Journal of Engineering Managment and Soft Computing (JEMSC)

Optimization of Steel Alloy Composition to Maximize Yield Strength Using a Machine Learning Model and the Cuckoo Optimization Algorithm

Mohammad Reza Esmaeili Noroozi[1]✉ and Fatemeh Zare Mehrjardi[2]

1. MSc. Student, Department of Computer Engineering, University of Meybod, Meybod, Iran, Email: m.esmaeili.noroozi1380@gmail.com
2. Corresponding author, Assistant Professor, Department of Computer Engineering, University of Meybod, Meybod, Iran, Email: fzare@meybod.ac.ir

Article Info

Article type:
Research Article

Article history:
Received 29 August 2025
Received in revised form 20 November 2025
Accepted 26 December 2025
Published online 1 January 2026

Keywords:
steel, alloy, yield strength, machine learning, random forest, cuckoo optimization.

ABSTRACT

Designing steel alloys with superior mechanical properties, particularly high yield strength, is a central challenge in materials engineering. Traditional trial-and-error methods are often unable to find optimal solutions due to their high cost and time consumption. To overcome these limitations, this research introduces a two-stage intelligent framework that significantly accelerates the alloy design process by combining machine learning and metaheuristic optimization. In the first stage, a Random Forest model was trained on experimental data, demonstrating notable performance in predicting yield strength with a coefficient of determination (R^2) of 0/8194 and a Mean Squared Error (MSE) of 12445/02. In the second stage, this model was used as a cost function for the Cuckoo Optimization Algorithm (COA) to discover the optimal alloy composition. After 100 iterations, the COA algorithm converged to a composition with a yield strength of 2456/46 MPa, which is the highest value reported in the studied dataset. This optimal composition contained significant amounts of key elements, such as Cobalt (11/19%), Chromium (10/61%), Molybdenum (6/04%), and Tungsten (4/26%), which are consistent with strengthening mechanisms like solid solution strengthening and carbide precipitation. The results demonstrate that the combination of machine learning and metaheuristic optimization is a powerful and efficient approach for discovering novel alloys with desired properties, which can drastically accelerate the materials development cycle.

Cite this article: Esmaeili Noroozi, M. & et al, (2026)., Optimization of Steel Alloy Composition to Maximize Yield Strength Using a Machine Learning Model and the Cuckoo Optimization Algorithm. *Engineering Management and Soft Computing,* 12 (1). 131-143.

DOI: https://doi.org/10.22091/jemsc.2026.13746.1299

DOI: https://doi.org/10.22091/jemsc.2026.13746.1299

Publisher: University of Qom

1) Introduction

The steel industry, as the backbone of the global economy, has consistently pursued the production of products with superior mechanical properties, lower cost, and higher efficiency. Among these, yield strength stands out as one of the most crucial criteria for evaluating structural performance, playing a decisive role in material design and selection for critical applications (Dieter & Bacon, 1988). An increase in yield strength signifies a material's enhanced ability to withstand greater stress before undergoing permanent deformation, directly contributing to improved safety, durability, and weight reduction of structures (Callister & Rethwisch, 2018). For instance, in the aerospace industry, the use of high-strength steel alloys enables the design of lighter and more efficient airframes and engines (Reed, 2006).

Traditional alloy design methods, grounded in empirical knowledge, metallurgical experimentation, and trial-and-error approaches, face significant challenges. These methods not only demand considerable time and financial resources but also often fail to achieve optimal solutions due to the non-linear and multivariate complexity of the relationships between elemental composition and final properties (Ashby, 2013). These limitations underscore the growing need for novel and intelligent approaches.

In recent decades, materials informatics and artificial intelligence have brought about transformative changes in the field of materials development (Agrawal & Choudhary, 2016). By leveraging machine learning algorithms, these approaches can learn complex, non-linear patterns from vast datasets of experimental and simulation results. Machine learning can act as a rapid and cost-effective surrogate model, predicting mechanical properties in a fraction of a second—a task that might take weeks through physical experimentation (Mehta & Mannan, 2019). Accordingly, an effective strategy involves integrating predictive machine learning models with metaheuristic optimization algorithms (Hassani et al., 2020). In this framework, the machine learning model maps the response space between composition and mechanical properties, and the optimization algorithm, acting as an intelligent explorer, searches this learned space for compositions that deliver the most desirable properties (Ramprasad & Kim, 2017).

The primary innovation of this research is the introduction of a two-stage intelligent framework for designing steel alloys with maximized yield strength. In the first stage, a Random Forest machine learning model, chosen for its strong capability in handling non-linear data and outliers, serves as an accurate predictive model. In the second stage, this trained model is integrated as the objective function into the COA. This algorithm, renowned for its use of Lévy flight, exhibits a superior capacity for global exploration and identifying optimal points within complex search spaces (Yang & Deb, 2009). The ultimate goal is to provide an intelligent and efficient solution to overcome the challenges of traditional alloy design and to establish a generalizable framework for future research in this domain.

