Real-time biomechanics using the finite element method and machine learning: review and perspective

Renzo Phellan,[1, a] Bahe Hachem,[2] Julien Clin,[2] Jean–Marc Mac–Thiong,[2] and Luc Duong[1]

[1] *ETS Montreal - University of Quebec - 3330 Hospital Drive NW - Montreal - QC - Canada*

[2] *Spinologics Inc. - 6750 Esplanade Avenue #290 - Montreal - QC - Canada*

(Dated: November 2020; Revised November 2020)

This is the peer reviewed version of the following article: Phellan, R., Hachem, B., Clin, J., Mac-Thiong, J.-M. and Duong, L. (2021), Real-time biomechanics using the finite element method and machine learning: Review and perspective. Med. Phys., 48: 7-18, which has been published in final form at https://doi.org/10.1002/mp.14602.

Running title: Real-time biomechanics review

Purpose: The finite element method (FEM) is the preferred method to simulate phenomena in anatomical structures. However, purely FEM-based mechanical simulations require considerable time, limiting their use in clinical applications that require real-time responses, such as haptics simulators. Machine learning (ML) approaches have been proposed to help with the reduction of the required time. The present paper reviews cases where ML could help to generate faster simulations, without considerably affecting the performance results.

Methods: This review details the ML approaches used, considering the anatomical structures involved, the data collection strategies, the selected ML algorithms, with corresponding features, the metrics used for validation, and the resulting time gains.

Results: A total of 41 references were found. ML algorithms are mainly trained with FEM-based simulations, in 32 publications. The preferred ML approach is neural networks, including deep learning, in 35 publications. Tissue deformation is simulated in 18 applications, but other features are also considered. The average distance error and mean squared error are the most frequently used performance metrics, in 14 and 17 publications, respectively. The time gains were considerable, going from hours or minutes for purely FEM-based simulations to milliseconds, when using ML.

Conclusions: ML algorithms can be used to accelerate FEM-based biomechanical simulations of anatomical structures, possibly reaching real-time responses. Fast and real-time simulations of anatomical structures, generated with ML algorithms, can help to reduce the time required by FEM-based simulations and accelerate their adoption in the clinical practice.

---

[a] Electronic mail: renzo.phellanaro@mail.mcgill.ca

I. INTRODUCTION

Biomechanical modelling (BM) of the behavior of anatomical structures under various conditions is an essential step for many research and clinical applications. The physical phenomenon being modelled is governed by corresponding partial differential equations (PDEs). BM is used to develop virtual reality simulators with haptic feedback, which are used to train surgeons on the surgical techniques required for minimally invasive surgery[1,2] and to assess innovative interventions for the treatment of heart failure[3]. BM is also used during preoperatory surgical planning, to estimate the risk of rupture of atherosclerotic plaques in vulnerable patients[4]. BM is used in augmented reality applications to deform virtual organ atlases of the liver[5–7], prostate[8], and stomach[9–11] during intraoperative navigation, which are then registered to the medical images of an specific patient. The organ atlases contain important additional information about the internal structure of the organs, such as the position of blood vessels and tumors, which aids the clinician during surgery[12]. Another BM application is the calculation of the response of bones to load during daily activities for orthopaedic implant design[13] and prediction of bone adaptation due to disuse[14]. Fatigue and consequent crack growth in cancellous bone, specifically in the femur, is also studied with BM, in order to prevent and analyze the causes of bone fracture[15,16]. Additionally, prediction of the spine deformation in patients that suffer adolescent idiopathic scoliosis (AIS) is also modelled with BM[17,18].

As it can be noted, the biomechanical modelling of the behavior of anatomical structures is important for many clinical and research applications. In order to achieve a suitable BM, a set of PDEs and their corresponding boundary conditions have to be satisfied. For example, Pellicer-Valero et al.[7] model the elastic bevahior of the liver in response to externally applied forces using an energy balance equation. The solution of this equation requires the total potential energy of the system to be minimum at the equilibrium. The same strategy is used by Madani et al.[19] to model stress in atherosclerotic arterial walls, by minimizing the total potential energy in a situation of equilibrium. A different approach is taken by Mendizabal et al.[20], where the elastic behavior of the breast in response to the presence of an ultrasound probe is modelled by considering the Dirichlet and Neumann boundary conditions.

One of the most widely used methods to solve the PDEs and generate biomechanical simulations is the finite element method (FEM), also referred to as finite element analysis or modelling. In general terms, the FEM provides a set of mathematical tools used to approximate the solution of PDEs that govern the biomechanical phenomenon being analyzed[21]. The FEM allows a proper understanding of how structures and tissues deform given a set of external stimuli and appropriate boundary conditions by discretizing the overall physical model with a representative mesh and evaluating the behaviour of each individual part of such a mesh[22]. By using the FEM, it is possible to retrieve information that is otherwise impossible to obtain experimentally. The FEM also allows the identification of localized deformation and it captures the internal stresses in structures. However, the FEM is also known to require considerable amounts of computational resources, which have been supported by the current advances in hardware and software design[22].

Motivated by the successful applications of the FEM to generate biomechanical simulations, some developments focused on clinical applications where real-time responses to the user are required. The strategies to increase the simulation speed in order to achieve real-time responses include using more powerful hardware elements, such as graphics processing units (GPU), and combine them with fast and parallelizable algorithms[22]. In this scenario, machine learning (ML) algorithms have gained relevance as many scientific works, such as the ones included in this review, have demonstrated that ML algorithms can provide robust, fast, and real-time responses for BM. The publications report a time reduction going from minutes, required to generate a simulation using only the FEM, to milliseconds, when using ML.

ML algorithms can be trained by using biomechanical simulations of the behavior of anatomical structures generated with the FEM. FEM-based simulations are computationally expensive to generate, but a trained ML algorithm can considerably reduce the time required. This reduction in time is possible thanks to the data-driven approach provided by ML algorithms, as an alternative option to the resolution of PDEs through the FEM. Once an ML algorithm is trained with enough data, it can generate a simulation considerably faster than when using the FEM. Besides, in some scenarios, ML can help the FEM by tackling redundant calculations. Or in cases where sequential computations are performed, ML can intervene in some of the steps and aid in accelerating FEM computations.

The present paper reviews the work that has been done in the field of ML to increase the speed of biomechanical simulation of anatomical structures and obtain real-time responses. It is expected that real-time responses would help to increase the use of biomechanical simulations in clinical applications to train new surgeons, plan and assist surgical interventions with simulated biomechanical response, predict the reactions and evolution of anatomical structures under various conditions and morphological states, amongst others.

II. REVIEW METHOD

The publications regarding the use of the FEM and ML for the biomechanical simulation of the behavior of anatomical structures included in this review are obtained using the Google Scholar, PubMed, and Web of Science search engines. All chosen search engines are currently being used in prestigious reviews and meta-analyses[23], and the search features and results they provide are known to complement each other[24].

All publications that mention the terms "finite element analysis", "finite element method", or "finite element modeling", together with the terms "deep learning" or "machine learning" are initially selected for this review. Additionally, publications that are cited on or cite the initial search results are also included if their content includes concepts related to the FEM and ML. All obtained results are then filtered, excluding the ones that do not refer to any anatomical structure. Also, publications before the year 2005 are excluded, as it is after 2005 that the use of ML algorithms to accelerate FEM-based simulations of anatomical structures has gained widespread. Additionally, particular attention is given to publications that report the time required to generate simulations using the FEM and ML-algorithms, and if they explicitly mention the term "real-time".

