Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 7792958, 9 pages
https://doi.org/10.1155/2022/7792958

Research Article

Machine Leaning-Based Optimization Algorithm for Myocardial Injury under High-Intensity Exercise in Track and Field Athletes

Guanguan Li

Department of Public Sports, Huanghe Jiaotong University, Jiaozuo, Henan 454950, China

Correspondence should be addressed to Guanguan Li; 2009040298@zjtu.edu.cn

Received 15 January 2022; Revised 8 April 2022; Accepted 19 April 2022; Published 9 May 2022

Academic Editor: Shahid Mumtaz

In order to train at high-intensity, athletics can again cause varying degrees of myocardial damage. Evaluating the balance between exercise myocardial injury and exercise intensity should actively prevent myocardial injury caused by high-intensity athletic training. In this paper, an intelligent optimization algorithm is used to investigate the degree of myocardial injury. The basic idea is to define the measured data and the output of the numerical model as an objective function of the structural parameters, to obtain the structural parameters by finding ways to continuously optimize the objective function to be close to the observed values, and to identify the injury based on the changes in these parameters before and after myocardial injury. The objective function can be defined in various ways, and the myocardial injury optimization algorithm can be chosen. In order to obtain the best computational results, numerical simulations of damage identification are performed using the objective function and three machine learning-based optimization algorithms. The computational results show that the combination of the objective function and the machine learning algorithms provides good accuracy and computational speed in identifying myocardial injury.

1. Introduction

Sports injury refers to a variety of physical injuries that occur during athletes' sports activities due to various improper sports activities. In the process of sports, sports injuries are easily ignored by athletes and coaches. In recent years, it has become an important task in modern sports to improve athletic performance and to reduce myocardial injury in athletes, which is the goal of modern sports. The combination of scientific training and physical fitness assessment is an important prerequisite for achieving the development goals of modern athletics, and in this context, how to accurately analyze the relationship between intensity training and myocardial injury in track and field athletes has become a major problem in this field. The modeling of the relationship between intensity training and myocardial injury in track and field athletes is an effective way to solve these problems and has attracted the attention of many researchers. Intensity training is necessary for competitive sports, but when the load exceeds the physiological limits of athletes, it can cause myocardial ischemia and hypoxia, which can lead to myocardial injury. The heart is the central command of the human body, and damage to the heart will inevitably reduce the athletic ability of the athlete and affect the training effect [1]. The contraction and diastole of the heart wall plays an important role in exercise. Long-term, systematic, load-appropriate exercise training can lead to adaptations in the morphology of the athlete's heart and increase the pumping function of the heart. A growing body of research today indicates that exercise training is not as effective as more intense stimulation. Intense training causes a disruption of endogenous protective substances in the myocardium. In addition, high-intensity training can cause damage to the heart muscle. This suggests that intense training can cause some damage to the myocardium, resulting in relatively localized ischemic and hypoxic myocardial changes in myocardial fibers [2–7].

Considering that myocardial damage recognition can be regarded as an optimization problem, intelligent optimization techniques have received attention in recent years, like Monarch Butterfly Optimization (MBO) [8], Slime Mould Algorithm (SMA) [9], Hunger Games Search (HGS)

[10], Runge Kutta method (RUN) [11], genetic algorithm (GA) [12], differential evolution algorithm (DE) [13], particle swarm algorithm (PSO) [14], and other optimization algorithms. Optimization technology is a mathematically based application technique for solving various engineering problems with optimal solutions. In summary, the optimization problem is to construct a suitable objective function such that the solution of this objective function taken to the extreme is what you require; let the computer discover the hidden relationships between the data instead of the human. The reason for using a computer is that the amount of data is so large that it far exceeds the processing power of the human brain. The goal is to find a way to get a solution that takes the extreme value of this objective function. Although their structures are very different, they all have in common that they have a large set of parameters just waiting for you to feed data for it to learn. The control parameters of the relevant algorithms are shown in Table 1 [15–24].

Some scholars proposed a lot of different machine learning-based optimization approaches to implement some optimization problems [25–30]. KAdam and Mahajan [31] tried to optimize the cutting temperature prediction model using GA to optimize the objective function, and the outcomes acquired through experimental test are likewise similar to the outcomes of GA. El et al. [32] compared the optimization models between PSO and GA. The results show that, for the majority of the proposed strategies, PSO is more efficient than the GA, whereas the latter is generally much more used for the optimization of detailed kinetic mechanisms. Zhou et al. [33] proposed an adaptive hierarchical update particle swarm optimization (AHPSO) algorithm, and one real-world optimization problem is employed to evaluate the AHPSO against eight typical PSO variants. Balasubramani et al. [34] proposed a PSO algorithm-based artificial neural network (ANN) model to optimize the adsorption process parameters. Zhong and Cheng [35] proposed an elite-guided hierarchical differential evolution algorithm. Finally, for the sake of evaluating the performance of the proposed method, sensitivity analysis to the size of elite individuals, efficiency analysis of the control parameters adaptive strategy, and comparisons with baseline methods on 29 universal benchmark function in terms of convergence accuracy and convergence speed have been taken out. All the obtained results show that the proposed model has excellent optimization performance. Tharwat and Hassanien [36] proposed a novel chaotic antlion optimization (CALO) algorithm to optimize the parameters of support vector machine (SVM) classifier, so that the classification error can be reduced. The experimental results proved that the proposed algorithm is capable of finding the optimal values of the SVM parameters and avoids the local optima problem. The results also demonstrated lower classification error rates compared with baseline methods. Although all of these algorithms have positive examples to support their effectiveness, the general applicability of these methods cannot be demonstrated only by the analysis and validation of a specific example. To compare the damage recognition effects of various intelligent optimization algorithms, this paper first defines four objective functions for structural damage recognition, then describes three well-known intelligent optimization algorithms, finally conducts numerical simulations of myocardial damage recognition using three typical types of structural models, and summarizes the comparison of the damage recognition effects of each objective function and intelligent optimization algorithm.

