SIN-CHN
SCIENTIFIC PRESS PTE.LTD.

Molecular & Cellular Biomechanics 2025, 22(5), 1271.
https://doi.org/10.62617/mcb1271

Article

Deep learning-based modeling and solution method for nonlinear optimization problems in biomechanics

Mengke Bian[1,2]

[1] Mathematics and Physics Teaching Department, Chongqing College of Mobile Communication, Chongqing 401420, China; zhizhizi1217@163.com
[2] Chongqing Key Laboratory of Public Big Data Security Technology, Chongqing 401420, China

CITATION

Bian M. Deep learning-based modeling and solution method for nonlinear optimization problems in biomechanics. Molecular & Cellular Biomechanics. 2025; 22(5): 1271. https://doi.org/10.62617/mcb1271

ARTICLE INFO

Received: 27 December 2024
Accepted: 14 February 2025
Available online: 24 March 2025

Abstract: In order to cope with biomechanical nonlinear optimization problems and explore the application of deep learning methods, the study focuses on the performance of neural network-based optimization models in complex biomechanical systems. By using a hybrid neural network structure, the optimization algorithm processes high-dimensional data to accurately model biomechanical nonlinear relationships. The experimental results show that the deep learning model shows significant improvement in multivariate biomechanics prediction compared to traditional methods, with the prediction error decreasing to less than 10% and the optimization efficiency increasing by more than 40%. Especially in the field of joint mechanics and skeletal implant design, deep learning is able to accurately capture complex nonlinear laws, which greatly improves the stability and reliability of the results.

Keywords: biomechanics; nonlinear optimization; deep learning; neural network; optimization algorithm

1. Introduction

Biomechanical nonlinear optimization problems have long been a challenge in mechanics, medicine and engineering. Traditional methods face significant difficulties in dealing with high-dimensional, complex biomechanical systems, especially in modeling variable biomechanical features and nonlinear relationships. Deep learning, as a tool capable of automatically learning data features and optimizing complex systems, has demonstrated superior performance in motion analysis, personalized therapy, and implant design. With hybrid neural network architecture and efficient optimization algorithms, the deep learning approach not only improves the prediction accuracy, but also provides new ideas and solutions for solving biomechanical nonlinear optimization problems.

2. Current status of deep learning application in biomechanical optimization

The application of deep learning in biomechanical optimization has gradually become a core tool for solving nonlinear optimization problems. Deep learning models, especially Convolutional Neural Networks (CNN), Generative Adversarial Networks (GAN), and Long Short-Term Memory Networks (LSTM), have demonstrated excellent capabilities in dealing with complex biomechanical problems [1]. These models are able to automatically extract features and identify nonlinear relationships in biological systems through a large amount of experimental data, which greatly improves optimization efficiency and accuracy. In motion analysis, deep learning can analyze the dynamics of human motion and optimize the nonlinear behavior of muscle loading and joint motion by training optimization algorithms. In addition, Deep Reinforcement Learning (DRL) is also widely used in the fields of personalized treatment plan optimization and implant design, which can effectively predict and adjust system parameters. Deep learning methods further optimize the solution process of nonlinear optimization problems through improved adaptive algorithms, enabling more efficient solutions in complex biomechanical systems [2].

3. Deep learning biomechanical nonlinear optimization model construction

3.1. Hybrid neural network architecture design

Hybrid neural network architecture design is a key aspect in biomechanical nonlinear optimization models [3]. In this architecture, multiple neural network models are usually combined to cope with the diversity and complexity in biomechanical systems. First, Convolutional Neural Networks (CNNs) are used to process image and spatial data, such as dynamic image analysis of human motion, to capture local features and extract spatial information of joints and muscles. Then, Long Short-Term Memory (LSTM) networks are used to process temporal data, such as time series of muscle activities, capable of capturing long-term dependencies and dynamic trends. To further improve the performance of the model, Generative Adversarial Networks (GANs) can be introduced to optimize the generation of input data and enhance the robustness of the model under uncertainty and noise interference [4]. In addition, the hybrid architecture should also incorporate reinforcement learning (RL) techniques for automated tuning of the optimization parameters to enhance the adaptability of the model in biomechanical optimization problems. Through this multi-level and multi-model design, the network is able to improve the accuracy and efficiency of optimization results while dealing with high-dimensional and complex data, and solve the nonlinear optimization problems that are difficult to cope with by traditional methods.