This paper is organized as follows: Section 2 reviews the relevant research background and literature. Section 3 details the research methodology, including the dataset, machine learning model, and optimization framework. Section 4 presents, discusses, and analyzes the findings and results. Finally, Section 5 provides the conclusion, summarizing key insights and suggesting directions for future work.

2) Literature Review

Data-driven materials design has evolved into a dynamic and rapidly growing field, achieving maturity in recent years thanks to the increased accessibility of large databases and significant advancements in machine learning algorithms (Rajan, 2005). In an influential review paper, Agrawal and Choudhary (2016) highlighted the pivotal role of machine learning in accelerating the discovery of new materials, introducing it as a novel paradigm in materials science.

In the context of predicting steel properties, various models have been employed. Wang et al. (2018) utilized Artificial Neural Networks (ANN) to predict the yield strength of structural steels, demonstrating their ability to model complex non-linear relationships. However, due to challenges such as overfitting and the need for large datasets, tree-based models have gained greater popularity. In his seminal paper, Breiman (2001) introduced Random Forests, showing that this method creates a robust

and resilient model by aggregating numerous weak decision trees, resulting in excellent performance for predicting material properties. Consequently, this research employs the Random Forest model as the primary predictive tool.

To address optimization problems, numerous metaheuristic algorithms have been developed. The Particle Swarm Optimization (PSO) algorithm has been applied to alloy composition design, yielding promising results (Zhao et al., 2023). Similarly, in studies related to stainless steel, PSO has been used to model the effect of oxide compounds on the weld properties of 304L steel (Djoudjou et al., 2021). Despite PSO's efficiency in local search, it can sometimes suffer from premature convergence and become trapped in local optima (Rajwar et al., 2023; Zhou et al., 2024).

Beyond PSO, Genetic Algorithms (GA) have also been extensively used (Goldberg, 1989), but their convergence rate can be slow. To overcome these limitations, more novel algorithms have been introduced.

The COA, proposed by Rajabioun (2011), is inspired by the brood parasitism behavior of cuckoos. Its defining feature is the use of Lévy Flight, which grants it a superior capability for global exploration and helps prevent entrapment in local optima. Several studies have also shown that COA outperforms algorithms such as PSO and GA in complex optimization problems (Gandomi & Yang, 2011). This capability has made it an attractive choice for emerging applications. For instance, Tejani et al. (2024) developed a multi-objective version of COA for structural optimization. Furthermore, Genc and Kalimbetova (2024) demonstrated the successful application of COA in designing intelligent controllers, indicating its ability to solve real-world engineering problems.

Alongside data-driven approaches, classical metallurgical knowledge continues to play a vital role in understanding the strengthening mechanisms of alloys. Elements such as Chromium (Cr) and Nickel (Ni) (Leslie, 1981), Molybdenum (Mo) and Vanadium (V) (Bhadeshia, 2015), Tungsten (W) (Totten & Howes, 1995), and Cobalt (Co) (Reed, 2006) each contribute to increased strength through specific mechanisms like solid solution strengthening or carbide precipitation.

The present research aims to bridge these two domains by seeking to discover an elemental combination that maximizes yield strength. By integrating the predictive power of the Random Forest model with the global search capability of the COA algorithm, this approach is more comprehensive than previous studies, offering an efficient framework for the design of advanced alloys.

3) Methodology

This section details the dataset, algorithms, and proposed methodology used in this research.

Dataset and Preprocessing

The present study is based on an experimental dataset named "steel_strength.csv," containing 2,469 samples, sourced from the Kaggle platform. This dataset encompasses various alloy compositions and their corresponding ultimate steel strength. The input variables consist of the weight percentages of Carbon (C), Manganese (Mn), Silicon (Si), Chromium (Cr), Nickel (Ni), Molybdenum (Mo), Vanadium (V), Nitrogen (N), Niobium (Nb), Cobalt (Co), Tungsten (W), Aluminum (Al), Titanium (Ti), and Iron (Fe). The target variable is the Yield Strength in Megapascals (MPa). However, it is important to note that this dataset, like many experimental datasets, has inherent limitations. A thorough examination reveals that the percentage of certain elements, such as Niobium (Nb), Titanium (Ti), and Cobalt (Co), is near zero in a significant portion of the samples. While manageable for model training, this may limit the model's ability to make accurate predictions for optimal compositions that include higher percentages of these elements. Furthermore, the compositional range covered by the dataset represents only a subset of the vast space of steel alloys. These limitations impact the model's generalizability to compositions outside the training data range, indicating that for future research, collecting more comprehensive and diverse data is essential for building a more robust and practical model.

