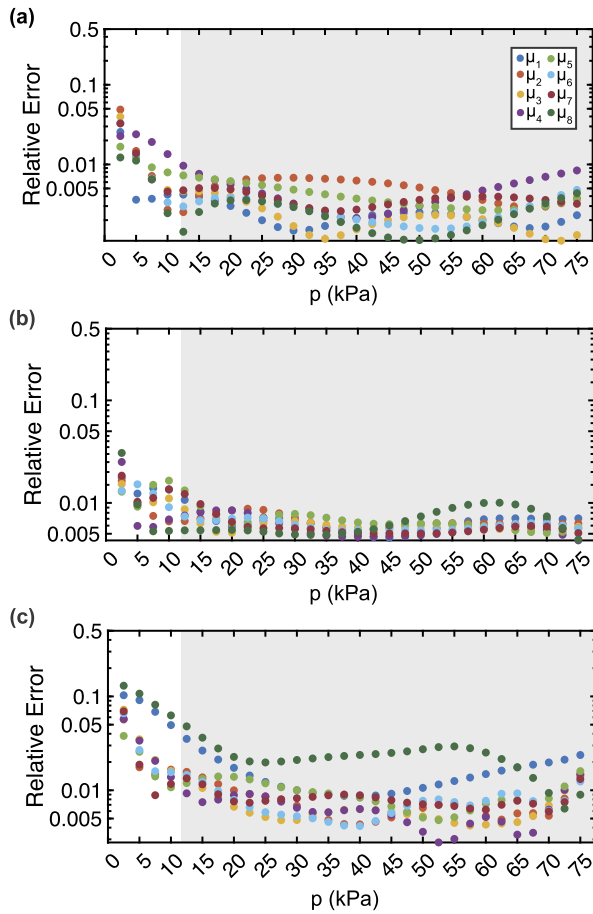


Fig. 8. Relative errors (Eqs. (10), (15), and (21)) of approximations produced for each fiber orientation parameter set (differentiated by color) at each luminal pressure $p_1, \dots, p_{30}$ by (a) the G-ROM using a POD basis of size $l = 11$, (b) the ML model with hyperparameters $LS = 128$ and $ND = 2$, and (c) the POD-ML model with $l = 11$, $LS = 128$ and $ND = 2$. The gray shading marks snapshots with lowest relative errors.

The ML approximation at lower pressure also had larger errors in the immediate region of the tear, but the error over most of the rest of the geometry was fairly uniform with patches of slightly larger and slightly smaller errors distributed randomly throughout (Fig. 9(b)). The exception was a thin band of smaller errors around the proximal opening which can be seen clearly on the interior of the geometry when viewing through the tear. While the POD-ML approximation had greater overall magnitudes of error, the patterns of error on the mesh reflected that of the G-ROM with the notable exception of the boundary region at the distal opening (Fig. 9(c)). Whereas the G-ROM had greater error on this boundary than elsewhere in the distal region, the POD-ML had smaller errors at the boundary than elsewhere. In the higher pressure case, the G-ROM once again exhibited slightly larger errors around the edges of the tear, though less pronounced than in the lower pressure case (Fig. 9(d)). The proximal region again had smaller errors than other regions, but the band around the mid region had slightly larger errors in this case. Overall, the magnitude of the error in the higher pressure G-ROM case was much smaller than in the lower pressure case. The higher pressure ML approximation also had a less pronounced increase in error around the edges of the tear when compared to its low pressure counterpart (Fig. 9(e)). Its other characteristics were very similar though, with randomly distributed patches of larger and smaller errors and a band of smaller errors around the proximal opening. As with the lower pressure case, the POD-ML approximation of the higher pressure snapshot had greater error magnitudes but similar patterning of error to the G-ROM in the mid region and the distal region away from the boundary

(Fig. 9(f)). Different from the G-ROM, the POD-ML again had smaller errors at the distal opening than elsewhere in the proximal region. In addition, the POD-ML exhibited errors in the proximal region away from the boundary which were similar to the errors of the mid region rather than smaller, as was the case for the G-ROM. The overall magnitude of errors for the G-ROM and POD-ML approximations changed quite substantially between lower and higher pressure cases, while the ML had similar overall error magnitudes in each of the pressure cases.

Computational cost. All values for computational cost assessment were produced by running the offline and online portions of the G-ROM and NN on the same 24-core Intel® Xeon® Gold 6248R CPU @ 3.00 GHz with 191 GB of usable RAM that was used to produce FOM results. No GPU hardware acceleration was implemented when generating our results (see discussion in Section 4).

