Published in final edited form as:

Biomed Signal Process Control. 2013 November 1; 8(6): 475–482. doi:10.1016/j.bspc.2013.04.004.

Application of neural networks for the prediction of cartilage stress in a musculoskeletal system

Yunkai Lu^a, Palgun Reddy Pulasani^b, Reza Derakhshani^b, and Trent M. Guess^{c,*}

^aCivil and Mechanical Engineering, University of Missouri – Kansas City, Kansas City, MO, USA

^bElectrical Engineering, University of Missouri – Kansas City, Kansas City, MO, USA

^cMechanical Engineering, University of Missouri – Kansas City, 350K Flarsheim Hall, 5110 Rockhill Road, Kansas City, MO, USA 64110

Abstract

Traditional finite element (FE) analysis is computationally demanding. The computational time becomes prohibitively long when multiple loading and boundary conditions need to be considered such as in musculoskeletal movement simulations involving multiple joints and muscles. Presented in this study is an innovative approach that takes advantage of the computational efficiency of both the dynamic multibody (MB) method and neural network (NN) analysis. A NN model that captures the behavior of musculoskeletal tissue subjected to known loading situations is built, trained, and validated based on both MB and FE simulation data. It is found that nonlinear, dynamic NNs yield better predictions over their linear, static counterparts. The developed NN model is then capable of predicting stress values at regions of interest within the musculoskeletal system in only a fraction of the time required by FE simulation.

Keywords

Finite element analysis; Musculoskeletal simulation; Neural networks; Cartilage stress

1. Introduction

Numerical simulations can enhance experimental results by predicting values that cannot be directly measured and by providing insight into the interaction of parts within a physical system such as the musculoskeletal system [1–4]. Computational simulations have become an important tool in improving our understanding of ligament injury, tissue damage, and prosthetic design, especially when in vivo measurement of internal forces is not available.

Musculoskeletal modeling based on reasonable assumptions and simplifications, such as treating the knee as a hinge joint or representing the ligaments as one-dimensional elastic elements [5], is common as it provides good prediction of system kinematics and net joint loading at an acceptable computational cost. The multibody (MB) method has been successfully used to model joint behavior [6–12], especially as it pertains to joint kinematics and contact forces. Recently anatomically based MB joint models have been implemented in body level musculoskeletal simulations [1,7]. However, because each part in the model is treated as a rigid body, MB models do not have the ability to predict stress or strain

distribution which can be the determinant factor in joint failure or cartilage degeneration [13,14].

The finite element (FE) method has a major advantage over the MB approach in evaluating the musculoskeletal system as it can provide tissue level information such as contact pressures and tissue stress. However, this advantage comes at the price of prolonged initial stage of model building and meshing, increased demand on the computational resource, and inevitably extended overall simulation time. Although it is very difficult to conduct FE simulations of the whole musculoskeletal system [15], much work has been performed in FE simulations involving systems with limited number of bone/cartilage/ligaments [16,17] and results have been in good agreement with experimental data.

Concurrent prediction of tissue level stress within body level musculoskeletal simulations would provide a valuable tool for understanding the relationships of muscle force and tissue level loading. Presented here is a method that utilizes data-driven neural network surrogate models to predict tissue level stress from MB system forces. These models are constructed and calibrated using contact forces from MB simulations and stress components from FE analyses.

Artificial neural networks (NN) have been successfully used in various biomechanical modeling scenarios [18]. NNs are universal function approximators that use supervised learning to create the desired nonlinear input–output relationship (black-box modeling). Given that, NNs learn from provided examples (data driven modeling) [19], it is important to have adequate and representative data to describe all aspects of the intended function between input and output. This ensures proper convergence of the learning algorithm that is used to find NN model parameters. Though NN training, typically performed using a form of gradient descent algorithm, is time and compute-intensive, the trained NN model executes much more efficiently.

