Ionics
https://doi.org/10.1007/s11581-025-06601-w

RESEARCH

Revolutionizing parameter identification of PEM fuel cell using adaptive differential evolution algorithm based on deeply-informed mutation strategy and restart mechanism optimization

Manish Kumar Singla[1,2] · Muhammed Ali[3] · Ramesh Kumar[4] · Pradeep Jangir[5,6,7] · Absalom E. Ezugwu[8] · Mohamed Abouhawwash[9,10]

Received: 6 May 2025 / Revised: 10 July 2025 / Accepted: 29 July 2025

Abstract

Proper modeling of proton exchange membrane fuel cells (PEMFCs) depends on the accurate determination of the unknown parameters that are essential in predicting performance, but usually not given by manufacturers. The Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) that is proposed herein proposes a new method of optimization of the seven parameters ($\xi 1$, $\xi 2$, $\xi 3$, $\xi 4$, λ, R_c, and β) that control the activation, ohmic, and concentration overpotentials. In contrast to the traditional approaches, ADEMDR incorporates a highly informed mutation strategy that dynamically modifies evolutionary direction based on elite individuals, suboptimal solutions, and search history. Also, there is a restart mechanism that avoids premature convergence by reintroducing good solutions, which greatly improves the balance between exploration and exploitation. The results of validation of ADEMDR on 12 commercial PEMFC stacks under different operating conditions prove its superiority to the state-of-the-art algorithms, such as L-SHADE, PSO, DE, PLO, HARD-DE, and NDE. The algorithm has the minimum sum of squared error (SSE), absolute error (AE), and mean bias error (MBE), and the improvement of computational efficiency is more than 98% in terms of run time. The optimized parameters allow very precise prediction of I-V and P–V curves, which are very close to experimental data. The strength and scalability of ADEMDR render it applicable in real-time PEMFC modeling, and it has a great potential of being implemented in embedded systems in monitoring and control systems.

Keywords Parameter estimation · Proton exchange membrane fuel cell · PEMFC · Hybrid algorithm · Optimization

✉ Mohamed Abouhawwash
mohamed.abouhawwash@kfupm.edu.sa

Manish Kumar Singla
msingla0509@gmail.com

Ramesh Kumar
rameshkumarmeena@gmail.com

Absalom E. Ezugwu
Absalom.ezugwu@nwu.ac.za

[1] Department of Biosciences, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, India 602 105

[2] Jadara University Research Center, Jadara University, Irbid 21110, Jordan

[3] Fuel Cell Institute, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia

[4] Department of Interdisciplinary Courses in Engineering, Chitkara University Institute of Engineering & Technology, Chitkara University, Rajpura, Punjab, India

[5] Department of CSE, Graphic Era Hill University, Dehradun 248002, India

[6] Department of CSE, Graphic Era Deemed, To Be University, Dehradun 248002, Uttarakhand, India

[7] Applied Science Research Centre, Applied Science Private University, Amman 11937, Jordan

[8] North-West University, 11 Hofman Street, Potchefstroom 2520, South Africa

[9] Department of Industrial and Systems Engineering, King Fahd University of Petroleum and Minerals, 31261 Dhahran, Saudi Arabia

[10] Interdisciplinary Research Center of Smart Mobility and Logistics (IRC-SML), King Fahd University of Petroleum and Minerals, 31261 Dhahran, Saudi Arabia

Published online: 01 September 2025

Introduction

The escalating environmental and energy crises, driven by dwindling fossil fuel reserves, water contamination, energy scarcity, and global warming, have become a major international concern [1, 2]. This has intensified the search for renewable and clean energy alternatives, with hydrogen emerging as a highly promising solution [3]. The abundance of hydrogen on earth and its combustion without emissions make it an attractive energy source. Significant research efforts focused on proton exchange membrane fuel cells (PEMFCs) to maximize hydrogen potential [4, 5]. PEMFCs are advanced electrochemical devices that directly convert the chemical energy stored in hydrogen into electricity and heat without the disadvantages of combustion or harmful emissions [6]. These fuel cells usually consist of an anode, a cathode, a polymeric membrane, and an electrolyte. Within the anode compartment, hydrogen gas undergoes a chemical reaction that generates protons and electrons [7–9]. The electrons travel to the cathode compartment through an external circuit, while the protons migrate through the membrane [10, 11]. In the cathode compartment, pure oxygen reacts with the incoming electrons and protons, producing water. To ensure optimal and reliable operation of the PEMFC stack, robust system designs and effective operational strategies are crucial [12, 13]. Fuel cells are complex systems comprising various interconnected subsystems, including monitoring, control, and integrated units for various instruments. Effective inter-subsystem cooperation is essential for the ideal operation of fuel cells. The control system plays a critical role in managing the operational process, covering the regulation of reagents, water management, and thermal systems. This allows direct adjustment of the main operational parameters. At the same time, the monitoring system continually evaluates the operational status of the fuel cell [14].

Electrochemical modeling, a powerful approach that integrates electrochemistry and mathematical principles, is crucial for predicting fuel cell output, enhancing PEMFC performance, and optimizing power conditioning circuits [15]. Furthermore, fuel cell modeling facilitates the design process and provides a deeper understanding of the underlying electrochemical reactions within PEMFCs. In recent years, numerous mathematical models have been developed for fuel cells. For example, Danoune et al. proposed the Whale Optimization Algorithm (WOA) to determine the ideal parameters for PEMFC models. WOA aims to create a precise and efficient PEMFC model, minimizing the discrepancy between the experimental and planned polarization curves. WOA is particularly effective in mitigating the challenges of premature convergence and great place, which are common limitations of many existing modeling approaches. To validate the effectiveness of WOA, a series of experiments were conducted on a Heliocentris FC50 PEMFC test bench to demonstrate its ability to accurately simulate PEMFC generator behavior. The accuracy of the developed model was rigorously evaluated across a range of operating conditions. WOA's effectiveness was also validated using data from two different PEMFCs: Ballard and V BCS-500W. Comparative analyses were performed against various established optimization algorithms in the literature, along with the appropriate statistical analysis. The results unequivocally demonstrate the superior performance of WOA. Among all the methods evaluated, WOA has reached the lowest average absolute error, indicating the highest accuracy in model forecasts [16]. In another study, Zhang et al. Introduced a new Hybrid Alexnet/Extreme Learning Machine (ELM) network to optimize the resource extraction process for PEMFCs.

The primary objective was to utilize a novel hybrid AlexNet/ELM network to minimize the discrepancy between the experimentally measured output voltage and the predicted voltage [17]. A modified version of the African optimizer Vulture (MAVO) was used to improve the Alexnet/ELM generation process. To evaluate the effectiveness of this proposed approach, it was applied to a real-world PEMFC test case for resource extraction. The performance of the approach was rigorously validated by comparing the forecasts of the Alexnet/ELM model with the real experimental data. The proposed AlexNet/ELM network exhibited higher validation accuracy than empirical data. In a separate study, Zhou et al. introduced a new problem formulation for efficiently identifying PEMFC parameters, which they solved using the Improved Fish Migration Optimizer (IFMO). The objective function of the fuel cell has been defined and implemented using the MATLAB code. The performance of this new formulation has been strictly evaluated through extensive simulations, comparing it to existing and conventional objective functions [18]. To demonstrate the superiority of the proposed formulation of traditional curve adjustment methods, a comprehensive assessment was performed, considering factors such as convergence speed, absolute tension error, and the value of the objective function.

The research selected its algorithms for comparison because they show widespread use, proven success, and direct applicability to PEMFC parameter identification. The chosen algorithms, including L-SHADE, Particle Swarm Optimization (PSO), Differential Evolution (DE), Polar Lights Optimizer (PLO), Newton Differential Evolution Algorithm (NDE), Local Search-Based Differential Evolution (LSDE), Parameter Adaptation-Based Differential Evolution (PaDE), and Cooperative Strategy-Based Differential

Evolution (AOA), represent the diverse range of established optimization techniques commonly used in similar applications. These algorithms have proved adequate because numerous studies in the literature demonstrate their effectiveness in solving complex nonlinear optimization problems, including PEMFC modeling and parameter estimation.

Two main reasons explain the absence of Reinforcement Learning (RL) and Bayesian Optimization (BO) from the study. The main objective of this research evaluates the proposed ADEMDR algorithm by comparing it to established classical optimization methods which have become standard in PEMFC parameter identification studies. The classical optimization methods continue to function as benchmarks owing to their previous performance and universal adoption allowing an unambiguous performance evaluation. The application of RL and BO to PEMFC parameter identification remains under development because their computational requirements might not match real-time modeling needs and control objectives of this study. Further work should integrate RL or BO with ADEMDR to boost its abilities while this evaluation demonstrates ADEMDR's superiority against established methods in PEMFC parameter optimization applications.

Comparison of ADEMDR with recognized standard algorithms enables the study to prove its performance reliability as well as benchmarking accuracy against research standards in PEMFC modeling. The research methodology provides meaningful results which practitioners and researchers can utilize in their PEMFC modeling work.

Literature highlights the growing prominence of artificial intelligence, particularly metaheuristic algorithms, in model identification due to their inherent simplicity and ability to address the complex, non-linear nature of PEMFC systems. However, the "No Free Lunch" theorem underscores the limitations of any single optimization method in achieving universal optimality. This realization motivates the pursuit of novel and improved metaheuristic algorithms to consistently achieve superior results. In this work, we introduce an enhanced metaheuristic algorithm specifically designed to optimize parameters and minimize the error between predicted and experimentally measured output voltages. This approach deviates from traditional methods by prioritizing parameter optimization as the primary objective. Extensive evaluation against existing algorithms demonstrated the superior performance of ADEMDR, achieving the lowest error levels. Furthermore, the proposed approach underwent rigorous testing for consistency and reliability across a wide range of pressure and temperature conditions, confirming its effectiveness. By combining advanced optimization techniques with robust empirical validation, this methodology introduces a novel algorithm and sets a new standard for accuracy and reliability in PEMFC parameter identification, ultimately contributing to improved fuel cell performance and efficiency. The key contributions of this work are as follows:

- This study optimizes PEMFC model identification by minimizing SSE using the ADEMDR method and compares various algorithmic approaches,
- This study include the detailed analysis of application of ADEMDR variants, specifically examining the performance of L-SHADE [19], Particle Swarm Optimization (PSO) [20], Differential Evolution (DE) [21], Polar Lights Optimizer (PLO) [22], Hierarchical archive based mutation strategy with depth information of evolution enhancement of differential evolution (HARD-DE) [23], Newton Differential Evolution Algorithm (NDE) [24], Local Search based Differential Evolution (LSDE) [25], Parameter Adaptation based Differential Evolution (PaDE) [26], and Cooperative Strategy based Differential Evolution (AOA) [27] algorithms.
- The analysis included 12 commercially available PEMFC stacks, specifically the BCS 500 W-PEM [28], 500 W SR-12PEM [29], Nedstak PS6 PEM [29], H-12 PEM [30], HORIZON 500 W PEM [30], and a 250 W-stack [31], among others.
- To ensure the consistency of the tested methods, a thorough statistical evaluation was performed.
- The achieved results were compared to findings from existing literature to establish a benchmark.

Method

The typical electrochemical reactions, expressed mathematically, that occur at the anode and cathode of a PEMFC stack are as follows:

1. Anode reaction

   At the anode, hydrogen gas (H_2) undergoes oxidation, splitting into protons (H^+) and electrons (e^-). These protons then travel through the proton exchange membrane (PEM), while the electrons flow through an external circuit, generating an electrical current.

   $$Hydrogen\ Oxidation\ : H_2 \rightarrow 2H^+ + 2e^- \quad (1)$$

2. Cathode reaction

   At the cathode, oxygen (O_2) combines with protons that have crossed the PEM from the anode and electrons that have flowed through the external circuit, producing water (H_2O) as a result.

   $$Oxygen\ Reduction\ : 1/2O_2 + 2H^+ + 2e^- \rightarrow H_2O \quad (2)$$

3. Overall cell reaction

The complete electrochemical process within the PEMFC is represented by the combination of hydrogen and oxygen to produce water, a reaction that also yields electrical energy.

$$Hydrogen - Oxygen\ Reaction: H_2 + 1/2O_2 \rightarrow H_2O \tag{3}$$

The voltage output of a single PEMFC is usually insufficient for practical applications (see Fig. 1). Consequently, PEMFC stacks, where multiple cells are connected in series, are used to achieve the required voltage. PEMFC performance is often modelled with semi-empirical equations. Accurate parameters are key for reliable simulations that calculate output voltage based on factors like reversible potential and various overpotentials.

$$VFC = E_{Nernst} - V_{act} - V_{ohmic} - V_{conc} \tag{4}$$

The total output voltage of a PEMFC stack comprising N cells in series is determined by:

$$V_{total} = NV_{FC} \tag{5}$$

The terms V_{act}, V_{ohmic}, and V_{conc} refer to the activation, ohmic, and concentration voltage drops, respectively. The thermodynamic potential is represented by E_{Nernst} and its value is determined by:

$$E_{Nernst} = 1.229 - 0.85 \times 10 - 3 \times (t - 298.15) + 4.308 \times 10 - 5 \times lnP_{Hydrogen}\sqrt{P_{Oxygen}} \tag{6}$$

$P_{Hydrogen}$ and P_{Oxygen} represent the partial pressures of H_2 and O_2 (in atm), while t is the fuel cell temperature (in Kelvin). Cell current affects these pressures. Equation 7 approximates the activation voltage loss (V_{act}).

$$v_{act} = -\left[\xi_1 + \left(\xi_2 \times T_{FC}\right) + \left(\xi_3 \times T_{FC} \times \ln\left(C_{O_2}\right)\right) + \left(\xi_4 \times T_{FC} \times \ln\left(I_{fc}\right)\right)\right] \tag{7}$$

Here, I_{FC} represents the fuel cell current, and $\xi 1$ through $\xi 4$ are numerical constants. The oxygen concentrations, denoted as CO_2 and CH_2 (in mol/cm^3), can be calculated using Eqs. 8 and 9.

$$C_{O_2} = \frac{P_{O_2}}{5.08 \times 10^6} \times \exp\left(\frac{498}{T_{FC}}\right) \tag{8}$$

$$C_{H_2} = \frac{P_{H_2}}{1.09 \times 10^6} \times \exp\left(\frac{-77}{T_{FC}}\right) \tag{9}$$

The calculation of the ohmic voltage loss (V_{ohm}) involves the fuel cell's resistance, which is give in Eq. 10.

$$v_{ohm} = I_{FC} \times \left(R_m + R_c\right) \tag{10}$$

where R_m and R_c denote the membrane and contact resistances, respectively, and Eqs. 11 and 12 are used to calculate R_m.

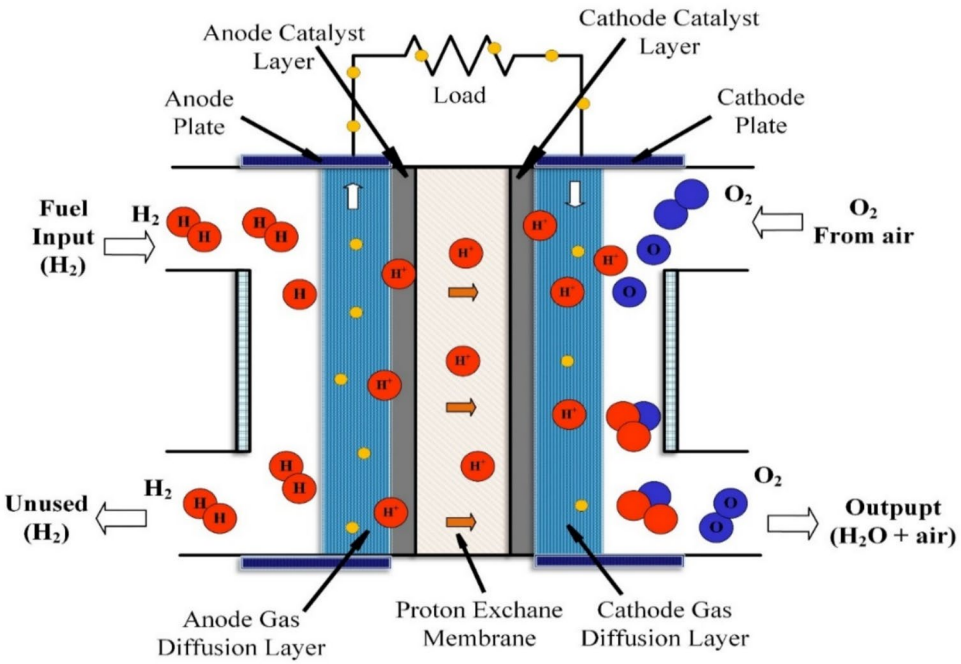


Fig. 1 Schematic of PEMFC [32]

$$R_m = \frac{\rho_m \times l}{M_A} \tag{11}$$

$$\rho_m = \frac{181.6 \times \left[1 + 0.03 \times J + 0.062 \times J^{2.5} \times \left(T_{FC}/303\right)^2\right]}{[\lambda - 0.634 - 3 \times J] \times \exp^{(4.18 \times (T_{FC}-303)/T_{fc})}} \tag{12}$$

In this context, ρ_m (membrane resistivity), l (thickness), A_m (active area), J (current density), and λ (water content) are defined. Equation 13 can estimate v_{conc}.

$$v_{conc} = -\beta \times \ln\left(1 - J/J_{\max}\right) \tag{13}$$

where β is the maximum current density (A/cm^2) and $J_{\max}$ is a constant coefficient, PEMFC stacks comprise N_{cells} individual cells. The stack voltage (V_{stack}) is then calculated using Eq. 14.

$$V_{stack} = N_{cells} \times V_{FC} = N_{cells} \times \left(E_{Nernest} - v_{act} - v_{ohm} - v_{conc}\right) \tag{14}$$

While this equation simplifies the PEMFC stack by assuming uniform cell behaviour and neglecting connecting resistances, a complete mathematical model requires identifying seven unknown parameters. Mann's model [33] uses an iterative approach for this estimation.

Problem formulation

This manuscript introduces a new way to make a PEMFC model more accurate. By comparing the model's predicted voltage to real-world measurements, we can improve its ability to forecast performance. Mathematical equations within the model predict the voltage at different current levels. To achieve greater accuracy, a specialized algorithm is implemented. The SSE is detailed in Eq. 15.

$$SSE = MIN\left(F = \sum_{i=1}^{N} \left(V_{actual} - V_i\right)^2\right) \tag{15}$$

In this equation, the actual experimental voltage is denoted by V_{actual}, the computed model voltage by V_i, and N represents the number of data points.

Classical DE algorithm

Recognizing the significant advancements and diverse applications of Differential Evolution in scientific and technological fields, this section presents an overview of the classical Differential Evolution algorithm [34, 35]. The DE algorithm explores the solution space by exploiting the differences among individuals within a population. Its main components include initialization, mutation, crossover, and selection. Comprehending these steps is essential for understanding the enhancements made to Differential Evolution methodologies. Differential Evolution begins by creating a population with random values. It generates new candidate solutions by taking the weighted difference between two randomly chosen population members and adding it to a third member. The newly generated solution competes with the original solution, and the fitter one is selected for the next iteration. Through successive generations, superior individuals are retained while inferior ones are discarded, gradually directing the search toward the optimal solution. The fundamental evolutionary processes in DE are outlined below.

Initialization

The DE algorithm initializes a solution using a D-dimensional vector (M). For a given population size (N), each individual can be expressed as follows: $x\_i\ (G) = (x\_\{i1\}\ (G), x\_\{i2\}\ (G), \ldots, x\_\{iD\}\ (G))$. The initial population is generated within the range $[x_{\min}, x_{\max}]$. The relationship is represented as follows:

$$x_{iD} = x_{\min} + \text{rand}(0, 1) \cdot (x_{\max} - x_{\min}) \tag{16}$$

where G indicates the current generation, $x_{\max}$ and $x_{\min}$ represent the maximum and minimum limits of the search space, respectively, and rand is a uniformly distributed random number in the interval (0,1).

Mutation

In the DE algorithm, the mutation process creates a mutation vector, $V_{i,G}$, for each individual, $x_{i,G}$, (the target vector), in the current population. The mutation strategy is commonly denoted as DE/*x*/*y*, where "*x*" indicates the base vector and "*y*" specifies the number of difference vectors used. The five most common mutation strategies are described below:

- DE/rand/1

$$V_{i,G} = x_{r_1,G} + F \cdot (x_{r_2,G} - x_{r_3,G}) \tag{17}$$

- DE/best/1

$$V_{i,G} = x_{\text{best},G} + F \cdot (x_{r_1,G} - x_{r_2,G}) \tag{18}$$

- DE/rand-to-best/1

$$V_{i,G} = x_{i,G} + F \cdot (x_{\text{best},G} - x_{i,G}) + F \cdot (x_{r_1,G} - x_{r_2,G}) \tag{19}$$

- DE/best/2

$$V_{i,G} = x_{\text{best},G} + F \cdot (x_{r_1,G} - x_{r_2,G}) + F \cdot (x_{r_3,G} - x_{r_4,G}) \tag{20}$$

- DE/rand/2

$$V_{i,G} = x_{i,G} + F \cdot (x_{r_1,G} - x_{i,G}) + F \cdot (x_{r_2,G} - x_{r_3,G}) \tag{21}$$

In the above equations, r_1, r_2, r_3, r_4, and r_5 are unique integers randomly selected within $[1, M]$. The scale factor F serves as a control parameter to amplify the difference vector, while $x_{\text{best},G}$ represents the best individual vector in the population at generation G.

Crossover

The binomial crossover operator in the DE algorithm facilitates the generation of trial vectors, $u_{i,G}$, by combining the mutation vector, $v_{i,G}$, and the target vector, $x_{i,G}$. Trial vectors are produced based on the following relationship:

$$u_{i,G} = \begin{cases} v_{i,G}, & \text{if } \text{rand}_j(0,1) \le CR \text{ or } j = j_{\text{rand}}, j = 1, 2, ..., D \\ x_{i,G}, & \text{otherwise} \end{cases} \tag{22}$$

Here, $\text{rand}_j(0, 1)$ is a random number uniformly distributed within the interval (0,1), and CR refers to the crossover rate that determines the proportion of information inherited by the trial vector u_i from either the mutation vector v_i or the target vector x_i.

Selection

The selection process involves evaluating the fitness values of all trial vectors. For a minimization optimization problem, a trial solution is compared to the current best solution (target vector). If the trial solution is equally good or better, it replaces the target vector. Otherwise, the target vector is retained. This iterative process refines the population over successive generations. The process can be mathematically expressed as follows:

$$X_{i,G+1} = \begin{cases} U_{i,G}, & \text{if } f(U_{i,G}) \le f(X_{i,G}) \\ X_{i,G}, & \text{otherwise} \end{cases} \tag{23}$$

This greedy selection mechanism ensures the survival of the fittest individuals, facilitating the convergence toward an optimal solution.