The offline (training) and online (approximation) times are listed in Table 2 alongside corresponding relative errors (Eqs. (9), (14), (20)) for the least expensive G-ROM ($l = 1$), the G-ROM used for snapshot and local error comparisons with the ML model ($l = 11$), the most expensive G-ROM ($l = 100$), the least expensive ML architecture ($LS = 8$, $ND = 2$), the ML architecture used for snapshot and nodal error comparisons with the G-ROM ($LS = 128$, $ND = 2$), the most expensive ML architecture ($LS = 2048$, $ND = 2$), and the POD-ML ($l = 11$, $LS = 128$, $ND = 2$). The least expensive G-ROM had an offline time of $O(10^{-1})$ s, and the G-ROM at comparable error performance to the ML, with hyperparameters $LS = 128$ and $ND = 2$, had an offline time of $O(1)$ s. Training the most expensive G-ROM, with $l = 100$ basis functions, only took $O(10^1)$ s. The mean offline times for all the ML models from least to most expensive were of $O(10^2)$ s, an order of magnitude larger than the offline time of the most expensive G-ROM. As for online times, the G-ROM with one basis function was, predictably, the least expensive to evaluate, taking $O(10^{-3})$ s to complete. The G-ROM with a POD basis of size 11 was not much more expensive, with an online time also of $O(10^{-3})$ s. By contrast, the G-ROM with a 100-dimensional POD basis had an online time of $O(10^{-1})$ s. The online times of the ML models with $LS = 8$ and $LS = 128$ were quite similar, both taking $O(10^{-2})$ s to produce approximations. The online time of the ML model with $LS = 2048$ was more similar to the G-ROM for $l = 100$, with an online time of $O(10^{-1})$ s. Lastly, the POD-ML model with $l = 1, \dots, 100$ generally outperformed the ML model in all cases, with the most expensive approximations ($l = 100$) having online times of $O(10^{-2})$ s and completing faster than the smallest ML model ($LS = 8$, $ND = 2$). Many of the POD-ML approximations ($l = 2, \dots, 65$) had online times of $O(10^{-3})$ s and performed either comparably or superior to their G-ROM counterparts.

We note that substantially more floating point operations (FLOPS) were required for an evaluation of the ML model mapping from parameter sets to full displacement fields than were required for the G-ROM. We speculate that the similarity in evaluation times was due to parallelization and optimization built into the established ML libraries we used in our study.

4. Discussion

In this study, we constructed an FE model of the rat vagina with geometry and boundary conditions meant to recreate the *ex vivo* experiments conducted in our lab [40]. The model accounts for observed differences in the microstructure of the proximal, mid, and distal vagina [30]. A pre-imposed elliptical tear with the major axis aligned with the axial direction of the organ was included in the geometry so the model could give us insight into the impact that a large tear had on the deformation of vaginal tissue. The results of our simulations showed that the initially elliptical tear became more circular as the pressure increased (Fig. 4) and that the opening of the tear was favored when the mean preferred fiber directions were more aligned in the hoop direction (Fig. 1). Both these findings are consistent with our experimental investigation

