Mathematical Biosciences and Engineering

MBE, 22(8): 2020–2038.
DOI: 10.3934/mbe.2025074
Received: 04 April 2025
Revised: 31 May 2025
Accepted: 18 June 2025
Published: 26 June 2025

https://www.aimspress.com/journal/MBE

Research article

Model identification of ventilation air pump utilizing Ridge-momentum regression and Grid-based structure optimization

Cong Toai Truong[1,2], Trung Dat Phan[1,2], Van Tu Duong[1,2], Huy Hung Nguyen[3] and Tan Tien Nguyen[1,2,*]

[1] Key Laboratory of Digital Control and System Engineering (DCSELab), Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam

[2] Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc City, Ho Chi Minh City, Vietnam

[3] Faculty of Electronics and Telecommunication, Sai Gon University

*** Correspondence:** Email: nttien@hcmut.edu.vn.

Abstract: Historically, the world has endured numerous respiratory pandemics, with the recent COVID-19 outbreak underscoring the significant importance of respiratory equipment and mechanical ventilators being no exception. Despite long-standing efforts in control and modeling system research, mechanical ventilators, especially the air generation unit, remain a significant challenge due to various factors and uncertainties (e.g., model structure, order selection, time-varying parameters, etc.). This paper presents a novel approach for identifying ARMA models, specifically in ventilation pumps, using Ridge regression modified with momentum (Ridge-M) and a grid search-based joint optimization strategy. The proposed algorithm effectively estimates model coefficients while simultaneously selecting the optimal AR and MA orders along with time-delay parameters. By integrating momentum into Ridge regression, the estimation process gains stability and improved convergence, particularly in handling abrupt system changes. The grid search framework ensures robust model selection by systematically evaluating candidate structures using the Akaike Information Criterion (AIC). Experimental validation with multiple input functions, including ramp and multistep signals, demonstrates that Ridge-M achieves superior performance in capturing dynamic system behaviors. Ridge-M reduces the root mean squared error (RMSE) by 2.7% on average across multistep inputs for both scenarios compared to recursive least squares and 6.8% compared to standard Ridge regression. However, standard Ridge outperforms Ridge-M for ramp inputs for both scenarios, reducing RMSE by 0.7%, indicating that momentum can slow adaptation to gradual variations.

Nonetheless, Ridge-M achieves the lowest overall average RMSE (31.6236) compared to RLS (34.1499) and standard Ridge regression (32.0247), confirming its superior balance between stability and adaptability in model identification. This work offers a lightweight and stable method that is well-suited for embedded applications where data is noisy, the system is time-varying, and computational resources are limited.

Keywords: ARMA model; Ridge regression; momentum; grid search; model identification

1. Introduction

In the field of robotics and control systems, designing an accurate and efficient controller remains a challenging task, particularly when system dynamics are unknown or highly complex [1]. Many advanced control strategies, such as model predictive control [2], sliding mode control [3], and optimal control [4], rely on precise mathematical models to ensure stability and robustness. Furthermore, accurately modeling system dynamics offers significant advantages across various domains, including trajectory tracking in robotics [5], state-of-charge prediction in battery management systems [6], modeling infinite impulse response systems in the field of digital filter design [7], parameters determination for solar photovoltaic panel [8] and permanent magnet synchronous machine [9], and energy consumption parameters prediction in plug-in hybrid electric vehicles [10], among other applications. However, in practical applications, such models are either readily or difficult to derive, necessitating system identification techniques. Beyond control engineering, system modeling plays a crucial role in spreading fields such as economics [11], biomedical engineering [12], and financial forecasting [13], where data-driven models enhance decision-making and predictive accuracy.

Among various types of models, a discrete-time model is evident in its suitability for implementation on digital devices like microcontrollers and computers [14]. Within this domain, the autoregressive moving average (ARMA) model has been extensively used in time-series described for single-input single-output (SISO) systems [1,14–18]. Introduced several decades ago, the ARMA model provides a compact representation of system dynamics, requiring fewer parameters compared to high-order differential equation models. As a result, ARMA models have become a widely adopted framework for system identification, especially in control applications that demand accurate estimation of system dynamics from observed data [20]. The effectiveness of ARMA models in capturing system dynamics has driven extensive research into identification techniques that estimate their parameters with high accuracy and computational efficiency [21].

Numerous methods have been developed for efficient ARMA model identification, focusing on estimating both model coefficients and orders for the autoregressive and moving average components. One of the earliest techniques, proposed by Graupe et al. [21], introduced a least squares (LS)-based method for sequentially estimating ARMA parameters. Subsequently, Li et al. [18] combined LS regression with an iterated estimation approach, demonstrating improved parameter accuracy. A major breakthrough came with the orthogonal search-based ARMA estimation method proposed by Paarmann and Korenberg [15], which employed an orthogonalized regression approach to iteratively refine parameter estimates, reducing variance compared to conventional LS methods. Further advancements included rational approximation techniques [17], which reformulated ARMA system identification as a convex optimization problem, improving convergence properties. In the late 1990s, Sabiti et al. [19] introduced a fast estimation method using maximum likelihood estimation for ARMA processes, reducing computational complexity while maintaining accuracy, making it suitable for large-scale datasets. More recently, studies such as [16] analyzed estimation algorithms for nonstationary ARMA processes, highlighting parameter accuracy variations under time-varying conditions and proposing a recursive least squares (RLS) identification algorithm. Beyond traditional estimation methods, recent research has explored nonlinear and quantized identification approaches. Lu and Chon [22] extended ARMA identification to nonlinear systems by minimizing hypersurface distance, while Marelli et al. [23] and Yu et al. [24] developed techniques for ARMA model identification using quantized and intermittent observations, improving estimation in low-resolution and energy-constrained systems. Ding et al. [25] further introduced an iterative filtering-based method for ARMA systems under noisy environments, demonstrating enhanced robustness in practical applications.

On the other hand, Ridge regression has emerged as a powerful tool in statistical modeling, particularly in addressing issues of multicollinearity and improving generalization in regression problems [26]. Specifically, it offers the advantage of stable estimates, especially for high-order AR models, which shrink coefficients, reducing variance and potentially improving generalization in out-of-sample forecasting [27]. Additionally, momentum-based optimization [28] could accelerate convergence in non-convex likelihood landscapes of ARMA models, reduce oscillations near optima, and improve stability of gradient-based methods for parameter estimation. Notably, the underutilization of Ridge and momentum methods in ARMA modeling is attributed to several key factors. Initially, ARMA modeling was developed primarily within traditional statistical settings between the 1970s and 1990s, focusing on interpretability and simplicity [29]. In contrast, Ridge regression and momentum-based optimization emerged later in the context of machine learning, addressing high-dimensional and nonlinear problems [30,31]. In essence, the ARMA model prioritizes selecting small model orders (p, q) to avoid overfitting [32], whereas Ridge regression assumes high-dimensional feature spaces and controls model complexity through coefficient shrinkage [33]. Moreover, ARMA estimation techniques, such as Maximum Likelihood or Yule-Walker are not designed to incorporate such modifications [34,35]; they typically involve relatively few parameters. As a result, conventional estimation approaches such as least squares (LS) or maximum likelihood (ML) are often sufficient. This diminishes the perceived need for regularization or momentum-driven techniques that are better suited for high-dimensional settings. Nonetheless, recent advancements have extended Ridge regression beyond traditional linear models to applications in recursive estimation [36], time-varying parameter modeling [37], and bilevel optimization [38], demonstrating its versatility in dynamic and high-dimensional systems. By introducing ℓ_2-norm penalty, Ridge regression stabilizes coefficient estimation, prevents overfitting, and enhances numerical conditioning [39]. Despite these developments, limited attention has been given to leveraging Ridge regression for ARMA model identification. As analyzed above, the ARMA model is widely used for time-series analysis, yet conventional estimation techniques, such as LS and ML, often struggle with multicollinearity and high variance in parameter estimation. While recursive Ridge regression techniques have been explored, their optimization strategies have been effectively employed for hyperparameter tuning [40], yet their integration with Ridge regression in ARMA identification is largely unexplored.