The first section of this review describes the main ML algorithms used for BM of anatomical structures found in the scientific literature. The next section details the main strategies used to obtain datasets required to build and evaluate the models provided by the ML algorithms. The datasets can be obtained from purely virtual FEM-based simulations, subjects, or a combination of both. Then, a list of strategies on how to split the available datasets on training, validation, and testing sets is provided in the following section, which also includes additional comments on how to properly train an ML algorithm. After that, this review offers a description of the principal input and output features that the ML algorithms use. Finally, the present work analyzes the metrics used for validation, and the time gains achieved when using ML. It is expected that this information would help researchers and clinicians to accelerate the adoption of biomechanical modeling in the clinical practice. A summary of the results of the present review is offered in Tables I and II.

Table 1 placeholder

Table 2 placeholder

III. MACHINE LEARNING ALGORITHMS USED AS AN ALTERNATIVE TO THE FINITE ELEMENT METHOD

Neural networks (NN), tree based algorithms, and support vector regression machines (SVR)[25] are the three main types of algorithms used to generate fast and real-time simulations of anatomical structures. A total of 35 out of the 41 reviewed works use a neural network to generate simulations, thus, making NN the most used type of algorithms. The second most popular type are tree based algorithms, with 6 publications, followed by SVR, with 4 publications. Tables I and II indicate the algorithms used in each application, in the algorithm column. It must be noted that some publications compare the use of more than one type of algorithm to generate the simulations.

In ML, neural networks are a model of computation that resembles the networks of neurons in the brain. In this ML model, a large number of computing elements or artificial neurons connect and interact with each[26]. The adoption of NN for clinical and research applications has increased considerably in recent years[27,28], partly due to the advances in hardware developments, such as GPUs, and the success obtained in many applications by DL[29]. However, the use of NN to simulate the biomechanical behavior of anatomical structures is previous to the invention of DL. Zhong et al. published early work on simulation of the deformations that the liver experiences under externally applied loads, using a cellular neural network[30]. Morooka et al. created the NeuroFEM neural network to study deformations of the liver[2,5] and stomach[11] in the presence of external forces. Other works focus on the use of feedforward and radial basis function (RBF) NN to simulate the behavior of various anatomical structures. Finally, seven reviewed publications use DL architectures, such as the U-Net[20,31], to generate simulations.

Tree based algorithms are the second most popular alternative to generate biomechanical simulations. Tree based algorithms represent knowledge by using a stratified model that includes a series of rules, which are often easy to interpret by humans[26]. The reviewed literature uses decision trees (DT)[3,32,33], extremely randomized trees (ERT)[34], and random forests (RF)[7,17], obtaining promising results. Only two of the publications directly compare the accuracy of the results obtained using RF and NN. In the first case, RF and a feedforward NN are used to estimate liver tissue displacement in the presence of external forces[7]. The feed forward NN displays a vastly superior performance when compared to RF in this particular case. In the second case, a combined approach is used to predict the evolution of the shape of the spine in subjects affected by AIS[17]. Initially, descriptors of the spine shape are calculated using either independent component analysis (ICA) or stacked denoising autoencoders (SDAE), based on a feedforward NN. Then, the descriptors are used as input to a RF algorithm to calculate the final prediction. Both combined options, ICA and RF, and SDAE and RF obtained similar performances.

Finally, SVR has also been used to generate biomechanical simulations of anatomical structures. SVR makes use of the concept of margins or support vectors in high dimensional spaces to tackle the complexity of problems involving the evaluation of multiple features[26]. One application of SVR is the estimation of the risk of rupture of cerebral aneurysms using SVR with a RBF kernel[35], by using hemodynamic parameters calculated with computational fluid dynamics (CFD) and other features as input. In this case, SVR is used as part of the method pipeline and not to replace the CFD calculations. A similar application was proposed by Liang et al.[36] to calculate the risk of rupture of ascending aortic aneurysms using SVR with a RBF kernel. The main difference with the previously cited work is that in this case, the FEM method is only used to generate an annotated dataset for the SVR algorithm. The SVR algorithm is used then for classification of aneurysms as either high or low risk, and also for regression to estimate the pressure risk ratio of the analyzed aneurysm. Lastly, 2 publications compare the performance of SVR with a RBF kernel and a NN with one hidden layer. Cilla et al.[4] estimate the Von Mises stress in the coronary artery by analyzing its geometrical features, obtaining a similar performance with both, SVR with a RBF kernel and NN with one hidden layer. Tonutti et al.[37] estimate the deformation of brain tumors affected by external forces. The accuracy of the results generated by SVR with a RBF kernel was greater than the accuracy obtained with the NN with one hidden layer.

IV. DATA COLLECTION

One crucial element when using ML algorithms is the availability of a proper dataset for training, validation, and testing purposes. In order to obtain the required dataset, the most common solution adopted in the literature is to use the FEM to generate it, with or without underlying data from subjects or physical phantoms, and this strategy is used in 32 of the 41 reviewed publications, as it can be seen on Tables I and II, in the data source column. Alternatively, annotated datasets obtained from subjects can also be used when available.

In some cases, a mesh representing the anatomical structure of interest is created according to statistical models, population sample data reported in the scientific literature, and suggestions of clinicians and other experts in the field. In this case, the mesh does not correspond to the anatomy of any particular subject. In order to generate the simulations, necessary assumptions and simplifications, included as boundary conditions, are made regarding the characteristics of the anatomical structure being analyzed. The FEM is then used to simulate the biomechanical behavior of this mesh in response to the studied phenomena in each application in order to generate a dataset. For example, Morooka et al.[5] generated a mesh representing the liver, assuming elastic properties, but disregarding the viscosity and plasticity of this organ. Also, Deo et al.[9] created a mesh representing the stomach, assuming it to be a Neo-hookean solid. Another example can be seen in the work of Hambli et al.[38], where a femur head is simulated, assuming elastic behavior and isotropic properties for this bone.

In more recent years, a transition from meshes that represent a population sample to meshes that represent individual subjects or physical phantoms has gained popularity. In order to generate the latter type of meshes, usually a medical image of the subject or physical phantom is acquired, the anatomical structure of interest is segmented, and the mesh is then generated by using the segmented organ data. Considering that the mesh corresponds to a subject or physical phantom, it is possible to associate additional data to the mesh, such as individualized material properties, which can be obtained with other techniques used to analyze the anatomical structure of interest. The main reason to use a mesh is that in many cases it is not possible or it is decided not to directly apply the selected phenomena to the anatomical structure being studied. Such decisions are linked to ethical limitations when the subject is alive; the logistics required to apply and record the results of many configurations of the studied phenomena; the possibility of damaging the anatomical structure or physical phantom, turning it unusable for other experiments; and the increase in variables that have to be accounted for in a physical setting, such as experimental variability and environmental factors. Consequently, the FEM is used in these cases to simulate the desired phenomena in meshes that represent anatomical structures reconstructed from medical images of subjects or physical phantoms, in order to generate a dataset without risking any damage to the original anatomical structure. Some examples in the literature include meshes reconstructed from computed tomography (CT) images of *ex vivo* discarded livers intended to be used for transplantation[39] or from magnetic resonance (MR) images of brains that presented a tumor[37]. Additionally, meshes were also reconstructed from ultrasound images of physical phantoms that represent the breast[20] and MR images of the prostate[8]. It should be noted that even when the meshes mentioned in this paragraph come from individual subjects or physical phantoms, the phenomena applied to them are simulations and follow the assumptions and simplifications associated to the FEM.

