Zero : Jurnal Sains, Matematika, dan Terapan
E-ISSN : 2580-5754; P-ISSN : 2580-569X
Volume 10, Number 1, 2026
DOI: 10.30829/zero.v10i1.28741
Page: 250-264

Adaptive Continuous Parameter Optimization of ARIMA using a Hybrid GA–PSO Approach for Time Series Forecasting

[1]Fitri Andini Ritonga
Department of Mathematics, Universitas Sumatera Utara, Medan, Indonesia

[2]Sutarman
Department of Mathematics, Universitas Sumatera Utara, Medan, Indonesia

[3]Syahriol Sitorus
Department of Mathematics, Universitas Sumatera Utara, Medan, Indonesia

Article Info

Article history:

Accepted 1 April 2026

Keywords:

ARIMA;
Continuous Optimization;
Genetic Algorithm;
Particle Swarm Optimization;
Time Series Forecasting.

ABSTRACT

Accurate forecasting of financial time series remains challenging due to non-stationarity, complex data patterns, and difficulties in parameter optimization within traditional models. Although ARIMA is widely used, its performance is often limited by static parameter estimation and sensitivity to evolving data structures. Existing metaheuristic-based approaches have attempted to address these issues; however, many lack adaptive mechanisms that account for varying data complexity. This study proposes a Continuous Hybrid ARIMA–Metaheuristic (GA–PSO) framework with adaptive parameter tuning guided by Model Complexity Assessment (MCA). The framework enables continuous optimization of ARIMA parameters, allowing the model to dynamically adapt to changing time-series characteristics. Empirical results demonstrate consistent improvements in forecasting performance compared to the baseline ARIMA model. For instance, in the Gold dataset (300 observations), the model achieved RMSE = 56.96, MAE = 41.86, and MAPE = 1.12%, indicating stable and accurate predictions. Statistical validation using the Diebold–Mariano test further confirms the significance of these improvements. The main contribution lies in the integration of adaptive GA–PSO optimization with complexity-aware tuning, which enhances both forecasting stability and responsiveness. However, the findings also indicate the presence of heteroscedasticity in several cases, suggesting that volatility dynamics are not fully captured by the current framework. This limitation highlights the need for incorporating volatility-aware models, such as ARIMA–GARCH, to better represent time-varying variance and improve forecasting robustness in future research.

Corresponding Author:

Fitri Andini Ritonga
Department of Mathematics,
Universitas Sumatera Utara
Email: fitriandini@students.usu.ac.id

1. INTRODUCTION

Time series forecasting plays a crucial role in decision-making across various fields, particularly in economic and financial systems characterized by rapid structural changes and high uncertainty. Financial time series typically exhibit complex patterns such as fluctuations, seasonality, long-term trends, non-stationarity, and periods of extreme volatility [1], [2]. These characteristics make accurate modeling challenging, as the underlying datagenerating process may evolve over time, causing forecasting models calibrated on historical observations to lose predictive stability.

Volatility is one of the most dominant characteristics of financial data. It reflects the degree of uncertainty and risk in asset price movements and often changes dynamically according to market conditions. As described in [3], olatility represents the evolution of stochastic return variability under changing market environments [4], [5], and plays an essential role in financial risk assessment and decision-making [6]. From a statistical perspective, volatility is closely related to heteroscedasticity, where the variance of residuals varies over time, potentially leading to inefficient parameter estimation and reduced forecasting accuracy [7].

Among classical statistical approaches, the Autoregressive Integrated Moving Average (ARIMA) model remains a widely used framework due to its capability to model non-stationary time series through differencing, as well as its combination of autoregressive and moving average structures [8], [9]. Despite its popularity, ARIMA mainly captures linear temporal dependencies and is not designed to represent conditional variance dynamics that frequently appear in financial data [10]. Consequently, forecasting performance may deteriorate when the data exhibit strong volatility or structural instability.

To address volatility dynamics, models such as the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) have been widely used to capture time-varying variance in financial time series. These models are effective in representing volatility clustering and conditional heteroscedasticity [11]. However, the primary focus of this study lies in improving the adaptability of forecasting models through continuous parameter optimization. This perspective allows the proposed approach to complement existing volatility modeling frameworks by enhancing model responsiveness to evolving data patterns.

Although metaheuristic optimization techniques such as Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) enhance the search for optimal parameters, they primarily focus on improving model calibration rather than addressing structural properties of the data, such as conditional heteroscedasticity. Consequently, optimization alone may not be sufficient when the underlying data exhibit significant volatility and dynamic variance patterns.

To overcome these limitations, many studies have explored hybrid forecasting frameworks that integrate statistical models with machine learning techniques or metaheuristic optimization algorithms. Previous research has demonstrated that such combinations can improve forecasting accuracy by enhancing parameter selection and capturing more complex data structures. For example,[10] Proposed a hybrid ARIMA–Singular Spectrum Analysis (SSA) model that outperformed the standard ARIMA model. Similarly, [12] combined ARIMA with machine learning approaches and reported improved predictive performance. In the context of metaheuristic optimization, [13]Applied Particle Swarm Optimization (PSO) to tune ARIMA parameters and obtained better forecasting accuracy, while [14] Integrated Long Short-Term Memory (LSTM) with the Seahorse Optimization Algorithm for stock market prediction.

Within this line of research, hybrid ARIMA–metaheuristic models aim to preserve the statistical interpretability of ARIMA while utilizing optimization algorithms to search for optimal model parameters. Genetic Algorithm (GA) is known for its ability to perform global exploration of the solution space, whereas Particle Swarm Optimization (PSO) provides efficient local exploitation and faster convergence toward promising solutions ([15]; [13]; [16]). These complementary characteristics make the combination of GA and PSO attractive for improving parameter optimization in time series forecasting models.

Despite these developments, two important limitations remain in the existing literature. First, most hybrid ARIMA–metaheuristic models rely on static parameter configurations after optimization, making them less adaptable to evolving data patterns. Second, these approaches generally do not incorporate mechanisms to explicitly account for volatility dynamics, even though financial time series are often characterized by heteroscedasticity and structural instability. This creates a research gap in developing forecasting frameworks that simultaneously address parameter adaptivity and dynamic data characteristics within a unified modeling structure [17], [18].

More critically, although financial time series are widely recognized to exhibit volatility clustering and heteroscedastic behavior, many existing ARIMA-based and hybrid metaheuristic models focus primarily on improving mean forecasting accuracy without explicitly incorporating variance dynamics. While models such as GARCH are specifically designed to capture time-varying volatility, their integration with adaptive metaheuristic-based ARIMA frameworks remains limited in the literature. As a result, adaptive optimization alone may be insufficient to fully represent the underlying structure of financial data, particularly in highly volatile environments where both mean and variance dynamics play a crucial role.

To address this issue, adaptive modeling strategies have begun to receive increasing attention. One approach is to incorporate rolling window mechanisms that allow models to be re-estimated using recent observations. Nevertheless, continuous parameter adaptation within such frameworks remains relatively limited, particularly when combined with structured mechanisms that regulate when and how optimization should be activated.

This study addresses that limitation by proposing a dynamic hybrid ARIMA–metaheuristic framework in which parameter optimization is conducted iteratively within a rolling window scheme. To regulate the optimization process, a Model Complexity Assessment (MCA) mechanism is introduced to evaluate the statistical complexity of the data before adaptive tuning is applied. MCA determines whether the baseline ARIMA structure remains adequate or whether hybrid optimization using GA and PSO should be activated. Model evaluation is conducted not only through forecasting accuracy metrics but also through residual diagnostic tests, including normality, autocorrelation, heteroscedasticity, and outlier detection[19].

