Metamodel-assisted metaheuristic for structural optimization problems

Érica C.R. Carvalho[1], Afonso C.C. Lemonge[2], Kenedy G.F. Júnior[1], Patrícia H. Hallak[2]

[1]*Graduate Program of Computational Modeling, Federal University of Juiz de Fora*
José Lourenço Kelmer St, São Pedro, Juiz de Fora, MG 36036-900, Brazil
ericacrcarvalho@gmail.com, kenedy.guedes@engenharia.ufjf.br
[2]*Dept. of Applied and Computational Mechanics, School of Engineering, Federal University of Juiz de Fora*
José Lourenço Kelmer St, São Pedro, Juiz de Fora, MG 36036-900, Brazil
afonso.lemonge@ufjf.br, patricia.hallak@ufjf.br

Abstract. Optimization problems are common in many different areas, especially engineering. The complexity of modern problems has led to the development of increasingly complex mathematical models, resulting in expensive simulation models. An alternative for solving these problems is population-based metaheuristics, especially those of natural inspiration. However, they usually require many evaluations to obtain a feasible or even satisfactory solution. In this context, the application of metamodels, or surrogate models, together with metaheuristics, has received the growing attention of researchers in several areas. The metamodels generate a simpler computational model to be used in parts of the optimization process, replacing the original model. This work presents an application strategy of metamodels within metaheuristics, which allows for computational cost reduction. The methodology is applied to structural optimization problems, demonstrating its applicability and establishing it as an alternative to improving solutions in the context of fixed-budget simulations.

Keywords: Structural Optimization Problems, Metaheuristics, Machine Learning, Metamodels

1 Introduction

Metaheuristic algorithms for optimal design of structures has attracted the interest of many researchers in recent decades [1–7]. In general, the problems were modeled simplified, disregarding specific characteristics of the studied phenomenon to reduce the complexity and the computational cost during the search process. However, over time, the evolution of computational resources allowed optimization problems through more reliable models to be solved. The optimization process may require the resolution of very complex models since most of these problems are modeled by partial differential or integral equations and need to be solved by accurate numerical methods (e.g., structural finite element analysis, computational fluid dynamics, etc.)[8]. However, these optimization procedures are usually highly computationally expensive once the computational time required by a single function evaluation is often counted in hours or even days for large-scale applications.

For all these reasons, an alternative available in the literature is the surrogate-based or metamodel-assisted optimization which has emerged as an unavoidable approach to tackling complex engineering optimization problems [9]. This approach uses a surrogate model to perform the simulations, get approximations and replace expensive high-fidelity simulations during structural optimization processes. These models seek to avoid using computationally expensive evaluations to obtain: (i) better results with the same processing time, or (ii) similar results more quickly. The most common metamodels are Polynomial regression, Neural Networks, and Kriging models [10].

This paper proposes a multi-objective particle swarm optimization algorithm assisted by a deep neural network model to be applied to constrained structural optimization problems. The structural problem consists in the minimization of the mass of truss structures and the maximization of the first natural frequency of vibration. The axial stresses and nodal displacements are the constraints. The computational experiment is on benchmark single-objective structural optimization problems widely discussed in the literature: the 10-bar truss. Pareto curves show non-dominated solutions to optimization problems with and without using the metamodel.

The paper's organization is as follows: Section 2 states the multi-objective structural optimization problem, an overview of metaheuristics and swarm intelligence algorithm, and metamodels concepts. The optimum design problem and metamodel-assisted approach are described in Section 3. Section 4 presents the numerical experiments, and summarizes the benefits of incorporating metamodels into structural optimization problems. Finally, conclusions and proposed future research are reported in Section 5.