Table 1: Control parameters.

Optimization algorithm	Parameter setting
GA algorithm	NP = 30; Pc = 0.7; Pm = 0.08; search Scope = [0,1]
PSO algorithm	NP = 30; $w = 0.8$; $c^1 = c_2 = 2$; search Scope = [0,1]; speed range = [0,1]
DE algorithm	NP = 30; $F = 0.5$; $Cr = 0.9$; search Scope = [0,1]

*Genetic algorithm (GA), differential evolution algorithm (DE), and particle swarm algorithm (PSO).

2. Finite Element Model of the Optimization Algorithm

The identification of the degree of myocardial injury needs to be achieved by continuous modification of the finite element model, which can be equivalent to the treatment of a nonlinear optimization problem, and therefore, the optimization module and the related theory in the finite element software will be introduced in detail in the following sections. The mathematical model of the optimization problem is

$$\min f(x), \quad x = (x_1, x_2, \ldots, x_n)^T, \tag{1}$$

$$\begin{aligned} \text{S.T. } g_i(x) &\le g_i, \quad i = 1, 2, \ldots, m_1, \\ h_i(x) &\ge h_i, \quad i = 1, 2, \ldots m_2, \\ \underline{w_i} \le w_i(x) &\le \overline{w_i}, \quad i = 1, 2, \ldots, m_3, \\ \underline{x_i} \le x_i &\le \overline{x}_i, \quad i = 1, 2, \ldots, n, \end{aligned} \tag{2}$$

where f(x) is the objective function and g_i, h_i, w_i are state variables with upper, lower, and double limits, respectively. X=x1, x2,. . ., xn are design variables and n is the number of design variables.

The design variables that satisfy (2) are called feasible solutions, while those that satisfy (1) are called optimal solutions.

Two optimization methods are provided in the finite element software, namely, the zero-order method and the first-order method. The zero-order method uses only the calculated values of the objective function and state variables, while the first-order method requires the calculation of the first-order inverse values of the objective function and state variables.

2.1. Zero-Order Method.

Basic concept of the zero-order method is to approximate the complex nonlinear optimization problem as a quadratic programming problem. For example, the objective function can be approximated by the following equation:

$$f(x) = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n}\sum_{i=1}^{n} b_{ij} x_i x_j. \tag{3}$$

The coefficients in the above equation can generally be determined by the least squares calculation, i. e.,

$$E^2 = \sum_{j=1}^{n_d} \phi_j \left(f_j - \overset{\wedge}{f_j} \right)^2, \tag{4}$$

where φj is the weight obtained at the ith design point and n_d is the total number of design points, which must meet $\mathrm{n_d} \geq \mathrm{n} + 2$.

There are five methods of taking the weight coefficients in (4):

(1) The objective function method, even if the smaller the objective function value, the greater the value of the weight at the design point

(2) The optimal design point method, that is, the closer to the optimal design point, the greater the value of the weight at the design point

(3) Feasible design point method, i.e., the weight of the feasible design point is greater than the weight of the infeasible design point

(4) Combination method, that is, the combination of the above methods

(5) Uniform method, i.e., all the weights are taken as 1

The optimization problem described by equation (1) is simplified as

$$\begin{aligned} \min \hat{f}(x), \quad & x = (x_1, x_2, \ldots, x_n)^T, \\ \text{S.T. } g_i(x) g_i, \quad & i = 1, 2, \ldots, m_1, \\ h_i(x) \geq h_i, \quad & i = 1, 2, \ldots m_2, \\ \underline{w_i} \leq w_i(x) \leq \overline{w_i}, \quad & i = 1, 2, \ldots, m_3, \\ \underline{x_i} \leq x_i \leq \overline{x}_i, \quad & i = 1, 2, \ldots, n. \end{aligned} \tag{5}$$

This constrained quadratic programming problem can be transformed into an unconstrained problem by using a penalty function, i.e.,

$$\begin{aligned} &\operatorname{Min} F(x, p_k) \\ &= \min \left[\hat{f}(x) + f_0 p_k \left(\sum_{i=1}^{n} X(x_i) + \sum_{i=1}^{m_1} G(x_i) + \sum_{i=1}^{m_2} W(x_i) \right) \right], \end{aligned} \tag{6}$$

where f_0 is the target reference function; p_k is the response surface parameter, whose value increases with the number of iterations; and X is the design variable to obtain penalty functions.

$$X(x_i) = \begin{cases} \dfrac{c_1 + c_2}{\left(\bar{x} - x_i\right)}, & \in x_i < \bar{x}, \\[2ex] \dfrac{c_3 + c_4}{\left(x_i - \bar{x}\right)}, & xi \geq \bar{x}. \end{cases} \tag{7}$$

At each iteration, the design variables take the values:

$$xj + 1 = x_b + C\left(x_b - x_j\right), \tag{8}$$

where xb is the most optimal design variable among the current design variables. C is the constant, being 0–1.