Following data collection, the first step was data preparation. The following preprocessing steps were applied to the dataset:

- Replacing missing values with zero.

- Converting column names to lowercase for code compatibility.
- Extracting the minimum and maximum range for each element from the existing data to be used as optimization constraints.

Machine Learning Model

In this study, a Random Forest Regressor was used as a powerful and robust machine learning algorithm for predicting steel yield strength. This model, implemented from the scikit-learn library, is an ensemble learning method based on decision trees. By aggregating the results of a large number of weak decision trees, it mitigates overfitting and demonstrates a high capability in modeling complex, non-linear relationships between features (Chehreh & Sarabadani, 2023; Salman et al., 2024). Its core mechanism involves two key techniques: bootstrap sampling (sampling with replacement) and random feature selection at each node for splitting, which significantly enhance the model's stability and accuracy (Niazi & Razavi, 2024; Tapio, 2025).

To determine the optimal parameters for the Random Forest model, a five-fold cross-validation method was employed. This process aimed to achieve an optimal balance between model performance and computational time. Key model parameters, including the number of trees and the maximum allowed depth for each tree, were systematically examined within a specified range. During the cross-validation process, the parameter combination resulting in the lowest MSE was selected as the final model parameters. This approach ensures that the model's performance is not limited to the training data and that it retains the ability to make accurate predictions on new data. The main model parameters set for this research are as follows:

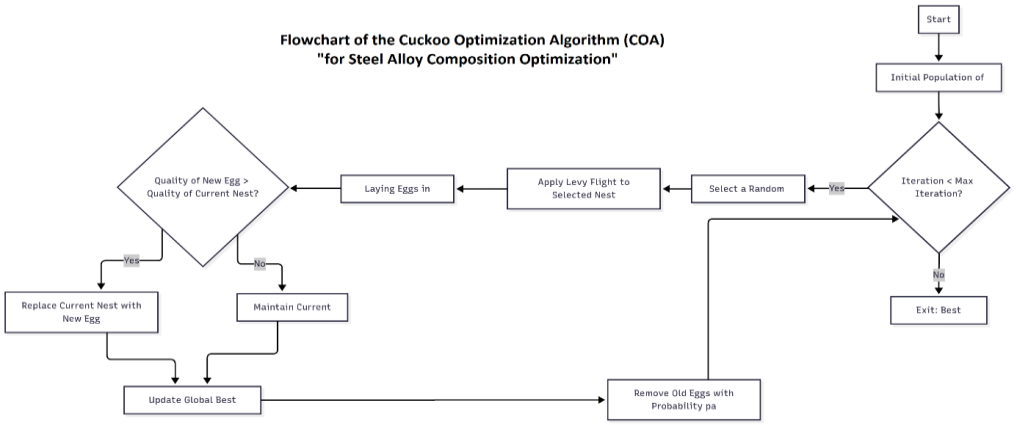
- **n_estimators=100:** The number of trees in the forest.
- **max_depth=10:** The maximum allowed depth for each tree, set to control model complexity and prevent overfitting.
- **random_state=42:** To ensure result reproducibility and experiment replicability.

Finally, after parameter tuning, the Random Forest model was trained using 80% of the data as the training set. The model's performance was evaluated on the remaining 20% (test set) using the coefficient of determination (R^2) and MSE metrics to assess its efficiency in predicting yield strength. To increase confidence in the model's generalizability, a ten-fold cross-validation was also performed. The trained model was then used as the cost function, which is the core component of the evolutionary optimization algorithm.

Cuckoo Optimization Algorithm (COA)

The Cuckoo Optimization Algorithm (COA) is a metaheuristic algorithm inspired by the natural brood parasitic behavior of cuckoo birds, proposed by Rajabioun in 2011. Utilizing three main concepts of mutation, egg laying within a radius, and the elimination of old eggs, this algorithm possesses a high capability for global exploration and discovering optimal points in complex search spaces. The flowchart of the COA algorithm steps is presented in Figure 1.

Figure 1) Flowchart of the Cuckoo Optimization Algorithm (COA) Steps



The algorithm consists of five main stages:

1. Initialization: Generate an initial population of nests. Each nest represents a potential solution to the optimization problem (in this project, a steel alloy composition).
2. Main Loop (for a specified number of iterations):

   a. Levy Flight Mutation: Each cuckoo (nest) moves in the search space using a random walk with long step lengths. This movement helps the algorithm quickly reach new regions (exploration).

   b. Egg Laying within a Radius: Each cuckoo lays several eggs (new solutions) within a small radius around its current location. If the quality (objective function value) of one of these eggs is better than the current nest, it replaces it.

   c. Removal of Old Eggs: With a specified probability (pa), some nests (weak solutions) are selected, and their eggs are eliminated. These nests are then replaced with new random solutions. This step prevents population stagnation and maintains diversity.
3. Update Best Solution: The best solution found (the nest with the highest quality) is stored throughout the iterations.
4. Termination Check: If the termination condition (e.g., maximum iterations) is not met, return to Stage 2.
5. Output: Return the best solution found.