In some other cases, data regarding the morphology, material properties, and behavior of anatomical structures from individual subjects are used directly to train a ML algorithm, without requiring any additional data generated with the FEM. For example, Ardestani et al.[40] accessed a public database containing the gait trials data of four participants with knee prostheses. Sensors installed in the knee prostheses can measure joint load data. All these data are used to train a feedforward NN capable of predicting gait patterns associated to joint load, and vice versa. Similarly, Komaris et al.[41,42] accessed a public database with kinematic measurements data of twenty-eight regular professional runners. Reflective markers attached to the lower-body of the runners and motion capture cameras are used to record the movement of the lower limbs. The data are then used to train a feedforward neural network (NN) to estimate ground reaction forces, and compare it with the forces measured by an instrumented treadmill.

Another publication[43] collected and prepared bovine pericardium samples. Microscopy images of the samples are then used to train deep learning (DL) models to classify tissue stiffness and to predict nonlinear anisotropic stress-strain curves. The strategy of obtaining data from animal samples is also used by Mouloodi et al.[44,45] as they use nine hydrated bones from cadavers of horses. Different loads are applied to the bones and measures of strain and displacement are registered. The collected data are used to train a NN that can estimate bone loading.

Alternatively, Garcia-Cano et al.[17] validated their approach with a private database of 150 patients collected at the Centre Hospitalier Universitaire Sainte-Justine, Montreal, Canada, with the spinal shape annotated with the help of a trained technician. The data are used to train a ML model that could predict the progression of spinal curve deformation in patients with AIS. Similarly, Guo et al.[18] used a private database with anthropometry records of more than three thousand patients affected by scoliosis, collected between the years of 1975 and 2014 at the Duchess of Kent Children's Hospital at Hong Kong. The data are used to train a DL algorithm, known as a long short-term memory network, which can predict the progression of spinal curve deformation in patients with AIS. Even when both mentioned works are on AIS, no comparison of their strategies has been done yet, to the best of our knowledge.

An exceptional case is noticed in the earliest work considered in this review[30], where a dynamic cellular NN is used to mimic the deformations experimented by the liver tissue when external forces are applied to it. In this case, the parameters of the network are not trained and their setting is left to the user.

V. TRAINING, VALIDATION, AND TESTING SETS FOR MACHINE LEARNING ALGORITHMS

Once the datasets are obtained, they should be assigned to either training, validation or testing sets, which are essential steps in the development of any ML algorithm. In general, the training set is used by an ML algorithm to learn the parameters of the model that describes the available data, the validation set is used to adjust the hyperparameters of the same ML algorithm, and the testing set is used to evaluate the performance of the ML algorithm with previously unseen data[46]. As the results when processing the testing set indicate how well the model learnt by the selected ML algorithm generalizes to new unseen data, these are commonly the results reported in the scientific literature.

One of the main strategies found in the analyzed papers to split the datasets consists of simply selecting a percentage for each set. For example, Cilla et al.[4] assigned 80%, 10%, and 10% of the datasets to the training, validation, and testing sets, respectively. Alternatively, Zadpoor et al.[47] use percentages of 90%, 5%, and 5%, and Komaris et al.[41] use percentages of 60%, 20%, and 20%. The right set of percentages depends on each application and no specific rule was found to select them. However, the amount of data available and the amount of data required by the ML algorithm being evaluated can influence the selected percentages. Particularly, NN require considerable amounts of data to generate suitable models, so the training percentages tend to present high values[47].

It is also possible to repeat the splitting of the datasets with fixed percentages for the training, validation, and test sets. The results of each round of splitting are averaged in order to obtain a more robust report of the performance of the ML algorithm being evaluated. These approach is known as cross-validation[48]. As the datasets are separated in folds that are assigned to either the training, validation or test sets, this approach can also be referred to as k-folding, where k is the number of folds. Liang et al.[36] use cross-validation with 10 folds to evaluate a method to predict the risk of ascending aortic aneurysm. In the particular case where the fold assigned to the testing set includes only one dataset, this approach is known as leave-one-out[48]. Jahya et al.[8] use leave-one-out to evaluate their model to predict deformations of the prostate.

During the present review, it was also observed that some scientific studies do not use a validation set to adjust the hyperparameters of the selected ML algorithm, and either use its default values or tune it empirically. For example, in some cases, the hyperparameters of NN, such as the number of layers, number of hidden units, activation functions, learning rate, number of epochs, and others are not tuned by using a validation set given the considerable computational requirement to train a NN. In those cases, common percentages of datasets assigned to the training and testing datasets are 70% and 30%[32], 80% and 20[20,31,35,45] or 90% and 10%[6,49], respectively.

Another very important aspect to consider besides selecting the training, validation, and testing sets to evaluate an ML algorithm is the avoidance of overfitting, as this phenomenon limits the capacity of the learnt model to be generalized and applied to process new unseen data. Overfitting occurs when an ML algorithm generates highly accurate results in the training set, but the learnt model performs poorly when applied to new unseen data[50].

The techniques used to reduce of overfitting are known in the scientific literature as regularization techniques[50]. When training NN, the most commonly used technique is dropout, and it consists in randomly dropping units of the network and its corresponding connections while the network is trained. As a consequence, the neural network is prevented from adapting too much to the training set[50]. Some examples of NN trained using dropout regularization can be found in[18,19,41,42].

L1 and L2 are alternative regularization techniques, also for NN, which seek to reduce the weight or importance assigned to each individual dataset when training a neural network[51]. As no dataset is given a considerable weight, the possibility of one or a few datasets driving the training of the model is mitigated. Pellicer-Valero et al.[7] use L1 and L2 regularization when training a neural network to simulate deformations of the liver.

Finally, it is also possible to artificially augment the datasets available for training, validation, and testing by adding noise and applying geometric transformations to the available datasets. This technique is known as data augmentation and it seeks to reduce the possibility of overfitting by exposing a machine learning algorithm to a set of datasets with artificially added variation[52]. Pfeiffer et al.[6] use this strategy to model the behavior of the soft tissue of the human organs during surgical navigation.

VI. INPUT AND OUTPUT FEATURES

The estimation of tissue deformation due to external forces is the most frequently observed analysis, in 18 of the 41 reviewed publications. On one hand, it is possible to estimate the deformation of the soft tissue of anatomical structures such as the liver[53], stomach[11], breast[20], and others, as they are affected by external forces applied with surgical instruments. In addition to the deformation experimented by the tissue, some applications also generate as output the reaction forces, which are used, for example, to send feedback to the user in haptics simulators[9,10,53]. On the other hand, it is also possible to estimate the deformation of harder structures, such as the bones as they are affected by load[45]. It was also observed, in 2 publications[44,47], that external forces can be calculated as the output of an ML algorithm that receives tissue deformation data as input. Both publications target bones and they estimate the applied load as output, by sending bone deformation data as input to an ML algorithm.

Another common output of the reviewed ML algorithms are specific tissue properties. First, bone properties, such as crack density and length, bone density, damage, and elastic modulus can be estimated by using ML algorithms to process data regarding the applied stress, load frequency, and other additional properties of the bones[15,16,38,54]. Second, articular cartilage mechanical and physical properties, such as friction, elasticity, and permeability can be calculated by ML algorithms by analyzing force-time data during cartilage indentation tests[55]. Third, the stiffness of ligaments in the foot can be predicted by ML algorithms by processing foot kinematic data[56]. Fourth, elastic properties of tissue samples of bovine heart are estimated by directly processing microscopic images of the samples, with ML algorithms[43]. Finally, material and hemodynamic parameters of models that describe the thoracic region of the aorta and the blood flow inside the aorta are estimated by using ML algorithms to analyze its geometry[57,58].