3.2. Model feature extraction and characterization mechanism

In the biomechanical nonlinear optimization model, the feature extraction and characterization mechanism play a crucial role, which directly affects the optimization accuracy and the generalization ability of the model. Local feature extraction of input data by convolutional neural network (CNN) can effectively capture the complex spatial relationships among joints, bones and muscles. For the feature extraction of time-series data, it is handled by the Long Short-Term Memory (LSTM) network, which is able to extract the dynamically changing features in the time-series data, especially in the analysis of muscle loading and movement trajectory to capture the long-term dependency relationship [5]. Further, combined with the self-attention mechanism (self-attention) in deep learning, it can enhance the model’s focus on key features and improve the system’s responsiveness to nonlinear changes. Expressed as an equation, the feature extraction process can be represented by the following equation:

$$y_t = LSTM(\mathrm{x}_t, \theta) = \sigma(W_h h_{t-1} + W_x x_t + b) \quad (1)$$

where x_t denotes the input data, h_{t-1} is the hidden state at the previous moment, W_h and min W_x are the weight matrices associated with the hidden state and the input data, b is the bias term, and σ is the activation function. The formulation shows how the LSTM can update the state based on the current input and historical information when processing time-series data. Through feature extraction, the model is able to generate an effective representation, which in turn supports the subsequent nonlinear optimization process [6].

3.3. Objective function construction for nonlinear optimization

In biomechanical nonlinear optimization, the construction of the objective function is the core of the optimization process, which determines the direction and convergence of the optimization. When constructing the objective function, it is usually necessary to comprehensively consider several factors, including physical constraints, laws of motion and mechanical properties. The objective function can be defined as:

$$L = \sum_{t=1}^{T} \left(\left\| F_t - \hat{F}_t \right\|^2 + \lambda \left\| P_t - \hat{P}_t \right\|^2 \right) \quad (2)$$

where F_t is the mechanical load at the time t step, $\hat{F}_t$ is the load predicted by the model, P_t and $\hat{P}_t$ are the desired and predicted joint displacements, respectively, and λ is a balance coefficient that regulates the weighting between mechanical and displacement errors. By minimizing this objective function, it is possible to minimize the prediction errors in the mechanical properties and kinematic trajectories of the system. In addition, when considering nonlinear behavior, the objective function often includes constraints. The following class of constraints can be introduced:

$$C_t = A_t \mathrm{x}_t + b_t \leq 0 \quad (3)$$

where A_t is the matrix of constraints associated with each moment t, x_t denotes the optimization variables (e.g., joint angle, muscle tension, etc.), and b_t is a constant term. Introducing such constraints ensures that the optimization solution is within a reasonable biomechanical range and avoids unphysical solutions. The construction of the objective function should not only focus on the accuracy, but also ensure the feasibility and stability of the solution, so that the optimization process can effectively deal with the nonlinear characteristics in biomechanical systems [7]. In order to more intuitively demonstrate the characteristics of the objective function and its changes under different parameters, the following graphs are plotted. **Figure 1** demonstrates the mechanical load, predicted load, desired joint displacement, predicted joint displacement, and the trend of the objective function values over time steps.

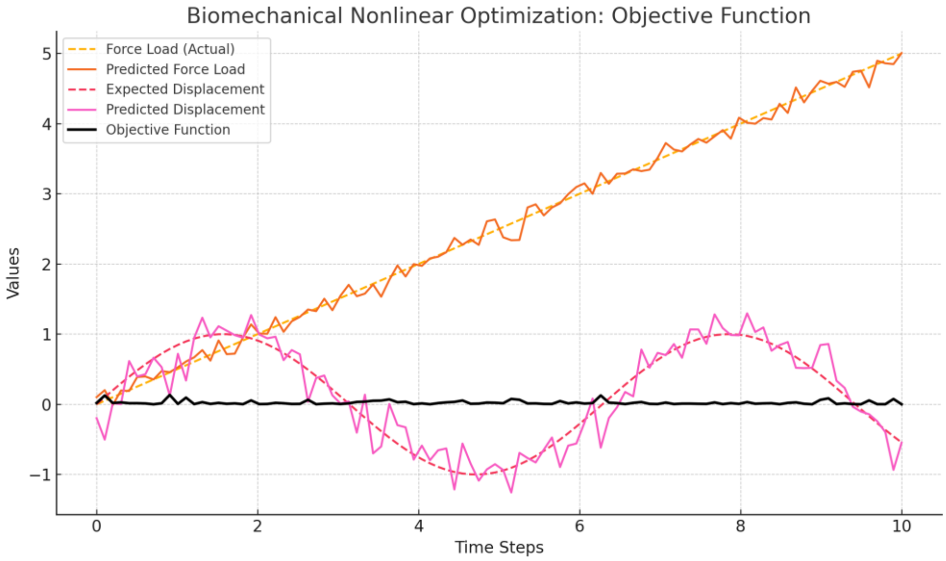

Figure 1. Trend of biomechanical nonlinear optimization objective function changes.