In the presented work, inputs to the NN surrogate model are rigid body reaction forces resulting from cartilage–cartilage contact and outputs are the resulting von Mises tissue stress. The goal of this work is to generate initial datasets from both MB and FE simulations and identify data-driven surrogate models that are capable of describing the stress patterns within the system. Ultimately, the developed surrogate models will be embedded in MB musculoskeletal simulations for concurrent prediction of muscle force and cartilage tissue stress.

2. Model development and methods

To demonstrate the concept of creating a NN surrogate model that captures tissue level behavior, a simplified model of the tibiofemoral joint (Fig. 1) is built in both the FE and MB framework to emulate the typical contact interaction between the femur cartilage and tibia cartilage [1,6–8,18]. The model includes a tibia plateau with a contour derived from medical images of a human subject and a semispherical femoral condyle with a radius of 20 mm. The femoral condyles can be closely approximated with ellipsoid shapes [20] and the radius of the condyle used in this study is consistent with previous studies that have measured the arc radius of the medial condyle to be in the 19–23 mm range [21–23].

Two independent models (MB and FE) were created for this study. Both models share the same loading, boundary, and geometries. The MB simulation yields contact forces between interfaces and reactions forces at the joints. The FE simulation can provide information on stresses and strains (not used in this study) in any region of interest.

2.1. Model description

The geometry of the model is shown in Fig. 2. The simulated femoral condyle (henceforth referred to as simply “condyle”) was modeled as a semi sphere with a radius of 20 mm. The tibia cartilage was modeled as a 40 mm by 40 mm square with a flat bottom surface. The contour of the top surface was mapped from the medial side of a human tibia cartilage geometry acquired from magnetic resonance imaging [6]. The material properties and contact modeling parameters used in this study are listed in Table 1.

Given the importance of adequate and representative training and validation data for NN surrogate modeling, 25 simulations were conducted representing all the combinations from five loading levels (250–750 N at a uniformly increasing step of 125 N) and five motion paths (Fig. 3) applied to the center of the indenter. Each simulation ran for 2 s with a time step of 0.04 s. The condyle was first pressed down vertically until the load was ramped up to its maximum in 0.1 s. The load was then maintained while the x and z displacement controlled motion was enforced at the center of the condyle through the rest of the simulation. The condyle was free to move in the vertical y -axis.

2.2. Multibody simulation in ADAMS

The MB models (Fig. 4) were created in ADAMS (Santa Ana, California) which conducts nonlinear rigid body dynamic simulations in a fraction of the time compared to FE.

In the MB models, a deformable contact was defined between the condyle and cartilage cells using the following equation:

$$F_c = k\delta^{\exp_c} + B(\delta)\dot{\delta} \quad (1)$$

where F_c is the contact force, k represents the contact stiffness, δ is the contact penetration depth, $\exp_c$ represents the contact force exponent, $B(\delta)$ is a damping coefficient and $\dot{\delta}$ is the penetration rate. The values of the parameters for Eq. (1) (listed in Table 1) are the results of tuning to match the maximum vertical displacement of the condyle in ADAMS to its FE counterpart for the case with 500 N load and motion path No. 1.

In order to find the reaction forces on the bottom surface, the cartilage geometry is diced into 20 by 20 cells each with a 1 mm $\times$ 1 mm cross-section using a custom ADAMS macro. Each of the 400 cells is a separate part connected to the “bone” via a fixed joint. Deformable contacts using (Eq. (1)) are then defined between each cell and the condyle. The time history of the reaction forces at each fixed joint in the x , y , and z directions are calculated and used as inputs to NN surrogate models.

2.3. FE analysis

FE simulations (Fig. 5) were conducted in LS-DYNA v971 (the mesh was generated using ABAQUS v10.2 (Simulia Inc., Providence, RI)). The model consisted of three parts – the tibia cartilage (blue), the condyle bone (red), and the condyle cartilage (green). The average thickness of the condyle cartilage which consisted of two layers of elements (Fig. 5) was 3.4 mm. Both the cartilages and condyle bone were modeled as elastic solids and their material properties are listed in Table 1.

