Journal Européen des Systèmes Automatisés
Vol. 59, No. 5, May, 2026, pp. 1363-1371
Journal homepage: http://iieta.org/journals/jesa

A CALM-ANN Hybrid Framework for Accurate Prediction in Complex Time Series

Samia Noureddine[1,2*], Abir Betka[2,3], Hani Benguesmia[4], Abida Toumi[2,5], Nazim Kazar[6], Naima Rahoua[5]

[1] Department of Industrial Pharmacy, Pharmacy Faculty, University of Health Sciences, Algiers 16000, Algeria
[2] Laboratory of Vision and Communication Systems, University of Biskra, Biskra 07000, Algeria
[3] Department of Electrical Engineering, University of El-Oued, El-Oued 39000, Algeria
[4] Laboratory, Electrical Engineering (LGE), Faculty of Technology, University of M'sila, University Pole, M'sila 28000, Algeria
[5] Department of Electrical Engineering, University of Biskra, Biskra 07000, Algeria
[6] The Research Center in Computer Sciences, Paris 1 Panthéon-Sorbonne University, Paris 75231, France

Corresponding Author Email: s.noureddine@univ-alger.dz

https://doi.org/10.18280/jesa.590515

Received: 6 March 2026
Revised: 15 May 2026
Accepted: 26 May 2026
Available online: 31 May 2026

Keywords:
optimization, Chaotic Augmented Lagrange Multiplier, Artificial Neural Networks, time series, prediction

ABSTRACT

In metaheuristics, local minima and instabilities are observed when dealing with nonlinear and chaotic data, making it difficult for them to explore the solution space adequately. In contrast, Multilayer Perceptron Artificial Neural Networks (MLP-ANNs) are susceptible to the impact of weight parameters and are less stable when dealing with chaotic and dynamic data. In this context, this work proposes a novel approach that combines the use of the Chaotic Augmented Lagrange Multiplier (CALM) in Artificial Neural Networks (CALM-ANN) during the learning process to overcome the identified drawbacks. In this approach, three steps are followed: the use of chaotic searches for escaping local minima, the use of guided initialization to improve starting conditions, and the use of Nelder-Mead for incorporating local searches. In addition, the CALM-ANN was developed as a specialized modification for time series forecasting, applied to International Airline Passengers (IAP) and World Health Organization (WHO) figures available in the Neglected Tropical Diseases (NTD) dataset. From the experimental observations, it can be clearly identified that the proposed work outperforms: Long Short-Term Memory (LSTM), Seasonal Autoregressive Integrated Moving Average (SARIMA), Prophet, and Support Vector Machines (SVM) methods in dealing with chaotic and complicated data sets, resulting in root mean square error (RMSE) values of 28.54 and 9.07 and the coefficient of determination (R^2) values of 0.854 and 0.913 for transportation and epidemiological datasets, respectively. The primary contribution of this study is the integration of chaotic search and augmented Lagrangian optimization into the training of ANNs within a unified framework.

1. INTRODUCTION

With increasingly complex systems, classic optimization techniques are actually experiencing the limitations of their applicability. Although they possess optimality, this is limited by computational complexity [1]. Contemporary metaheuristics, on one hand, possess intelligent exploration ability, but they have the disadvantage of exploration and exploitation trade-off, especially in time series prediction tasks, whereby nonlinearity and noise pose complex problems of pattern relevance extraction [2, 3].

Our work constitutes an original contribution because it presents the Chaotic Augmented Lagrange Multiplier Optimization Algorithm combined with Artificial Neural Networks (CALM-ANN), a hybrid model that improves this classic paradigm [4, 5], as we rigorously validate our model on test functions [6] and, at the same time, demonstrate its application on complex time series. That is, series that cover a broad spectrum of problems, ranging from air traffic data—often considered a classic example due to its pronounced seasonality and trend [7] to time series related to Neglected Tropical Diseases (NTD) [8], which are characterized by noise and the irregular behavior of epidemics, as well as all other series that fall between these two extremes.

The originality of CALM-ANN is based on three fundamental pillars: a hybrid optimization scheme integrating augmented Lagrange multipliers and chaotic dynamics to combine mathematical rigor and robust exploration; a carrier wave initialization mechanism ensuring stable convergence; and exhaustive validation combining classic benchmarks (Rosenbrock, Rastrigin) and real-world temporal prediction.

By implementing CALM-ANN on various time series, we demonstrate that our methodology outperforms previous approaches, as it can transform the weaknesses of classical paradigms into opportunities for innovation.

The primary contributions of the study include the following:

1. The CALM optimization strategy, which is a robust

technique utilizing random initialization, chaotic exploration, and the augmented Lagrangian method; and
2. The CALM-ANN optimization strategy, whereby an ANN is optimized via random initialization, chaotic exploration, and the augmented Lagrangian method.

The results clearly show that the utilization of chaotic exploration contributes to achieving stability in the convergence process, while constraint-based optimization guarantees dependable predictive power. In this regard, the suggested model, CALM-ANN, successfully integrates exploration and exploitation in all instances, thus emphasizing the importance of each element of the model.

This study introduces methodological and empirical contributions to hybrid optimization and neural time-series forecasting.

First, we propose CALM, a constraint-aware hybrid optimizer combining augmented Lagrange multipliers, chaos-driven exploration, and feasibility-guided initialization within a unified framework.

Second, we reformulate ANN learning as a constrained optimization problem and integrate CALM directly into the training process, allowing weights and biases to be learned through a stability-oriented search mechanism rather than gradient-only updates.

Third, we evaluate the proposed framework both on standard optimization benchmarks and on real-world time series forecasting tasks, including transportation and epidemiological datasets characterized by nonlinearity, noise, and structural instability.