2.2. First-Order Method. The first-order method can directly transform the constrained optimization problem into an unconstrained optimization problem, but it needs to be under the action of penalty function as follows:

$$\begin{aligned} Q(x, q) = {} & \frac{f(x)}{f_0} + \sum_{i=1}^{n} P_x(x_i) \\ & + q\left(\sum_{i=1}^{m_1} P_G(gi) + \sum_{i=1}^{m_2} P_H(h_i) + \sum_{i=1}^{m_3} P_W(w_i) \right), \end{aligned} \tag{9}$$

where Q is dimensionless, unconstrained objective function; P_X is the design of variable penalty functions, using the out-point method. P_G, P_H, and P_W are state variable penalty functions, using the interior point method.

The difference between the first-order method and the zero-step method is that the direction of advance and the step size are determined in the iterative calculation, and the design variables are taken in the iteration.

$$x_{j+1} = x_j + s_j d_j, \tag{10}$$

where d_j is the (j+1)th iteration yielding the forward method and s_j is the (j+1)th iteration getting the previous progress length.

In the finite element method, the forward direction is determined by the conjugate gradient method with the iterative formula.

Here, d_j, $\mathrm{d}_{j\text{-}1}$ denote the jth and (j–1)th iteration forward directions, respectively.

$\nabla\mathrm{Q}(x_j, \mathrm{q})$ and $\nabla\mathrm{Q}(\mathrm{x}_{j\text{-}1}, \mathrm{q})$ denote the jth and j-1th iteration objective function gradients, respectively.

After determining the forward direction, the forward progress length is determined by the following equation:

$$\min Q\left(x_j + s_j d_j, q\right). \tag{11}$$

The above is the process of objective function modification and optimization of finite element model. The optimization algorithm can quickly determine the optimal objective function.

3. Intelligent Algorithms

3.1. Optimization Algorithm. GA is a global optimization algorithm based on Mendelian genetics and Darwinian evolution to simulate the biological evolution process in nature. The GA algorithm simulates the biological evolutionary process by abstracting the biological population as an effective approach to a set of optimization problems, referred to as the population, and each effective solution in the population is called an individual, mapping the natural environment as the solution space, using "survival of the

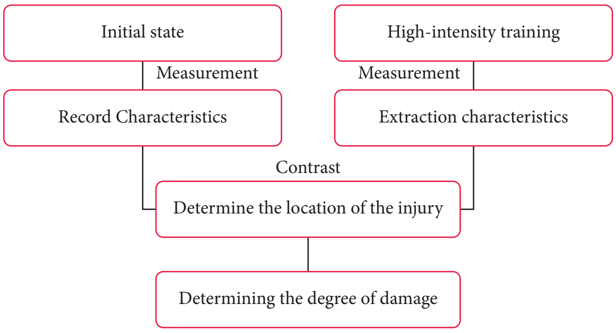
Initial state
High-intensity training
Measurement
Measurement
Record Characteristics
Extraction characteristics
Contrast
Determine the location of the injury
Determining the degree of damage

Figure 1: Computational flow of the optimization algorithm.

fittest" as the selection mechanism for the optimal individual and using genetics and variation for the self-adjustment of the individual. The advantages of the GA algorithm in solving NP-complete problems, nonlinear, multipeak function optimization, and multiobjective function optimization has attracted a lot of attention from researchers in various fields. After decades of development, GA algorithms have been deeply involved in various scientific research and engineering applications and have become an important branch of intelligent optimization algorithms. The particle swarm optimization (PSO) algorithm is a heuristic optimization algorithm, inspired by the behavior of a flock of birds during foraging. For this process, the PSO algorithm abstracts each bird as a possible optimal solution to the optimization problem, called a "particle," and the "particle." The PSO algorithm uses real number coding for easy understanding and implementation, and it is computationally efficient and can be effectively used to solve global multiobjective optimization problems. The basic idea of this algorithm is similar to that of genetic algorithm, which searches for the optimal solution by simulating the natural biological evolution mechanism of "survival of the fittest." In the first International Competition on Evolutionary Optimization (ICEO) held by IEEE, the DE algorithm achieved the 3rd place in terms of computational speed, but the top two algorithms are deterministic computational methods and have limited applications, so the DE Storn and price analyzed the computation of nine standard functions and showed that the iteration efficiency and robustness of the DE algorithm outperformed the annealed Nelder and Mead strategy (ANM), the adaptive simulated annealing algorithm (ASA), and the annealed Nelder and Mead algorithm (AMA). The DE algorithm has the features of simple principle, real number encoding, easy implementation, parallel computation, etc. It has better efficiency and robustness in solving the minimization of nonlinear and nondifferentiable continuous functions, which attracts wide attention and is applied in various fields.

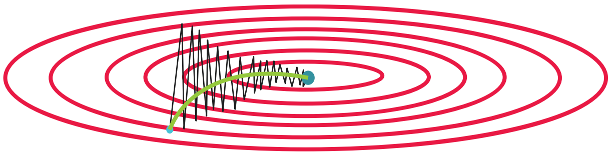

Figure 2: Schematic diagram of the optimized path.

