

PAPER

LASSO-Based Identification of Sparse Autoregressive Exogenous Input Model Coefficients: Application to Automated AMBU-Bag Resuscitator System

Trung Dat Phan^{1,2} ,
 Cong Toai Truong^{1,2},
 Huy Hung Nguyen³,
 Tan Tien Nguyen^{1,2},
 Van Tu Duong^{1,2}  

¹Key Laboratory of Digital Control and System Engineering (DCSELab), Faculty of Mechanical Engineering, Ho Chi Minh University of Technology (HCMUT), Ho Chi Minh City, Vietnam

²Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam

³Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology and Engineering, Ho Chi Minh City, Vietnam

dvtu@hcmut.edu.vn

ABSTRACT

Accurate system identification is important role in the development of reliable control approaches for automated respiratory devices. Nonetheless, conventional identification approaches suffer from overfitting and sensitivity to noise. Therefore, in this paper, a novel sparse autoregressive exogenous input identification scheme based on the least absolute shrinkage and selection operator is proposed for the automated AMBU-bag resuscitator (AABR). Specifically, it incorporates decoupled regularization terms for autoregressive and input-related components, as well as enables robust parameter estimation without requiring normalization of input data. The grey wolf optimizer (GWO) is used to optimize regularization parameters by minimizing prediction errors on validation datasets. The experimental data were obtained from a real AABR system with three excitation signals: cosine, parabolic, and ramp, corresponding to smooth periodic, second-order polynomial, and linear profiles, respectively. The smooth periodic profile dataset was used for training, while the others were employed for validation. As a result, the proposed model achieved consistent prediction accuracy, with the root mean square errors (RMSEs) of 0.177893, 0.195727, and 0.150998. Conversely, the conventional recursive least squares (RLS) method exhibited severe overfitting, with RMSE increasing from 0.009821 to 11.436522.

KEYWORDS

autoregressive exogenous input (ARX) model, least absolute shrinkage and selection operator (LASSO) regression, model identification, sparse estimation, AMBU-bag resuscitator (ABR), respiratory mechanics

1 INTRODUCTION

Automated respiratory support devices have gained significant importance in contemporary healthcare infrastructure, especially for emergency ventilation

Phan, T. D., Truong, C. T., Nguyen, H. H., Nguyen, T. T., Duong, V. T. (2026). LASSO-Based Identification of Sparse Autoregressive Exogenous Input Model Coefficients: Application to Automated AMBU-Bag Resuscitator System. *International Journal of Online and Biomedical Engineering (iJOE)*, 22(10), pp. 90–108. <https://doi.org/10.3991/ijoe.v22i10.61627>

Article submitted 2026-03-20. Revision uploaded 2026-05-30. Final acceptance 2026-05-31.

© 2026 by the authors of this article. Published under CC-BY.

and portable life support [1]. Automated AMBU-bag resuscitators (AABRs) [2] have attracted significant interest in recent years as a cost-effective alternative to traditional mechanical ventilators for ventilatory support [3–4]. In AABR devices, the air flow delivered to the patient is generated using a mechanical compression system driven by an electromechanical actuator. The system's dynamics are determined by the combined effects of the actuator's movement, air flow resistance, and the lungs' elastic properties [5]. Therefore, precise mathematical modeling of the ventilatory system is required for controller design, trajectory tracking, and system safety analysis.

Several system identification techniques have been widely used to obtain a mathematical model for a dynamic system based on input/output measurements obtained from a real system [6–9]. Among these techniques, linear parametric models such as the autoregressive with exogenous input (ARX) model have been commonly used due to their simplicity and suitability for discrete-time control systems. The most commonly used parameter estimation techniques for system identification based on the ARX model include least squares (LS) and recursive least squares (RLS). These techniques have been widely used in various fields for system identification in engineering systems [10–11]. However, in the case of electromechanical ventilation systems, it has been observed that these techniques may lead to overfitting and be sensitive to measurement noise when redundant regressors are present in the model. Dong et al. (2025) [12] demonstrated that LS/RLS methods in ARX models are prone to overfitting when the number of regressor is large and are highly sensitive to measurement noise. Chen and Li (2025) [13] showed that conventional LS/RLS methods in ARX identification are vulnerable to noise and redundant regressors, making RLS unstable and biased in complex electromechanical systems. This clearly illustrates the limitations of RLS in complex electromechanical systems.

In order to alleviate these problems, sparse regression methods have been increasingly studied in system identification research. Among them, the most popular method is the least absolute shrinkage and selection operator (LASSO), in which an $\mathcal{L}_1$ -norm regularization term is added to the objective function to realize sparse solutions for system identification problems [14]. Many studies have shown the effectiveness of sparse system identification in control systems, signal processing, and machine learning [15–16]. Although sparse system identification methods have shown their superiority in system identification problems, the performance of LASSO is heavily affected by the choice of regularization parameters. In particular, Gillariose et al. (2025) [17] emphasized that while LASSO provides effective sparse solutions, it suffers from key drawbacks. Its performance strongly depends on the choice of regularization parameters, often leading to bias and instability when predictors are highly correlated. Moreover, parameter selection via cross-validation increases computational cost, limiting its applicability in complex systems. Indeed, in most existing studies, the parameters are manually set or chosen through cross-validation methods, which can lead to suboptimal solutions and increased computational cost.

Recently, a novel application of metaheuristic optimization algorithms has been proposed as a promising approach to automatically optimize hyperparameters in machine learning and system identification problems [18]. Some optimization algorithms, such as particle swarm optimization (PSO), genetic algorithms (GA), and grey wolf optimizer (GWO), have been successfully implemented in solving optimization problems for model parameters in complex search spaces [19]. Among these

optimization techniques, GWO has received significant attention due to its simplicity, strong global search capability, and rapid convergence. Pham and Nguyen [20] applied GWO combined with ANN to optimize HVAC system operations, demonstrating improved energy efficiency. Nonetheless, the application of metaheuristic optimization algorithms in conjunction with sparse identification in automated ventilation system modeling is still in its infancy. Nevertheless, there are still some important research gaps in the field of sparse system identification. First, most studies have been performed for generic control systems or signal processing problems, whereas the AABR system is rarely studied using sparse identification approaches. Secondly, the majority of traditional identification methods based on LASSO depend on the normalized data and set a single regularization parameter for all coefficients of the system. Such penalty results in unbalanced shrinkage effect when applying it to the raw physical system. In control system, the scale and physical unit of the exogenous input is greatly different from the one of the observed output and applying a single penalty on the un-normalized data lead to extremely unbalanced shrinkage effect as coefficients related to small scale variables be largely shrunk while other variables are less sparsely penalized. Researchers then usually normalize the data before estimation, but normalization can alter the physical scale of the system and the resulting model is not directly physically interpreted. Lastly, there is a lack of experimental verification of sparse system identification algorithms applied to ventilators with various excitation inputs.

To address these limitations, this paper proposes a LASSO-based ARX identification framework for modeling an AABR system. The proposed approach uses decoupled regularization parameters for the autoregressive and exogenous input terms. This allows for effective sparse estimation without the need for data normalization. The GWO is used for the optimization of the hyperparameters by minimizing the prediction errors for the validation data. The proposed approach is experimentally tested using a physical AABR system with different excitation signals, which comprise cosine, parabolic, and ramp. The main contributions of this work can be summarized as follows:

- Development of a LASSO-based ARX identification framework for modeling AABR systems.
- Automatic hyperparameters tuning using the GWO method to improve identification performance.
- Introduction of decoupled input–output regularization terms to enable robust estimation with unnormalized experimental data.
- Experimental validation on a real AABR platform using multiple excitation trajectories.

2 METHODOLOGY

2.1 Problem formulation

In a standard autoregressive moving average (ARMA) structure, the autoregressive component links the current system state directly to its past outputs. Meanwhile, the moving average component represents the effects of past random disturbances or system noise. As a specific extension of this concept, the ARX model integrates external control actions. It mathematically describes the linear relationship between

the current system output and past input-output signals. The scalar representation of the ARX model is given by:

$$y(k) = \sum_{i=1}^n a_i^* y(k-i) + \sum_{j=1}^m b_j^* u(k-j) + \epsilon(k), \quad (1)$$

where $n \in \mathbb{N}^*$ and $m \in \mathbb{N}^*$ are the orders of the autoregressive (AR) and exogenous input (X) parts, respectively; $y(k) \in \mathbb{R}$ and $y(k-i) \in \mathbb{R}$ are the measured outputs of the model at time instants k and $k-i$; $u(k-j) \in \mathbb{R}$ represents the control input at time instant $k-j$; $a_i^*, b_j^* \in \mathbb{R}$ are the coefficients of the autoregressive part and exogenous input parts, respectively; and $\epsilon(k) \sim \mathcal{N}(0, \sigma^2)$ denotes white noise with zero mean.