The objective of this research is to develop a forecasting framework that remains robust under conditions of high volatility and structural instability. The main contribution of this study is the development of a continuous adaptive hybrid ARIMA–metaheuristic framework that integrates rolling window updating, model complexity assessment, and GA–PSO-based parameter optimization. This framework enhances model adaptability to evolving data patterns while maintaining statistical interpretability. However, it is important to note that the current approach focuses primarily on mean dynamics, and the integration of volatility-aware models remains an important direction for future research.

2. RESEARCH METHOD

This study designs and tests time series forecasting models using adaptive optimization strategies. It focuses on enhancing ARIMA performance by integrating GA and PSO with adaptive parameter tuning in a continuous optimization framework. The ARIMA model provides the statistical foundation, while metaheuristics optimize parameters numerically, dynamically adjusting algorithm settings according to incoming data. Consequently, optimization operates continuously, adapting to data fluctuations rather than as a single static procedure.

The datasets used in this study are obtained from the Kaggle platform, which provides publicly accessible financial time series data. Specifically, three datasets are selected, representing different market characteristics: Gold (commodity price), AAPL (Apple Inc. stock), and BNGA (financial asset). For each dataset, the "Close" price is used as the primary variable, as it reflects the final trading value at each time step and is commonly employed in financial forecasting studies. These datasets are chosen to capture varying levels of volatility and structural complexity, enabling a comprehensive evaluation of the proposed adaptive hybrid ARIMA–metaheuristic framework under different market conditions. However, it is important to note that financial datasets are inherently affected by external factors such as macroeconomic events, market sentiment, and policy changes, which are not explicitly modeled in this study. Therefore, the forecasting performance may vary when applied to different time periods or asset classes.

To further evaluate the robustness of the proposed framework, each dataset is partitioned into multiple rolling window configurations with varying lengths, specifically 100, 200, and 300 observations. This variation allows the model to be tested under different data availability conditions, where smaller windows emphasize adaptability to recent patterns, while larger windows provide more stable statistical estimation. By comparing model performance across different window sizes, this study assesses the ability of the adaptive hybrid framework to maintain forecasting accuracy under varying levels of data complexity and structural stability. This design also enables analysis of how window size influences model sensitivity to volatility and structural changes in financial time series.

2.1 Data Preprocessing

The dataset is preprocessed to satisfy statistical assumptions required by time series modeling. Missing values are removed, and linear interpolation is applied when series length adjustments are needed. Stationarity is ensured via the Augmented Dickey–Fuller (ADF) test, with differencing order d selected adaptively. Data are then split chronologically into 80% training and 20% testing subsets, preserving temporal dependencies for sequential rolling window forecasting and model optimization [20], [21]

2.2 Rolling Window Initialization

The rolling window initialization stage establishes the structural foundation of the continuous hybrid ARIMA–metaheuristic optimization framework. In this stage, the objective is not merely to partition the dataset but to construct a dynamic estimation mechanism that enables sequential model updating as new observations become available [22], [23].

Let $y_1, y_2, \dots, y_T$ denote a time series consisting of Tobservations. For a predefined window length n, the first rolling window is defined as

$$W_1 = \{y_1, y_2, \dots, y_n\} \tag{1}$$

The window then shifts sequentially along the time axis, producing the following window.

$$W_2 = \{y_2, y_3, \dots, y_{n+1}\} \tag{2}$$

This process continues iteratively until the final window

$$W_{T-n} = \{y_{T-n}, \dots, y_T\} \tag{3}$$

At each window W_T , the forecasting model generates a prediction for the next observation using a one-step-ahead forecasting scheme

$$\hat{y}_{t+1} = f(W_t) \quad (4)$$

where f denotes the forecasting function determined by the ARIMA-metaheuristic hybrid model. The one-step-ahead forecasting strategy is adopted because it allows the model to respond more adaptively to rapid structural changes and volatility, which are common characteristics in financial time series.

For the first window, a baseline ARIMA structure is established, and the feasible search space for parameters is defined. In PSO, particle positions and velocities are randomly initialized within this space, while in GA, an initial population of chromosomes is generated and evaluated based on forecasting error. In the hybrid framework, the optimal parameters from the previous window are retained to initialize the next, improving convergence stability compared to conventional rolling window forecasting. Forecasting performance is consistently assessed using RMSE, MAE, and MAPE throughout the optimization process.

2.3 Model Complexity Assessment (MCA)

The Model Complexity Assessment (MCA) determines whether a conventional ARIMA model suffices or if adaptive metaheuristic optimization is needed, serving as a decision-control mechanism that regulates hybrid optimization within the forecasting framework.
Formally, MCA can be defined as a mapping function.

$$\text{MCA} : y \rightarrow \{0,1\} \quad (5)$$

The MCA output determines the forecasting strategy: 0 indicates the baseline ARIMA is sufficient, while activates hybrid ARIMA–GA–PSO optimization. Structural complexity is assessed using a vector of statistical diagnostic indicators.

$$\text{k} = (k_1, k_2, k_3, k_4) \quad (6)$$

where k_1 denotes non-stationarity detected using the Augmented Dickey–Fuller (ADF) test, k_2 represents heavy-tail behaviour measured through kurtosis, k_3 indicates heteroscedasticity identified using the ARCH test, and k_4 represents residual autocorrelation evaluated using the Ljung–Box test.
Each diagnostic indicator is converted into a discrete signal.

$$k_i = \begin{cases} 1, & \text{if the complexity condition is satisfied} \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

These signals are then aggregated to construct a global complexity index

$$c = \sum_{i=1}^{4} w_i k_i \quad (8)$$

where c represents the overall complexity score and $w_i$denotes the weight assigned to each indicator. A higher value of $c$indicates a higher level of structural complexity in the time series.
Based on this index, the MCA determines the appropriate forecasting strategy using the following decision rule:

$$\text{Decision} = \begin{cases} \text{Baseline ARIMA}, & c<\delta \\ \text{Hybrid ARIMA-Metaheuristic}, & c\geq\delta \end{cases} \quad (9)$$

The selection of the threshold parameter δ is based on statistical significance criteria derived from diagnostic tests. Specifically, indicators such as the Augmented Dickey–Fuller (ADF) test, Ljung–Box test, and ARCH test are evaluated using standard significance levels (e.g., 5%).

The weights w_i are assigned to reflect the relative importance of each diagnostic indicator in capturing structural complexity. In this study, equal weighting is adopted to avoid bias toward a specific statistical property, ensuring a balanced assessment of complexity.

Additionally, preliminary experiments were conducted to validate that the chosen threshold provides a stable trade-off between unnecessary optimization and responsiveness to structural changes.
where δ represents the predefined complexity threshold.

In this framework, MCA operates as a deterministic control mechanism that regulates the activation of metaheuristic optimization. When the complexity level exceeds the threshold, the hybrid optimization process is activated, and the parameter search space is expanded:

$$\Theta = \begin{cases} \Theta_{\text{basic}}, & c<\delta \\ \Theta_{\text{ectended}}, & c\geq\delta \end{cases} \quad (10)$$

The framework dynamically applies hybrid optimization only when structural complexity warrants it, balancing forecasting accuracy and computational efficiency [24], [25], [26].

2.4 Hybrid ARIMA-Metaheuristic Model Construction (GA-PSO)

The hybrid ARIMA–Metaheuristic model combines the global exploration capability of the Genetic Algorithm (GA) with the local exploitation ability of Particle Swarm Optimization (PSO). The objective of this hybrid framework is to automatically determine the optimal ARIMA parameters. $\theta = (p, d, q)$ that minimizes the forecasting error [27].