2 Methods

2.1 The multi-objective structural optimization problem

Although single-objective structural optimization problems are commonly found in the literature, the formulation of optimization problems involving multiple objectives appears naturally due to the presence of two or more conflicting objective. The multi-objective structural optimization problem considered here consists in finding the set of cross-sectional areas $\mathbf{x} = \{A_1, A_2, ..., A_N\}$ which minimizes the weight of the structure and maximizes its first natural frequency as it follows

$$\min \quad W(\mathbf{x}) = \sum_{i=1}^{N} \rho A_i L_i, \quad \text{and} \quad \max \quad \omega_1(\mathbf{x}), \tag{1}$$

subject to the normalized displacements constraints

$$\frac{|u_j|}{\bar{u}} - 1 \leq 0, \quad j = 1, \ldots, M, \tag{2}$$

and the normalized stress constraints

$$\frac{|\sigma_i|}{\bar{\sigma}} - 1 \leq 0, \quad i = 1, \ldots, N, \tag{3}$$

where $W(\mathbf{x})$ is the weight of the structure, ρ is the specific mass of the material, L_i is the length of $i-$th bar of the structure, $\omega_1(\mathbf{x})$ is the first natural frequency of vibration, and u_j and σ_i are respectively the nodal displacement of the $j-$th translational degree of freedom, and the stress of the $i-$th bar. $\bar{u}$ is the maximum displacement for each nodal point, and $\bar{\sigma}$ is the allowable stress for the material. M is the number of translational degrees of freedom, and N is the total number of bars in the truss structure.

2.2 Metaheuristics and particle swarm algorithm

The last 20 years have witnessed vast growth in a non-deterministic based optimizer called metaheuristic. Physical or natural phenomena have inspired metaheuristic algorithms and are a broad family of optimization methods to find accurate solutions to complex optimization problems when exact methods are not applicable [11]. This class of approaches includes, among others, Genetic Algorithm, Simulated Annealing, Ant Colony Optimization, and Particle Swam Optimization.

Many approaches dedicated to designing and implementing these algorithms to solve multi-objective optimization problems (MOPs) have shown excellent results in solving several problems. Life in society offers more chances of survival as it facilitates hunting and gathering food, and reduces the possibility of attack by predators, among others [12].

One of these algorithms is inspired by the social behavior of birds flocking in search of food and is known as Particle Swarm Optimization (PSO). PSO was introduced by Eberhart & Kennedy [13] and widely used in the literature. The multi-objective version of PSO entitled Multi-objective Craziness based Particle Swarm Optimization (MOCRPSO) operates with a special code based on CRPSO (Kar *et al.* [14]) and incorporates a crowding distance mechanism, non-dominated solutions, and an external archive, together with a mutation operator based on the MOPSO-CD (Raquel & Naval [15])).

2.3 Metamodels

Metamodel emerged around the 1970s (Blanning [16]). It can be defined as a simplified model of the original model of the problem so that its role is to replace or approximate functions that require a lot of computational effort to obtain the results. A model is a simplified idealization of phenomena in the real world, which nevertheless significantly reproduces observed behavior. On the other hand, the metamodel is an abstraction, highlighting the properties of the model itself.

Many metamodels approaches are available in the literature, such as Artificial neural networks [17], Kriging or Gaussian processes [18], Radial Basis Functions [19], Support vector machines [20], among others. Alternatively, several surrogates may be derived from physical or numerical simplifications of the original model, in which case they are more strongly problem-dependent.

Metamodels should be simple, general, and keep the number of control parameters as small as possible. One of the first studies involving metamodel-assisted metaheuristics optimization was presented by Jin and Sendhoff [21]. Since then, several works have been developed regarding the use of metamodels as an alternative for solving engineering problems [22–26]. Furthermore, metamodels are also used for solving high-cost optimization problems [27–29]. Regis [30] presents a survey of surrogate-assisted methods for computationally expensive constrained global optimization problems.

Despite promising results obtained through metamodels, each one needs a specific dataset for proper construction. Depending on the number of variables and the size of the solution space, the number of high-fidelity simulations required will directly influence the correct construction of the metamodel. In addition, the metamodel selection depends on the formulated problem and the case study or structure to be optimized. Therefore, knowing the relationships between the type of metamodels and other categorical variables related to structural design optimization would be of great interest (Negrin *et al.* [31]).