Fourth, we analyze the contribution of each component of the proposed framework through comparative experimentation and performance analysis. These contributions position the CALM-ANN framework as a robust alternative for learning and forecasting in complex dynamical systems. This paper is structured in a way that the next part is the theoretical context where the underpinning of the proposed approach is drawn. Then, the following part presents the experimentation and the detailed implementation of the CALM-ANN model along with the assessment done on different time series. Finally, the last section outlines the main conclusions and discusses future perspectives and potential research directions.

2. THEORETICAL CONTEXT

Comparative contextualization of CALM-ANN against GWO, ABC, and Long Short-Term Memory (LSTM) has been strengthened:

- CALM integrates constraint management and chaotic exploration within a unified learning mechanism, whereas GWO and ABC treat exploration and exploitation as separate phases.
- Unlike LSTM, which relies on gradient-based backpropagation and is sensitive to hyperparameter tuning, CALM-ANN reformulates neural network training as a constrained optimization problem, offering greater robustness to noisy and nonlinear data.

In this section, we present the theoretical foundations underlying our proposed approach. We first introduce the concepts and principles that motivate the development of the CALM-based framework, highlighting the limitations of traditional optimization and learning paradigms that inspired our contribution. This theoretical background serves as a basis for understanding the structure, mechanisms, and advantages of the CALM algorithm [9].

2.1 Description of the Chaotic Augmented Lagrange Multiplier Optimization Algorithm

The CALM method is structured around three complementary phases, each enhanced by a specific mathematical formulation:

•Phase 1: unique fusion: augmented Lagrange multipliers manage constraints, while chaotic maps ensure diversified exploration of the search space, optimization is based on the augmented Lagrangian function [10, 11] presents in Eq. (1):

$$\mathcal{L}(x,\lambda,\mu)=f(x)+\sum_{j=1}^{m}\lambda_j\, g_j(x)+\frac{1}{2}\sum_{j=1}^{m}\mu_j\, g_j(x)^2 \qquad (1)$$

where, $f(x)$ is the objective function, $g_j(x)$ are the constraints ($g_j(x) \leq 0$), λ_j and μ_j are the multipliers and penalty coefficients respectively.

•Phase 2: integration of Feasible Chaotic Weighting (FCW): chaotic weighting is based on the logistic map, used to guide exploration [10-12]:

$$z_{n+1}=rz_n(1-z_n);\qquad 3.57<r\leq 4 \qquad (2)$$

where, $z_n \in (0,1)$ is the chaotic state at iteration n, which this sequence guides the exploration to effectively cover the search space. FCW is then used to construct a robust initial point, limiting sensitivity to initial conditions before local optimization.

•Phase 3: updating multipliers and penalties [13]: parameters related to constraints are updated by:

$$\lambda_j^{(k+1)}=\lambda_j^{(k)}+\mu_j^{(k)}\, g_j(x^{(k)});\ \mu_j^{(k+1)}=\gamma\mu_j^{(k)},\ \gamma>1 \qquad (3)$$

$x^{(k)}$: solution obtained at iteration k, and γ: penalty increase factor, ensuring gradual convergence towards a feasible solution. Algorithm 1 demonstrates the CALM method's performance:

Algorithm 1 presents the pseudocode for the CALM algorithm, describing its three main phases and the complete execution of the algorithm.

Algorithm 1: Chaotic Augmented Lagrange Multiplier (CALM) Algorithm

Initialization: $(f;\ g;\ x_0;\ \lambda_0;\ \mu_0;\ z_0;\ r;\ \max\ \_iter)$
$k \leftarrow 0$
//AUGMENTED LAGRANGIAN FUNCTION:
$$\mathcal{L}(x,\lambda,\mu)=f(x)+\sum_{j=1}^{m}\lambda_j\, g_j(x)+\frac{1}{2}\sum_{j=1}^{m}\mu_j\, g_j(x)^2$$
//CALM ALGORITHM:

While $k < \max\ \_iter$ && $not\ converged$
// **Phase 1**: chaotic exploration
$z \leftarrow r * z * (1-z)$ // update chaotic state (logistic map)
$x_{cand} \leftarrow ProposeForm(x,z)$ // generate chaos-

guided candidate

// **Phase 2:** robust starting point via FCW
$x_{init} \leftarrow FCW(\mathcal{L}, x_{cand})$ // improvement of candidate by FCW

// **Phase 3:** Local optimization + parameter update
$x_{new} \leftarrow NelderMead\_Optimize(\mathcal{L}, x_{init})$ // Local optimization of the augmented Lagrangian function

// Update multipliers and penalties
$$\lambda_j \leftarrow \lambda_j + \mu_j \times g_j(x_{new})$$
$$\mu_j \leftarrow \gamma \times \mu_j$$
// convergence
$$x \leftarrow x_{new}$$
$$k \leftarrow k + 1$$
return x

2.2 Multilayer perceptron neuron network optimized by the Chaotic Augmented Lagrange Multiplier

We apply the CALM algorithm to optimize the training of a multilayer perceptron neuron network optimized (CALM-ANN, or Chaotic Augmented Lagrange Multiplier and Multilayer Perceptron (CALM-MLP)). CALM generates the weights and biases that minimize the error, and these parameters are then used in the network.

The process is iterative and continues until the stopping criterion is met. The quality of the solution depends on the initial weights, biases and the size of the population [14].

Initialization: $X = \lfloor w_{i,j},\ b_{i,j} \rfloor_{M \times (nw+nb)}$ with M the population size. nw, and nb numbers of weights, and biases respectively, the network comprises: I input neurons (number of features), H = 2I + 1 hidden neurons according to Kolmogorov's theory, proposed by Nakamura [15], and O output neurons (number of classes), the total dimension is:

$$D = (I \times H) + (O \times H) + H + O \quad (4)$$

The sigmoid function is used in the hidden and output layers for its differentiability and its interval [0, 1], suitable for probabilistic outputs.