2.5 Genetic Algorithms

Genetic Algorithms (GA) are evolution-based metaheuristic methods inspired by natural selection. In the Genetic Algorithm, ARIMA parameters are encoded as binary chromosomes representing the parameter vector [15].

$$\theta = (p, d, q) \tag{11}$$

Each chromosome corresponds to a candidate ARIMA model. The encoding process transforms discrete ARIMA parameters into a binary representation to enable combinatorial exploration in the search space.
For example, the autoregressive parameter p is encoded as

$$p = \sum_{j=0}^{L_p - 1} b_j 2^j, \qquad b_j \in \{0,1\} \tag{12}$$

where b_j denotes the binary bit at position j, and L_p represents the chromosome length used to encode the parameter p. This representation allows the GA to explore the parameter space in a discrete combinatorial manner and reduces numerical bias that may arise from continuous initialization.

The GA population at generation $t$is defined as

$$P^{(t)} = \{\theta_1^{(t)}, \theta_2^{(t)}, \ldots, \theta_N^{(t)}\} \tag{13}$$

where N denotes the population size and each individual $\theta_i^{(t)} = (p_i, d_i, q_i)$ represents a candidate ARIMA parameter vector obtained after decoding the binary chromosome.
The fitness of each individual is evaluated using the forecasting loss function. $L(\theta)$. The best individual at generation $t$is defined as

$$\theta_{best}^{(t)} = \arg\min_{\theta \in P^{(t)}} L(\theta) \tag{14}$$

To preserve high-quality solutions, an elitism mechanism is employed. The elite set is defined as

$$\varepsilon^{(t)} = \{\theta_{(1)}^{(t)}, \theta_{(2)}^{(t)}, \ldots, \theta_{(k)}^{(t)}\} \tag{15}$$

where $k$denotes the number of elite individuals retained in each generation. The elitism mechanism guarantees that the best solutions are deterministically preserved in the next generation.

$$P^{(t+1)} \supseteq \varepsilon^{(t)} \tag{16}$$

ensuring that high-quality solutions are not lost during the evolutionary process.
After the GA optimization stage, the best chromosome

$$b^* = (b_1, b_2, \ldots, b_L) \tag{17}$$

is decoded into the ARIMA parameter vector

$$\theta_{GA} = (p_{GA}, d_{GA}, q_{GA}) \tag{18}$$

This solution acts as the initial reference for the PSO refinement stage.

2.6 Particle Swarm Optimization (PSO)

$$x_i^{(0)} = \theta_{GA} + \varepsilon_i, \qquad \varepsilon_i \sim U(-\delta, \delta) \tag{19}$$

where $x_i^{(0)} \in \mathbb{R}^3$ represents the initial particle position and δ controls the local exploration radius. Since ARIMA parameters are discrete integers while PSO operates in continuous space, a projection operator is applied:

$$\tilde{x}_i^{(t)} = \prod (x_i^{(t)}) \tag{20}$$

where

$$\prod (x) = (\text{round}(p), \text{round}(d), \text{round}(q)) \tag{21}$$

This operator maps the continuous particle position to a feasible ARIMA parameter vector satisfying.

$$\tilde{x}_i = (\tilde{p}_i, \tilde{d}_i, \tilde{q}_i) \tag{22}$$

Is used to construct an ARIMA model $ARIMA(\tilde{p}_i, \tilde{d}_i, \tilde{q}_i)$. The model is estimated, and the resulting prediction error is used to compute the fitness value.
After the PSO iteration convergence, the final optimized parameter vector is defined as

$$\theta^* = \arg\min_{\tilde{x}_i} f(\tilde{x}_i) \tag{23}$$

where

$$\theta^* = (p^*, d^*, q^*) \tag{24}$$

Represent the optimal ARIMA parameter configuration.
In this hybrid framework, the GA determines a promising region of the parameter search space through global exploration, while PSO refines the solution through local exploitation around the best GA candidate.

2.7 Algorithm Parameter Settings

To ensure reproducibility, the metaheuristic parameters are explicitly defined based on literature and preliminary experiments. GA is configured with a moderate population and generation size, with crossover and mutation probabilities set to balance exploration and diversity, and elitism preserving top solutions. PSO uses a swarm with inertia weight $\omega = 0.7$, cognitive coefficient $c_1 = 1.5$, and social coefficient $c_2 = 1.5$, regulating exploration and exploitation. The feasible ARIMA parameter space is bounded as $p \in [0, 5]$, $d \in [0, 2]$, and $q \in [0, 5]$ to avoid overly complex models and reduce overfitting. All settings remain constant across rolling windows to ensure consistent optimization and fair comparison.

Table 1. Metaheuristic Parameter Settings

Parameter	Value	Description
GA population size	12	Number of individuals in the GA population
GA generations	8	Maximum number of GA iterations
GA Mutation Rate	0.25	Probability of mutation in GA
PSO particles	6	Number of particles in the PSO swarm
PSO iterations	4	Maximum PSO iteration
Inertia weight (w)	0.5	Control exploration-exploitation tradeoff
Cognitive coefficient (c1)	1.5	Individual learning factor
Social coefficient (c2)	1.5	Global learning factor
Rolling window size	(100,200,300)	Evaluated window sizes for robustness analysis

Parameter values were chosen based on empirical evaluation and prior studies in metaheuristic time-series forecasting. The framework employs adaptive tuning, adjusting optimization parameters dynamically according to the complexity of each data window. Forecasting was assessed across rolling windows W = 100, 200, and 300, showing that smaller windows enhance responsiveness to local fluctuations, while larger windows yield more stable estimates. The hybrid ARIMA–GA–PSO model maintains consistent performance across all configurations, demonstrating the robustness of the proposed framework.

2.8 Algorithmic Procedure

The algorithm integrates statistical modeling, complexity assessment, and hybrid metaheuristic optimization within a continuous rolling-window forecasting scheme. The overall workflow of the proposed forecasting framework is summarized in Algorithm 1.

Algorithm 1: Continuous Hybrid ARIMA–GA–PSO Forecasting Framework

Input:
Time series observations $y_t = \{y_1, y_2, ..., y_T\}$ rolling window size $W \in \{100, 200, 300\}$
Output:
One-step-ahead forecast values $\hat{y}_{t+1}$
Perform data preprocessing, including data cleaning, interpolation for missing observations, and stationarity testing.
Determine the differencing order d using the Augmented Dickey–Fuller (ADF) test to ensure stationarity of the series.
Initialize the rolling window of size W.
For each rolling window t:
    Apply the Model Complexity Assessment (MCA) to evaluate the statistical complexity of the time series.
    If MCA = 0, estimate the baseline ARIMA model using the current window.
    If MCA = 1, MCA=1 Activate the hybrid metaheuristic optimization process:
        Initialize a Genetic Algorithm (GA) population representing candidate ARIMA parameter chromosomes.
        For each generation $g = 1, ..., G$:
            Evaluate the forecasting fitness using the RMSE objective function.
            Apply genetic operators: selection, crossover, and mutation.
            Preserve elite individuals to maintain high-quality solutions.
        Obtain the best GA solution θ_{GA}.
        Initialize Particle Swarm Optimization (PSO) particles in the neighborhood of θ_{GA}.

For each PSO iteration $i = 1, \ldots, I$:
Update particle velocities and positions.
Project particle positions onto the feasible ARIMA parameter space.
Evaluate the fitness of each particle using the forecasting error.
Obtain the final optimized parameter vector θ^*
Estimate the ARIMA model using the optimized parameters θ^* .
Generate a one-step-ahead forecast $\hat{y}_{t+1}$
Shift the rolling window forward and repeat the process until the end of the dataset.