In practice, the true orders and coefficients of the model (1) are generally unknown. Therefore, a prediction model is constructed to estimate the system output using historical input-output data. The parameters of this model are estimated using the LASSO method to mitigate overfitting and improve robustness in the presence of noise. The prediction model is expressed as:

$$\hat{y}(k) = \sum_{i=1}^{n^*} \hat{a}_i y(k-i) + \sum_{j=1}^{m^*} \hat{b}_j u(k-j), \quad (2)$$

where $\hat{a}_i \in \mathbb{R}$ and $\hat{b}_j \in \mathbb{R}$ are the estimated coefficients of a_i and b_j , respectively; $n^* \in \mathbb{N}^*$ and $m^* \in \mathbb{N}^*$ are the estimated orders of the autoregressive part and the exogenous input part, respectively and $n^* \geq m^*$; $\hat{y}(k)$ is the estimated output at the time k .

For compact representation, the parameter vectors and regressor vectors are defined as:

$$\hat{\mathbf{a}} := [\hat{a}_1 \quad \hat{a}_2 \quad \cdots \quad \hat{a}_{n^*}]^T \in \mathbb{R}^{n^*}, \quad (3)$$

$$\hat{\mathbf{b}} := [\hat{b}_1 \quad \hat{b}_2 \quad \cdots \quad \hat{b}_{m^*}]^T \in \mathbb{R}^{m^*}, \quad (4)$$

$$\mathbf{x}(k) = [y(k) \quad y(k-1) \quad \cdots \quad y(k-n^*+1)] \in \mathbb{R}^{1 \times n^*}, \quad (5)$$

$$\mathbf{g}(k) = [u(k) \quad u(k-1) \quad \cdots \quad u(k-m^*+1)] \in \mathbb{R}^{1 \times m^*}. \quad (6)$$

From Eqs. (3)–(6), the prediction model in (2) can be reformulated into a compact linear regression form, which serves as the basis for the subsequent sparse estimation procedure:

$$\hat{y}(k) = \mathbf{x}(k-1)\hat{\mathbf{a}} + \mathbf{g}(k-1)\hat{\mathbf{b}}. \quad (7)$$

Considering N observed samples, the stacked regression matrices $\mathbf{X}$ and $\mathbf{U}$ are defined as:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}(n^*) \\ \mathbf{x}(n^*+1) \\ \vdots \\ \mathbf{x}(N-1) \end{bmatrix} = \begin{bmatrix} y(n^*) & y(n^*-1) & \cdots & y(1) \\ y(n^*+1) & y(n^*) & \cdots & y(2) \\ \vdots & \vdots & \ddots & \vdots \\ y(N-1) & y(N-2) & \cdots & y(N-n^*) \end{bmatrix} \in \mathbb{R}^{(N-n^*) \times n^*}, \quad (8)$$

$$\mathbf{U} = \begin{bmatrix} \mathbf{g}(n^*) \\ \mathbf{g}(n^* + 1) \\ \vdots \\ \mathbf{g}(N - 1) \end{bmatrix} = \begin{bmatrix} u(n^*) & u(n^* - 1) & \cdots & u(n^* - m^* + 1) \\ u(n^* + 1) & u(n^*) & \cdots & u(n^* - m^* + 2) \\ \vdots & \vdots & \ddots & \vdots \\ u(N - 1) & u(N - 2) & \cdots & u(N - m^*) \end{bmatrix} \in \mathbb{R}^{(N-n^*) \times m^*}. \quad (9)$$

Consequently, the predicted output vector $\hat{\mathbf{Y}}$ for the entire dataset is reformulated as:

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\mathbf{a}} + \mathbf{U}\hat{\mathbf{b}}, \quad (10)$$

where $\hat{\mathbf{Y}} = [\hat{y}(n^* + 1) \ \hat{y}(n^* + 2) \ \cdots \ \hat{y}(N)] \in \mathbb{R}^{N-n^*}$. Similarly, the actual output vector is likewise defined as $\mathbf{Y} = [y(n^* + 1) \ y(n^* + 2) \ \cdots \ y(N)] \in \mathbb{R}^{N-n^*}$.

2.2 Model coefficients estimation with LASSO regularization

In this study, the parameter estimation is formulated using a LASSO-based objective function that combines the residual sum of squares (RSS) with $\mathcal{L}_1$ -norm regularization:

$$\mathcal{J}(\hat{\mathbf{a}}, \hat{\mathbf{b}}, \lambda_a, \lambda_b) = \underbrace{\frac{1}{2} \|\mathbf{Y} - \mathbf{X}\hat{\mathbf{a}} - \mathbf{U}\hat{\mathbf{b}}\|_2^2}_{\mathcal{J}_r} + \underbrace{\lambda_a \|\hat{\mathbf{a}}\|_1}_{\mathcal{J}_a} + \underbrace{\lambda_b \|\hat{\mathbf{b}}\|_1}_{\mathcal{J}_b}, \quad (11)$$

where $\lambda_a > 0$ and $\lambda_b > 0$ are positive regularization terms governing the sparsity imposed on the model coefficients. $\mathcal{J}_r$ encapsulates the RSS and enforces the least-squares regression criterion, while the $\mathcal{L}_1$ -regularization terms $\mathcal{J}_a$ and $\mathcal{J}_b$ promote sparsity in the estimated parameters.

By invoking the extreme value theorem to perform coordinate-wise minimization of the LASSO objective function, one arrives at the following first-order optimality conditions:

$$\frac{\partial \mathcal{J}}{\partial \hat{a}_i} = \frac{\partial \mathcal{J}_r}{\partial \hat{a}_i} + \frac{\partial \mathcal{J}_a}{\partial \hat{a}_i}, \quad (12)$$

$$\frac{\partial \mathcal{J}}{\partial \hat{b}_j} = \frac{\partial \mathcal{J}_r}{\partial \hat{b}_j} + \frac{\partial \mathcal{J}_b}{\partial \hat{b}_j}. \quad (13)$$

By decomposing the residual to isolate the specific term $\hat{a}_i$ and $\hat{b}_j$, the partial derivatives of $\mathcal{J}_r$ are expressed as:

$$\frac{\partial \mathcal{J}_r}{\partial \hat{a}_i} = -\mathbf{X}(:, i)^T (\mathbf{Y} - \mathbf{X}\hat{\mathbf{a}} - \mathbf{U}\hat{\mathbf{b}} + \mathbf{X}(:, i)\hat{a}_i) + \mathbf{X}(:, i)^T \mathbf{X}(:, i)\hat{a}_i, \quad (14)$$

$$\frac{\partial \mathcal{J}_r}{\partial \hat{b}_j} = -\mathbf{U}(:, j)^T (\mathbf{Y} - \mathbf{X}\hat{\mathbf{a}} - \mathbf{U}\hat{\mathbf{b}} + \mathbf{U}(:, j)\hat{b}_j) + \mathbf{U}(:, j)^T \mathbf{U}(:, j)\hat{b}_j. \quad (15)$$

Taking partial derivative of the weighted $\mathcal{L}_1$ -regularization terms, one can obtain that

$$\frac{\partial \mathcal{J}_a}{\partial \hat{a}_i} = \lambda_a \frac{\partial}{\partial \hat{a}_i} (|\hat{a}_i|), \quad (16)$$

$$\frac{\partial \mathcal{J}_b}{\partial \hat{b}_j} = \lambda_b \frac{\partial}{\partial \hat{b}_j} (|\hat{b}_j|). \quad (17)$$

Since the function $f(x) = |x|$ is non-differentiable at $x = 0$, its gradient is undefined in the classical sense. By employing the concept of subdifferential calculus, the coordinate-wise partial derivatives of the weighted $\mathcal{L}_1$ -regularization terms can be formally expressed as:

$$\lambda_a \frac{\partial}{\partial \hat{a}_i} (|\hat{a}_i|) = \begin{cases} -\lambda_a & \text{if } \hat{a}_i < 0, \\ [-\lambda_a, \lambda_a] & \text{if } \hat{a}_i = 0, \\ \lambda_a & \text{if } \hat{a}_i > 0, \end{cases} \quad (18)$$

$$\lambda_b \frac{\partial}{\partial \hat{b}_j} (|\hat{b}_j|) = \begin{cases} -\lambda_b & \text{if } \hat{b}_j < 0, \\ [-\lambda_b, \lambda_b] & \text{if } \hat{b}_j = 0, \\ \lambda_b & \text{if } \hat{b}_j > 0. \end{cases} \quad (19)$$

These formulations rigorously enforce sparsity within the LASSO framework by enabling the estimated coefficients $\hat{a}_i$ and $\hat{b}_j$ to be driven exactly to zero whenever their corresponding gradient values reside within the subdifferential interval at the origin.