3 Metamodel-assisted approach for optimization

3.1 Optimum design problem

For the optimization of truss structures, it is applied to a well-known structural multi-objective optimization problem named as 10-bar truss (Gellatly and Berke [32]). The geometrical data is given in Figure 1. The specific weight is ρ = 0.1 lb/in^3, and the material's Young's modulus is E = 10^4 ksi. Vertical downward loads of 100 kips are applied at nodes 2 and 4, the stress in each bar is limited to $\pm$ 25 ksi, and the displacement in each free node in each direction should not exceed 2 inches. A non-structural mass of 1000 lb is attached to the free nodes. The lower and upper bounds for the cross-sectional areas are defined between 0.1 and 40 in^2.

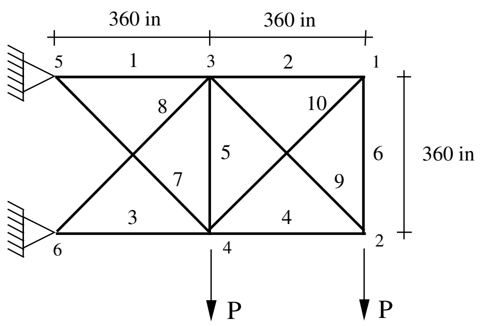


Figure 1. 10-bar truss, taken from Lemonge *et al.*[33].

3.2 Deep neural network-based metamodeling

This section presents the scheme of the integration model with the optimization algorithm. The MOCRPSO algorithm is introduced to resolve the structural optimization problem assisted by a deep neural network (DNN). The multi-layer neural network is a set of mathematical relationships between the inputs and outputs through a training process. The network learns through analyzing the training data by adjusting weights and biases (Hamdia *et al.*[34]). In recent years, a number of studies have proved the effectiveness in applying DNN to structural optimization problems [35–38].

A neural network with one hidden layer is called a shallow neural network, whereas a neural net with two or more hidden layers is called a deep neural network. This study establishes a DNN model to predict the eight displacements of the 10-bar truss (4 free nodes in each direction). After obtaining the approximate displacements, the stresses in each bar are calculated. DNN architecture with fully connected layer is depicted in Figure 2. There are an input layer with 10 neurons (one for each area), an output layer with 8 neurons (one for each displacement), and two hidden layers with 24 neurons each. The units of the pra present layer are connected to all units in the previous layer via weights and bias and use a sigmoid as an activation function.

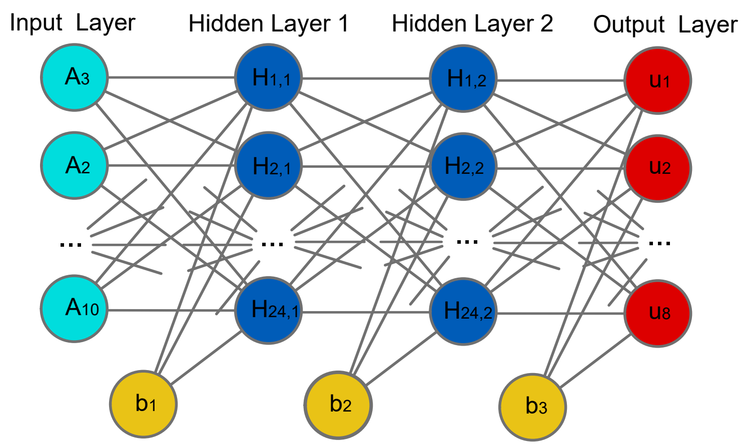


Figure 2. Architecture of the deep neural network used in this study.

The deep neural network used here was developed by the authors using C programming language. No framework or library was used. The entire architecture, equations, and derivatives were built, allowing the network to improve.

In summary, the steps of the approach presented consists of four main blocks as below:

1. Some samples (A) are generated, and then the displacement responses (u) are collected using the finite element method (FEM) (Hughes [39]) through the analysis phase.
2. Build a DNN model and training, then take the well-trained network as a surrogate model of the constraints.
3. An equivalent optimization problem is constructed based on the original objective functions and the DNN model.
4. The MOCRPSO algorithm is introduced to resolve the optimization problem.