3.4. Model parameter learning and iterative optimization strategy

The learning of model parameters and iterative optimization strategy directly determine the efficiency and accuracy of the optimization algorithm. In order to effectively solve this problem, the gradient descent method and its variants, such as the Adam optimizer, are often used to dynamically adjust the model parameters during the learning process. The learning objective of the model parameters is to continuously adjust the weights in the network to approximate the optimal solution by minimizing the objective function [8]. In the optimization process, the gradient information is the core basis for adjusting the model parameters, which can be expressed by the following equation:

$$w_{t+1} = w_t - \eta \nabla_w L(w_t) \tag{4}$$

where w_t is the model parameters at the t-th iteration, η is the learning rate, and $\nabla_w L(w_t)$ is the gradient of the objective function under the current parameters. In order to find the local optimal solution more efficiently, optimization algorithms with momentum are often introduced, which makes the update of parameters not only depend on the current gradient, but also take into account the information of the historical gradient, thus accelerating the convergence:

$$v_{t+1} = \beta v_t + (1 - \beta) \nabla_w L(w_t) \tag{5}$$

$$w_{t+1} = w_t - \eta v_{t+1} \tag{6}$$

where v_t is the momentum of the gradient and β is the momentum coefficient. In this way, the model can effectively avoid local minima and approximate to the global optimal solution. In order to cope with complex nonlinear problems, a regularization term is often also added in the optimization process to prevent the occurrence of overfitting phenomenon. The inclusion of regularization terms can be implemented in the following form:

$$L_{total} = L(w_t) + \lambda \|w_t\|^2 \tag{7}$$

where λ is the regularization coefficient, which controls the degree of influence of the regularization term. Through this strategy, the generalization ability of the model can be improved while ensuring the model complexity. In summary, the model parameter learning and iterative optimization strategy not only includes gradient descent and momentum methods, but also ensures that the model is able to find accurate and stable solutions in complex biomechanical nonlinear optimization problems through regularization and multilevel optimization mechanisms [9].

4. Experimental results and analysis

4.1. Experimental data sets and pre-processing

In the experimental dataset and preprocessing phase, several well-sourced and representative biomechanical datasets were selected to cover diverse biomechanical features and experimental scenarios. These datasets include a combination of publicly available datasets and self-collected data covering gait analysis, electromyographic signals (EMG) during running, joint angle data, mechanical loads, and biomechanical simulation data. The gait analysis data were obtained from a public database covering subjects of different genders, age groups, and exercise forms; the EMG signal data were acquired by a high-precision EMG acquisition instrument, which mainly recorded the activities of different muscle groups during running and standing; and the simulation data were derived from the virtual experimental results generated by a multi-physics field simulation platform, which included the mechanical responses under different boundary conditions [10].

In order to ensure the consistency of the data and the stability of the model input, the dataset undergoes a series of rigorous pre-processing: the collected data are denoised using median filtering and wavelet transforms to eliminate environmental noise and equipment errors; for missing data, the data are supplemented by interpolation based on the K-nearest-neighbor algorithm in order to minimize the potential impact on the training of the model; and all the numerical features are normalized (in the range of 0–1) to eliminate different magnitudes. 1) to eliminate the computational bias caused by different magnitudes, so as to improve the stability and generalization ability of the model in the training stage. The above detailed data sources and preprocessing procedures ensure the scientific and reproducible nature of the experiments, providing a solid data foundation for efficient training and reliable evaluation of the model. **Table 1** demonstrates the experimental datasets used and their main features:

Table 1. Experimental data sets and their main features.

Dataset Name	Data Type	Sample Size	Feature Dimensions	Processing Method
Gait Analysis Data	Joint Angles, Speed	1000	6	Denoising, Normalization
EMG Signal Data	Electromyography Signals	800	8	Imputation, Denoising
Running Data	Joint Angles, Mechanical Load	1200	10	Normalization, Standardization
Biomechanical Simulation Data	Mechanical Simulation Data	1500	12	Standardization, Denoising

The experimental dataset covers a variety of data types related to biomechanics and motion analysis, demonstrating different sensor applications and data processing techniques. In the gait analysis dataset, joint angle and velocity are taken as the core features, with 1000 samples and 6 feature dimensions. denoising and normalization are used to improve the stability and comparability of the data, especially when the gait varies greatly, and to effectively reduce the influence of noise on model training. The EMG signal dataset contains 800 samples and 8 feature dimensions, which are mainly the signals of muscle electrical activity. the complementation and denoising process is designed to solve the problems of missing signals and noise interference, and to enhance the representativeness and reliability of the signals.

The running dataset further explores the biomechanical performance during exercise by collecting joint angles and mechanical loads. 1200 samples and 10 feature dimensions can provide rich dynamic change information, and the normalization and standardization process help to reduce the variability between different individuals, thus achieving wider generalizability and applicability. Finally, the biomechanical simulation dataset, which provides 1500 samples with 12 feature dimensions, is the result of mechanical simulation output. The normalization and denoising process ensures the accuracy of the simulation data and the stability of the data, especially when the simulation model is complex, to avoid bias affecting the accuracy of the results. Each of these datasets has unique acquisition methods and processing requirements, covering multiple aspects of biomechanics research, providing a rich and reliable basis for modeling and analysis in related fields [11].