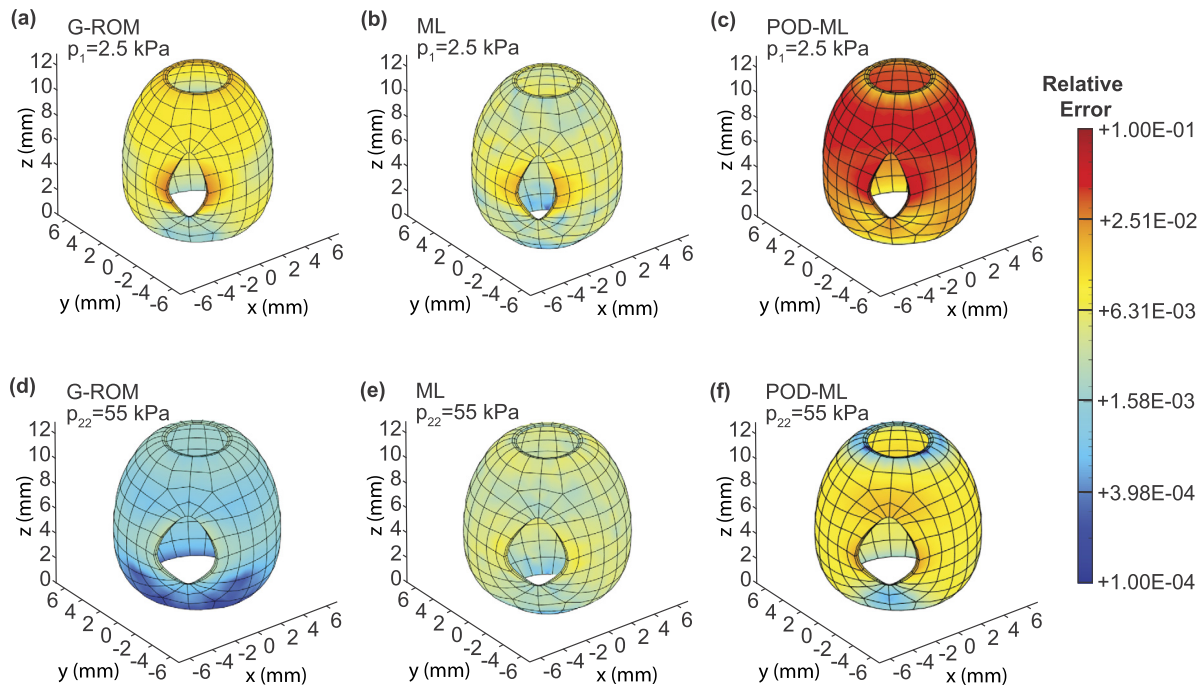


Fig. 9. Nodal relative error plotted on the surface of deformed mesh for approximations of snapshots via G-ROM with a POD basis with $l = 11$ at $p_1 = 2.5$ kPa (a) and $p_{22} = 55$ kPa (d), via ML with hyperparameters $LS = 128$ and $ND = 2$ for snapshots at $p_1 = 2.5$ kPa (b) and $p_{22} = 55$ kPa (e), and via POD-ML for snapshots at $p_1 = 2.5$ kPa (c) and $p_{22} = 55$ kPa (f), all with mean preferred fiber orientations, μ_1 ($\beta_d = \beta_m = 35^\circ$ and $\beta_p = 55^\circ$).

Table 2

Offline times, online times, and relative errors for the FOM, the G-ROM with a POD basis with $l = 1$, $l = 11$, and $l = 100$, ML models with architectures defined by hyperparameters $LS = 8$, 128, and 2048 with $ND = 2$, and the POD-ML model with $l = 11$ POD basis functions. The FOM offline time is the full cost of simulation and the FOM online time is the order of the cost of the final Newton-Raphson iterations for the 240 parameter combinations.

Model Reduction Technique	Offline Time (s)	Online Time (s)	Relative Error
FEM	3.827×10^4	$O(10)$	N/A
G-ROM: $l = 1$	0.479	0.005	5.095
G-ROM: $l = 11$	1.641	0.009	1.189×10^{-1}
G-ROM: $l = 100$	13.650	0.112	9.364×10^{-5}
ML: $LS = 8$, $ND = 2$	104.947	0.015	1.069
ML: $LS = 128$, $ND = 2$	108.181	0.017	1.461×10^{-1}
ML: $LS = 2048$, $ND = 2$	476.234	0.127	2.785×10^{-1}
POD-ML: $l = 11$	14.919	0.009	3.193×10^{-1}

exploring the toughening mechanisms of the vagina in the rat model [30].

Although both the selected geometry (i.e., the hollow prolate spheroid) and boundary conditions (e.g., uniform luminal pressure) of the FE model well replicate the experimental conditions of the *ex vivo* vagina, they are not a realistic representation of the shape and boundary conditions of the *in vivo* vagina. To create an FE model that accurately represents *in vivo* conditions, we are currently using magnetic resonance imaging (MRI) to obtain detailed images of the vagina within the entire reproductive system in rodents. These new data will be incorporated into FE models to reconstruct the geometry and reproduce both boundary and loading conditions of the reproductive system. Vaginal tears were assumed to be static elliptical holes in the geometry of our FE simulations, as done by others to study the mechanical impact of episiotomy [7]. This assumption was supported by our experimental work showing that the vagina is a very tough organ with a microstruc-

ture that prevents tear propagation. Experimentally, pre-imposed tears along the axial direction of the rat vagina were observed to propagate under inflation at a mean ($\pm$ std. dev.) pressure of 40 ± 10 kPa [30]. To improve the realism of the FE simulations, we plan to create FE models that can not only simulate the deformations of the torn vagina but also the progressive propagation of tears. One method which could allow us to simulate tear propagation in the vagina is the cohesive zone model for fracture mechanics [41]. The implementation of such model would still require that we pre-imposed the site of the tear and the direction of tear propagation. To simulate tearing behavior which is entirely controlled by the mechanics of our model, we instead would need to implement damage modeling into our material constitutive equations and delete elements from the mesh when they meet some set failure criteria [42,43].