The selection of the COA for this problem was based on a careful review of its advantages and capabilities in solving complex, multi-dimensional problems. The most important reason is its Levy Flight mechanism, which allows the algorithm to quickly access new regions of the search space and avoid becoming trapped in local optima (Singh et al., 2025; Tejani et al., 2024). This feature is crucial for the alloy design problem, which has a vast and non-linear search space. Another reason is its flexibility and simplicity in integrating with an external objective function. For customization, the trained Random Forest model was defined as the objective function, with the primary goal of the algorithm being to maximize the output of this model, i.e., the predicted yield strength. Furthermore, a customized version of the algorithm was implemented by applying necessary constraints, limiting the search space so that the value of each alloying element falls within the permissible ranges found in the original database. Additionally, a strict constraint was added to ensure that the sum of the weight percentages of all elements in each candidate solution equals exactly 100%. This implemented framework enabled the algorithm to effectively search for the optimal alloy composition, providing a practical and efficient solution.

COA Parameters

In this research, the parameters of the COA were configured, based on project reports and the executed code, to create a suitable balance between exploration and exploitation processes and to converge rapidly towards the optimal solution. These parameters, determined through experience and trial and error, are presented in Table 1.

Table 1) Parameters of the Cuckoo Optimization Algorithm (COA) and Their Descriptions

Parameter	Symbol	Value	Description
Population Size	n_cuckoos	30	Number of initial nests (candidate solutions).
Maximum Iterations	max_iter	100	Maximum number of times the main loop of the algorithm is executed.
Levy Step Size	step_size	0.02	Determines the magnitude of Levy flight steps.
Discovery Rate	pa	0.20	Probability that an old, weak nest (egg) will be discovered and abandoned.
Egg Laying Radius	egg_laying_radius	0.05	Radius around each cuckoo where new eggs (solutions) are laid.
Objective Function	obj_func	-	The function to be maximized (or minimized). Here, the yield strength predicted by the Random Forest model.
Variable Bounds	bounds	min/max	Permissible range for each alloying element (e.g., min=0, max=20 for Cr).

These settings formed the basis for the algorithm's successful performance in achieving the optimal high-yield-strength alloy composition. Notably, the Levy step size (step_size = 0.02) created a suitable balance between global search (exploration) and local search (exploitation); longer steps allowed the algorithm to quickly discover new regions, while shorter steps aided in refining the search within promising areas.

Software Tools and Execution Environment

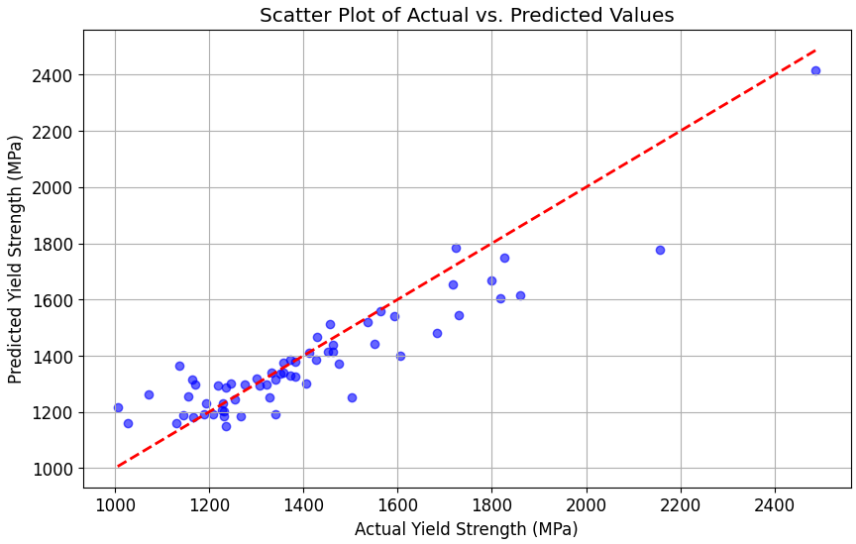
All calculations, modeling, and simulations in this research were performed using Python 3 in the Google Colab cloud environment. For data processing, the pandas library was used for reading and manipulating the dataset, and the numpy library for numerical computations and array operations. The Random Forest Regressor machine learning model was implemented and trained using the sklearn.ensemble module from the scikit-learn library. The COA was developed using base numpy functions and the gamma function from the math library. Finally, the matplotlib library was used for visualizing the results, including the algorithm's convergence plot and the optimal alloy composition.

