Temperature Prediction for Photovoltaic Inverters Using Particle Swarm Optimization-Based Symbolic Regression: A Comparative Study

Fabian Alonso Lara-Vargas[1], Jesús Águila-León[2], Carlos Vargas-Salgado[3], Oscar J. Suarez[4]
Universidad Pontificia Bolivariana, Montería, Colombia[1]
University Institute of Energetic Engineering, Universitat Politècnica de València, Valencia, Spain[1,3]
Department of Water and Energy Studies, Universidad de Guadalajara, Guadalajara, Mexico[2]
Mechatronics Engineering Department, Universidad de Pamplona, Pamplona, Colombia[4]

***Abstract*—Accurate temperature modeling is crucial for maintaining the efficiency and reliability of solar inverters. This paper presents an innovative application of symbolic regression based on particle swarm optimization (PSO) for predicting the temperature of photovoltaic inverters, offering a novel approach that balances accuracy and computational efficiency. The study evaluates the performance of a PSO-based symbolic regression model compared to multiple linear regression (MLR) and a symbolic regression model based on genetic algorithms (GA). The models were developed using a dataset that included inverter temperature, active power, and DC bus voltage, collected over a year in hourly intervals from a rooftop photovoltaic system in a tropical region. The dataset was divided, with 70% used for training and the remaining 30% for testing. The symbolic regression model based on PSO demonstrated superior performance, achieving lower values of the root mean square error (RMSE) and mean absolute error (MAE) of 3.97 and 3.31, respectively. Furthermore, the PSO-based model effectively captured the nonlinear relationships between variables, outperforming the MLR model. It also exhibited greater computational efficiency, requiring fewer iterations than traditional symbolic regression approaches. These findings open new possibilities for real-time monitoring of photovoltaic inverters and suggest future research directions, such as generalizing the PSO model to different environmental conditions and inverter types.**

Keywords*—Particle swarm optimization; photovoltaic inverters; multiple linear regression; symbolic regression; temperature prediction

I. Introduction

Photovoltaic systems play a crucial role in reducing the carbon footprint compared to traditional energy sources. These systems provide a sustainable alternative that helps reduce future CO_2 emissions, which is critical to combating global warming [1]. Solar inverters are essential for photovoltaic systems because they convert direct current (DC) from solar panels into alternating current (AC) for commercial and residential use [2]. In the context of electrical systems, solar inverters are critical in facilitating grid integration and improving overall system efficiency [3]. In a global context where the photovoltaic capacity has increased 41% annually since 2009 [4], improving thermal management in solar inverters is critical to achieving energy sustainability goals. Accurate temperature modeling in solar inverters facilitates predicting and controlling thermal conditions, which is crucial for optimizing their performance and preventing failures [5].

Currently, the use of advanced thermal models combined with sensors and control algorithms is becoming increasingly important to ensure efficient thermal management in these devices [6]. Proper temperature control of a solar inverter is essential to maintaining the efficiency and longevity of these systems [7]. Inaccurate temperature modeling of solar inverters can significantly impact their performance and reliability, affecting these devices' thermal management and operational efficiency [8]. Additionally, erroneous temperature predictions can lead to suboptimal thermal management strategies, resulting in energy losses and reduced efficiency of solar inverters [9].

Previous studies have used various statistical and machine learning methods to model physical processes, energy production processes, and related parameters with varying degrees of success [10]–[13]. In [10], multiple machine-learning approaches have been applied, including multiple linear regression, decision trees, random forests, support vector machines, and neural networks. Otherwise, [11] presents a mathematical model of multigenic genetic programming (GP) designed to forecast the flow of the Blackwater River. To enhance its accuracy, this model has been optimized using the particle swarm optimization (PSO) algorithm. The model achieved an R^2 coefficient value of 0.96059 in the training set and 0.94296 in validation; however, it required 150 iterations in the adjustment parameters using PSO, which may result in increased computational expense. In addition, [12] proposes a method for predicting photovoltaic inverter temperatures using a hybrid neural network model that integrates convolutional neural networks (CNN) with long short-term memory (LSTM) networks. This CNN-LSTM approach significantly enhances temperature prediction accuracy, as demonstrated by an improvement in R^2 metrics from 0.92 to 0.96 compared to actual values. Finally, models based on neural networks (CNN-LSTM) and lumped thermal networks [13], have shown improvements in accuracy but face limitations in complexity and computational [14]. These limitations underscore the need for more efficient and robust methods for predictive solar inverter temperature modeling, aiming to enhance operational efficiency and extend the lifespan of PV systems. This study explores whether a PSO-based symbolic regression model can offer a more accurate and computationally efficient alternative to existing models.

Symbolic regression (SR) is a powerful tool for modeling complex systems, offering improvements in precision [15] and robustness [16]. Traditional genetic programming (GP), a widely used SR method, leverages evolutionary principles to evolve mathematical expressions over multiple generations [17]. However, GP and similar traditional methods often face challenges related to efficiency and convergence. In contrast, algorithms based on swarm intelligence, such as particle swarm optimization (PSO) and firefly algorithms, have emerged as promising alternatives for symbolic regression tasks [18]. Inspired by the collective behavior of social organisms, these algorithms navigate large and complex search spaces, typical of symbolic regression problems [19]–[21]. Despite their potential, limited studies have explored the application of PSO-based symbolic regression for modeling solar inverter temperatures or compared its effectiveness with traditional approaches like Multiple Linear Regression (MLR) or Genetic Algorithm (GA)-based symbolic regression. Analyzing this approach could identify more precise, interpretable, and robust temperature prediction methods, thereby enhancing thermal management in solar inverters.