2.9 Computational Complexity and Scalability

The computational complexity of the proposed hybrid framework is primarily dominated by repeated estimation of the ARIMA model during the optimization process.
For the Genetic Algorithm (GA) stage, the computational cost can be approximated as

$$O(G_{GA}.P_{GA}.C_{ARIMA}) \tag{25}$$

Where G_{GA} denotes the number of GA generations, P_{GA} represents the population's size, and C_{ARIMA} corresponds to the computational cost required to estimate a single ARIMA model.
Similarly, the Particle Swarm Optimization (PSO) stage exhibits a complexity of

$$O(G_{PSO}.P_{PSO}.C_{ARIMA}) \tag{26}$$

Where G_{PSO} is the number of PSO iterations and P_{PSO} denotes the number of particles in the swarm.

ARIMA estimation dominates the computational cost, while the GA-guided PSO search improves efficiency by focusing on promising regions and accelerating convergence.

The computational cost of the framework grows roughly linearly with the number of rolling windows. The MCA mechanism enhances efficiency by triggering hybrid optimization only for statistically complex segments, avoiding unnecessary computations and maintaining scalability for longer time series.

2.10 Mechanisms to Prevent Overfitting

To enhance robustness and generalization, the proposed hybrid ARIMA–GA–PSO framework incorporates multiple mechanisms to mitigate overfitting. A rolling window strategy trains the model on a fixed-length segment. To enhance robustness and generalization, the proposed hybrid ARIMA–GA–PSO framework incorporates multiple mechanisms to mitigate overfitting. A rolling window strategy trains the model on a fixed-length segment ($W \in \{100,200,300\}$, capturing recent patterns while reducing reliance on outdated observations. Candidate ARIMA parameters are evaluated using a validation subset within each window, ensuring fitness reflects generalization rather than training error alone. Search parameters of GA and PSO are adaptively adjusted to balance exploration and exploitation, avoiding unnecessarily complex or unstable models [28]. Finally, residual diagnostics, including Ljung–Box, Breusch–Pagan, and Jarque–Bera tests, verify that residuals behave as white noise. Collectively, these mechanisms maintain predictive accuracy while preserving statistical reliability and generalizability. capturing recent patterns while reducing reliance on outdated observations. Candidate ARIMA parameters are evaluated using a validation subset within each window, ensuring fitness reflects generalization rather than training error alone. Search parameters of GA and PSO are adaptively adjusted to balance exploration and exploitation, avoiding unnecessarily complex or unstable models. Finally, residual diagnostics, including Ljung–Box, Breusch–Pagan, and Jarque–Bera tests, verify that residuals behave as white noise [29]. Collectively, these mechanisms maintain predictive accuracy while preserving statistical reliability and generalizability.

2.11 Role of Parameter Tuning in the Hybrid Framework

Parameter tuning in this study is formulated as an optimization problem for determining the ARIMA parameter vector [30].

$$\theta^* = \arg\min_{\theta \in \Theta} L(\theta) \tag{27}$$

Where Θ denotes the feasible parameter space and $L(\theta)$ represents the forecasting loss function.
Each candidate solution is evaluated using the Root Mean Squared Error (RMSE)

$$L(\theta_i) = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t(\theta_i))^2} \tag{28}$$

Where y_t denotes the observed value and $\hat{y}_t(\theta_i))^2$ is the forecast produced by the ARIMA model with parameter vector θ_i .

Thus, the metaheuristic algorithms do not replace the ARIMA forecasting mechanism but function as optimization controllers that search for parameter configurations yielding the lowest forecasting error.

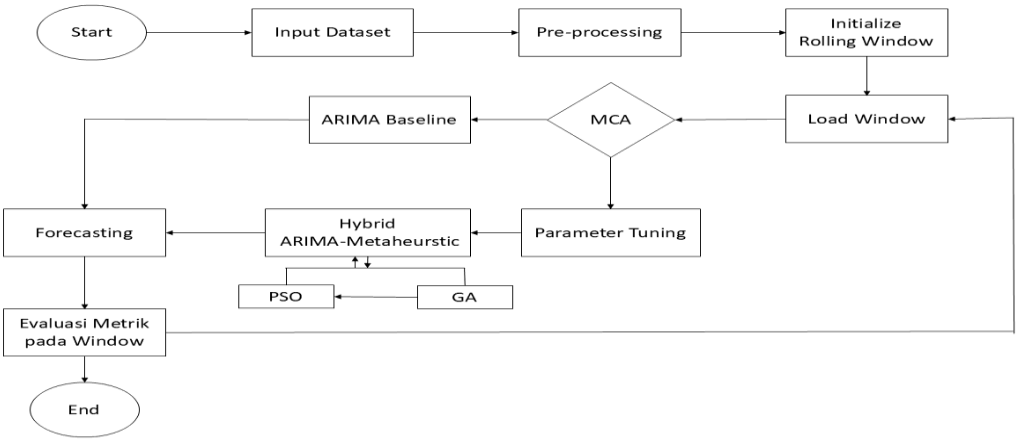

Figure 1. Research flowchart

The flowchart was developed to clarify the relationship between processes in the study, particularly the decision-making mechanism through the MCA and the integration of the ARIMA–Metaheuristic hybrid approach in the rolling window scheme. This diagram helps illustrate the adaptive and sustainable optimization flow.

2.12 Performance Evaluation Metrics

To measure the accuracy level of the model prediction, three evaluation metrics are used, namely.

Root Mean Squared Error (RMSE), which is used to measure the magnitude of prediction errors, gives a greater penalty to extreme errors [31], [32].

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \quad (29)$$

Where y_t is the actual value at time-t, and $\hat{y}_t$ is the predicted value with n observations.

Mean Absolute Error (MAE) is used to measure the average absolute error without giving additional weight to extreme errors, thus providing an overview of the level of prediction deviation.

$$MAE = \frac{1}{N}\sum_{t=1}^{n}| y_t - \hat{y}_t | \quad (30)$$

Meanwhile, Mean Absolute Percentage Error (MAPE) is used to measure prediction errors in percentage form, making it easier to interpret evaluation results relative to actual values.

$$MAPE = \frac{100}{N}\sum_{t-1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \quad (31)$$

These three metrics collectively provide a comprehensive evaluation of model performance. Algorithm parameters are adaptively updated as data characteristics change, ensuring stable performance under high volatility and non-stationary conditions. Conceptually, adaptive tuning operates at the algorithm level, while hybrid GA–PSO optimization addresses model-level parameter selection, clarifying methodological contributions and avoiding overlap between tuning and optimization.

3. RESULT AND ANALYSIS

This section presents a comprehensive evaluation of the ARIMA–Metaheuristic hybrid model with continuous optimization across multiple financial time series and window lengths. The analysis proceeds in stages, including model performance comparison and residual assessment to validate underlying assumptions.