By substituting Eqs. (14)–(17) into Eqs. (12)–(13), and subsequently setting the resulting expressions to zero, one can obtain:

$$-\mathbf{X}(:, i)^T (\mathbf{Y} - \mathbf{X}\hat{\mathbf{a}} - \mathbf{U}\hat{\mathbf{b}} + \mathbf{X}(:, i)\hat{a}_i) + \mathbf{X}(:, i)^T \mathbf{X}(:, i)\hat{a}_i + \lambda_a \frac{\partial}{\partial \hat{a}_i} (|\hat{a}_i|) = 0, \quad (20)$$

$$-\mathbf{U}(:, j)^T (\mathbf{Y} - \mathbf{X}\hat{\mathbf{a}} - \mathbf{U}\hat{\mathbf{b}} + \mathbf{U}(:, j)\hat{b}_j) + \mathbf{U}(:, j)^T \mathbf{U}(:, j)\hat{b}_j + \lambda_b \frac{\partial}{\partial \hat{b}_j} (|\hat{b}_j|) = 0. \quad (21)$$

Solving equations (20) and (21) respectively for $\hat{a}_i$ and $\hat{b}_j$, one obtains the update rules for the model coefficients:

$$\hat{a}_i = \frac{\gamma_a(i) - \lambda_a \frac{\partial}{\partial \hat{a}_i} (|\hat{a}_i|)}{\beta_a(i)}, \quad (22)$$

$$\hat{b}_j = \frac{\gamma_b(j) - \lambda_b \frac{\partial}{\partial \hat{b}_j} (|\hat{b}_j|)}{\beta_b(j)}, \quad (23)$$

where

$$\gamma_a(i) \triangleq \mathbf{X}(:, i)^T (\mathbf{Y} - \mathbf{X}\hat{\mathbf{a}} - \mathbf{U}\hat{\mathbf{b}} + \mathbf{X}(:, i)\hat{a}_i), \quad (24)$$

$$\gamma_b(j) \triangleq \mathbf{U}(:, j)^T (\mathbf{Y} - \mathbf{X}\hat{\mathbf{a}} - \mathbf{U}\hat{\mathbf{b}} + \mathbf{U}(:, j)\hat{b}_j), \quad (25)$$

$$\beta_a(i) \triangleq \mathbf{X}(:, i)^T \mathbf{X}(:, i), \quad (26)$$

$$\beta_b(j) \triangleq \mathbf{U}(:, j)^T \mathbf{U}(:, j). \quad (27)$$

Using Eqs. (18)–(19), the closed-form parameter update laws given in (22) and (23) can be explicitly reformulated into the following piecewise-defined expressions:

$$\hat{a}_i = \begin{cases} \frac{\gamma_a(i) + \lambda_a}{\beta_a(i)} & \text{if } \gamma_a(i) < -\lambda_a, \\ 0 & \text{if } -\lambda_a \leq \gamma_a(i) \leq \lambda_a, \\ \frac{\gamma_a(i) - \lambda_a}{\beta_a(i)} & \text{if } \gamma_a(i) > \lambda_a, \end{cases} \quad (28)$$

$$\hat{b}_j = \begin{cases} \frac{\gamma_b(j) + \lambda_b}{\beta_b(j)} & \text{if } \gamma_b(j) < -\lambda_b, \\ 0 & \text{if } -\lambda_b \leq \gamma_b(j) \leq \lambda_b, \\ \frac{\gamma_b(j) - \lambda_b}{\beta_b(j)} & \text{if } \gamma_b(j) > \lambda_b. \end{cases} \quad (29)$$

By introducing the soft-thresholding operator $S(a, b)$, a nonlinear shrinkage function widely employed in sparse estimation, for nonnegative arguments $a, b \geq 0$, and defined explicitly as

$$S(a, b) = \text{sign}(a) \max(|a| - b, 0) \quad (30)$$

one can reformulate the coordinate-wise update equations (28) and (29) into a compact and computationally elegant representation. Specifically, the estimated model coefficients take the form:

$$\hat{a}_i = \frac{S(\gamma_a(i), \lambda_a)}{\beta_a(i)}, \quad (31)$$

$$\hat{b}_j = \frac{S(\gamma_b(j), \lambda_b)}{\beta_b(j)}. \quad (32)$$

The procedural workflow for ARX model identification leveraging the LASSO regularization framework in conjunction with the coordinate descent optimization algorithm is systematically summarized and presented in Algorithm 1 in the form of a pseudo-code representation.

Algorithm 1: ARX Model Identification using LASSO and Coordinate Descent Algorithms

Step 1. Acquire the input-output data sequences $y(k)$ and $u(k)$ for $k = 1, 2, \dots, N$, and specify the convergence tolerance δ .

Step 2. Specify the maximum candidate orders n^* and m^* , the regularization parameters λ_a and λ_b , the threshold δ . Randomly assign initial values to the model coefficients.

Step 3. While $\|\hat{\mathbf{a}}_\tau^T - \hat{\mathbf{a}}_{\tau-1}^T\|_\infty$ or $\|\hat{\mathbf{b}}_\tau^T - \hat{\mathbf{b}}_{\tau-1}^T\|_\infty > \delta$:
 At each position i :
Step 3.1. Calculate $\gamma_a(i)$, $\gamma_b(j)$, $\beta_a(i)$, $\beta_b(j)$ for each individual coordinate using (24)–(27);
Step 3.2. Update the AR coefficients $\hat{\mathbf{a}}_i$ and the X coefficients $\hat{\mathbf{b}}_j$ according to Eqs. (31), (32);

Step 4. Finish the process.

The convergence tolerance δ serves as a critical hyperparameter in the coordinate descent algorithm, dictating the fundamental trade-off between estimation accuracy and computational complexity. As defined in Step 4, the stopping criterion evaluates the infinity norm $\mathcal{L}_\infty$ of the coefficient updates, ensuring the algorithm only terminates when the maximum absolute change across all parameters is strictly bounded by δ . Selecting an excessively large δ induces premature convergence, halting the optimization before the AR and exogenous dynamics are fully captured. Conversely, a strictly infinitesimal δ dramatically amplifies the computational burden and may trap the algorithm in an endless loop due to floating-point precision limits or inherent measurement noise. Following the standard algorithmic configurations widely adopted in regularized machine learning libraries (e.g., the default tolerance in GLMNET [21]), the threshold in this study was empirically set to $\delta = 10^{-4}$.

2.3 Grey wolf optimizer for regularization terms selection

In practical scenarios, discrepancies in the physical units of measurement between the input and output variables inherently lead to differences in their numerical ranges. Such disparities can significantly influence the performance and convergence behavior of the algorithm, particularly if the regularization parameters λ_a and λ_b are chosen without appropriate scaling. Therefore, this study employs the GWO algorithm to determine the optimal values of λ_a and λ_b that minimize the LASSO loss function.

The GWO algorithm solves the optimization problem that satisfies the condition:

$$\boldsymbol{\lambda}^* = \underset{\boldsymbol{\lambda} \in \Psi}{\operatorname{argmin}} \mathcal{J}(\boldsymbol{\lambda}), \quad (33)$$

where $\boldsymbol{\lambda}^* \triangleq [\lambda_a^* \ \lambda_b^*] \in \mathbb{R}^{1 \times 2}$ denotes the optimal parameter vector; $\Psi = \{(\lambda_a, \lambda_b) \in \mathbb{R}^2 \mid 0 < \lambda_a \leq \lambda_a^u, 0 < \lambda_b \leq \lambda_b^u\}$; λ_a^u and λ_b^u are the upper bounds of the hyperparameters.

The GWO algorithm is inspired by the pack hunting behavior of a population of grey wolves consisting of P individuals, where each individual is indexed by $p \in \{1, 2, \dots, P\}$. These individuals represent a set of candidate design solutions $\boldsymbol{\lambda}^p(v) \triangleq [\lambda_a^p(v) \ \lambda_b^p(v)] \in \mathbb{R}^{1 \times 2}$ at iteration v . Among them, three leading individuals, referred to as the alpha wolf, beta wolf, and delta wolf, have position vectors denoted by $\boldsymbol{\lambda}^\alpha(v)$, $\boldsymbol{\lambda}^\beta(v)$, and $\boldsymbol{\lambda}^\delta(v)$, respectively, with corresponding objective function values $\mathcal{J}(\boldsymbol{\lambda}^\alpha(v))$, $\mathcal{J}(\boldsymbol{\lambda}^\beta(v))$ and $\mathcal{J}(\boldsymbol{\lambda}^\delta(v))$. These three individuals represent the best solutions

found within the population at each iteration and are characterized by the following inequality:

$$\begin{cases} \mathcal{J}(\lambda^1(v)) \leq \mathcal{J}(\lambda^2(v)) \leq \mathcal{J}(\lambda^3(v)) \leq \dots \leq \mathcal{J}(\lambda^P(v)) \\ \lambda^\alpha(v) := \lambda^1(v) \\ \lambda^\beta(v) := \lambda^2(v) \\ \lambda^\delta(v) := \lambda^3(v) \end{cases} \quad (34)$$

Arranging the set of vectors $\lambda^p(v)$ row-wise forms the population position matrix $\Lambda(v)$:

$$\Lambda(v) = \begin{bmatrix} \lambda^1(v) \\ \lambda^2(v) \\ \vdots \\ \lambda^P(v) \end{bmatrix} \in \mathbb{R}^{P \times 2}. \quad (35)$$

The three leading individuals govern the movement of the entire population during the search for the optimal solution. The remaining individuals update their search positions according to an encircling strategy based on the positions of the three leaders.