Stress calculation is also possible by using ML algorithms. Geometrical features of the aorta, coronary, and other arteries are used as input data to estimate the value of the Von Mises stress of plaque and the artery wall[4,19]. Alternatively, data regarding the cartilage load level and subject motion path is used to estimate the value of the Von Mises stress of the involved articulation cartilage[33]. Lastly, stress, pressure, and volume in the heart are simulated by using an ML algorithm to analyze load and material properties data[3].

Finally, some more specific input and output pairs are also present in the reviewed works. These include gait estimation using knee joint implant data[40]; aneurysm rupture risk estimation using shape features[36], geometrical parameters, and wall shear stress data[35]; spinal curve progression estimation using spinal shape data[17] and anthropometry records[18] from patients affected by AIS; zero-pressure shape estimation using aorta geometry data; and ground reaction forces simulation using kinematic measurements as input to the chosen ML algorithm. A summary of all the observed input and output features is included in Tables I and II, in the input features and output features columns.

VII. PERFORMANCE METRICS

Performance metrics are required to quantify the results obtained while training and evaluating a model built with ML algorithms. The present section describes the main performance metrics used in different applications that use ML algorithms to simulate the biomechanical behavior of anatomical structures. A summary of the metrics is also provided in Tables I and II, in the metrics column.

The principal metric used for validation and testing purposes of ML algorithms in the reviewed literature is the mean squared error (MSE), also presented as the value of its squared root or root mean squared error ($RMSE$), so that $RMSE = \sqrt{MSE}$. The MSE is calculated as the average of the differences between the resulting values generated by the chosen ML algorithm ($Y = \{y_i\}, 1 \leq i \leq N$) and their corresponding reference values ($X = \{x_i\}, 1 \leq i \leq N$), squared, as it is shown in Equation 1.

$$MSE(X,Y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2 \tag{1}$$

One of the main advantages of the MSE is being considerably simple to use, with no parameters to tune and low computational complexity. It also exhibits many useful mathematical properties for comparison and optimization of results, such as nonnegativity, identity, symmetry, triangular inequality, convexity, and differentiability[59]. Lastly, thanks to its frequency of use in many scientific publications, it allows performance comparisons with other proposed ML methods.

Some other publications have also used the mean absolute error (MAE) as a performance metric, defined in Equation 2. In some cases, each individual error being averaged has a value smaller than 1, such that when this value is squared during the computation of the MSE, an even smaller value is obtained. Too small values might not be accurately represented due to limitations of the hardware used for experiments, thus, leading to erroneous calculations[60]. In these cases, the MAE is a reasonable alternative.

$$MAE(X,Y) = \frac{1}{N}\sum_{i=1}^{N}|x_i - y_i| \tag{2}$$

Additionally, it is also possible to calculate the normalized values of both the $RMSE$ and the MAE. In the first case, the corresponding metric is the normalized $RMSE$ ($NRMSE$), presented in equation 3 as a function of the $RMSE$[58].

$$NRMSE(X,Y) = \frac{RMSE}{\max(x_i) - \min(x_i)} \times 100\% \tag{3}$$

In the other case, the corresponding metric is the normalized MAE ($NMAE$), presented in equation 4 as a function of the MAE[43,49,57,58].

$$NMAE(X,Y) = \frac{MAE}{\max(x_i) - \min(x_i)} \times 100\% \tag{4}$$

In both cases, the normalization process allows a direct comparison between the $NRMSE$ or $NMAE$ of different estimated parameters.

Another metric based in the MAE found in the literature for biomechanical simulation using ML algorithms is the relative MAE ($RMAE_1$)[4], defined in Equation 5.

$$RMAE_1(X,Y) = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - x_i}{x_i}\right| \tag{5}$$

Notice that x_i corresponds to the reference values of the parameters being estimated. Similarly, to the $NMAE$, the $RMAE$ allows a direct comparison of the performance measurement when estimating different parameters.

In an alternative formulation of the $RMAE_2$[10], the denominator is set to be the maximum reference value ($\max(x_i)$), as it is shown in Equation 6.

$$RMAE_2(X,Y) = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - x_i}{\max(x_i)}\right| \tag{6}$$

A different metric of the performance of an ML algorithm is the Pearson correlation coefficient (PCC)[61] between the ML generated values Y and their corresponding reference values X. There are many ways to present the formula of the PCC, and one of them is indicated in Equation 7.

$$PCC(X,Y) = \frac{\sum_{i=1}^{N} x_i y_i - \frac{\sum_{i=1}^{N} x_i \sum_{i=1}^{N} y_i}{N}}{\sqrt{\sum_{i=1}^{N} x_i^2 - \frac{(\sum_{i=1}^{N} x_i)^2}{N}}\sqrt{\sum_{i=1}^{N} y_i^2 - \frac{(\sum_{i=1}^{N} y_i)^2}{N}}} \tag{7}$$

The PCC measures the strength of the linear relation between X and Y, and its value falls in the range between -1 and 1. A PCC value of -1 reflects a perfectly negative relation, a value of 1 reflects a perfectly positive relation, and a value of 0 indicates the absence of a linear relation between X and Y[61]. Differently from the previous metrics of error, the PCC does not directly measure the difference between estimated Y and reference X values, but rather provides information regarding the linear relation between X and Y.

Finally, there are cases where the biomechanical simulations are used to predict the individual displacement of elements of a group in space and time. These elements can be, for example, nodes of a mesh that represents the soft tissue of the liver[6,7], the breast[20], the brain[37], the prostate[8] or the stomach[11], or points along a line that represents the shape of the spine[17].

In the cases described in the previous paragraph, error metrics based in the distance between corresponding, simulated and reference, elements are commonly used. It is possible to define Z, a vector where each entry corresponds to the space and time coordinates of an element displaced according to an ML-based simulation, and W, a vector where each entry corresponds to the space and time coordinates of the same element displaced according to a reference model. Then, a commonly used metric referred to as the average distance error (ADE)[11] is calculated as the average Euclidian distance between corresponding elements in W and Z. Similarly, it is possible to report the minimum ($MinDE$) and maximum distance errors ($MaxDE$), which correspond to the minimum and maximum Euclidian distance between elements of W and Z[2]. Additionally, the percentage of samples with a distance error lower than a predefined threshold ($PSDE$) is considered a relevant metric when evaluating the performance of an ML algorithm[33].

It has to be noted, as a final note, that this section lists the most commonly used metrics of performance found in the reviewed literature. However, there are more performance metrics suitable for specific applications, which are not included in this section.

VIII. REAL-TIME AND TIME GAINS

As this work is focused on real-time biomechanical simulations, it is important to mention that the term real-time does not have the same particular meaning for each application. Tanembaum defines a real-time operating system as a system where time is a key parameter, such that responses must happen within a precise range of time[62]. The specific value of the range of time is usually expressed in milliseconds in the reviewed publications, but some authors also consider a few seconds as real-time. Particularly, a real-time biomechanical simulation of the liver reported a response time of 3 milliseconds[31], while a real-time biomechanical simulation of the heart reported response times between 5 and 11 seconds[3].

It should also be noted that the response time required for a real-time simulation does not decrease depending on how recent a publication is. The present review found that the response time required for a real-time simulation depends strictly on the type of application being studied. Thus, considering this flexibility in the definition of the concept of real-time, an application is assumed to have reached real-time when its authors report it that way, under their particular requirements.