3.1 Data Description and Experimental Scenarios

Table 1. Forecasting performance for Gold dataset under different window size

Rolling Window Size (n observations)	Model	ARIMA	RMSE	MAE	MAPE (%)	Improvement (%)
Gold (300)	ARIMA Baseline	0,1,1	365.99			
	Continuous Hybrid GA-PSO		56.96	41.86	1.12%	84.4%
Gold (200)	ARIMA Baseline	1,1,2	475.96			

	Continuous Hybrid GA-PSO		62.66	46.08	1.19%	86.8%
Gold (100)	ARIMA Baseline	2,1,2	292.36			
	Continuous Hybrid GA-PSO		81.35	63.83	1.59%	72.2%

Table 2. Forecasting performance for AAPL dataset under different window size

Rolling Window Size (n observations)	Model	ARIMA	RMSE	MAE	MAPE (%)	Improvement (%)
AAPL (300)	ARIMA Baseline	0,1,1	14.52			
	Continuous Hybrid GA-PSO		2.50	2.04	1.11%	82.7%
AAPL (200)	ARIMA Baseline	0,1,1	16.88			
	Continuous Hybrid GA-PSO		2.68	2.20	1.17%	84.1%
AAPL (100)	ARIMA Baseline	1,1,1	4.74			
	Continuous Hybrid GA-PSO		2.72	2.21	1.15%	42.6%

Table 3. Forecasting performance for BNGA dataset under different window size

Rolling Window Size (n observations)	Model	ARIMA	RMSE	MAE	MAPE (%)	Improvement (%)
BNGA (300)	ARIMA Baseline	0,1,2	101.29			
	Continuous Hybrid GA-PSO		17.84	13.83	0.80%	82.4%
BNGA (200)	ARIMA Baseline	2,1,0	27.84			
	Continuous Hybrid GA-PSO		17.62	13.87	0.81%	36.7%
BNGA (100)	ARIMA Baseline	0,1,2	13.91			
	Continuous Hybrid GA-PSO		16.65	12.00	0.69%	-19.7%

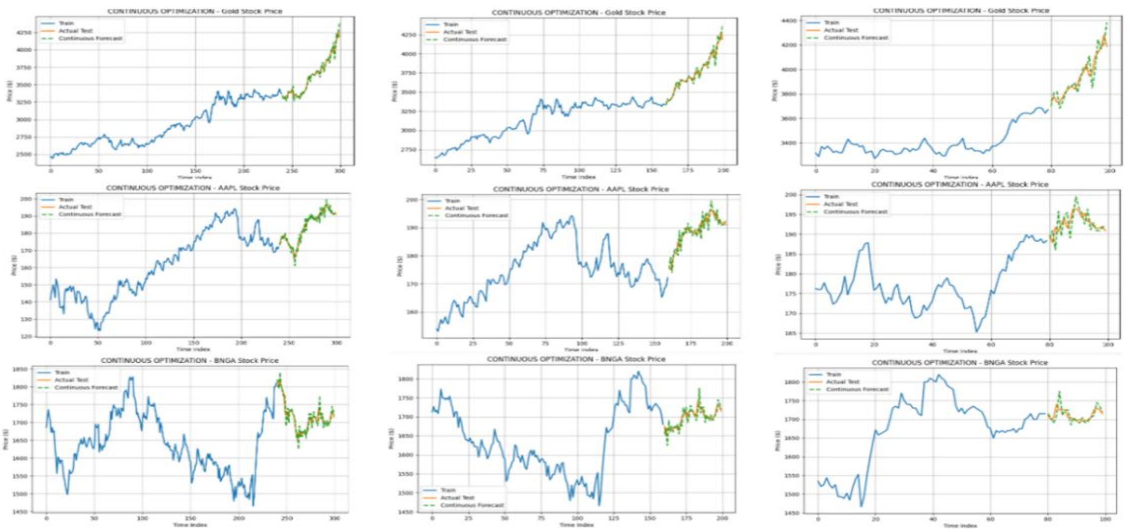

Figure 2. Performance comparison between ARIMA baseline and Continuous Hybrid GA-PSO across rolling windows, illustrating the improvement in forecasting accuracy obtained through adaptive optimization.

Table 1-3 presents the forecasting performance of the continuous hybrid ARIMA-metaheuristic model in comparison with the baseline ARIMA model across several datasets and rolling window configurations. The financial time series analyzed in this study were obtained from publicly available datasets on the Kaggle platform, including Gold price data, Apple Inc. stock prices (AAPL), and BNGA stock prices. All datasets consist of daily closing prices, which represent a commonly used observation frequency in financial time series forecasting studies.

The Gold price dataset covers the period from 30 August 2000 to 17 October 2025 and contains 6,307 daily observations. The Apple Inc. (AAPL) dataset spans from 2 January 2019 to 29 December 2023 with a total of 1,259 observations, while the BNGA dataset ranges from 18 August 2020 to 15 August 2025 and consists of 1,201 observations. These datasets originate from different financial markets and time periods, introducing variations in their statistical characteristics and providing a suitable setting for evaluating the robustness of the proposed forecasting approach.

To maintain a consistent experimental setup across datasets with different lengths, the forecasting experiments were conducted using rolling windows of 100, 200, and 300 observations. This design allows the model to be evaluated under varying amounts of historical information while ensuring comparability across datasets.

The results show that the proposed continuous hybrid GA–PSO framework significantly improves forecasting performance across most datasets and window configurations, with improvement rates exceeding 80% in several cases. For the Gold and AAPL datasets, the hybrid model consistently outperforms the baseline ARIMA model across all window sizes, demonstrating strong adaptability to both stable and moderately volatile conditions.

However, for the BNGA dataset with a smaller window size (n = 100), the hybrid model exhibits a performance degradation, as indicated by a negative improvement value (-19.7%). This suggests that under low data availability and potentially lower complexity conditions, the adaptive optimization process may introduce instability or overfitting, leading to reduced predictive accuracy.

Overall, the hybrid model tends to show improved forecasting performance relative to the baseline ARIMA model. The improvement is more noticeable for datasets with relatively higher volatility, such as the Gold series, and for larger rolling windows in the AAPL and BNGA datasets. These findings indicate that the effectiveness of the hybrid framework is closely related to data complexity and window size. Larger windows provide more stable statistical estimation, while smaller windows increase sensitivity to short-term fluctuations, which may negatively impact optimization performance.

Table 4. Descriptive Statistics of Continuous Hybrid GA-PSO Model Residuals

Rolling Window Size (n observations)	Total Residual	Mean	Std Deviation	Min	Max	Range	MAR	MAD
Gold (300)	60	-1.12	56.95	-193.60	126.59	320.200	30.15	32.44
Gold (200)	40	-2.09	62.63	-193.60	126.59	320.300	32.75	33.75
Gold (100)	20	-5.90	81.13	-193.60	126.59	320.200	47.34	46.44
AAPL (300)	60	-0.03	2.50	-5.78	5.75	11.53	1.70	1.68
AAPL (200)	40	-0.10	2.68	-5.78	5.75	11.53	1.70	1.73
AAPL (100)	20	-0.08	2.72	-5.04	5.75	10.79	1.82	1.91
BNGA (300)	60	-0.33	17.83	-50.00	45.00	95.00	10.00	10.00
BNGA (200)	40	0.37	17.62	-50.00	45.00	95.00	12.50	12.50
BNGA (100)	20	-0.50	16.65	-50.00	30.00	80.00	10.00	10.00

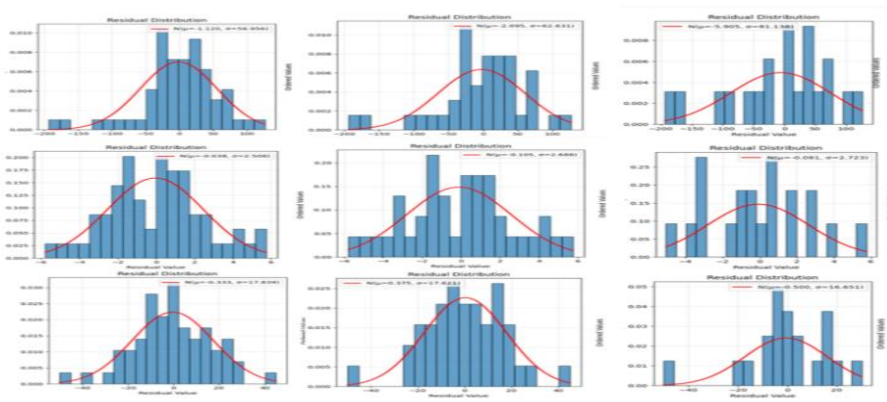

Figure 3. Residual chart

The average residual value reported in Table 4 indicates that the proposed forecasting framework does not exhibit a noticeable systematic bias in its predictions. However, the standard deviation of the residuals tends to increase for smaller rolling windows, suggesting that prediction errors become more variable when the model relies on limited historical observations. This behavior is particularly evident in the Gold dataset with a window size of 100, where the highest residual volatility is observed. Descriptive statistics of the residuals from the continuous hybrid GA–PSO model are summarized in Table 5 to provide further insight into the bias and dispersion of the prediction errors. From a practical standpoint, these results have implications for financial forecasting tasks such as short-term market prediction, portfolio monitoring, and trading signal generation.