In the following section, more details of the presented approach are given, including preliminary tests on the metamodel and the parameterization of the optimization algorithm used.

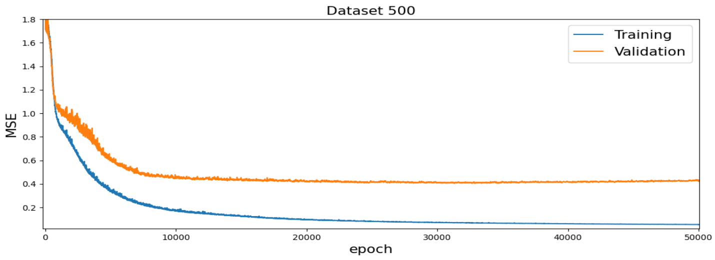
4 Numerical experiments

In supervised learning, the dataset is labeled, and the best DNN model is determined by training to identify the optimal parameters (weights and bias). They can get sufficiently good approximations when the amount of data is considerable. In this context, three datasets were used to investigate the effects of the number of samples, considering 500, 1000, and 7000 (cross-sectional areas and displacements). A new dataset, including 1000 samples, was used for model validation in all datasets. The model was trained by gradient descent optimizer with a learning rate starting at 0.2 and decreasing until 0.057 according to the number of epochs, batch-size of 25 and 50000 epochs. After this stage, each dataset's mean square error (MSE) progression throughout the iterations was observed, as depicted in Figure 3. Table 1 presents the best MSE and their average obtained in the three datasets' sample training and validation process.

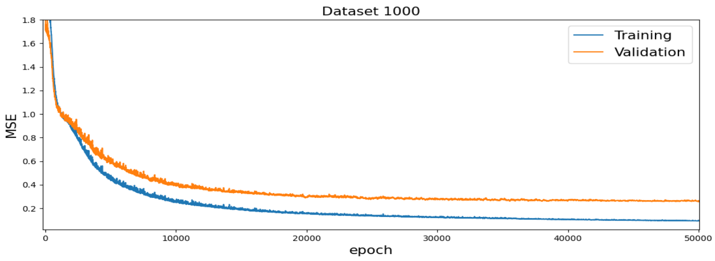
Learning for the 500-sample dataset is the fastest among the three sets. However, the validation error is the largest over time. The best MSE obtained from validation is presented for the dataset with 7000 samples since it brought minor errors between the training and validation. Therefore, the dataset with 7000 samples was chosen here.

The previous steps consist of the first two main blocks of the approach, as presented in Section 3.2. After that, the remaining steps are conducted as follows: the initial population was randomly generated considering the maximum number of objective function evaluations is 50000 (50 particles and 1000 generations), and the number of independent runs is 45. The MOCRPSO was developed using C++ language and its parameters are: $c_1 = c_2 = 2.05$, $v^{craziness} = 0.001$, $Pcr = 0.5$, external file limit $ARQ = 500$, and global neighborhood topology. It is important to remember that in this stage, the structure was not analyzed by the FEM during the evolutionary process.

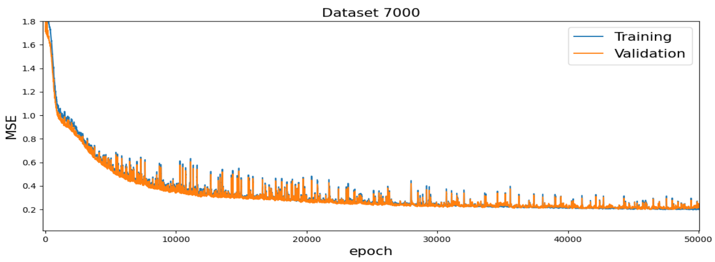
The results obtained by the MOCRPSO algorithm and the metamodel-assisted approach using the DNN (MOCRPSO-DNN) of the 10-bar truss are presented in Figure 4. The Pareto curves are displayed in Figure 4a and are similar, demonstrating the quality and variety of solutions obtained. Figure 4b shows a zoom of an interesting



(a)



(b)



(c)

Figure 3. The convergence history of MSE with different size of data.