Tables I and II indicate the time required to generate a biomechanical simulation using either the FEM (time FEM column) or an ML algorithm (time ML column). The time FEM is reported in minutes (*min*) and the time ML is reported in milliseconds (*ms*), in order to keep an organized presentation. All publications where the authors indicate their application generates real-time results have their corresponding time ML in bold. It should be noted that only 14 of the 41 reviewed publications claim to generate real-time results, and 22 of them do not report either the time FEM, time ML, or both. However, in all 19 publications that report both, the time FEM and the time ML, the former is considerably longer than the latter. Consequently, there is evidence in the scientific literature supporting that ML can help to accelerate the biomechanical simulations of the behavior of anatomical structures, obtaining real-time responses in most cases.

IX. DISCUSSION

The objective of the present paper is to provide a review of the publications that use ML algorithms and the FEM to generate biomechanical simulations of the behavior of anatomical structures under a set of predefined conditions. The main reason to use ML algorithms is the considerable time gains obtained, without compromising the overall performance and robustness of the simulations.

The main ML algorithms used in the publications included in this review, NN algorithms are the most frequent choice. One of the main reasons for this selection is that the biomechanical behavior of various anatomical structures can be represented by non-linear functions, which NN have been demonstrated to approximate in an accurate way[29], as NN can include non-linear functions as part of their formulation.

Some tree based algorithms, such as the ones used in the reviewed literature, are also suitable to approximate non-linear functions[63], and they are the second most frequent choice to simulate the biomechanical behavior of anatomical structures. One advantage of tree based algorithms is that they can be represented by tree diagrams, which are relatively easy to be understood by the user[64]. However, in the 2 publications that NN and tree based algorithms were compared to predict the biomechanical behavior of an anatomical structure, NN displayed an either similar[17] or superior performance[7]. Particularly, the publication where NN displayed a similar performance to tree based algorithms processed vectors with hundreds of thousands of elements to describe each training element, while the publication where NN displayed a vastly superior performance processed vectors with hundreds of millions of elements. This fact suggests that NN, specifically deep learning algorithms such as the one used in[7], might be more suitable to handle massive amounts of data, as it has already been reported in other publications[65].

SVR is the third most frequently used ML algorithm in the literature. SVR is known to be a linear regression algorithm, which needs to project non-linear data in a higher dimensional space, by using different kernels, in order to be able to process the data[25]. This strategy is proven to be useful and in the 2 publications that compare SVR and NN algorithms, SVR displays similar[4] or more accurate[37] results than NN. However, it must be noted that the NN algorithms used for comparison included only one hidden layer, which do not included considerably deeper architectures which are currently available as part of the DL approaches[29]. A comparison between SVR and DL approaches may lead to different results and conclusions.

The mentioned ML algorithms require datasets to be trained, validated, and tested. However, obtaining a suitable dataset can be a challenging task when studying anatomical structures, considering the ethical and legal regulations to access and store data, the amount of work required to annotate the data in the case of supervised learning, and the logistics required to guarantee that the selected dataset accurately represents the target population, particularly during the study of anatomical structures of living subjects.

One strategy to obtain the required datasets is to create meshes that represent the anatomical structure of interest and to simulate their behavior under the selected biomechanical phenomena via the FEM. The finite element models are generated according to mathematical models, population sample data reported in other publications, and suggestions of clinicians. The main advantages of this solution are that the ethical and legal regulations that apply to the study are considerably reduced, as no subjects are directly involved in the study. Also, the annotations can usually be generated automatically as part of the FEM-based simulation, and the representativeness of the dataset can be set to an acceptable level by using reported population sample data. Additionally, even when they take a considerable amount of time, FEM-based simulations can be generated in vast amounts and with different parameter settings automatically, which is convenient for data augmentation strategies, commonly used to train ML algorithms such as deep learning (DL)[29].

Even when datasets created using mathematical models, population sample data, and suggestions of clinicians may provide a reasonable approximation of the anatomical structure of interest and its biomechanical behavior, they represent averaged population sample data, which may correspond to no individual subject in the studied population. In order to guarantee that the biomechanical simulations correspond to a subject, many works capture the shape and material properties of the anatomical structure being studied in subjects by using medical imaging modalities and other techniques for material analysis. However, applying the desired biomechanical phenomena may still not be feasible due to ethical reasons, logistics, possible damage to the anatomical structure, and increase of variables to be controlled. Consequently, the FEM is still required to simulate the biomechanical behavior.

The main disadvantage is that the FEM generates simulations by using models designed by a human subject and limited by a set of assumptions and simplifications applied to the biomechanical behavior being studied. Then, in the best case scenario, the ML algorithm will learn the simplified model, which might not be the most suitable choice to represent the behavior of the target anatomical structure. This characteristic limits one of the main advantages of ML algorithms, which is directly letting the machine learn the most suitable model to solve a task. Additionally, the practice of using values reported in the literature to set the parameters of a FEM-based model can limit the generalizability of the ML model to new unreported cases.

In order to avoid imposing any assumptions or simplifications to the model learned by a ML algorithm, the most suitable alternative is to apply it directly to datasets obtained from subjects. Nevertheless, it is difficult to obtain data regarding the shape and materials of an anatomical structure and its biomechanical behavior. In the publications analyzed, only 8 of them could obtain all the mentioned data from subjects. The main strategies used include accessing public and private databases and using animal samples.

One important consideration to have is that, even when ML can deliver faster biomechanical simulations than the FEM in the reviewed publications, this does not imply that the

FEM will eventually be discarded. First, the FEM is still required in many cases to generate annotated training datasets for the ML algorithms, for example when *in vivo* experiments are not possible due to ethical reasons, or when there is a risk of damaging the physical samples or phantoms used to generate the dataset, or when data augmentation strategies requiring an underlying mathematical model are being used. Some publications have also considered integrating both ML algorithms and the FEM as an hybrid approach to their particular problem. For example, Hambli et al.[16,38,54] published various studies where a hybrid method combines the FEM to estimate bone properties at macroscopic scale with NN to estimate bone properties at mesoscale.

Once suitable datasets have been obtained, they should be split in training, validations, and testing sets. A commonly observed approach is to select fixed percentages for each set. The training and evaluation process can be repeated many times in order to obtain robust results, following a technique known as cross-validation. Alternatively, some publications prefer to avoid the generation of a validation set and adjust the hyperparameters of the selected ML model either empirically or by using default values. XXX

The present review also lists the main input and output features used to train, validate, and test different ML applications. It was noted that some input and output pairs are frequently used as a pattern to study different anatomical structures. For example, in the haptics field, it is common to estimate deformations and reaction forces in various anatomical structures when an external force is applied. Another common pattern is the estimation of tissue properties of an anatomical structure after analyzing its biomechanical behavior. It is expected that the identified patterns will help researchers and clinicians to apply ML algorithms to their particular requirements and also to create new application patterns.

Regarding performance metrics, it is always very important to provide a quantitative evaluation of any FEM or ML based biomechanical simulation. These metrics allow the user to compare and select the most suitable approach for their particular problem and take and informed decision. Many publications prefer to use metrics that provide information about the mean error, such as the MAE, MSE or $RMSE$[66]. Nevertheless, it is possible to use the PCC to obtain additional information regarding the strength of the linear relation between the distribution of the reference data and the obtained results. When the concept of displacement in space and time is involved in the evaluation of an ML algorithm, it was observed that metrics for distance error are used. The ADE provides information regarding the mean distance error information, while the $MinDE$ and $MaxDE$ correspond to the minimum and maximum distance error, respectively.