To bridge this gap, this paper proposes an enhanced Ridge regression framework that incorporates a momentum-based regularization strategy, improving upon traditional Ridge methods [41] by penalizing deviation not from zero but from a dynamically updated momentum vector. This not only mitigates high-variance parameter estimates but also accelerates convergence and stabilizes the estimation process. Furthermore, the optimal ARMA model order and input delay are systematically selected via an AIC-based grid search, automating model structure selection to ensure neither overfitting nor underfitting, an approach not addressed in existing literature, to the best of our knowledge. By combining momentum-based updates with joint optimization of model structure, our approach delivers a unified solution that enhances parameter estimation stability and adaptability in dynamic systems, an integration not addressed in existing literature on recursive Ridge schemes [42,43]. The main contributions of this paper are as follows:

- The proposed Ridge-M method enhances parameter estimation stability and accuracy, addressing the limitations of conventional least squares and RLS approaches.
- By incorporating momentum, the proposed approach mitigates parameter fluctuations and accelerates convergence, particularly for control systems with abrupt changes.
- A systematic optimization process is utilized to determine the optimal ARMA order and time delay, ensuring the accuracy and reliability of the identification model.
- The proposed method is evaluated across different input scenarios, including ramp and multistep functions, demonstrating its superiority over standard Ridge regression and RLS in handling sudden changes and improving model consistency.

2. Materials and methods

2.1. System description

An ARMA model based on input-output experimental data observations in a multi-step-ahead manner can be expressed as follows:

$$y_t = \sum_{i=1}^{p} \theta_i y_{t-i} + \sum_{j=0}^{q-1} \varphi_j u_{t-j-d} + \varepsilon_t, \tag{1}$$

where $\{y_t\}_{t=1}^{N}$, y_t and y_{t-i} represent the model output at the times t and $t-i$, respectively. The term u_{t-j-d} denotes the control input at the time $t-j$ delayed by d, while θ_i and φ_j are the coefficients of the p-order autoregressive and q-order moving average components, respectively. The output observations are subjected to white noise with zero mean and a normal Gaussian distribution.

The system model in (1) can be expressed in a linear regression vector form using N observations as follows:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w} + \boldsymbol{\varepsilon}, \tag{2}$$

where the model output vector is given by

$$\boldsymbol{y} \coloneqq [y_1, y_2, \dots, y_N]^T \in \mathbb{R}^N, \tag{3}$$

subjected to the white noise vector

$$\boldsymbol{\varepsilon} := [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_N]^T \in \mathbb{R}^N. \tag{4}$$

Then, the $(p+q)$-dimensional model coefficient vector is defined as

$$\boldsymbol{w} := \left[\theta_1, \theta_2, \dots, \theta_p, \varphi_0, \varphi_2, \dots, \varphi_{q-1}\right]^T \in \mathbb{R}^{p+q}, \tag{5}$$

which is associated with a Toeplitz-structured feature matrix constructed from all input-output observations, given by

$$\boldsymbol{X} := \begin{bmatrix} 0 & 0 & \cdots & 0 & u_{1-d} & 0 & \cdots & 0 \\ y_1 & 0 & \cdots & 0 & u_{2-d} & u_{1-d} & \cdots & 0 \\ y_2 & y_1 & \cdots & 0 & u_{3-d} & u_{2-d} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \cdots & 0 \\ y_p & y_{p-1} & \cdots & y_1 & u_{p+1-d} & u_{p-d} & \cdots & u_{p-d-q+2} \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ y_{N-1} & y_{N-2} & \cdots & y_{N-p} & u_{N-d} & u_{N-1-d} & \cdots & u_{N-q-d+1} \end{bmatrix} \in \mathbb{R}^{N\times(p+q)}. \tag{6}$$

Assumption 1. The control input sequence $\{u_t\}$ is assumed to be persistently exciting, ensuring that the input data spans a sufficiently rich subspace, preventing any column in $\boldsymbol{X}$ from being a linear combination of the others. Additionally, the number of observations must satisfy $N \gg p+q$, ensuring that $\boldsymbol{X}$ contains enough data points to avoid rank deficiency. Under these conditions, the Toeplitz-structured feature matrix $\boldsymbol{X}$ is full column rank, i.e., $\mathrm{rank}(\boldsymbol{X}) = p+q$.

Assumption 2. It is assumed that there exists a positive constant $\mho$ such that the maximum value of the control input sequence satisfies $\{u_t\}_{\max} \le \mho$.

Remark 1. In practice, the system model in (2) contains an unknown model coefficient vector with an implicit dimension, which must be identified using observation data to minimize the difference between the true model output and the estimated output.

Let $\widehat{\boldsymbol{w}}$ be an estimated coefficient vector of $\boldsymbol{w}$. The estimated output $\widehat{\boldsymbol{y}}$ of the true output vector $\boldsymbol{y}$ can then be derived by the following auxiliary model

$$\widehat{\boldsymbol{y}} = \boldsymbol{X}\widehat{\mathbf{w}}, \tag{7}$$

where the estimated coefficient vector is defined as

$$\widehat{\mathbf{w}} = \left[\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_p, \hat{\varphi}_0, \hat{\varphi}_2, \dots, \hat{\varphi}_{q-1}\right]^T \in \mathbb{R}^{p+q}. \tag{8}$$

2.2. Formation of coefficient identification

This paper proposes an identification algorithm for model coefficients using modified Ridge regression, which incorporates a residual sum-of-squares term, and a $\mathcal{L}_2$-norm regularization term with momentum. The estimated coefficient vector is determined by selecting sufficient small criteria to minimize the residual between the true model output and the estimated output.

The modified Ridge regression loss function $J(\widehat{\boldsymbol{w}}) \in \mathbb{R}$ is defined as

$$J(\widehat{\boldsymbol{w}}) = \frac{1}{2}(\|\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{w}}\|_2^2 + \gamma\|\widehat{\boldsymbol{w}} - \boldsymbol{v}\|_2^2), \tag{9}$$

where γ is a positive regularization hyperparameter, i.e., $\gamma \in \mathbb{R}^+$, which controls the shrinkage of model coefficients that have minimal impact on the estimated output and regulates the strength of the momentum regularization.

The modified loss function introduces $\boldsymbol{v} \in \mathbb{R}^{p+q}$ as the accumulated velocity, which is updated iteratively using the following rule

$$\boldsymbol{v}_{k+1} = \beta\boldsymbol{v}_k + (1-\beta)\widehat{\boldsymbol{w}}_k, \tag{10}$$

with $\beta \in [0, 1]$ defined as a momentum weight which controls the retention of the past velocity, and $\widehat{\boldsymbol{w}}_k$ is updated according to the extremum value theorem.

Taking the derivative of the modified loss function with respect to $\widehat{\boldsymbol{w}}^T$ and applying the vector calculus property $\partial\boldsymbol{x}^T\boldsymbol{A}/\partial\boldsymbol{x} = \boldsymbol{A}^T$, it can be obtained that

$$\begin{aligned}\frac{\partial J(\widehat{\boldsymbol{w}})}{\partial\widehat{\boldsymbol{w}}^T} &= \frac{1}{2}\frac{\partial}{\partial\widehat{\boldsymbol{w}}^T}\left((\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{w}})^T(\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{w}}) + \gamma(\widehat{\boldsymbol{w}} - \boldsymbol{v})^T(\widehat{\boldsymbol{w}} - \boldsymbol{v})\right) \\ &= \frac{1}{2}\frac{\partial}{\partial\widehat{\boldsymbol{w}}^T}\left(\boldsymbol{y}^T\boldsymbol{y} - 2\boldsymbol{y}^T\boldsymbol{X}\widehat{\boldsymbol{w}} + \widehat{\boldsymbol{w}}^T\boldsymbol{X}^T\boldsymbol{X}\widehat{\boldsymbol{w}} + \gamma(\widehat{\boldsymbol{w}}^T\widehat{\boldsymbol{w}} - 2\widehat{\boldsymbol{w}}^T\boldsymbol{v} + \boldsymbol{v}^T\boldsymbol{v})\right) \\ &= -\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{X}^T\boldsymbol{X}\widehat{\boldsymbol{w}} + \gamma\widehat{\boldsymbol{w}} - \gamma\boldsymbol{v}.\end{aligned} \tag{11}$$

Setting this expression to zero to find the extremum, the recurrence update rule for the estimated coefficient vector is given by

$$\widehat{\boldsymbol{w}}_{k+1} = (\boldsymbol{X}^T\boldsymbol{X} + \gamma\boldsymbol{I})^{-1}(\boldsymbol{X}^T\boldsymbol{y} + \gamma\boldsymbol{v}_k), \tag{12}$$

where $\boldsymbol{I}$ represents the identity matrix with appropriate dimensions.

Lemma 1. *Let $\boldsymbol{X} \in \mathbb{R}^{N\times(p+q)}$ be a real, unsymmetric matrix in a finite-dimensional Euclidean space. The Gramian matrix, defined as $\boldsymbol{X}^T\boldsymbol{X} \in \mathbb{R}^{(p+q)\times(p+q)}$, is symmetric, positive definite, and self-adjoint matrix. Consequently, all its eigenvalues are real and positive. Thus, the matrix $\boldsymbol{X}^T\boldsymbol{X} + \gamma\boldsymbol{I}$ is invertible for any $\gamma > 0$.*

Proof: See Appendix A.