Table 1. MSE values with different size of dataset.

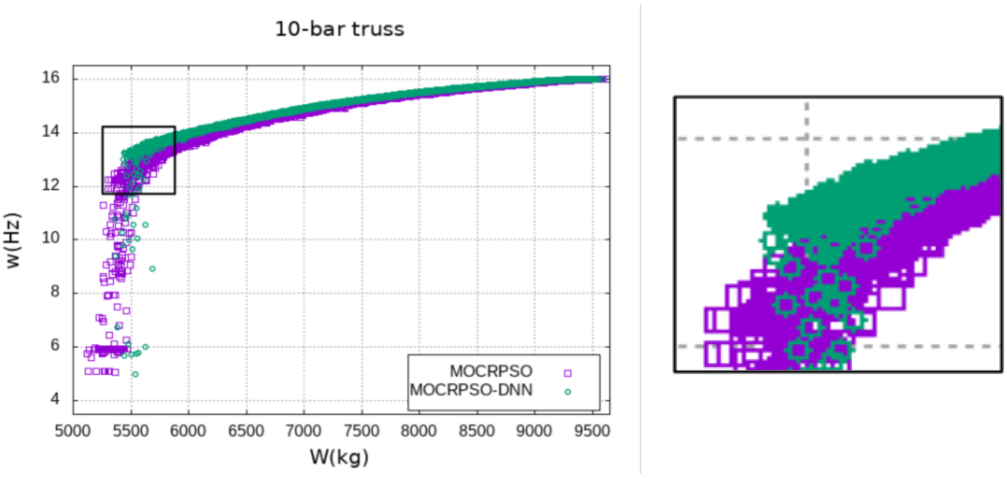
	Best MSE		Final MSE	
Size data	Training	Validation	Training	Validation
500	0.0541	0.4046	0.0543	0.4240
1000	0.0923	0.2572	0.0929	0.2572
7000	0.1973	0.1992	0.2082	0.2221

region within the multi-objective problem. In this way, decision-makers can choose any solution that suits them within the presented region. The execution time was not considered at this moment due to the 10-bar truss is a problem with a low computational cost and obtains good results in an acceptable time. Other experiments considering structures with more bars will be conducted in order to examine the computational cost in both approaches.

5 Concluding remarks and future work

In this work, we demonstrated that applying nature-inspired metaheuristic algorithm to DNN in the context of obtaining approximations of nodal displacements can achieve good results without using the finite element method during the evolutionary process. The obtained solutions are competitive and demonstrated the efficiency of the proposed approach.

However, there are many issues that can still be improved to obtain better results, both in the context of DNN architecture and parameters, as well as in the use of other types of metamodels. Future works intend to apply the approach for solving other problems considering large-scale optimization problems, and expensive simulation models. And yet, improve the prediction accuracy of the surrogates model by combining them to form a ensemble.



(a) Pareto curves of the MOCRPSO algorithm and the MOCRPSO-DNN. (b) Preferred region of the Pareto curves.

Figure 4. Results obtained for the 10-bar truss.

Acknowledgements. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brazil (CAPES) – Finance Code 001, Brazilian Agencies CNPq (Grants 308105/2021-4 and 303221/2022-4), and FAPEMIG (Grant TEC PPM-00174-18 and APQ-00869-22).