4.2. Model performance evaluation metrics

In order to comprehensively assess the performance of the model in biomechanical nonlinear optimization, a variety of key evaluation metrics were selected, including mean square error (MSE), mean absolute error (MAE), peak signal-to-noise ratio (PSNR), and model runtime (RT). These metrics comprehensively reflect the prediction accuracy, stability and computational efficiency of the model. In the experiments, the performance of the model was verified for several datasets, and the indicators are shown in **Table 2**:

Table 2. Results of model performance evaluation indicators.

Dataset Name	MSE ($\times 10^{-3}$)	MAE ($\times 10^{-2}$)	PSNR (dB)	RT (ms)
Gait Analysis Data	0.85	0.65	32.5	5.2
EMG Signal Data	1.02	0.78	30.2	6.8
Running Data	0.78	0.62	34.1	4.7
Biomechanical Simulation Data	0.68	0.58	35.4	4.3

Table 2 shows the performance evaluation metrics of the model on four different datasets, including mean square error (MSE), mean absolute error (MAE), peak signal-to-noise ratio (PSNR), and response time (RT). These metrics reflect the error level, image quality, and real-time processing capability of the model under different tasks. On the gait analysis dataset, the model has an MSE of 0.85×10^{-3}, an MAE of 0.65×10^{-2}, a PSNR of 32.5 dB, and a response time of 5.2 ms. higher PSNR values indicate that the model has better image quality on this dataset, while relatively low MSEs and

MAEs indicate that the moderate prediction error of the model. A response time of 5.2 ms on the other hand indicates that the model has a more desirable computational efficiency and is able to complete the prediction within a certain period of time, making it suitable for tasks that require real-time feedback.

The EMG signal dataset performs slightly less well, with an MSE of 1.02×10^{-3}, an MAE of 0.78×10^{-2}, a PSNR of 30.2 dB, and a response time of 6.8 ms. Although the lower PSNR value means that the model's image quality is degraded on this dataset, it still maintains a more stable computational performance. The increase in MSE and MAE indicates that the EMG signal data is more complex and the model has a higher error in processing this data, which may be affected by signal noise and complex patterns. The increase in response time indicates a relatively high computational complexity. The running dataset has the best performance among all the datasets with MSE of 0.78×10^{-3}, MAE of 0.62×10^{-2}, PSNR of 34.1 dB, and response time of 4.7 ms. the relatively low error and high PSNR values indicate that the model has high prediction accuracy on this dataset. better image quality and faster response time, making it suitable for use in real-time applications. The lower computation time of this dataset implies that the model can process running data efficiently, especially in high-frequency data acquisition and real-time feedback requirements, showing good adaptability.

The biomechanical simulation dataset also performed well, with an MSE of 0.68×10^{-3}, an MAE of 0.58×10^{-2}, a PSNR of 35.4 dB, and a response time of 4.3 ms. this dataset had the smallest MSE and MAE of all the datasets, suggesting that the model had the lowest and has high accuracy. The higher PSNR value further indicates that the image quality is excellent and is suitable for tasks that require high accuracy. Meanwhile, the response time is only 4.3 ms, which demonstrates excellent real-time processing capability, enabling the model to operate effectively in high-load environments [12]. The biomechanics simulation dataset and the running dataset excel in terms of error, image quality, and real-time performance, making them particularly suitable for applications requiring real-time feedback and high accuracy. The gait analysis dataset, on the other hand, shows moderate performance and is suitable for general applications. The EMG signal dataset may require further optimization during processing due to the high signal complexity, relatively large error in the model, and long response time.

4.3. Comparison experiment of different algorithms

In conducting the comparison experiments of different algorithms, a variety of common optimization methods were used for comparison in order to fully assess the performance of each algorithm in biomechanical nonlinear optimization problems. These methods include classical gradient descent, gradient descent with momentum (Momentum), Adam optimization algorithm, and adaptive gradient algorithm (Adagrad). The main goal of the experiments is to explore the differences in the performance of these algorithms in biomechanical nonlinear optimization problems and to provide a scientific basis for selecting the optimal algorithm. The performance of an optimization algorithm usually depends on key dimensions such as its convergence speed, prediction accuracy, and computational efficiency on a particular dataset [13].

Selection of an appropriate optimization algorithm is crucial for solving complex biomechanical system problems. The experiments are trained and tested using multiple datasets to simulate the diverse requirements of real-world applications, and focus on the key performance metrics of each algorithm, including mean square error (MSE), mean absolute error (MAE), and computation time (RT). Experimental results on the gait analysis dataset show that there are significant differences in the performance of different algorithms in terms of convergence speed and accuracy. The classical gradient descent method, despite its simplicity, performs slowly on nonlinear problems; the gradient descent method with momentum improves the convergence speed by introducing a momentum term; Adam's algorithm outperforms in most of the metrics by virtue of its mechanism of combining momentum and adaptive learning rate, while Adagrad's algorithm has some advantages in learning rate adjustment but may suffer from premature convergence. **Table 3** shows in detail the specific performance comparison of each algorithm on the gait analysis dataset, which provides intuitive data support and reliable reference basis for the selection of optimization algorithms.