Overall, the cartilage was meshed using 8326 hexahedral elements. Contact was defined between the two parts (condyle and cartilage). The second order element formulation was adopted to avoid spurious zero energy mode shape (hourglassing) during the explicit dynamic simulation and to ensure accuracy of the results. A mesh convergence study was conducted and it was determined that convergence was achieved using the mesh density in

Fig. 5 in which the elements on the cartilage had an average length of about 1 mm. The nodes on the bottom surface of the tibia cartilage were fixed in all three directions. The motion control in the x and z direction and the load in the y direction were applied at the center of the top surface of the condyle bone.

After the simulation is complete, the time history data of the von Mises stress within each of the 1600 elements forming the top layer of the tibia cartilage were extracted. Then the 1600 elements were evenly divided into 400 square blocks each of which was made of 2×2 adjacent elements. This was done in order to match the total number (400) of cartilage cells in the ADAMS simulation. The output data of the FE simulations were the time history of the maximal von Mises stress within each of the 400 blocks.

2.4. NN model

NNs have been used to create fast surrogate models [24,25] through nonlinear data-driven curve fitting (reaction force-tissue stress system identification). The correlation coefficient r was used to identify how well the desired (MB–FE generated ground truth) vs. the actual (NN-predicted) output correlate. An r value closer to 1 indicates a good correlation and thus modeling, whereas an r closer to 0 signifies poor correlation and modeling.

Since the type and structure of surrogate models directly affect their learning and generalization capacity [19,26], several linear regression, static feed-forward networks, and (dynamic) focused time delay neural networks (fTDNN) with different number of neurons were examined. Each neural network surrogate model is composed of an input, a hidden, and an output layer. We mainly examined one hidden layer configurations given the forthcoming modeling accuracy figures, combined with our desire to minimize computational complexity, as well as the reported theoretical and practical adequacy of neural networks with a single hidden layer [19,26–28]. Input and output dimensionality (number of input and output nodes of the NN surrogate) were 1200 and 400, respectively, since there were 400 cartilage cells with three reaction forces as the input and one von Mises stress as the desired output at each cell. The number of nodes in the hidden layer was chosen using a semi-exhaustive approach to find the optimal hidden layer size according to the following principles: it is known that the curve fitting and modeling versatility of a NN is increased with its hidden layer size. On the other hand, while improving the accuracy on training datasets, such hidden layer enlargement reduces the generalization power of the NN model, i.e., it reduces its accuracy over previously unseen validation and test datasets [19,26–28]. Given that there are no rigorous theoretical methods for finding the optimal hidden layer size, we started from a smaller hidden layer size and incrementally added nodes to it until a higher r and lower MSE was achieved. More specifically, for static NNs, the number of hidden layer nodes was increased from 10 to 200 in increments of 10. For the fTDNNs, several models with 130–320 hidden layer nodes in increments of 10 and tap delay lengths from 1 to 10 were examined. Eventually, the networks with higher r and lower MSE were found.

Given the nonlinear regression nature of the problem at hand, the NN nonlinearities (transfer functions) were set as sigmoidal for hidden layer and linear for output layer nodes [26]. Linear regression results were also calculated as a simple baseline for comparison purposes when evaluating various NN surrogate model outputs with those of the gold standard in this study, i.e., the FE reference dataset.

In the end, we combined the results of the NNs with the best correlation coefficient values in the form of simple average (sum rule) committees [26,29] as well as stacked generalization fusion. Such combinations produce better results, especially for unseen data. In the sum rule, the idea is to average model outputs so that the uncorrelated portions of their errors cancel

out. Stacked generalization is a more advanced combination method that uses supervised learning to train the combining function for merging the output of participating models into the desired output. It usually uses a training set different from those used to train individual models. A network is trained with all the targets obtained from a few selected networks with the best fits and provided to fusion function. The stacked generalization output usually has less error than the individual networks.