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] S. Gholizadeh, H. Asadi, and A. Baghchevan. Optimal design of truss structures by improved multi-objective firefly and bat algorithms. *Int. J. Optim. Civil Eng*, vol. 4, n. 3, pp. 415–431, 2014.
[2] J. S. Angelo, H. S. Bernardino, and H. J. Barbosa. Ant colony approaches for multiobjective structural optimization problems with a cardinality constraint. *Advances in Engineering Software*, vol. 80, pp. 101–115, 2015.
[3] V. Ho-Huu, T. Nguyen-Thoi, T. Truong-Khac, L. Le-Anh, and T. Vo-Duy. An improved differential evolution based on roulette wheel selection for shape and size optimization of truss structures with frequency constraints. *Neural computing and applications*, vol. 29, n. 1, pp. 167–185, 2018.
[4] H. Assimi, A. Jamali, and N. Nariman-Zadeh. Multi-objective sizing and topology optimization of truss structures using genetic programming based on a new adaptive mutant operator. *Neural Computing and Applications*, vol. 31, pp. 5729–5749, 2019.
[5] J. P. G. Carvalho, É. C. Carvalho, D. E. Vargas, P. H. Hallak, B. S. Lima, and A. C. Lemonge. Multi-objective optimum design of truss structures using differential evolution algorithms. *Computers & Structures*, vol. 252, pp. 106544, 2021.
[6] J.-S. Chou and D.-N. Truong. Multiobjective forensic-based investigation algorithm for solving structural design problems. *Automation in Construction*, vol. 134, pp. 104084, 2022.
[7] É. C. Carvalho, J. P. G. Carvalho, H. S. Bernardino, A. C. Lemonge, P. H. Hallak, and D. E. Vargas. Solving multi-objective truss structural optimization problems considering natural frequencies of vibration and automatic member grouping. *Evolutionary Intelligence*, vol. , pp. 1–26, 2022.
[8] R. Filomeno Coelho. Metamodels for mixed variables based on moving least squares: Application to the structural analysis of a rigid frame. *Optimization and Engineering*, vol. 15, pp. 311–329, 2014.
[9] J. W. Bandler, S. Koziel, and K. Madsen. surrogate modeling and space mapping for engineering optimization. *Optimization and engineering*, vol. 9, pp. 307–310, 2008.
[10] R. D. Bäckryd, A.-B. Ryberg, and L. Nilsson. Multidisciplinary design optimisation methods for automotive structures. *International Journal of Automotive and Mechanical Engineering*, vol. 14, n. 1, pp. 4050–4067, 2017.
[11] G. R. Zavala, A. J. Nebro, F. Luna, and C. A. Coello Coello. A survey of multi-objective metaheuristics applied to structural optimization. *Structural and Multidisciplinary Optimization*, vol. 49, pp. 537–558, 2014.
[12] Y.-f. Zhu and X.-m. Tang. Overview of swarm intelligence. In *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, volume 9, pp. V9–400. IEEE, 2010.