Theorem 1. *Assume that there exists an optimal solution $\boldsymbol{w}^*$ that minimizes the modified loss function in* (9) and satisfies $\boldsymbol{X}\boldsymbol{w}^* = \boldsymbol{y}$. *Given the hyperparameters $\lambda > 0$ and $\beta < 1$, the estimated coefficient vector $\widehat{\boldsymbol{w}}$ converges asymptotically to $\boldsymbol{w}^*$ when applying the update rule* (12) *and the accumulated momentum* (10).

Proof: See Appendix B.

2.3. Joint optimization for coefficient Orders and delayed time

In practice, determining the ARMA model order and input delay time solely from experimental observations can be challenging, implying that the orders of the AR and MA components are unknown and must be identified. Rather than relying on heuristic approaches, this study employs grid search (GS) over a predefined range and evaluates each candidate model using the Akaike information criterion (AIC) to systematically determine the optimal ARMA order (p, q). The AIC metric balances model fit and complexity, ensuring the selection an order that generalizes well. The key principle of the GS is to discretize the design space into a finite number of partitions, evaluate the corresponding loss function for each, and identify the optimal design where the extremum is located.

Let $\mathcal{P} = \{p_{min}, \dots, p_{max}\} \in \mathbb{N}^+$, $\mathcal{Q} = \{q_{min}, \dots, q_{max}\} \in \mathbb{N}^+$, and $\mathcal{D} = \{d_{min}, \dots, d_{max}\} \in \mathbb{N}^+$ define the candidate sets for the AR order, the MA order and input delay time, respectively. The optimal order pair (p^*, q^*, d^*) is chosen by minimizing AIC, given by

$$(p^*, q^*, d^*) = \underset{(p,q,d)\in\mathcal{G}}{\operatorname{argmin}} \operatorname{AIC}(p, q, d), \tag{13}$$

where $\mathcal{G} = \mathcal{P} \times \mathcal{Q} \times \mathcal{D}$ is the search grid. The AIC loss function for each candidate model is computed as follows:

$$\operatorname{AIC}(p, q) = N \log(\delta) + \alpha(p + q + d), \tag{14}$$

where $\alpha > 0$ is a regularization weight that controls the trade-off between model complexity and goodness of fit in the AIC, and δ is the estimated variance of residuals, which is defined as

$$\delta = \frac{1}{N} \|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2. \tag{15}$$

According to the theoretical framework discussed above, the identification algorithm for the ARMA model with input time delay is summarized in Algorithm 1.

ALGORITHM 1: Ridge regression with momentum for ARMA model identification

1. Set $\gamma, \beta, \alpha,$ max iterations, tolerance ϵ. For $t \leq 0$, set all initial values to zero.
2. Collect the input-output data $u(t)$, $y(t)$.
3. Set candidate ranges for $p, q,$ and d.
4. For each combination of $(p, q, d) \in \mathcal{G}$:
   - Construct Toeplitz-structured feature matrix $\boldsymbol{X}$ based (6);
   - Iterate until convergence or reaching max iterations:
     - Compute $\hat{\boldsymbol{w}}$ using the update rule (12);
     - Update the momentum term using (10) and $\hat{\boldsymbol{w}}$;
     - Check for convergence: $\|\hat{\boldsymbol{w}}_t - \hat{\boldsymbol{w}}_{t-1}\| < \epsilon$:

       If satisfied or max iterations reached, terminate.

- Compute model error using the AIC-based loss function (14);

5. Identify p^*, q^*, d^* and corresponding optimal coefficient vector $\hat{\boldsymbol{w}}$ that minimizes the model error.

3. Results

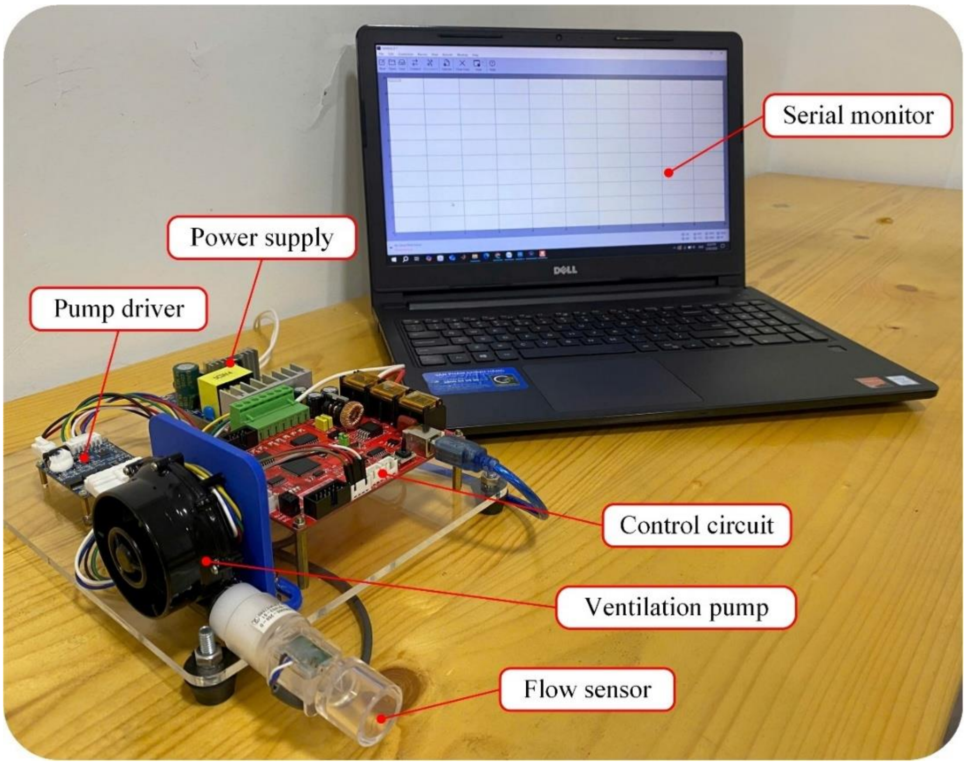

Figure 1. Experimental setup.

The experimental apparatus used to collect data is depicted in Figure 1. The system consists of a brushless DC ventilation pump, a proximity flow sensor SFM3300, a pump driver, and a customized control circuit that includes an ATMEGA2560A microcontroller with a fixed sampling period of 0.01 seconds. PWM signals from the microcontroller regulated the pump's input voltage through a driver, thereby controlling airflow. The flow sensor continuously measures airflow and transmits the data to the control circuit through I2C communication. This data is stored on the computer and used for the system identification using the proposed algorithm. The entire system was powered by a regulated 24V DC power supply, and the pump outlet discharged directly into ambient air. On this system's measurement uncertainty, noise is presented mainly from the inherent properties of the SFM3300 sensor, with the typical noise level of 3% of the measured value. Additionally, zero-point offset noise has a typical number of 0.1 liters per minute at standard conditions. To comprehensively evaluate the system's response, four distinct control inputs are sequentially applied to the pump driver to observe the airflow: two ramp functions with different slopes and two multistep functions with varying levels, designated as A-Ramp, B-Ramp, A-Multistep, and B-Multistep.