The estimated search positions of the entire population at iteration v , when guided by the alpha wolf, are characterized by the distance matrix $\mathbf{D}_\alpha(v)$ and the position matrix $\Lambda_1(v)$:

$$\mathbf{D}_\alpha(v) = |\mathbf{C}_1(v) \odot \lambda^\alpha(v) - \Lambda(v)| \in \mathbb{R}^{P \times 2}, \quad (36)$$

$$\Lambda_1(v) = \lambda^\alpha(v) - \mathbf{A}_1(v) \odot \mathbf{D}_\alpha(v) \in \mathbb{R}^{P \times 2}. \quad (37)$$

where

$$\mathbf{A}_1(v) = 2a\mathbf{R}_1(v) - a \cdot \mathbf{1}_{P \times 2} \in \mathbb{R}^{P \times 2}, \quad (38)$$

$$\mathbf{C}_1(v) = 2\mathbf{R}_2(v) \in \mathbb{R}^{P \times 2}. \quad (39)$$

The coefficient a decreases linearly from 2 to 0 over successive iterations; $\mathbf{1}_{P \times 2}$ denotes a $P \times 2$ matrix with all elements equal to one, and $\mathbf{R}_1(v)$, $\mathbf{R}_2(v) \in \mathbb{R}^{P \times 2} \sim \text{Uniform}(0, 1)$ are matrices containing random coefficients within the interval $[0, 1]$. Similarly, the estimated position matrices of the population when guided by the beta and delta individuals are defined as follows:

$$\mathbf{D}_\beta(v) = |\mathbf{C}_2(v) \odot \lambda^\beta(v) - \Lambda(v)| \in \mathbb{R}^{P \times 2}, \quad (40)$$

$$\Lambda_2(v) = \lambda^\beta(v) - \mathbf{A}_2(v) \odot \mathbf{D}_\beta(v) \in \mathbb{R}^{P \times 2}, \quad (41)$$

$$\mathbf{D}_\delta(v) = |\mathbf{C}_3(v) \odot \lambda^\delta(v) - \Lambda(v)| \in \mathbb{R}^{P \times 2}, \quad (42)$$

$$\Lambda_3(v) = \lambda^\delta(v) - \mathbf{A}_3(v) \odot \mathbf{D}_\delta(v) \in \mathbb{R}^{P \times 2}. \quad (43)$$

The values of $\mathbf{A}_2(v)$, $\mathbf{A}_3(v)$, $\mathbf{C}_2(v)$, and $\mathbf{C}_3(v)$ are determined in the same manner as $\mathbf{A}_1(v)$ and $\mathbf{C}_1(v)$. Due to this randomness, the population continuously explores new extrema during the convergence process guided by the three leading individuals.

The actual position vector of the entire population at iteration v is obtained as the average of $\mathbf{\Lambda}_1(v)$, $\mathbf{\Lambda}_2(v)$ and $\mathbf{\Lambda}_3(v)$.

$$\mathbf{\Lambda}(v) = \frac{1}{3}[\mathbf{\Lambda}_1(v) + \mathbf{\Lambda}_2(v) + \mathbf{\Lambda}_3(v)] \in \mathbb{R}^{P \times 2}. \quad (44)$$

After updating the positions, the entire population evaluates the new objective function values and subsequently updates the positions of the three leading individuals. The algorithm proceeds according to the pseudocode presented in Algorithm 2.

Algorithm 2: GWO Algorithm for Regularization Terms Selection

Step 1. Initialize P , the maximum number of iterations v_{max} , the search bounds λ_a^u and λ_b^u , set $v = 0$.
Step 2. Randomly initialize $\mathbf{\Lambda}(0)$, evaluate $\mathcal{J}(\mathbf{\Lambda}^p(0))$ for each individual, and select $\lambda^\alpha(0)$, $\lambda^\beta(0)$, $\lambda^\gamma(0)$.
Step 3. While $\mathbf{\Lambda}(v) - \mathbf{\Lambda}(v-1) < \delta$:
 Step 3.1. Update $a = 2 \left(1 - \frac{v}{v_{max}} \right)$.
 Step 3.2. Generate random matrices $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3, \mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3$.
 Step 3.3. Update the population positions according to Eqs. (36)–(44).
 Step 3.4. Re-evaluate the objective function and update $\lambda^\alpha, \lambda^\beta, \lambda^\gamma$.
 Step 3.5. Set $v = v + 1$.
Step 4. Return λ^α .

3 RESULTS AND DISCUSSION

The data acquisition system is illustrated in Figure 1. The gear-driven gripper mechanism compresses the AMBU bag using a DC motor, which is driven by a control circuit based on the Arduino DUE board with a fixed sampling period of 1 ms. The controller generates PWM signals to regulate the motor motion and thus control the airflow delivered to the respiratory system. The air from the ventilator is connected to a Michigan Instrument training test lung (TTL) module. The TTL module is used to simulate the mechanical properties of the human lungs. In the TTL module, a compliance spring is used to simulate the lung compliance, while a resistance adapter is connected in the breathing circuit to simulate the airway resistance. In this setup, the airway resistance and lung compliance were set to 5 cmH₂O/l/s and 50 ml/cmH₂O, respectively. An exhaust valve is incorporated to release air into the environment during exhalation. To collect the dataset for the identification algorithm, the gripper angle was controlled to move from the zero position to an angle of 20 degrees within 1.5 s, following three trajectories: a cosine function, a parabolic function, and a ramp function, referred to as a smooth periodic profile, a second-order polynomial profile, and a linear profile, respectively. The gripper then returned to 0 degrees, and the process was repeated for six cycles. The tidal volume flowing into and out of the TTL module was recorded using the Sensirion SFM3300 airflow sensor. Therefore, the input and output of the control system are the PWM duty cycle and the tidal volume, respectively. The smooth periodic profile dataset was used for model identification, while the other two datasets were used to validate the estimated model.

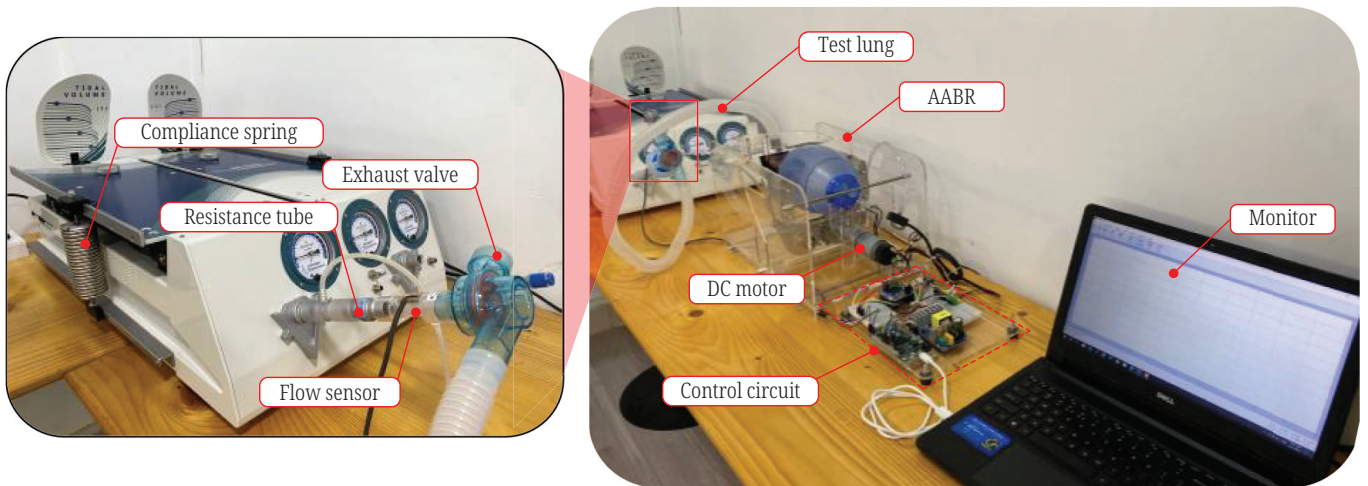


Fig. 1. Data acquisition system setup

The transfer function of the AABR system can be expressed as $G(s) = K/(s^2 + p_1s + p_2)$ in the continuous-time domain [22]. Under discretization using a zero-order hold (ZOH), the ARX model of $G(s)$ has orders $n = 2$ and $m = 2$. Therefore, $n^* = m^* = 2$ will be selected as the regression orders for the input and output of the prediction model. Let $\text{RMSE}_v(\cdot)$ denote the RMSE evaluated on the second-order polynomial profile dataset. The GWO optimization problem is then formulated as $\lambda^* = \arg \min_{\lambda \in \Psi} \text{RMSE}_v(\lambda)$.