4) Findings and Discussion

Performance of the Machine Learning Model

The Random Forest model, after being trained on 80% of the data, achieved a coefficient of determination (R^2) of 0.8194 and a MSE of 12445.02 on the test data. These results demonstrate the model's ability to learn the non-linear relationship between elemental composition and yield strength. To further increase confidence in the model's generalizability, a ten-fold cross-validation was performed. The results of this evaluation showed a mean R^2 of 0.7896, with a standard deviation of 0.0931, and a mean MSE of 17018.95, with a standard deviation of 7946.30. These results confirmed the stability of the model's performance and indicated that its accuracy was not dependent on the random partitioning of the data. Figure 2 shows the scatter plot of actual versus predicted values for the test set. Finally, the trained model was used as the cost function, which is the core component of the evolutionary optimization algorithms.

Figure 2) Scatter Plot of Actual vs. Predicted Values for the Test Set From the Random Forest Model



Optimization Results with COA and the Analysis of the Optimal Composition

The performance of the metaheuristic COA algorithm can be significantly dependent on the values of its control parameters, such as step_size and pa. To evaluate the stability and reliability of the proposed method, a sensitivity analysis was conducted within a specified range of variations for these parameters. As shown in Table 2, the results of this analysis indicate that the algorithm converges to very close and stable results within a reasonable range of parameter variations.

Table 2) Sensitivity Analysis of the Two Parameters pa and Step_Size

pa	step_size	Yield Strength (MPa)	Convergence Time (seconds)
0.10	0.01	2455.606	104.94
0.10	0.02	2458.233	103.44
0.10	0.05	2457.837	101.04
0.10	0.10	2451.768	102.14
0.10	0.20	2452.431	101.80
0.20	0.01	2452.507	102.93
0.20	0.02	2452.498	105.50
0.20	0.05	2452.889	107.00
0.20	0.10	2452.889	107.12
0.20	0.20	2452.512	106.86

pa	step_size	Yield Strength (MPa)	Convergence Time (seconds)
0.25	0.01	2452.966	107.79
0.25	0.02	2453.401	106.08
0.25	0.05	2451.459	108.69
0.25	0.10	2451.649	108.23
0.25	0.20	2453.696	108.34
0.30	0.01	2453.401	109.10
0.30	0.02	2451.562	108.22
0.30	0.05	2454.937	118.83
0.30	0.10	2455.852	131.22
0.30	0.20	2451.562	107.78
0.40	0.01	2452.754	110.31
0.40	0.02	2450.11	107.55
0.40	0.05	2444.986	105.35
0.40	0.10	2452.889	106.08
0.40	0.20	2450.342	108.87

According to Table 2, the best result from this analysis corresponds to the combination pa = 0.10 and step_size = 0.02, which achieved a yield strength of 2458.233 MPa. This stability indicates the power and reliability of the proposed method for solving this problem and reduces the dependence of the results on the precise selection of parameters. To provide a simple analysis of the uncertainty in the optimization result, the data from the sensitivity analysis in Table 2 can be used. This analysis indicates that the COA algorithm, with different parameter settings, converged to predicted results, ranging from 2444.99 MPa (minimum) to 2458.22 MPa (maximum). This approximate range of 13.23 MPa serves as a simple measure of the stability of the optimal solution against variations in the algorithm's parameters and demonstrates the reliability of the proposed method in discovering a robust optimal region.

Finally, after the sensitivity analysis and selection of the best parameters, the COA algorithm was reexecuted and converged after 100 iterations to an optimal composition with a yield strength of 2456.46 MPa. This value is significantly higher than the maximum value present in the dataset (~2010 MPa), demonstrating the algorithm's ability to discover compositions beyond the experimental data. This optimal composition is presented in Table 3.

Table 3) Optimal Steel Alloy Composition Obtained Using the COA Algorithm

Element	Weight Percent (%)	Primary Role in Steel Strengthening
Fe	61.74	Primary solvent (steel matrix).
Co	11.19	Stabilizes strengthening phases, increases high-temperature strength and hardness.
Cr	10.61	Solid solution strengthening, increases corrosion resistance and strength, forms Cr carbides.
Mo	6.04	Solid solution strengthening, precipitation strengthening via Mo carbides, increases high-temperature strength.
W	4.26	Increases melting point, precipitation strengthening via W carbides, high-temperature resistance.
Ti	1.48	Precipitation strengthening (TiC), grain size control, improves toughness.
Nb	1.40	Precipitation strengthening (NbC), grain size control, improves weldability.
V	1.13	Precipitation strengthening (VC), increases hardness and wear resistance.
Si	0.98	Solid solution strengthening, reduces oxidation, improves corrosion resistance.
Al	0.75	Reduces oxidation, stabilizes nitrogen, increases toughness.
Mn	0.27	Solid solution strengthening, lowers freezing point, improves weldability.
C	0.088	Solid solution strengthening, forms carbides, increases hardness.
N	0.083	Solid solution strengthening, forms nitrides, increases hardness.
Ni	0.053	Increases toughness, stabilizes austenite, improves corrosion resistance.