Evaluation: Performance is measured using the root mean square error (RMSE), the absolute mean error (MAE), the coefficient of determination (R^2), and Pi-Score presents the performance index-score (Peak Periods Score) evaluating the difference between the actual values y and the predicted values$\hat{y}$ [15], according to Eqs. (5)-(8):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)} \quad (5)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |\, y_i - \hat{y}_i \,| \quad (6)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \dot{y})^2} \quad (7)$$

$$Pi - Score = \alpha(1 - nRMSE) + (1 - \alpha) \times R^2; \alpha = 0.5 \quad (8)$$

where, n is the number of observations.

The factors that affect the computational complexity of CALM-MLP primarily involve neural network evaluation and local optimization.

In every iteration, the algorithm's complexity is O(N, W), where N stands for the number of patterns, and W stands for the total number of weights and biases.

The per-iteration complexity of CALM-MLP is O (N x W), where N is the number of training samples and W is the total number of weights and biases. The chaotic update step adds only O(W) overhead per iteration. The Nelder-Mead step introduces $O(W^2)$ cost per call but is invoked infrequently. In comparison, LSTM back propagation through time has complexity $O(N \times T \times H^2)$, scaling poorly with sequence length. Scalability to very large datasets remains a direction for future work.

Figure 1 illustrates the CALM-MLP flowchart, showing how the CALM model is integrated into the neural network architecture used in the time series experiment.

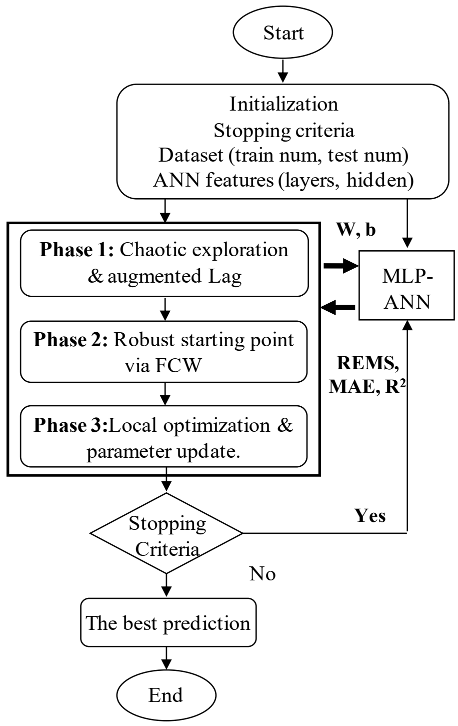

Figure 1. Multilayer perceptron neuron network optimized by the Chaotic Augmented Lagrange Multiplier in Artificial Neural Networks (CALM-ANN) method flowchart

2.3 Time series analysis

A time series is a sequence of observations $\{x_1, x_2, \ldots\ldots, x_n\}$ indexed by time (minute, hour, day, year, etc.), the objective of the analysis is to model these data in order to extract their structure and perform tasks such as forecasting, regression, classification, or explanation [16-18].

Forecasting can be formulated as an approximation problem [19]:

$$y(t+d) \approx f\left(y(t), y(t_1), y(t_2), \ldots\ldots, y(t_n)\right) \quad (9)$$

where, d denotes the forecast horizon, f: Rn → R estimation function of $y(t+d)$ of y at time $(t+d)$, given n observations of y up to time t.

In practice, future values are often affected by random noise [20]:

$$y(t+d)=f\left(y(t),y(t_1),y(t_2),\ldots\ldots,y(t_n)\right)+\in; \quad \in\approx N\left(0,\alpha^2\right) \tag{10}$$

where, N is a normal distribution with meaning zero and variance α^2.

2.4 Positioning relative to existing methodologies

Existing hybrid optimization methodologies that use metaheuristics in conjunction with Multilayer Perceptron Artificial Neural Networks (MLP-ANNs) have typically done so through hyperparameters or weight initialization. However, most of these methodologies have addressed constraint handling, exploration, and learning as discrete processes. The proposed framework in this paper integrates constraint management, chaotic exploration, and local optimization in a single learning framework. In addition to that, while chaotic metaheuristics and augmented Lagrangian optimization have been studied as individual methodologies in the literature, their use in conjunction with neural networks in the context of time series prediction is less well explored. The proposed framework in this paper seeks to address that by incorporating feasibility control and exploration in the learning of the MLP-ANN.

3. EXPERIMENTAL SECTION

To ensure reproducibility and reliability, a single and unified protocol was used for all experiments. Each optimization method has been executed several times in order to account for stochastic effects. For evaluating the performance of the models, we used the RMSE, MAE, and R^2 measures, and therefore, a fair comparison between the methods and datasets has been ensured. For all models, the data division used for training and testing has been fixed at 70 and 30, respectively, and all models have been subjected to the same preprocessing steps.

For a fair evaluation, the parameters of the compared algorithms were set according to the literature recommendations. All experiments have been performed in the same computational conditions to exclude possible performance bias due to hardware differences. The experiments were implemented in MATLAB (Release 2018) on a PC running Windows 10 (64-bit) and the equipped with an Intel Core i5-7200U processor (2.50 GHz, dual-core, 4 logical processors) and 8 GB of RAM.

3.1 Tests on benchmark functions

The robustness of the method was validated using six benchmark functions [21], regrouped into unimodal high-dimensional functions (F1−F2), multimodal high-dimensional functions (F3–F4), and the multimodal low-dimensional functions (F5–F6), whose global optimum is known. For each function, the population size is set at 100 candidate solutions and the number of iterations is identical for all comparison methods, CALM was compared with established algorithms such as Particle Swarm Optimization (PSO), Gray Wolf Optimizer (GWO), Chaotic Algorithm (CA), Artificial Bee Colony (ABC), Sine Cosine Algorithm (SCA), and Genetic Algorithm (GA) [22, 23]. Performance was measured using three indicators: the mean, the standard deviation, and the relative ranking of the solutions. According to Table 1, our CALM method can find the global optimal solution, its variants and the other cited metaheuristics in all the benchmark functions, only for F6, it is classed in the second place after GWO algorithm. This means that CALM is able to effectively balance exploration and exploitation, preventing it from falling into local solutions. In contrast, GWO and GA performed very similarly to CALM, but were less stable.