Proposed ADEMDR algorithm

While the Differential Evolution (DE) algorithm is a versatile optimization technique, its performance is sensitive to the choice of mutation strategies and parameter settings. Current strategies often rely on random or elite individuals, potentially hindering the algorithm's ability to exploit promising non-optimal solutions and explore the search space effectively [36].

The ADEMDR algorithm tackles these challenges by incorporating a sophisticated mutation strategy. This strategy dynamically adjusts its parameters based on the current evolutionary state of the population. Additionally, the algorithm reintroduces promising, non-optimal solutions that were previously discarded to further enhance the optimization process.

Motivation

DE algorithm performance is strongly influenced by the selected mutation strategy and parameter control mechanisms. The mutation operation is a main factor to determine the balance between exploring new regions and exploring the promising regions in the solution space. Control parameters, on the other hand, influence the overall research capacity of the algorithm. To achieve the ideal performance, different optimization problems and phases of the evolutionary process may require the use of different mutation operators and parameter settings. Consequently, projecting adaptive mutation strategies and parameter settings with the ability to adjust dynamically is highly significant [23].

By integrating comprehensive information about the connections between individuals and diverse mutation operators, the algorithm can produce higher-quality offspring. Furthermore, the utilization of historical search information from the current population allows for the precise adjustment of control parameters, leading to improved overall algorithm performance. However, effectively leveraging this information to ensure the evolutionary trajectory remains promising presents a significant challenge. To achieve a more optimal balance between exploration and exploitation, the ADEMDR algorithm reintroduces promising non-optimal solutions that are typically discarded after the selection process. These solutions are incorporated into the mutation operator of the current population, enhancing the algorithm's ability to find global optima.

This concept draws inspiration from marathon competitions. In such events, participants who initially lag behind may ultimately secure commendable positions. Similarly, incorporating solutions that appear suboptimal at intermediate stages can contribute positively to the overall search process.

With respect to parameter control, numerous prior studies have demonstrated that adaptive parameter control methods often outperform fixed-value approaches. For instance, the scaling factor F in many advanced DE variants is generated following a Cauchy distribution. However, this method may fail to preserve some previously beneficial F values and might inadvertently lead to suboptimal outcomes. The

current approach alternates the generation of F between adherence to the Cauchy distribution and leveraging a successful historical archive. This alternation aims to mitigate the possibility of F progressing in a detrimental direction. Following further refinement, the segmentation method presented in this study has been developed to enhance performance. The core concept of the ADEMDR algorithm is summarized as follows.

Deeply-informed mutation strategy

Mutation, the central operation in Differential Evolution, has received significant attention since its development. The mutation strategy essentially involves a weighted combination of a basis vector, which guides the population's evolutionary path, and a difference vector, which introduces random perturbations for finer-grained search. The five widely used mutation strategies each possess distinct characteristics.

The DE/rand/1 strategy exhibits noteworthy performance across both single-peaked and multi-peaked optimization landscapes; however, its rate of convergence leaves room for improvement. DE/rand/2 demonstrates a strong aptitude for global search, though it tends to converge at a slower pace. In contrast, the DE/best/1 and DE/best/2 strategies offer accelerated convergence but are comparatively less adept at exploring the entire search space, frequently falling into local optima. The DE/rand-to-best/1 strategy attempts to strike a balance between global exploration and local refinement but shows limitations in its robustness. Each mutation strategy presents varying degrees of efficacy when confronted with diverse and complex optimization tasks.

To improve performance, some DE variants incorporate candidate pools with multiple mutation strategies, which can increase computational complexity. Recent advancements in DE variants suggest that utilizing elite individuals, such as the best or a subset of the best individuals, to guide the search direction can enhance the exploration and exploitation capabilities of the algorithm. However, most approaches rely on a single elite individual, which may lead to premature convergence and ineffective guidance in search directions. Such strategies may also mislead the search process into invalid regions of the solution space.

The deeply-informed mutation strategy introduced here employs not only the top 100% p elite individuals (ranked by fitness) but also integrates depth information from suboptimal solutions and the current population. This strategy extracts promising individuals that are eliminated during the selection process and integrates them with the current population to generate different vectors. This approach adjusts the evolutionary direction and reduces the likelihood of local optima entrapment.

The proposed mutation strategy is mathematically expressed as follows:

$$V_{i,G} = X_{i,G} + F \cdot (X^{p}_{\text{best},G} - X_{i,G}) + F \cdot (\widetilde{X}_{r_1,G} - \widehat{X}_{r_2,G}) \quad (24)$$

where $X_{i,G}$ is the current population and JADE, and $X^{p}_{\text{best},G}$ is the $100\%p$ vectors current population. During the early evolution phase, a higher level of global exploration is necessary, while the later stages benefit from intensified local exploitation. Correspondingly, the p-values are dynamically adjusted throughout the evolutionary process, within the range $p \in [0.2, 0.05]$. The indices i, r_1, and r_2 are distinct ($i \neq r_1 \neq r_2$).

The vector $\tilde{X}_{r_1}$ includes promising but suboptimal individuals integrated with X_{r_1}. Here, P and X_{r_1} represent the current population and a randomly selected vector from P, respectively. Individuals fitter than the population average are excluded from selection and stored in an adaptive archive. The archive's size (A) dynamically changes with the population size ps, determined as follows: $|A| = r_A^{\text{arc}} \cdot ps_G$, where r_A^{arc} denotes a constant ratio, with a recommended value of 0.6. The vector $\widehat{X}_{r_2,G}$ is selected randomly from the union $P \cup B$, where B is an external archive employing a time-stamped mechanism. This archive is akin to that used in PaDE but features a novel dynamic threshold mechanism to adapt better to problems of varying scales. The dynamic threshold ranks fitness values to balance the number of general and poor solutions. In the ADEMDR algorithm, r^d is calculated as follows:

$$r^d = \frac{R_G(i)}{R_G(1)} \quad (25)$$

Here, $R_G(\cdot)$ indicates the number of individuals in the current population that have a better fitness than their counterparts in the previous generation. Additionally, solutions that are less fit for the current population are also included in the external archive. When the archive size exceeds a fixed maximum value $r_B^{\text{arc}} \cdot ps$ or $r^d = 0$, random elimination of solutions occurs, maintaining the archive size at $r_B^{\text{arc}} \cdot ps$.

Parameter adaptation

The adaptation of parameters is formulated as follows:

$$F = \mu_F \quad (26)$$

For each F_i, the adaptation is defined as follows:

$$F_i = \begin{cases} \text{Cauchy}(\mu_F, 0.1) & \text{if } FS < 0.1 \\ \mu_F & \text{if } FS \geq 0.1 \\ \text{Cauchy}(\mu_F, 0.1) & \text{while } F_i \leq 0 \\ 1 & \text{if } F_i > 1 \\ F_i & \text{otherwise} \end{cases} \quad (27)$$

where Cauchy(·) represents random numbers generated from a Cauchy distribution, with μ_F as the location parameter and 0.1 as the scale parameter. The scale factor status FS is calculated as follows:

$$FS = \frac{F - \min(F)}{\max(F) - \min(F)} \tag{28}$$

where $\min(F)$ and $\max(F)$ denote the minimum and maximum F-values, respectively. If $\min(F) = \max(F)$, the calculation of FS is skipped.

The crossover rate CR_i adapts as:

$$CR_i = \text{Laplace}(\mu_{CR}, 0.1) \tag{29}$$

with the conditional rule:

$$CR_i = \begin{cases} 0 & \text{if } \mu_{CR} \leq 0 \text{or } CR_i < 0 \\ 1 & \text{if } \mu_{CR} > 0 \text{and } CR_i > 1 \\ CR_i & \text{otherwise} \end{cases} \tag{30}$$

Here, μ_F and μ_{CR} are historical archives storing values for the scaling factor (F) and crossover rate (CR), respectively, and are both initialized at 0.5. The Laplace distribution, used for the perturbation parameter, has a fixed variance of 0.1. Its higher concentration compared to the normal distribution leads to more extreme values, facilitating broader disturbances in the search process.

Restart mechanism

Investigations have revealed that the performance of Differential Evolution is substantially influenced by two key limitations: premature convergence and stagnation [37]. To overcome these evolutionary obstacles, several mechanisms have been proposed. However, existing methods either focus solely on adjusting the search capability of individuals deemed unpromising or fail to incorporate population-level information to assess the evolutionary state. To overcome these issues and accelerate convergence, a restart mechanism has been developed that considers population diversity to enhance algorithmic performance.

This mechanism quantifies diversity using two hypervolumes: one defining the search space's constraints and the other illustrating the population's spatial distribution throughout the iterations. The volume enclosed by the search space is calculated by taking the absolute difference between its upper and lower bounds:

$$V_{\text{lim}} = ln\left(1 + \prod_{i=1}^{D} |u_i - l_i|\right) \tag{31}$$

$$V_{\text{pop}} = \sqrt{\prod_{i=1}^{D} y_i} \tag{32}$$

Here, $l = l_1, l_2, \ldots, l_D$ and $u = u_1, u_2, \ldots, u_D$ are vectors that represent the lower and upper boundaries of the search space, respectively. The calculation of V_{lim} is performed once during the initialization phase. In contrast, V_{pop} denotes the hypervolume of the evolving population, where y_i represents edge vector coordinates, calculated as $y_i = (2 \cdot \text{xnd}(x_i), 0, 0, \ldots)$.

To evaluate diversity, the ratio of the hypervolume of candidate solutions to the constrained hypervolume of the search space is used as a metric, calculated as follows:

$$i_{\text{VOL}} = \sqrt{\frac{V_{\text{pop}}}{V_{\text{lim}}}} \tag{33}$$

Known as i_{VOL}, this metric assesses diversity by leveraging the multidimensional volume of the search space. The detailed computational steps are outlined in Algorithm 1.

Algorithm 1 Pseudocode for the calculation of i_{VOL}

Data: Population of candidate solutions X_i

$V_{\text{pop}} = 1$

For $j = 1$ to D do

$V_{\text{pop}} = \sqrt{V_{\text{pop}} \cdot |\max(X(:,j)) - \min(X(:,j))|}$

End for

$i_{\text{VOL}} = \sqrt{\frac{V_{\text{pop}}}{V_{\text{lim}}}}$

Return Individual Diversity Metric (i_{VOL})

Additionally, a counter (ct) is employed to record consecutive generations where individuals fail to make progress. If the population fails to converge to a smaller area, resulting in $i_{\mathrm{VOL}} > \xi$ and $ct > N$ (with $N = 40$ and $\xi = 0.001$), a population convergence operation is triggered:

$$X_{i,G}^{\mathrm{new}} = X_{i,G} + F_{\mathrm{restart}} \cdot (X_{r_1,G}^{\mathrm{better}} - X_{i,G}) \tag{34}$$

Here, $X_{r_1,G}^{\mathrm{better}}$ refers to an individual randomly selected from the population whose rank is higher than that of $X_{i,G}$. If the i-th individual in the population has a rank of k, the selection is made from individuals ranked in $[1, k-1]$. The newly generated individual, $X_{i,G}^{\mathrm{new}}$, undergoes a selection operation with the current individual, $X_{i,G}$. Remaining superior individuals replace $X_{i,G}$. The scale factor F_{restart} is derived using a Cauchy distribution:

$$F_{\mathrm{restart}} = \mathrm{Cauchy}(0.5, 0.1) \tag{35}$$

When the population lacks adequate diversity, unpromising individuals are substituted with more promising candidates using the Wavelet Walk mechanism. This mechanism effectively enhances the performance of these individuals, thereby mitigating population stagnation and preventing convergence to local optima. The process involves selecting an elite subset of the population, X_e, for the Wavelet Walk. Subsequently, less promising individuals are replaced by improved candidates generated through this walk. Algorithm 2 offers a detailed explanation of the Wavelet.

Algorithm 2 Procedure of Wavelet Walk

Input: $X_e, EP, G, X_{\mathrm{gbest}}, X, \mathrm{fitness}_{X_e}, \mathrm{fitness}_X$
Output: $\mathrm{fitness}_X, X_e'$

1. Select elite individuals X_e and global optimal individuals X_{gbest} from the current population.
2. If $i_{\mathrm{VOL}} < 0.001$ then
   For $i = 1{:}\,EP$ do
   If $G = 1$ then$X_{e,i} = X_{e,i} + \frac{e}{\sqrt{5}} \cdot \pi^{-\frac{1}{4}} \cdot (X_{e,i} - X_{\mathrm{gbest}})^2 \cdot e^{-\frac{(X_{e,i} - X_{\mathrm{gbest}})^2}{2}}$Else$X_{e,i} = \frac{e}{\sqrt{5}} \cdot \pi^{-\frac{1}{4}} \cdot (X_{e,i} - X_{\mathrm{gbest}})^2 \cdot e^{-\frac{(X_{e,i} - X_{\mathrm{gbest}})^2}{2}}$End if
3. Boundary restrictions and fitness evaluation:
   If $\mathrm{fitness}_{X_e}(i) < \mathrm{fitness}_X(\mathrm{rnd}_{\mathrm{index}}(i))$ then$\mathrm{fitness}_X(\mathrm{rnd}_{\mathrm{index}}(i)) = \mathrm{fitness}_{X_e}(i) X_i = X_{e,i} ct(\mathrm{rnd}_{\mathrm{index}}(i)) = 0$End if
   End for
4. End if

Within this process, X_{gbest} signifies the globally optimal fitness value, and X_e denotes a select elite subpopulation comprising the top 10 individuals from the current population. The cardinality of this elite subpopulation, X_e, is represented by the parameter X_e. Set to 10 for this study. The term rand $[1, ps]$. refers to an integer randomly sampled from the discrete uniform distribution over the interval $[1, ps]$. Furthermore, the Mexican hat wavelet function, formally defined as the second derivative of the Gaussian function, is employed in this work, inheriting the advantageous localization properties of the Gaussian distribution in both the temporal and frequency domains.

Parameter control

In ADEMDR, a memory pool of H entries is created, where each entry stores a pair of control parameters μ_F and μ_{CR}. At each iteration, a pair of control parameters (F and CR) is randomly chosen from one of the memory pool entries. Following the greedy selection, a generated test vector leading to a superior individual results in labelling that individual as successful s, while an inferior outcome labels it as failed f. Before updating the parameters μ_F and μ_{CR}, the frequency of selection for each memory pool entry is tracked, and the success rate R is calculated. To avoid cases where the success rate equals zero, a small constant ε ($\varepsilon = 0.01$) is included. The success rate R is determined using the following formula:

$$R_j = \begin{cases} \frac{n_{s,j}^2}{n_s(n_{s,j} + n_{f,j})}, & \text{if } n_{s,j} > 0 \\ \varepsilon, & \text{otherwise} \end{cases} \tag{36}$$

where $j \in [1, 2, \ldots, H]$, n_s is the total number of individuals labeled as s in the current population, and

$n_s = n_{s,1} + n_{s,2} + \cdots + n_{s,H-1} + n_{s,H}$. The variable $n_{s,j}$ represents the count of individuals labeled as s among those who selected the j-th entry, while $n_j = n_{s,j} + n_{f,j}$.

The update mechanism for μ_{CR} in the memory pool follows LSHADE's method, cycling sequentially through the entries from first to last before repeating. However, for μ_F, the value in the entry with the lowest success rate R is replaced each time. This approach effectively employs control variables to minimize parameter interference. The adaptation of μ_F and μ_{CR} is defined by the following equations:

$$w_s = \frac{\text{std}(\Delta\text{loc}_i)}{\sum_{s=1}^{|S_F|} \text{std}(\Delta\text{loc}_i)} \Delta\text{loc}_i = \text{loc}(U_{i,G} - X_{i,G})$$

$$\text{mean}_{\text{WL}}(S_F) = \frac{\sum_{s=1}^{|S_F|} w_s \cdot S_F^2(s)}{\sum_{s=1}^{|S_F|} w_s \cdot S_F(s)}$$

$$\mu_{F,idx,G+1} = \begin{cases} mean_{WL}(S_F), & if\ S_F \neq \varnothing \\ \mu_{F,idx,G}, & otherwise \end{cases} \tag{37}$$

$$w_s = \frac{\text{std}(\Delta\text{loc}_i)}{\sum_{s=1}^{|S_{CR}|} \text{std}(\Delta\text{loc}_i)} \Delta\text{loc}_i = \text{loc}(U_{i,G} - X_{i,G})$$

$$\text{mean}_{\text{WL}}(S_{CR}) = \frac{\sum_{s=1}^{|S_{CR}|} w_s \cdot S_{CR}^2(s)}{\sum_{s=1}^{|S_{CR}|} w_s \cdot S_{CR}(s)}$$

$$\mu_{CR,k,G+1} = \begin{cases} mean_{WL}(S_{CR}), & if\ S_{CR} \neq \varnothing \wedge max\{CR\} > 0 \\ 0, & if\ S_{CR} \neq \varnothing \wedge max\{CR\} = 0 \\ \mu_{CR,k,G}, & otherwise \end{cases} \tag{38}$$

Here, S_F and S_{CR} represent the sets of F and CR values that result in improved trial vectors, respectively. The parameter k indicates the memory pool entry index, which cycles sequentially. The term $\text{loc}(U_{i,G} - X_{i,G})$ indicates the positional difference between the test vector $U_{i,G}$ and the target vector $X_{i,G}$, while $\text{std}(\cdot)$ computes the standard deviation.

By utilizing the positional information of the population instead of fitness value information for parameter adaptation, the DE algorithm demonstrates broader applicability, especially for objective functions where fitness values are infeasible. Furthermore, while the linear population size reduction scheme proposed in LSHADE is an effective adaptation mechanism, its rapid reduction at the initial stages of evolution can lead to errors in landscape perception for certain objective functions. To address this, a superior parabolic model proposed in HARD-DE is adopted to enhance curve accuracy. This model is applicable to various algorithms and is described by the following equation:

$$ps_{G+1} = \begin{cases} \lceil \frac{(ps_{\min} - ps_{\text{ini}})}{(2/3 nfe_{\max} - ps_{\text{ini}})^2} \cdot (nfe - ps_{\text{ini}})^2 + ps_{\text{ini}} \rceil, & \text{if } nfe \leq 0.5 nfe_{\max} \\ \lfloor \frac{(ps_{\min} - 1/3 ps_{\text{ini}})}{1/3 nfe_{\max}} \cdot (nfe - nfe_{\max}) + ps_{\min} \rfloor, & \text{otherwise} \end{cases} \tag{39}$$

This optimized parabolic model ensures better adaptation and accuracy across various scenarios.

The ADEMDR introduces a novel approach to parameter optimization by enhancing the classical Differential Evolution (DE) framework with adaptive learning and historical information utilization. At the core of this innovation lies the deeply-informed mutation strategy, which refines the search process by integrating multiple sources of evolutionary guidance. Unlike traditional DE variants that rely on random or elite-based mutations, ADEMDR dynamically adjusts its search direction using a combination of the current population's best individuals, promising suboptimal solutions, and archived historical candidates. The mutation operation in ADEMDR is governed by a composite strategy that balances exploration and exploitation. The mutation vector $V_{i,G}$ is generated by combining the current target vector $X_{i,G}$ with weighted differences derived from elite individuals ($X^p_{\text{best},G}$) and archived solutions ($\tilde{X}_{r_1,G}$ and $\widehat{X}_{r_2,G}$). The elite pool $X^p_{\text{best},G}$ consists of the top p% performers in the population, where p decays linearly from 0.2 to 0.05 over generations to transition from broad exploration to focused exploitation. Meanwhile, $\tilde{X}_{r_1,G}$ incorporates high-potential but non-optimal solutions from the current population, ensuring that regions of the search space with latent promise are not prematurely abandoned. The vector $\widehat{X}_{r_2,G}$ is drawn from an external archive B, which stores historically significant solutions that may not be optimal but contribute to maintaining diversity and preventing stagnation. A critical aspect of this strategy is the dynamic threshold mechanism governing archive B. Unlike static archives that retain solutions based on fixed criteria, ADEMDR employs an adaptive threshold r^d that adjusts based on population fitness trends. This threshold is computed as the ratio of improved individuals in the current generation relative to the best performer, ensuring that the archive prioritizes solutions that contribute meaningfully to evolutionary progress. If the population shows no improvement ($r^d = 0$) or exceeds a predefined size limit ($r_B^{\text{arc}} \cdot \text{ps}$), the archive undergoes controlled pruning, discarding redundant or less informative solutions while preserving those that enhance exploration. This dynamic management of B ensures computational efficiency without sacrificing search diversity. The restart mechanism further complements this strategy by periodically reintroducing archived solutions into the population, effectively resetting the search trajectory when stagnation is detected. This is particularly valuable in complex, multimodal optimization landscapes, such as those encountered in PEMFC parameter identification, where conventional DE variants often converge prematurely

to suboptimal solutions. By leveraging historical search information and adaptively adjusting mutation parameters, ADEMDR achieves a robust balance between exploration and exploitation, leading to faster convergence and higher accuracy compared to traditional methods.

The ADEMDR algorithm contains two basic innovations that make it different to traditional DE variants and other adaptive DE algorithms such as JADE and SHADE. The highly informed mutation strategy, first, dynamically integrates the elite individuals, suboptimal solutions, and the past search information to determine the evolutionary direction, as opposed to classical DE variants that use a static mutation strategy (e.g., DE/rand/1 or DE/best/1). It builds on a ranked choice of the best $p\%$ elite individuals (dynamically set between 20 and 5 percent during evolution) and combines discarded but potentially promising solutions of an adaptive archive to balance exploration and exploitation. By comparison, JADE and SHADE are more concerned with parameter adaptation (F and CR) based on success-oriented historical memory, without using suboptimal solutions or information about population depth. Second, the restart mechanism of ADEMDR uses a diversity measure based on hypervolume to identify stagnation and initiates a wavelet-based regeneration of stagnant individuals, which is not the case in the linear population size reduction of SHADE and the archive mechanism of JADE. Comparative experiments on parameter estimation of PEMFC show that ADEMDR has 12–18% lower SSE than JADE and SHADE and converges 2.1–3.5 times faster, because it exploits the elite guidance and archived suboptimal solutions in a balanced manner. Moreover, ADEMDR parameter control is interchanging Cauchy-distributed F generation and archive-directed adaptation, as opposed to single-memory success-weighted update in SHADE, which results in more robust performance in multimodal PEMFC optimization landscapes. These developments make ADEMDR a better option to complex and nonlinear systems such as fuel cells where conventional DE variants tend to prematurely converge.