With regard to our simplified model reduction techniques, our results appear to indicate that the G-ROM outperformed the ML in all our metrics of comparison, with the exception of the POD-ML having comparable speed to smaller-dimensional G-ROMs. The Pareto plots shown in Fig. 7(a) and both the offline and online times for the G-ROM and ML methods listed in Table 2 seem to favor the G-ROM as having superior performance. The most expensive G-ROM considered, with $l = 100$ POD basis functions, had an offline time which was an order of magnitude smaller than the least expensive ML model and an error that was 4 orders of magnitude smaller than any of the ML models. While online time for this G-ROM was one order of magnitude slower than most of the ML models and two orders of magnitude slower than most of the POD-ML models, it was comparable to the online time of the most expensive ML model. The lowest error approximation from the ML models ($LS = 128$, $ND = 5$) was outperformed by the G-ROM with $l = 11$ POD basis functions. We do, however, note that the ML approximations had slightly more competitive error performance for the lower pressure snapshots from 2.5–7.5 kPa as the G-ROM only achieved superior performance in this range with $l \geq 14$ POD basis functions.

We speculate that the superior performance of the G-ROM in this case may be due to it being inherently physics-informed as it is constructed from the linearized equations of the nonlinear elasticity problem, whereas the ML must relate model inputs and outputs without

knowledge of the physics. However, this does not mean that the G-ROM is the preferred method of order reduction for all use-cases. The linearized G-ROM presented in this study could perform better than ML only in the context where the governing equations of the problem are known and the goal is to reconstruct the existing snapshots on which the models are trained. Having an ML technique which is not dependent on known governing equations can be an advantage as it is more easily adaptable to performing approximations with, for example, experimental data sets. The ML model could perform these approximations without the need to first replicate the experimental results in FE models as would be necessary for G-ROM. In addition, the G-ROM used in this study is constructed from the linear systems produced by the final iterations of the Newton-Raphson method applied to the nonlinear system of equations of our nonlinear elasticity problem. Thus, the G-ROM we used here has difficulties being extended to the predictive regime as components of the linearized system at new parameters can only be obtained via running additional FE simulations, decreasing the improved efficiency of G-ROM. Therefore, prediction of new snapshots with our G-ROM for parameters not included in the training data would require methods such as the subspace interpolation used by Niroomandi et al. [15] and Pfaller et al. [19] or precomputation of G-ROM bases for a set of “most probable” load states, also used by Niroomandi et al. [15]. Alternatively, the difficulties in the predictive regime could be addressed by constructing a fully nonlinear G-ROM. However, that would certainly increase the overall computational cost of the G-ROM. By contrast, the ML model does not require additional steps to perform such extrapolative predictions. However, an ML model which is not physics-informed may have worse performance when making predictions outside of the training regime than would a comparable physics-informed approach. Therefore, the type of ML model used in this study may not be as reliable as a nonlinear G-ROM for making predictions, but it does have an advantage in comparative ease of implementation. We also note that the ML has a general advantage in terms of usability due to open source tool-kits like *scikit-learn* and *TensorFlow*.

The superior speed performance of the G-ROM compared to the ML could be due to the implementation of both techniques on a CPU rather than a graphics processing unit (GPU). Typically, ML models are implemented using hardware acceleration to leverage the GPU and reduce computational cost. From cursory testing with hardware acceleration we found that our ML models with LS up to 1000 had online times of $O(10^{-3})$ s. Our choice not to conduct all of *TensorFlow*'s calculations on the CPU instead of GPU had two motivations. First, we do not yet have a *TensorFlow* implementation of our G-ROM methods which can benefit from hardware acceleration, so our ML results needed to be run on a CPU for the sake of a fair comparison. Second, we found that our GPU (NVIDIA® Quadro RTX™ 5000) would exhaust its available memory when computing NN models with $LS = 2048$. Our results indicated that $LS > 128$ yielded diminished returns in performance, so we acknowledge that such large hidden layers may not be necessary for problems of our type. As this project advances, we will narrow our focus to optimize NNs with smaller LS for which hardware acceleration is viable.