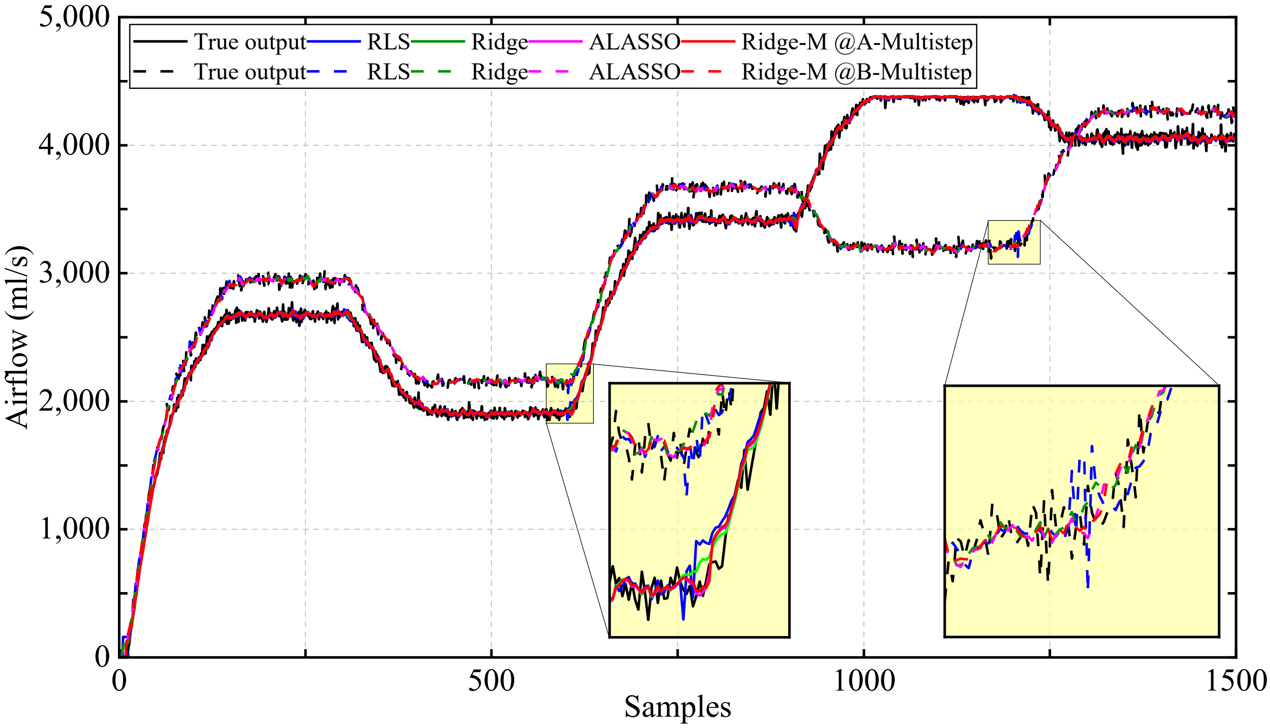

Figure 2. Estimated outputs from different identification algorithms using model coefficients obtained from training data corresponding to the A-Multistep input function.

Initially, data observation excited by the A-Multistep function is used to identify model coefficients, AR and MA orders, and time delay using the proposed algorithm (Ridge-M) with the hyperparameter $\gamma = 1000$, and $\alpha = 12$. The candidate sets are defined as $\mathcal{P} = \{1, \dots, 6\}, \mathcal{Q} = \{1, \dots, 6\}$, and $\mathcal{D} = \{0, \dots, 6\}$. Regarding the hyperparameter β, as demonstrated in Theorem 1, it determines the convergence capability of the estimated coefficient vector $\boldsymbol{w}$ toward the ground truth vector $\boldsymbol{w}^*$, provided that $\beta \in [0,1]$. Therefore, β does not significantly affect the estimation error but does influence the convergence speed of the algorithm. As a result, the authors suggest choosing a β value close to 0, with a recommended value of 0.01, to achieve the fastest convergence and reduce the computational time required by the proposed algorithm. The influence of β on the convergence speed of the proposed algorithm is described in Figure 3. Consequently, the identification results obtained using the proposed algorithm yield estimated coefficients for an ARMA model with orders ($p = 5, q = 6$ and $d = 4$). For comparative analysis, the same dataset is employed to estimate model coefficients for a fixed ARMA(5,6) structure using standard Ridge regression without momentum (Ridge) with the same hyperparameter γ, recursive least squares (RLS), and adaptive least absolute shrinkage and selection operator (ALASSO). For cross-validation, the coefficients obtained from the A-Multistep function are further validated using data from other input functions.

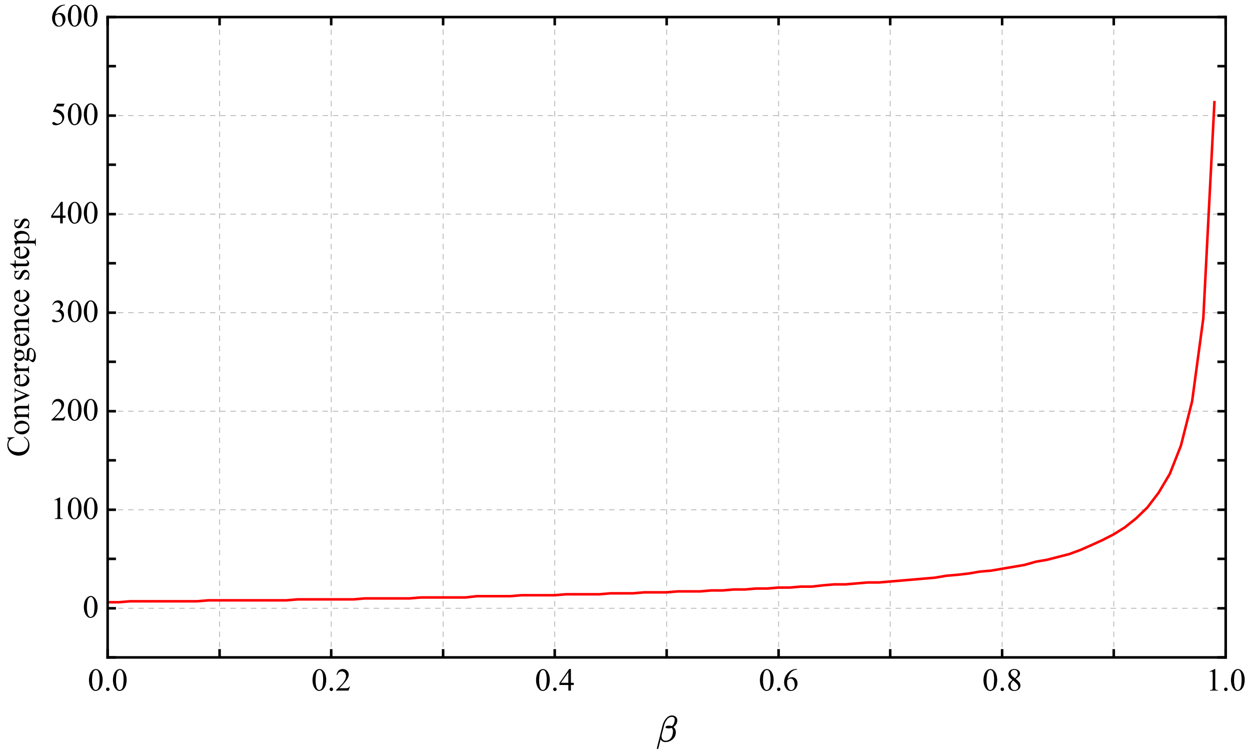

Figure 3. Influence of β on the convergence speed of the proposed algorithm.

Figure 2 illustrates the airflow generated by the ventilation pump alongside its estimated output, predicted using three coefficient models validated with data from the A-Multistep and B-Multistep functions for a fixed ARMA (5,6,4). It can be seen that the estimated outputs from the four models closely align with the true output. However, the output estimated by RLS exhibits greater fluctuation compared to the other algorithms, particularly when the step input function changes levels. To further assess the performance of the proposed algorithm, the root mean squared error (RMSE) metric is used, defined as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2}. \tag{16}$$

The three estimated models identified using the A-Multistep function are validated through cross-validation using data from other input functions, with RMSE values presented in Table 1. Compared to the Ridge model, the Ridge-M model achieves the lower RMSE for multistep functions (34.6729 and 33.7559), while the Ridge model performs best for ramp functions (28.8954 and 26.1665). This suggests that momentum enhances the handling of sudden changes by smoothing updates over time but may hinder adaptation to gradual changes, as seen in ramp inputs. However, when compared to modern techniques such as LASSO, the RMSE of Ridge-M is slightly higher for the A-Multistep, A-Ramp, and B-Ramp input functions, but remains highly competitive relative to the other two algorithms. Overall, Ridge-M remains competitive with an average RMSE of 31.0934, indicating that while momentum is beneficial, it is not always essential. Meanwhile, RLS consistently underperforms, likely due to weaker regularization or slower adaptation. Although highly adaptive, it is prone to instability, making it less suitable for environments with sudden changes or noise. This is further evidenced by the estimated coefficients in Figure 4, where RLS exhibits significant fluctuations over time, particularly during abrupt changes.

Table 1. Performance comparison of RLS, Ridge, ALASSO, and Ridge-M using training data from the A-Multistep input function for the ARMA (5,6,4) model.