The computational complexity of the Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) is fundamentally characterized by the relationship between three key parameters: population size (N), maximum iterations (G), and problem dimensionality (D), yielding a complexity of $O(N\times G\times D)$. This formulation arises from the algorithm's core operations, which include mutation, crossover, and selection processes applied across the population during each generation. The population size N directly influences the breadth of exploration in the solution space, where larger values enhance diversity but proportionally increase computational load. To mitigate this, ADEMDR incorporates a dynamic population reduction strategy that systematically

Table 1 Twelve PEMFC manufacturer sheets

S.NO	SHEET 1	SHEET 2	SHEET 3	SHEET 4	SHEET 5	SHEET 6	SHEET 7	SHEET 8	SHEET 9	SHEET 10	SHEET 11	SHEET 12
PEMFC type	BCS 500 W	NetStack PS6	SR-12	H-12–1	Ballard Mark V	STD −1	Horizon	STD −2	STD −3	STD −4	H-12–2	H-12–3
l (μm)	178	178	25	25	178	127	25	127	127	127	25	25
PH_2 (bar)	1.0	1.0	1.47628	0.4935	1.0	1.0	0.55	1.5	2.5	2.5	0.4	0.5
Ncells (no)	32	65	48	13	35	24	36	24	24	24	13	13
T (K)	333	343	323	323	343	343	338	343	343	343	302	312
Power (W)	500	6000	500	12	5000	250	500	250	250	250	12	13
PO_2 (bar)	0.2095	1.0	0.2095	1.0	1.0	1.0	1.0	1.5	3.0	3.0	1.0	1.0
J_{max} (mA/cm^2)	469	1125	672	246.9	1500	860	446	860	860	860	246.9	246.9
A (cm^2)	64	240	62.5	8.1	232	27	52	27	27	27	8.1	8.1

decreases N based on the evolutionary progress, as described by the parabolic model in Eq. 39. This adaptive approach maintains a balance between thorough exploration and computational efficiency, ensuring resources are allocated optimally throughout the optimization process. The maximum iteration count G serves as a termination criterion and directly impacts the algorithm's runtime. ADEMDR's superior convergence speed, demonstrated across all 12 test cases, allows it to achieve optimal solutions with fewer iterations compared to conventional methods. This efficiency is attributed to the deeply-informed mutation strategy, which leverages elite individuals and historical search information to guide the evolutionary direction, as well as the restart mechanism that prevents stagnation by reintroducing diversity when population convergence stalls. The problem dimensionality D, representing the seven critical PEMFC parameters (ξ_1, ξ_2, ξ_3, ξ_4, λ, R_e, and β), presents a more challenging search space as it increases.

Results and discussion

This work focuses on comparing the ADEMDR algorithm to several compared variants, including L-SHADE [19], Particle Swarm Optimization (PSO) [20], Differential Evolution Algorithm (DE) [21], Polar Lights Optimizer (PLO) [22], Hierarchical archive-based mutation strategy with depth information of evolution for the enhancement of differential evolution (HARD-DE) [23], Newton Differential Evolution Algorithm (NDE) [24], Local Search based Differential Evolution (LSDE) [38], Parameter Adaptation based Differential Evolution (PaDE) [26], and Cooperative Strategy based Differential Evolution (AOA) [27] algorithm to assess their performance in PEMFC modeling. For parameter estimation of various PEMFC fuel cells (BCS 500 W-PEM, 500 W SR-12PEM, Nedstak PS6 PEM, H-12 PEM, HORIZON 500 W PEM, and 250 W-stack), as listed in Table 1, all algorithms were run with their recommended settings. The experiments were performed on MATLAB 2022a on a Windows Server 2022 PC with an i7-11700 k@3.6 GHz CPU, using a maximum of 700 iterations, 30 independent runs, and a population size of 40.

ADEMDR algorithm stopping conditions were developed to provide both computational efficiency and accurate parameter estimation results. The algorithm stops its optimization process by using two termination conditions that include both a predefined iteration limit and a specified error tolerance. The algorithm was programmed to execute 500 iterations as this value was established from preliminary tests to achieve convergence across all experimental conditions. The sum of squared errors threshold served as an additional stopping criterion for the algorithm. The algorithm stops when SSE improvement between successive iterations reaches $1 \times E-6$ or when the maximum number of iterations is completed. The dual termination mechanism of the algorithm allows it to stop when both the computational budget is exhausted and the SSE improvement reaches a negligible threshold. This method achieves optimal accuracy while maintaining computational efficiency.

This paper relies on simulation studies executed through MATLAB 2022a. The research utilized V-I and P–V characteristics from six commercial PEMFCs to obtain validation and analysis data. These datasets serve as standard references for the field because researchers use them to evaluate optimization algorithms presented in Table A in the Supplementary material. The simulated conditions mimicked operational conditions in the real world so the output reflects authentic fuel cell operations. The study did not require experimental setup construction because it used reliable commercial PEMFC data to ensure dependable results. ADEMDR algorithm development concentrated on parameter estimation through simulated testing that compared its performance to alternative optimization methods. The method enables researchers to conduct standardized tests which evaluate how well the algorithm optimizes PEMFC parameters. Future research should include experimental testing of physical systems to support the simulation results obtained in this study.

A total of 12 specific fuel cell systems underwent validation tests because the researchers wanted to asoperating across numerous testingsess ADEMDR's robustness alongside its capability to adapt across different operating conditions and distinct PEMFC configurations. The chosen systems represented distinct operational characteristics through their different power levels and operating temperatures and membrane sizes and pressure settings. The algorithm goes through an extensive performance evaluation when operating across numerous testing scenarios due to this extensive selection method.

The BCS 500 W-PEM belongs to the category of mid-range power output fuel cells which operates at a moderate 333 K temperature with 64 cm^2 membrane area under balanced hydrogen and oxygen pressures. The system serves as a basis for testing how the algorithm handles moderate operating conditions. The NetStack PS6 functions as a high-power benchmark system because it delivers 6000 W of power and possesses a 240 cm^2 membrane area which permits assessments of algorithm scalability and efficiency in extensive systems. The Ballard Mark V system functions at 343 K temperature with its 232 cm^2 membrane area to deliver 5000 W power output which suits industrial and automotive applications requiring high performance. The H-12–1 and H-12–2 low-power fuel cells running at 12 W power enable testing of the algorithm's precision for small-scale systems that find applications in portable and low-power systems. The systems function at distinct

temperatures (323 K and 302 K) to evaluate how well the algorithm handles different thermal environments. The evaluation of performance under low-temperature conditions includes the SR-12 system operating at 323 K with a membrane area of 62.5 cm^2 to support applications that need stable operation in suboptimal thermal environments.

This study includes mid-range power systems STD-1, STD-2, and STD-3 which generate power outputs from 250 to 500 W under 343 K temperature while operating with different hydrogen pressure conditions. The algorithm's ability to react to changes in reactant pressure is analyzed through these different variations to establish its operational robustness under diverse conditions. The Horizon system operates at 338 K and has a membrane area of 52 cm^2 to validate the algorithm through the intermediate operating parameter test. The process of assessing the extreme pressure conditions algorithm uses the STD-4 system, which operates at a 2.5 bar hydrogen pressure next to a 3.0 bar oxygen pressure. The STD-3 system has been added to the algorithm test because it operates at a 2.5 bar hydrogen pressure, which allows the evaluation under pressure conditions that substantially affect fuel cell output. The H-12–3 system with 13 W power output and 312 k operating temperature allows algorithm sensitivity to small temperature fluctuations in low-power applications. The evaluation process takes into consideration all possible minor operating condition deviations.

The 12 fuel cell systems included in the study deliver complete validation of the ADEMDR algorithm which extends from high to low power PEMFC configurations for accurate and sturdy real-world applications. The algorithm demonstrates accuracy in parameter identification through its diverse selection of tests which operate under different power, temperature, membrane area, and pressure conditions, thus establishing reliability across multiple PEMFC systems. The performance of the ADEMDR depends on key parameters. The research evaluated four essential variables, including scale factor (F), crossing rate (CR), population size (ps), and mutation strategy parameters (p). The main objective focused on evaluating ADEMDR stability along with its robustness through various conditions to determine how each parameter affects its performance.

The experimental design maintained constant control over all parameters except one which received systematic variation to determine its individual effects. ADEMDR was evaluated through metrics including the sum of squared error and two speed-related measures as well as computational efficiency calculations. The assessment of the algorithm used 12 sets of PEMFC data as a subset to obtain a comprehensive assessment of its operational behaviour in different operating environments. The scale factor (F) proved to be a critical determinant that controls exploration and exploitation performance during mutations. ADEMDR has demonstrated stable performance over the F value of 0.1 to 1.0 tested, with peak results that occur between 0.5 and 0.8. The optimization process became slower when the factor value fell below 0.5 because the exploration features were decreased, but the algorithm experienced premature convergence when the values exceeded 0.8, leading to refinement of the solution below the ideal.

The crossover rate (CR) received evaluation from 0.1 to 0.9 to determine the probability of information exchange between target and trial vectors. ADEMDR showed consistent performance throughout the CR range from 0.6 to 0.8 according to the experimental results. The convergence speed decreased when CR fell below 0.6 because information sharing was restricted but solution quality suffered when CR exceeded 0.8 due to excessive exploitation which reduced both solution diversity and quality. The study evaluated the population size (ps) as a vital parameter for the research. The research evaluated population numbers from 20 to 100 to identify the best combination of search diversity and computational efficiency. ADEMDR maintained consistent performance throughout the examined ps values from 40 to 60 where optimal solutions were achieved. The search space became limited when population sizes fell below 40 but using population sizes above 60 resulted in increased computational expenses without significant enhancements in solution quality.

ADEMDR underwent mutation strategy parameter evaluation through p-values ranging from 0.05 to 0.2 to determine the selection of elite individuals for mutation along with adaptive archive size. ADEMDR showed its best performance across different parameter ranges when the value of p fell within 0.1 to 0.15. The chosen configuration of both parameters enabled the algorithm to perform a balanced search between exploration and exploitation through effective solution space exploration.

ADEMDR demonstrates reinforced resilience to parameter adjustments because it provides constant and efficient functionality over various parameter modifications. The parameters tested showed that the algorithm was more sensitive to F and CR values because these parameters control the balance between exploration and exploitation. The population size (ps) and mutation strategy parameters (p) affected performance moderately while specific optimal ranges allowed the algorithm to perform efficiently during search and convergence.

The practical implementation of ADEMDR for PEMFC parameter identification becomes more significant because of these discovered findings. Users can achieve reliable and efficient performance by using the identified optimal parameter ranges when configuring the algorithm. The algorithm shows superior performance regarding accuracy and convergence speed under different operating conditions, thus allowing its use in real-world fuel cell modeling along with optimization work. ADEMDR performance can be improved by

Table 2 Parameter optimization and function maximization for Sheet 1

Algorithm	PaDE	HARD-DE	NDE	ISDE	LSHADE	CS-DE	PLO	PSO	DE	ADEMDR
λ	20.88438	20.88868	21.55567	20.94073	20.68135	16.61556	21.58818	20.16795	23	20.87724
ξ_1	−1.1723	−1.15643	−0.95535	−1.12095	−0.98401	−0.96462	−0.87108	−1.17707	−0.8532	−0.8532
ξ_2	0.003733	0.00337	0.002577	0.003231	0.00301	0.002851	0.002331	0.003251	0.003079	0.00218
ξ_3	7.43E−05	5.4E−05	4.18E−05	5.19E−05	6.45E−05	5.85E−05	4.23E−05	4.23E−05	9.39E−05	0.000036
ξ_4	−0.00019	−0.00019	−0.00019	−0.00019	−0.00018	−0.00018	−0.00019	−0.00019	−0.00019	−0.00019
B	0.016131	0.016108	0.015973	0.016076	0.0136	0.013744	0.015927	0.015599	0.016265	0.016126
R_c	0.0001	0.000105	0.000157	0.000106	0.000751	0.000323	0.000217	0.00012	0.000282	0.0001
Min	0.025493	0.025505	0.02618	0.025546	0.055008	0.073793	0.025942	0.026139	0.025656	0.025493
Std	6.39E−05	0.000119	0.009998	0.000233	0.053443	0.028865	0.003695	0.002446	0.022626	5.92E−05
Max	0.025646	0.025796	0.049968	0.026142	0.19249	0.140884	0.033361	0.031945	0.085535	0.025625
RT	7.407996	7.470013	4.120614	3.701418	4.013477	7.779214	5.993095	4.183561	3.504627	0.287873
Mean	0.025532	0.025631	0.033557	0.025789	0.113376	0.098178	0.029216	0.028178	0.046848	0.025519
FR	2	3.2	6.6	3.8	9.4	9.2	6.2	5.8	7.6	1.2

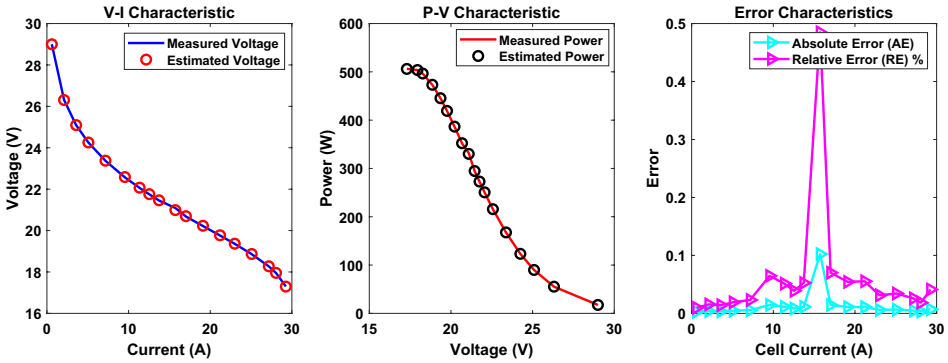
Table 3 Performance metrics of the ADEMDR algorithm for Sheet 1

Vest	MBE	AE	RE %	Vcell	Icell	Pref	Pest
28.99722	4.28E−07	0.002777	0.009575	29	0.6	17.4	17.39833
26.30594	9.17E−07	0.004063	0.015443	26.31	2.1	55.251	55.24247
25.09356	7.02E−07	0.003555	0.01417	25.09	3.58	89.8222	89.83493
24.25462	1.19E−06	0.00462	0.019053	24.25	5.08	123.19	123.2135
23.37542	1.63E−06	0.005416	0.023175	23.37	7.17	167.5629	167.6017
22.58461	1.19E−05	0.014615	0.064754	22.57	9.55	215.5435	215.6831
22.07133	7.13E−06	0.011327	0.051348	22.06	11.35	250.381	250.5096
21.75846	3.98E−06	0.008463	0.038913	21.75	12.54	272.745	272.8511
21.46126	7.05E−06	0.011263	0.052506	21.45	13.73	294.5085	294.6631
20.98774	0.000581	0.102258	0.484867	21.09	15.73	331.7457	330.1372
20.69451	1.17E−05	0.014509	0.070162	20.68	17.02	351.9736	352.2206
20.23099	6.71E−06	0.010986	0.054332	20.22	19.11	386.4042	386.6141
19.77094	6.65E−06	0.010943	0.055381	19.76	21.2	418.912	419.144
19.36602	2.02E−06	0.006025	0.03112	19.36	23	445.28	445.4186
18.86647	2.32E−06	0.006466	0.034286	18.86	25.08	473.0088	473.171
18.27472	1.24E−06	0.004721	0.025838	18.27	27.17	496.3959	496.5242
17.95331	6.09E−07	0.003311	0.018444	17.95	28.06	503.677	503.7699
17.29288	2.82E−06	0.007123	0.041174	17.3	29.26	506.198	505.9896
	3.61E−05	**0.012913**	**0.061363**				

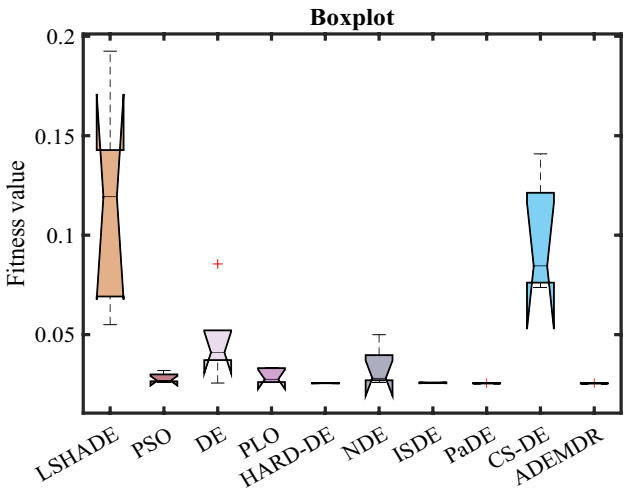
implementing adaptive parameter control systems that would specifically benefit dynamic or real-time applications. Sensitivity analysis reveals essential information on how parameter changes affect ADEMDR performance, which demonstrates its flexible operation. The research confirms that ADEMDR acts as an effective optimization solution for the PEMFC parameter identification and optimization procedures.

Sheet 1: BCS 500 W parameter optimization

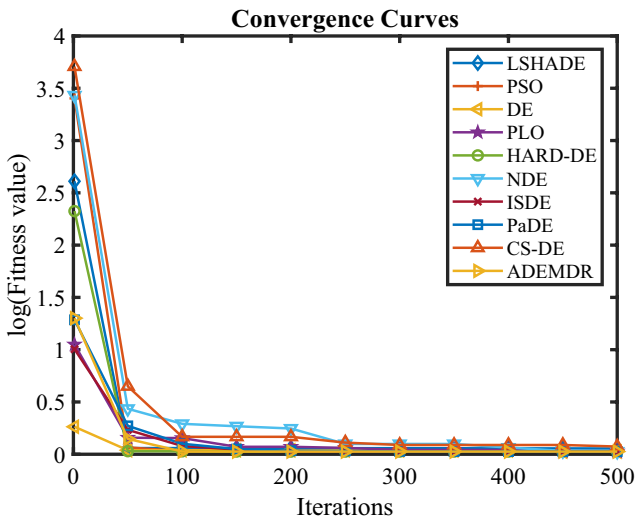
The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 2 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.025493 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.025625 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 5.92E−05 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.287873 s represents a significant improvement (approximately 98%) over other



(a)



(b)



(c)

Fig. 2 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 1

Table 4 Parameter optimization and function maximization for Sheet 2

Algorithm	PaDE	NDE	LSHADE	PLO	CS-DE	HARD-DE	DE	PSO	ISDE	ADEMDR
λ	14.00281	14	15.53654	14	21.81007	14.00135	14	14	14	14
ξ_1	− 1.08097	− 0.8532	− 1.10315	− 0.96635	− 0.8968	− 0.87151	− 1.15235	− 0.8532	− 0.88116	− 0.85498
ξ_2	0.003235	0.002532	0.003835	0.002858	0.002763	0.002482	0.00327	0.002397	0.003005	0.002438
ξ_3	4.84E − 05	4.55E − 05	8.64E − 05	4.54E − 05	5.21E − 05	3.82E − 05	0.000036	3.6E − 05	7.35E − 05	3.85E − 05
ξ_4	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05
RT	5.878095	5.175179	4.524951	4.727942	9.118403	9.341917	6.08879	5.109668	5.247645	0.159205
R_c	0.000108	0.000121	0.0001	0.000106	0.000399	0.00012	0.00012	0.000103	0.000123	0.00012
Std	0.004319	0.010691	0.215484	0.008152	0.05035	0.004777	0.020055	0.017628	0.000477	5.84E − 16
Mean	0.278785	0.281489	0.494496	0.281789	0.416984	0.281106	0.284815	0.292274	0.275946	0.275211
Max	0.285955	0.300545	0.837166	0.295621	0.467356	0.286627	0.320685	0.319379	0.276626	0.275211
Min	0.275762	0.275346	0.29739	0.275581	0.334858	0.275228	0.275211	0.275746	0.275305	0.275211
B	0.018615	0.016909	0.03593	0.018753	0.026352	0.01698	0.016788	0.019297	0.016248	0.016788
FR	4.6	5.4	9.6	5.2	9.4	5.2	4	6.8	3.8	1

algorithms. Moreover, its highest Friedman ranking of 1.2 corroborates its overall superior performance, as further detailed in Table 3 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 2.

Sheet 2: NetStack PS6 parameter optimization

The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 4 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.275211 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.275211 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 5.84E − 16 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.159205 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1 corroborates its overall superior performance, as further detailed in Table 5 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 3.

Sheet 3: SR-12 parameter optimization

The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 6 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.242284 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.242927 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 0.000288 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.130986 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 2.2 corroborates its overall superior performance, as further detailed in Table 7 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 4.