Forecasting models that produce relatively unbiased residuals can contribute to more stable decision support. Nevertheless, the higher variance observed in shorter windows indicates that forecasts generated from limited data may carry greater uncertainty, which should be carefully considered in risk management contexts. Another important aspect is the computational trade-off associated with the hybrid optimization framework. The integration of GA and PSO allows a broader exploration of the parameter space and can improve parameter selection compared with conventional estimation methods. However, this benefit comes at the cost of increased computational complexity, since each candidate solution requires repeated estimation of the ARIMA model. In practice, the hybrid approach may therefore be more suitable for offline forecasting analysis or periodic model recalibration rather than high-frequency real-time applications. Finally, several limitations should be acknowledged. Despite improvements in parameter optimization, residual diagnostics suggest that certain datasets may still exhibit heteroscedastic behavior and potential remaining autocorrelation.

Table 5. Residual Distribution

Rolling Window Size (n observations)	Skewness	Curtosis	Empirical Rule Check					
			±1σ	Expected	±2σ	Expected	±3σ	Expected
Gold (300)	-0.8846	1.8899	76.7%	68.3%	93.3%	95.5%	98.3%	99.7%
Gold (200)	-0.8440	1.4605	72.5%	68.3%	92.5%	95.5%	97.5%	99.7%
Gold (100)	-0.6812	-0.0776	70.0%	68.3%	90.0%	95.5%	100.0%	99.7%
AAPL (300)	0.1284	-0.2644	66.7%	68.3%	93.3%	95.5%	100.0%	99.7%
AAPL (200)	0.0319	-0.4817	65.0%	68.3%	95.0%	95.5%	100.0%	99.7%
AAPL (100)	0.1678	-0.5404	60.0%	68.3%	95.0%	95.5%	100.0%	99.7%
BNGA (300)	-0.1084	0.2185	68.3%	68.3%	95.0%	95.5%	100.0%	99.7%
BNGA (200)	-0.1410	0.5811	70.0%	68.3%	95.0%	95.5%	100.0%	99.7%
BNGA (100)	-0.9117	1.8392	80.0%	68.3%	95.0%	95.5%	95.0%	99.7%

Table 5 shows that AAPL and BNGA residuals are relatively symmetric with kurtosis near the normal distribution, whereas Gold exhibits negative skewness and larger deviation, reflecting high volatility. Residual normality tests were conducted to assess model stability and the validity of statistical inference, although strict normality is not required for time series forecasting. Empirical analysis uses three financial datasets: Gold (commodity price), AAPL (Apple Inc. stock), and BNGA (financial asset), representing different volatility levels and market complexities.

Table 6. Residual Autocorrelation Test

Rolling Window Size (n observations)	Ljung-Box test		
	Lag	p-value	Diagnostic
Gold (300)	5	0.0048	Autocorrelation Detected
	10	0.0133	Autocorrelation Detected
	15	0.0160	Autocorrelation Detected
Gold (200)	5	0.0487	Autocorrelation Detected
	10	0.0948	No Autocorrelation
	15	0.0892	No Autocorrelation
Gold (100)	5	0.1331	No Autocorrelation
	10	0.2418	No Autocorrelation
	15	0.2774	No Autocorrelation
AAPL (300)	5	0.0001	AutocorrelationDetected
	10	0.0000	Autocorrelation Detected
	15	0.0000	Autocorrelation Detected
AAPL (200)	5	0.003	Autocorrelation Detected
	10	0.0000	Autocorrelation Detected
	15	0.0000	Autocorrelation Detected
AAPL (100)	5	0.0528	No Autocorrelation
	10	0.0831	No Autocorrelation
	15	0.0971	No Autocorrelation
BNGA (300)	5	0.0016	Autocorrelation Detected
	10	0.0001	Autocorrelation Detected
	15	0.0015	Autocorrelation Detected
BNGA (200)	5	0.0030	Autocorrelation Detected
	10	0.0017	Autocorrelation Detected
	15	0.0199	Autocorrelation Detected
BNGA (100)	5	0.0199	Autocorrelation Detected
	10	0.0735	No Autocorrelation
	15	0.1329	No Autocorrelation

Table 7. Residual Heteroscedasticity Test

Rolling Window Size (n observations)	Breusch-Pagan test			White test		
	LM	p-value	Diagnostic	LM	p-value	Diagnostic
Gold (300)	14.6849	0.0001	Heteroscedastic Detected	25.5368	0.0000	Heteroscedastic Detected
Gold (200)	12.8080	0.0003	Heteroscedastic Detected	18.9145	0.0001	Heteroscedastic Detected
Gold (100)	6.7078	0.0096	Heteroscedastic Detected	9.1590	0.0103	Heteroscedastic Detected
AAPL (300)	0.0018	0.9663	Homoscedastic	2.0624	0.3566	Homoscedastic
AAPL (200)	0.5213	0.4703	Homoscedastic	0.7312	0.6938	Homoscedastic
AAPL (100)	0.7765	0.3782	Homoscedastic	7.4226	0.0244	Heteroscedastic Detected
BNGA (300)	0.1977	0.6565	Homoscedastic	6.3629	0.6565	Heteroscedastic Detected
BNGA (200)	0.1786	0.6726	Homoscedastic	27.8623	0.0000	Heteroscedastic Detected
BNGA (100)	8.4185	0.0037	Heteroscedastic Detected	15.5317	0.0004	Heteroscedastic Detected

Both tests in Tables 6 and 7 indicate that the residuals are homoscedastic. Rejection at a p-value < 0.05 indicates heteroscedasticity, which is commonly found in financial data with high volatility. The results of the Breusch-Pagan test and White test show that the gold dataset residuals are consistently heteroscedastic across all windows. AAPL is relatively homoscedastic, while BNGA shows more dominant heteroscedasticity in small windows. These findings confirm that residual volatility is an inherent characteristic of financial data and opens up opportunities for the development of other models.

Table 8. Residual Performance of the ARIMA Hybrid Model – Metaheuristics

Rolling Window Size (n observations)	Residual			
	RMSE	MAE	R-Square	MAPE
Gold (300)	56.96	41.86	0.9558	1.1321%
Gold (200)	52.66	46.08	0.9277	1.1987%
Gold (100)	81.35	63.83	0.7527	1.6038%
AAPL (300)	2.50	2.04	0.9196	1.1129%
AAPL (200)	2.68	2.20	0.7486	1.1713%
AAPL (100)	2.72	2.21	-5.56	1.1517%
BNGA (300)	17.84	13.83	0.7926	0.8068%
BNGA (200)	17.63	13.87	0.3510	0.8175%
BNGA (100)	16.65	12.00	-0.5443	0.6987%

Table 8 reinforces the previous findings that the ARIMA-Metaheuristic hybrid model performs best on large and medium windows. The decrease in R-Square on small windows confirms the limitations of historical data in the adaptation optimization process.