[13] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, pp. 39–43. IEEE, 1995.
[14] R. Kar, D. Mandal, S. Mondal, and S. P. Ghoshal. Craziness based particle swarm optimization algorithm for fir band stop filter design. *Swarm and Evolutionary Computation*, vol. , 2012.
[15] C. R. Raquel and P. C. Naval Jr. An effective use of crowding distance in multiobjective particle swarm optimization. In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pp. 257–264. ACM, 2005.
[16] R. W. Blanning. The sources and uses of sensitivity information. *Interfaces*, vol. 4, n. 4, pp. 32–38, 1974.
[17] S. Ferrari and R. F. Stengel. Smooth function approximation using neural networks. *IEEE Transactions on Neural Networks*, vol. 16, n. 1, pp. 24–38, 2005.
[18] M. T. Emmerich, K. C. Giannakoglou, and B. Naujoks. Single-and multiobjective evolutionary optimization assisted by gaussian random field metamodels. *IEEE Transactions on Evolutionary Computation*, vol. 10, n. 4, pp. 421–439, 2006.
[19] K. Giannakoglou. Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence. *Progress in Aerospace sciences*, vol. 38, n. 1, pp. 43–76, 2002.
[20] V. Kecman. *Learning and soft computing: support vector machines, neural networks, and fuzzy logic models*. MIT press, 2001.
[21] Y. Jin and B. Sendhoff. Fitness approximation in evolutionary computation-a survey. In *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation*, pp. 1105–1112, 2002.
[22] M.-E. Miranda-Varela and E. Mezura-Montes. Surrogate-assisted differential evolution with an adaptive evolution control based on feasibility to solve constrained optimization problems. In *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*, pp. 809–822. Springer, 2016.
[23] de R. Paula Garcia, de B. S. L. Lima, and de A. C. Castro Lemonge. A surrogate assisted differential evolution to solve constrained optimization problems. In *2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, pp. 1–6. IEEE, 2017.
[24] M. Moustapha and B. Sudret. A unified framework for two-level reliability-based design optimization using metamodels. *ICASP13 Proceedings*, vol. , 2019.
[25] L. Krischer and M. Zimmermann. Decomposition and optimization of linear structures using meta models. *Structural and Multidisciplinary Optimization*, vol. 64, n. 4, pp. 2393–2407, 2021.
[26] J. Yang, H. Jensen, and J. Chen. Structural optimization under dynamic reliability constraints utilizing probability density evolution method and metamodels in augmented input space. *Structural and Multidisciplinary Optimization*, vol. 65, n. 4, pp. 1–26, 2022.
[27] R. Shi, L. Liu, T. Long, Y. Wu, and Y. Tang. Filter-based adaptive kriging method for black-box optimization problems with expensive objective and constraints. *Computer Methods in Applied Mechanics and Engineering*, vol. 347, pp. 782–805, 2019.
[28] L. Chen, H. Qiu, L. Gao, C. Jiang, and Z. Yang. Optimization of expensive black-box problems via gradient-enhanced kriging. *Computer Methods in Applied Mechanics and Engineering*, vol. 362, pp. 112861, 2020.
[29] H. T. Mai, J. Kang, and J. Lee. A machine learning-based surrogate model for optimization of truss structures with geometrically nonlinear behavior. *Finite Elements in Analysis and Design*, vol. 196, pp. 103572, 2021.
[30] R. G. Regis. A survey of surrogate approaches for expensive constrained black-box optimization. In *Optimization of Complex Systems: Theory, Models, Algorithms and Applications*, pp. 37–47. Springer, 2020.
[31] I. Negrin, M. Kripka, and V. Yepes. Metamodel-assisted design optimization in the field of structural engineering: A literature review. In *Structures*, volume 52, pp. 609–631. Elsevier, 2023.
[32] R. A. Gellatly and L. Berke. Optimal structural design. Technical report, DTIC Document, 1971.
[33] A. C. Lemonge and H. J. Barbosa. An adaptive penalty scheme for genetic algorithms in structural optimization. *International Journal for Numerical methods in engineering*, vol. 59, n. 5, pp. 703–736, 2004.
[34] K. M. Hamdia, X. Zhuang, and T. Rabczuk. An efficient optimization approach for designing machine learning models based on genetic algorithm. *Neural Computing and Applications*, vol. 33, pp. 1923–1933, 2021.
[35] Y. Yu, T. Hur, J. Jung, and I. G. Jang. Deep learning for determining a near-optimal topological design without any iteration. *Structural and Multidisciplinary Optimization*, vol. 59, n. 3, pp. 787–799, 2019.
[36] D. W. Abueidda, S. Koric, and N. A. Sobh. Topology optimization of 2D structures with nonlinearities using deep learning. *Computers & Structures*, vol. 237, pp. 106283, 2020.
[37] H. T. Kollmann, D. W. Abueidda, S. Koric, E. Guleryuz, and N. A. Sobh. Deep learning for topology optimization of 2d metamaterials. *Materials & Design*, vol. 196, pp. 109098, 2020.
[38] H. Nguyen, T. Vu, T. P. Vo, and H.-T. Thai. Efficient machine learning models for prediction of concrete strengths. *Construction and Building Materials*, vol. 266, pp. 120950, 2021.
[39] T. J. Hughes. *The finite element method: linear static and dynamic finite element analysis*. DoverPublications.com, 2012.