On the other hand, various methods have been developed to estimate temperature based on power loss calculations, input and output parameters, and thermal models [22], [23]. These methods utilize the relationship between electrical parameters and thermal behavior to provide real-time temperature estimations [24].

From the above discussion, the following research question arises: ¿what extent can a symbolic regression algorithm based on Particle Swarm Optimization (PSO) enhance the accuracy of temperature modeling in solar inverters compared to MLR and GA-based symbolic regression models? This study specifically aims to compare the predictive accuracy of PSO-based symbolic regression with MLR and GA-based models for solar inverter temperature prediction, using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) as evaluation metrics.

The novelty of the work is summarized as follows: (1) the creation of an innovative symbolic regression model using particle swarm optimization (PSO), which achieves higher accuracy than traditional models like multiple linear regression (MLR) and genetic algorithm (GA)-based approaches; (2) the model's enhanced capability to capture nonlinear relationships between critical variables, such as active power and DC voltage, essential for complex systems like solar inverters; and (3) its computational efficiency, demonstrated by fewer iterations and shorter execution times compared to advanced methods.

The paper outline is as follows: Section II presents the data collection methodology, and the construction of the PSO-based symbolic regression algorithm is described. Section III presents the temperature modeling of the solar inverter and the evaluation of the PSO-based symbolic regression model compared to the MLR and GA-based symbolic regression models. Section IV contains the discussion. Finally, conclusions are drawn in Section IV.

II. Methods and Materials

This section describes the methodology used for the analysis. The study requires data on the temperature of the solar inverter, its active power, and DC bus voltage, as well as information from the photovoltaic system. This data serves as input for developing the symbolic regression model based on PSO, multiple linear regression model, and GA-based symbolic regression model. The results are then compared with the measured temperature of the solar inverter. Fig. 1 illustrates the methodology for developing and evaluating these models.

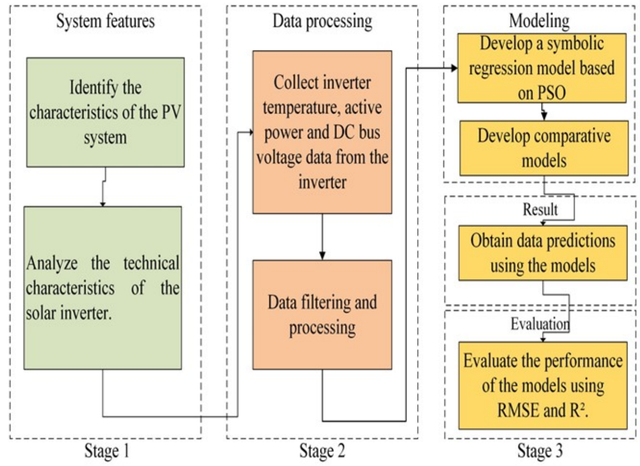

Fig. 1. Methodology followed in carrying out the models and the evaluations.

A. Characteristics of the PV System

The photovoltaic system is located on the rooftop of a building in Montería, Colombia, at coordinates 8°48'13.5" N, 75°51'0.45" W. It connects to the building's electrical grid through an indoor substation on the roof. The inverter is housed inside the rooftop structure, and Fig. 2 shows the current installation setup.

Fig. 2. Three-phase solar inverters and grid-connected electrical system.

B. Technical Characteristics of the Solar Inverter

The photovoltaic system comprises two solar inverters. The installation incorporates a transformer at the inverter output to facilitate grid connection. The solar inverter under analysis is the Yaskawa PVI 36TL-480. Table I describes the technical characteristics of the solar inverter and the transformer.

TABLE I. TECHNICAL CHARACTERISTICS OF THE INSTALLATION

Solar inverter and transformer	
Item	**Detail**
Inverter Rated Power	36 kW
Inverter Power Input Voltage Range	540-800 VDC
Inverter Ambient Temperature Range	(-25 °C to +60 °C)
Transformer for coupling to the electrical network	80 kW 460 V/120 V

C. Data Acquisition

The data extracted from the solar inverter includes the inverter temperature in (ºC), active power in (kW), and DC bus voltage in (V). The data are stored hourly on a cloud-based platform. The volume of data encompasses one year of storage. The transmission system uses a GSM module to transmit the data to a cloud-based computer system from which it can be downloaded for analysis. Fig. 3 illustrates the process of data transmission and storage.

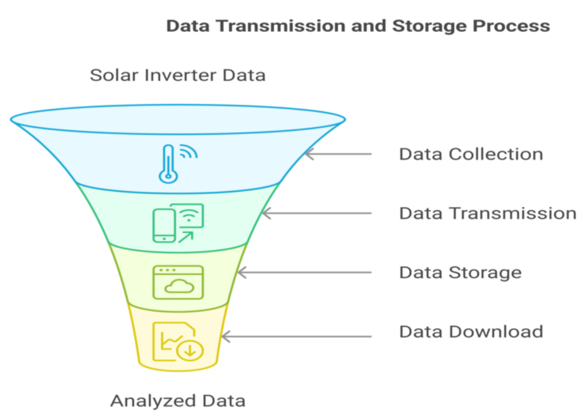

Fig. 3. Phases of the solar inverter's data transfer and storage process.

D. Data Filtering and Processing

The filtration and processing of the data followed these steps:

1) Identifying and eliminating outliers and inaccurate or incomplete data: Atypical data points substantially deviating from normal experimental parameters were classified as outliers. Specifically, any measurement that fell beyond two standard deviations from the adjusted median was designated as an outlier.

2) Manual review: Each data point was meticulously reviewed to identify and correct apparent errors or inconsistencies.