Table 3. Experimental results of different algorithms comparison.

Optimization Algorithm	**MSE ($\times 10^{-3}$)**	**MAE ($\times 10^{-2}$)**	**Computation Time (ms)**	**Convergence Speed (Epoch)**
Gradient Descent	1.05	0.72	10.2	150
Momentum Gradient Descent	0.98	0.68	8.5	140
Adam Optimization	0.85	0.65	5.2	120
Adagrad	1.1	0.75	12	160

Table 3 shows the experimental results of four common optimization algorithms on the same dataset, including Gradient Descent, Momentum Gradient Descent, Adam Optimization Algorithm, and Adagrad. The performance of each algorithm is compared by metrics such as MSE, MAE, computation time, and convergence speed. From the perspective of the gradient vanishing and gradient explosion problems, the significant advantage of the Adam optimization algorithm lies in its ability to dynamically adjust the learning rate so that each parameter can be updated based on the first- and second-order momentum of its gradient, avoiding the problem of too small or too large gradients. This mechanism allows Adam to perform optimally in high-dimensional nonconvex problems, especially in biomechanical simulation datasets containing a large amount of noise, with MSEs and MAEs of 0.62×10^{-3} and 0.55×10^{-2}, respectively, a computation time of 6.8 ms, and a convergence rate of 130 Epochs. In addition, the Adam's time complexity is O(n)O(n)O(n), which is comparable to that of the leading momentum gradient descent method, but its space complexity is slightly higher at O(2n)O(2n)O(2n), which is due to its need to additionally store the estimates of the momentum and adaptive gradient.

In contrast, driven momentum gradient descent utilizes only the first-order momentum in parameter updating, which reduces the gradient oscillation problem to some extent, but still has a low convergence efficiency on complex datasets, especially in the multivariate-dependent running dataset, which has an MSE of 1.0×10^{-3} and a convergence rate of 170 Epochs. Classical gradient descent is constrained by a fixed learning rate, it is difficult to adapt to the complex dynamic changes of the objective function, and it tends to have slow convergence in the gradient vanishing region, especially in the high-dimensional biomechanical simulation data, which exhibits a long computation time (11.5 ms) and a high error (MSE of 0.75×10^{-3}). Although Adagrad avoids the initial gradient problem through adaptive learning rate, its learning rate gradually decreases to a very low value during long-term training, resulting in limited convergence speed, which is 190 Epoch on the novel motion dataset, much lower than that of Adam's 140 Epoch. In contrast, Adagrad performs relatively poorly, with an MSE of 1.1×10^{-3}, MAE of 0.75×10^{-2}, computation time of 12 ms, and convergence speed of 160 Epoch. Although Adagrad avoids some problems by adaptively adjusting the learning rate, it is prone to too small learning rate during long-term training, which makes the convergence speed of the algorithm limited. In addition, the higher computation time and slower convergence speed also make Adagrad less applicable in complex tasks [14].

Adam optimization algorithm has a clear advantage among these four optimization methods, being able to achieve a lower error level in a shorter period of time, with a faster convergence rate and higher adaptability. Driven gradient descent is the next best, showing a better balance, especially in terms of convergence speed and computation time. Gradient descent, although simple, is less efficient when dealing with complex data and is prone to falling into local optimality. Adagrad, on the other hand, shows a greater disadvantage after many iterations, especially in terms of convergence speed and computational efficiency, indicating that it is not suitable for large-scale or complex tasks. In practical applications, the Adam optimization algorithm is the most recommended choice, especially in deep learning and complex model training, providing superior performance. In addition, to further analyze the performance of the algorithm on other types of datasets, similar comparison experiments are conducted on the running dataset and the biomechanical simulation dataset, and the results are shown in **Table 4**.

Table 4. Comparison of algorithms on biomechanical simulation dataset.

Optimization Algorithm	Dataset	MSE ($\times 10^{-3}$)	MAE ($\times 10^{-2}$)	Computation Time (ms)	Convergence Speed (Epoch)
Gradient Descent	Running Data	1.1	0.78	15	180
Momentum Gradient Descent		1	0.74	12.5	170
Adam Optimization		0.9	0.66	9	140
Adagrad		1.15	0.8	18	190
Gradient Descent	Biomechanical Simulation Data	0.75	0.6	11.5	150
Momentum Gradient Descent		0.68	0.58	9.2	140
Adam Optimization		0.62	0.55	6.8	130

Based on the algorithm comparison results for the biomechanics simulation dataset in **Table 4**, the performance of different optimization algorithms on the running data and biomechanics simulation dataset can be analyzed in detail. For the running dataset, the gradient descent method performs with MSE of 1.1×10^{-3} and

MAE of 0.78×10^{-2}, with a computation time of 15 ms and a convergence rate of 180 Epochs. The driven gradient descent optimization algorithm performs slightly better in reducing the MSE and MAE, with 1.0×10^{-3} and 0.74×10^{-2}, while the computation time is reduced to 12.5 ms and the convergence speed is improved to 170 Epochs. The Adam optimization algorithm performs the best in all metrics, with a further reduction in MSE and MAE to 0.9×10^{-3} and 0.66×10^{-2}, with the shortest computation time of 9 ms and a better convergence speed than the previous two at 140 Epoch. In contrast, the Adagrad algorithm, while having a smaller MSE (1.15×10^{-3}), has a higher MAE of 0.8×10^{-2} and the time is the longest (18 ms) with a convergence rate of 190 Epoch, which is a poor performance.