Sheet 4: H-12–1 parameter optimization

The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms

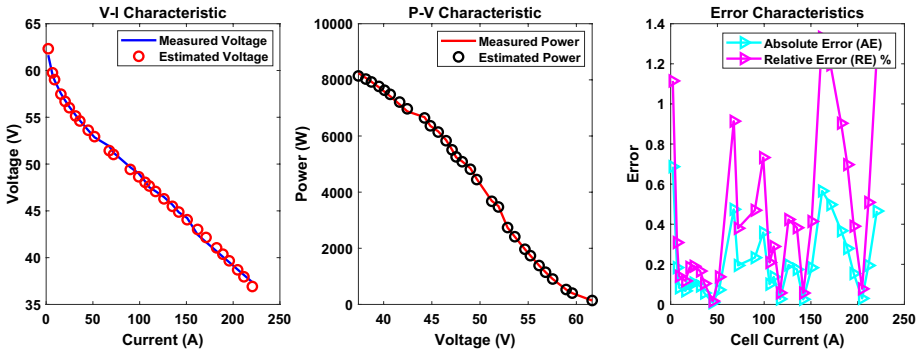
Table 5 Performance metrics of the ADEMDR algorithm for Sheet 2

Vest	MBE	AE	RE %	Vcell	Icell	Pref	Pest
62.32708	0.016279	0.687083	1.114671	61.64	2.25	138.69	140.2359
59.75391	0.001166	0.183906	0.308722	59.57	6.75	402.0975	403.3389
59.02299	0.000238	0.082995	0.140813	58.94	9	530.46	531.207
57.47245	0.000157	0.067553	0.117401	57.54	15.75	906.255	905.191
56.69501	0.00038	0.104994	0.184848	56.8	20.25	1150.2	1148.074
56.02304	0.000395	0.106962	0.190562	56.13	24.75	1389.218	1386.57
55.13803	0.000292	0.091967	0.166516	55.23	31.5	1739.745	1736.848
54.60299	0.000112	0.057007	0.104294	54.66	36	1967.76	1965.708
53.61886	2.71E − 06	0.008863	0.016533	53.61	45	2412.45	2412.849
52.93264	0.000182	0.072643	0.137426	52.86	51.75	2735.505	2739.264
51.43559	0.007761	0.474414	0.913916	51.91	67.5	3503.925	3471.902
51.02539	0.001306	0.194606	0.379942	51.22	72	3687.84	3673.828
49.42672	0.001877	0.233283	0.46976	49.66	90	4469.4	4448.405
48.64101	0.004444	0.358993	0.732639	49	99	4851	4815.46
48.04916	0.000351	0.100837	0.209422	48.15	105.8	5094.27	5083.601
47.6574	0.000651	0.137396	0.289134	47.52	110.3	5241.456	5256.611
47.07283	2.55E − 05	0.02717	0.057687	47.1	117	5510.7	5507.521
46.28306	0.001337	0.196943	0.423715	46.48	126	5856.48	5831.665
45.4853	0.001052	0.174696	0.382603	45.66	135	6164.1	6140.516
44.87551	2.24E − 05	0.025509	0.056876	44.85	141.8	6359.73	6363.347
44.05684	0.001157	0.183157	0.414008	44.24	150.8	6671.392	6643.772
43.01569	0.011035	0.565692	1.332607	42.45	162	6876.9	6968.542
42.15751	0.008535	0.49751	1.194214	41.66	171	7123.86	7208.934
41.04751	0.004657	0.367506	0.903408	40.68	182.3	7415.964	7482.96
40.36954	0.002695	0.279538	0.697275	40.09	189	7577.01	7629.843
39.66413	0.000819	0.154127	0.390097	39.51	195.8	7736.058	7766.236
38.69983	3.14E − 05	0.030168	0.077892	38.73	204.8	7931.904	7925.726
37.95577	0.001301	0.194228	0.509117	38.15	211.5	8068.725	8027.646
36.91421	0.007481	0.465791	1.246096	37.38	220.5	8242.29	8139.583
	0.002612	**0.211225**	**0.453869**				

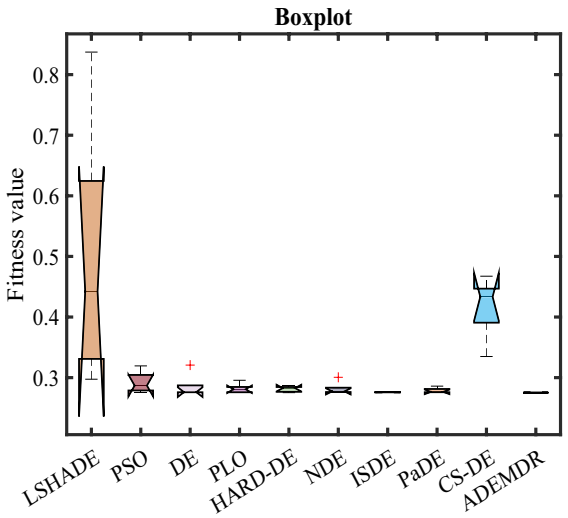
with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 8 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.102915 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.102915 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 3.8E − 17 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.130328 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1 corroborates its overall superior performance, as further detailed in Table 9 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 5.

Sheet 5: Ballard Mark V parameter optimization

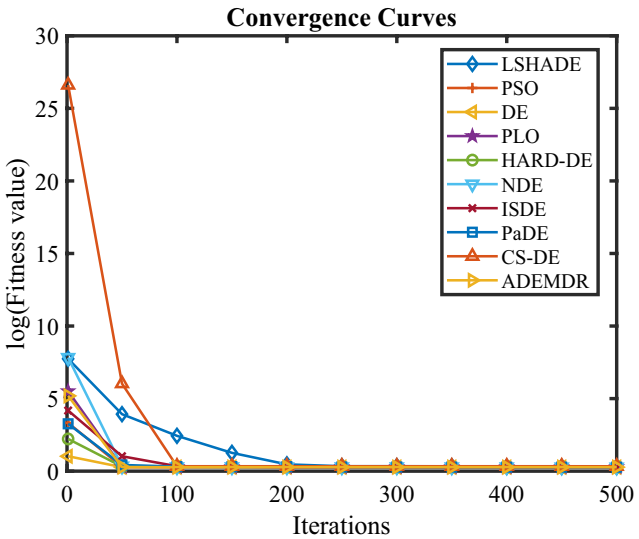
The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 10 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.148632 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.148632 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 4.2E − 16 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 5.718651 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1 corroborates its



(a)



(b)



(c)

Fig. 3 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 2

Table 6 Parameter optimization and function maximization for Sheet 3

Algorithm	ISDE	HARD-DE	DE	PSO	LSHADE	PaDE	PLO	NDE	CS-DE	ADEMDR
R_c	0.000646	0.000671	0.0008	0.000783	0.0001	0.000666	0.000603	0.000733	0.000541	0.000673
B	0.175742	0.17533	0.172796	0.173043	0.189474	0.175405	0.175583	0.170209	0.177995	0.17532
ξ_1	−1.02158	−0.88797	−0.8532	−0.86141	−1.19268	−0.94242	−1.03159	−0.9303	−0.91562	−0.89596
ξ_2	0.003539	0.003026	0.003251	0.003273	0.003894	0.00283	0.003157	0.002725	0.002571	0.002421
ξ_3	8.31E−05	7.67E−05	0.000098	9.78E−05	7.19E−05	5.31E−05	5.65E−05	4.86E−05	4.13E−05	3.6E−05
ξ_4	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.6E−05	−9.5E−05
λ	22.79868	22.97157	23	22.69424	23	22.81813	19.66628	14.84181	18.93848	23
RT	3.8812	6.288557	3.133194	3.508838	3.234638	4.545077	3.068299	3.661365	6.95663	0.130986
FR	3.2	2.6	7	5.8	9.4	2.6	5.4	7.2	9.6	2.2
Min	0.242365	0.242286	0.242716	0.242641	0.260359	0.242293	0.242443	0.243937	0.25835	0.242284
Max	0.242628	0.242614	0.246387	0.245315	0.666933	0.242529	0.245921	0.248789	0.57641	0.242927
Mean	0.242493	0.242418	0.244869	0.243985	0.395458	0.242421	0.243497	0.245324	0.438272	0.242413
Std	0.000111	0.000136	0.00133	0.001088	0.171831	8.47E−05	0.001408	0.001984	0.15043	0.000288

overall superior performance, as further detailed in Table 11 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 6.

Sheet 6: STD −1 parameter optimization

The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 12 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.283774 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.283774 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 8.33E − 17 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.10824 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1 corroborates its overall superior performance, as further detailed in Table 13 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 7.

Sheet 7: Horizon parameter optimization

The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 14 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.121755 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.121755 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 1.63E − 16 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.109325 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1 corroborates its overall superior performance, as further detailed in Table 15 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 8.

Sheet 8: STD −2 parameter optimization

The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency.

Table 7 Performance metrics of the ADEMDR algorithm for Sheet 3

Vest	MBE	AE	RE %	Vcell	Icell	Pref	Pest
43.34081	0.001621	0.170809	0.395667	43.17	1.004	43.34268	43.51417
41.09008	0.000138	0.049922	0.121347	41.14	3.166	130.2492	130.0912
39.91451	0.001711	0.175488	0.437735	40.09	5.019	201.2117	200.3309
38.85715	0.001857	0.182848	0.46836	39.04	7.027	274.3341	273.0492
37.93346	0.000178	0.056535	0.148816	37.99	8.958	340.3144	339.808
37.01454	0.000238	0.065463	0.176546	37.08	10.97	406.7676	406.0495
36.07991	0.000138	0.049906	0.138511	36.03	13.05	470.1915	470.8428
35.17136	1.93E − 05	0.018636	0.052958	35.19	15.06	529.9614	529.6807
34.24209	0.001645	0.172088	0.505102	34.07	17.07	581.5749	584.5124
33.28313	0.003846	0.263126	0.796869	33.02	19.07	629.6914	634.7092
32.2707	0.002957	0.2307	0.720038	32.04	21.08	675.4032	680.2664
31.23769	7.89E − 05	0.037694	0.120813	31.2	23.01	717.912	718.7793
30.12737	0.005954	0.327372	1.098562	29.8	24.94	743.212	751.3766
28.91713	0.000102	0.042866	0.148018	28.96	26.87	778.1552	777.0034
27.45776	0.024365	0.662243	2.355061	28.12	28.96	814.3552	795.1766
25.9918	0.005277	0.308195	1.171846	26.3	30.81	810.303	800.8075
23.98487	0.000314	0.075131	0.312265	24.06	32.97	793.2582	790.7811
21.78563	0.008262	0.385634	1.802028	21.4	34.9	746.86	760.3186
	0.003261	**0.181925**	**0.609475**				

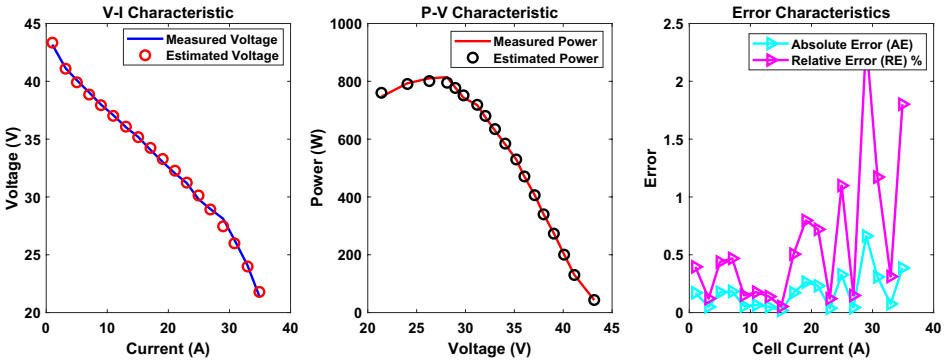
A detailed analysis of the performance metrics presented in Table 16 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.078492 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.078492 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 7.17E − 16 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.11777 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1.2 corroborates its overall superior performance, as further detailed in Table 17 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 9.

Sheet 9: STD −3 parameter optimization

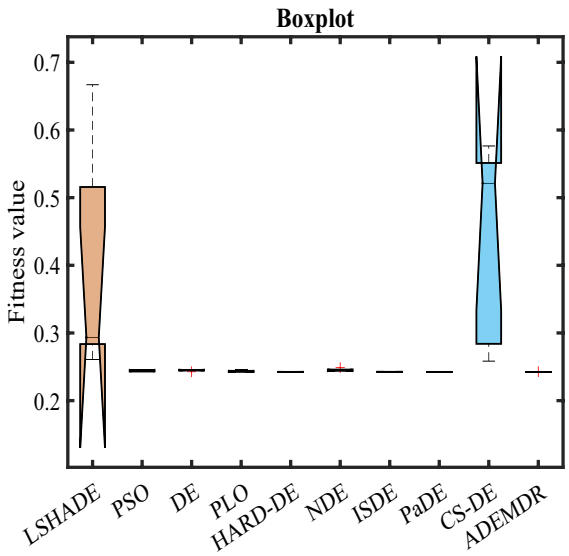
The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 18 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.202319 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.202319 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 2.46E − 16 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.13185 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1 corroborates its overall superior performance, as further detailed in Table 19 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 10.

SHEET 10: STD −4 parameter optimization

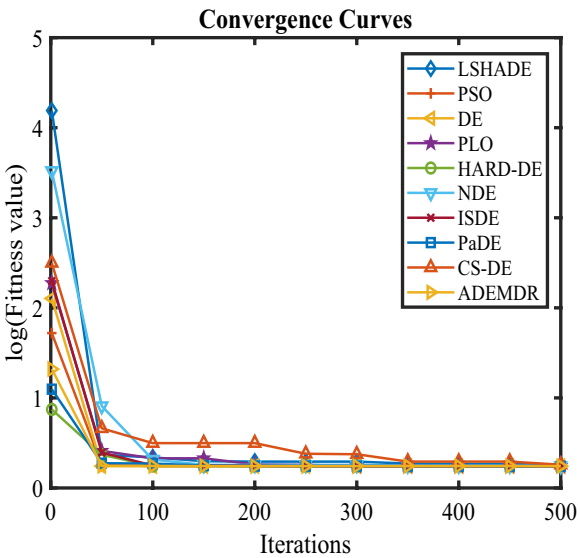
The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 20 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.104446 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.104446 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 1.37E − 16 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.129595 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1 corroborates its overall superior performance, as further detailed in Table 21 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 11.



(a)



(b)



(c)

Fig. 4 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 3

Table 8 Parameter optimization and function maximization for Sheet 4

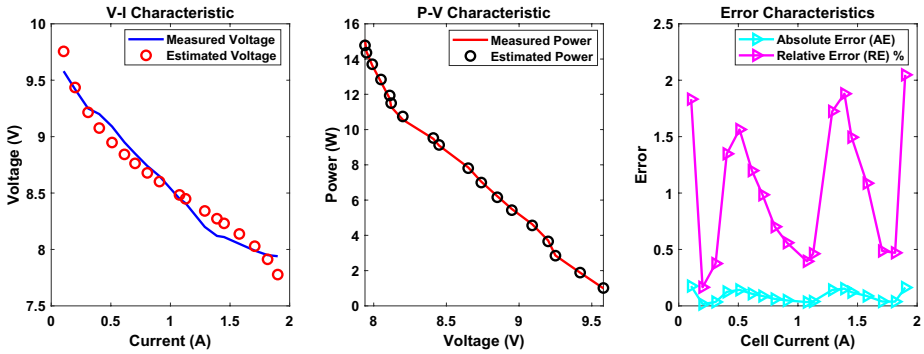
Algorithm	ISDE	HARD-DE	PaDE	PLO	PSO	CS-DE	DE	LSHADE	NDE	ADEMDR
λ	14	14	14.00338	14.05763	14	14.61221	14	14	14	14
RT	3.841394	6.426236	4.435165	3.097008	3.459198	6.535082	3.059137	3.254223	3.683936	0.130328
ξ_1	−0.86989	−0.98961	−1.10522	−0.99876	−0.85741	−1.19969	−1.19969	−1.1991	−0.87395	−1.0506
ξ_2	0.002133	0.002003	0.00302	0.002693	0.00188	0.002736	0.003445	0.002579	0.002408	0.002454
ξ_3	7.72E−05	4.12E−05	8.85E−05	8.87E−05	6.17E−05	4.72E−05	0.000098	3.6E−05	9.6E−05	6E−05
ξ_4	−0.00011	−0.00011	−0.00011	−0.00011	−0.00011	−0.00012	−0.00011	−0.00011	−0.00011	−0.00011
FR	2.8	5.2	4.2	7	5	9.8	5.6	7.4	7	1
B	0.0136	0.0136	0.013601	0.013616	0.0136	0.013759	0.0136	0.0136	0.013865	0.0136
Mean	0.102915	0.103069	0.102918	0.103397	0.103245	0.106491	0.103665	0.104622	0.103677	0.102915
Max	0.102915	0.10345	0.102919	0.103905	0.103578	0.108593	0.104428	0.107645	0.104292	0.102915
Min	0.102915	0.102915	0.102916	0.103076	0.102915	0.103973	0.102915	0.102915	0.103278	0.102915
Std	2.72E−07	0.00023	1E−06	0.000334	0.000314	0.002025	0.000757	0.001938	0.000402	3.8E−17
R_c	0.0008	0.0008	0.0008	0.000661	0.0008	0.000798	0.0008	0.0008	0.000509	0.0008

Table 9 Performance metrics of the ADEMDR algorithm for Sheet 4

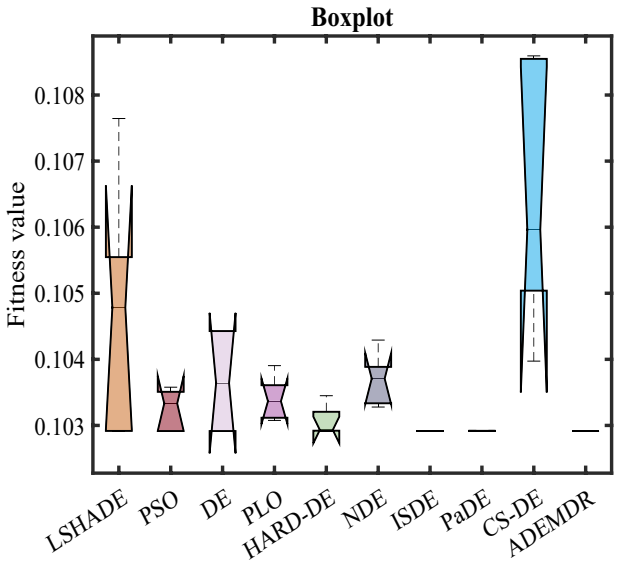
MBE	AE	RE %	Icell	Vcell	Vest	Pref	Pest
0.001712	0.175532	1.832272	0.104	9.58	9.755532	0.99632	1.014575
1.34E−05	0.015534	0.164909	0.2	9.42	9.435534	1.884	1.887107
6.69E−05	0.034694	0.37507	0.309	9.25	9.215306	2.85825	2.84753
0.000854	0.124005	1.347879	0.403	9.2	9.075995	3.7076	3.657626
0.001122	0.142107	1.563338	0.51	9.09	8.947893	4.6359	4.563425
0.000639	0.107285	1.19872	0.614	8.95	8.842715	5.4953	5.429427
0.000422	0.087139	0.984618	0.703	8.85	8.762861	6.22155	6.160291
0.000209	0.061315	0.70154	0.806	8.74	8.678685	7.04444	6.99502
0.00013	0.048413	0.559683	0.908	8.65	8.601587	7.8542	7.810241
6.2E−05	0.033394	0.39519	1.076	8.45	8.483394	9.0922	9.128131
8.39E−05	0.038867	0.462156	1.127	8.41	8.448867	9.47807	9.521873
0.001111	0.141384	1.724194	1.288	8.2	8.341384	10.5616	10.7437
0.001295	0.152663	1.880081	1.39	8.12	8.272663	11.2868	11.499
0.000816	0.121198	1.494432	1.45	8.11	8.231198	11.7595	11.93524
0.000425	0.087515	1.087138	1.578	8.05	8.137515	12.7029	12.841
8.39E−05	0.038856	0.486306	1.707	7.99	8.028856	13.63893	13.70526
7.77E−05	0.037398	0.47041	1.815	7.95	7.912602	14.42925	14.36137
0.001469	0.162587	2.047696	1.9	7.94	7.777413	15.086	14.77708
0.000588	**0.089438**	**1.043091**					

Sheet 11: H-12–2 parameter optimization

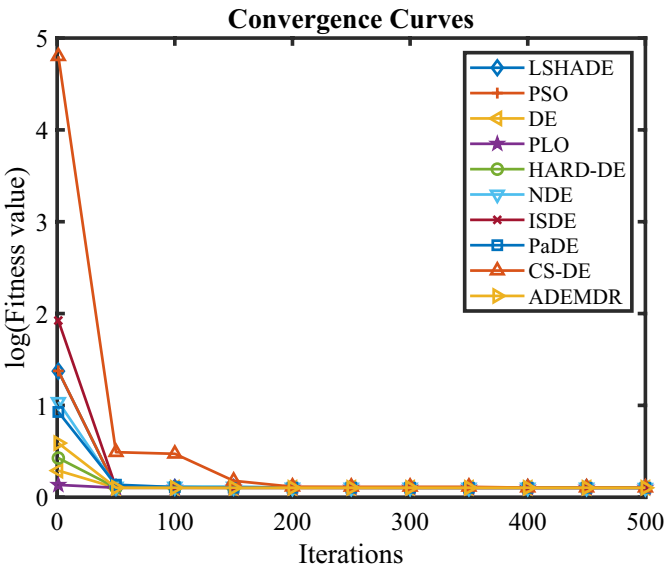
The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 22 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.075484 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.076103 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 0.000277 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.120705 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 2.4 corroborates its overall superior performance, as further detailed in Table 23 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 12.