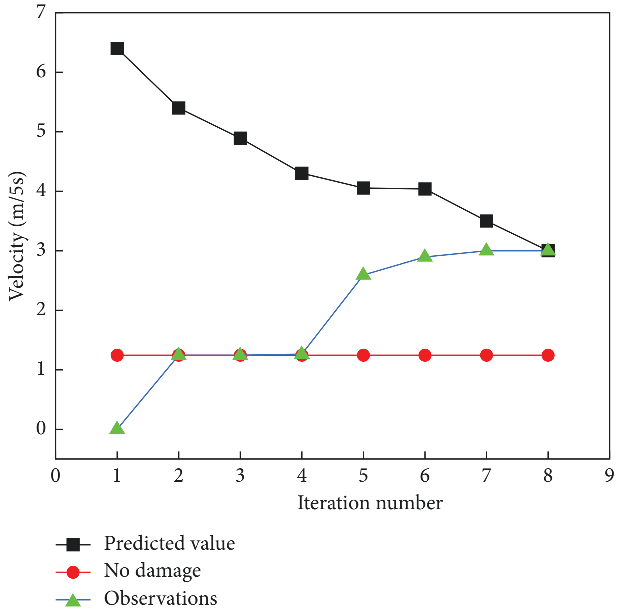

Figure 3: Damage-free velocity and observation and prediction iterations.

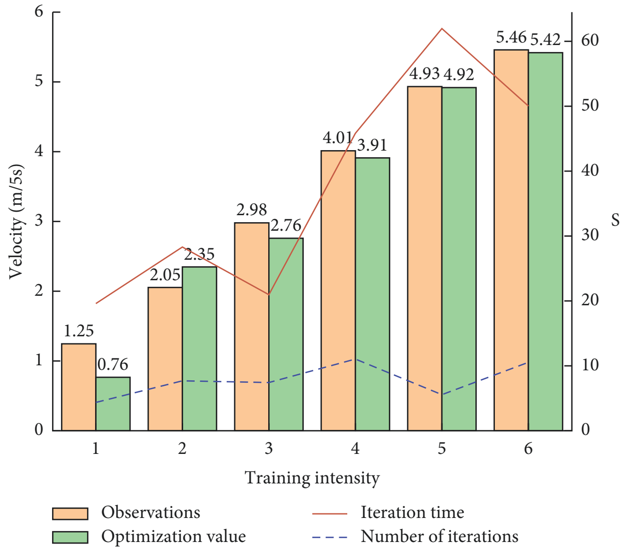

Figure 4: Observed and statistical values at different intensities in terms of iterations.

3.2. Trial-and-Error Judgment Algorithm. The intelligent trial-and-error learning algorithm consists of two main steps: (1) the creation step of the behavior-performance mapping table; and (2) the adaptation step. The creation of the behavior-performance mapping table is done by the multidimensional archive of phenotypic Elites algorithm. The MAP-Elites algorithm is based on the evolutionary algorithm proposed by core process is also the selection and mutation of elite solutions. The adaptation process is also done by the new algorithm. This algorithm is called the map-based Bayesian