Table 1. Optimization results obtained with Chaotic Augmented Lagrange Multiplier (CALM) and other metaheuristics for test functions

Functions		PSO	GWO	CA	ABC	SCA	GA	CALM
$f_1=\sum_{i=1}^{n} x_i^2$	Mean	0.3970	0	0.14262	1.46E-106	3.49E-19	0	0
	Std	0.3209	0	0.00335	7.32E-106	2.59E-19	0	0
	Rank	7	1	6	4	5	1	1
$f_2=\sum_{i=1}^{n}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$	Mean	0.0352	0	0.03905	1.46E-115	6.03E-22	0	0
	Std	0.1624	0	0.01022	7.09E-115	5.70E-22	0	0
	Rank	6	1	7	4	5	1	1
$f_3=-20exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2}\right)-exp\left(\frac{1}{D}\sum_{i=1}^{D}cos(2\pi x_i)\right)+20+e$	Mean	13.4428	8.88e-16	20.06828	1.98E-05	19.3972766	8.8818e-16	8.8818e-16
	Std	0.7387	0	0.225689	2.67E-05	0.44255796	0	0
	Rank	5	1	7	4	6	1	1
$f_4=\frac{1}{4000}\sum_{i=1}^{D}x_i^2-\prod_{i=1}^{D}cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	Mean	43.5010	0	29.95350	0.00994193	1.16815179	0	0
	Std	11.9431	0	0.559410	0.0411774	0.06518776	0	0
	Rank	6	1	5	2	4	1	1
$f_5=-\sum_{i=1}^{4}c_i\,exp\left(-\sum_{j=1}^{6}a_{ij}(x_j-p_{ij})^2\right)$	Mean	-3.2602	-3.2504	-1.89728	-3.31404604	-3.32189167	-3.2839	-3.320
	Std	0.0606	0.05962	0.421318	0.02773574	0.0003203	0.0566	0.1e-03
	Rank	6	7	8	3	2	4	1

$f_6 = -\sum_{i=1}^{10}[(X - a_i)(X - a_i)^T + c_i]^{-1}$	Mean	-8.5370	-10.534	-1,72097	-10.2298128	-5.12848079	-9.0222	-10.5
	Std	3.3054	7.28e-04	0,595891	1.11788668	3.63E-15	2.4782	0
	Rank	7	1	9	4	8	6	2

Note: Particle Swarm Optimization (PSO), Gray Wolf Optimizer (GWO), Chaotic Algorithm (CA), Artificial Bee Colony (ABC), Sine Cosine Algorithm (SCA), and Genetic Algorithm (GA)

CALM achieves the best rank (1st) on five of the six functions, with GWO ranking 2nd on F6. The pairwise comparisons between CALM and GWO/ABC have been expanded; CALM constraint-aware exploration provides an advantage, particularly in functions with complex multimodal landscapes (F3K F6).

3.2 Application of the Chaotic Augmented Lagrange Multiplier and Multilayer Perceptron method to time series

In the second experiment, the proposed method was applied to a concrete time series regression problem and compared to other methods [24, 25] such as LSTM, Seasonal Autoregressive Integrated Moving Average (SARIMA), Prophet, and Support Vector Machines (SVM). The algorithms effectively predict future values from historical data. We used two datasets: International Airline Passengers (IAP) (1949–1960, 144 monthly observations) to test for trend and seasonality [7], and WHO's NTD to evaluate noise, gaps, and epidemic episodes [8]. For the IAP dataset, the test set consisted of the last 24 months (2 years), while the training set included the remaining observations. For the NTD dataset, the test set covered the last 52 weeks (1 year), and the training set consisted of the remaining data.

For the IAP dataset, strong multiplicative seasonality and trend component make it challenging for linear models such as SARIMA. For the NTD dataset, irregular epidemic episodes, missing observations, and high inter-week variability challenge models with rigid parametric assumptions. The chaotic search mechanism of CALM (via the logistic map) enables diverse exploration of the weight space, helping the neural network escape local optima arising from high variability and noise.

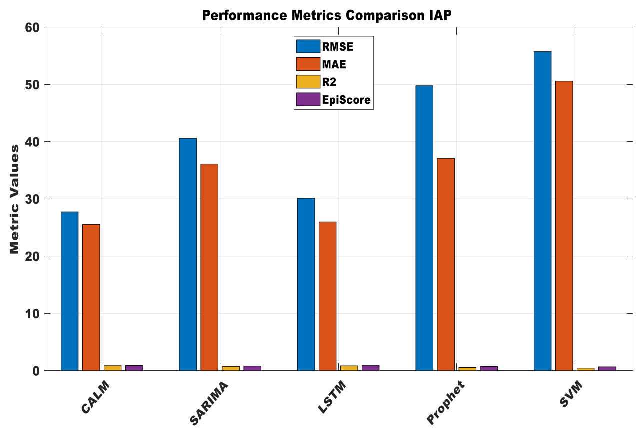

Figure 2. Performance metrics comparison- International Airline Passengers (IAP) data

3.3 Discussion of results

Table 2 and Figure 2 illustrate that the proposed hybrid approach CALM-MLP performs significantly better for predicting Airline Passengers dataset, yielding a value of RMSE = 28.54 compared to other methods. Also, the CALM-MLP gave a value of MAE = 24.82 outperforming SARIMA, LSTM, Prophet, and SVM which gave MAE values of 42.02, 32.14, 49.77, and 51.79 respectively, furthermore, CALM-MLP yielded a coefficient of determination $R^2 = 0.854$ and a peak-score Pic-Score=0.884, these results indicate that our method can accurately estimate seasonal peaks and real information within the dataset, as visually confirmed in Figures 3 and 4. The LSTM method also produced results close to those of CALM- MLP. In contrast, the other methods showed lower coefficients of determination and weaker performance for peak periods, although the SARIMA method is generally considered as a good model for time series data.