3. Results

3.1. Computational time

Simulations of the MB, FE, and NN models were run on a desktop computer (Intel (R) Xeon (R) CPU X5482@3.2 GHz, 16 GB memory, and Windows 7 Enterprise 64 bit operating system) using MSC ADAMS 2011, LS-DYNA 971, and MATLAB 2012a, respectively. Depending on the load level and the motion path selected, there was a slight variance in the simulation time amongst the 25 cases. A 2 s MB simulation took between 28 and 35 s to run in ADAMS. The running time for one FE simulation ranged from 48,480 s to 52,320 s (13 h 28 min to 14 h 32 min). The typical running time for the NN models was under 2 s and is shown in Table 2. Each NN simulation was run 20 times and the simulation times of the 10th–20th runs were averaged.

3.2. MB simulation

The reaction forces along all three coordinate directions for selected elements and are displayed in Fig. 6.

3.3. FE analysis

Dynamic explicit analysis was performed using double precision solver to improve accuracy. As an example, the von Mises stresses for selected elements are displayed in Fig. 7.

3.4. Evaluation of the NN models

The von Mises stress data from FE simulations (surrogate model output ground truth) were paired with the three dimensional reaction forces of the corresponding cells from the MB simulations (surrogate model input ground truth). 84% of all the simulation data (21 out of 25 FE-generated datasets, each with 861 time samples) were used for NN training. After training, the NN was tested using the remaining 16% of the data not seen during training (the four remaining FE-generated datasets, each with 861 time samples). During training, 70% of the aforesaid training dataset points were randomly chosen for gradient descent training algorithm and the remaining 30% were reserved for validation-based early stopping. The early stopping of the gradient decent algorithm avoids overtraining, i.e., when the NN memorizes the training data to the detriment of its modeling power over unseen data (generalization). Scaled conjugate gradient training algorithm was the preferred gradient descent method to train all the NNs given its higher speed and lower computational complexity [30]. The NN results, shown in Tables 2 and 3 and Fig. 8, indicate that NN outputs closely agree with the actual values from MB–FE simulations.

For fTDNNs, the dynamic (temporal) property of the network comes from the tapped delay lines that are focused at the input of the NN providing limited memory, similar to finite impulse response filters. It was hypothesized that the system being modeled here was almost memoryless and the improvements from fTDNNs were expected to be minimal. The results in Table 3 confirm this assumption, as the MSE and correlation coefficient values of fTDNNs are similar to those of the static NNs.

For the linear surrogate model (baseline linear regression), a three-dimensional linear model was fitted to 80% of the data that were chosen by random selection as the training set. The remaining 20% data were used to test the model, yielding a training MSE of 0.1082 MPa of the von Mises stress and a test MSE of 0.1709 MPa. The predicted vs. target correlation coefficients were 0.9681 for training and 0.9139 for test data.

Static committees were successfully formed by averaging three networks with 80, 100, and 160 hidden nodes (sum rule). The dynamic committees included five networks with 160, 230, 240, 280, and 320 hidden nodes and tap delay lengths of 1 or 5.

Tables 2 and 3 show results of static vs. dynamic NN models for three different working configurations: individual neural nets, stacked generalization fusion, and simple committee using sum rule (averaging outputs). Best results with dynamic neural nets (fTDNNs) were achieved at a tap delay line of 1 (equivalent to a memory depth of one sampling cycle, or $T_s = 20$ ms). Yet the mentioned dynamic model was not significantly better than the static NNs. This confirms the hypothesis that the input–output process for surrogate model is memoryless. It was also observed that stacked generalization was effective for static NNs, but not for fTDNNs.

In summary, it is advantageous to choose a dynamic network over a static network for a MB system in motion. For the model under study, because the motion of the condyle is not highly dynamic and the material models for the cartilage and such are memoryless (i.e., their behavior in one cycle is not related to what happened to them in previous cycles), the improvement from fTDNN is not significant. Hence static networks will suffice. Nonlinear NNs are recommended due to the nonlinear nature of the contact analysis in combination with nonlinear hyper-elastic material models (currently still under study).