Input function	RLS	Ridge	ALASSO	Ridge-M
A-Multistep	37.6166	35.5458	33.623	34.6729
B-Multistep	36.5255	34.5349	33.896	33.7559
A-Ramp	30.6618	28.8954	29.221	29.2187
B-Ramp	28.3006	26.1665	26.553	26.7261
Average	33.2761	31.2856	31.050	31.0934

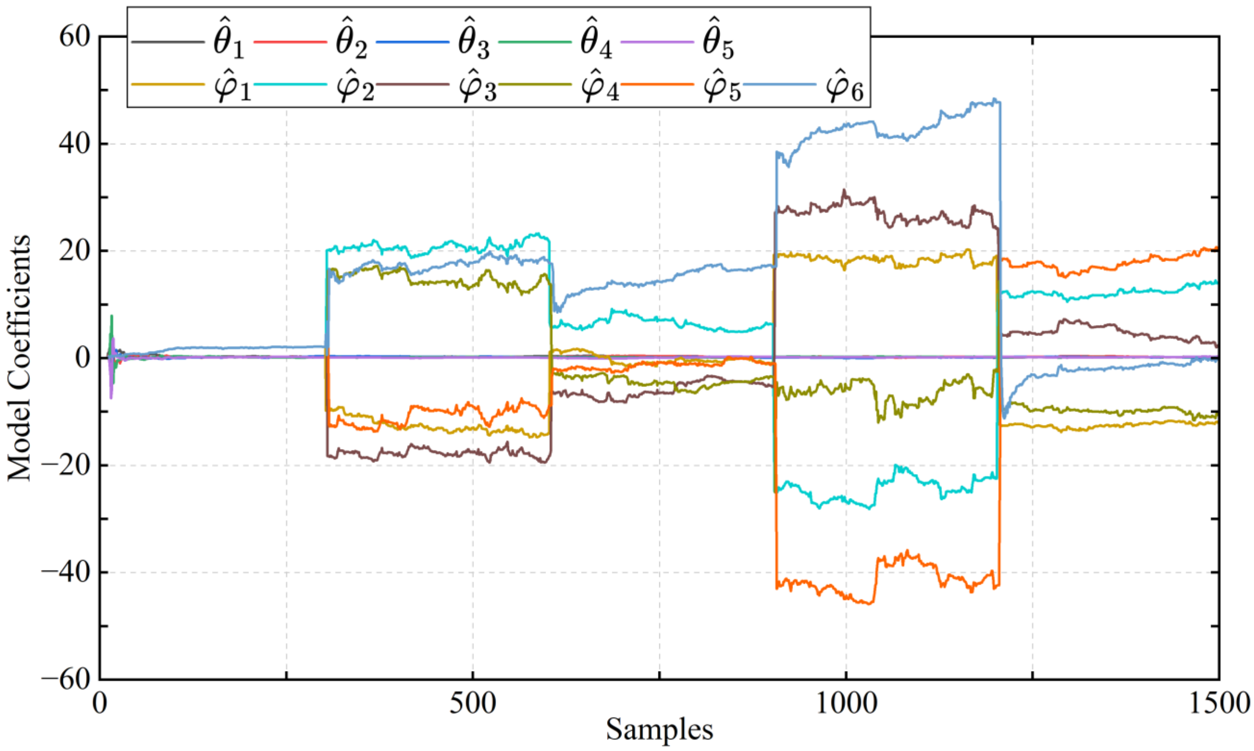

Figure 4. Coefficients obtained by RLS.

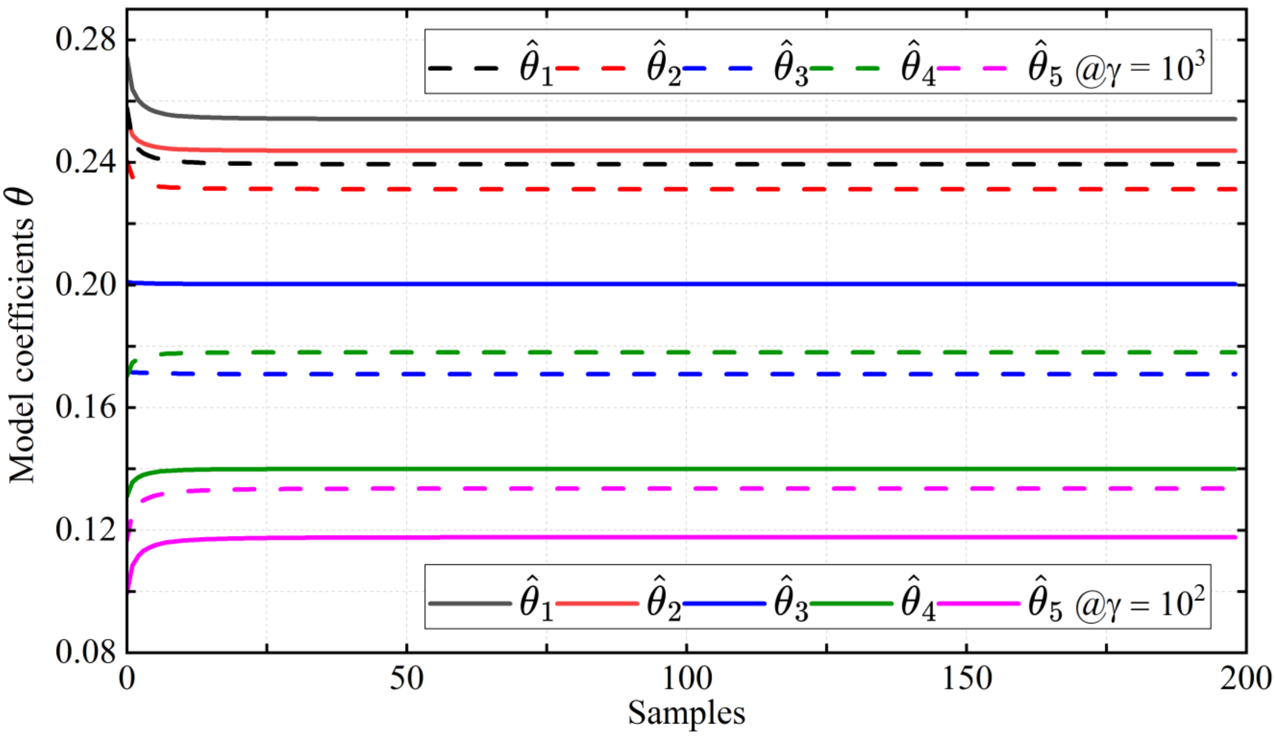

Figure 5. Model coefficients $\hat{\theta}_i$ obtained by Ridge-M with different hyperparameter γ.

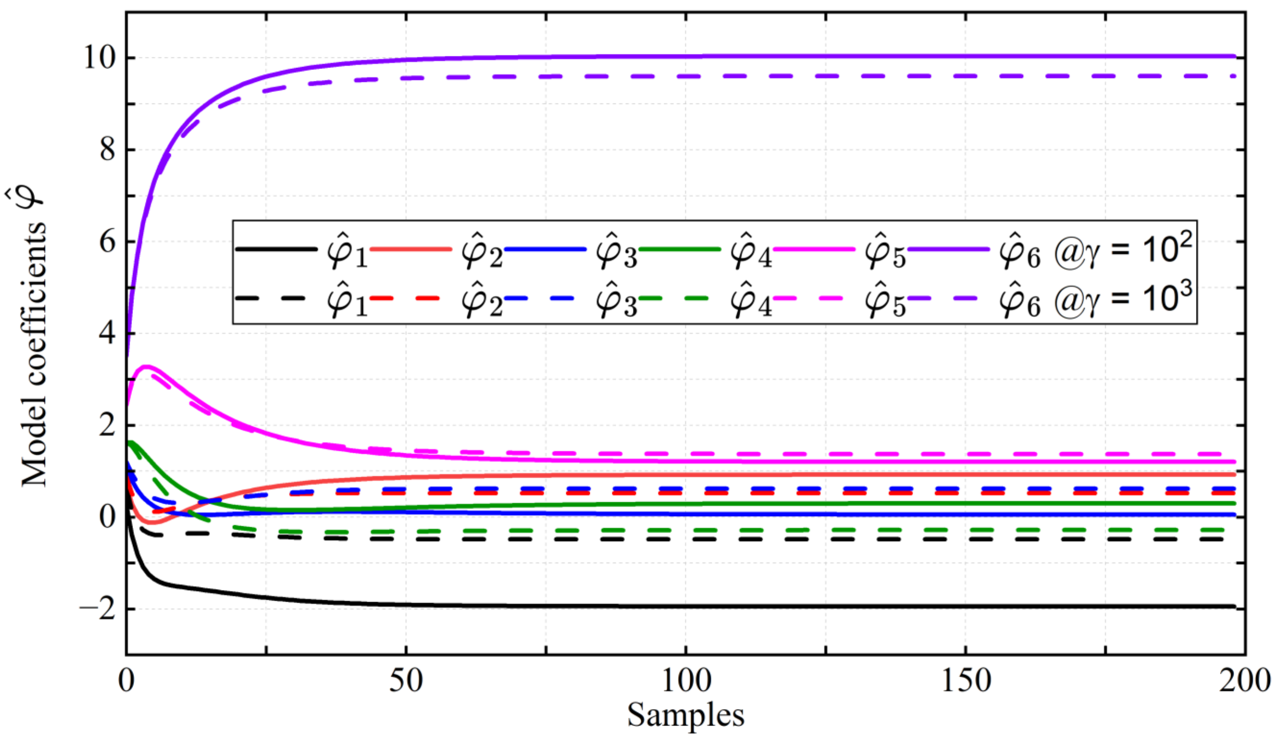

Figure 6. Model coefficients $\hat{\varphi}_i$ obtained by Ridge-M with different hyperparameter γ.