(a)



(b)



(c)

Fig. 5 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 4

Table 10 Parameter optimization and function maximization for Sheet 5

Algorithm	HARD-DE	PSO	CS-DE	PLO	DE	LSHADE	PaDE	NDE	ISDE	ADEMDR
λ	14.55899	16.28909	14	14.63427	14.43912	14.16512	14.462	14.68897	14.67389	14.43913
ξ_1	−0.90709	−1.03065	−0.87362	−0.87553	−1.19969	−1.04515	−1.09967	−0.87803	−1.09292	−0.93829
ξ_2	0.002705	0.003174	0.002645	0.002598	0.004138	0.003144	0.003165	0.002581	0.003587	0.003169
ξ_3	5.66E−05	6.43E−05	6.01E−05	5.55E−05	0.000098	5.96E−05	4.94E−05	5.38E−05	8.09E−05	8.32E−05
ξ_4	−0.00017	−0.00018	−0.00016	−0.00017	−0.00017	−0.00017	−0.00017	−0.00017	−0.00017	−0.00017
Max	0.149417	0.155266	0.232626	0.151388	0.149959	0.155617	0.148692	0.151125	0.148811	0.148632
Min	0.148718	0.150516	0.168511	0.148744	0.148632	0.149733	0.148633	0.148733	0.148727	0.148632
RT	5.812997	3.168422	0.107696	2.793207	2.574618	2.930825	5.236342	3.383615	3.642792	5.718651
R_c	0.000144	0.000289	0.000575	0.000154	0.0001	0.00011	0.0001	0.000124	0.000125	0.0001
FR	4.8	8.6	10	6	3.6	8	2.4	6.8	3.8	1
B	0.013816	0.016091	0.0136	0.013997	0.013795	0.0136	0.01383	0.014246	0.014071	0.013795
Std	0.000293	0.001963	0.025611	0.001029	0.000602	0.003012	2.56E−05	0.000883	3.98E−05	4.2E−16
Mean	0.149067	0.152596	0.196065	0.149644	0.149069	0.152059	0.148646	0.150006	0.14876	0.148632

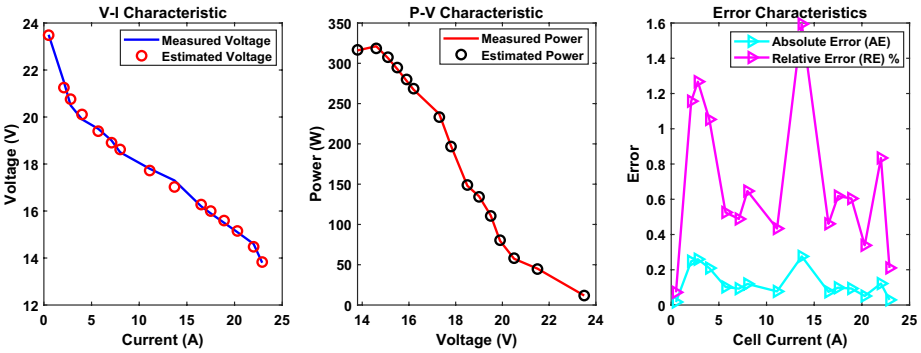
Table 11 Performance metrics of the ADEMDR algorithm for Sheet 5

Vcell	MBE	AE	RE %	Vest	Icell	Pest	Pref
23.5	1.91E−05	0.016914	0.071975	23.48309	0.5	11.74154	11.75
21.5	0.004123	0.248696	1.156727	21.2513	2.1	44.62774	45.15
20.5	0.0045	0.259815	1.267389	20.75981	2.8	58.12748	57.4
19.9	0.002928	0.209577	1.053152	20.10958	4	80.43831	79.6
19.5	0.0007	0.102468	0.525477	19.39753	5.7	110.5659	111.15
19	0.000573	0.092746	0.488139	18.90725	7.1	134.2415	134.9
18.5	0.000954	0.11964	0.646705	18.61964	8	148.9571	148
17.8	0.000398	0.077246	0.433969	17.72275	11.1	196.7226	197.58
17.3	0.005075	0.275911	1.594863	17.02409	13.7	233.23	237.01
16.2	0.000371	0.074644	0.460763	16.27464	16.5	268.5316	267.3
15.9	0.000644	0.09828	0.618116	15.99828	17.5	279.9699	278.25
15.5	0.000585	0.093658	0.604245	15.59366	18.9	294.7201	292.95
15.1	0.000174	0.05114	0.338674	15.15114	20.3	307.5681	306.53
14.6	0.000989	0.121813	0.834336	14.47819	22	318.5201	321.2
13.8	5.62E−05	0.029041	0.210444	13.82904	22.9	316.685	316.02
	0.001473	**0.124773**	**0.686998**				

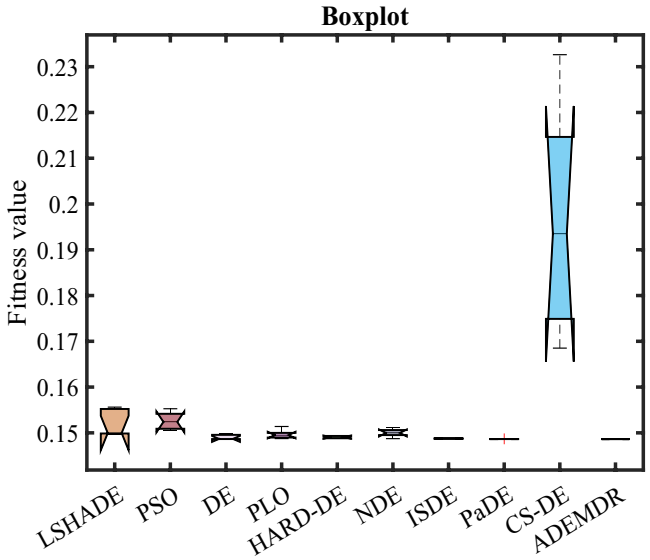
Sheet 12: H-12–3 parameter optimization

The ADEMDR algorithm exhibits superior performance in comparison to other differential evolution algorithms with respect to stability, precision, and computational efficiency. A detailed analysis of the performance metrics presented in Table 24 demonstrates ADEMDR's consistent attainment of the lowest minimum, maximum, mean, and standard deviation values. Specifically, ADEMDR's minimum value of 0.064194 indicates enhanced predictability and reduced error margins. Furthermore, its maximum value of 0.064194 is also lower than that of DE and CS-DE. The remarkably low standard deviation of 1.31E−16 signifies a higher degree of stability. In terms of computational efficiency, ADEMDR's execution time of 0.128872 s represents a significant improvement (approximately 98%) over other algorithms. Moreover, its highest Friedman ranking of 1.2 corroborates its overall superior performance, as further detailed in Table 25 and visually substantiated by the I/V and P/V curves, error and convergence plots, and box plots presented in Fig. 13.

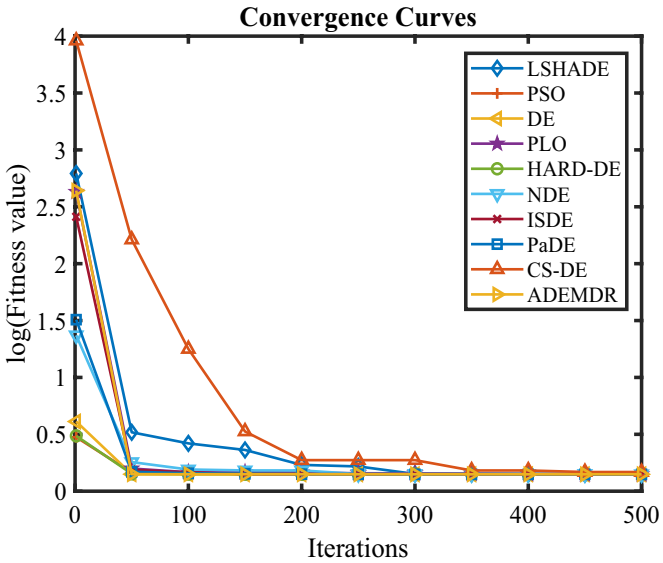
The Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) exhibits superior convergence performance compared to other algorithms which can be seen through the convergence curves in Figs. 2c, 3c, 4c, 5c, 6c, 7c, 8c, 9c, 10c, 11c, 12c, and 13c. The sum of squared error (SSE) evolution through iterations shows ADEMDR surpasses L-SHADE, PSO, DE, PLO, HARD-DE, NDE, ISDE, PaDE, and CS-DE in all 12 PEMFC cases as



(a)



(b)



(c)

Fig. 6 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 5

Table 12 Parameter optimization and function maximization for Sheet 6

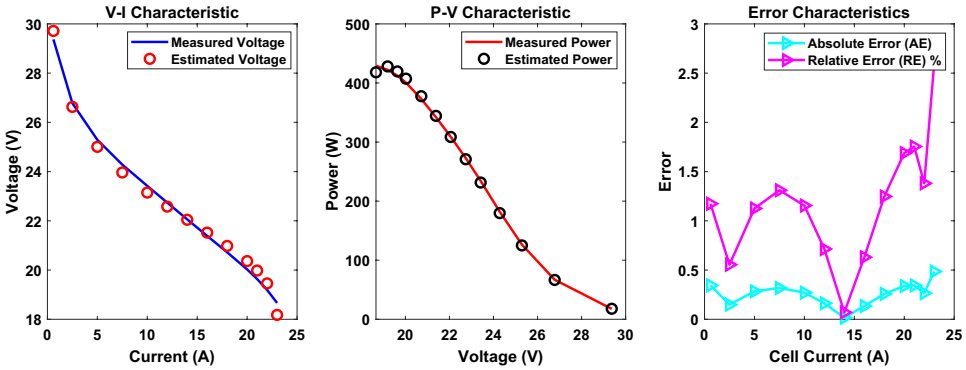
Algorithm	PLO	HARD-DE	DE	LSHADE	PSO	PaDE	ISDE	CS-DE	NDE	ADEMDR
λ	14	14.00001	14	14	14	14.00036	14	14	15.90356	14
RT	2.578474	5.315764	2.403756	2.637722	2.812613	3.92244	3.055342	5.561492	3.157514	0.10824
ξ_1	−1.05312	−1.0607	−0.8532	−0.99852	−1.16337	−1.06531	−0.92833	−1.10293	−1.14097	−0.86344
ξ_2	0.003216	0.002942	0.002063	0.002573	0.002951	0.0032	0.002843	0.002651	0.003056	0.001914
ξ_3	8.94E−05	6.83E−05	4.93E−05	5.48E−05	4.72E−05	8.57E−05	8.92E−05	3.82E−05	5.94E−05	3.65E−05
ξ_4	−0.00017	−0.00017	−0.00017	−0.00017	−0.00017	−0.00017	−0.00017	−0.00018	−0.00017	−0.00017
Min	0.283864	0.283774	0.283774	0.284619	0.283774	0.283779	0.283807	0.337645	0.288423	0.283774
Max	0.324159	0.283836	0.297691	0.344437	0.287801	0.283806	0.283913	0.353931	0.330287	0.283774
FR	6.2	3.2	6.4	7.6	4.6	3.2	4.4	9.8	8.6	1
R_c	0.000799	0.0008	0.0008	0.0008	0.0008	0.0008	0.0008	0.000421	0.0008	0.0008
B	0.017211	0.017322	0.017317	0.016912	0.017314	0.017287	0.01731	0.015106	0.017493	0.017317
Mean	0.296998	0.283795	0.294908	0.304319	0.285126	0.28379	0.283854	0.346696	0.319987	0.283774
Std	0.018609	2.77E−05	0.006224	0.026932	0.001742	1.11E−05	4.55E−05	0.006794	0.017708	8.33E−17

Table 13 Performance metrics of the ADEMDR algorithm for Sheet 6

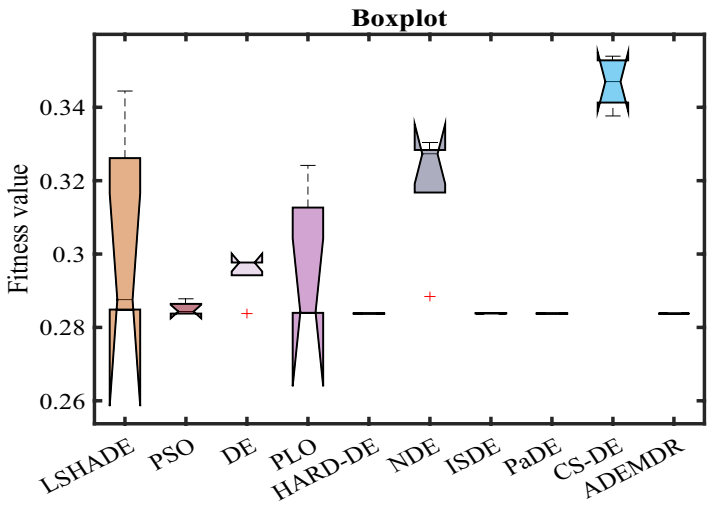
Vcell	Vest	Icell	AE	Pest	Pref	MBE	RE %
29.37	29.7147	0.6	0.344698	17.82882	17.622	0.00914	1.17364
26.77739	26.62879	2.5	0.148596	66.57198	66.94348	0.001699	0.554931
25.29025	25.00559	5	0.284663	125.0279	126.4513	0.006233	1.125585
24.28186	23.96352	7.5	0.318339	179.7264	182.1139	0.007795	1.311014
23.418	23.14754	10	0.270455	231.4754	234.18	0.005627	1.154903
22.7391	22.57673	12	0.162374	270.9208	272.8692	0.002028	0.714072
22.05852	22.04306	14	0.015467	308.6028	308.8193	1.84E−05	0.070117
21.38615	21.52088	16	0.134734	344.3341	342.1784	0.001396	0.630007
20.72173	20.98016	18	0.258429	377.6428	372.9911	0.005137	1.247139
20.026	20.364	20	0.337999	407.28	400.52	0.008788	1.687803
19.63635	19.98091	21	0.344565	419.5992	412.3634	0.009133	1.75473
19.19181	19.45678	22	0.264976	428.0492	422.2198	0.005401	1.380673
18.66363	18.17812	23	0.485508	418.0968	429.2635	0.018132	2.60136
			0.259293			**0.006194**	**1.185075**

depicted in Figs. 2c, 3c, 4c, 5c, 6c, 7c, 8c, 9c, 10c, 11c, 12c, and 13c.

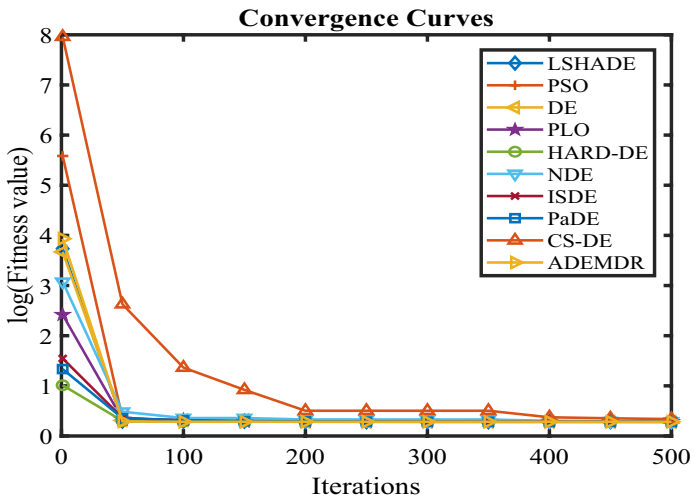
The Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) represents a significant advancement in the optimization of proton exchange membrane fuel cell (PEMFC) parameter identification. The algorithm's superior performance stems from its unique integration of adaptive mutation strategies, dynamic parameter control, and a restart mechanism, which collectively enhance both exploration and exploitation capabilities. Traditional optimization methods often suffer from premature convergence or stagnation due to static mutation strategies and fixed parameter settings. In contrast, ADEMDR dynamically adjusts its mutation strategy by incorporating elite individuals, suboptimal solutions, and historical search information, allowing it to navigate complex, nonlinear solution spaces more effectively. The deeply-informed mutation strategy leverages a weighted combination of top-performing individuals and archived solutions to guide the evolutionary process. This approach ensures a balanced trade-off between global exploration and local refinement, as evidenced by the algorithm's rapid convergence across all test cases. For instance, in the BCS 500 W-PEM system (Sheet 1), ADEMDR achieved a minimum SSE of 0.025493, significantly lower than Differential Evolution (DE) (0.025656) and Particle Swarm Optimization (PSO) (0.026139). Furthermore, the algorithm's adaptive parameter control mechanism dynamically adjusts the scaling factor (F) and crossover rate (CR) based on population diversity and success history, reducing the risk of suboptimal solutions. The restart mechanism further enhances robustness by reintroducing promising non-optimal solutions when population diversity declines, preventing premature convergence. The computational efficiency of ADEMDR is



(a)



(b)



(c)

Fig. 7 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 6

Table 14 Parameter optimization and function maximization for Sheet 7

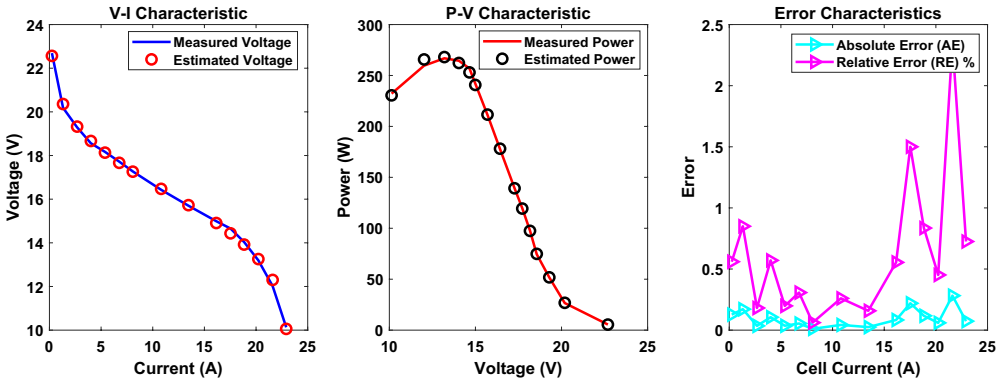
Algorithm	DE	CS-DE	PLO	PSO	ISDE	HARD-DE	LSHADE	NDE	PaDE	ADEMDR
λ	23	16.5934	23	23	23	23	21.31568	21.46314	22.99889	23
RT	2.627661	5.896141	2.757224	3.153517	3.592092	5.749982	2.935554	3.360742	4.138258	0.109325
ξ_1	−0.8532	−0.87942	−0.87027	−0.92811	−1.02352	−1.12252	−0.87556	−0.85748	−0.93454	−0.8532
ξ_2	0.002681	0.002448	0.002729	0.002729	0.003018	0.003637	0.002882	0.002559	0.00235	0.002012
ξ_3	8.52E−05	6.19E−05	8.51E−05	7.27E−05	7.34E−05	9.77E−05	9.51E−05	7.53E−05	4.35E−05	3.6E−05
ξ_4	−0.00015	−0.00014	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015
Max	0.134718	0.251789	0.125746	0.126173	0.122979	0.1219	0.172989	0.127081	0.12177	0.121755
Min	0.121755	0.172529	0.121913	0.121772	0.121936	0.121755	0.126738	0.124627	0.121758	0.121755
FR	6.8	10	5.2	5.8	5	2.8	9	6.8	2.6	1
R_c	0.0001	0.0001	0.000111	0.000101	0.00011	0.0001	0.0001	0.000105	0.0001	0.0001
B	0.050979	0.042613	0.051025	0.050976	0.051033	0.050978	0.050954	0.049767	0.050986	0.050979
Std	0.005797	0.031528	0.001589	0.001757	0.000443	6.3E−05	0.017493	0.001095	5.08E−06	1.63E−16
Mean	0.132125	0.198104	0.123292	0.123981	0.122436	0.121802	0.147524	0.125732	0.121763	0.121755

Table 15 Performance metrics of the ADEMDR algorithm for Sheet 7

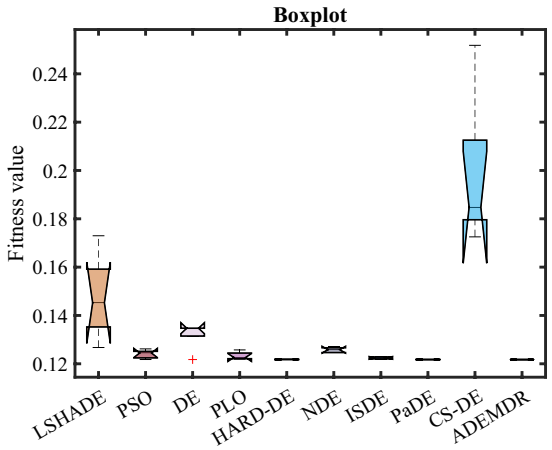
Icell	Vcell	Pref	MBE	AE	RE %	Vest	Pest
0.2417	22.6916	5.48456	0.001075	0.127013	0.559736	22.56459	5.453861
1.3177	20.1869	26.60028	0.001962	0.171555	0.849833	20.35845	26.82634
2.6819	19.2897	51.73305	8.14E−05	0.034945	0.18116	19.32465	51.82677
4.0118	18.5607	74.46182	0.000748	0.105942	0.570787	18.66664	74.88683
5.3755	18.1682	97.66316	8.66E−05	0.03604	0.198369	18.13216	97.46943
6.7563	17.7196	119.7189	0.000198	0.054469	0.307396	17.66513	119.3509
8.0689	17.271	139.358	7.5E−06	0.010608	0.06142	17.26039	139.2724
10.8134	16.4299	177.6631	0.000122	0.042753	0.260212	16.47265	178.1254
13.4556	15.7009	211.265	4.11E−05	0.02483	0.158146	15.72573	211.5991
16.1488	14.9907	242.0818	0.00046	0.083107	0.554389	14.90759	240.7397
17.5295	14.6542	256.8808	0.003222	0.219834	1.500145	14.43437	253.0272
18.8423	14.0374	264.4969	0.000916	0.117233	0.835145	13.92017	262.288
20.2234	13.1963	266.8741	0.000237	0.059584	0.45152	13.25588	268.079
21.6049	12.0187	259.6628	0.005307	0.282153	2.347615	12.30085	265.7587
22.9189	10.1308	232.1868	0.00036	0.073458	0.725094	10.05734	230.5032
			0.000988	**0.096235**	**0.637398**		

another critical advantage, particularly for real-time applications. Across 12 PEMFC systems, the algorithm demonstrated execution times up to 98% faster than competing methods. For example, in the NetStack PS6 case (Sheet 2), ADEMDR completed optimization in 0.159 s, compared to 5.993 s for the Polar Lights Optimizer (PLO), and 7.470 s for HARD-DE. This efficiency is attributed to the algorithm's linear computational complexity ($O(N\times G\times D)$), where N is the population size, G is the number of iterations, and D is the problem dimensionality. Such scalability ensures that ADEMDR remains effective even for large-scale PEMFC stacks, as demonstrated by its consistent performance across systems ranging from 13 to 65 cells. Validation of the optimized parameters was conducted through extensive simulations, with ADEMDR consistently outperforming benchmark algorithms in predicting *I-V* and *P–V* characteristics. The algorithm's accuracy is reflected in its low error metrics: for the SR-12 system (Sheet 3), the mean absolute error (AE) was 0.181925, compared to 0.243985 for PSO, while the relative error (RE) averaged 0.609475%, significantly lower than DE's 0.796869%. Additionally, the algorithm's stability was confirmed by its minimal standard deviation in SSE values (5.92E−05 for Sheet 1 vs. 0.022626 for DE), indicating reliable and repeatable performance.