4. Discussion

The geometries used in this study were based on human knee geometries, but were simplified to facilitate FE computation speed and data collection. The top surface of the tibia was obtained from magnetic resonance images of a medial tibia plateau, but the bottom surface was flat. The femoral condyle was modeled as a semisphere with a radius derived from femoral condyle geometries from the same knee used for the tibia surface. The von Mises stress predicted for this study is consistent with other knee studies. In a subject specific finite element model of the knee and lower leg, Liu and Zhang [31] predicted peak von Mises stresses in the femoral cartilage that ranged from 2.57 MPa to 2.24 MPa depending on loading conditions at the foot. Static forces from the major muscles crossing the knee and ankle were applied to the Liu and Zhang model during simulation. In their FE model, Li et al. [32] predicted peak knee cartilage von Mises stress between 1 and 1.5 MPa for a static compressive force across the knee of 1400 N. A Young's modulus of 5 MPa was used for these predictions.

In the FE model, individual nodes are connected to neighboring nodes. In the MB model, individual rigid body cartilage cells are not connected to neighboring cells. As a result, cells that are not in direct contact with the condyle will not be subjected to external forces. However, FE nodes at the same location will be subjected to normal and shear stresses due to connections to neighboring nodes and the requirement of equilibrium. By including inputs from all tibia cartilage cells, the NN model will predict stress even in MB cells that are not directly in contact with the condyle.

The advantages of the approach used in this study include: (1) the ability to embed the NN models within larger scale models for concurrent multi-scale simulations, (2) improvements in computational efficiency, and (3) the extendibility to various new loading situations and

materials. FE analysis provides insight to the localized stress and strains at an increased computational cost. Even the convergence study of the mesh density may take days to finish in order to find the optimal mesh sizes. The study here introduced a different route using neural networks to predict the stresses. With the successful application of this method, evaluation of the stress distribution can be obtained in minutes instead of hours or days as it would take the traditional FE analysis to run. In addition, the input–output structure of the NN allows it to be embedded in MB simulations for concurrent multi-scale simulations. For example, body level muscle forces and tissue level stress can be concurrently predicted in the same model.

To build a network that is capable of generating data of high accuracy, it is beneficial to train the neural network with as much data as possible. More training data, however, will require more computation time. Hence, an optimization study is in place to find the balance between good accuracy and size of input data pool.

5. Conclusions

The study here demonstrates a methodology that fully utilizes the efficiency of the MB simulation and the versatility of the NN prediction. Only limited cases of FE simulation need to be run to generate stress/strain data along with the corresponding force data in MB simulations to train and build NN models. The constructed NN models use future force inputs from MB simulations under various different loading scenarios to predict the stress/strain distribution in those situations. The computation time and cost are drastically reduced through this process as compared to the traditional FE analysis.

Acknowledgments

Funding for portions of this project were provided by the Missouri Life Sciences Research Board (Award # 09-1078) and the National Institute of Arthritis and Musculoskeletal and Skin Diseases (Award # RAR061698A).

References

1. Guess TM. Forward dynamics simulation using a natural knee with menisci in the multi-body framework. *Multibody System Dynamics*. 2012; 28:37–53.
2. Nigg, BM.; Herzog, W. *Biomechanics of the Musculo-Skeletal System*. 3rd ed.. New York: Wiley; 2007.
3. Tawhai M, Bischoff J, Einstein D, Erdemir A, Guess T, Reinbolt J. Multiscale modeling in computational biomechanics. *Engineering in Medicine and Biology Magazine IEEE*. 2009; 28:41–49.
4. Stylianou AP, Guess TM, Cook JL. Development and validation of a multibody model of the canine stifle joint. *Comput Methods Biomech Biomed Engin*. 2012 <http://dx.doi.org/10.1080/10255842.2012.684243>.
5. Bloemker KH, Guess TM, Maletsky L, Dodd K. Computational knee ligament modeling using experimentally determined zero-load lengths. *The Open Biomedical Engineering Journal*. 2012; 6:33–41. [PubMed: 22523522]
6. Guess TM, Liu H, Bhashyam S, Thiagarajan G. A multibody knee model with discrete cartilage prediction of tibio-femoral contact mechanics. *Computer Methods in Biomechanics and Biomedical Engineering*. 2011; 16:256–270. [PubMed: 21970765]
7. Guess TM, Stylianou A. Simulation of anterior cruciate ligament deficiency in a musculoskeletal model with anatomical knees. *The Open Biomedical Engineering Journal*. 2012; 6:23–32. [PubMed: 22470411]
8. Guess TM, Thiagarajan G, Kia M, Mishra M. A subject specific multibody model of the knee with menisci. *Medical Engineering & Physics*. 2010; 32:505–515. [PubMed: 20359933]