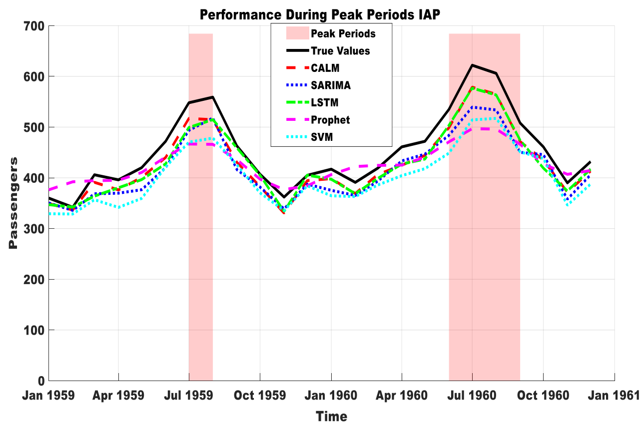

Figure 3. Performance during the season peak - International Airline Passengers (IAP) data

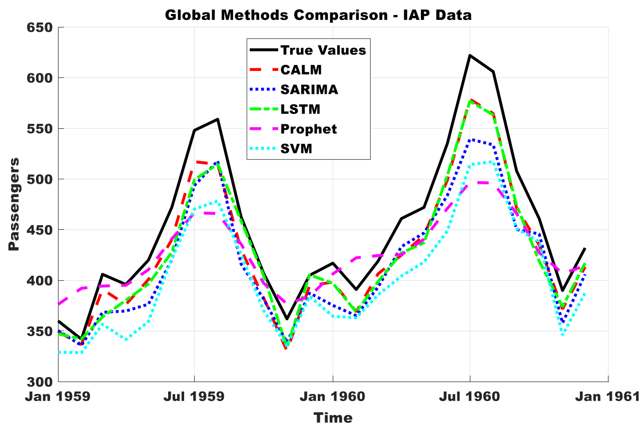

Figure 4. Global comparison methods- International Airline Passengers (IAP) data

Figure 5 shows the reliability curve of the different prediction models using the IAP dataset. Most points are located near the diagonal, which represents a perfect prediction. The distribution of prediction errors is displayed in

Figure 6, where the suggested approach shows fewer extreme values and a smaller error range than the other models. Furthermore, Figure 7 presents the temporal evolution of prediction performance, demonstrating that proposed method maintains relatively stable accuracy over different periods, while the other methods show greater fluctuations.

Table 2. Comparison of performance on the International Airline Passengers (IAP) enrollment time series

Method	RMSE	MAE	R^2	Pic-Score
CALM-MLP	28.54	24.82	0.854	0.884
SARIMA	42.02	38.55	0.683	0.787
LSTM	32.14	29.62	0.815	0.860
Prophet	49.77	37.07	0.556	0.729
SVM	51.79	47.30	0.519	0.701

Note: Support Vector Machines (SVM); root mean square error (RMSE); absolute mean error (MAE); Long Short-Term Memory (LSTM); Seasonal Autoregressive Integrated Moving Average (SARIMA); Chaotic Augmented Lagrange Multiplier and Multilayer Perceptron (CALM-MLP)

These results indicate that CALM-MLP adapts better to variations in the data and offers more consistent forecasting performance and stable prediction performance over time.

A second publicly available dataset on NTD dataset was analyzed in order to strengthen the experimental validation of the proposed method.

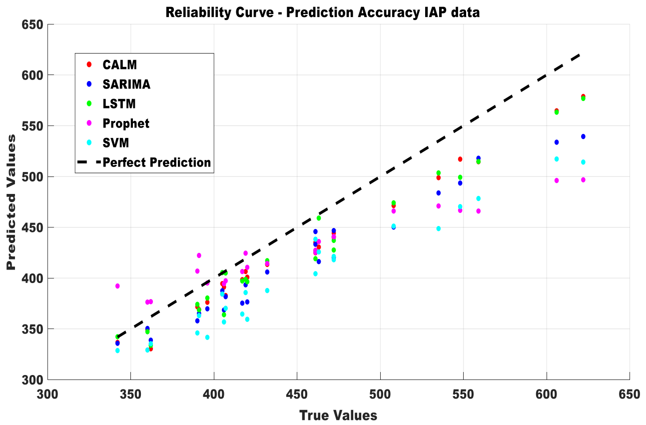

Figure 5. Reliability curve - prediction accuracy - International Airline Passengers (IAP) data

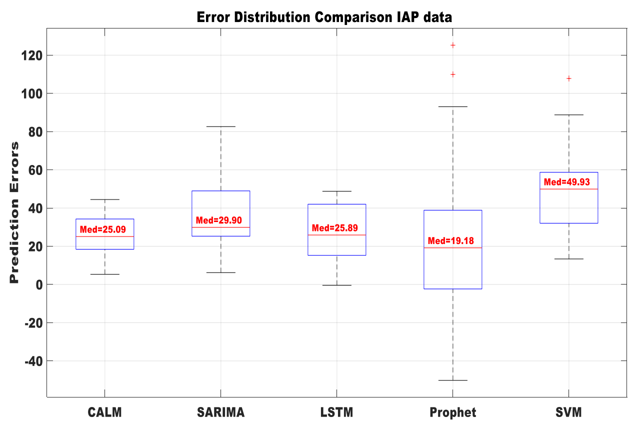

Figure 6. Prediction error distribution-International Airline Passengers (IAP) data

Table 3 and Figure 8 represent the performance metrics of evaluation models and reveal the remarkable performance of CALM-MLP in predicting NTD, CALM-MLP achieved very low values of RMSE = 9.07, and of MAE = 7.75, along with strong performance of $R^2 = 0.897$, and predicted the epidemic peak of 0.914. These results illustrate that the proposed approach can precisely predict the epidemic peak and real values of NTD dataset. The LSTM model yielded results close to those of CALM-MLP. On the other hand, the remaining methods showed weaker performance. Figure 9 highlights the particular accuracy of the proposed method during epidemic phases, which strengthens its leading position. Figures 10 and 11 demonstrate their ability to accurately track actual temporal trends compared to the other methods. The prediction error distribution shown in Figure 12 confirms the robustness of its predictions. In addition, Figure 13 represents the temporal performance evolution and shows that the proposed method maintains good performance over time.