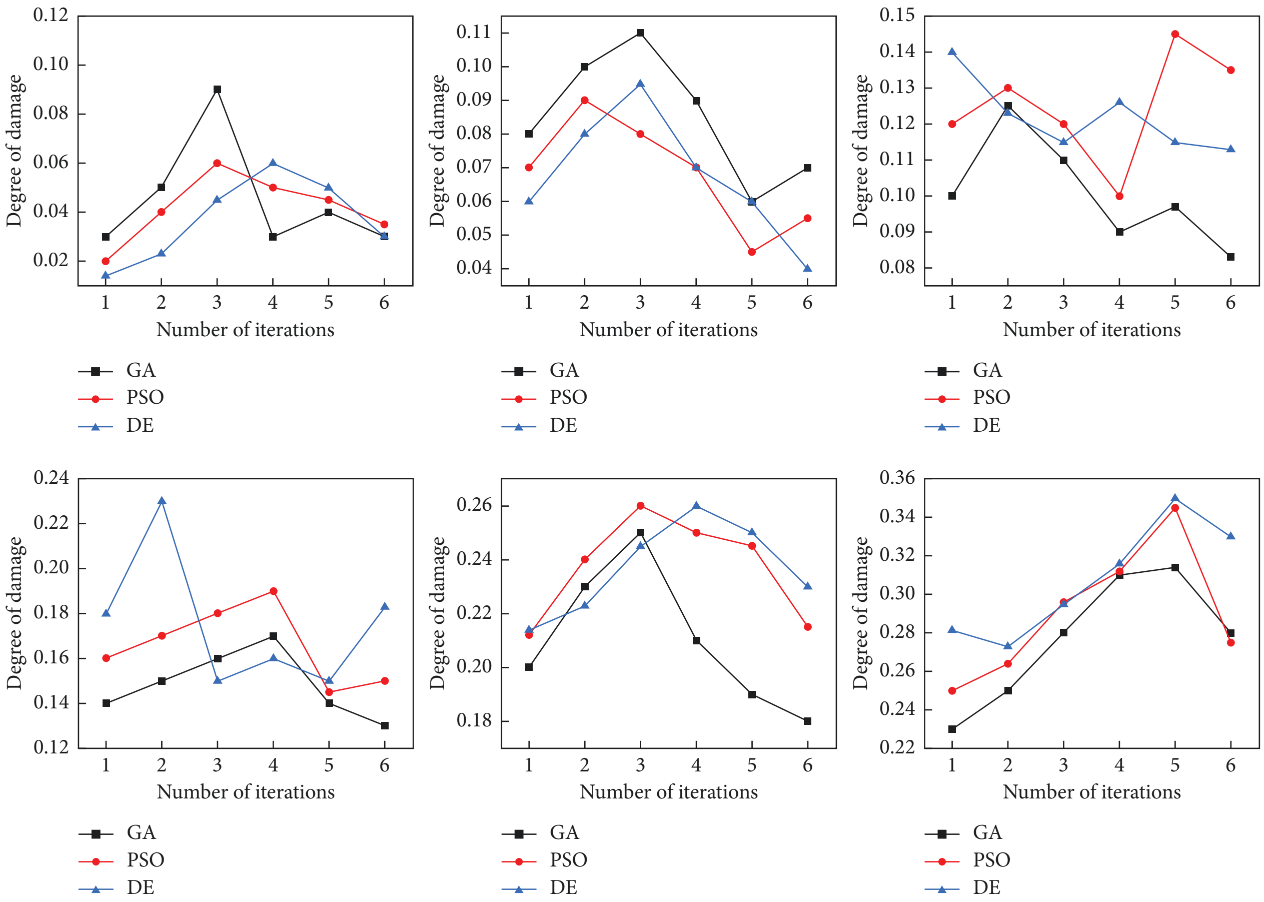

Figure 5: Damage at different strengths for each optimization algorithm.

optimization algorithm (M-BOA, map-based Bayesian optimization algorithm). The M-BOA algorithm is based on the proposed Bayesian optimization algorithm, with the difference that M-BOA starts from a map table and incorporates the information from the mapping table into the Bayesian optimization process. The entire process is computationally presented as shown in Figure 1.

4. Numerical Calculation Examples

The objective function proposed by the minimization optimization algorithm is used to solve to derive the location and degree and to compare the damage identification effect under different objective functions, in addition to considering 1% uncertainty factor. The heartbeat motion model is used as the numerical simulation object, and the data of the inherent frequency in the vertical direction of the structure are taken for state identification. There are many methods in the optimization process, as shown in Figure 2.

However, the most important thing is that we look for the most suitable method to perform the optimization process. Based on this, we tried three optimization algorithms for comparison to understand the optimization, iteration, and accuracy of the myocardial injury process.

We first selected a particular myocardial state for validation. Generally, we guess that myocardial injury occurs or tends to occur with increasing exercise intensity, and here, the exercise speed is used to represent the exercise intensity. In this paper, the model of myocardial injury at different exercise intensities created in the simulation tool V-REP is used for validation. Then, we first select the velocity as the evaluation of the validated injured myocardium, while we assume that it will appear in a certain state after the injury. From the qualitative analysis, the effect of exercise speed on myocardial injury is relatively large, as shown in Figure 3.

This is a prediction process for damage, shown in Figure 3 from the starting moment of performing the adaptation process. The horizontal coordinate is the number of iterations. The lower line represents the change in the maximum observed value after each select-observation-update iteration, and it is known that the observed performance value increases after the number of iterations. The upper line shows the maximum predicted performance value given in the behavior-performance mapping table after each selection-observation-update iteration. It can be known that the maximum predicted value decreases with the increase of the number of iterations because when a steep drop in performance is monitored, the observed value of this steep drop in performance is updated by the update process of the Gaussian model to make a corresponding decrease in other behaviors around its corresponding behavior.

As can be seen from Figure 3, this damage process found the target compensation behavior after only 7 iterations. 1.255 (m/5s) in the absence of injury and 6.442

(m/5s) in the myocardial injury velocity. The first three iterations of the damage recovery process found the best velocity without damage, and the last compensation behavior found resulted in a myocardial damage travel velocity of 3.022 (m/5s), a 42% improvement in computational efficiency.

In the above process, it has been demonstrated that the optimization algorithm can help to determine the statistical or observed values that the optimization algorithm can obtain with. The following can be further understood by the motion algorithm at different motion intensities as shown in Figure 4.

The change of the number of iterations is not very obvious with the increase of the training intensity, but the computation time is increased relatively obviously.

By verifying the effectiveness of the objective function and intelligent optimization algorithm above in structural damage identification, the damage models at different intensities were selected to use the velocity of motion as input, and the calculated damage states were used as measurement data, and all the information of the model before the damage was known, and the objective functions were optimized using the GA algorithm, PSO algorithm, and DE algorithm to identify the damage degree DK of each layer, and the damage degree DK was defined is the relative change of stiffness or elastic modulus before and after the damage; DK takes the value of [0,1], DK = 0 means no damage, and DK = 1 means complete damage. Table 1 shows the parameter settings of various optimization algorithms, and the best identification result is taken as the final identification result of the optimization algorithm in order to compare various identification results fairly under the same calculation scale as shown in Figure 5.

The comparison of the above optimization algorithms shows that each optimization algorithm reaches convergence within a certain number of iteration steps. The average number of convergence iterations under each algorithm is shown in Figure 6.

The iterative efficiency diagram provides an understanding of the iterative efficiency of each algorithm, and further, we need to understand the iterative computational accuracy of each algorithm, as shown in Figure 7.

The optimized model can further understand the circumferential damage pattern of myocardium by exercise intensity as shown in Figure 8.

Figure 8 shows the extent of myocardial damage in each direction at a given moment.

It is clearly visible that the degree of damage varies around the entire myocardium. The beating time of the heart muscle was monitored several times at different fluctuation frequencies, as shown in Figure 9.

Figure 9 shows that the heartbeat frequency is consistent for multiple monitoring of the same state, while the peak myocardial damage is reached at a frequency of 10. Therefore, it is necessary to optimize the algorithm to always remember the injury frequency to prevent the injury during the usual training process. And, the algorithm is optimized to keep in the low injury moment.