The estimated parameters corresponding to each candidate point (λ_a, λ_b) are computed using the smooth periodic profile dataset. The search space of the GWO algorithm is bounded by $\Psi = \{(\lambda_a, \lambda_b) \in \mathbb{R}^2 \mid 0.002 \leq \lambda_a \leq 2, 0.002 \leq \lambda_b \leq 2\}$.

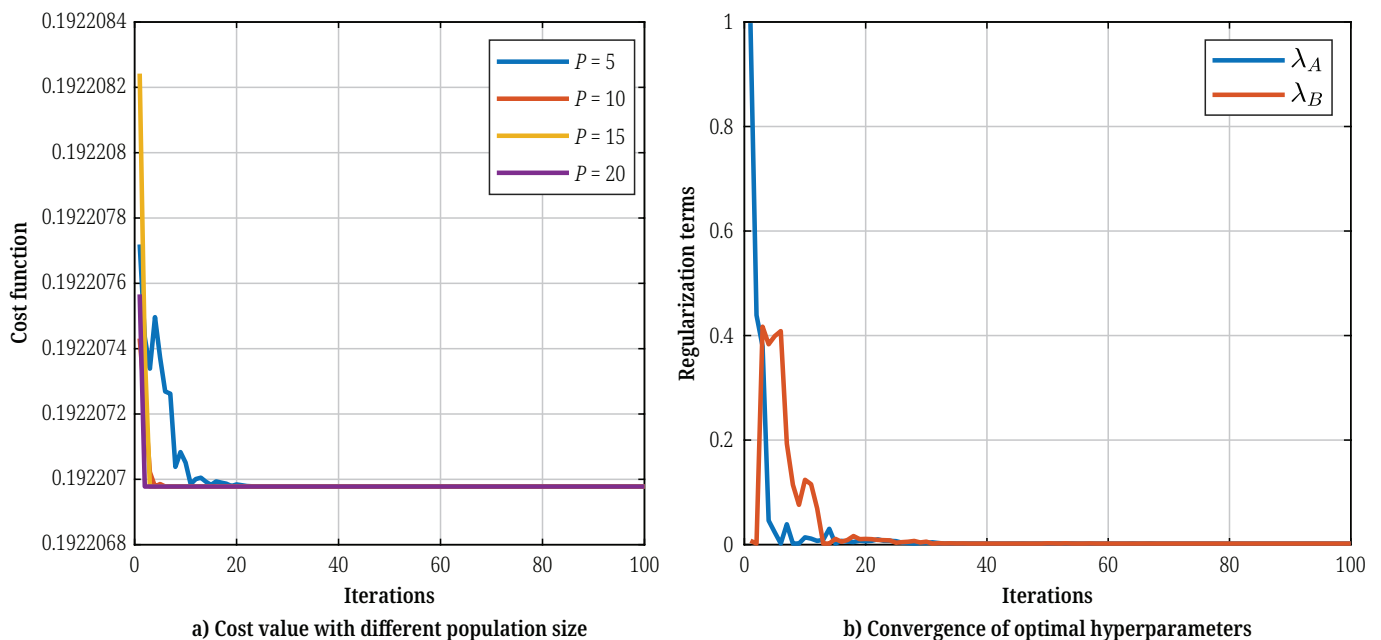


Fig. 2. (Continued)

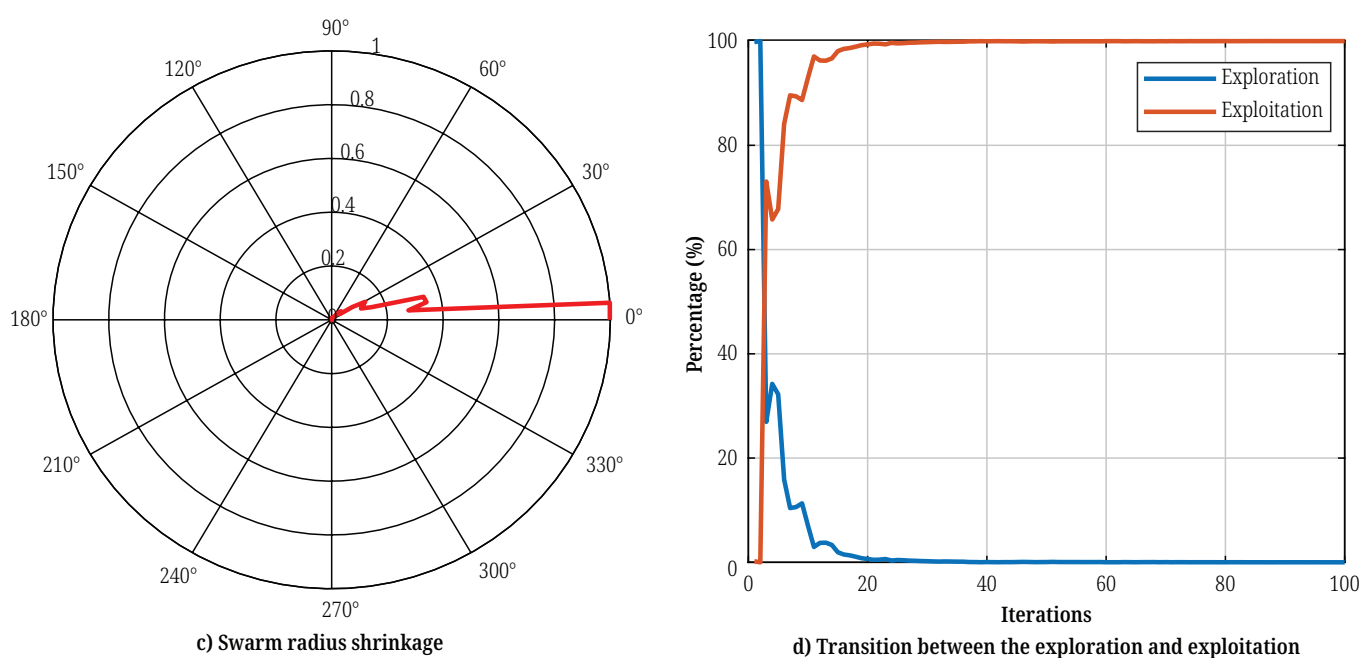


Fig. 2. Convergence and search behavior of GWO

The optimization performance and search behavior of the GWO in tuning the regularization parameters are comprehensively evaluated in Figure 2. To ascertain the algorithm's robustness, the cost function's convergence trajectory was initially investigated across varying population sizes, as depicted in Figure 2a. Notably, the algorithm demonstrates exceptional computational efficiency; even with a remarkably diminutive swarm size of $P = 5$, the objective function successfully escapes initial fluctuations and sharply converges to the global minimum within merely 20 iterations. This rapid stabilization underscores the viability of the proposed approach for practical system identification tasks where minimizing computational overhead is critical.

Consequently, subsequent behavioral analyses were conducted utilizing the highly efficient configuration of $P = 5$. Figure 2b illustrates the convergence profiles of the hyperparameters. While λ_b exhibits pronounced early-stage oscillations, suggesting a broader global search requirement for the moving average terms, both parameters seamlessly lock into their optimal configuration once the steady state is achieved. Ultimately, the GWO algorithm converged to an optimal parameter set of $\lambda_a = 0.002$ and $\lambda_b = 0.002$, yielding a minimized loss function of 0.192207 ml on the second-order polynomial profile dataset.

The underlying mechanics driving this robust convergence are further elucidated by examining the algorithmic phase transitions. As evidenced by Figure 2d, the optimization process undergoes a clear paradigm shift. During the initial phase, exploration strongly dominates, empowering the wolf pack to extensively scan the parameter space and evade local optima. Subsequently, around the 10th iteration, a critical crossover occurs where exploitation aggressively takes precedence. This intensive local fine-tuning phase is visually corroborated by the polar coordinate representation in Figure 2c, which reveals a rapid contraction of the swarm's search radius toward the origin, definitively confirming the precise and stable localization of the optimal hyperparameters.