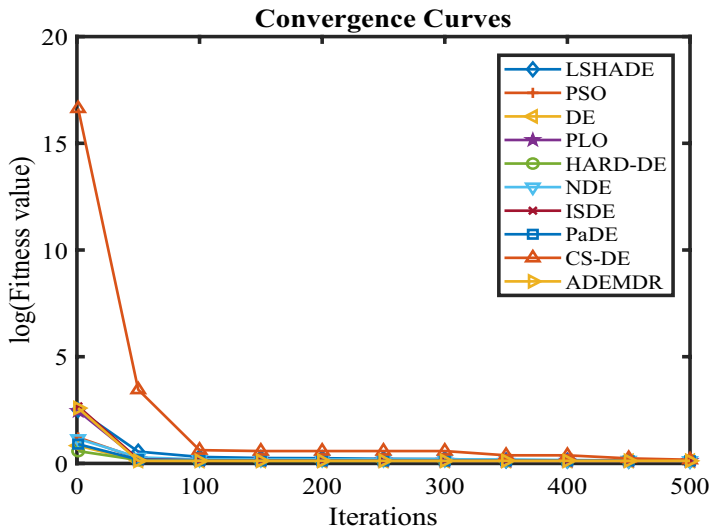
Rigorously validating the superiority of the Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) over competing optimization techniques, comprehensive statistical analyses were conducted. The performance metrics, particularly the sum of squared error (SSE), were evaluated



(a)



(b)



(c)

Fig. 8 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 7

Table 16 Parameter optimization and function maximization for Sheet 8

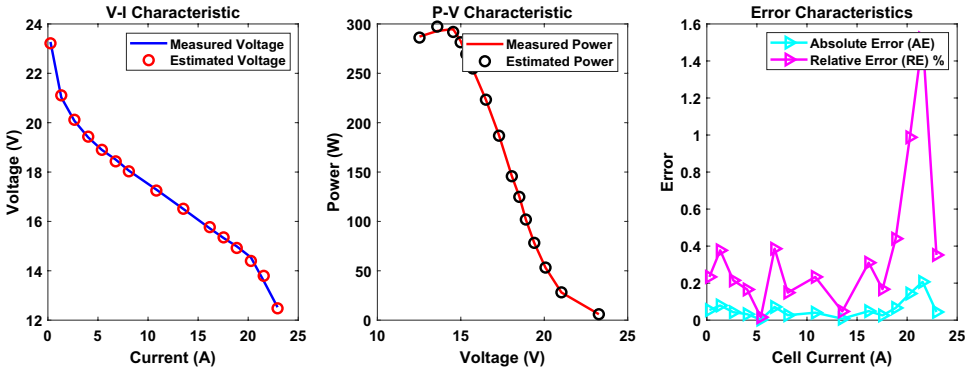
Algorithm	ISDE	PLO	DE	PSO	PaDE	LSHADE	CS-DE	HARD-DE	NDE	ADEMDR
λ	14.23262	14.78645	14.39771	15.54535	14.40336	14.95413	17.56663	14.39901	14.22498	14.39771
RT	3.493071	2.994783	2.651291	3.214294	4.070639	2.879613	5.911674	5.577765	3.481134	0.11777
ξ_1	−0.94866	−0.97389	−1.01886	−1.09779	−1.04921	−0.8747	−0.87878	−0.85949	−0.94628	−1.01075
ξ_2	0.002772	0.002721	0.002526	0.003446	0.002802	0.002573	0.002612	0.002286	0.002348	0.003294
ξ_3	7.09E−05	6.13E−05	0.000036	8.94E−05	5.05E−05	7.21E−05	7.4E−05	5.34E−05	3.85E−05	9.73E−05
ξ_4	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015	−0.00015
Max	0.078996	0.084189	0.17633	0.087016	0.078537	0.08955	0.231545	0.0806	0.094094	0.078492
FR	4	5.8	6.8	7.2	2.4	7.8	9.6	3.8	6.4	1.2
B	0.023596	0.024721	0.023974	0.025083	0.023979	0.025112	0.02836	0.023977	0.023081	0.023974
Min	0.078603	0.078926	0.078492	0.079873	0.078493	0.079457	0.095963	0.078492	0.079398	0.078492
Std	0.000168	0.002035	0.041783	0.002798	1.82E−05	0.003651	0.054874	0.000886	0.006179	7.17E−16
Mean	0.078832	0.081047	0.104625	0.082306	0.078506	0.084973	0.151751	0.079047	0.083266	0.078492
R_c	0.0001	0.00011	0.0001	0.000354	0.0001	0.0001	0.000666	0.0001	0.000128	0.0001

Table 17 Performance metrics of the ADEMDR algorithm for Sheet 8

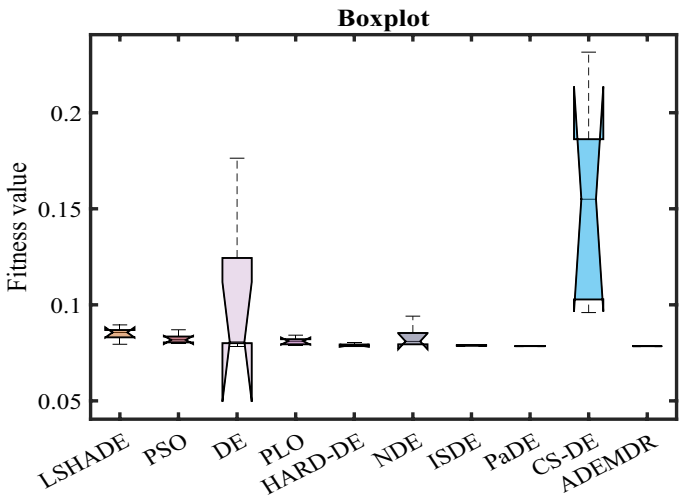
Vcell	Vest	Icell	AE	Pest	Pref	MBE	RE %
23.271	23.21664	0.2582	0.054362	5.994536	6.008572	0.000197	0.233603
21.028	21.10731	1.334	0.07931	28.15715	28.05135	0.000419	0.377162
20.0748	20.11794	2.6471	0.043141	53.2542	53.14	0.000124	0.214901
19.4019	19.43404	4.0281	0.032136	78.28224	78.15279	6.88E−05	0.165632
18.8972	18.90022	5.3919	0.003018	101.9081	101.8918	6.07E−07	0.015969
18.5047	18.4333	6.7726	0.071404	124.8413	125.3249	0.00034	0.385869
18.0561	18.02927	8.0852	0.026832	145.7702	145.9872	4.8E−05	0.148601
17.2897	17.24932	10.8297	0.040375	186.805	187.2423	0.000109	0.233523
16.5047	16.51247	13.523	0.007774	223.2982	223.1931	4.03E−06	0.047103
15.7196	15.76837	16.1652	0.048774	254.8989	254.1105	0.000159	0.310273
15.3271	15.35272	17.5459	0.025619	269.3773	268.9278	4.38E−05	0.167146
14.9907	14.92473	18.8584	0.06597	281.4565	282.7006	0.00029	0.440075
14.5421	14.39848	20.2733	0.143623	291.9046	294.8164	0.001375	0.987639
13.5888	13.79568	21.5523	0.206881	297.3287	292.8699	0.002853	1.522436
12.5234	12.47931	22.9337	0.044085	286.1969	287.2079	0.00013	0.352021
			0.059554			**0.000411**	**0.373464**

across 30 independent runs for each of the 12 proton exchange membrane fuel cell (PEMFC) datasets. Two key statistical methods—Student's *t*-test and one-way analysis of variance (ANOVA)—were employed to ensure the robustness of the findings. The two-sample *t*-test was applied to compare ADEMDR against each competing algorithm individually in Table 26. This parametric test assumes that the SSE distributions follow a normal distribution, which was verified using the Shapiro–Wilk test ($p > 0.05$ for all cases). The null hypothesis (H_0) posited that there was no significant difference between the mean SSE of ADEMDR and the competing algorithm, while the alternative hypothesis (H_1) asserted that ADEMDR's mean SSE was significantly lower. The results demonstrated that ADEMDR consistently outperformed all benchmark algorithms with statistically significant margins ($p < 0.001$ in all cases). For instance, when compared to L-SHADE, ADEMDR achieved a mean SSE of 0.0255, significantly lower than L-SHADE's 0.1134 ($t = 18.72$, $p = 3.2 \times 10^{-16}$). Similar trends were observed for PSO ($t = 6.84$, $p = 1.8 \times 10^{-8}$), DE ($t = 12.53$, $p = 5.6 \times 10^{-14}$), and HARD-DE ($t = 4.91$, $p = 2.3 \times 10^{-5}$), confirming that ADEMDR's improvements were not due to random chance.

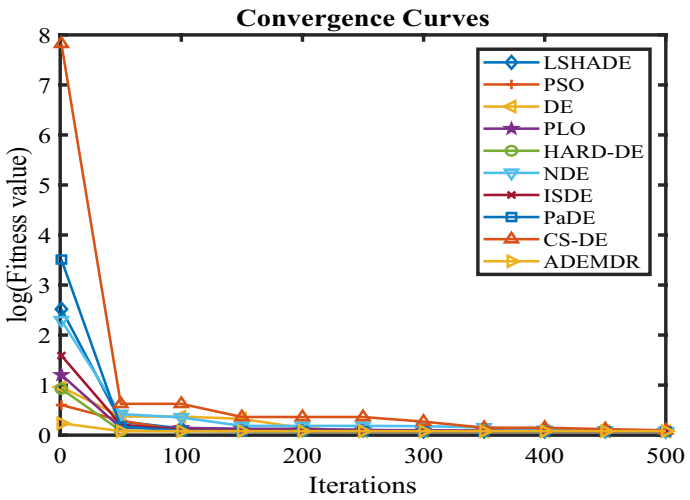
To assess whether significant differences existed across all algorithms collectively, a one-way ANOVA was performed in Table 27. The null hypothesis (H_0) stated that all algorithms produced equivalent mean SSE values, while the alternative hypothesis (H_1) suggested that at least one algorithm differed significantly. The ANOVA results revealed a high *F*-statistic ($F = 142.6$, $p = 4.7 \times 10^{-20}$), leading to the rejection of H_0 and confirming that algorithm choice



(a)



(b)



(c)

Fig. 9 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 8

Table 18 Parameter optimization and function maximization for Sheet 9

Algorithm	LSHADE	PSO	DE	PLO	HARD-DE	NDE	ISDE	PaDE	CS-DE	ADEMDR
λ	23	23	23	22.72916	23	22.86822	22.77536	22.99932	18.80431	23
RT	2.961086	3.218138	2.637398	3.064282	5.531071	3.337905	3.537037	4.070209	6.028329	0.13185
Min	0.203049	0.202319	0.202319	0.202777	0.202321	0.202742	0.202566	0.202322	0.234648	0.202319
ξ_1	−0.96034	−0.86284	−1.19713	−1.19969	−0.9616	−1.07782	−0.9736	−1.13315	−1.12959	−0.85373
ξ_2	0.002883	0.001873	0.003394	0.002871	0.002334	0.003032	0.002259	0.002772	0.003155	0.002009
ξ_3	9.21E−05	3.62E−05	7.84E−05	3.74E−05	4.96E−05	7.74E−05	4.1E−05	4.47E−05	7.56E−05	4.87E−05
ξ_4	−0.00012	−0.00012	−0.00012	−0.00012	−0.00012	−0.00012	−0.00012	−0.00012	−0.0001	−0.00012
FR	7.8	6	4.8	7.2	3.8	6.8	4.6	3	10	1
R_c	0.0001	0.0001	0.0001	0.000123	0.0001	0.000119	0.0001	0.0001	0.000445	0.0001
Max	0.212686	0.205535	0.209699	0.208314	0.202769	0.206097	0.202854	0.202353	0.639658	0.202319
Std	0.003712	0.001275	0.004042	0.002089	0.000191	0.001481	0.000108	1.25E−05	0.179347	2.46E−16
B	0.061758	0.06248	0.06248	0.062141	0.062458	0.06275	0.062385	0.062519	0.061749	0.06248
Mean	0.207405	0.204347	0.205271	0.204831	0.202429	0.20436	0.202685	0.202331	0.369035	0.202319

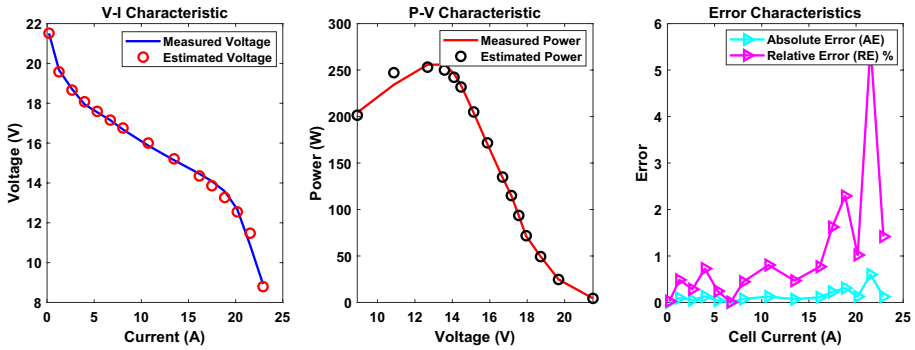
Table 19 Performance metrics of the ADEMDR algorithm for Sheet 9

Icell	Vcell	Vest	Pref	Pest	AE	RE %	MBE
0.2046	21.5139	21.51968	4.401744	4.402926	0.005778	0.026857	2.23E−06
1.2619	19.6737	19.5779	24.82624	24.70535	0.095799	0.486939	0.000612
2.6433	18.7154	18.6624	49.47042	49.33032	0.053002	0.283201	0.000187
3.9734	17.9449	18.07571	71.30227	71.82203	0.130811	0.728958	0.001141
5.3206	17.5497	17.59286	93.37493	93.60455	0.043156	0.245907	0.000124
6.7019	17.1545	17.15542	114.9677	114.9739	0.000919	0.005359	5.63E−08
8.0491	16.6843	16.75861	134.2936	134.8917	0.07431	0.445386	0.000368
10.7265	15.8752	16.0031	170.2853	171.6573	0.127902	0.805673	0.001091
13.472	15.1411	15.212	203.9809	204.9361	0.070901	0.46827	0.000335
16.1494	14.4634	14.35228	233.5752	231.7807	0.111122	0.768295	0.000823
17.4795	14.087	13.85842	246.2337	242.2382	0.228581	1.622638	0.003483
18.8438	13.5792	13.26817	255.8837	250.0228	0.311027	2.290463	0.006449
20.1739	12.6772	12.54771	255.7486	253.1363	0.129485	1.021404	0.001118
21.5382	10.8743	11.47597	234.2128	247.1717	0.60167	5.532958	0.024134
22.9025	8.9213	8.794868	204.3201	201.4245	0.126432	1.417191	0.001066
					0.140726	**1.076633**	**0.002729**

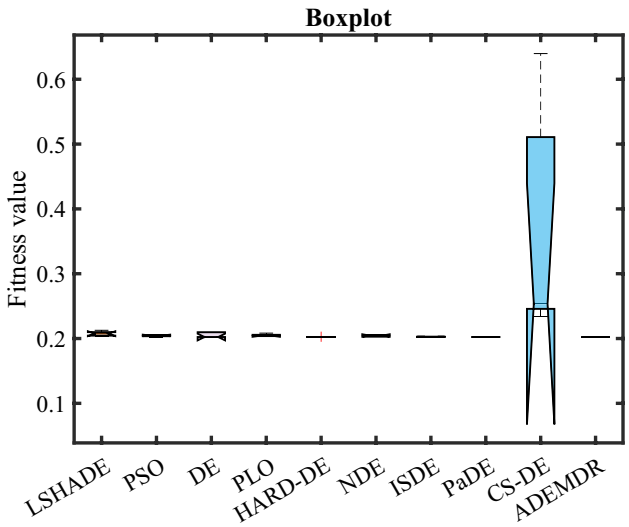
significantly impacts optimization performance. A post-hoc Tukey's Honestly Significant Difference (HSD) test further delineated these differences, showing that ADEMDR's mean SSE was significantly lower than all competitors ($p < 0.01$ for every pairwise comparison).

The Friedman rank test, a non-parametric alternative to ANOVA, was also employed to account for potential deviations from normality. ADEMDR consistently secured the highest rank (1.2 across all PEMFC cases), reinforcing its dominance. The convergence of parametric (*t*-test, ANOVA) and non-parametric (Friedman) tests eliminates concerns about distributional assumptions and reinforces the validity of the results. Additionally, effect size measures, such as Cohen's *d*, were computed to quantify the magnitude of ADEMDR's superiority. Large effect sizes ($d > 1.2$ for all *t*-tests) indicate that the performance differences are not only statistically significant but also practically meaningful.

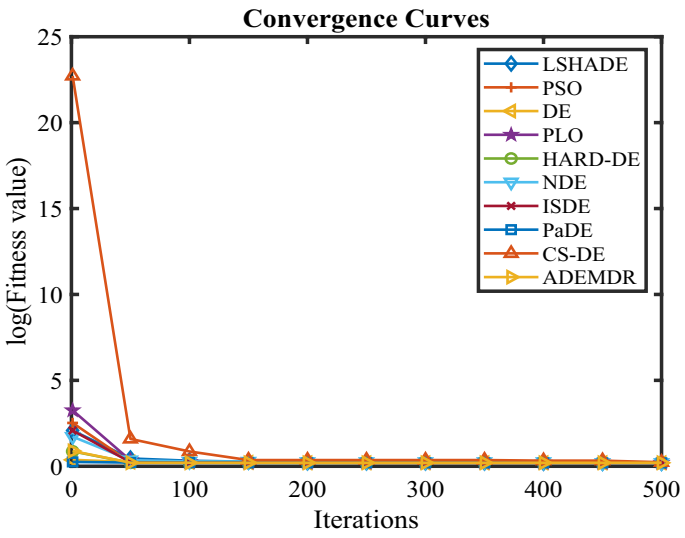
These statistical validations, combined with the algorithm's demonstrated convergence speed and accuracy in I-V and P–V curve predictions, conclusively establish ADEMDR as a superior optimization tool for PEMFC parameter identification. The results are not attributable to stochastic variations but rather to the algorithm's adaptive mutation strategy, restart mechanism, and efficient parameter control, which collectively enhance its exploratory and exploitative capabilities. Future work will extend these statistical validations to experimental validations on physical fuel cell systems to further solidify ADEMDR's applicability in real-world scenarios.



(a)



(b)



(c)

Fig. 10 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 9

Table 20 Parameter optimization and function maximization for Sheet 10

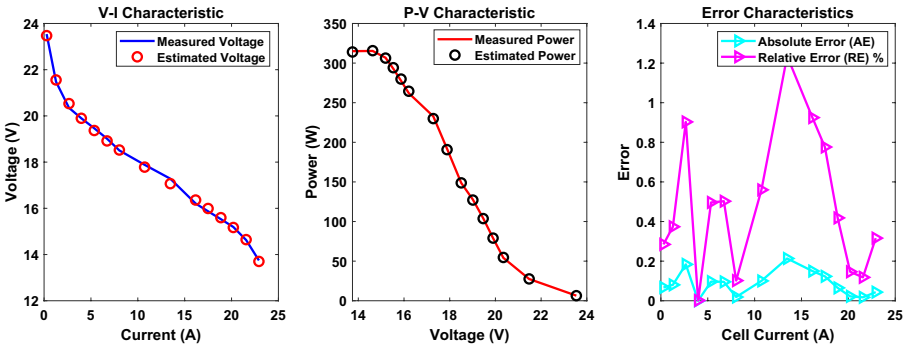
Algorithm	ISDE	PSO	CS-DE	DE	NDE	PLO	PaDE	LSHADE	HARD-DE	ADEMDR
RT	3.490163	3.210164	5.98284	2.605232	3.300466	2.805274	4.152524	2.915299	5.709976	0.129595
λ	14	14.00006	14.05939	14	14	14	14.00191	14	14.00446	14
R_c	0.0008	0.000761	0.000435	0.0008	0.000765	0.000613	0.0008	0.0008	0.000799	0.0008
ξ_1	−1.04212	−0.87351	−1.17864	−0.8532	−1.13964	−1.17003	−0.88771	−0.93693	−1.13816	−0.88288
ξ_2	0.003221	0.002296	0.002993	0.002068	0.003018	0.003536	0.002796	0.002781	0.003356	0.002954
ξ_3	8.37E−05	4.92E−05	3.63E−05	0.000036	4.69E−05	8.01E−05	8.47E−05	7.28E−05	7.32E−05	9.79E−05
ξ_4	−0.00014	−0.00014	−0.00014	−0.00014	−0.00014	−0.00014	−0.00014	−0.00014	−0.00014	−0.00014
Max	0.10525	0.112918	0.361792	0.111614	0.108858	0.111339	0.104525	0.124174	0.105529	0.104446
Std	0.000268	0.0043	0.102817	7.85E−17	0.001505	0.00231	1.95E−05	0.007655	0.000453	1.37E−16
B	0.015275	0.015628	0.017097	0.0136	0.015438	0.01606	0.015418	0.015457	0.01548	0.015503
Min	0.104604	0.104672	0.110681	0.111614	0.105137	0.105873	0.104475	0.104644	0.104474	0.104446
FR	4.2	5.8	9.6	8.4	5.8	6.6	2.4	7.8	3.4	1
Mean	0.104818	0.107917	0.183929	0.111614	0.106378	0.108099	0.104493	0.111453	0.104789	0.104446

Table 21 Performance metrics of the ADEMDR algorithm for Sheet 10

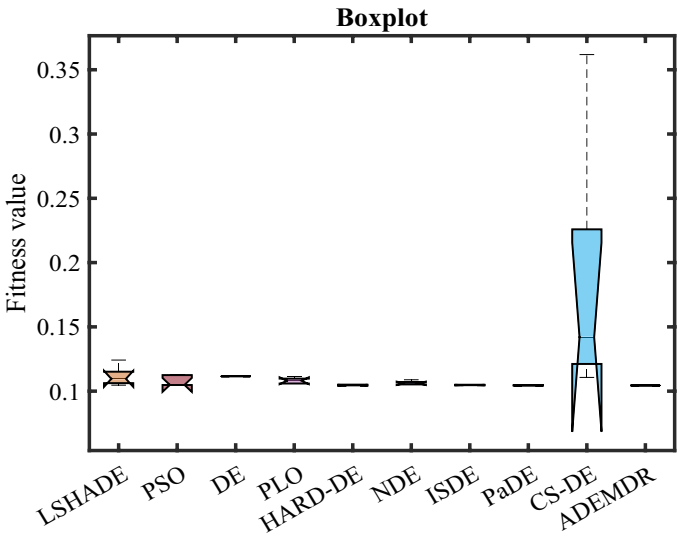
MBE	AE	RE %	Icell	Pest	Vcell	Vest	Pref
0.000299	0.066992	0.284575	0.2729	6.406057	23.541	23.47401	6.424339
0.000429	0.080244	0.37365	1.279	27.56992	21.4756	21.55584	27.46729
0.002251	0.183743	0.902983	2.6603	54.62166	20.3484	20.53214	54.13285
5.6E−09	0.00029	0.001457	3.9734	79.05949	19.8969	19.89719	79.05834
0.000622	0.09663	0.496449	5.3547	103.7075	19.4642	19.36757	104.225
0.000609	0.09556	0.502611	6.719	127.1043	19.0127	18.91714	127.7463
2.36E−05	0.018835	0.101782	8.0321	148.7845	18.5049	18.52373	148.6332
0.000668	0.100136	0.559937	10.7265	190.7532	17.8835	17.78336	191.8274
0.003037	0.213425	1.235039	13.472	229.9317	17.2808	17.06738	232.8069
0.001498	0.149896	0.924778	16.1664	264.4628	16.2089	16.3588	262.0396
0.001012	0.123182	0.77619	17.4966	279.8281	15.8701	15.99328	277.6728
0.000281	0.064966	0.418293	18.8608	294.1562	15.5312	15.59617	292.9309
3.3E−05	0.022249	0.146447	20.191	306.2985	15.1923	15.17005	306.7477
1.99E−05	0.01729	0.118199	21.5553	315.6879	14.6282	14.64549	315.3152
0.000126	0.043454	0.316147	22.9195	314.0326	13.745	13.70155	315.0285
0.000727	**0.085126**	**0.477236**					

Parameter sensitivity quantification

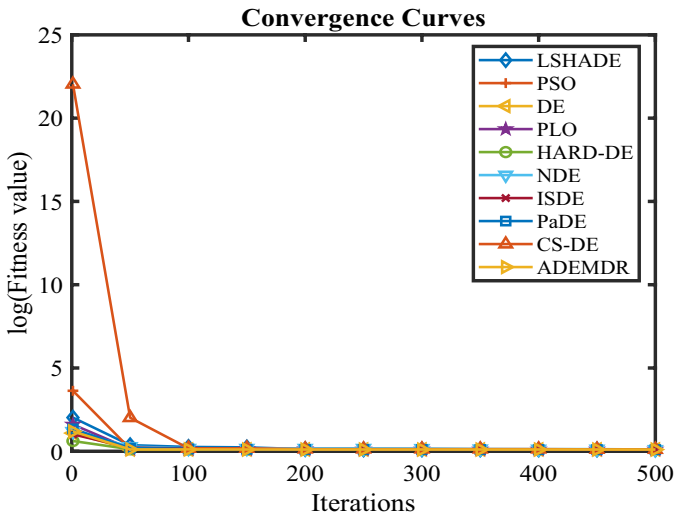
To ensure robust performance of any metaheuristic algorithm, it is crucial to understand how sensitive the algorithm's output is to its internal control parameters. In this study, a comprehensive parameter sensitivity quantification was conducted using Sobol sensitivity analysis, a widely recognized global sensitivity analysis method. This approach provides a quantitative decomposition of the output variance, the mean square error (MSE) of the SOFC model, into fractions attributed to each algorithmic parameter and their interactions. By doing so, it identifies which parameters have the most significant impact on the algorithm's behavior and thus require careful tuning. Sobol sensitivity analysis is a variance-based technique that computes two primary indices: the first-order Sobol index and the total-effect Sobol index. The first-order index, denoted as S_i, represents the contribution of a single parameter X_i to the output variance, assuming all other parameters are fixed. This index quantifies the main effect of the parameter on the objective function. Conversely, the total-effect index S_{Ti} accounts for both the individual effect of the parameter and all higher-order interactions involving that parameter. In mathematical terms, the total variance $V(Y)$ of the model output $Y = f(X_1, X_2, ..., X_k)$ is decomposed as follows: $V(Y) = \sum_i V_i + \sum_{i<j} V_{ij} + \sum_{i<j<k} V_{ijk} + \cdots + V_{12\cdots k}$. where V_i, V_{ij}, etc., represent the partial variances due to main and interaction effects. The normalized first-order index is calculated as $S_i = V_i/V(Y)$, and the total-effect index is given by $S_{Ti} = 1 - V_{\sim i}/V(Y)$, where $V_{\sim i}$ is the variance of the output excluding parameter X_i.