Both model order reduction techniques, the G-ROM and ML methods, struggled to approximate FE simulations at lower luminal pressure. In Fig. 8, we compared approximations of individual snapshots at pressure values $p_1, \dots, p_{30}$ using the G-ROM with $l = 11$ POD basis functions, POD-ML of the same dimension, and the ML model with $LS = 128$ and $ND = 2$. All approximations exhibited worse error performance for snapshots at pressures from 2.5–7.5 kPa, but the G-ROM and POD-ML at $l = 11$ had worse error performance than the ML and more variation in error between fiber orientation combinations. This difficulty in approximating the FOM at lower pressures may be the result of the highly nonlinear anisotropic elastic behavior of vaginal tissue. According to the HGO constitutive model implemented in this study, the deformations of the vaginal tissue are determined by both

the arrangement of the fibers comprising the tissue and the surrounding isotropic ground substance. Such deformations increase exponentially with the applied pressure, changing significantly from low pressures to high pressures based on the contributions of the fibers and ground substance.

When comparing G-ROM and ML approximations at comparable levels of overall relative error, we also found that the profiles of localized errors for the two methods were quite different (Fig. 9). Figs. 9(a), 9(b), and 9(c) all showed larger values of error around the edges of the tear as expected given that the tear region experiences larger deformations and greater stress concentrations than other parts of the model. However, Figs. 9(a) and 9(c) had generally larger magnitudes of error over most of the geometry compared to Fig. 9(b). This agrees with the results for errors at different values of pressure shown in Fig. 8, wherein the G-ROM and POD-ML approximations tended to perform worse at lower pressures than the ML approximations. Notably, the G-ROM and POD-ML approximations in Fig. 9(a) and 9(c) also had clear patterns of local error in regions away from the tear. One could observe smaller errors at the proximal opening, slightly larger errors in the region of transition from the proximal to mid region, a band of smaller errors again in the mid region, and then larger errors throughout the distal region up to the distal opening. For the POD-ML, the error dropped again at the boundary on the distal opening. Comparatively, the ML approximation errors in Fig. 9(b) did not have the same sort of regional distinctions away from the tear. There were small patches of larger or smaller local error dispersed over the geometry and only a thin band of smaller errors around the proximal opening. A similar disparity was observed between Fig. 9(e) and Figs. 9(d) and 9(f). The G-ROM approximation in Fig. 9(d) had alternating bands of smaller errors in the proximal region, larger errors in the mid region, and smaller errors again in the distal region. The POD-ML approximation in Fig. 9(f) had its smallest errors at the proximal and distal openings and the error appeared to increase gradually when approaching the middle of the geometry from either end. In contrast, the ML approximation in Fig. 9(e) had patches of larger or smaller local error and a band of smaller errors around the proximal opening in similar fashion to Fig. 9(b). It seems likely that these differences in local error between models are due to the fact that the G-ROM and POD-ML relied on the POD basis, whereas the ML did not. For the POD-ML in particular, the multiplication of the approximated POD coefficients by the known POD basis may have imparted characteristics of the boundary conditions stored in the POD basis onto the resulting displacement field prediction which were not captured by the direct parameter-to-displacement mapping. This may account for the local errors of the ML approximations having a comparatively random distribution that was much less affected by regional proximity to boundaries. However, it is still not clear why the error at the distal boundary appeared to decrease for the POD-ML and increase for the G-ROM.

The “patchy” nature of the ML model error in Fig. 9 also highlights a potential path for improving the ML models by building spatial structure into the approximation. This can be done by penalizing large deviations between values at nearby nodes – rewarding smoothness – as part of the loss function. A natural first step toward encoding this locality is to consider the mass matrix from the FE construction, which characterizes connections in the mesh. Of course, once we have added the mass matrix into the ML loss function, it would be natural to try to incorporate other operators from the FE in a similar way to penalize model outputs that violate the underlying partial differential equations. This is the approach of “physics-informed” machine learning; see, e.g., [44] for background and [45] for one implementation using *TensorFlow*. The low-dimensional inputs and high-dimensional outputs also point to the potential for using data reduction techniques such as what we have implemented with POD-ML or auto-encoders [46] (see also, e.g., [47, Section 6]) to reduce the dimension of the target data before fitting with ML. Our trials with a POD-ML models yielded improvements in computational cost compared to our ML models which mapped to the full displacement field. However, POD-ML did not provide any benefits

in terms of model accuracy. We did attempt some preliminary investigation of auto-encoder methods for ML, but they did not seem to materially improve the accuracy or, due to the cost of decoding, the online time for this problem case. We plan to further develop our implementation of POD-ML and perform a more extensive exploration of auto-encoder methods to build more effective ML models going forward.