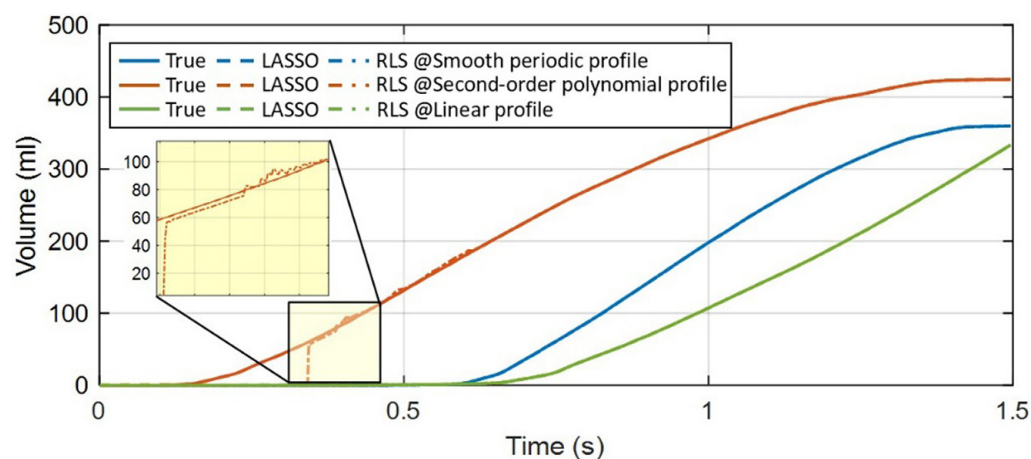


Fig. 3. Estimated output from the LASSO and RLS algorithm with three datasets

Table 1. Performance comparison of LASSO and RLS with different datasets

Dataset	LASSO		RLS	
	RMSE	MAE	RMSE	MAE
Smooth periodic profile (train)	0.177893	0.157929	0.009821	0.006356
Second-order polynomial profile (validate)	0.195727	0.160596	11.436522	3.886867
Linear profile (validate)	0.150998	0.139875	0.021425	0.014557
Average	0.174872	0.1528	3.822590	1.302593

To comprehensively evaluate its performance and advantages, the proposed LASSO algorithm is compared against the conventional RLS algorithm, with the results depicted in Figure 3 and Table 1. Both algorithms were executed on three different datasets, and the accuracy of the estimated models was measured by using the RMSE and mean absolute error (MAE) criteria. Figure 3 shows graphical representations of the predicted outputs of both algorithms, LASSO and RLS, alongside the actual ground truth for the smooth periodic, second-order polynomial, and linear profiles. Here, the blue line represents the training dataset, and the other lines represent the validation datasets.

It is obvious that the predicted outputs of the LASSO model are successful in capturing the transient trends of the ground truth for all the datasets. This strong tracking capability of the LASSO model with the predicted output validates the exceptional generalization capability of the LASSO-based predictive model for the control system, eliminating the occurrence of overfitting phenomena altogether. On the contrary, though the predicted output of the RLS algorithm tries to track the actual data trajectory, the occurrence of severe overfitting in the case of the second-order polynomial profile data leads to a significant deviation in the prediction, especially during the initial 0.6 seconds, as validated in the metrics obtained in Table 1. On average across the three datasets, the LASSO algorithm produces an outstandingly low average value for the RMSE of 0.174872 ml and an MAE value as low as 0.1528 ml. On the contrary, the RLS algorithm produces much inferior values for the RMSE and MAE values, which shoot up to an average value of 3.822590 ml and an MAE value of 1.302593 ml, respectively. In general, even though the RLS algorithm produces much lower values for the prediction errors in comparison to LASSO for the smooth periodic and linear profile datasets, it suffers from severe overfitting. This is because

the RLS algorithm is fitting too much to the noise and disturbances present in the datasets and is unable to capture the underlying dynamics of the system. This is also due to the severe fluctuations in the values for the RMSE and MAE values for the RLS algorithm across the datasets. For example, the value for the RMSE shoots up drastically from a mere 0.0098 for the RLS algorithm in the training set to a value as high as 11.4365 for the second-order polynomial profile validation set. In a stark contrast to the above, the LASSO algorithm shows exemplary stability and robustness with highly consistent error margins, with the value of the RMSE being closely bounded between the range of 0.15 and 0.19 for all the datasets, whether seen.

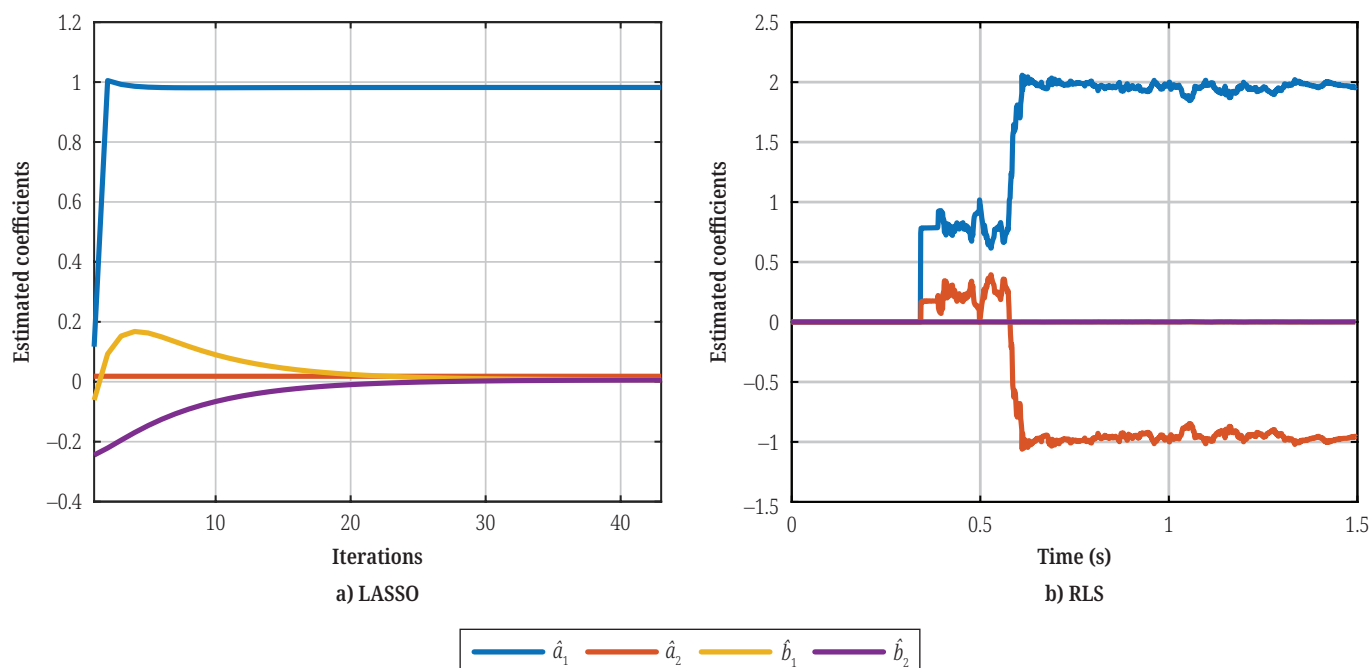


Fig. 4. Coefficients evolution of the LASSO and RLS algorithm

For a more detailed understanding of the underlying mechanisms that lead to such a performance gap between the two estimators, Figure 4 shows a plot of the evolution of the estimated coefficients obtained by both identification methods. It is clear from Figure 4a that a smooth convergence profile is obtained by the LASSO-based method for the estimated coefficients. The coefficients converge smoothly and steadily to their steady-state values and then plateau after about 30 optimization iterations without any significant oscillations in between. This smooth convergence profile may be due to the regularization effect that is incorporated by the LASSO method by using the $\mathcal{L}_1$ penalty term in the cost function formulation. On the other hand, Figure 4b shows a plot of the estimated coefficients obtained by the RLS algorithm. It is clear that significant oscillations exist in the estimated coefficients obtained by this method due to the presence of some transient spikes in between, especially around the 0.5–0.6 s interval. Such variations indicate that the RLS estimator is more sensitive to measurement noise and short-term disturbances. Since the standard RLS formulation lacks an explicit regularization mechanism, the algorithm continuously adjusts its coefficients in response to instantaneous variations in the data. As a result, the estimated parameters may deviate from the underlying control system dynamics, which can lead to degraded prediction accuracy. In contrast, the LASSO-based method provides more stable parameter estimates and better captures the dominant dynamics of the control system.