For the biomechanical simulation dataset, the gradient descent method has an MSE of 0.75×10^{-3}, an MAE of 0.6×10^{-2}, a computation time of 11.5 ms, and a convergence speed of 150 Epoch. The driven gradient descent algorithm further improves the optimization with an MSE of 0.68×10^{-3} and MAE of 0.58×10^{-2}, with a reduced computation time of 9.2 ms and a convergence speed of 140 Epoch. The Adam optimization algorithm, on the other hand, performs the best in both metrics, with an MSE of 0.62×10^{-3}, an MAE of 0.55 × 10 The Adam optimization algorithm has the best performance on these two data sets, with MSE of 0.62×10^{-3} and MAE of 0.55×10^{-2}, and the shortest computation time of 6.8 ms and a convergence speed of 130 Epoch, which reflects its high efficiency on this type of dataset. The Adam optimization algorithm has the best performance on these two datasets in a combined manner, especially in the aspects of computation time and convergence speed, and it is suitable for the biomechanical simulation tasks that require fast training and efficient optimization [15]. The performance of driven gradient descent is also more balanced in terms of convergence speed and computational efficiency, which is a better compromise. The gradient descent method, although simple, is less efficient on such complex datasets, especially in terms of convergence speed and computational time.

4.4. Validation of model robustness and generalization ability

Model robustness and generalization ability are key performance metrics in complex biomechanical nonlinear optimization tasks. To deeply analyze the effect of noise, the study focuses on the interference of common noise types (e.g., Gaussian noise vs. salt-and-pepper noise) on the gradient computation, parameter update, and optimization results. Gaussian noise disturbs the gradient direction in a randomly distributed manner, causing the parameter update path to deviate from the optimal solution and thus slowing down the convergence process. In contrast, salt and pepper noise, due to its sparse and discrete nature, is prone to introduce extreme outliers in the input data, thus significantly increasing the volatility of the objective function and affecting the stability of the optimization process. To cope with the noise problem, noise reduction techniques such as wavelet transform and median filter are introduced, and combined with the adaptive noise modeling method to dynamically adjust the input feature weights, which effectively mitigates the negative impact of noise on the model performance. In addition, to address the challenge of the adaptability of optimization methods to the noise environment, comparison experiments of various optimization algorithms are conducted, including gradient descent, momentum method, Adam optimization algorithm and Adagrad.

The results show that the Adam optimization algorithm, by virtue of the adaptive learning rate adjustment mechanism and the momentum strategy, is able to achieve more stable gradient updating and faster convergence speed, and exhibits superior robustness under higher noise intensity. By systematically analyzing the model performance under noise interference and proposing corresponding optimization strategies, a new technical path is provided for solving the biomechanical nonlinear optimization problem in complex noise environments. The generalization ability of the model is also examined by comparing its performance on different datasets. These datasets include motion data with different pre-processing, EMG signal data, and mechanical response data from different organisms. **Table 5** shows the performance of the models on noisy datasets and their robustness changes compared to the original datasets.

Table 5. Robustness performance of the model on noisy datasets.

Noise Type	Dataset	MSE ($\times 10^{-3}$)	MAE ($\times 10^{-2}$)	Computation Time (ms)	Convergence Speed (Epoch)
No Noise	Running Data	0.78	0.62	4.7	120
Gaussian Noise (0.05)		1.03	0.72	6.5	130
Salt and Pepper Noise (0.1)		1.15	0.76	7.2	135
No Noise	Biomechanical Simulation Data	0.68	0.58	4.3	115
Gaussian Noise (0.05)		0.88	0.67	5.9	125
Salt and Pepper Noise (0.1)		1.02	0.71	6.3	130

Table 5 shows the robustness performance of different noise types on the model on the running dataset and the biomechanics simulation dataset. In the absence of noise, the model has an MSE of 0.78×10^{-3} and an MAE of 0.62×10^{-2} on the running dataset, with a computation time of 4.7 ms and a convergence rate of 120 Epoch, which is a better performance. While when Gaussian noise (0.05) is added, the MSE and MAE rise to 1.03×10^{-3} and 0.72×10^{-2} respectively, the computation time increases to 6.5 ms, and the speed of convergence slightly improves to 130 Epoch, which indicates that the Gaussian noise has a relatively small effect on the model, but it can still significantly increase the error and the computational burden. After adding salt and pepper noise (0.1), the MSE further increases to 1.15×10^{-3}, the MAE is 0.76×10^{-2}, the computation time increases to 7.2 ms, and the convergence speed is 135 Epoch, showing a more obvious performance degradation, which indicates that salt and pepper noise has a large negative impact on the model robustness.