9. Bei Y, Fregly BJ. Multibody dynamic simulation of knee contact mechanics. *Medical Engineering & Physics*. 2004; 26:777–789. [PubMed: 15564115]
10. Caruntu DI, Hefzy MS. 3-D anatomically based dynamic modeling of the human knee to include tibio-femoral and patello-femoral joints. *ASME Journal of Biomechanical Engineering*. 2004; 126:44–53.
11. Fisk J, Wayne J. Development and validation of a computational musculoskeletal model of the elbow and forearm. *Annals of Biomedical Engineering*. 2009; 37:803–812. [PubMed: 19156525]
12. Spratley E, Wayne J. Computational model of the human elbow and forearm: application to complex varus instability. *Annals of Biomedical Engineering*. 2011; 39:1084–1091. [PubMed: 21132368]
13. Dekel S, Weissman SL. Joint changes after overuse and peak overloading of rabbit knees in vivo. *Acta Orthopaedica Scandinavica*. 1978; 49:519–528. [PubMed: 216227]
14. McCann L, Ingham E, Jin Z, Fisher J. Influence of the meniscus on friction and degradation of cartilage in the natural knee joint. *Osteoarthritis and Cartilage*. 2009; 17:995–1000. [PubMed: 19328878]
15. Erdemir A, Guess TM, Halloran J, Tadeipalli SC, Morrison TM. Considerations for reporting finite element analysis studies in biomechanics. *Journal of Biomechanics*. 2012; 45:625–633. [PubMed: 22236526]
16. Peña E, Calvo B, Martínez MA, Palanca D, Doblaré M. Finite element analysis of the effect of meniscal tears and meniscectomies on human knee biomechanics. *Clinical Biomechanics*. 2005; 20:498–507. [PubMed: 15836937]
17. Yao J, Salo AD, Lee J, Lerner AL. Sensitivity of tibio-menisco-femoral joint contact behavior to variations in knee kinematics. *Journal of Biomechanics*. 2008; 41:390–398. [PubMed: 17950743]
18. Mishra M, Derakhshani R, Paiva GC, Guess TM. Nonlinear surrogate modeling of tibio-femoral joint interactions. *Biomedical Signal Processing and Control*. 2011; 6:164–174.
19. Haykin, SO. *Neural Networks and Learning Machines*. 3rd ed. New York: Prentice Hall; 2009.
20. Li K, Tashman S, Fu F, Harner C, Zhang X. Automating analyses of the distal femur articular geometry based on three-dimensional surface data. *Annals of Biomedical Engineering*. 2010; 38:2928–2936. [PubMed: 20496005]
21. Nuno N, Ahmed AM. Three-dimensional morphometry of the femoral condyles. *Clinical Biomechanics (Bristol, Avon)*. 2003; 18:924–932.
22. Siebold R, Axe J, Irrgang JJ, Li K, Tashman S, Fu FH. A computerized analysis of femoral condyle radii in ACL intact and contralateral ACL reconstructed knees using 3D CT. *Knee Surgery, Sports Traumatology. Arthroscopy*. 2010; 18:26–31.
23. Howell SM, Howell SJ, Hull ML. Assessment of the radii of the medial and lateral femoral condyles in varus and valgus knees with osteoarthritis. *The Journal of Bone and Joint Surgery*. 2010; 92:98–104. American Volume. [PubMed: 20048101]
24. Sherwood, J.; Derakhshani, R.; Guess, T. A comparative study of linear and nonlinear data-driven surrogate models of human joints; Region 5 Conference, 2008 IEEE; 2008. p. 1-6.
25. Song R, Tong KY. Using recurrent artificial neural network model to estimate voluntary elbow torque in dynamic situations. *Medical & Biological Engineering & Computing*. 2005; 43:473–480. [PubMed: 16255429]
26. Príncipe, JC.; Euliano, NR.; Lefebvre, WC. *Neural and Adaptive Systems: Fundamentals Through Simulations*. New York: Wiley; 1999.
27. Alpaydin, E. *Introduction to Machine Learning: Adaptive Computation and Machine Learning*. Cambridge, MA: MIT Press; 2004.
28. Bishop, CM. *Pattern Recognition and Machine Learning*. Cambridge, UK: Springer Science; 2006.
29. Lipnickas A, Korbicz J. Adaptive selection of neural networks for a committee decision. *Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Proceedings of the Second IEEE International Workshop*. 2003:109–114.
30. Møller, MF. *A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning*. Denmark: Aarhus University Publishing; 1990.