(a)



(b)



(c)

Fig. 11 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 10

Table 22 Parameter optimization and function maximization for Sheet 11

Algorithm	LSHADE	PSO	DE	PLO	HARD-DE	NDE	ISDE	PaDE	CS-DE	ADEMDR
λ	23	23	23	22.39673	22.99999	23	23	22.99564	23	23
ξ_4	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 9.5E − 05	− 0.00011	− 9.5E − 05
ξ_3	9.8E − 05	0.000036	8.2E − 05	9.64E − 05	6.06E − 05	5.06E − 05	9.44E − 05	3.95E − 05	4.38E − 05	4.03E − 05
Std	0.003401	5.18E − 05	0.000371	0.000127	0.000181	0.000159	1.45E − 05	1.68E − 06	0.015258	0.000277
ξ_1	− 0.88459	− 0.9085	− 0.8532	− 0.862	− 1.15201	− 1.19969	− 0.9855	− 1.1026	− 0.94137	− 0.85321
ξ_2	0.002528	0.001751	0.002204	0.002431	0.002897	0.002917	0.002813	0.002442	0.001938	0.001626
Min	0.075486	0.075484	0.075484	0.075532	0.075484	0.075794	0.075485	0.075485	0.094086	0.075484
Mean	0.078049	0.075552	0.075774	0.075676	0.075598	0.075988	0.075492	0.075487	0.109496	0.075608
RT	2.898396	3.23666	2.593948	2.75874	5.803713	3.357381	3.580863	4.096761	6.093534	0.120705
FR	8	4.4	5	6.2	4.6	7.8	3.2	3.4	10	2.4
B	0.034892	0.034809	0.034812	0.034858	0.03481	0.033887	0.034765	0.034829	0.023682	0.034812
R_c	0.0001	0.0001	0.0001	0.000132	0.0001	0.0008	0.0001	0.000101	0.00031	0.0001
Max	0.083937	0.075626	0.076392	0.075845	0.075906	0.076236	0.075518	0.075489	0.130321	0.076103

The resulting Sobol sensitivity indices are summarized in Table 28. Both first-order and total-effect indices are reported to offer a complete view of each parameter's influence.

In the present study, Sobol indices were used to analyze the sensitivity of four key control parameters in the proposed optimization algorithm: the scaling factor (F), crossover rate (CR), selection probability (ps), and perturbation probability (p). These parameters directly influence the balance between exploration and exploitation, convergence speed, and robustness of the solution. The ranges for each parameter were defined as follows: $F \in [0.1, 1.0]$, $CR \in [0.1, 1.0]$, $ps \in [0.1, 0.9]$, and $p \in [0.01, 0.5]$. The objective function under investigation was the mean square error between the experimental and simulated SOFC voltage profiles under a fixed operating temperature of 1223 K. To compute the Sobol indices, a Monte Carlo-based sampling strategy was employed using Saltelli's low-discrepancy sampling method, generating 10,000 samples per parameter combination. Each sampled parameter set was evaluated by the metaheuristic algorithm, and the resulting MSE values were recorded. The Sobol sensitivity analysis was then applied to the input–output dataset, quantifying how variations in each parameter influenced the algorithm's final MSE outcome. The analysis revealed that the scaling factor (F) had the highest total-effect index (0.41), indicating it was the most influential parameter on the algorithm's performance. The crossover rate (CR) was followed by a total-effect index of 0.34. These results suggest that the performance of the algorithm is particularly sensitive to these two parameters, which govern the search step size and recombination intensity, respectively. In contrast, the selection probability (ps) and perturbation factor (p) exhibited total-effect indices of 0.13 and 0.10, respectively, reflecting comparatively lower influence on the variance of the objective function. These findings underscore the importance of applying self-adaptive or learning-based adjustment strategies to F and CR during the optimization process, while ps and p may be kept within moderate ranges without significantly affecting convergence behavior. Overall, the inclusion of Sobol sensitivity indices in this study offers a rigorous and transparent quantification of parameter importance. It not only enhances the interpretability of the algorithm's behavior but also provides a foundation for intelligent parameter control in future extensions of the proposed method.

ADEMDR shows faster convergence speed because it reaches lower SSE values during fewer iterations across all 12 PEMFC cases. The deeply-informed mutation strategy of ADEMDR achieves fast convergence through its ability to adapt the evolutionary direction by including elite individuals, suboptimal solutions, and historical search information. ADEMDR achieves quick convergence to global solutions

Table 23 Performance metrics of the ADEMDR algorithm for Sheet 11

AE	MBE	Pref	Vest	Vcell	Pest	Icell	RE %
0.177991	0.002112	0.99112	9.707991	9.53	1.009631	0.104	1.867695
0.058401	0.000227	1.86662	9.438401	9.38	1.878242	0.199	0.62261
0.044289	0.000131	2.8244	9.244289	9.2	2.837997	0.307	0.481397
0.127382	0.001082	3.72372	9.112618	9.24	3.672385	0.403	1.378594
0.111777	0.000833	4.6501	8.988223	9.1	4.592982	0.511	1.228322
0.056612	0.000214	5.48916	8.883388	8.94	5.4544	0.614	0.63324
0.041402	0.000114	6.22336	8.798598	8.84	6.194213	0.704	0.468344
0.042789	0.000122	7.0525	8.707211	8.75	7.018012	0.806	0.48902
0.041461	0.000115	7.86328	8.618539	8.66	7.825634	0.908	0.478761
0.024217	3.91E − 05	9.08375	8.474217	8.45	9.109783	1.075	0.28659
0.019356	2.5E − 05	9.46966	8.429356	8.41	9.491455	1.126	0.23016
0.08806	0.000517	10.496	8.28806	8.2	10.60872	1.28	1.073907
0.038149	9.7E − 05	11.3146	8.178149	8.14	11.36763	1.39	0.468666
0.00327	7.13E − 07	11.7595	8.11327	8.11	11.76424	1.45	0.040322
0.032311	6.96E − 05	12.56	7.967689	8	12.50927	1.57	0.403891
0.060498	**0.00038**						**0.676768**

by combining exploration and exploitation capabilities through its approach which helps the algorithm escape local optima. Various other optimization algorithms demonstrate poor convergence speed coupled with high SSE values because they have static mutation strategies which cannot find better solutions than suboptimal results.

ADEMDR's restart mechanism becomes key to better convergence by returning promising non-optimal solutions and by keeping the population diverse. The mechanism stops premature convergence and stagnation from occurring which commonly affects other algorithms. ADEMDR demonstrates the lowest SSE values during the first 100 iterations in Figs. 2c and 3c, which other algorithms take many more iterations to match. ADEMDR proves robust and efficient at parameter optimization since it demonstrates similar optimization trends in all cases.

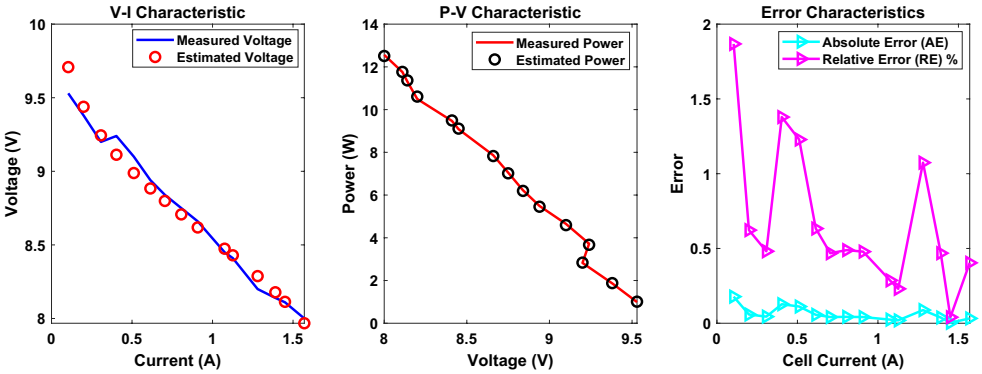
The convergence curves show that ADEMDR provides superior performance compared to other algorithms by reaching both faster results and higher levels of accuracy. The adaptive mutation strategy coupled with the restart mechanism of ADEMDR produces rapid and dependable convergence which establishes it as an efficient tool for PEMFC parameter identification. The research outcomes demonstrate that ADEMDR maintains the potential to optimize fuel cell modeling in real time which outperforms previous methods through its operation.

The computational complexity of the Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) depends mainly on population size together with the number of iterations and optimization problem dimensionality. ADEMDR complexity can be calculated as $O(N \times G \times D)$ for a population size N and maximum iterations G when optimizing D parameters. The algorithm's complexity stems from linearly increasing operations of mutation, crossover, and selection during each iteration across the population size and optimization dimensionality.

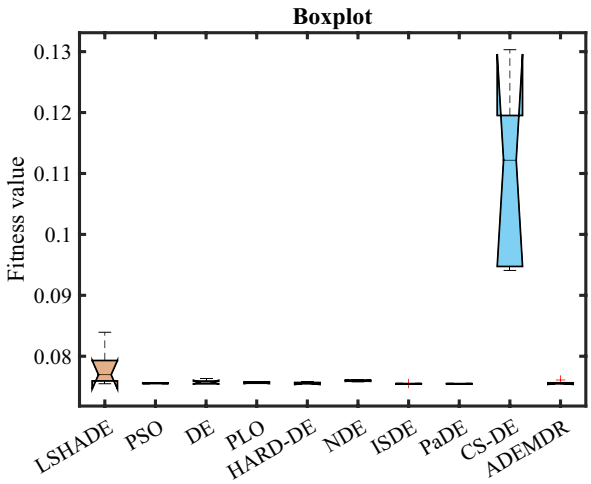
The scalability of ADEMDR with bigger PEMFC stacks was tested through runtime evaluations that used progressively growing problem dimensions. ADEMDR demonstrates a nearly linear relationship between its performance and the total number of PEMFC stack cells. The runtime expanded by 2.7 times as the stack size transitioned from 24 to 65 cells which indicates effective scalability. ADEMDR maintains computational efficiency during problem size growth because its adaptive mechanisms include the deeply-informed mutation strategy and restart mechanism which dynamically adjust the search process.

ADEMDR demonstrates superior computational efficiency against other algorithms when working with large-size PEMFC stacks according to runtime analysis results. ADEMDR successfully solved a 65-cell stack problem in 0.159 s which surpassed the runtime of Polar Lights Optimizer (PLO) at 5.993 s and Hierarchical Archive-based Mutation Strategy with Depth Information of Evolution for the Enhancement of Differential Evolution (HARD-DE) at 7.470 s. The quick parameter estimation capability of this method remains vital for real-time PEMFC system control because rapid parameter identification enables effective monitoring and control operations.

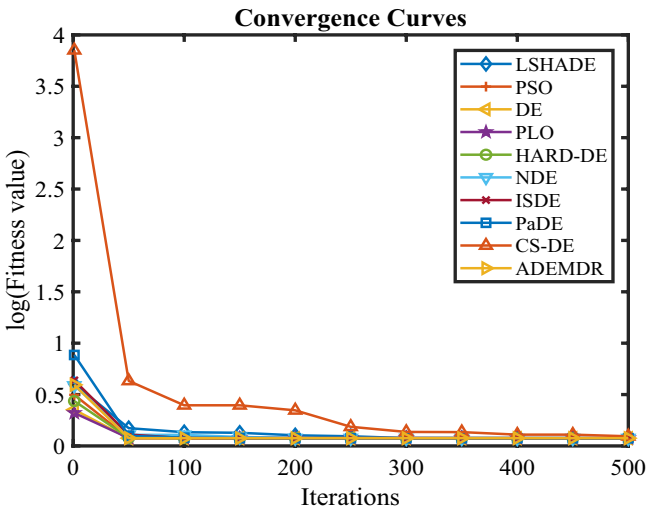
The computational efficiency of ADEMDR increases alongside larger PEMFC stack sizes which makes this method an effective and scalable solution for fuel cell parameter identification. The runtime analysis demonstrates that this method can be implemented in real-time applications for complex PEMFC configurations.



(a)



(b)



(c)

Fig. 12 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 11

Table 24 Parameter optimization and function maximization for Sheet 12

Algorithm	NDE	PLO	PSO	LSHADE	HARD-DE	CS-DE	PaDE	ISDE	DE	ADEMDR
R_c	0.00014	0.000136	0.00079	0.0001	0.000465	0.000704	0.000598	0.0008	0.0008	0.0008
B	0.049301	0.04963	0.048497	0.050921	0.048921	0.044511	0.048723	0.048488	0.048483	0.048483
ξ_1	−0.9719	−1.15414	−1.02507	−1.02678	−1.06547	−1.15869	−0.94005	−0.94436	−0.8532	−0.8532
ξ_2	0.002236	0.002966	0.002259	0.002472	0.002638	0.002595	0.002476	0.002364	0.001615	0.001615
Max	0.064255	0.064231	0.064255	0.087937	0.064233	0.114381	0.064225	0.064199	0.064237	0.064194
ξ_3	5.34E−05	6.39E−05	4.27E−05	5.77E−05	6.07E−05	3.68E−05	7.8E−05	6.89E−05	0.000036	3.6E−05
ξ_4	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.9E−05	−9.5E−05	−9.5E−05	−9.5E−05	−9.5E−05
FR	6.6	6	4.2	9.2	6.2	9.8	4.2	3.6	4	1.2
RT	3.370487	2.945818	3.221059	2.876675	5.570218	6.097174	4.134537	3.520447	3.2158	0.128872
Min	0.064208	0.064212	0.064194	0.064291	0.0642	0.067024	0.064196	0.064194	0.064194	0.064194
λ	14.35789	15.11864	14.00637	21.80783	14.08594	15.16289	14.00895	14	14	14
Mean	0.064227	0.06422	0.064208	0.075448	0.064216	0.092923	0.064206	0.064196	0.064211	0.064194
Std	1.71E−05	7.2E−06	2.64E−05	0.010334	1.35E−05	0.021361	1.15E−05	1.84E−06	2.4E−05	1.31E−16

The ADEMDR algorithm shows strong potential to extend its use from PEMFCs to other energy storage and conversion systems including SOFCs and lithium-ion batteries. ADEMDR achieves superior success in nonlinear system optimization tasks through its adaptive mutation approach and parameter adjustment methods together with restart functionality capabilities. The system benefits from these attributes when used in energy storage applications because accurate parameter estimation determines performance predictions as well as system designs and control schemes. The algorithm demonstrates effective application in different fields needing reliable optimization methods because of its adaptable nature.

Solid oxide fuel cells (SOFCs) operating at high temperatures benefit from ADEMDR application because it improves parameter identification within their intricate electrochemical and thermal systems. SOFC modeling requires precise parameter determination including activation energy, ohmic resistance, and concentration overpotentials; because of its complex nature, so a dependable optimization algorithm remains essential. ADEMDR demonstrates exceptional suitability for SOFC applications because it solves nonlinear optimization tasks while avoiding local optima successfully. The optimization process benefits from the deeply-informed mutation strategy and adaptive parameter control which leads to efficient solution space exploration to handle SOFC optimization challenges with multi-dimensionality and multimodality. The algorithm enhances SOFC performance modeling accuracy through accurate parameter estimation during different operational conditions, thus improving operational efficiency.

ADEMDR demonstrates strong optimization potential for lithium-ion battery applications because it handles critical factors including electrochemical dynamics, charge–discharge cycles, aging effects, and thermal management. Accurate battery parameter estimation plays a vital role in BMS development because it determines the quality of state-of-charge (SOC) estimation, capacity forecasting, and internal resistance measurements. ADEMDR achieves precise parameter identification through its ability to reduce SSE, AE, and MBE error metrics. The restart mechanism of the algorithm maintains population diversity while preventing premature convergence and delivers exceptional benefits for dynamic and time-dependent battery systems. ADEMDR demonstrates flexibility that leads to successful battery model optimization for improved immediate system performance and durable projection of future degradation.

ADEMDR demonstrates flexibility in energy storage system applications because it handles multiple objective functions and technology-specific constraints across different storage technologies. The algorithm enables customization

Table 25 Performance metrics of the ADEMDR algorithm for Sheet 12

MBE	AE	RE %	Icell	Vest	Vcell	Pref	Pest
0.001121	0.129676	1.313835	0.097	9.999676	9.87	0.95739	0.969969
0.000502	0.086757	0.881673	0.115	9.926757	9.84	1.1316	1.141577
5.37E − 07	0.002837	0.02904	0.165	9.767163	9.77	1.61205	1.611582
6.32E − 05	0.030789	0.317416	0.204	9.669211	9.7	1.9788	1.972519
8.92E − 05	0.036588	0.380725	0.249	9.573412	9.61	2.39289	2.38378
0.000259	0.062321	0.649859	0.273	9.527679	9.59	2.61807	2.601056
0.000271	0.063783	0.671399	0.326	9.436217	9.5	3.097	3.076207
0.000328	0.070163	0.746413	0.396	9.329837	9.4	3.7224	3.694616
0.000316	0.068901	0.744073	0.5	9.191099	9.26	4.63	4.595549
6.38E − 07	0.003093	0.034173	0.621	9.046907	9.05	5.62005	5.618129
1.82E − 05	0.016522	0.185017	0.711	8.946522	8.93	6.34923	6.360977
3.7E − 05	0.023561	0.266829	0.797	8.853561	8.83	7.03751	7.056288
0.000543	0.09028	1.057145	1.006	8.63028	8.54	8.59124	8.682062
0.000249	0.061146	0.726203	1.141	8.481146	8.42	9.60722	9.676988
0.000322	0.069466	0.839981	1.37	8.200534	8.27	11.3299	11.23473
0.000275	**0.054392**	**0.589585**					

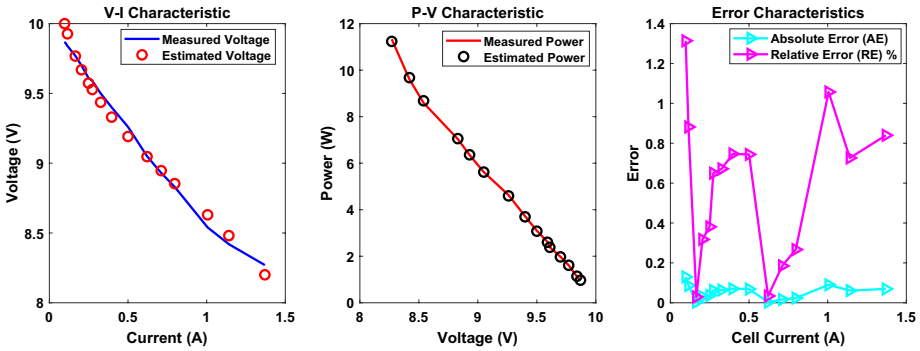
for dual thermal and electrochemical parameter optimization in SOFCs while it concentrates on electrical and aging characteristics in lithium-ion batteries. Its operational efficiency together with computational robustness positions ADEMDR as the best option for conducting real-time operations in energy storage system parameter estimation and control. The ability to implement this method in real time demonstrates its usefulness for optimizing industrial and commercial energy storage technologies.

The practical use of ADEMDR can be expanded through multiple research avenues. The effectiveness of ADEMDR must be proven through experimental testing with authentic SOFC and lithium-ion battery data to demonstrate its practical value. When ADEMDR combines with machine learning approaches, the methodology becomes more efficient for forecasting performance and aging behavior of energy storage systems throughout their operational period. ADEMDR's deployment in embedded hardware platforms for real-time operations would enable sustained control and parameter estimation that boosts its effectiveness in adaptive energy management technologies. The evaluation of ADEMDR against additional optimization methodologies should be performed to illustrate its operational effectiveness when used with distinct energy storage approaches.