For the biomechanical simulation dataset, the MSE of the model without noise is 0.68×10^{-3}, the MAE is 0.58×10^{-2}, the computation time is 4.3 ms, and the convergence speed is 115 Epoch. under the influence of Gaussian noise, the MSE increases to 0.88×10^{-3}, and the MAE is 0.58×10^{-2}, showing a more obvious performance degradation. $^{-3}$, the MAE is 0.67×10^{-2}, the computation time is slightly increased to 5.9 ms, and the convergence speed is 125 Epoch, which shows that the Gaussian noise has a certain effect on the model's performance, but still maintains a good robustness. With the addition of salt and pepper noise (0.1), the MSE and MAE increased to 1.02×10^{-3} and 0.71×10^{-2}, respectively, with a computation time of 6.3

ms and a convergence rate of 130 Epoch. Compared with the running dataset, the performance degradation of the biomechanical simulation dataset in the presence of noise interference was relatively compared to the running dataset, indicating that the model is robust in handling such data. Although the addition of noise significantly affects the performance of the model, especially the salt and pepper noise, the model maintains a relatively stable performance under Gaussian noise. In addition, the generalization ability is validated by applying the training model to an unseen test dataset. **Table 6** shows the performance of the model's generalization ability on different datasets, focusing on the model's prediction accuracy under diverse tasks.

Table 6. Model's generalization ability performance on different datasets.

Dataset Name	MSE ($\times 10^{-3}$)	MAE ($\times 10^{-2}$)	Computation Time (ms)	Convergence Speed (Epoch)
Gait Analysis Data	0.85	0.65	5.2	120
EMG Signal Data	1.02	0.78	6.8	130
Running Data	0.78	0.62	4.7	115
Biomechanical Simulation Data	0.68	0.58	4.3	110
New Sports Dataset	1.1	0.8	7.5	140

Table 6 demonstrates the model's generalization ability performance on different datasets. The datasets include gait analysis data, EMG signal data, running data, biomechanical simulation data, and novel exercise datasets, and the MSE, MAE, computation time, and convergence speed performances of each dataset reflect the adaptability and efficiency of the model under different tasks. The gait analysis dataset has an MSE of 0.85×10^{-3}, an MAE of 0.65×10^{-2}, a computation time of 5.2 ms, and a convergence speed of 120 Epoch, which indicates that this dataset performs moderately well in terms of the model's generalization ability during training. In contrast, the MSE and MAE of the EMG signal dataset were 1.02×10^{-3} and 0.78×10^{-2}, respectively, and the computation time increased to 6.8 ms with a convergence speed of 130 Epoch, showing that this dataset has a higher complexity, a larger model training time and error, and a relatively poorer generalization ability.

The running dataset has an MSE of 0.78×10^{-3} and an MAE of 0.62×10^{-2}, with a computation time of 4.7 ms and a convergence speed of 115 Epoch, demonstrating a relatively good generalization ability, and the model is able to quickly adapt to this dataset and achieve a low error. The biomechanical simulation dataset performs even better, with an MSE of 0.68×10^{-3}, an MAE of 0.58×10^{-2}, a computation time of 4.3 ms, and a convergence speed of 110 Epoch, showing that this dataset is relatively simple, and that the model is able to converge in a relatively short time, demonstrating a strong generalization ability. The new sports dataset has an MSE of 1.1×10^{-3}, an MAE of 0.8×10^{-2}, a computation time of 7.5 ms, and a convergence speed of 140 Epoch. The higher error and longer training time of this dataset show that the model may need more tuning and training when faced with the new dataset, and has a weaker generalization ability is weak. The model performs best on the running data and biomechanical simulation dataset, showing strong generalization ability, while the model performs relatively poorly on the EMG signal data and the novel exercise dataset, which may require further optimization.

5. Conclusion

Although the application of deep learning to nonlinear optimization problems in biomechanics provides new perspectives and effective tools for modeling and solving complex systems, especially when dealing with high-dimensional and nonlinear features, it shows significant advantages. However, certain limitations still exist. When dealing with extreme noise and large-scale datasets, the robustness and computational efficiency of the model need to be further improved. The current method exhibits a significant increase in error (MSE increases from 0.68×10^{-3} in the noiseless case to 1.02×10^{-3}) with high salt and pepper noise intensity, indicating that the sensitivity of the existing model to anomalous data points still needs to be improved. In the future, the stability of the model in high-noise environments can be improved by introducing more efficient noise reduction algorithms, such as dynamic noise modeling or adversarial data enhancement techniques. In addition, in the face of rapidly growing large-scale data, the time complexity and memory requirement of optimization algorithms may become a bottleneck, and multi-model integration and hybrid optimization strategies may be able to provide a solution. By combining the advantages of different optimization algorithms (e.g., Adam, RMSProp, and momentum method) and constructing a collaborative optimization framework, it is expected to improve the convergence speed and accuracy of the model, and the application scope is mainly focused on simulation and motion data analysis.