In contrast, Figures 5–6 show that the coefficients estimated by Ridge-M converge smoothly and remain stable across different regularization parameters ($\gamma = 10^2$ and $\gamma = 10^3$), demonstrating robustness and effective regularization. The choice of γ affects the convergence rate but maintains stability. Higher γ values result in smaller coefficient magnitudes and slower adaptation, indicating stronger regularization that smooths parameter updates. Lower γ values allow for faster adaptation but may introduce slight variations. However, even at $\gamma = 10^2$, Ridge-M remains significantly more stable than RLS.

The convergence of estimated coefficients across samples is illustrated in Tables 2–3. For both values of γ, the coefficients initially fluctuate but gradually converge as the number of samples increases. A larger γ provides smoother and more stable convergence, making it preferable in scenarios requiring strong regularization to prevent overfitting or noise amplification. A smaller γ allows for quicker adaptation but may introduce minor instability. The optimal γ depends on the trade-off between adaptability and stability required for the specific application.

Table 2. Estimated coefficients obtained by Ridge-M with the hyperparameter $\gamma = 100$.

Samples	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$	$\hat{\theta}_4$	$\hat{\theta}_5$	$\hat{\varphi}_1$	$\hat{\varphi}_2$	$\hat{\varphi}_3$	$\hat{\varphi}_4$	$\hat{\varphi}_5$	$\hat{\varphi}_6$
1	0.2741	0.2544	0.2010	0.1309	0.0993	0.2656	0.6870	1.0586	1.6253	2.4464	3.5124
30	0.2542	0.2438	0.2002	0.1399	0.1175	-1.8097	0.7191	0.1062	0.1514	1.6625	9.7322
60	0.2541	0.2437	0.2002	0.1399	0.1176	-1.9294	0.8885	0.0941	0.2341	1.2853	9.9951
90	0.2541	0.2437	0.2002	0.1399	0.1176	-1.9440	0.9156	0.0677	0.2797	1.2188	10.0305
120	0.2541	0.2437	0.2002	0.1399	0.1176	-1.9465	0.9219	0.0583	0.2928	1.2051	10.0368
150	0.2541	0.2437	0.2002	0.1399	0.1176	-1.9471	0.9237	0.0556	0.2963	1.2020	10.0381
180	0.2541	0.2437	0.2002	0.1399	0.1176	-1.9472	0.9241	0.0548	0.2971	1.2013	10.0384

Table 3. Estimated coefficients obtained by Ridge-M with the hyperparameter $\gamma = 1000$.

Samples	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$	$\hat{\theta}_4$	$\hat{\theta}_5$	$\hat{\varphi}_1$	$\hat{\varphi}_2$	$\hat{\varphi}_3$	$\hat{\varphi}_4$	$\hat{\varphi}_5$	$\hat{\varphi}_6$
1	0.2577	0.2404	0.1719	0.1703	0.1165	0.6334	0.8759	1.1954	1.6386	2.4921	3.5901
30	0.2394	0.2313	0.1709	0.1780	0.1334	-0.4414	0.5021	0.5347	-0.3196	1.6726	9.3935
60	0.2393	0.2313	0.1709	0.1780	0.1335	-0.4818	0.5225	0.6162	-0.3051	1.4141	9.5805
90	0.2393	0.2313	0.1709	0.1780	0.1335	-0.4822	0.5200	0.6180	-0.2884	1.3792	9.6003
120	0.2393	0.2313	0.1709	0.1780	0.1335	-0.4819	0.5197	0.6168	-0.2844	1.3736	9.6030
150	0.2393	0.2313	0.1709	0.1780	0.1335	-0.4819	0.5198	0.6164	-0.2835	1.3726	9.6035
180	0.2393	0.2313	0.1709	0.1780	0.1335	-0.4819	0.5198	0.6163	-0.2833	1.3725	9.6035

Additionally, the impact of γ on model performances, with and without momentum, is presented in Table 4. For small values ($\gamma = 10^{-1}{\sim}10^{0}$), RMSE remains relatively stable across both models, and the identified ARMA orders and delay remain unchanged, indicating minimal influence of momentum in this region. For moderate values ($\gamma = 10^{3}{\sim}10^{5}$), RMSE improves significantly at $\gamma = 10^{4}$ (best MSE = 25.8757) without momentum, while the performance with momentum is slightly worse but remains competitive. Moreover, momentum enhances robustness, as RMSE remains relatively stable despite variations in $p, q,$ and $d.$ At higher values, begins to increase, with a sharp rise observed at $\gamma = 10^{9}$ without momentum, whereas the model with momentum remains more stable. This highlights that momentum helps mitigate over-regularization effects, leading to a more consistent model structure. Overall, momentum enhances the identification algorithm's robustness, making it less sensitive to variations in the hyperparameter during model structure changes.

Table 4. Effectiveness of the hyperparameter γ with and without momentum for model coefficients trained using the A-Multistep function and validated with the B-Ramp function.

	Without momentum				With momentum			
γ	RMSE	p	q	d	RMSE	p	q	d
10^{-1}	26.726	5	6	4	26.7261	5	6	4
10^{0}	26.7253	5	6	4	26.7261	5	6	4
10^{1}	26.7187	5	6	4	26.7261	5	6	4
10^{2}	26.6686	5	6	4	26.7261	5	6	4
10^{3}	26.4337	5	4	6	26.7261	5	6	4
10^{4}	25.8757	4	5	0	26.7261	5	6	4
10^{5}	26.0333	4	6	0	26.7124	5	6	4
10^{6}	26.6083	3	1	0	26.6247	5	4	6
10^{7}	26.6624	3	1	0	26.3392	5	6	6
10^{8}	27.5191	3	1	0	26.0679	4	6	0
10^{9}	37.466	6	1	0	26.6497	3	1	0

In what follows, data obtained from the A-Ramp function is utilized to estimate model coefficients, AR and MA orders, and input time delay using the proposed Ridge-M algorithm with a hyperparameter of $\gamma = 1000$ and $\alpha = 12$. The proposed algorithm identifies an ARMA model with estimated orders of ($p = 5, q = 1$ and $d = 0$). For comparative evaluation, the same dataset is utilized to estimate model coefficients for a predefined ARMA (5,1) structure using standard Ridge regression without momentum (Ridge) with the same γ value, as well as RLS and ALASSO. To assess generalization, the coefficients identified from the A-Ramp function are further validated using data from other input functions.

Figure 7 depicts the airflow generated by the ventilation pump alongside its estimated output, predicted using four coefficient models validated with data from the A-Ramp and B-Ramp functions. It can be seen that all four models (RLS, Ridge, ALASSO, Ridge-M) follow the true output closely. Notably, in the case where the system input varies linearly without sudden changes, unlike the multistep input function, the predicted airflow from Ridge, Ridge-M, and ALASSO is nearly identical. However, there are still some differences in RMSE values over the entire dataset, as presented in Table

5.