31. Liu X, Zhang M. Redistribution of knee stress using laterally wedged insole intervention: finite element analysis of knee-ankle-foot complex. *Clinical Biomechanics* (Bristol, Avon). 2013; 28:61–67.
32. Li G, Lopez O, Rubash H. Variability of a three-dimensional finite element model constructed using magnetic resonance images of a knee for joint contact stress analysis. *Journal of Biomechanical Engineering*. 2001; 123:341–346. [PubMed: 11563759]

Fig. 1.

An example of the intended use of the neural network model within a MB musculoskeletal model of the knee (a) and the prototype MB model used in this study (b). The contact force (F_c) resulting from contact with the condyle (or femur in the knee model) results in reaction forces at the bone cartilage interface (F_x , F_y , F_z) for each tibia cartilage cell. These reaction forces provide input to the NN model that then predicts the von Mises stress at each cell location.

Fig. 2.
Creation of model geometries used in this study.

Fig. 3.
Top view of the five motion paths used in the MB/FE simulations.

Fig. 4.
The MB model as built (a), and at an intermediate step of indenter motion showing the contact area (b).

Fig. 5.
Model in LS-DYNA.

Fig. 6. Selected cartilage cells of MB model (a), and three dimensional reaction forces at cell fixed joints (b). All results are from MB simulation with 500 N load and No. 4 motion path of indenter.

Fig. 7. Selected cartilage elements on top surface of FE model (a), contour plot of von Mises stress at one instant of motion (b), and von Mises stresses for the selected elements (c). All results are from FE simulation with 500 N load and No. 4 motion path of indenter.

Fig. 8.
NN model prediction vs. FE simulation results of von Mises stresses of selected elements.

Table 1

Material properties and contact parameters used in the simulation.

Young's modulus	15.0 MPa
Poisson's ratio	0.47
Density	10^{-6} kg/mm ³
Contact stiffness, k	10 N/mm
Contact damping coefficient, $B(\delta)$	2.55 N s/mm
Contact force exponent, $\exp$	1.5

Table 2

Test correlation coefficients and simulation times.

	Individual		Stacked generalization		Committee	
	Test <i>R</i>	Average execution time (s)	Test <i>R</i>	Average execution time (s)	Test <i>R</i>	Average execution time (s)
Dynamic NN	0.891	0.071	0.894	1.390	0.914	1.364
Static NN	0.889	0.016	0.927	0.569	0.899	0.543

Table 3

Results using fTDNN and static NN (train and test MSE are in MPa²).

Network type	Hidden nodes	Tap delay length	Train MSE	Test MSE	Train R	Test R
Best fTDNN network	280	1	0.111	0.210	0.966	0.883
Best fTDNN committee			0.108	0.171	0.968	0.914
Best static network	80		0.170	0.192	0.918	0.889
Best static committee			0.152	0.180	0.935	0.903