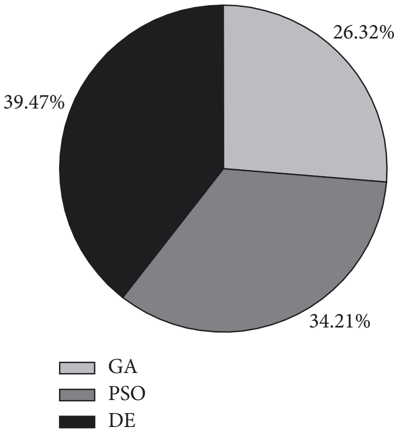

Figure 6: Comparison of iteration efficiency of different optimization algorithms.

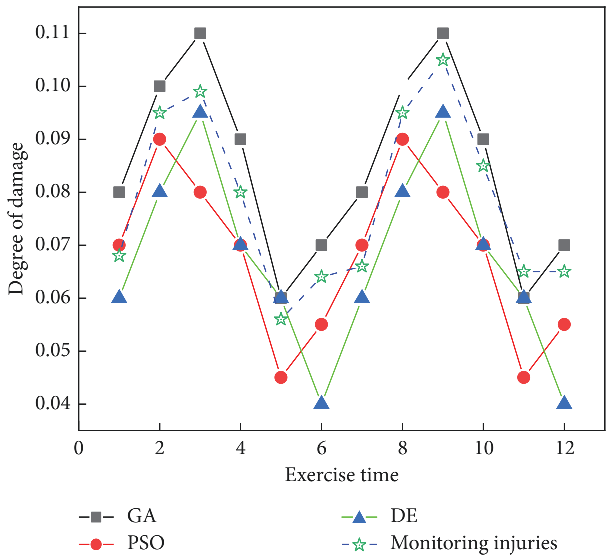

Figure 7: Comparison of calculation and monitoring of different algorithms.

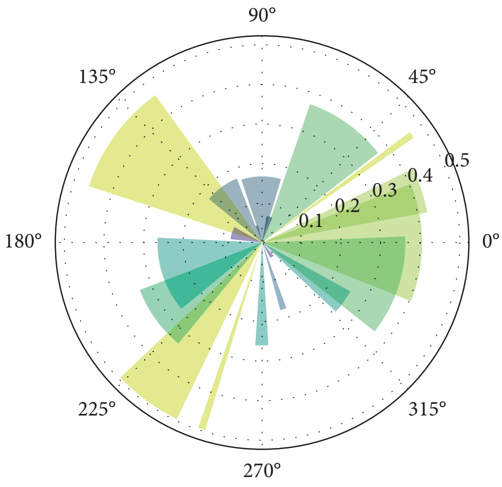

Figure 8: Extent of perimyocardial injury.

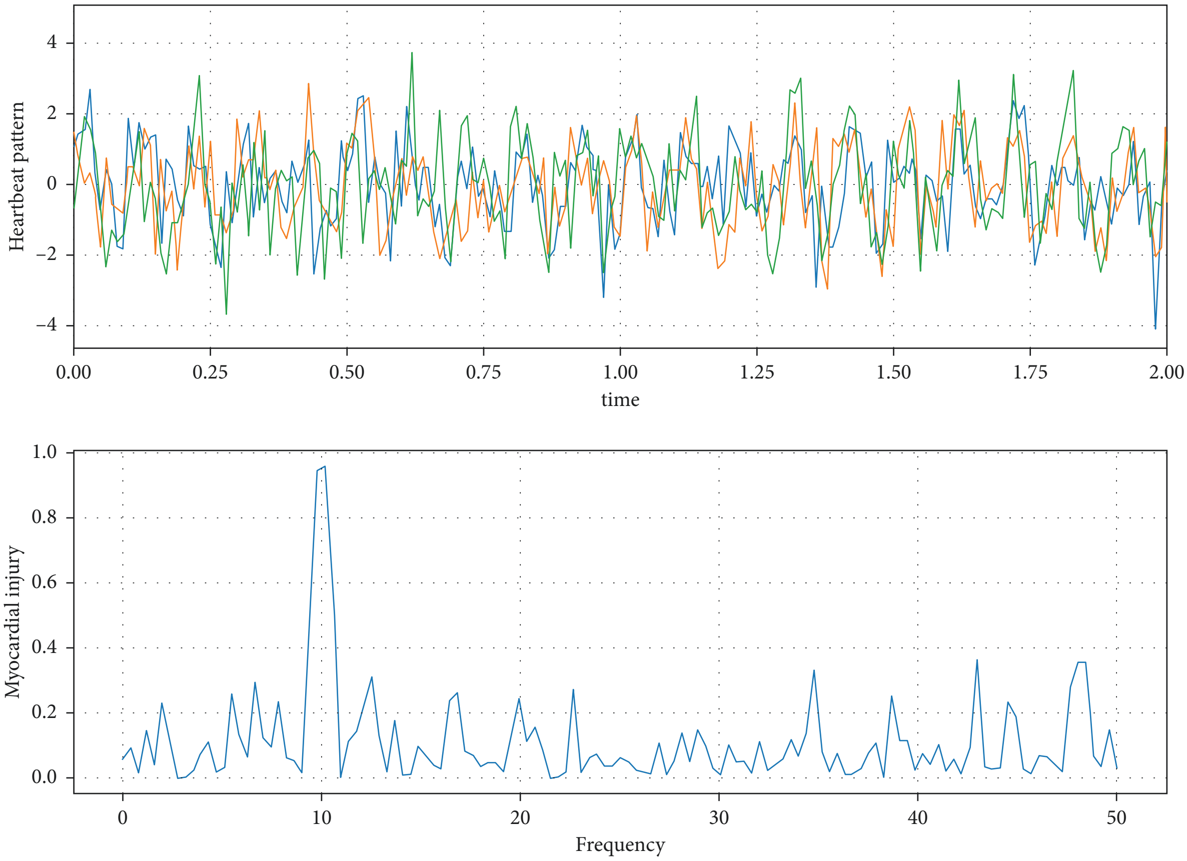

Figure 9: Change in frequency and myocardial injury under multiple monitoring.