E. Algorithm Description

The PSO-based symbolic regression algorithm models nonlinear relationships by evolving algebraic expressions through particle swarm optimization. It integrates the following key steps:

1) Variable and function definition: Internal symbolic variables x1 and x2 are defined using Python's SymPy library. A list of mathematical functions (e.g. trigonometric, logarithmic) with random constants is constructed.

2) Expression generation: Random symbolic expressions are created by combining 3–6 terms from the predefined function list, operating on x1 and x2.

3) Fitness evaluation: Each expression is evaluated using RMSE between predicted and actual values. Invalid outputs (e.g., division by zero) are penalized with high error values.

4) PSO Update: Particle positions and velocities are updated based on personal best (pbest) and global best (gbest) solutions, guided by inertia (w=0.7), cognitive (c1=1.5), and social (c2=1.5) factors.

5) Optimization loop: The algorithm iteratively refines expressions over 25 iterations, balancing exploration and exploitation to minimize RMSE.

6) Data handling: Training (70%) and testing (30%) datasets are split using pandas. Results are visualized using matplotlib to track RMSE evolution.

F. Developing a Symbolic Regression Model Based on PSO

Symbolic regression is a tool that models complex nonlinear relationships between variables by identifying algebraic expressions that best fit the dynamics of a given system [25]. The PSO-based symbolic regression algorithm consists of the following subroutines in Fig. 4.

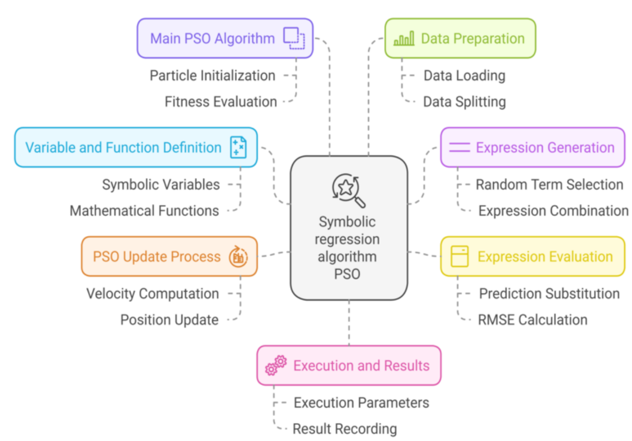

Fig. 4. Flowchart of symbolic regression algorithm based on particle swarm optimization (PSO).

The following section provides a detailed description of each subroutine the algorithm utilizes for its operation. The program used to develop the algorithm in Spyder 6 is an integrated development gateway (IDE) designed explicitly for scientific programming in Python [26].

Subroutines: Definition of Variables and Mathematical Functions
Input: No external inputs are required; the symbolic variables x1 and x2 are generated internally.
Output: The symbolic variables (x1, x2) are defined as mathematical symbols. The list of mathematical functions encompasses algebraic, trigonometric, and advanced operations (e.g. sin, log, sqrt), some with random constants.
Process: The symbolic variables x1 and x2 are defined utilizing the Sympy library in Python. A list of functions operating on x1 and x2 is constructed, incorporating random constants and predefined mathematical operations.

Subroutines: Generation of Mathematical Expressions
Input: Symbolic variables x1, x2. A predefined list of mathematical functions contains algebraic operations and mathematical functions applicable to x1 and x2.
Output: Symbolic mathematical expression combining several functions randomly selected from the list.
Process: A random number of terms is chosen (between 3 and 6). Functions are randomly selected from the list of mathematical functions. Each function operates on the symbolic variables x1 and x2. The resulting terms are summed to form a single symbolic expression.

Subroutines: Evaluation of Expressions
Input: A mathematical expression generated in a previous step. Arrays for x1, x2 (independent variables), and y (dependent variable).
Output: A numerical value representing the difference between the predicted values (from the symbolic expression) and the actual y values (RMSE).
Process: Each pair of values x1 and x2 are substituted into the symbolic expression to calculate the predicted value. If an invalid value (e.g. division by zero or NaN) occurs, assign a high error (inf) to the expression. Compares the predicted values with the actual values of y using the formula RMSE.

Subroutines: Update of Velocity and Position in PSO
Input: Current velocity and position representing the state of a particle. Personal best (pbest) and global best (gbest). PSO parameters (w, c1, c2).
Output: Updated velocity and position for the particle.
Process: Calculate the new velocity. Update the position.

Subroutines: Main PSO Algorithm
Input: Training and test data (x1, x2, y). PSO parameters: Number of particles, iterations, and constants (w, c1, c2).
Output: Best expression: The symbolic expression with the lowest RMSE. Best RMSE: The RMSE of the best expression. RMSE evolution: A record of RMSE values over iterations.
Process: Initialization: Generate initial particles (expressions) and velocities and evaluate their fitness. Optimization Loop: For each iteration: - Evaluate RMSE for each particle. - Update pbest and gbest based on fitness. - Adjust velocity and position for each particle. - Record the best RMSE for the iteration. Stop when the maximum iterations or target RMSE is reached.

Subroutines: Data Loading and Preparation
Input: Path to an Excel file containing data.
Output: Training and test datasets split into x_1, x_2, and y.
Process: Load data using `pandas.read_excel`. Extract columns for x_1, x_2, and y. Split data into 70% training and 30% test sets.

Subroutines: Results Visualization
Input: RMSE evolution data for each execution.
Output: A graph showing RMSE over iterations for all executions.
Process: Convert iterations to a time scale for the X-axis. Plot RMSE evolution for each execution using `matplotlib`. Label axes, add a title, and display the graph.

The equipment used for model development had the following technical specifications, as can be seen in Table II.