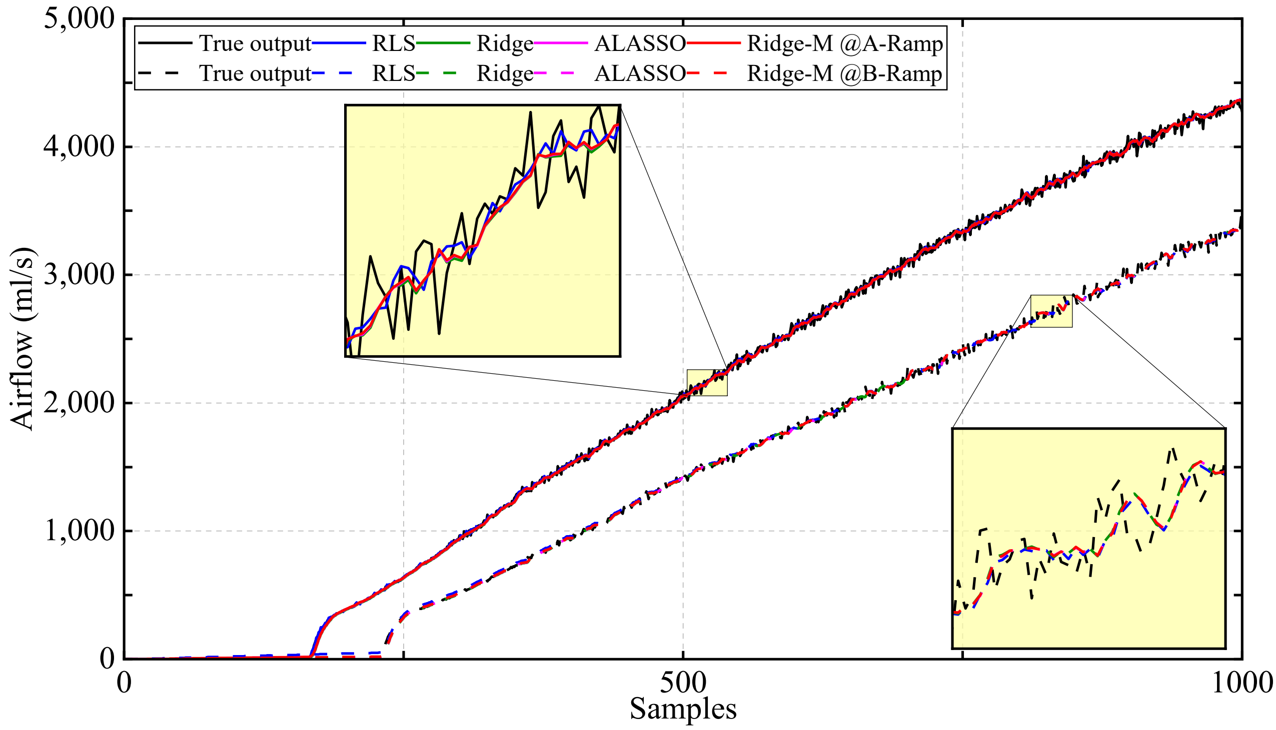

Figure 7. Estimated outputs from different identification algorithms using model coefficients obtained from training data corresponding to the A-Ramp input function.

Table 5. Performance comparison of RLS, Ridge, ALASSO, and Ridge-M using training data from the A-Ramp input function for the ARMA (5,1) model.

Input function	RLS	Ridge	ALASSO	Ridge-M
A-Multistep	40.4569	39.3918	38.325	38.0383
B-Multistep	39.7511	38.3543	37.298	37.3048
A-Ramp	30.3365	28.1481	28.120	27.9784
B-Ramp	29.5502	25.1616	25.295	25.2943
Average	35.0237	32.7639	32.259	32.1539

Between the Ridge and Ridge-M models, it is evident that Ridge performs best for ramp functions, while Ridge-M achieves the lowest RMSE for multistep functions, indicating its advantage in handling sudden changes due to momentum. Notably, Ridge-M outperforms Ridge on the training ramp (A-Ramp) due to momentum-driven smoothing, but it generalizes worse to the unseen B-Ramp. This suggests that while momentum helps reduce oscillations and fit gradual trends, it may lead to overfitting and slower adaptation to different ramp slopes. Ridge, without momentum, updates coefficients more directly, making it more adaptable to new ramp dynamics. If the test conditions closely match the training data, Ridge-M is beneficial; otherwise, Ridge provides better generalization. Overall, Ridge-M demonstrates the lowest average RMSE (32.1539), establishing itself as the most consistent model across different input functions. Ridge follows closely with an RMSE of 32.7639, suggesting that momentum provides only marginal improvement. RLS, with the highest RMSE

(35.0237), further confirms its inferior performance in both scenarios. Similarly, ALASSO also exhibits lower performance compared to Ridge-M when the training data is based on ramp input functions. However, the difference is not significant, and its performance remains highly competitive with average RMSE of 32.259.

4. Conclusions

This paper introduces a novel identification algorithm for ARMA models, using a ventilation pump as a representative case study to demonstrate its effectiveness. This algorithm employs ridge regression with momentum (Ridge-M), addressing the challenge of identifying model coefficients and structure from data with unknown orders p, q and input time delay d. A key contribution is the integration of grid search joint optimization for model order selection simultaneously, which effectively optimizes discrete parameters, ensuring robust structure identification.

The experimental study demonstrates the effectiveness of the proposed Ridge-M algorithm compared to standard ridge regression, adaptive LASSO, and recursive least squares across different input scenarios, including ramp and multistep functions. Results show that Ridge-M outperforms both methods in handling sudden changes, achieving the lowest RMSE for multistep inputs by leveraging momentum for smoother updates. However, for ramp inputs, standard Ridge performs slightly better, as momentum may slow adaptation to gradual trends. Overall, this work advances ARMA system identification by introducing a Ridge-M-based approach with structured model selection, offering a practical and efficient solution for data-driven modeling in control and forecasting applications.

Beyond system identification, this work paves the way for designing discrete model-based controllers based on the identified models. Additionally, the development of a digital twin using Ridge-M is proposed to enhance system accuracy and prediction reliability. These future directions aim to further leverage the identified model for control, simulation, and real-time monitoring applications, solidifying the practical impact of the proposed approach. Moreover, in the broader healthcare context, the proposed modeling forming also holds potential for integration into various types of mechanical ventilators, enabling personalized respiratory support tailored to patient-specific conditions and advancing the development of adaptive, patient-centric medical technologies.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under grant number B2023-20-01. We acknowledge the support of time and facilities from Key Laboratory of Digital Control and System Engineering (DCSELab), Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for this study.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. D. Graupe, D. J. Krause, J. B. Moore, J. B. Moore, Identification of autoregressive Moving-average parameters of time series, *IEEE Transact Automat Control*, **20** (1975), 104–107. https://doi.org/10.1109/TAC.1975.1100855
2. H. Djouadi, K. Ouari, Y. Belkhier, H. Lehouche, M. Bajaj, V. Blazek, Improved robust model predictive control for PMSM using backstepping control and incorporating integral action with experimental validation, *Results Eng.*, **23** (2024), 102416. https://doi.org/10.1016/j.rineng.2024.102416
3. N. M. Alyazidi, A. F. Bawazir, A. S. Al-Dogail, Robust integral sliding mode control for pressure management in multi-phase flow systems Discrete-time control Integral sliding mode control (ISMC) Multi-phase flow systems (MPFS) Pressure regulation Hammerstein model Nonlinear dynamics System identification, *Results Eng.*, **25** (2024), 104024. https://doi.org/10.1016/j.rineng.2025.104024
4. W. Fan, J. Xiong, Successive over relaxation for model-free LQR control of discrete-time Markov jump systems, *Automatica*, **171** (2024), 111919. https://doi.org/10.1016/j.automatica.2024.111919
5. J. Xu, M. Wang, G. Zhang, Trajectory tracking control of an underactuated unmanned underwater vehicle synchronously following mother submarine without velocity measurement, *Adv. Mechan. Eng.*, **7** (2015), 1–11. https://doi.org/10.1177/1687814015595340
6. Q. D. Le, Q. M. Lam, M. N. Huynh, H. H. Nguyen, V. T. Duong, Hardware-in-the-Loop simulation for online identification of Lithium-ion battery model parameters and state of charge estimation, *Results Eng.*, **25** (2025), 104509. https://doi.org/10.1016/j.rineng.2025.104509
7. S. Ekinci, D. Izci, Enhancing IIR system identification: Harnessing the synergy of gazelle optimization and simulated annealing algorithms, *e-Prime – Adv. Electr. Eng. Electron. Energy*, **5** (2023), 100225. https://doi.org/10.1016/j.prime.2023.100225
8. S. K V, D. G, S. M. Sulthan, Adaptive Particle Swarm Optimization based improved modeling of Solar Photovoltaic module for parameter determination, *e-Prime – Adv. Electr. Eng. Electron. Energy*, **8** (2024), 100621. https://doi.org/10.1016/j.prime.2024.100621
9. A. Srivastava, An application of kho-Kho optimization algorithm to estimation parameter of permanent magnet synchronous machine, *e-Prime – Adv. Electr. Eng. Electron. Energy*, **6** (2023), 100309. https://doi.org/10.1016/J.PRIME.2023.100309
10. B. P. Adedeji, Energy parameter modeling in plug-in hybrid electric vehicles using supervised machine learning approaches, *e-Prime – Adv. Electr. Eng. Electron. Energy*, **8** (2024), 100584. https://doi.org/10.1016/J.PRIME.2024.100584
11. Y. Tang, H. Ding, X. Shan, X. Wang, Application of the novel three-stage DEA model to evaluate total-factor energy efficiency: A case study based on 30 provinces and 8 comprehensive economic zones of China, *Results Eng.*, **20** (2023), 101417. https://doi.org/10.1016/j.rineng.2023.101417
12. M. Rayati, R. Nasiri, and M. N. Ahmadabadi, Improving muscle force distribution model using reflex excitation: Toward a model-based exoskeleton torque optimization approach, *IEEE Transact Neural Syst. Rehab. Eng.*, **31** (2023), 720–728. https://doi.org/10.1109/TNSRE.2022.3230795
13. C. Remlinger, C. Alasseur, M. Brière, J. Mikael, Expert aggregation for financial forecasting, *J. Finan. Data Sci.*, **9** (2023), 1–21. https://doi.org/10.1016/j.jfds.2023.100108
14. N. S. Gehlot, P. J. Alsina, A discrete model of induction motors for real-time control applications, *IEEE Transactions on Industrial Electronics*, **40** (1993), no. 3, 317–325.