5. Summary and Recommendations

By studying the characteristics and extent of myocardial injury under high-intensity sports training, combining clinical experimental follow-up and physiological index test methods for myocardial performance testing and injury assessment, the prevention and treatment of myocardial injury is achieved, and the physical health of athletes is safeguarded. The results of the changes of myocardial physiological index parameters under high-intensity sports training exercise are analyzed, and the significant difference characteristics of heart rate beating of athletes with high-intensity sports training are used as test indexes to analyze the functional indexes of myocardial injury under prolonged aerobic energy supply, and the parametric system structure of myocardial function of athletes under high-intensity sports training is obtained. The best damage identification method was selected from various objective functions and optimization algorithms, and the acceleration time response, frequency, vibration, and flexibility matrices were defined as the objective functions. The results show that the acceleration time response is more suitable as the objective function than the frequency, vibration type, and flexibility matrix. Using the DE algorithm shows better search ability than using GA algorithm or PSO algorithm; the combination of DE algorithm and the objective function based on the acceleration time response has the best accuracy force for damage identification.

The analysis of the conclusions enables to give relevant recommendations mainly in terms of not sustaining high-intensity training. In short, the athletic sports workers and sports medicine workers should pay enough attention to the training of athletes because scientific training is an important method for athletes and coaches to reduce sports injuries. Thus, through reasonable training methods and scientific sports medicine guidance, the quality of training and health of athletes can be doubly guaranteed.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares no conflicts of interest.

Acknowledgments

This work was supported by the 2021 University-Level Curriculum Ideological and Political Education Teaching Reform Research Project key project“Analysis on the Integration of Education and College Sports Team Training under the concept of “Curriculum Ideology and Politics” (number: HHJTXY-2021szzd04).

References

[1] V. Jahmunah, E. Y. K. Ng, T. R. San, and U. R. Acharya, "Automated detection of coronary artery disease, myocardial infarction and congestive heart failure using GaborCNN model with ECG signals," *Computers in Biology and Medicine*, vol. 134, Article ID 104457, 2021.

[2] M. M. J. Teo, F. H. Chang, and Y. N. Lin, "Sports injuries and risk factors for table tennis among nonprofessional collegiate athletes: a cross-sectional study," *Medicine*, vol. 100, no. 50, 2021.

[3] M. Jarraya, C. A. Blauwet, M. D. Crema et al., "Sports injuries at the Rio de Janeiro 2016 summer paralympic games: use of diagnostic imaging services," *European Radiology*, pp. 1–12, 2021.

[4] J. Galvao, B. Davis, M. Tilley, E. Normando, M. R. Duchen, and M. F. Cordeiro, "Unexpected low-dose toxicity of the universal solvent DMSO," *The FASEB Journal*, vol. 28, no. 3, pp. 1317–1330, 2014.

[5] S. Sharma, R. Killedar, D. Bagewadi, and P. Shindhe, "Protocol based management of common sports injuries by integrated approach of Sandhi Marmabhighata - an open labeled clinical trial," *Journal of Ayurveda and Integrative Medicine*, vol. 12, no. 1, pp. 119–125, 2021.

[6] A. Dixit, N. Dandu, C. J. Hadley, L. N Nazarian, S. B Cohen, and M Ciccotti, "Ultrasonographic technique, appearance, and diagnostic accuracy for common elbow sports injuries," *JBJS reviews*, vol. 8, no. 11, Article ID e1900219, 2020.

[7] M. Hadizadeh, I. D. M. Azri, H. Mohafez, E. B. Hafiz, Y. Sugajima, and M. D. M. Azri, "Association between levels of sports participation and oral injuries among combat athletes," *Revista Brasileira de Medicina do Esporte*, vol. 27, no. 1, pp. 75–79, 2021.

[8] Y. Feng, S. Deb, G.-G. Wang, and A. H. Alavi, "Monarch butterfly optimization: a comprehensive review," *Expert Systems with Applications*, vol. 168, Article ID 114418, 2021.

[9] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: a new method for stochastic optimization," *Future Generation Computer Systems*, vol. 111, pp. 300–323, 2020.

[10] F. Kutlu Onay and S B. Aydemi̇r, "Chaotic hunger games search optimization algorithm for global optimization and engineering problems," *Mathematics and Computers in Simulation*, vol. 192, pp. 514–536, 2022.

[11] Z. A. Anastassi and T. E. Simos, "An optimized Runge-Kutta method for the solution of orbital problems," *Journal of Computational and Applied Mathematics*, vol. 175, no. 1, pp. 1–9, 2005.

[12] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia Tools and Applications*, vol. 80, no. 5, pp. 8091–8126, 2021.

[13] W. Deng, S. Shang, X. Cai, H. Zhao, Y. Song, and J. Xu, "An improved differential evolution algorithm and its application in optimization problem," *Soft Computing*, vol. 25, no. 7, pp. 5277–5298, 2021.