5. Conclusions

This work presented FE simulations that illustrated the deformations of the rat vagina during *ex vivo* inflation testing. Differently from our previous study [22], the FE simulations described the effect of a pre-imposed tear on the mechanical response of the vaginal tissue. These simulations, for eight different sets of material parameters and thirty values of luminal pressure, were used as snapshots to implement and compare two fundamentally different strategies: a simplified, linearized G-ROM and a straightforward ML approach with various NN architectures. In addition, we combined elements of both strategies to implement a POD-ML approach which was compared to both the G-ROM and the ML approaches.

Each technique was used to produce approximations of the snapshots, and these approximations were compared in terms of accuracy and both offline and online computational costs. It was found that, when approximating the FE-based snapshots in the reconstructive regime, G-ROM was both more accurate and less computationally expensive than the ML models. The ML model for which approximations had the lowest error was outperformed in terms of error by the G-ROM with a POD basis of size $l = 11$, and the G-ROM approximations at comparable levels of error had lower online times than those from ML models. When using a POD-ML approach, the online computational cost of the ML strategy was comparable to the G-ROM, but the POD-ML did not improve on the error performance of the ML models. The offline times for training ML models were an order of magnitude larger than the offline time needed to train even the most expensive G-ROM with a POD basis with $l = 100$, and the offline training time of the POD-ML was one order of magnitude greater than the offline time of the G-ROM for the same size of POD basis.

To the authors' knowledge, this comparison of basic numerical and data-driven model reduction techniques for approximating the deformations of anisotropic hyperelastic soft biological tissues is the first of its kind. While these findings suggest that the G-ROM is a superior order reduction method for this specific reconstructive task, there may be more complex forms of NN or other ML methods which would be more competitive than those employed here, and ML techniques could still be preferred for certain model reduction use-cases due to their ease of implementation and adaptability to the predictive regime. Just as the ML methods chosen for the G-ROM *versus* ML comparison were fairly simple, more advanced G-ROM methods (e.g., using a classic ROM discretization of the fully nonlinear equations and equipping the resulting ROM with hyperreduction [35,48,49]) could also be evaluated in future comparisons of computational cost and accuracy. Additionally, future inquiries could compare the predictive capabilities of ML models to a predictive regime technique for G-ROM such as the POD interpolation described in Remark 2.2. In general, the successful implementation of both techniques for our simplified model of *ex vivo* vaginal tissue deformations indicated potential for their application to more realistic simulations. Thus, future studies will need to investigate model reduction techniques for the approximation of FE models of the vagina with more realistic geometries and boundary conditions, as well as tear initiation and propagation, to better reproduce the *in vivo* characteristics of this important reproductive organ and accurately simulate vaginal tearing during childbirth. The results herein laid a solid foundation for further investigations of G-ROM and ML as competing options for real-time simulation of maternal birth trauma.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the National Science Foundation under grant #1929731 and #2135683.