The Diebold–Mariano (DM) test was conducted to assess whether the Hybrid GA–PSO model significantly improves forecasting accuracy. Table 7 summarizes the results across datasets and rolling window sizes [33].

Table 9. Diebold-Mariano test result for forecast accuracy comparison

Rolling Window Size (n observations)	DM Statistic	p-value	Significance
Gold (300)	-5.2685	< 0.001	significant
Gold (200)	-6.3523	< 0.001	significant
Gold (100)	-3.8563	< 0.001	significant
AAPL(300)	-8.3686	< 0.001	significant
AAPL(200)	-11.5488	< 0.001	significant
AAPL (100)	-2.8917	0.0093	significant
BNGA (300)	-12.3823	< 0.001	significant
BNGA (200)	-3.4025	0.0016	significant
BNGA (100)	0.6424	0.5283	Not significant

The predominance of negative Diebold–Mariano (DM) statistics indicates that the forecast errors produced by the hybrid model are systematically lower than those of the baseline ARIMA model. The statistical significance observed in most cases suggests that the improvement is not due to random variation but reflects a consistent enhancement in predictive performance across multiple configurations.

The observed performance improvement can be attributed to the adaptive nature of the proposed hybrid framework, which dynamically adjusts model parameters in response to evolving data patterns. This enables the model to better capture short-term structural changes that are not effectively handled by static ARIMA configurations. However, the degradation observed in certain cases, particularly for smaller window sizes, suggests that excessive sensitivity to recent fluctuations may lead to unstable parameter estimation. This indicates a trade-off between adaptability and robustness, especially under limited data conditions.

While the proposed framework improves forecasting performance through adaptive parameter optimization, it does not explicitly model volatility dynamics such as conditional heteroscedasticity. In contrast, models such as the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) are specifically designed to capture time-varying variance and volatility clustering in financial time series. Therefore, the proposed approach should be viewed as complementary rather than substitutive to volatility-aware models. Integrating adaptive optimization with volatility modeling frameworks, such as ARIMA–GARCH, may further enhance forecasting performance, particularly in highly volatile environments. This distinction highlights that while GARCH-type models focus on modeling conditional variance, the proposed framework enhances the adaptability of the mean forecasting process, suggesting that combining both approaches could address both structural and volatility-related challenges simultaneously.

From a practical perspective, the proposed adaptive framework is particularly relevant for real-time forecasting applications, such as financial trading systems and risk management platforms, where model responsiveness to changing market conditions is critical. The rolling window mechanism combined with continuous optimization enables the model to operate in dynamic environments, making it suitable for large-scale systems that require frequent model updates without manual recalibration. However, the computational cost associated with continuous optimization may become a limiting factor in high-frequency or large-scale deployment scenarios, highlighting the need for efficient implementation strategies.

The results also suggest that model effectiveness is closely linked to data complexity. In highly volatile datasets, adaptive optimization provides significant benefits by capturing non-stationary behavior. Conversely, in more stable datasets, excessive adaptation may introduce unnecessary variability, leading to performance degradation. This highlights the importance of balancing model flexibility and stability, which remains a key challenge in time series forecasting.

4. CONCLUSION

This study presents a continuous hybrid ARIMA–metaheuristic framework that combines Genetic Algorithms and Particle Swarm Optimization with adaptive parameter tuning guided by Model Complexity Assessment (MCA). The findings demonstrate that the proposed approach improves forecasting performance compared to the standard ARIMA model, particularly when sufficient historical data are available to support stable parameter estimation. The framework also exhibits the ability to adapt to evolving data patterns, which is essential in dynamic financial time series environments. However, this adaptability introduces trade-offs. The continuous optimization process increases computational cost, which may limit scalability in large-scale or high-frequency applications. In addition, smaller rolling windows may lead to unstable parameter estimation due to increased sensitivity to short-term fluctuations. Furthermore, empirical results indicate the presence of heteroscedasticity in several datasets, suggesting that the variance of the residuals is not constant over time. A key limitation of the proposed framework is that it does not explicitly model volatility dynamics, such as conditional heteroscedasticity, which are fundamental characteristics of financial time series. As a result, the model may not fully capture variance-driven behavior in highly volatile environments. Therefore, future research should focus on integrating the proposed adaptive optimization mechanism with volatility-aware models, such as ARIMA–GARCH, to jointly capture both mean and variance dynamics. Such integration is expected to enhance forecasting robustness and provide a more comprehensive modeling framework for complex financial systems. Additionally, further work may explore improvements in computational efficiency to support real-time and large-scale forecasting applications. Overall, the proposed hybrid approach offers a flexible and adaptive forecasting framework, but its effectiveness can be further strengthened by incorporating volatility modeling and optimizing computational performance.

5. REFERENCES

[1] S. Jenčová, P. Vašaničová, M. Košíková, and M. Miškufová, "A Time Series Approach to Forecasting Financial Indicators in the Wholesale and Retail Trade," *World*, vol. 6, no. 1, p. 5, Jan. 2025, doi: 10.3390/world6010005.

[2] P. Naivasha, G. Musumba, P. Gikunda, and J. Wandeto, "Model-based evaluation of synthetic financial time series data: A comparative study with multi-metric validation," *Array*, vol. 29, p. 100684, 2026, doi: https://doi.org/10.1016/j.array.2026.100684.

[3] L. Rubio, A. Palacio Pinedo, A. Mejía Castaño, and F. Ramos, "Forecasting volatility by using wavelet transform, ARIMA and GARCH models," *Eurasian Economic Review*, vol. 13, no. 3, pp. 803–830, 2023, doi: 10.1007/s40822-023-00243-x.

[4] David P. Lundquist and Daniel J. Eck, "Volatility Forecasting Using Similarity-based Parameter Correction and Aggregated Shock Information," 2024.

[5] S. T. Enow, "Exploring stochastic volatility in financial market," *International Journal of Research in Business and Social Science (2147- 4478)*, vol. 14, no. 1, pp. 74–79, Feb. 2025, doi: 10.20525/ijrbs.v14i1.3837.

[6] E. S. Gunnarsson, H. R. Isern, A. Kaloudis, M. Risstad, B. Vigdel, and S. Westgaard, "Prediction of realized volatility and implied volatility indices using AI and machine learning: A review," *International Review of Financial Analysis*, vol. 93, p. 103221, 2024, doi: https://doi.org/10.1016/j.irfa.2024.103221.

[7] F. Audrino, F. Sigrist, and D. Ballinari, "The impact of sentiment and attention measures on stock market volatility," *Int. J. Forecast.*, vol. 36, no. 2, pp. 334–357, 2020, doi: https://doi.org/10.1016/j.ijforecast.2019.05.010.

[8] R. A. Al-Qazzaz and S. A. Yousif, "High performance time series models using auto autoregressive integrated moving average," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 27, no. 1, pp. 422–430, Jul. 2022, doi: 10.11591/ijeecs.v27.i1.pp422-430.

[9] A. T. Mustafa and O. S. A.-D. Al-Yozbaky, "Forecasting energy demand and generation using time series models: A comparative analysis of classical, grey, fuzzy, and intelligent approaches," *Franklin Open*, vol. 12, p. 100350, 2025, doi: https://doi.org/10.1016/j.fraope.2025.100350.