[14] S. Almahdi and S. Y. Yang, "A constrained portfolio trading system using particle swarm algorithm and recurrent reinforcement learning," *Expert Systems with Applications*, vol. 130, pp. 145–156, 2019.

[15] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2008.

[16] R. Mallipeddi, P. N. Suganthan, Q. K. Pan, and M. F. Tasgetiren, "Differential evolution algorithm with ensemble of parameters and mutation strategies," *Applied Soft Computing*, vol. 11, no. 2, pp. 1679–1696, 2011.

[17] K. M. Hamdia, X. Zhuang, and T. Rabczuk, "An efficient optimization approach for designing machine learning models based on genetic algorithm," *Neural Computing & Applications*, vol. 33, no. 6, pp. 1923–1933, 2021.

[18] H. Liang, J. Zou, K. Zuo, and M. J. Khan, "An improved genetic algorithm optimization fuzzy controller applied to the wellhead back pressure control system," *Mechanical Systems and Signal Processing*, vol. 142, Article ID 106708, 2020.

[19] J. F. Schutte, J. A. Reinbolt, B. J. Fregly, R. T. Haftka, and A. D. George, "Parallel global optimization with the particle swarm algorithm," *International Journal for Numerical Methods in Engineering*, vol. 61, no. 13, pp. 2296–2315, 2004.

[20] H.-C. Tsai and Y.-H. Lin, "Modification of the fish swarm algorithm with particle swarm optimization formulation and communication behavior," *Applied Soft Computing*, vol. 11, no. 8, pp. 5367–5374, 2011.

[21] H. M. Gomes, "Truss optimization with dynamic constraints using a particle swarm algorithm," *Expert Systems with Applications*, vol. 38, no. 1, pp. 957–968, 2011.

[22] N. Singh, S. B. Singh, and E. H. Houssein, "Hybridizing salp swarm algorithm with particle swarm optimization algorithm for recent optimization functions," *Evolutionary Intelligence*, pp. 1–34, 2020.

[23] D. Yazdani, B. Nasiri, A. Sepas-Moghaddam, and M. R. Meybodi, "A novel multi-swarm algorithm for optimization in dynamic environments based on particle swarm optimization," *Applied Soft Computing*, vol. 13, no. 4, pp. 2144–2158, 2013.

[24] W. Yu and F. Zhao, "Predictive study of ultra-low emissions from dual-fuel engine using artificial neural networks combined with genetic algorithm," *International Journal of Green Energy*, vol. 16, no. 12, pp. 938–946, 2019.

[25] E. Avard, I. Shiri, G. Hajianfar et al., "Non-contrast cine cardiac magnetic resonance image radiomics features and machine learning algorithms for myocardial infarction detection," *Computers in Biology and Medicine*, vol. 141, Article ID 105145, 2022.

[26] M. Karimi-Mamaghan, M. Mohammadi, P. Meyer, A. M. Karimi-Mamaghan, and E.-G. Talbi, "Machine learning at the service of meta-heuristics for solving combinatorial optimization problems: a state-of-the-art," *European Journal of Operational Research*, vol. 296, no. 2, pp. 393–422, 2022.

[27] Y. Ouyang, C. Yu, G. Yan, and J. Chen, "Machine learning approach for the prediction and optimization of thermal transport properties," *Frontiers of Physics*, vol. 16, no. 4, pp. 1–16, 2021.

[28] Y. Ma, R. Han, and W. Wang, "Portfolio optimization with return prediction using deep learning and machine learning," *Expert Systems with Applications*, vol. 165, Article ID 113973, 2021.

[29] B. H. Abed-alguni and D. Paul, "Island-based Cuckoo search with elite opposition-based learning and multiple mutation methods for solving optimization problems," *Soft Computing*, pp. 1–20, 2022.

[30] S. Loeffler and J. Starobin, "Reaction-diffusion informed approach to determine myocardial ischemia using stochastic in-silico ECGs and CNNs," *Computers in Biology and Medicine*, vol. 136, Article ID 104635, 2021.

[31] B. J. Kadam and K. A. Mahajan, "Optimization of cutting temperature in machining of titanium alloy using response

surface method, genetic algorithm and taguchi method," *Materials Today Proceedings*, 2021.

[32] E. El Rassy, A. Delaroque, P. Sambou, H. R. Chakravarty, and A. Matynia, "On the potential of the particle swarm algorithm for the optimization of detailed kinetic mechanisms. comparison with the genetic algorithm," *The Journal of Physical Chemistry A*, vol. 125, no. 23, pp. 5180–5189, 2021.

[33] S. Zhou, L. Sha, S. Zhu, and L. Wang, "Adaptive hierarchical update particle swarm optimization algorithm with a multi-choice comprehensive learning strategy," *Applied Intelligence*, vol. 52, pp. 1853–1877, 2021.

[34] K. Balasubramani, N. Sivarajasekar, S. Muthusaravanan et al., "Efficient removal of antidepressant Flupentixol using graphene oxide/cellulose nanogel composite: Particle swarm algorithm based artificial neural network modelling and optimization," *Journal of Molecular Liquids*, vol. 319, Article ID 114371, 2020.

[35] X. Zhong and P. Cheng, "An elite-guided hierarchical differential evolution algorithm," *Applied Intelligence*, vol. 51, no. 7, pp. 4962–4983, 2021.

[36] A. Tharwat and A. E. Hassanien, "Chaotic antlion algorithm for parameter optimization of support vector machine," *Applied Intelligence*, vol. 48, no. 3, pp. 670–686, 2018.