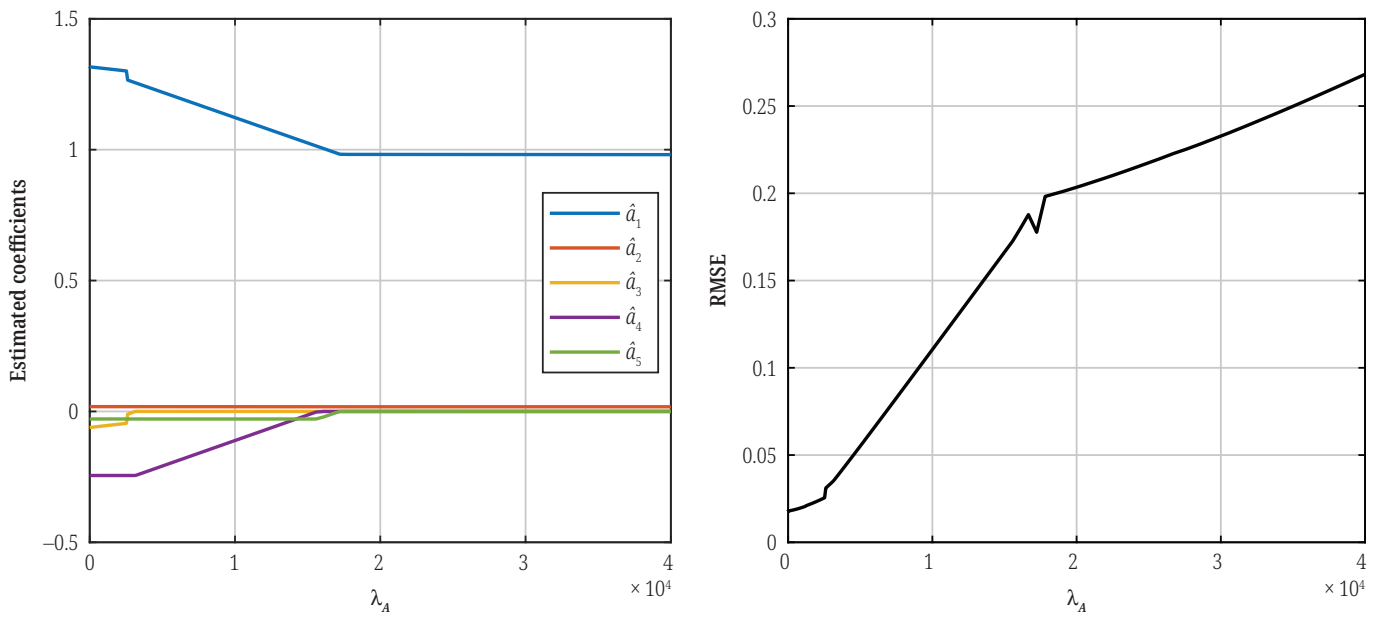


Fig. 5. Regularization path of autoregressive coefficients when $\lambda_b = 0.002$, $\lambda_a \in [0.002; 4 \times 10^4]$

To investigate the system dynamics' sensitivity to the regularization hyperparameters, the coefficient shrinkage paths and their corresponding impacts on prediction accuracy were analyzed independently. Figure 5 illustrates the evolution of the autoregressive parameters $\hat{a}_i$ and the resulting RMSE of the training process as λ_a is swept from 0.002 to 40,000 on a logarithmic scale, with λ_b held constant at 0.002. As the penalty increases, higher-order terms are swiftly compressed to zero, successfully isolating the dominant first- and second-order inertia of the AABR actuation mechanism. However, the RMSE degrades monotonically along this path. This strictly ascending error trend indicates that the physical system relies heavily on its autoregressive properties. Penalizing the AR coefficients too aggressively forces the model to ignore the inherent mechanical momentum, immediately sacrificing prediction fidelity.

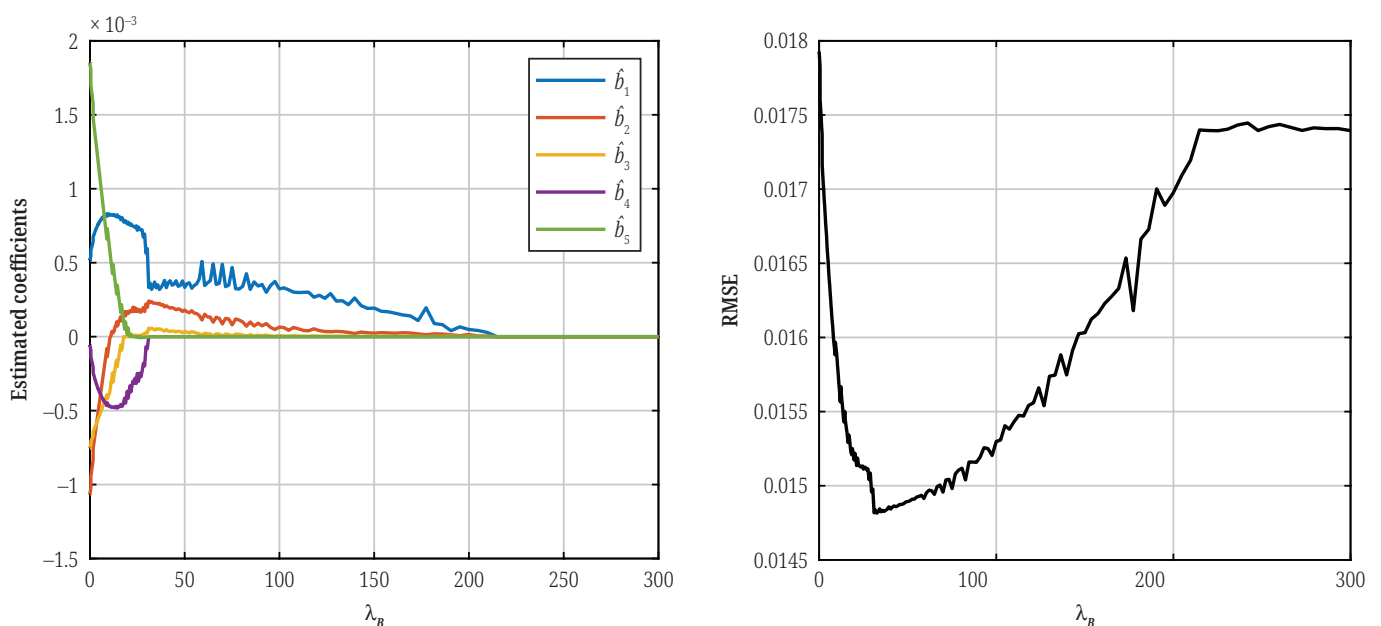


Fig. 6. Regularization path of exogenous input coefficients when $\lambda_a = 0.002$, $\lambda_b \in [0.002; 300]$

A distinctly different and important phenomenon is noted in relation to the exogenous input coefficients in Figure 6. When λ_b increases, the redundant higher-order input parameters are again eliminated. Yet, it is noted that instead of a deterioration in the RMSE values, a distinct global minimum in the error values is observed at λ_b values around 25, and then increases again. This U-shaped error curve essentially represents a bias-variance trade-off in system identification problems. When low values of λ_b are imposed as a penalty term, it is noted that the model is overfitting the high-frequency variations and noise in the input PWM signal. Conversely, when a moderate value of λ_b penalty is imposed in the model, it essentially filters out this noise-fitting effect and leads to a minimum in the validation error values.

4 CONCLUSION

With the integration of LASSO and GWO for the automated tuning of hyperparameters, the system identification problem has been successfully addressed. The effectiveness of the proposed solution has been verified by experimental evaluations that clearly demonstrated the decisive superiority of the proposed solution compared with the traditional RLS-based solution. The traditional RLS solution exhibited aggressive memory effects and locally noisy behavior, as reflected by the erratic variation in parameters and the significant problem of overfitting for the validation data sets. Specifically, where the RMSE shot up to 11.43 ml, the proposed solution has been effective in filtering out the disturbances. The smooth convergence of parameters and the tight bounds of the prediction errors from 0.15 to 0.19 ml for completely different sets of breathing trajectories are also significant. Moreover, the proposed solution has been effective in providing physical insights into the system behavior. Notably, the analysis of the regularized path has typically effectively illustrated the role of the optimal solution in suppressing high-frequency input noise while retaining the actuator's inherent momentum. Despite these specifically theoretical and experimental results, the current methodology is based on offline data collection, which was validated using static simulated lung conditions with fixed compliance and airway resistance values. However, in real-world clinical conditions, human respiratory mechanics are nonlinear.

In future work, the focus of the subsequent studies is on the implementation of the closed-loop control strategy, while the extremely stable, time-invariant nominal model obtained from the current LASSO framework is the perfect starting point for the design of sophisticated trajectory-tracking controllers, such as Iterative Learning Control. Subsequent studies on the ventilator's performance are expected to be conducted on the physical device, while the nonlinear time-varying lung mechanics are taken into account, in order to ensure the mechanical robustness as well as the patient's safety.

5 ACKNOWLEDGEMENT

We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for supporting this study.