TABLE II. TECHNICAL CHARACTERISTICS OF THE COMPUTER EQUIPMENT

Laptop Computer and Software	
Item	**Detail**
CPU	8x1.9 GHz
RAM	32GB DDR4
RAM Speed	4267 MHz
Software	Spyder 6

G. Development of the MRL and GA-Based Symbolic Regression Model

Multiple linear regression (MLR) is a statistical technique investigating the relationship between a single dependent variable and several independent variables [27]. It aims to determine the equation of a line that minimizes the total squared deviations between the predicted and actual values of the dependent variable [27].

On the other hand, symbolic regression based on genetic algorithms (GA) is a computational technique that combines genetic algorithms with symbolic regression to discover the mathematical expressions that best fit a given dataset [17].This approach leverages the evolutionary principles of genetic algorithms to explore the space of potential mathematical models to identify the most accurate and interpretable representation of underlying data relationships [28].

The performance of the PSO-based symbolic regression model was compared with that of a multiple linear regression (MLR) model and a GA-based symbolic regression model using the same training data set at 70% and testing at the remaining 30%. The PSO-based and GA-based symbolic regression models employed an identical set of mathematical functions in their equation formulation processes.

The generated models were evaluated using 30% of the data set aside for testing. The main goal is to balance having enough training data to create robust models and sufficient test data to assess model performance [29].

H. Evaluation of Model Performance

Model performance was evaluated using RMSE and MAE metrics.

RMSE: This indicator evaluates the size of the discrepancies between the model's predicted values ($V_{\text{predicted}}$) and the actual values (V_{target}), as illustrated in Eq. (1) [30]. A lower RMSE indicates higher model accuracy.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(V_{\text{predicted},i} - V_{\text{target},i})^2} \tag{1}$$

MAE: Mean Absolute Error (MAE) is a widely used method in model validation in various fields of study. MAE quantifies the average error size in a set of predictions without considering their directionality [31]. Eq. (2) describes the Mean Absolute Error (MAE) as follows:

$$MAE = \frac{1}{n}\sum_{k=1}^{n}|y_k - \hat{y}_k| \quad (2)$$

The number of observations is denoted by n, y_k represents the actual value, and $\hat{y}_k$ denotes the predicted value.

III. Results

This section presents the primary results of the study. The results are organized into two subsections: The first subsection describes the process of building the multiple linear regression model, the PSO-based symbolic regression model, and the GA-based symbolic regression model. The second subsection analyzes and contrasts the results of comparing these models with the measured data of the solar inverter temperature.

A. Model Construction

The distribution of training and test data is examined before modeling. Fig. 5 and Fig. 6 illustrate the data distribution.

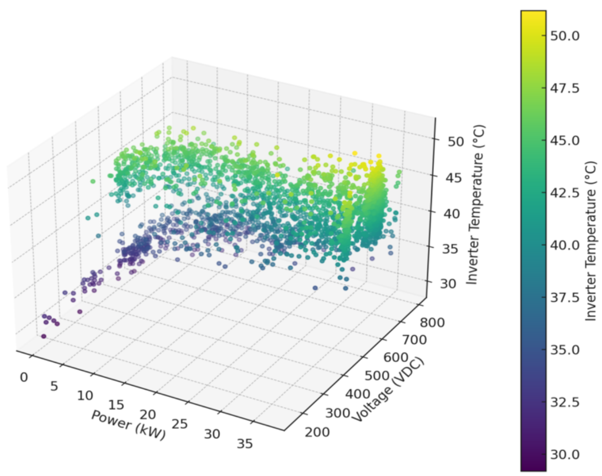

Fig. 5. Distribution of training data.

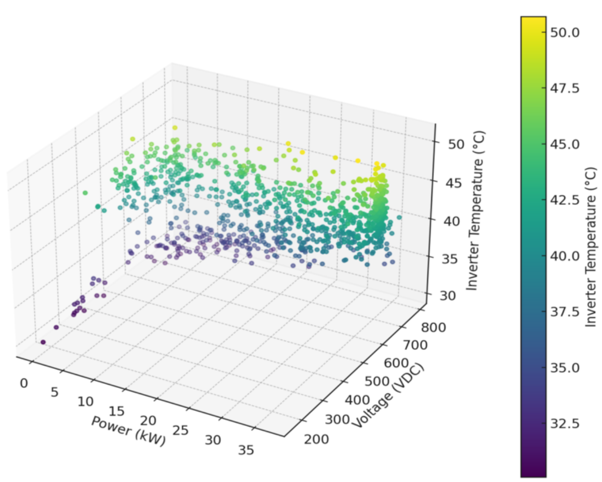

Fig. 6. Distribution of test data.

The dispersion and clustering of diverse data points across distinct regions of the three-dimensional space indicate complex relationships among the variables of active power, DC voltage, and inverter temperature. The following section describes the process of constructing the models:
A multiple linear regression model was established with training data and evaluated with the test dataset. The analysis revealed the corresponding coefficients in Equation (3) as follows:

$$MLR = 38.9104 + W \cdot 0.15981 + VD \cdot 0.0011, \quad (3)$$

where W represents the active power of the solar inverter in kW, and VD denotes the DC bus voltage in volts DC.

Using the training data, a symbolic regression model based on GA was established and evaluated with the test dataset. The parameters utilized for this model are illustrated in Table III.

TABLE III. Performance Parameters of the Algorithm based on Symbolic Regression based on GA

Input parameters	
Item	**Detail**
Population size	30
Number of generations	25
Initial mutation probability	0.5
Selection percentage	0.5
Crossover method	Subexpression Crossover
Number of algorithm executions	10

The optimal equation resulting from 10 executions of the algorithm was Eq. (4).

$$\text{Model}_{SR\_GA} = |VD|^{0.38} + \sqrt{VD} + 4.75, \quad (4)$$

where VD denotes the DC bus voltage in volts DC. Fig. 7 shows the result of the best RMSE value for each execution and the corresponding time.