References

- [1] E. Samuelsson, L. Ladfors, B.G. Lindblom, H. Hagberg, A prospective observational study on tears during vaginal delivery: occurrences and risk factors, *Acta Obstet. Gynecol. Scand.* 81 (1) (2002) 44–49.
- [2] C. Phillips, A. Monga, Childbirth and the pelvic floor: “the gynaecological consequences”, *Rev. Gynaecolog. Pract.* 5 (1) (2005) 15–22.
- [3] L.A. Smith, N. Price, V. Simonite, E.E. Burns, Incidence of and risk factors for perineal trauma: a prospective observational study, *BMC Pregnancy Childbirth* 13 (1) (2013) 1–9.
- [4] S. Chen, M.J. Grimm, Childbirth computational models: characteristics and applications, *J. Biomech. Eng.* 143 (5) (2021).
- [5] D.A. Oliveira, M.P.L. Parente, B. Calvo, T. Mascarenhas, R.M. Natal Jorge, A biomechanical analysis on the impact of episiotomy during childbirth, *Biomech. Model. Mechanobiol.* 15 (6) (2016) 1523–1534.
- [6] D.A. Oliveira, M.P.L. Parente, B. Calvo, T. Mascarenhas, R.M. Natal Jorge, A holistic view of the effects of episiotomy on pelvic floor, *Int. J. Numer. Methods Biomed. Eng.* 33 (12) (2017) e2892.
- [7] D. Oliveira, M.V. Pouca, J. Ferreira, T. Mascarenhas, Episiotomy: the biomechanical impact of multiple small incisions during a normal vaginal delivery, *Interface Focus* 9 (5) (2019) 20190027.
- [8] J.S. Hesthaven, G. Rozza, B. Stamm, *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*, Springer, 2015.
- [9] P. Holmes, J.L. Lumley, G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge, 1996.
- [10] A. Quarteroni, A. Manzoni, F. Negri, *Reduced Basis Methods for Partial Differential Equations: An Introduction*, vol. 92, Springer, 2015.
- [11] S. Volkwein, *Proper Orthogonal Decomposition: Theory and Reduced-Order Modelling*, Lecture Notes, University of Konstanz, 2013, <http://www.math.uni-konstanz.de/numerik/personen/volkwein/teaching/POD-Book.pdf>.
- [12] E. Cueto, F. Chinesta, Real time simulation for computational surgery: a review, *Adv. Model. Simul. Eng. Sci.* 1 (1) (2014) 1–18.
- [13] F. Dogan, M.S. Celebi, Real-time deformation simulation of non-linear viscoelastic soft tissues, *Simulation* 87 (3) (2011) 179–187.
- [14] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Real-time deformable models of non-linear tissues by model reduction techniques, *Comput. Methods Programs Biomed.* 91 (3) (2008) 223–231.
- [15] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Accounting for large deformations in real-time simulations of soft tissues based on reduced-order models, *Comput. Methods Programs Biomed.* 105 (1) (2012) 1–12.
- [16] S. Niroomandi, D. González, I. Alfaro, F. Bordeu, A. Leygue, E. Cueto, F. Chinesta, Real-time simulation of biological soft tissues: a PGD approach, *Int. J. Numer. Methods Biomed. Eng.* 29 (5) (2013) 586–600.
- [17] D. Bonomi, A. Manzoni, A. Quarteroni, A matrix discrete empirical interpolation method for the efficient model reduction of parametrized nonlinear PDEs: application to nonlinear elasticity problems, in: *MATHICSE, Ecole Polytechn. Fédérale de Lausanne, Lausanne, Switzerland, Tech. Rep. 14* (2016) 2016.
- [18] D. Bonomi, A. Manzoni, A. Quarteroni, A matrix DEIM technique for model reduction of nonlinear parametrized problems in cardiac mechanics, *Comput. Methods Appl. Mech. Eng.* 324 (2017) 300–326.
- [19] M.R. Pfäler, M. Cruz Varona, J. Lang, C. Bertoglio, W.A. Wall, Using parametric model order reduction for inverse analysis of large nonlinear cardiac simulations, *Int. J. Numer. Methods Biomed. Eng.* 36 (4) (2020) e3320.
- [20] A. Radermacher, S. Reese, Proper orthogonal decomposition-based model reduction for non-linear biomechanical analysis, *Int. J. Mater. Eng. Innov.* 4 (2) (2013) 149–165.
- [21] A. Radermacher, S. Reese, A comparison of projection-based model reduction concepts in the context of nonlinear biomechanics, *Arch. Appl. Mech.* 83 (8) (2013) 1193–1213.
- [22] W. Snyder, J.A. McGuire, C. Mou, D.A. Dillard, T. Iliescu, R. De Vita, Data-driven variational multiscale reduced order modeling of vaginal tissue inflation, *Int. J. Numer. Methods Biomed. Eng.* (2022) e3660.
- [23] F. Martínez-Martínez, M.J. Rupérez-Moreno, M. Martínez-Sober, J.A. Solves-Llorens, D. Lorente, A.J. Serrano-López, S. Martínez-Sanchis, C. Monserrat, J.D. Martín-Guerrero, A finite element-based machine learning approach for modeling the mechanical behavior of the breast tissues under compression in real-time, *Comput. Biol. Med.* 90 (2017) 116–124.