Finally, in terms of the time required to generate a simulation, it was evidenced that in all publications that report it, the time required by simulations generated used only the FEM is considerably longer than the time required by any method that includes ML algorithms. Even more, simulations generated using ML algorithms can provide real-time responses, as defined according to their particular application requirements. Based on these facts, a combined strategy using FEM models to generate datasets offline and ML algorithms trained on those datasets to provide real-time responses can help to accelerate the adoption of biomechanical simulations in the clinical practice, in situations that require fast responses and robustness.

X. CONCLUSION

This work presents a review of the research being done to generate faster biomechanical simulations of the behavior of anatomical structures by using machine learning algorithms and the finite element method, without compromising the performance of the simulations. A total of 41 publications were reviewed, covering a wide spectrum of anatomical structures. For each publication, this review provides details regarding the used ML algorithms, the strategies used to obtain datasets required by those ML algorithms, the techniques employed to split the datasets in training, validation, and testing sets, the selected input and output features for each ML algorithm, the performance metrics considered, and the time gains obtained. In all cases that reported the time required to generate the simulations, ML algorithms were always a faster alternative to purely FEM-based simulations. Even more, 14 publications claimed to have achieved real-time responses by using ML algorithms. It is expected that these fast responses will help to accelerate the adoption of biomechanical simulations in the clinical practice, for applications such as surgical training, planning, intervention, and follow up of patients.

ACKNOWLEDGEMENTS

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), Engage Grants 543780-19 and of Spinologics inc.

REFERENCES

[1] C. Basdogan, M. Sedef, M. Harders, and S. Wesarg, “VR-based simulators for training in minimally invasive surgery,” IEEE Computer Graphics and Applications **27**, 54–66 (2007).

[2] K. Morooka, Y. Nakasuka, R. Kurazume, X. Chen, T. Hasegawa, and M. Hashizume, “Navigation system with real-time finite element analysis for minimally invasive surgery,” in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (IEEE, 2013) pp. 2996–2999.

[3] Y. Dabiri, A. V. d. Velden, K. L. Sack, J. S. Choy, G. S. Kassab, and J. Guccione, “Prediction of left ventricular mechanics using machine learning,” Frontiers in physics **7**, 117 (2019).

[4] M. Cilla, J. Martinez, E. Pena, and M. Á. Martínez, “Machine learning techniques as a helpful tool toward determination of plaque vulnerability,” IEEE Transactions on Biomedical Engineering **59**, 1155–1161 (2012).

[5] K. Morooka, X. Chen, R. Kurazume, S. Uchida, K. Hara, Y. Iwashita, and M. Hashizume, “Real-time nonlinear FEM with neural network for simulating soft organ model deformation,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, 2008) pp. 742–749.

[6] M. Pfeiffer, C. Riediger, J. Weitz, and S. Speidel, “Learning soft tissue behavior of organs for surgical navigation with convolutional neural networks,” International journal of computer assisted radiology and surgery **14**, 1147–1155 (2019).

[7] O. J. Pellicer-Valero, M. J. Rupérez, S. Martínez-Sanchis, and J. D. Martín-Guerrero, “Real-time biomechanical modeling of the liver using machine learning models trained on finite element method simulations,” Expert Systems with Applications **143**, 113083 (2020).

[8] A. Jahya, M. Herink, and S. Misra, “A framework for predicting three-dimensional prostate deformation in real time,” The International Journal of Medical Robotics and Computer Assisted Surgery **9**, e52–e60 (2013).

[9]D. Deo and S. De, "Phyness: A physics-driven neural networks-based surgery simulation system with force feedback," in *World Haptics 2009-Third Joint EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems* (IEEE, 2009) pp. 30–34.

[10]S. De, D. Deo, G. Sankaranarayanan, and V. S. Arikatla, "A physics-driven neural networks-based simulation system (Phynness) for multimodal interactive virtual environments involving nonlinear deformable objects," Presence: Teleoperators and Virtual Environments **20**, 289–308 (2011).

[11]K. Morooka, T. Taguchi, X. Chen, R. Kurazume, M. Hashizume, and T. Hasegawa, "A method for constructing real-time FEM-based simulator of stomach behavior with large-scale deformation by neural networks," in *Medical Imaging 2012: Image-Guided Procedures, Robotic Interventions, and Modeling*, Vol. 8316 (International Society for Optics and Photonics, 2012) p. 83160J.

[12]K. Konishi, M. Nakamoto, Y. Kakeji, K. Tanoue, H. Kawanaka, S. Yamaguchi, S. Ieiri, Y. Sato, Y. Maehara, S. Tamura, *et al.*, "A real-time navigation system for laparoscopic surgery based on three-dimensional ultrasound using magneto-optic hybrid tracking configuration," International Journal of Computer Assisted Radiology and Surgery **2**, 1–10 (2007).

[13]F. Galbusera, A. Fantigrossi, M. T. Raimondi, R. Assietti, M. Sassi, and M. Fornari, "Biomechanics of the C5-C6 spinal unit before and after placement of a disc prosthesis," Biomechanics and Modeling in Mechanobiology **5**, 253–261 (2006).

[14]A. Vahdati and G. Rouhi, "A model for mechanical adaptation of trabecular bone incorporating cellular accommodation and effects of microdamage and disuse," Mechanics Research Communications **36**, 284–293 (2009).

[15]R. Hambli, "Apparent damage accumulation in cancellous bone using neural networks," Journal of the mechanical behavior of biomedical materials **4**, 868–878 (2011).

[16]R. Hambli and N. Hattab, "Application of neural network and finite element method for multiscale prediction of bone fatigue crack growth in cancellous bone," in *Multiscale Computer Modeling in Biomechanics and Biomedical Engineering* (Springer, 2013) pp. 3–30.

[17]E. García-Cano, F. A. Cosío, L. Duong, C. Bellefleur, M. Roy-Beaudry, J. Joncas, S. Parent, and H. Labelle, "Prediction of spinal curve progression in adolescent idiopathic sco-

liosis using random forest regression," Computers in biology and medicine **103**, 34–43 (2018).

[18] X. Guo, S. Xu, Y. Wang, J. P. Y. Cheung, and Y. Hu, "Prediction model of scoliosis progression bases on deep learning," in *International Conference on Cyber-Living, Cyber-Syndrome and Cyber-Health* (Springer, 2019) pp. 431–440.

[19] A. Madani, A. Bakhaty, J. Kim, Y. Mubarak, and M. R. Mofrad, "Bridging finite element and machine learning modeling: stress prediction of arterial walls in atherosclerosis," Journal of biomechanical engineering **141** (2019).

[20] A. Mendizabal, E. Tagliabue, J.-N. Brunet, D. Dall'alba, P. Fiorini, and S. Cotin, "Physics-based deep neural network for real-time lesion tracking in ultrasound-guided breast biopsy," (2019).

[21] S. Brenner and R. Scott, *The mathematical theory of finite element methods*, 1st ed., Vol. 15 (Springer Science & Business Media, 2007).

[22] D. Marinkovic and M. Zehn, "Survey of finite element method-based real-time simulations," Applied Sciences **9**, 2775 (2019).

[23] M. Gusenbauer and N. R. Haddaway, "Which academic search systems are suitable for systematic reviews or meta-analyses? Evaluating retrieval qualities of Google Scholar, PubMed, and 26 other resources," Research Synthesis Methods **11**, 181–217 (2020).