[10] Z. F. Althobaiti, "Improved forecasting of carbon dioxide emissions using a hybrid SSA ARIMA model based on annual time series data in Bahrain," *Sci. Rep.*, vol. 15, no. 1, p. 25699, 2025, doi: 10.1038/s41598-025-11343-w.

[11] A. Bhambu, K. Bera, S. Natarajan, and P. N. Suganthan, "High frequency volatility forecasting and risk assessment using neural networks-based heteroscedasticity model," *Eng. Appl. Artif. Intell.*, vol. 149, p. 110397, 2025, doi: https://doi.org/10.1016/j.engappai.2025.110397.

[12] V. Arumugam and V. Natarajan, "Time Series Modeling and Forecasting Using Autoregressive Integrated Moving Average and Seasonal Autoregressive Integrated Moving Average Models," *Instrumentation Mesure Metrologie*, vol. 22, no. 4, pp. 161–168, Aug. 2023, doi:10.18280/i2m.220404

[13] Castillo, Mauricio Soto, Ricardo Crawford, Broderick Castro, Carlos Olivares, and Rodrigo, "A Knowledge-Based Hybrid Approach on Particle Swarm Optimization Using Hidden Markov Models," *Mathematics*, vol. 9, p. 1417, Jun. 2021, doi:https://doi.org/10.3390/math9121417.

[14] B. Gülmez, "A Hybrid Approach for Stock Market Price Forecasting Using Long Short-Term Memory and Seahorse Optimization Algorithm," *Annals of Data Science*, pp. 1–27, May 2025, doi: 10.1007/s40745-025-00609-9.

[15] Ecer, Fatih Ardabili, Sina Band, Shahab S. Mosavi, and Amir, "Training Multilayer Perceptron with Genetic Algorithms and Particle Swarm Optimization for Modeling Stock Price Index Prediction," *Entropy*, vol. 22, p. 1239, Oct. 2020, doi:https://doi.org/10.3390/e22111239.

[16] R. B. B. de Holanda and J. F. L. de Oliveira, "Swarm-Based Ensembles for Time Series Residual Forecasting," in *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, 2020, pp. 595–602. 10.1109/ICTAI50040.2020.00097.

[17] E. H. Houssein, M. K. Saeed, G. Hu, and M. M. Al-Sayed, "Metaheuristics for Solving Global and Engineering Optimization Problems: Review, Applications, Open Issues and Challenges," Dec. 01, 2024, *Springer Science and Business Media B.V.* doi: 10.1007/s11831-024-10168-6.

[18] M. Á. Castán-Lascorz, A. Alcaide-Moreno, and J. Arroyo, "Optimizing Rotary Cement Kiln modelling: A comparative analysis of metaheuristics in a real-world application," *Results in Engineering*, vol. 25, p. 103945, 2025, doi: https://doi.org/10.1016/j.rineng.2025.103945.

[19] Y. Baig and S. Ahmed, "Advancing wind speed forecasting accuracy with a novel ARIMA-GA hybrid model and enhanced parameter optimization," *Sādhanā*, vol. 51, no. 1, p. 54, 2026, doi: 10.1007/s12046-026-03071-2.

[20] M. L. Hossain, S. M. N. Shams, and S. M. Ullah, "Time-series and deep learning approaches for renewable energy forecasting in Dhaka: a comparative study of ARIMA, SARIMA, and LSTM models," *Discover Sustainability*, vol. 6, no. 1, Dec. 2025, doi: 10.1007/s43621-025-01733-5.

[21] Z. Mohammed, C. Anas, and M. El Hammoumi, "A hybrid learning framework for forecasting uncertainty and adaptive inventory planning in retail supply chains," *Supply Chain Analytics*, vol. 13, p. 100180, 2026, doi: https://doi.org/10.1016/j.sca.2025.100180.

[22] Z. Zhang, X. Mo, H. Li, Z. Zhang, and P. Guo, "A multi-step hybrid forecasting model based on optimized secondary decomposition and TCN-BiLSTM with residual learning," *Discover Computing*, vol. 29, no. 1, p. 108, 2026, doi: 10.1007/s10791-026-10011-5.

[23] W. Bao, Y. Cao, Y. Yang, H. Che, J. Huang, and S. Wen, "Data-driven stock forecasting models based on neural networks: A review," *Information Fusion*, vol. 113, p. 102616, 2025, doi: https://doi.org/10.1016/j.inffus.2024.102616.

[24] Y. Dong, L. Du, and G. Li, "Hybrid Ship Design Optimization Framework Integrating a Dual-Mode CFD–Surrogate Mechanism," *Applied Sciences*, vol. 15, no. 19, p. 10318, Sep. 2025, doi: 10.3390/app151910318.

[25] M. Gwabavu, R. C. Bansal, and A. Bryce, "Hybrid Intelligent Optimisation for Onshore Wind Farm Forecasting," *Smart Grids and Sustainable Energy*, vol. 10, no. 3, p. 65, 2025, doi: 10.1007/s40866-025-00293-x.

[26] K. Yousef, B. Yuce, and A. He, "A Hybrid Deep Learning Framework for Wind Speed Prediction with Snake Optimizer and Feature Explainability," *Sustainability (Switzerland)*, vol. 17, no. 12, Jun. 2025, doi: 10.3390/su17125363.

[27] M. Almsallti, A. B. Alzubi, and O. R. Adegboye, "Hybrid Metaheuristic Optimized Extreme Learning Machine for Sustainability Focused CO2 Emission Prediction Using Globalization-Driven Indicators," *Sustainability (Switzerland)*, vol. 17, no. 15, Aug. 2025, doi: 10.3390/su17156783.

[28] W. Zhao, Z. H. Azizul, C. S. Woo, W. Kuang, and Y. Li, "Potential-driven multi-learning particle swarm optimisation," *Swarm Evol. Comput.*, vol. 96, p. 101993, 2025, doi: https://doi.org/10.1016/j.swevo.2025.101993.

[29] M. Jain, V. Saihjpal, N. Singh, and S. B. Singh, "An Overview of Variants and Advancements of PSO Algorithm," Sep. 01, 2022, *MDPI*. doi: 10.3390/app12178392.

[30] M. Esro, S. K. Subramaniam, A. F. T. Ibrahim, Y. J. Kumar, S. A. Anas, and S. Rajkumar, "A Comparative Analysis of Time-Series Models of ARIMA and Prophet IoT-Based Flood Forecasting in Sungai Melaka," *Advance Sustainable Science, Engineering and Technology*, vol. 7, no. 4, Aug. 2025, doi: 10.26877/asset.v7i4.1048.

[31] A. Nyabadza and D. Brabazon, "Machine Learning-Based Recommender System for Pulsed Laser Ablation in Liquid: Recommendation of Optimal Processing Parameters for Targeted Nanoparticle Size and Concentration Using Cosine Similarity and KNN Models," 2025, doi: 10.3390/10.3390/cryst15070662.

[32] I. Mawardi *et al.*, "Hybrid early warning system: Integration of Z-score and machine learning for predicting financial performance of IRB in Indonesia," *Journal of Open Innovation: Technology, Market, and Complexity*, vol. 12, no. 1, p. 100694, 2026, doi: https://doi.org/10.1016/j.joitmc.2025.100694.

[33] S. Diaz *et al.*, "A hybrid LSTM-XGBoost model with residual correction for air quality prediction using SSA," *Air Qual. Atmos. Health*, vol. 18, no. 12, pp. 3991–4008, 2025, (2025). https://doi.org/10.1007/s11869-025-01867-5.