- [24] D. Lorente, F. Martínez-Martínez, M.J. Rupérez, M.A. Lago, M. Martínez-Sober, P. Escandell-Montero, J.M. Martínez-Martínez, S. Martínez-Sanchis, A.J. Serrano-López, C. Monserrat, J.D. Martín-Guerrero, A framework for modelling the biomechanical behaviour of the human liver during breathing in real time using machine learning, *Expert Syst. Appl.* 71 (2017) 342–357.
- [25] M. Tonutti, G. Gras, G.-Z. Yang, A machine learning approach for real-time modelling of deformation in image-guided neurosurgery, *Artif. Intell. Med.* 80 (2017) 39–47.
- [26] A. Madani, A. Bakhty, J. Kim, Y. Mubarak, M.R.K. Mofrad, Bridging finite element and machine learning modeling: stress prediction of arterial walls in atherosclerosis, *J. Biomech. Eng.* 141 (8) (2019).
- [27] S. Fresca, A. Manzoni, L. Dedé, A. Quarteroni, POD-enhanced deep learning-based reduced order models for the real-time simulation of cardiac electrophysiology in the left atrium, *Front. Physiol.* 12 (2021) 679076.
- [28] J.S. Hesthaven, S. Ubbiali, Non-intrusive reduced order modeling of nonlinear problems using neural networks, *J. Comput. Phys.* 363 (2018) 55–78.
- [29] T.C. Gasser, R.W. Ogden, G.A. Holzapfel, Hyperelastic modelling of arterial layers with distributed collagen fibre orientations, *J. R. Soc. Interface* 3 (6) (2006) 15–35.
- [30] J.A. McGuire, J.L. Monclova, A.C.S. Coariti, C.A. Stine, K.C. Toussaint Jr., J.M. Munson, D.A. Dillard, R. De Vita, Tear propagation in vaginal tissue under inflation, *Acta Biomater.* 127 (2021) 193–204.
- [31] N.-H. Kim, *Introduction to Nonlinear Finite Element Analysis*, Springer Science & Business Media, 2014.
- [32] L. Lapidus, G.F. Pinder, *Numerical Solution of Partial Differential Equations in Science and Engineering*, John Wiley & Sons, 1982.
- [33] L. Sirovich, Turbulence and the dynamics of coherent structures. I. Coherent structures, *Q. Appl. Math.* 45 (3) (1987) 561–571.
- [34] S. Niroomandi, I. Alfaro, D. Gonzalez, E. Cueto, F. Chinesta, Real-time simulation of surgery by reduced-order modeling and X-FEM techniques, *Int. J. Numer. Methods Biomed. Eng.* 28 (5) (2012) 574–588.
- [35] A. Radermacher, S. Reese, POD-based model reduction with empirical interpolation applied to nonlinear elasticity, *Int. J. Numer. Methods Eng.* 107 (6) (2016) 477–495.
- [36] H.V. Ly, H.T. Tran, Modeling and control of physical processes using proper orthogonal decomposition, *Math. Comput. Model.* 33 (2001) 223–236.
- [37] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, *arXiv:1412.6980*, 2017.
- [38] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: large-scale machine learning on heterogeneous systems, software available from tensorflow.org, <https://www.tensorflow.org/>, 2015.
- [39] S.E. Ahmed, S. Pawar, O. San, A. Rasheed, T. Iliescu, B.R. Noack, On closures for reduced order models – a spectrum of first-principle to machine-learned avenues, *Phys. Fluids* 33 (9) (2021) 091301.
- [40] J.A. McGuire, C.L. Crandall, S.D. Abramowitch, R. De Vita, Inflation and rupture of vaginal tissue, *Interface Focus* 9 (4) (2019) 20190029.
- [41] M. Elices, G. Guinea, J. Gómez, J. Planas, The cohesive zone model: advantages, limitations and challenges, *Eng. Fract. Mech.* 69 (2) (2002) 137–163.
- [42] J.C. Simo, J.W. Ju, Strain- and stress-based continuum damage models–I. Formulation, *Int. J. Solids Struct.* 23 (7) (1987) 821–840.
- [43] P. Liu, J. Zheng, Recent developments on damage modeling and finite element analysis for composite laminates: a review, *Mater. Des.* 31 (8) (2010) 3825–3834.
- [44] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707.
- [45] E. Haghighat, R. Juanes, SciANN: a Keras/TensorFlow wrapper for scientific computations and physics-informed deep learning using artificial neural networks, *Comput. Methods Appl. Mech. Eng.* 373 (2021) 113552.
- [46] J. Tencer, K. Potter, A tailored convolutional neural network for nonlinear manifold learning of computational physics data using unstructured spatial discretizations, *SIAM J. Sci. Comput.* 43 (4) (2021) A2581–A2613.
- [47] M.Z. Alom, T.M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M.S. Nasrin, M. Hasan, B.C. Van Essen, A.A.S. Awwal, V.K. Asari, A state-of-the-art survey on deep learning theory and architectures, *Electronics* 8 (3) (2019) 292.
- [48] C. Farhat, T. Chapman, P. Avery, Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models, *Int. J. Numer. Methods Eng.* 102 (2015) 1077–1110.
- [49] C. Farhat, S. Grimberg, A. Manzoni, A. Quarteroni, Computational bottlenecks for PROMs: precomputation and hyperreduction, in: *Model Order Reduction Volume 2: Snapshot-Based Methods and Algorithms*, 2020, pp. 181–244.