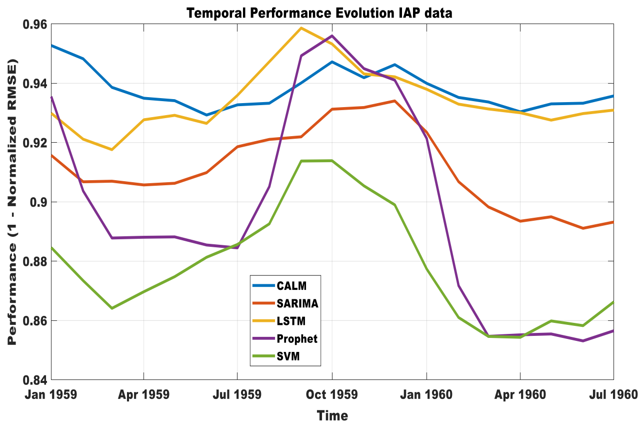

Figure 7. Temporal performance evolution- International Airline Passengers (IAP) data

Table 3. Comparison of performance on the WHO's Neglected Tropical Diseases (NTD) enrollment time series

Method	RMSE	MAE	R^2	Epi-Score
CALM-MLP	9.07	7.75	0.897	0.914
SARIMA	13.93	11.18	0.757	0.836
LSTM	12.73	10.55	0.797	0.857
Prophet	29.83	25.17	0.000	0.453
SVM	9.82	16.06	0.508	0.708

Note: Long Short-Term Memory (LSTM); Seasonal Autoregressive Integrated Moving Average (SARIMA); root mean square error (RMSE); root mean square error (RMSE); Chaotic Augmented Lagrange Multiplier and Multilayer Perceptron (CALM-MLP)

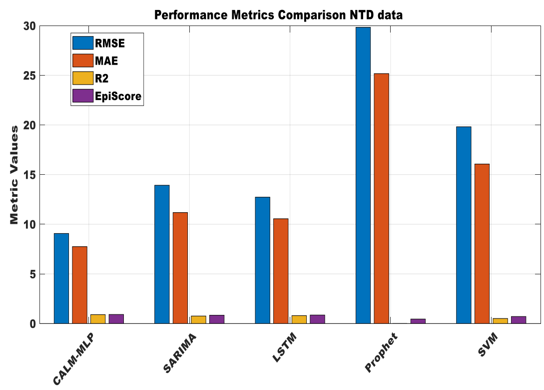

Figure 8. Performance metrics comparison- Neglected Tropical Diseases (NTD) data

Both IAP and NTD datasets exhibit nonlinear dynamics that violate SARIMA core assumptions, yielding 0.683 and RMSE = 42.02 on the IAP dataset. Regarding LSTM: its competitive performance (RMSE = 32.14, 0.815 on IAP) is expected given its recurrent memory mechanism. However, LSTM is sensitive to learning rate, hidden units, and training epochs. CALM-ANN performance advantage is expected to widen in scenarios with smaller training sets or higher noise levels, where LSTM gradient-based training becomes less stable.

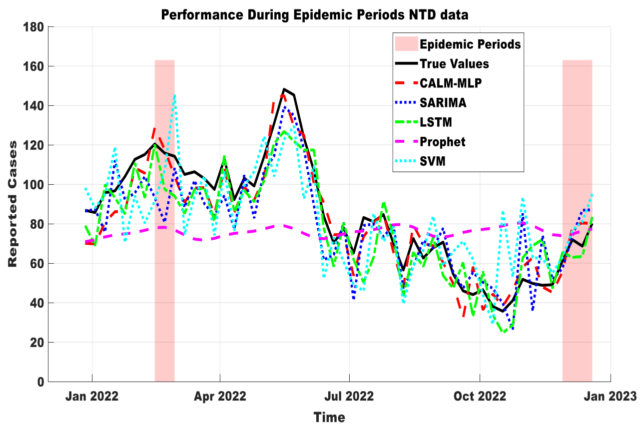

Figure 9. Performance during epidemic periods- Neglected Tropical Diseases (NTD) data

For larger datasets (thousands of observations), the current implementation would benefit from parallelization of the population-based evaluation phase. Potential extensions include:

•GPU-accelerated population evaluation;

•Mini-batch variants of the Lagrangian update for streaming or large-scale data;

•Adaptive population sizing strategies to balance exploration depth with computational budget. CALM-ANN is particularly well-suited for moderate-scale datasets with strong nonlinearity.

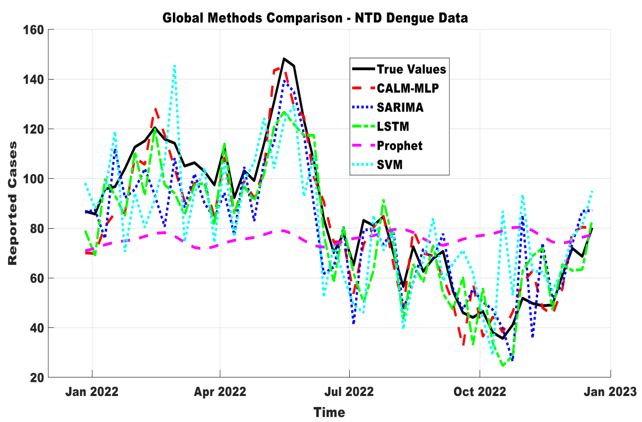

Figure 10. Global comparison methods - Neglected Tropical Diseases (NTD)

Even though the CALM-ANN model is validated using specific time series related to transportation and epidemiology, its wider applicability allows for its application to other kinds of time series. The properties of noise, nonlinearity, and variability inherent in time series relating to finance, energy, and environment make this methodology applicable in various fields. However, some drawbacks of this methodology should not be overlooked. In the first place, a limited number of observations were used in this research. Future research would benefit from employing very large samples of time series. Secondly, an ANNs model employed here is rather simple, and further research could focus on more complicated ANNs, including those associated with deep learning techniques.