[24] M. Shultz, "Comparing test searches in PubMed and Google Scholar," Journal of the Medical Library Association **95**, 442 (2007).

[25] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support vector regression machines," in *Advances in neural information processing systems* (1997) pp. 155–161.

[26] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms* (Cambridge university press, 2014).

[27] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," Medical image analysis **42**, 60–88 (2017).

[28] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," Briefings in bioinformatics **19**, 1236–1246 (2018).

[29] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature **521**, 436–444 (2015).

[30] Y. Zhong, B. Shirinzadeh, G. Alici, and J. Smith, "Cellular neural network based deformation simulation with haptic force feedback," in *9th IEEE International Workshop on Advanced Motion Control, 2006.* (IEEE, 2006) pp. 380–385.

[31] A. Mendizabal, P. Márquez-Neila, and S. Cotin, "Simulation of hyperelastic materials in real-time using deep learning," Medical image analysis **59**, 101569 (2020).

[32] M. Rupérez, F. Martínez-Martínez, M. Martínez-Sober, M. Lago, D. Lorente, P. Bakic, A. Serrano-López, S. Martínez-Sanchis, C. Monserrat, and J. Martín-Guerrero, "Modeling the mechanical behavior of the breast tissues under compression in real time," in *European Congress on Computational Methods in Applied Sciences and Engineering* (Springer, 2017) pp. 583–592.

[33] J. D. Martín-Guerrero, M. J. Rupérez-Moreno, F. Martinez-Martínez, D. Lorente-Garrido, A. J. Serrano-López, C. Monserrat, S. Martínez-Sanchis, and M. Martínez-Sober, "Machine learning for modeling the biomechanical behavior of human soft tissue," in *IEEE 16th International Conference on Data Mining Workshops* (IEEE, 2016) pp. 247–253.

[34] F. Martínez-Martínez, M. J. Rupérez-Moreno, M. Martínez-Sober, J. Solves-Llorens, D. Lorente, A. Serrano-López, S. Martínez-Sanchis, C. Monserrat, and 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," Computers in biology and medicine **90**, 116–124 (2017).

[35] A. Aranda and A. Valencia, "Study on cerebral aneurysms: Rupture risk prediction using geometrical parameters and wall shear stress with CFD and machine learning tools," Machine Learning and Applications: An International Journal (MLAIJ) **5**, 1–13 (2018).

[36] L. Liang, M. Liu, C. Martin, J. A. Elefteriades, and W. Sun, "A machine learning approach to investigate the relationship between shape features and numerically predicted risk of ascending aortic aneurysm," Biomechanics and modeling in mechanobiology **16**, 1519–1533 (2017).

[37] M. Tonutti, G. Gras, and G.-Z. Yang, "A machine learning approach for real-time modelling of tissue deformation in image-guided neurosurgery," Artificial intelligence in medicine **80**, 39–47 (2017).

[38] R. Hambli, "Application of neural networks and finite element computation for multiscale simulation of bone remodeling," Journal of biomechanical engineering **132** (2010).

[39]D. Lorente, F. Martínez-Martínez, M. J. Rupérez, M. Lago, M. Martínez-Sober, P. Escandell-Montero, J. M. Martínez-Martínez, S. Martínez-Sanchis, A. J. Serrano-López, C. Monserrat, *et al.*, "A framework for modelling the biomechanical behaviour of the human liver during breathing in real time using machine learning," Expert Systems with Applications **71**, 342–357 (2017).

[40]M. M. Ardestani, Z. Chen, L. Wang, Q. Lian, Y. Liu, J. He, D. Li, and Z. Jin, "A neural network approach for determining gait modifications to reduce the contact force in knee joint implant," Medical engineering & physics **36**, 1253–1265 (2014).

[41]D.-S. Komaris, E. Pérez-Valero, L. Jordan, J. Barton, L. Hennessy, B. O'Flynn, and S. Tedesco, "Predicting three-dimensional ground reaction forces in running by using artificial neural networks and lower body kinematics," IEEE Access **7**, 156779–156786 (2019).

[42]D.-S. Komaris, E. Pérez-Valero, L. Jordan, J. Barton, L. Hennessy, B. O'Flynn, and S. Tedesco, "A comparison of three methods for estimating vertical ground reaction forces in running," in *Proceedings of the 38th International Conference on Biomechanics in Sport* (International Society of Biomechanics in Sports, 2020) pp. 1–4.

[43]L. Liang, M. Liu, and W. Sun, "A deep learning approach to estimate chemically-treated collagenous tissue nonlinear anisotropic stress-strain responses from microscopy images," Acta biomaterialia **63**, 227–235 (2017).

[44]S. Mouloodi, H. Rahmanpanah, C. Burvill, and H. M. Davies, "Prediction of load in a long bone using an artificial neural network prediction algorithm," Journal of the Mechanical Behavior of Biomedical Materials **102**, 103527 (2019).

[45]S. Mouloodi, H. Rahmanpanah, C. Burvill, and H. M. Davies, "Prediction of displacement in the equine third metacarpal bone using a neural network prediction algorithm," Biocybernetics and Biomedical Engineering (2019).

[46]B. D. Ripley, *Pattern recognition and neural networks* (Cambridge university press, 2007).

[47]A. A. Zadpoor, G. Campoli, and H. Weinans, "Neural network prediction of load from the morphology of trabecular bone," Applied Mathematical Modelling **37**, 5260–5276 (2013).

[48]M. Kuhn, K. Johnson, *et al.*, *Applied predictive modeling*, Vol. 26 (Springer, 2013).

[49]L. Liang, M. Liu, C. Martin, and W. Sun, "A machine learning approach as a surrogate of finite element analysis–based inverse method to estimate the zero-pressure geometry of human thoracic aorta," International journal for numerical methods in biomedical engineering **34**, e3103 (2018).

[50]N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," The journal of machine learning research **15**, 1929–1958 (2014).

[51]S. J. Nowlan and G. E. Hinton, "Simplifying neural networks by soft weight-sharing," Neural computation **4**, 473–493 (1992).

[52]J. Kukačka, V. Golkov, and D. Cremers, "Regularization for deep learning: A taxonomy," arXiv preprint arXiv:1710.10686 (2017).

[53]W. Abdelrahman, S. Farag, S. Nahavandi, and D. Creighton, "A comparative study of supervised learning techniques for data-driven haptic simulation," in *2011 IEEE International Conference on Systems, Man, and Cybernetics* (IEEE, 2011) pp. 2842–2846.

[54]R. Hambli, "Numerical procedure for multiscale bone adaptation prediction based on neural networks and finite element simulation," Finite elements in analysis and design **47**, 835–842 (2011).

[55]V. Arbabi, B. Pouran, G. Campoli, H. Weinans, and A. A. Zadpoor, "Determination of the mechanical and physical properties of cartilage by coupling poroelastic-based finite element models of indentation with artificial neural networks," Journal of biomechanics **49**, 631–637 (2016).

[56]R. D. Chande, R. H. Hargraves, N. Ortiz-Robinson, and J. S. Wayne, "Predictive behavior of a computational foot/ankle model through artificial neural networks," Computational and mathematical methods in medicine **2017** (2017).

[57]M. Liu, L. Liang, and W. Sun, "Estimation of in vivo constitutive parameters of the aortic wall using a machine learning approach," Computer methods in applied mechanics and engineering **347**, 201–217 (2019).