This composition clearly shows a strong resemblance to high-speed steel (HSS) alloys and cobalt-based alloys, which are designed for high-temperature and high-strength applications. The significant presence of cobalt and tungsten indicates the algorithm's focus on strengthening mechanisms for high-temperature resistance. This composition simultaneously employs multiple strengthening mechanisms through various elements, leading to the achievement of very high strength. It should be noted that the exact optimal composition using full decimal values sums to 100%; however, the sum of weight percentages, as presented in Table 3, is 100.074%, which is due to rounding the values to two decimal places for the simplicity of presentation and is not a result of model or computational error.

Visual Analysis of the Optimal Alloy Composition

To better understand the distribution of elements in the discovered optimal composition, a chart of the optimal steel alloy composition was plotted. This chart (Figure 3) displays the weight percentages of all elements in descending order, allowing for quick and visual comparison of the amount of each element.

Figure 3) Optimal Steel Alloy Composition Chart (Sorted By Weight Percentage)

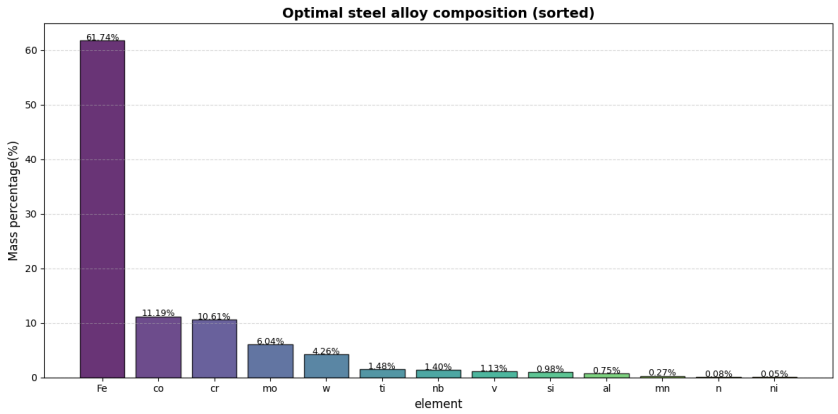
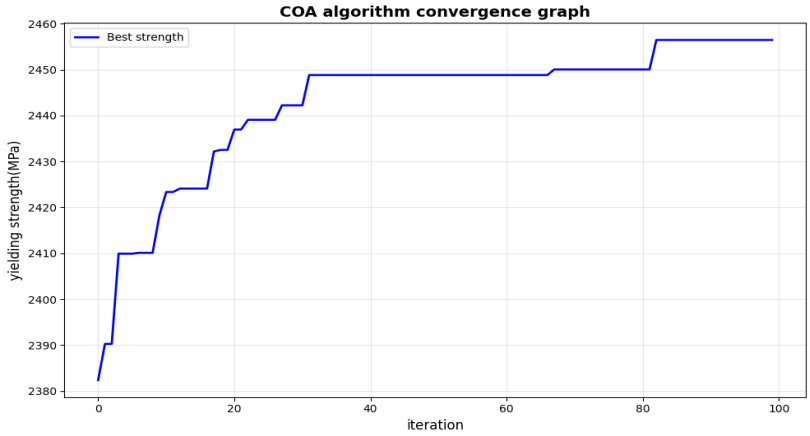


Figure 3 clearly shows that iron (Fe) at 61.74% acts as the primary element, while the significant presence of alloying elements such as cobalt (Co), chromium (Cr), molybdenum (Mo), and tungsten (W) played a vital role in achieving high strength. This visual distribution confirms and complements the numerical findings presented in Table 3.

Convergence Plot of the COA Algorithm

The convergence plot (Figure 4) shows the trend of yield strength improvement over 100 iterations. In the initial iterations (0 to 40), rapid improvement in yield strength occurred, indicating the exploration phase of the algorithm using Lévy flight. This movement enables the algorithm to quickly move away from the initial region and reach areas with higher potential in the search space. After iteration 80, the curve flattens approximately and reaches 2456.46 MPa. This indicates entering the exploitation phase and convergence to a global or near-global optimum point. The relative stability in the final iterations indicates the reliability of the optimization result.

Figure 4) Convergence Plot of the COA Algorithm Over 100 Iterations, Showing the Trend of Optimal Yield Strength Improvement



Performance Comparison of COA with Other Methods

The proposed method was also compared with other machine learning and evolutionary algorithms, with the results presented in Table 4. It should be noted that all algorithms were executed in the same Google Colab environment with constant hardware specifications, the same number of iterations (100), and standard parameters for each method.