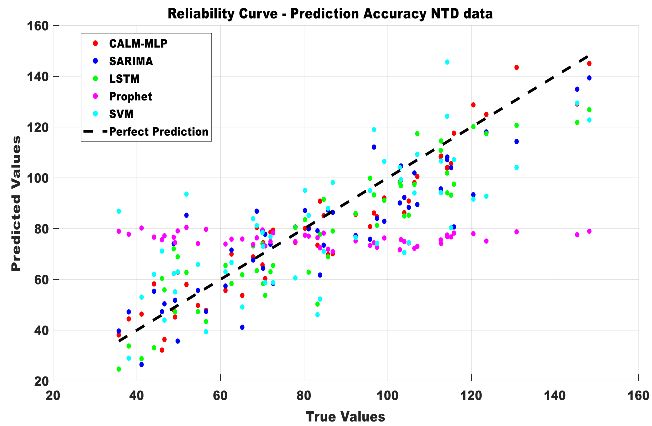

Figure 11. Reliability curve - prediction accuracy - Neglected Tropical Diseases (NTD) data

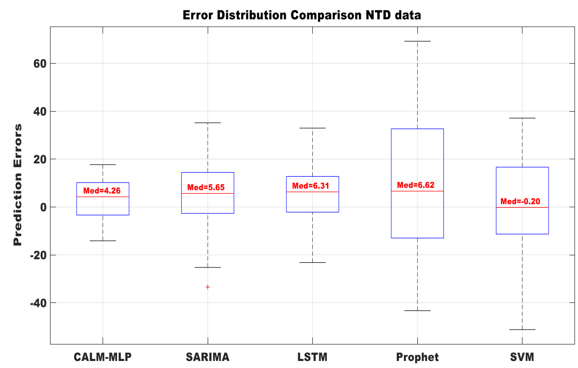

Figure 12. Prediction error distribution - Neglected Tropical Diseases (NTD) data

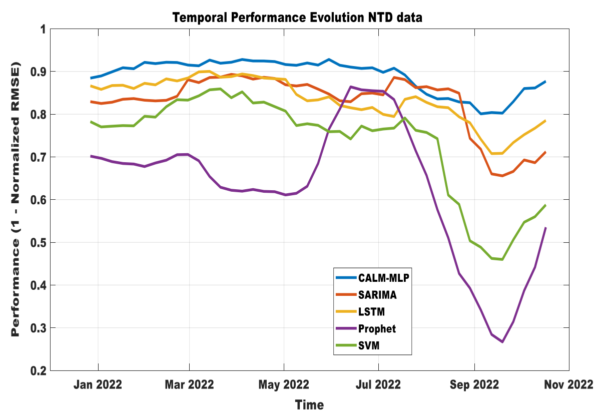

Figure 13. Temporal performance evolution - Neglected Tropical Diseases (NTD) data

4. CONCLUSION AND PERSPECTIVES

This article presented the CALM method, an optimization strategy combining improved Lagrange multipliers, chaotic maps, and the FCW mechanism, experimental results show that this approach offers a robust balance between exploration and exploitation and outperforms several benchmark test functions. Integrated into an ANN, it has also demonstrated good effectiveness for real time series, both in terms of accuracy and convergence speed.

The application of international airline passenger data confirms CALM-ANN's ability to model complex chains by highlighting trends and seasonality, and it has also been used on WHO data on NTD, further demonstrating its importance for epidemiological surveillance.

For transportation systems, CALM-ANN's ability to accurately capture seasonal peaks (Pic-Score = 0.884) makes it a strong candidate for passenger demand forecasting, supporting capacity planning and resource allocation. For epidemiological surveillance, the model's performance during epidemic peaks (Epi-Score = 0.914 on NTD data) enables earlier detection of outbreak surges. Future directions:

- GPU-based parallelization;
- Integration with CNN-LSTM or Temporal Convolutional Networks;
- Adaptive chaotic search updates dynamically adjusting the logistic map parameter r;
- Encoding operational constraints directly into the augmented Lagrangian learning process.

In terms of application, the proposed CALM-ANN model can assist with decision-making in real-world systems, particularly for systems relating to transport and epidemiology, where accurate and reliable predictions play a vital role.

Further research may be conducted to investigate combining the algorithm of CALM with deep learning techniques to improve modeling of nonlinear time series processes. Other chaotic applications related to improving optimization process stability can possibly be explored. The application of such a method in industrial setups, especially in the fields of energy and cybersecurity, would be very promising.

REFERENCES

[1] Li, G., Zhang, T., Tsai, C.Y., Yao, L., Lu, Y., Tang, J. (2024). Review of the metaheuristic algorithms in applications: Visual analysis based on bibliometrics. Expert Systems with Applications, 255: 124857. https://doi.org/10.1016/j.eswa.2024.124857

[2] Nourani, V. (2024). Review of stochastic modeling: A thorough guide to evaluate, pre-process, model and compare time series with MATLAB software by Hossein Bonakdari and Mohammad Zeynoddin. Journal of Hydrologic Engineering, 29(2). https://doi.org/10.1061/JHYEFF.HEENG-6195

[3] Jovanovic, L., Zivkovic, M., Bacanin, N., Bozovic, A., Bisevac, P., Antonijevic, M. (2024). Metaheuristic optimized electrocardiography time-series anomaly classification with recurrent and long-short term neural networks. International Journal of Hybrid Intelligent Systems, 20(4): 275-300. https://doi.org/10.3233/HIS-240