[58]L. Liang, W. Mao, and W. Sun, "A feasibility study of deep learning for predicting hemodynamics of human thoracic aorta," Journal of Biomechanics **99**, 109544 (2020).

[59]Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? A new look at signal fidelity measures," IEEE signal processing magazine **26**, 98–117 (2009).

[60]R. Phellan, T. Lindner, M. Helle, A. X. Falcão, T. W. Okell, and N. D. Forkert, "A methodology for generating four-dimensional arterial spin labeling MR angiography virtual phantoms," Medical image analysis **56**, 184–192 (2019).

[61]D. M. Lane, D. Scott, M. Hebl, R. Guerra, D. Osherson, and H. Zimmer, *An Introduction to Statistics* (Rice University, 2017).

[62] A. S. Tanenbaum and H. Bos, *Modern operating systems*, 3rd ed. (Pearson, 2015).

[63] M. Ryo and M. C. Rillig, "Statistically reinforced machine learning for nonlinear patterns and variable interactions," Ecosphere **8** (2017).

[64] A. Cutler, D. R. Cutler, and J. R. Stevens, "Random forests," in *Ensemble machine learning* (Springer, 2012) pp. 157–175.

[65] Y. Bengio, I. Goodfellow, and A. Courville, *Deep learning*, Vol. 1 (MIT press, 2017).

[66] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?–arguments against avoiding rmse in the literature," Geoscientific model development **7**, 1247–1250 (2014).

[67] Y. Lu, P. R. Pulasani, R. Derakhshani, and T. M. Guess, "Application of neural networks for the prediction of cartilage stress in a musculoskeletal system," Biomedical signal processing and control **8**, 475–482 (2013).

Ref.	**Year**	**Structure**	**Data**	**Algorithm**	**Input features**	**Output features**	**Metrics**	**Time FEM** (min)	**Time ML** (ms)
30	2006	Liver	-	Cellular NN	External force	Deformation	Qualitative	-	-
5	2008	Liver	FEM	NeuroFEM NN	External force	Deformation	ADE	0.5	**28**
9	2009	Stomach	FEM	RBF NN	External force	Deformation Reaction force	$RMAE$	-	**7**
38	2010	Femur	FEM	NN	Applied stress Load frequency	Bone properties	MSE	16	1000
10	2011	Stomach	FEM	RBF NN	External force	Deformation Reaction force	ADE $RMAE$	0.3	**4**
54	2011	Femur	FEM	Neuromod NN	Applied stress Load frequency	Bone properties	MSE	-	-
15	2011	Femur	FEM	Neuromod NN	Applied stress Bone properties	Apparent damage	MSE	1800	**1000**
53	2011	Liver	FEM	NN	External force	Deformation Reaction force	MAE PCC $RMAE$ $RMSE$	-	-
4	2012	Coronary artery	FEM	Feedforward NN SVR RBF	Geometrical features	Von Mises stress	PCC $RMAE$	-	-
16	2012	Femur	Subjects FEM	Neuromod NN	Applied stress Bone properties	Crack density and length	MSE	1380	1000
11	2012	Stomach	FEM	NeuroFEM NN	External force	Deformation	ADE	0.5	**28**
8	2013	Prostate	Phantom FEM	Feedforward NN	External force	Deformation	ADE	11	**40**
67	2013	Cartilage	FEM	Feedforward NN Focused time delay NN	Load level Motion path	Von Mises stress	MSE PCC	780	690
2	2013	Liver	Subjects FEM	NeuroFEM NN	External force	Deformation	ADE $MinDE$ $MaxDE$	-	-
47	2013	Bone	FEM	Feedforward NN	Deformation	External force	MAE MSE PCC $RMAE$	-	-
40	2014	Knee joint	Subjects	Feedforward NN	Knee joint load	Gait	MSE	-	-
55	2016	Cartilage	FEM	NN	Force-time data	Cartilage properties	MSE PCC	-	110
33	2016	Liver Breast	Subjects FEM	Ensembles of decision trees	External force	Deformation	ADE $PSDE$	-	**430**
56	2017	Foot	FEM	Feedforward NN RBF NN	Foot kinematics	Ligament stiffness	MSE PCC	-	-
43	2017	Bovine heart	Subjects	Deep NN	Microscopy images	Tissue elastic properties	MAE $NMAE$	-	10000
39	2017	Liver	Subjects FEM	Deep NN	External force	Deformation	ADE	1	2890
34	2017	Breast	Subjects FEM	Decision trees Extremely randomized trees Random forests	External force	Deformation	ADE $PSDE$	120	**85**

TABLE I. Summary of the publications included in this review, sorted by year of publication. The time ML is indicated in bold for publications that claim to have achieved real-time responses (part 1).

Ref.	**Year**	**Structure**	**Data source**	**Algorithm**	**Input features**	**Output features**	**Metrics**	**Time FEM (min)**	**Time ML (ms)**
37	2017	Brain tumor	Subjects FEM	NN SVR	External force	Deformation	ADE MSE	150	**7**
36	2017	Thoracic aorta	Subjects FEM	SVR RBF	Shape features	Aneurysm rupture risk	$RMSE$	30	-
35	2018	Brain aneurysm	Subjects FEM	SVR RBF	Geometrical parameters Wall shear stress	Aneurysm rupture risk	-	-	-
17	2018	Spine	Subjects	Feedforward NN Random forests	Spinal shape	Spinal curve progression	$RMSE$	-	-
49	2018	Thoracic aorta	Subjects FEM	Feedforward NN	Aorta geometry	Zero-pressure shape	MAE $NMAE$	27	1000
32	2018	Breast	FEM	Decision trees Extremely randomized trees Random forests	External force	Deformation	ADE	120	**240**
3	2019	Heart	Subjects FEM	Cubist trees XGBoost trees	Load Tissue properties	Stress Pressure and volume	MAE	20 60000	**5000** **11000**
18	2019	Spine	Subjects	Long short-term memory NN	Anthropometry records	Spinal curve progression	MSE	-	-
41	2019	Lower body	Subjects	Feedforward NN	Kinematic measurements	Ground reaction forces	MAE $RMSE$	-	-
57	2019	Thoracic aorta	Subjects FEM	NN	Aorta geometry	Tissue properties	MAE MSE $NMAE$	120	**1000**
19	2019	Arteries	FEM	Deep NN	Geometry and pressure parameters	Von Mises stress	MAE MSE	-	-
20	2019	Breast	Phantom FEM	U-Net NN	External force	Deformation	ADE	0.02	**3**
44	2019	Equine bone	Subjects	NN	Deformation Strain	External force	MSE PCC	-	-
45	2019	Equine bone	Subjects	NN	External force Strain	Deformation	MSE PCC	330	123000
6	2019	Liver	FEM	U-Net NN	External force	Deformation	ADE MSE	-	**20**
42	2020	Lower body	Subjects	Feedforward NN	Kinematic measurements	Ground reaction forces	PCC $RMSE$	-	-
58	2020	Thoracic aorta	Subjects FEM	Deep NN	Aorta geometry	Hemodynamic parameters	MAE $NMAE$ $NRMSE$ $RMSE$	15	1000
31	2020	Liver	Subjects FEM	U-Mesh NN	External force	Deformation	ADE	0.01	**3**
7	2020	Liver	Subjects FEM	Feedforward NN Random forests	External force	Deformation	ADE MAE MSE PCC $PSDE$	-	**4**

TABLE II. Summary of the publications included in this review, sorted by year of publication. The time ML is indicated in bold for publications that claim to have achieved real-time responses (part 2).