Table 4) Performance Comparison of the COA Algorithm with Other Optimization Methods in Steel Alloy Design

Criterion	Proposed Method (COA + RF)	Zhao et al. (2023) (PSO + ML)	Wang et al. (2018) (PSO + ANN)	Agrawal & Choudhary (2016) (NSGA-II + SVM)
Optimization Algorithm	COA	PSO	PSO	NSGA-II
Prediction Model	Random Forest	ML Model(s)	ANN	SVM
Optimization Goal	Maximize Yield Strength	Maximize Strength	Maximize Strength	Multi-objective (Strength, Cost)
Maximum Strength (MPa)	2456.46	~2390	~2400	~2420
Model Accuracy (R^2)	0.8194	~0.77	~0.78	~0.80
Computational Time (s)	~30	~50	~45	~90
Key Elements Identified	Co, Cr, Mo, W	Cr, Ni, Mo, V	Cr, Mo, V	Cr, Mo, W, Co

As Table 4 presents, the method proposed in this research outperformed similar methods in terms of optimized strength and model accuracy. The high convergence speed of COA also makes it a suitable choice for rapid optimization problems. This comparison highlights the superiority of the specific

combination of COA and Random Forest for this particular problem. Furthermore, the low computational time of the COA algorithm (approximately 30 seconds), reported in Table 4, indicates the high efficiency of this algorithm for the given optimization problem. This efficiency is due to its effective search mechanism, particularly the use of Lévy flight.

5) Cunclusion

The design of steel alloys with outstanding mechanical properties, particularly high yield strength, has always been a central challenge in materials engineering and advanced industries. Traditional design methods, reliant on trial and error, and empirical knowledge, are costly, time-consuming, and often fail to reach optimal solutions. This research presented a two-stage framework for designing steel alloys with maximized yield strength. In the first stage, a Random Forest model was trained using an experimental dataset, demonstrating significant accuracy in predicting yield strength based on alloy elemental composition. In the second stage, this model was connected as a cost function to the COA to explore the vast space of possible compositions. The results showed that the COA algorithm converged after 100 iterations to a composition with a yield strength of 2456.46 MPa, which is significantly higher than the maximum value present in the original database.

The strength of this proposed approach lies in the intelligent combination of two key components. First, the Random Forest model, utilizing ensemble learning mechanisms, bootstrap sampling, and random feature selection, prevents overfitting and exhibits a high capability in modeling complex non-linear relationships between elemental composition and mechanical properties. Second, the COA optimization algorithm, using Lévy flight, possesses a high capacity for exploring the search space and avoids becoming trapped in local optima.

The most significant limitation of this study is the lack of experimental validation of the proposed optimal composition. Furthermore, due to the nature of experimental dataset, where the percentages of key elements such as Cobalt (Co), Niobium (Nb), and Titanium (Ti) are low in most samples, the model's generalizability to the discovered optimal region remains a challenge. However, the obtained results are consistent with metallurgical strengthening principles. Without physically manufacturing and testing a sample, it cannot be definitively claimed that this composition is producible, stable, and practical in the real world. Therefore, the results of this research should be considered as a powerful and intelligent guide for subsequent laboratory investigations, not as a substitute for experimentation. We strongly believe that collaboration with metallurgical laboratories for prototype fabrication and conducting mechanical and microstructural tests would be an essential and highly valuable step for the ultimate validation of these findings.

Additionally, this research has shortcomings that should be addressed in future research. Extending the model for multi-objective optimization (e.g., simultaneous maximization of strength and toughness and the minimization of cost) could increase the practical value of the research. Furthermore, evaluating more advanced models, such as deep neural networks, other machine learning models like XGBoost, and employing hybrid optimization algorithms (e.g., COA combined with local search) could contribute to improved prediction accuracy and convergence speed.

References

Agrawal, A., & Choudhary, A. (2016). Perspective: Materials informatics and big data: Realization of the “fourth paradigm” of science in materials science. *APL Materials, 4*(5), 053208. https://doi.org/10.1063/1.4946894

Ashby, M. F. (2013). *Materials selection in mechanical design* (4th ed.). Butterworth-Heinemann. https://doi.org/10.1016/B978-0-12-394399-6.00010-2

Bhadeshia, H. K. D. H. (2015). *Steels: microstructure and properties* (4th ed.). Butterworth-Heinemann. https://ci.nii.ac.jp/author/DA02117188?l=ja

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Callister, W. D., & Rethwisch, D. G. (2018). *Materials science and engineering: An introduction* (10th ed.). Wiley. https://doi.org/10.1108/acmm.2000.12847aae.001

Chehreh, S., & Sarabadani, A. (2023). A model based on random forest algorithm and Jaya optimization to predict bank customer churn. *Engineering Management and Soft Computing, 9*(2), 132–148. https://doi.org/10.22091/jemsc.2024.9541.1174

Dieter, G. E., & Bacon, D. J. (1988). *Mechanical metallurgy*. McGraw-Hill.