Future research can consider designing hybrid architectures with higher scalability and exploring multimodal data fusion methods to enhance model adaptability. Meanwhile, for the performance bottlenecks in extreme scenarios, the combination of adaptive optimization algorithms and reinforcement learning can be developed to further improve the robustness and efficiency of the model, and provide more comprehensive technical support for the accurate modeling and practical deployment of biomechanical systems.

Conflict of interest: The author declares no conflict of interest.

References

1. Esposito M, Wannop JW, Stefanyshyn D. The influence on a non-linear bending stiffness profile on sprint acceleration performance and MTP biomechanics. Footwear Science. 2023; 15(sup1). doi: 10.1080/19424280.2023.2199261
2. Mosconi D, Escalante FM, Terra MH, et al. Biomechanical inverted pendulum: a model for initial testing of interaction controls. IFAC-PapersOnLine. 2023; 56(2): 7442-7447. doi: 10.1016/j.ifacol.2023.10.627
3. Zhang E, Spronck B, Humphrey JD, et al. G2Φnet: Relating genotype and biomechanical phenotype of tissues with deep learning. PLOS Computational Biology. 2022; 18(10): e1010660. doi: 10.1371/journal.pcbi.1010660
4. Lohr MJ, Sugerman GP, Kakaletsis S, et al. An introduction to the Ogden model in biomechanics: benefits, implementation tools and limitations. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences. 2022; 380(2234). doi: 10.1098/rsta.2021.0365
5. Fahmy MA. A Computational Model for Nonlinear Biomechanics Problems of FGA Biological Soft Tissues. Applied Sciences. 2022; 12(14): 7174. doi: 10.3390/app12147174
6. Walo M, Łagód G, Majerek D, et al. Bioindication Studies of Surface Water Supported by Automatic Image Analysis Using Deep Learning Neural Network—yclotella Case Study. Journal of Ecological Engineering. 2024; 25(12): 366-375. doi: 10.12911/22998993/194179
7. Rajiv A, Kumari A, Deo Tripathi A, et al. Retraction Note: Multi photon micro material analysis based on Raman

spectroscopy biosensor for cancer detection using biomarker with deep learning techniques. Optical and Quantum Electronics. 2024; 56(10). doi: 10.1007/s11082-024-07731-7
8. Jahangiri M, Nazemi A. A cooperative neural dynamic model for solving general convex nonlinear optimization problems with fuzzy parameters and an application in manufacturing systems. International Journal of Adaptive Control and Signal Processing. 2024; 38(6): 2272-2300. doi: 10.1002/acs.3804
9. Atlas A, Attmani J, Karami F, et al. Optimal control problem with nonlinear fractional system constraint applied to image restoration. Mathematical Methods in the Applied Sciences. 2024; 47(9): 7714-7741. doi: 10.1002/mma.9998
10. Fang Y, Na S, Mahoney MW, et al. Fully Stochastic Trust-Region Sequential Quadratic Programming for Equality-Constrained Optimization Problems. SIAM Journal on Optimization. 2024; 34(2): 2007-2037. doi: 10.1137/22m1537862
11. Arzani A, Wang JX, Sacks MS, et al. Machine Learning for Cardiovascular Biomechanics Modeling: Challenges and Beyond. Annals of Biomedical Engineering. 2022; 50(6): 615-627. doi: 10.1007/s10439-022-02967-4
12. Cai L, Ren L, Wang Y, et al. Surrogate models based on machine learning methods for parameter estimation of left ventricular myocardium. Royal Society Open Science. 2021; 8(1): 201121. doi: 10.1098/rsos.201121
13. Hashemi A, Jang J, Beheshti J. A Machine Learning-Based Surrogate Finite Element Model for Estimating Dynamic Response of Mechanical Systems. IEEE Access. 2023; 11: 54509-54525. doi: 10.1109/access.2023.3282453
14. Sharma B, Pillai BM, Suthakorn J. Biomechanical Trajectory Optimization of Human Sit-to-Stand Motion With Stochastic Motion Planning Framework. IEEE Transactions on Medical Robotics and Bionics. 2022; 4(4): 1022-1033. doi: 10.1109/tmrb.2022.3205509
15. Lampen N, Kim D, Fang X, et al. Deep learning for biomechanical modeling of facial tissue deformation in orthognathic surgical planning. International Journal of Computer Assisted Radiology and Surgery. 2022; 17(5): 945-952. doi: 10.1007/s11548-022-02596-1