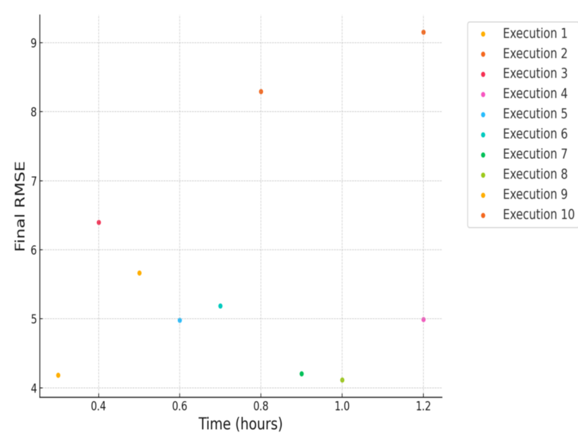

Fig. 7. Evolution of the RMSE of the Model RS GA over time.

Furthermore, selecting PSO parameters considers computational efficiency and model robustness; consequently, 30 particles and 10 executions are chosen for the algorithm. Concurrently, the inertia coefficient (0.7) and cognitive and social factors (1.5) are established to achieve an optimal balance between exploration and exploitation. As shown in Table IV, the selected inertia value allows particles to maintain sufficient momentum to escape local minima without affecting convergence.

TABLE IV. Operating Parameters of the Algorithm

Item	Detail
Number of iterations	25
Particle numbers	30
Inertia factor	0.7
Cognitive factor	1.5
Social factor	1.5
Number of algorithm executions	10

Fig. 8 illustrates the evolution of the RMSE over the iterations, and Fig. 9 shows the RMSE graphed over time.

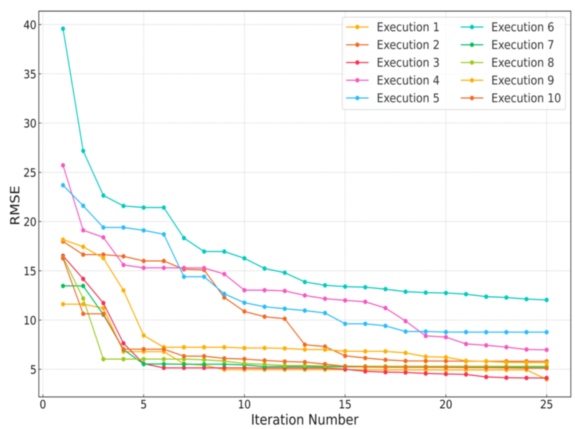

Fig. 8. Development of RMSE as a function of the number of iterations for different algorithm executions.

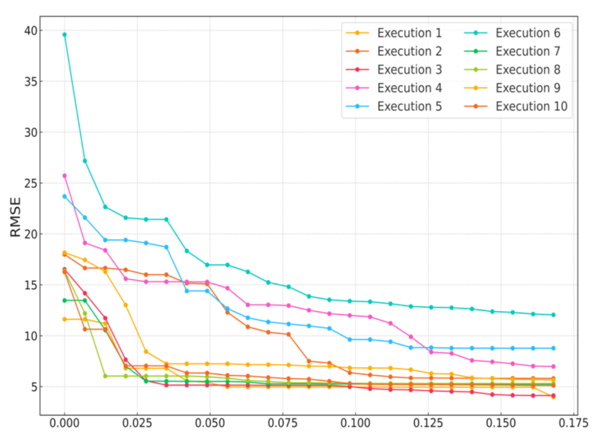

Fig. 9. Development of RMSE as a function of time for different algorithm executions.

The algorithm reduces the RMSE from initially high values (average values close to 20 in the first iterations) to final values between 3.97 and 12.04. This demonstrates the PSO's ability to identify symbolic equations that capture nonlinear relationships in the data. The progressive decrease in RMSE over the iterations confirms the algorithm's effectiveness in exploring and exploiting the search space, as confirmed in previous literature [11].

Regarding computing time, the algorithm completed 25 iterations within 0.175 hours per execution. The RMSE reduction reached a value of 3.97, positioning it as a competitive method compared to techniques such as neural networks or traditional multivariate regression for problems with nonlinear relationships [11]. The 25-iteration limit in the symbolic regression algorithm based on PSO allows for a brief and controlled execution time (0.175 hours), balancing computational efficiency and precision in searching solutions.

Additionally, using 30 particles balances exploration capacity and computational cost. This relatively modest size appears sufficient to adequately explore the search space and identify effective solutions to symbolic regression problems. Furthermore, a low inertia factor (0.5) favors local exploitation in advanced phases of the algorithm. This contributes to the rapid convergence observed in the RMSE.

A social factor of (1) means that the particles have a reduced tendency to follow the global best solution of the swarm, promoting a certain level of diversity in search. Combined with the high cognitive factor, this balance allows for more precise local convergence. The value of the cognitive factor (3) indicates that particles prioritize their own experiences. Eq. 5 illustrates the iterative process:

$$\begin{aligned} \text{Model RS PSO} = {} & 0.0940W - 0.0483VD \\ & + 0.2599\log(|VD| + 0.00001) \\ & - 0.4005\sin(W) + 0.4099\cos(VD) \\ & + 0.0049\tan(W) + 2.69206\sqrt{|VD|} \\ & + 0.52838\arctan(W) + 1.0941, \end{aligned} \tag{5}$$

where W represents the active power of the solar inverter in kW and VD denotes the DC bus voltage. The combination of nonlinear terms indicates that PSO can uncover relationships that traditional approaches, such as multiple linear regression, cannot capture.