https://doi.org/10.1109/41.232211
15. L. D. Paarmann and M. J. Korenberg, Estimation of the parameters of an ARMA signal model based on an orthogonal search, *IEEE Transact. Automat. Control*, **37** (1992), 347–352. https://doi.org/10.1109/9.119634
16. F. Ding, Y. Shi, T. Chen, Performance analysis of estimation algorithms of nonstationary ARMA processes, *IEEE Transact. Signal Process.*, **54** (2006), 1041–1053. https://doi.org/10.1109/TSP.2005.862845
17. S. U. Pillai, Theodore I. Shim, D. C. Youla, A new technique for ARMA-system identification and rational approximation, *IEEE Transact. Signal Process.*, **41** (1993), 1281–1304. https://doi.org/10.1109/78.205730
18. S. Li, Y. Zhu, B. W. Dickinson, A comparison of two linear methods of estimating the parameters of ARMA models, *IEEE Transact. Automat. Control*, **34** (1989), 915–917. https://doi.org/10.1109/9.29444
19. J. K. Sabiti, A fast estimation method for ARMA processes, *Automatica*, **32** (1996), 235–239. https://doi.org/10.1016/0005-1098(96)85553-1
20. T. Wang, X. Li, J. Guo, Y. Zhao, Identification of ARMA models with binary-valued observations, *Automatica*, **149** (2023), 110832. https://doi.org/10.1016/j.automatica.2022.110832
21. B. R. Pramod, A. Regressive, S. C. Bose, System identification using ARMA modeling and neural networks, **115** (1993), 487–491. https://doi.org/10.1115/1.2901794
22. S. Lu, K. H. Chon, Nonlinear autoregressive and nonlinear autoregressive moving average model parameter estimation by minimizing hypersurface distance, *IEEE Transact. Signal Process.*, **51** (2003), 3020–3026. https://doi.org/10.1109/TSP.2003.818999
23. D. Marelli, K. You, M. Fu, Identification of ARMA models using intermittent and quantized output observations, *Automatica*, **49** (2013), 360–369. https://doi.org/10.1016/j.automatica.2012.11.020
24. C. Yu, K. You, L. Xie, Quantized identification of ARMA systems with colored measurement noise, *Automatica*, **66** (2016), 101–108. https://doi.org/10.1016/j.automatica.2015.12.013
25. F. Ding, L. Xu, F. E. Alsaadi, T. Hayat, Iterative parameter identification for pseudo-linear systems with ARMA noise using the filtering technique, *IET Control Theory Appl.*, **12** (2018), 892–899. https://doi.org/10.1049/iet-cta.2017.0821
26. Y. Wang, W. Zhang, M. Fan, Q. Ge, B. Qiao, X. Zuo, Regression with adaptive lasso and correlation based penalty, *Appl. Math. Model.*, **105** (2022), 179–196. https://doi.org/10.1016/j.apm.2021.12.016
27. S. Basu, G. Michailidis, Regularized estimation in sparse high-dimensional time series models, *Ann. Statist.*, **43** (2015), 1535–1567. https://doi.org/10.1214/15-AOS1315
28. S. S. Mannelli, P. Urbani, Analytical study of momentum-based acceleration methods in paradigmatic high-dimensional non-convex problems, *Adv. Neural Inform. Process. Syst.*, **34** (2021), 187–199. https://hal.science/hal-04425461v1
29. G. E. P. Box, G. M. Jenkins, Some recent advances in forecasting and control, *J. Royal Statist. Soc. Series C Appl. Statist.*, **17** (1968), 91–109. https://doi.org/10.2307/2985674
30. Y. Shigeto, I. Suzuki, K. Hara, M. Shimbo, Y. Matsumoto, Ridge regression, hubness, and zero-shot learning, *Mach. Learn. Knowl. Discovery Databases*, **9284** (2015), 135–151. https://doi.org/10.1007/978-3-319-23528-8_9
31. R. T. Nakatsu, Validation of machine learning ridge regression models using Monte Carlo, bootstrap, and variations in cross-validation, *J. Intell. Syst.*, **32** (2023), 20220224.

https://doi.org/10.1515/jisys-2022-0224
32. P. M. T. Broersen, S. de Waele, Finite sample properties of ARMA order selection, *IEEE Transact. Instrument. Measurement*, **53** (2004), 645–651. https://doi.org/10.1109/TIM.2004.827058
33. B. Yüzbaşı, M. Arashi, S. Ejaz Ahmed, Shrinkage estimation strategies in generalised Ridge regression Models: Low/High-dimension regime, *Int. Statist. Rev.*, **88** (2020), 229–251. https://doi.org/10.1111/insr.12351
34. J. A. Cadzow, ARMA Time Series Modeling: An effective method, *IEEE Transact. Aerospace Electron. Syst.*, **AES-19** (1983), 49–58. https://doi.org/10.1109/TAES.1983.309419
35. J. A. Cadzow, ARMA Modeling of Time Series, *IEEE Transact. Pattern Anal. Mach. Intell.*, **PAMI-4** (1982), 124–128. https://doi.org/10.1109/TPAMI.1982.4767216
36. A. Godichon-Baggioni, W. Lu, B. Portier, Recursive ridge regression using second-order stochastic algorithms, *Comput. Statist. Data Anal.*, **190** (2024), 107854. https://doi.org/10.1016/j.csda.2023.107854
37. P. Goulet Coulombe, Time-varying parameters as ridge regressions, *Int. J. Forecast.*, **41** (2025), 982–1002. https://doi.org/10.1016/j.ijforecast.2024.08.006
38. G. Maroni, L. Cannelli, D. Piga, Gradient-based bilevel optimization for multi-penalty Ridge regression through matrix differential calculus, *European J. Control*, **81** (2023), 101150. https://doi.org/10.1016/j.ejcon.2024.101150
39. W. Jamil, A. Bouchachia, Iterative ridge regression using the aggregating algorithm, *Pattern Recogn. Letters*, **158** (2022), 34–41. https://doi.org/10.1016/j.patrec.2022.04.021
40. H. Gao, R. Diao, Z. Huang, Y. Zhong, Y. Mao, W. Tang, Parameter identification of SVG using multilayer Coarse-to-fine grid searching and particle swarm optimization, *IEEE Access*, **10** (2022), 77137–77146. https://doi.org/10.1109/ACCESS.2022.3192538
41. S. W. Mahmood, G. T. Basheer, Z. Y. Algamal, Improving kernel ridge regression for medical data classification based on meta-heuristic algorithms, *Kuwait J. Sci.*, **52** (2025), 100408. https://doi.org/10.1016/j.kjs.2025.100408
42. M. Mahadi, T. Ballal, M. Moinuddin, U. M. Al-Saggaf, A recursive least-squares with a time-varying regularization parameter, *Appl. Sci.,* **12** (2022), 2077. https://doi.org/10.3390/APP12042077
43. C. Illg, T. Kösters, S. Kiroriwal, O. Nelles, Adaptive system identification with regularized FIR Models, *IFAC-PapersOnLine*, **55** (2022), 1–6. https://doi.org/10.1016/J.IFACOL.2022.07.279
44. M. Neumann, R. J. Plemmons, Convergent nonnegative matrices and iterative methods for consistent linear systems, *Numer. Math.*, **31** (1978), 265–279. https://doi.org/10.1007/BF01397879
45. V. I. Hasanov, S. A. Hakkaev, Convergence analysis of some iterative methods for a nonlinear matrix equation, *Comput. Math. Appl.*, **72** (2016), 1164–1176. https://doi.org/10.1016/j.camwa.2016.06.035
46. A. S. Householder, On the Convergence of Matrix Iterations, *JACM*, **3** (1956), 314–324. https://doi.org/10.1145/320843.320851