6 REFERENCES

- [1] I. R. Drennan, M. R. Neth, and B. K. Walsh, “Advancements in ventilation during cardiac resuscitation,” *Crit. Care Clin.*, vol. 42, no. 1, pp. 131–147, 2026. <https://doi.org/10.1016/j.ccc.2025.08.007>
- [2] C. T. Truong, K. H. Huynh, V. T. Duong, H. H. Nguyen, L. A. Pham, and T. T. Nguyen, “Model-free volume and pressure cycled control of automatic bag valve mask ventilator,” *AIMS Bioeng.*, vol. 8, no. 3, pp. 192–207, 2021. <https://doi.org/10.3934/bioeng.2021017>
- [3] K. Iyengar, S. Bahl, R. Vaishya, and A. Vaish, “Challenges and solutions in meeting up the urgent requirement of ventilators for COVID-19 patients,” *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, vol. 14, no. 4, pp. 499–501, 2020. <https://doi.org/10.1016/j.dsx.2020.04.048>
- [4] S. Saisukirtha, M. Omprakash, S. Shane Steffy, and S. Dilip Kumar, “Smart ventilation bag with adjustable oxygen range and automated pressure regulation,” in *3rd International Conference on Intelligent Data Communication Technologies and Internet of Things, IDCIoT*, 2025, pp. 1014–1019. <https://doi.org/10.1109/IDCIOT64235.2025.10914958>
- [5] C. T. Truong, K. H. Huynh, V. T. Duong, H. H. Nguyen, L. A. Pham, and T. T. Nguyen, “Linear regression model and least square method for experimental identification of AMBU bag in simple ventilator,” *International Journal of Intelligent Unmanned Systems*, vol. 11, no. 3, pp. 378–395, 2022. <https://doi.org/10.1108/IJIUS-07-2021-0072>
- [6] C. T. Truong, T. D. Phan, M. H. Nguyen, V. T. Duong, H. H. Nguyen, and T. T. Nguyen, “Ensemble bidirectional long short-term memory network identification for nonlinear autoregressive exogenous model: Application to dual double-acting piston pump,” *IEEE Access*, vol. 13, pp. 174445–174461, 2025. <https://doi.org/10.1109/ACCESS.2025.3612891>
- [7] M. A. Khan, D.-E.-Z. Baig, H. Ali, and F. R. Albogamy, “Optimized System Identification (SI) of Brushless DC (BLDC) motor using data-driven modeling methods,” *Scientific Reports*, vol. 15, no. 1, Art. no. 8497, 2025. <https://doi.org/10.1038/s41598-025-93444-0>
- [8] C. T. Truong, T. D. Phan, V. T. Duong, H. H. Nguyen, and T. T. Nguyen, “Model identification of ventilation air pump utilizing ridge-momentum regression and grid-based structure optimization,” *Mathematical Biosciences and Engineering*, vol. 22, no. 8, pp. 2020–2038, 2025. <https://doi.org/10.3934/mbe.2025074>
- [9] N. Arridu, M. Lippi, M. Franceschelli, and A. Gasparri, “On ensemble learning models for autoencoder-based identification of nonlinear dynamical systems,” *IEEE Access*, vol. 13, pp. 169071–169082, 2025. <https://doi.org/10.1109/ACCESS.2025.3611914>
- [10] N. Xu and F. Ding, “Recursive estimation algorithms based on the least squares and their convergence for a class of time-varying systems,” *Nonlinear Dynamics*, vol. 111, no. 19, pp. 18191–18213, 2023. <https://doi.org/10.1007/s11071-023-08816-w>
- [11] K. Gajula, L. K. Marepalli, X. Yao, and L. Herrera, “Recursive least squares and adaptive kalman filter-based state and parameter estimation for series arc fault detection on DC microgrids,” *IEEE J. Emerg. Sel. Top. Power Electron.*, vol. 10, no. 4, pp. 4715–4724, 2022. <https://doi.org/10.1109/JESTPE.2021.3135409>
- [12] A. Dong, A. Starr, and Y. Zhao, “End-to-end identification of autoregressive with exogenous input (ARX) models using neural networks,” *Machine Intelligence Research*, vol. 22, no. 1, pp. 117–130, 2025. <https://doi.org/10.1007/s11633-024-1523-3>
- [13] X. Chen and J. Li, “Multi-sensor recursive EM algorithm for robust identification of ARX models,” *Sensors (Basel)*, vol. 25, no. 22, Art. no. 7060, 2025. <https://doi.org/10.3390/s25227060>
- [14] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *J. R. Stat. Soc. Series B Stat. Methodol.*, vol. 58, no. 1, pp. 267–288, 1996. <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>

- [15] H. Zhou, C. Ibrahim, W. X. Zheng, and W. Pan, "Sparse Bayesian deep learning for dynamic system identification," *Automatica*, vol. 144, Art. no. 110489, 2022. <https://doi.org/10.1016/j.automatica.2022.110489>
- [16] N. V. N. Sowjanya and M. Palanivelan, "Study of sparse optimization for system identification and machine learning," in *Algorithms in Advanced Artificial Intelligence*, London: CRC Press, 2025, pp. 731–737. <https://doi.org/10.1201/9781003641537-111>
- [17] J. Gillariose, J. Joseph, and C. Chesneau, "Lasso and ridge regression: A comprehensive review of applications and developments in machine learning," *International Journal of Data Science and Analytics*, vol. 21, no. 1, 2025. <https://doi.org/10.1007/s41060-025-00957-y>
- [18] J. Singh, J. K. Sandhu, and Y. Kumar, "Metaheuristic-based hyperparameter optimization for multi-disease detection and diagnosis in machine learning," *Service Oriented Computing and Applications*, vol. 18, no. 2, pp. 163–182, 2024. <https://doi.org/10.1007/s11761-023-00382-8>
- [19] Y. Liu, A. As'array, M. K. Hassan, A. A. Hairuddin, and H. Mohamad, "Review of the grey wolf optimization algorithm: Variants and applications," *Neural Computing and Applications*, vol. 36, no. 6, pp. 2713–2735, 2024. <https://doi.org/10.1007/s00521-023-09202-8>
- [20] V. H. S. Pham and V. K. Nguyen, "A study on optimization of HVAC system in buildings using gray wolf optimizer (GWO) and artificial neural network (ANN)," *Asian Journal of Civil Engineering*, vol. 24, no. 8, pp. 3743–3757, 2023. <https://doi.org/10.1007/s42107-023-00747-6>
- [21] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *J. Stat. Softw.*, vol. 33, no. 1, pp. 1–22, 2010. <https://doi.org/10.18637/jss.v033.i01>
- [22] C. T. Truong, T. D. Phan, V. T. Duong, H. H. Nguyen, and T. T. Nguyen, "L1 adaptive control for a low-cost bag-valve-mask ventilator in volume-controlled continuous mandatory ventilation mode: Design, simulation and experiment," *Eur. J. Control*, vol. 87, Art. no. 101409, 2025. <https://doi.org/10.1016/j.ejcon.2025.101409>

7 AUTHORS

Trung Dat Phan received his B.E. degree from Mechatronics Engineering, Ho Chi Minh City University of Technology, Vietnam in 2023. He is currently working at the Key Laboratory of Digital Control and System Engineering, Ho Chi Minh City, Vietnam. His fields of interest include mechatronics, medical equipment, application control, and computer-aided engineering (E-mail: ptdat.sdh241@hcmut.edu.vn).

Cong Toai Truong received his B.E. degree from Mechatronics Engineering, University of Technology and Education Ho Chi Minh City, Vietnam in 2019, his M.S. degree from Mechatronics Engineering, Ho Chi Minh City University of Technology, Vietnam in 2021. He is currently working at the Key Laboratory of Digital Control and System Engineering, Ho Chi Minh City, Vietnam. His fields of interest include mechatronics, medical equipment, manufacturing design and industrial automation (E-mail: tctoai.sdh232@hcmut.edu.vn).

Huy Hung Nguyen received the M.Sc. degree from Ho Chi Minh City University of Technology, Ho Chi Minh, Vietnam in 2001 and the Ph.D. degree in Mechanical Design Engineering from Pukyong National University, Busan, Korea in 2018. He joined the Faculty of Electronics and Telecommunication, Sai Gon University in 2011 as a lecturer. He is currently working at the Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology and Engineering, Ho Chi Minh City, Vietnam. He has been teaching several undergraduate and graduate courses such as electronic circuits, Automation control theory and Advanced

Mathematical Engineering. His current research interests include Control Theory, Power Electronics, and industrial equipment (E-mail: ngghung@sgu.edu.vn).

Tan Tien Nguyen received his B.S. degree in from Mechanical Engineering, Ho Chi Minh City University of Technology, Vietnam in 1990, his M.S. degree, and his Ph.D. degree in Mechatronics Engineering, Pukyong National University, Busan, Korea in 1998 and 2001, respectively. Since 1990, he has been a lecturer in the Department of Mechanical Design at HCMUT, and since 2005, he has been a lecturer in the Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology, Vietnam National University Ho Chi Minh City, Vietnam. Since 12.2018, he has held the position of the Director of the Key Laboratory of Digital Control and System Engineering. His current research areas include control theory, robotics and automation, electromechanical systems and their applications in industrial automation (E-mail: nttien@hcmut.edu.vn).

Van Tu Duong received his B.S. degree from Mechanical Engineering, University of Technical Education Ho Chi Minh City, Vietnam in 2009, his M.S. degree from Mechanical Engineering, Ho Chi Minh City University of Technology, Vietnam in 2012 and his Ph.D. degree in Mechanical Design Engineering, Pukyong National University, Busan, Korea in 2016. He is currently a lecturer at the Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology, Vietnam National University Ho Chi Minh City, Vietnam. His fields of interest include adaptive Control, manufacturing design, and application control (E-mail: dvtu@hcmut.edu.vn).