Table V shows the RMSE and MAE results for the training dataset of multiple linear regression, symbolic regression model based on PSO, and GA-based symbolic regression model.

TABLE V. Comparison of Results with Training Data

Models and Metrics		
Item	**RMSE**	**MAE**
Model Multiple Linear Regression (MLR)	4.22	3.55
Symbolic Regression Model based on PSO (SR PSO)	3.97	3.36
Symbolic Regression Model based on GA (SR GA)	4.59	3.78

Furthermore, Table VI presents the corresponding results for the test dataset.

TABLE VI. Comparison of Results with Test Data

Models and Metrics		
Item	**RMSE**	**MAE**
Model Multiple Linear Regression (MLR)	4.52	3.73
Symbolic Regression Model based on PSO (SR PSO)	4.12	3.31
Symbolic Regression Model based on GA (SR GA)	4.80	3.66

The RMSE and MAE metrics indicate that the symbolic regression model based on PSO demonstrated superior training and test data performance, achieving RMSE values of 3.97 and MAE of 3.31. Fig. 10, Fig. 11, and Fig. 12 illustrate the behavior of the Model MRL, Model SR PSO, and Model SR GA with training data.

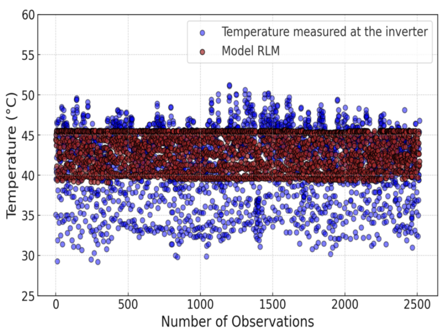

Fig. 10. Comparison of the temperature measured and the Model MLR with training data.

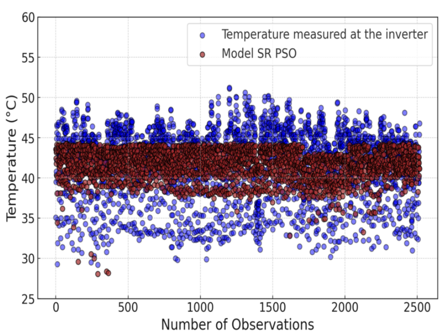

Fig. 11. Comparison of the temperature measured and the Model RS PSO with training data.

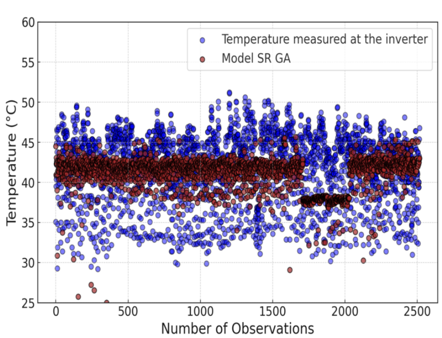

Fig. 12. Comparison of the temperature measured and the Model RS GA with training data.

Fig. 13, Fig. 14, and Fig. 15 show the behavior of Model MRL, Model SR PSO, and Model SR GA with training data. The model MLR represents a linear approximation with more substantial errors in scenarios where the relationships between variables and inverter temperature are not strictly linear, resulting in more significant deviations between predicted and actual values. The model SR PSO typically aligns more closely with actual inverter temperatures and exhibits reduced bias across multiple observations. Additionally, the Model SR GA model demonstrates inferior performance compared to the MLR and RS PSO models in evaluation metrics.

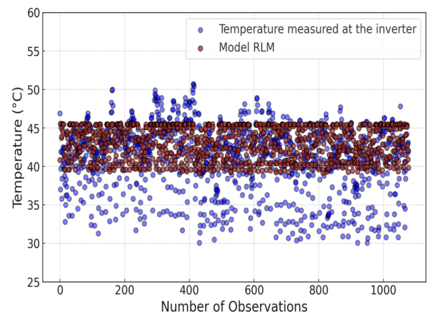

Fig. 13. Comparison of temperature measured and the MLR model with test data.

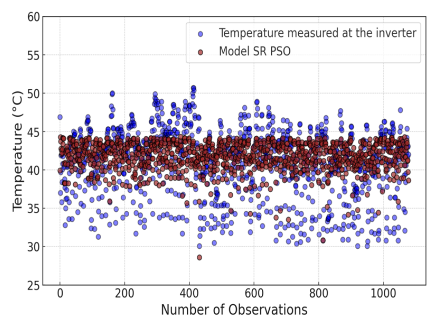

Fig. 14. Comparison of temperature measured and the Model RS PSO with test data.

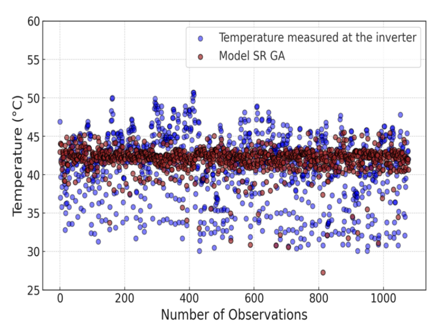

Fig. 15. Comparison of temperature measured and the Model RS PSO with test data.

Fig. 16, Fig. 17, and Fig. 18 show the performance of the MLR, RS PSO, and SR GA models based on the training data, while Fig. 19, Fig. 20, and Fig. 21 illustrate the behavior with test data.