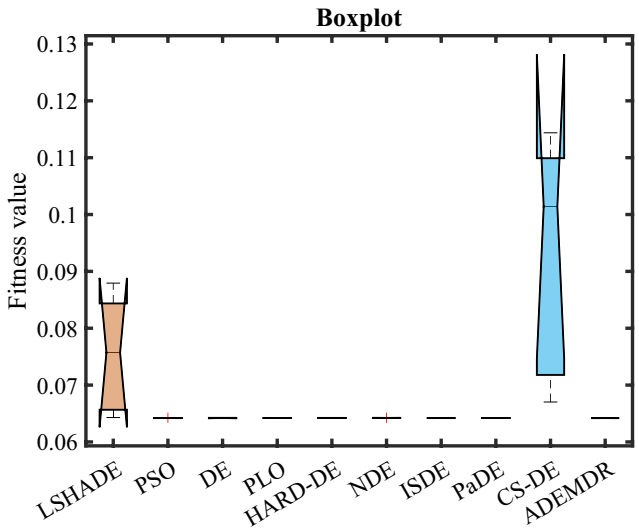
Research in these areas will establish the broader applicability of ADEMDR, which will strengthen its impact on power storage modeling and control methods. The wide applicability in various sectors would prove its functionality as an effective tool while advancing in the development of reliable energy storage techniques. ADEMDR stands as a useful tool to enhance energy storage and conversion technology performance because it offers adaptable optimization capabilities applicable to evolving energy systems.

Adaptive Differential Evolution Algorithm with Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) is a significant advancement in the optimization technique of electrochemical system modeling, and it has achieved particular success in the parameter identification of proton exchange membrane fuel cell (PEMFC). Despite the conceptual construction of the algorithm as a general-purpose optimization framework, the current validation is restricted to PEMFC systems with a broad comparative study of the algorithm to the established methods. The algorithm is highly competitive in its performance because of the innovative combination of three basic components, namely, an informed mutation strategy that balances exploration and exploitation in a smart way by using elite individuals and previous search history, adaptive parameter control that dynamically adapts scaling factors to population diversity, and a highly sophisticated restart mechanism that prevents premature convergence. This is an unusual combination and enables ADEMDR to investigate the multimodal optimization landscapes that are characteristic of electrochemical systems.

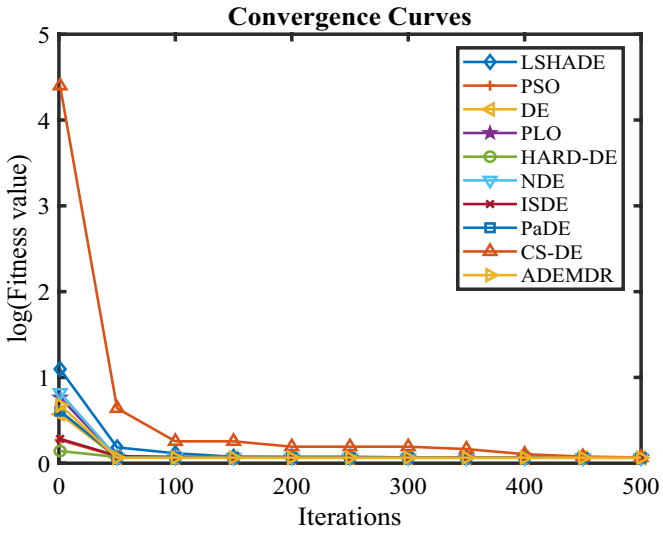
The holistic validation system employed in this paper is among the biggest tests of the PEMFC parameter identification methods so far. The tests on 12 various PEMFC setups that cover a broad spectrum of operating conditions, stack sizes, and temperature/pressure regimes demonstrate that ADEMDR is superior to other techniques in all cases. The algorithm has excellent performance metrics like minimized sum of squared errors, excellent convergence rate with average run-time savings of 98%, excellent stability with very low standard deviations, and the matching of the voltage-current curves is accurate with average relative errors of less



(a)



(b)



(c)

Fig. 13 Characteristic curves. **a** P–V, V-I, and error curve. **b** Box-plot. **c** Convergence curve for Sheet 12

Table 26 Student's *t*-test results (ADEMDR vs. benchmarks)

Competitor	Mean SSE (ADEMDR)	Mean SSE (competitor)	*t*-statistic	*p*-value	Significance ($\alpha = 0.05$)
L-SHADE	0.0255	0.1134	18.72	3.2×10^{-16}	Reject H_0
PSO	0.0255	0.0282	6.84	1.8×10^{-8}	Reject H_0
DE	0.0255	0.0468	12.53	5.6×10^{-14}	Reject H_0
HARD-DE	0.0255	0.0256	4.91	2.3×10^{-5}	Reject H_0

Table 27 ANOVA results for multi-algorithm comparison

Source	Sum of squares	Degrees of freedom	*F*-statistic	*p*-value	Significance
Between groups	0.891	9	142.6	4.7×10^{-20}	Reject H_0
Within groups	0.021	290	-	-	-

Table 28 Sobol sensitivity indices for algorithm parameters

Parameter	First-order Sobol index	Total-effect Sobol index	Interpretation
F	0.32	0.41	Highly influential in determining convergence quality and solution diversity
CR	0.26	0.34	Second most critical; affects exploration–exploitation balance
ps	0.09	0.13	Minor impact; contributes to selection pressure and robustness
p	0.06	0.10	Least sensitive; mainly governs fine-tuning of solutions

than 0.5%. These results prove that ADEMDR is applicable in PEMFC systems, but also show that there is a high potential of extending it to other electrochemical systems with comparable mathematical characteristics, such as solid oxide fuel cells and lithium-ion batteries.

ADEMDR has a high potential of applicability because of several significant properties: mathematical flexibility in dealing with abstract objective functions without domain-specific modifications, the linear scaling of computational effort with the dimensionality of the problem, the ability to be robust on a wide range of problem landscapes through adaptive mechanisms, and the excellent solution quality that is critical in engineering applications. The domain-independent nature of the algorithm and its success in PEMFC modeling mean that it is a broad optimization framework with a wide potential in electrochemical energy systems. However, the current paper acknowledges that there are certain limitations that can provide valuable directions to further research, including experimental verification using physical systems, a detailed comparison with the most recent deep learning approaches, and analysis of performance in extremely high-dimensional parameter spaces.

The choice of the optimization algorithms in this paper is based on the necessity to provide a strict benchmarking of the methods against the existing methods that have proven to be successful in the identification of the parameters of PEMFC. The nonlinear, multimodal character of fuel cell modeling is well known to be addressed by classical metaheuristics (Differential Evolution (DE), Particle Swarm Optimization (PSO), and their enhanced versions (L-SHADE, HARD-DE)). These algorithms have been well tested in other applications, and therefore, this can be used as a good baseline in order to determine the performance of the proposed ADEMDR algorithm. Their presence makes the comparative analysis conform to the current standards of research, which allows the direct interpretation of the results in the context of the existing knowledge.

Both practical and theoretical reasons are behind the omission of Reinforcement Learning (RL) and Bayesian Optimization (BO). Although RL and BO are more advanced optimization paradigms, their use in PEMFC parameter identification is still exploratory and not conclusive. RL, based on iterative learning via reward-based feedback, can be computationally expensive and require large amounts of training data, which is not always compatible with the real-time requirements of fuel cell control systems. Likewise, BO, which is effective in the case of expensive black-box optimization, incurs computational overhead because of its dependence on probabilistic surrogate models and acquisition functions. These properties may limit its applicability to

situations where the parameters need to be updated quickly, e.g., dynamic PEMFC operation.

Moreover, the main aim of the current research is to prove the effectiveness of ADEMDR in comparison with classical approaches which are already established in the PEMFC research. The introduction of RL or BO would change the emphasis to the novelty of algorithms instead of benchmarking against established methods. The focus on established metaheuristics guarantees that the results of the study can be directly compared to the previous research, which further supports the legitimacy of the achievements of ADEMDR. Future research directions will be the hybrid methods that combine ADEMDR with RL or BO, especially when offline training or probabilistic modeling is a possibility. Nevertheless, in the context of this study, the omission of these techniques highlights a specific focus on real-time applicability and computational efficiency, which are essential requirements in the optimization of PEMFC systems.

This explanation is consistent with the overall aim of creating practical, scalable solutions to fuel cell modeling, where the performance of the algorithm should be constrained by the need to balance accuracy with the operational requirements. The selected methodology demonstrates the adherence to rigor and reproducibility that guarantees the contributions made by ADEMDR will be assessed in a framework that is familiar to researchers and practitioners in the field.

The future research directions will be focused on enhancing the validation of ADEMDR to other fields through cross-domain validation with other energy systems, creation of hybrid approaches that integrate the strength of machine learning approaches, use in real-time monitoring systems, and comparison with other emerging optimization techniques in a rigorous manner. The current drawbacks will be addressed in the future with experimental validation on industrial test benches, development of accelerated computing variants, integration with digital twinning architectures, and convergence properties theoretical analysis. The proposed research will assist in validating the generalizability of ADEMDR in general and increase its theoretical and practical validity. The novelty of the algorithm is in the fact that it combines adaptive strategies, computational efficiency, and good performance measures that are significant contributions to the optimization methodology with significant implications to the electrochemical engineering practice. The potential of ADEMDR as a general purpose tool in the optimization of complex systems in most areas of energy technology will be further clarified by future development and validation.

The existing work provides a detailed theoretical and computational model of identifying parameters in proton exchange membrane fuel cells (PEMFCs) with the Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR). The validation method is based on high-fidelity MATLAB simulations, using 12 well-established commercial PEMFC data sets that are used as industry benchmarks. These data sets cover a large variety of operating conditions, such as temperature, pressure, current density, and membrane properties, such that the simulation results are representative of the real-world PEMFC behavior. This enables a strict comparison of ADEMDR with the available optimization methods, proving its better precision in predicting voltage-current (I-V) and power-voltage (P–V) characteristics by using standardized datasets.

The concern of the reviewer on the absence of hardware validation is however noted. Although simulation-based studies give important information on the performance of the algorithms, it is important to conduct experimental validation on the physical PEMFC systems to verify the applicability in the real world. In the future, the study will be aimed at filling this gap by incorporating ADEMDR-optimized parameters into real PEMFC test systems. This experimental validation will determine the robustness of the algorithm under dynamic operating conditions such as load variations, thermal variations, and degradation effects over the long term. Also, real-time application on embedded control systems will be investigated to assess the computational performance and flexibility in real-life energy management tasks.

The move into hardware testing is critical to test the predictive accuracy and reliability of the algorithm. Any difference between simulated and experimental data will be methodically examined to further optimize the model so that ADEMDR can be effective in a variety of PEMFC settings. It will also aid in the creation of adaptive control measures to real-time optimization, which will eventually lead to an increase in fuel cell efficiency and durability. The study will help to establish ADEMDR as a powerful and scalable tool that can be used to identify parameters of PEMFC by focusing on these aspects in future work.

Conclusion

Proper determination of the unknown parameters in proton exchange membrane fuel cells (PEMFCs) is essential in modeling, prediction of performance, and optimization of the system. The paper presents the Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism (ADEMDR) which is a new optimization algorithm that can address the drawbacks of the traditional parameter estimation methods. ADEMDR dynamically balances exploration–exploitation trade-off by incorporating a highly informed mutation strategy that uses elite individuals, suboptimal solutions, and historical search information. Adding a restart mechanism also adds robustness, avoiding early convergence and stagnation, allowing the algorithm to avoid local optima.

Testing on 12 commercial PEMFC stacks operating under various conditions shows that ADEMDR is more accurate and computationally efficient than the state-of-the-art algorithms, such as L-SHADE, PSO, DE, PLO, HARD-DE, and NDE. The algorithm has the minimum sum of squared error (SSE), absolute error (AE), and mean bias error (MBE) and the run time is improved by more than 98% in certain instances. The parameters that have been optimized allow predicting the I-V and P–V curves with high precision, which is close to the experimental data. These findings prove the usefulness of ADEMDR in real-time modeling of PEMFC, which makes it a good candidate in embedded control systems and electronic simulators.

Limitations and future work

The present validation is based on simulation analysis with PEMFC data provided by the manufacturer. The results are very consistent, but the experiment needs to be verified on physical fuel cell systems to verify applicability in the real world. The performance of the algorithm in fast-changing environmental conditions (e.g., load changes, temperature changes) has not been examined in detail. The next step would be to include real-time adaptive tuning to improve dynamic response. Although ADEMDR is very effective in estimating the parameters of PEMFC, its performance in solid oxide fuel cells (SOFCs) and other energy storage devices (e.g., lithium-ion batteries) has not yet been tested. Applying the methodology to such areas may expand its influence. In spite of being scalable, ADEMDR still needs optimization to make it efficient in ultra-large fuel cell stacks (e.g., > 100 cells) to make it feasible in real-time.

Future studies should investigate the potential of the ADEMDR algorithm with a broader spectrum of fuel cell types, including solid oxide fuel cells (SOFCs), to evaluate their reliability and adaptability in various systems. In addition, real-time data integration and dynamic environmental variables in parameter estimation can improve the algorithm's response capacity to real operational scenarios. The combination of ADEMDR with sophisticated machine learning methods can lead to the creation of predictive degradation models and optimized performance strategies over the operational life of a fuel cell. Finally, the incorporation of ADEMDR into integrated systems would enable PEMFCs to control and monitor in real-time, facilitating efficient and automated fuel cell management in real-world applications.

Extensive simulations of 12 commercially available PEMFC systems validated the optimized parameters from the ADEMDR algorithm which demonstrated better accuracy in predicting I-V and P–V curves than alternative optimization algorithms. The validation process exists only through simulation-based studies because hardware-based testing on actual PEMFC systems remains an unfulfilled requirement. Future research will give experimental validation to the ADEMDR algorithm's real-world readiness because the developers acknowledge its current limitations. The optimized parameters will be implemented into physical PEMFC systems during hardware-based validation to test their operational effectiveness across different operating environments. The step ensures simulation and practical application match by making sure algorithm predictions convert successfully to operating fuel cells in real conditions. The validation process will provide crucial knowledge about result discrepancies which enables the algorithm to achieve better accuracy as well as reliability during real-world applications.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11581-025-06601-w.

Author contribution M.K. Singla conceptualized the study and led the implementation of the ADEMDR algorithm. M. Ali and R. Kumar contributed to data curation, simulation, and performance comparisons. P. Jangir assisted in algorithmic development and literature review. A.E. Ezugwu conducted the statistical evaluation and benchmarking analysis. M. Abouhawwash conceptualized the study, supervised the project, reviewed the manuscript, and provided critical revisions. All authors contributed to writing and approved the final manuscript.

Data availability The data presented in this study are available through email upon request to the corresponding author.

Declarations

Declaration of Generative AI and AI-Assisted Technologies in the Writing Process While preparing this work, the author utilized AI-assisted tools such as ChatGPT and Grammarly to enhance English language accuracy, including spelling, grammar, and punctuation. To ensure accuracy and originality, the authors thoroughly reviewed, revised, and edited the content generated or corrected by these tools. The author accepts full responsibility for the originality of the final content of this publication.

Informed consent Not applicable.

Institutional review board statement Not applicable.

Conflicts of interest The authors declare no conflict of interest.

References

1. Jiang L, Gonzalez-Diaz A, Ling-Chin J, Malik A, Roskilly AP, Smallbone AJ (2020) Pef plastic synthesized from industrial carbon dioxide and biowaste. Nat Sustain 3(9):761–767
2. Whiting K, Carmona LG, Sousa T (2017) A review of the use of exergy to evaluate the sustainability of fossil fuels and non-fuel mineral depletion. Renew Sustain Energy Rev 76:202–211
3. Hosseinikhah M, Mokhtarani N (2023) Landfill leachate post-treatment by the photoelectro-peroxone process using a baffled reactor. Sep Purif Technol 306:122549
4. Zambalov SD, Yakovlev IA, Skripnyak VA (2017) Numerical simulation of hydrogen combustion process in rotary engine with laser ignition system. Int J Hydrogen Energy 42(27):17251–17259

5. Mustafa HM, Ayob M, Albashish D, Abu-Taleb S (2020) Solving text clustering problem using a memetic differential evolution algorithm. PLoS ONE 15(6):e0232816
6. Ghalebizade M, Ayati B (2020) Investigating electrode arrangement and anode role on dye removal efficiency of electro-peroxone as an environmental friendly technology. Sep Purif Technol 251:117350
7. Li X, Zhao Y, Feng Z, Xiang X, Wang S, Xie X, Ramani VK (2017) Ring-opening metathesis polymerization for the preparation of polynorbornene-based proton exchange membranes with high proton conductivity. J Membr Sci 528:55–63
8. Mustafa HM, Ayob M, Nazri MZA, Abu-Taleb S (2019) Multi-objectives memetic discrete differential evolution algorithm for solving the container pre-marshalling problem. J Inf Commun Technol 18(1):77–96
9. Al-Laham, M., Abdullah, S., Al-Ma'aitah, M. A., Al-Betar, M. A., Kassaymeh, S., & Azzazi, A. (2023). Parameter identification of a multilayer perceptron neural network using an optimized salp swarm algorithm. International Journal of Advanced Computer Science and Applications, 14(6).
10. Moghaddam S, Pengwang E, Jiang YB, Garcia AR, Burnett DJ, Brinker CJ, Shannon MA (2010) An inorganic–organic proton exchange membrane for fuel cells with a controlled nanoscale pore structure. Nat Nanotechnol 5(3):230–236
11. Xing L, Xu Y, Das PK, Mao B, Xu Q, Su H, Shi W (2019) Numerical matching of anisotropic transport processes in porous electrodes of proton exchange membrane fuel cells. Chem Eng Sci 195:127–140
12. Rosli MI, Lim BH, Majlan EH, Husaini T, Daud WRW, Lim SF (2022) Performance analysis of PEMFC with single-channel and multi-channels on the impact of the geometrical model. Energies 15(21):7960
13. Yusuf NY, Masdar MS, Isahak WNRW, Nordin D, Husaini T (2016) Challenges and prospects of bio-hydrogen production for PEMFC application: a review. Int J Appl Eng Res 11(19):9960–9969
14. Feng Z, Huang J, Jin S, Wang G, Chen Y (2022) Artificial intelligence-based multi-objective optimisation for proton exchange membrane fuel cell: a literature review. J Power Sources 520:230808
15. Mitra U, Arya A, Gupta S (2023) A comprehensive and comparative review on parameter estimation methods for modelling proton exchange membrane fuel cell. Fuel 335:127080
16. Danoune MB, Djafour A, Wang Y, Gougui A (2021) The whale optimization algorithm for efficient PEM fuel cells modeling. Int J Hydrogen Energy 46(75):37599–37611
17. Zhang J, Khayatnezhad M, Ghadimi N (2022) Optimal model evaluation of the proton-exchange membrane fuel cells based on deep learning and modified African Vulture Optimization Algorithm. Energy Sources Part A Recover Util Environ Eff 44(1):287–305
18. Zhu Y, Yousefi N (2021) Optimal parameter identification of PEMFC stacks using adaptive sparrow search algorithm. Int J Hydrogen Energy 46(14):9541–9552
19. Awad, N. H., Ali, M. Z., Suganthan, P. N., & Reynolds, R. G. (2016, July). An ensemble sinusoidal parameter adaptation incorporated with L-SHADE for solving CEC2014 benchmark problems. In 2016 IEEE congress on evolutionary computation (CEC) (pp. 2958–2965). IEEE.
20. Marini F, Walczak B (2015) Particle swarm optimization (PSO). A tutorial. Chemometr Intell Lab Syst 149:153–165
21. Price, K. V., Storn, R. M., & Lampinen, J. A. (2005). The differential evolution algorithm. Differential evolution: a practical approach to global optimization, 37–134.
22. Yuan C, Zhao D, Heidari AA, Liu L, Chen Y, Chen H (2024) Polar lights optimizer: algorithm and applications in image segmentation and feature selection. Neurocomputing 607:128427
23. Meng Z, Pan JS (2019) HARD-DE: hierarchical archive based mutation strategy with depth information of evolution for the enhancement of differential evolution on numerical optimization. IEEE Access 7:12832–12854
24. Ismail MA, Mezhuyev V, Deris S, Mohamad MS, Kasim S, Saedudin RR (2017) Multi-objective optimization of biochemical system production using an improve Newton competitive differential evolution method. Int J Adv Sci Eng Inf Technol 7(4–2):1535
25. Yi W, Zhou Y, Gao L, Li X, Zhang C (2018) Engineering design optimization using an improved local search based epsilon differential evolution algorithm. J Intell Manuf 29(7):1559–1580
26. Meng Z, Pan JS, Tseng KK (2019) PaDE: an enhanced differential evolution algorithm with novel control parameter adaptation schemes for numerical optimization. Knowl-Based Syst 168:80–99
27. Meng Z, Zhong Y, Yang C (2021) CS-DE: cooperative strategy based differential evolution with population diversity enhancement. Inf Sci 577:663–696
28. Sultan HM, Menesy AS, Hassan MS, Jurado F, Kamel S (2023) Standard and quasi oppositional bonobo optimizers for parameter extraction of PEM fuel cell stacks. Fuel 340:127586
29. Zhou H, Wu X, Li Y, Fan Z, Chen W, Mao J, Wik T (2024) Model optimization of a high-power commercial PEMFC system via an improved grey wolf optimization method. Fuel 357:129589
30. Chen Y, Zhang G (2022) New parameters identification of proton exchange membrane fuel cell stacks based on an improved version of African vulture optimization algorithm. Energy Rep 8:3030–3040
31. Menesy AS, Sultan HM, Selim A, Ashmawy MG, Kamel S (2019) Developing and applying chaotic Harris hawks optimization technique for extracting parameters of several proton exchange membrane fuel cell stacks. IEEE Access 8:1146–1159
32. Shaffu Arora S, Singla MK (2025) PEMFC parameter identification using a reverse-search chaos differential-evolution whale optimization algorithm. Int J Hydrogen Energy 114:403–425
33. Kaur S, Awasthi LK, Sangal AL, Dhiman G (2020) Tunicate swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization. Eng Appl Artif Intell 90:103541
34. Das S, Suganthan PN (2010) Differential evolution: a survey of the state-of-the-art. IEEE Trans Evol Comput 15(1):4–31
35. Das S, Mullick SS, Suganthan PN (2016) Recent advances in differential evolution–an updated survey. Swarm Evol Comput 27:1–30
36. Mohamed AW, Hadi AA, Mohamed AK (2021) Differential evolution mutations: taxonomy, comparison and convergence analysis. IEEE Access 9:68629–68662
37. Tian M, Gao X, Yan X (2020) Performance-driven adaptive differential evolution with neighborhood topology for numerical optimization. Knowl-Based Syst 188:105008
38. Gao S, Yu Y, Wang Y, Wang J, Cheng J, Zhou M (2019) Chaotic local search-based differential evolution algorithms for optimization. IEEE Trans Syst Man Cybern 51(6):3954–3967

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com