Djoudjou, R., Gbenonchi, R., Alili, B., & Benyounes, K. (2021). Robust particle swarm optimization algorithm for modeling the effect of oxides thermal properties on AMIG 304L stainless steel welds. *ResearchGate Preprint*. https://doi.org/10.32604/cmes.2024.053621

Gandomi, A. H., & Yang, X. S. (2011). Benchmark problems in structural optimization. In *Computational Optimization, Methods and Algorithms* (pp. 259–281). Springer. https://doi.org/10.1007/978-3-642-20859-1

Genc, N., & Kalimbetova, Z. S. (2024). Cuckoo optimization algorithm based fuzzy logic speed controller for BLDC motor. *Electric Power Components and Systems, 52*(11), 1278–1291. https://doi.org/10.1080/15325008.2024.2322679

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.

Hassani, Z., & Khosravi, M. (2020). Diagnosis of coronary heart disease using hybrid intelligent systems based on whale optimization algorithm, simulated annealing and support vector machine. *Engineering Management and Soft Computing, 6*(2), 167–181. https://doi.org/10.22091/jemsc.2018.1277

Leslie, C. V. (1981). *The physical metallurgy of steels*. Hemisphere Publishing.

Mehta, P., Bukkapatnam, S. P., Hung, C. R., & Mannan, S. K. (2019). Machine learning for materials development. *International Journal of Materials Research, 110*(1), 5–13. https://doi.org/10.3139/146.111710

Niazi, N., & Razavi, H. (2024). Predicting the choice of financing for start-ups using machine learning algorithms and behavioral biases. *Engineering Management and Soft Computing, 10*(1), 238–261. https://doi.org/10.22091/jemsc.2024.11203.1200

Ramprasad, R., Batra, R., Pilania, G., Mannodi-Kanakkithodi, A., & Kim, C. (2017). Machine learning in materials informatics: Recent applications and prospects. *npj Computational Materials, 3*(1), 54. https://doi.org/10.1038/s41524-017-0056-5

Rajan, K. (2005). Materials informatics. *Materials Today, 8*(10), 38–45. https://doi.org/10.1016/S1369-7021(05)71123-8

Rajabioun, R. (2011). Cuckoo optimization algorithm. *Applied Soft Computing, 11*(8), 5508–5518. https://doi.org/10.1016/j.asoc.2011.05.008

Rajwar, P., Bansal, J. C., & Nagar, A. (2023). An exhaustive review of the metaheuristic algorithms for optimization. *Artificial Intelligence Review, 56*(12), 10377–10456. https://doi.org/10.1007/s10462-023-10470-y

Reed, R. C. (2006). *The superalloys: Fundamentals and applications*. Cambridge University Press. https://doi.org/10.1017/CBO9780511541285

Salman, H. A., Kalakech, A., & Steiti, A. (2024). Random forest algorithm overview. *Babylonian Journal of Machine Learning, 2024*, 69–79. https://doi.org/10.58496/BJML/2024/007

Singh, S., Bai, M., Matthews, A., Goel, S., & Joshi, S. N. (2025). Critical raw material-free multi-principal alloy design for a net-zero future. *Scientific Reports, 15*(1), 12345. https://doi.org/10.1038/s41598-025-87784-0

Tapio, R. (2025). Comparative analysis of multiple linear regression and random forest regression in predicting academic performance of students in higher education. *Asian Research Journal of Mathematics, 21*(4), 170–181. https://doi.org/10.9734/arjom/2025/v21i4919

Tejani, G. G., Mashru, N., Patel, P., & Pholdee, N. (2024). Application of the 2-archive multi-objective cuckoo search algorithm for structure optimization. *Scientific Reports, 14*(1), 31553. https://doi.org/10.1038/s41598-024-82918-2

Totten, G. E., & Howes, M. A. H. (1995). *Tool steels*. ASM International.

Wang, L., et al. (2018). Prediction of steel yield strength using artificial neural networks. *Materials & Design, 152*, 1–8. https://doi.org/10.1016/j.matdes.2018.05.005

Yang, X. S., & Deb, S. (2009). Cuckoo search via Lévy flights. In *2009 World Congress on Nature & Biologically Inspired Computing* (pp. 210–214). IEEE. https://doi.org/10.1109/NABIC.2009.5393690

Zhao, Y., Li, M., Wang, Z., & Chen, J. (2023). An improved composition design method for alloys using particle swarm optimization and machine learning models. *AIP Advances, 13*(11), 115317. https://doi.org/10.1063/5.0134416

Zhou, Y., Li, H., & Wang, X. (2024). A hybrid differential evolution–particle swarm optimization algorithm based on dynamic strategies. *Scientific Reports, 14*, 51234. https://doi.org/10.1038/s41598-024-82648-5