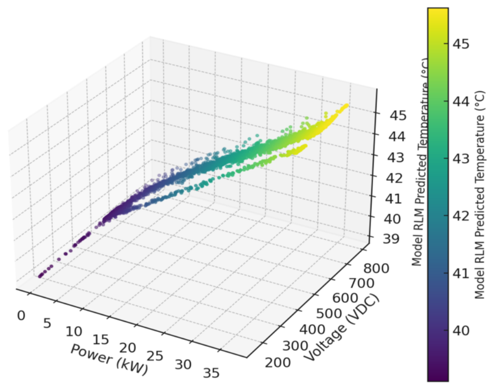

Fig. 16. Distribution of the data predicted by the model MLR using the training data.

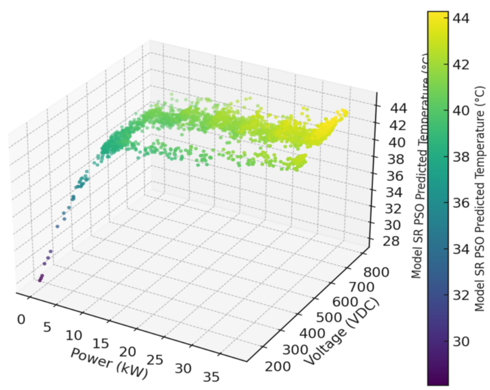

Fig. 17. Distribution of data predicted by the model RS PSO using the training data.

While multiple linear regression (MLR) presents a simplified linear trend with a limited prediction range (39-45 °C) that does not capture the full complexity of the actual data (30-50 °C), the symbolic regression model with PSO demonstrates superior capability in reproducing the non-linear nature of interactions among inverter power, voltage, and temperature, as evidenced by its improved fit to variable gradients and more realistic prediction range (28-44°C).

Both the symbolic regression model response based on PSO and the one based on GA exhibited similar RMSE values; however, the regression developed with PSO achieved the lowest value after 10 executions in both algorithms.

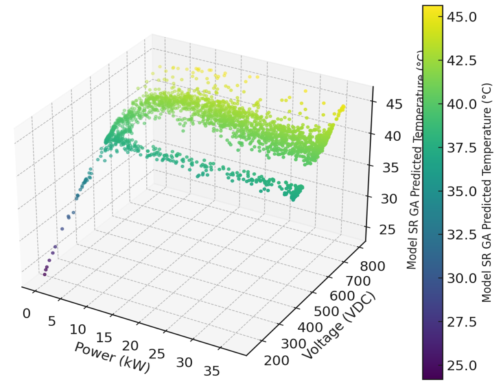

Fig. 18. Distribution of data predicted by the model RS GA using the training data.

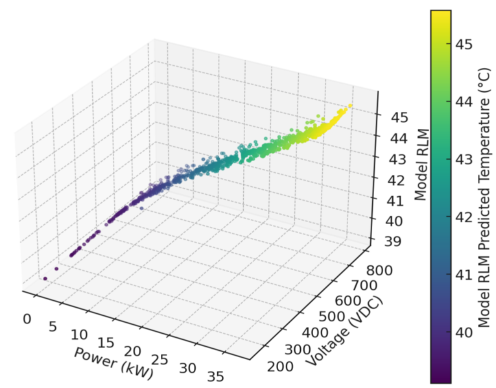

Fig. 19. Distribution of data predicted by the model MLR using test data.

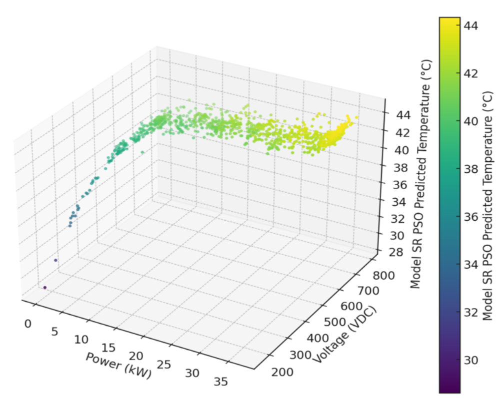

Fig. 20. Distribution of data predicted by the model RS PSO using the test data.

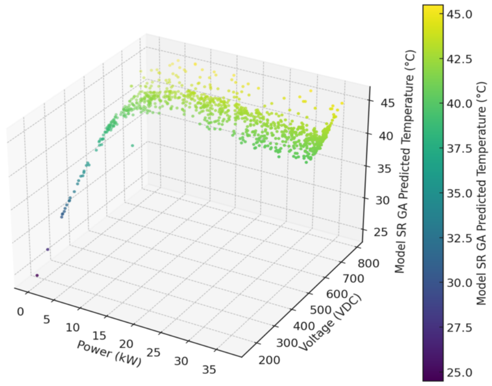

Fig. 21. Distribution of data predicted by the model RS GA using the test data.

IV. Discussion

The rapid convergence of the PSO-based model and its reduced computational cost compared to the GA-based approach render it an attractive option for symbolic regression tasks. Nevertheless, PSO may experience premature convergence and stagnation in complex search spaces, corroborating recent findings [32].

The response surface generated by PSO more accurately reflects the local variations and curvature observed in the experimental data. This suggests that this method is more appropriate for modeling complex physical systems where relationships between variables are inherently non-linear, as with temperature in solar inverters.

The results validate the efficacy of PSO in symbolic regression; however, its tendency to converge prematurely may result in suboptimal solutions if not managed appropriately.

V. Conclusion

The study shows that a symbolic regression model based on particle swarm optimization (PSO) outperforms multiple linear regression (MLR) and a Symbolic regression model based on GA in predicting the internal temperature of solar inverters, as evidenced by lower RMSE and MAE values. The SR PSO model's ability to capture nonlinear relationships between active power (W), DC bus voltage (VD), and temperature represents a significant advantage given the complex nature of solar inverter systems. However, the RS PSO model may require more computational resources.