[4] Bardhan, A., Asteris, P.G. (2023). Application of hybrid ANN paradigms built with nature inspired meta-heuristics for modelling soil compaction parameters. Transportation Geotechnics, 41: 100995. https://doi.org/10.1016/j.trgeo.2023.100995

[5] Jnr, E.O.N., Ziggah, Y.Y., Rodrigues, M.J., Relvas, S. (2022). A hybrid chaotic-based discrete wavelet transform and Aquila optimisation tuned-artificial neural network approach for wind speed prediction. Results in Engineering, 14: 100399. https://doi.org/10.1016/j.rineng.2022.100399

[6] Noureddine, S., Zineeddine, B., Toumi, A., Betka, A., Benharkat, A.N. (2022). A new predictive medical approach based on data mining and Symbiotic Organisms Search algorithm. International Journal of Computers and Applications, 44(5): 465-479. https://doi.org/10.1080/1206212X.2020.1809825

[7] Kaidi, R., Omara, H., Lazaar, M., Al Achhab, M. (2025). Enhancing tabular data analysis for classification of airline passenger satisfaction using TabNet deep neural network. International Journal of Electrical and Computer Engineering, 15(3): 3362-3372. https://doi.org/10.11591/IJECE.V15I3.PP3362-3372

[8] Khew, C., Akbar, R., Mohd-Assaad, N. (2025). Progress and challenges for the application of machine learning for neglected tropical diseases. F1000Research, 12: 287. https://doi.org/10.12688/f1000research.129064

[9] Yousaf, M.Z., Raza, A., Abbas, G., Ullah, N., Al-Ahmadi, A.A., Yasin, A.R., Jamil, M. (2022). MTDC grids: A metaheuristic solution for nonlinear control. Energies, 15(12): 4263. https://doi.org/10.3390/en15124263

[10] Tougma, A., Kaboré, A., Somé, K. (2024). Hyperbolic augmented lagrangian algorithm for multiobjective optimization problems. Gulf Journal of Mathematics, 16(2): 151-170. https://doi.org/10.56947/gjom.v16i2.1876

[11] AlikhaniKoupaei, J., Firouznia, M. (2021). A chaos-based constrained optimization algorithm. Journal of Ambient Intelligence and Humanized Computing, 12(11): 9953-9976. https://doi.org/10.1007/s12652-020-02746-w

[12] Mumtahina, U., Alahakoon, S., Wolfs, P. (2024). Hyperparameter tuning of load-forecasting models using metaheuristic optimization algorithms—A systematic review. Mathematics, 12(21): 3353. https://doi.org/10.3390/math12213353

[13] Cerone, V., Fosson, S.M., Pirrera, S., Regruto, D. (2025). A new framework for constrained optimization via feedback control of lagrange multipliers. IEEE Transactions on Automatic Control, 70(11): 7141-7156. https://doi.org/10.1109/TAC.2025.3568651

[14] Kowalski, P.A., Kucharczyk, S., Mańdziuk, J. (2025). Constrained hybrid metaheuristic algorithm for probabilistic neural networks learning. Information Sciences, 713: 122185. https://doi.org/10.1016/j.ins.2025.122185

[15] Ansar, T., Ashraf, W.M. (2025). Comparison of Kolmogorov–Arnold networks and multi-layer perceptron for modelling and optimisation analysis of energy systems. Energy and AI, 20: 100473. https://doi.org/10.1016/j.egyai.2025.100473

[16] Chicco, D., Warrens, M.J., Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. Peerj Computer Science, 7: e623. https://doi.org/10.7717/peerj-cs.623

[17] Zhao, L., Wang, C., Lin, H., Huang, R., Ma, H., Guo, C. (2026). CRMPA-TimesNet: Hyperparameter optimization with a modified metaheuristic for time series forecasting. The Journal of Supercomputing, 82(8): 458. https://doi.org/10.1007/s11227-026-08586-3

[18] Jun, T., Yasin, A.B., Mansor, N.S.B. (2024). Enhancing efficiency and accuracy of optimization techniques in time series data prediction using machine learning: A systematic literature review. Journal of Computer Science and Technology Studies, 6(5): 10-32996. https://doi.org/10.32996/jcsts.2024.6.5.11

[19] Ramadevi, B., Bingi, K. (2022). Chaotic time series forecasting approaches using machine learning techniques: A review. Symmetry, 14(5): 955. https://doi.org/10.3390/sym14050955

[20] Chou, J.S., Truong, D.N. (2021). Multistep energy consumption forecasting by metaheuristic optimization of time-series analysis and machine learning. International Journal of Energy Research, 45(3): 4581-4612. https://doi.org/10.1002/er.6125

[21] Naser, M.Z., Al-Bashiti, M.K., Tapeh, A.T.G., Naser, A., et al. (2025). A review of benchmark and test functions for global optimization algorithms and metaheuristics. Wiley Interdisciplinary Reviews: Computational Statistics, 17(2): e70028. https://doi.org/10.1002/wics.70028

[22] Abbassi, R., Trojovský, P., Mansor, Z., Trojovská, E., et al. (2025). Cuckoo optimization algorithm via Grey Wolf Optimizer for usage in engineering optimization and optimal power flow with renewable energy sources. Scientific Reports, 15(1): 37629. https://doi.org/10.1038/s41598-025-21515-3

[23] Anuar, M.A., Ibrahim, R., Zainal, N., Rejab, M.M., Hachimi, H. (2025). A comparative study of metaheuristic optimization algorithms on distinct benchmark functions. Journal of Soft Computing and Data Mining, 6(1): 69-85. https://doi.org/10.31972/iceit2024.013

[24] Guo, L., Fang, W., Zhao, Q., Wang, X. (2021). The hybrid PROPHET-SVR approach for forecasting product time series demand with seasonality. Computers & Industrial Engineering, 161: 107598. https://doi.org/10.1016/j.cie.2021.107598

[25] Zhang, G.P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing, 50: 159-175. https://doi.org/10.1016/S0925-2312(01)00702-0