According to the analyzed literature, this study investigated the application of the Particle Swarm Optimization (PSO) algorithm in symbolic regression, a domain predominantly utilizing Genetic Programming (GP) and neural networks. Although the surface modeling of the training and test data was not very close, the model highlighted the computational efficiency and potential adaptability of PSO to symbolic regression tasks.

Future research could examine the generalizability of the SR PSO model across different solar inverter types and environmental conditions, incorporate additional variables such as ambient temperature and humidity, and explore possible improvements through parameter optimization. In addition, comparing the SR PSO model with other advanced machine learning models and integrating it into real-time monitoring systems could further improve its practical application in photovoltaic systems.

Acknowledgment

This work is supported by Universidad Pontificia Bolivariana, Seccional Monteria. The authors also wish to express their gratitude for their collaboration in developing this research at Universitat Politècnica de València, Universidad de Guadalajara, and Universidad de Pamplona.

References

[1] M. Nasr, M. Ibrahim, and A. El Berry, "Solar Photovoltaic Panels Fault Detection Due to Thermal Effects," *Journal of International Society for Science and Engineering*, vol. 6, no. 2, pp. 26–35, 2024. [Online]. Available: www.isse.org.eg

[2] R. Madake and S. Dhanaraj, "Photovoltaic inverters experimentally validate power quality mitigation in electrical systems," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 36, no. 2, pp. 715–723, nov. 2024. doi: 10.11591/ijeecs.v36.i2.pp715-723.

[3] T. Liang et al., "Implementation and Applications of Grid-Forming Inverter with SiC for Power Grid Conditioning," *IEEJ Journal of Industry Applications*, 2023. doi: 10.1541/ieejjia.22006702.

[4] L. Kruitwagen, K. Story, J. Friedrich, L. Byers, S. Skillman, and C. Hepburn, "A global inventory of photovoltaic solar energy generating units," *Nature*, vol. 598, no. 7882, 2021. doi: 10.1038/s41586-021-03957-7.

[5] T. Ivanov and R. Stanev, "Mathematical model of photovoltaic inverters," in *11th Electrical Engineering Faculty Conference, BulEF 2019*, 2019. doi: 10.1109/BulEF48056.2019.9030705.

[6] Z. Wang, "A Thermal Management Strategy for Inverter System Based on Predictive Control," *International Journal of Heat and Technology*, vol. 41, no. 1, 2023. doi: 10.18280/ijht.410115.

[7] D. Mathew and R. Naidu, "A review on single-phase boost inverter technology for low power grid integrated solar PV applications," *Ain Shams Engineering Journal*, vol. 15, no. 2, 2024. doi: 10.1016/j.asej.2023.102365.

[8] A. Ribeiro et al., "Junction Temperature Prediction and Lifetime Assessment in a PV Inverter Using a 10-year Mission profile," *Eletrônica de Potência*, vol. 29, pp. e202438, oct. 2024. doi: 10.18618/REP.e202438.

[9] A. Luiz Perin, A. Krenzinger, and C. Massen Prieb, "Modelagem térmica de um inversor fotovoltaico conectado à rede," in *Anais Congresso Brasileiro de Energia Solar - CBENS*, 2016. doi: 10.59627/cbens.2016.1978.

[10] Z. Zheng, "Introduction to Solar Technology and its Environmental Importance," *Applied and Computational Engineering*, vol. 98, no. 1, pp. 47–56, nov. 2024. doi: 10.54254/2755-2721/98/20241116.

[11] A. Sheta, A. Abdel-Raouf, K. Fraihat, and A. Baareh, "Evolutionary Design of a PSO-Tuned Multigene Symbolic Regression Genetic Programming Model for River Flow Forecasting," 2023. [Online]. Available: www.ijacsa.thesai.org

[12] H. Dong and H. Liu, "Research on Photovoltaic Inverter Temperature Prediction Method Based on CNN-LSTM," in *9th International Conference on Intelligent Computing and Signal Processing, ICSP 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1647–1651. doi: 10.1109/ICSP62122.2024.10743359.

[13] A. Sangwongwanich, H. Wang, and F. Blaabjerg, "Reduced-Order Thermal Modeling for Photovoltaic Inverters Considering Mission Profile Dynamics," *IEEE Open Journal of Power Electronics*, vol. 1, 2020. doi: 10.1109/OJPEL.2020.3025632.

[14] M. Karim et al., "Explainable AI for Bioinformatics: Methods, Tools and Applications," *Briefings in Bioinformatics*, vol. 24, no. 5, 2023. doi: 10.1093/bib/bbad236.

[15] B. Cohen, B. Beykal, and G. Bollas, "Dynamic System Identification from Scarce and Noisy Data Using Symbolic Regression," in *Proceedings of the IEEE Conference on Decision and Control*, 2023. doi: 10.1109/CDC49753.2023.10383906.

[16] Z. Bastiani, R. Kirby, J. Hochhalter, and S. Zhe, "Complexity-Aware Deep Symbolic Regression with Robust Risk-Seeking Policy Gradients," 2024. [Online]. Available: http://arxiv.org/abs/2406.06751.

[17] N. Jiang and Y. Xue, "Racing Control Variable Genetic Programming for Symbolic Regression," 2023. [Online]. Available: http://arxiv.org/abs/2309.07934.

[18] M. Aliwi, S. Aslan, and S. Demirci, "Firefly Programming for Symbolic Regression Problems," in *28th Signal Processing and Communications Applications Conference*, 2020. doi: 10.1109/SIU49456.2020.9302201.

[19] V. Khaidurov, "Methods and Algorithms of Swarm Intelligence for the Problems of Nonlinear Regression Analysis and Optimization of Complex Processes, Objects, and Systems: Review and Modification of Methods and Algorithms," *System Research in Energy*, 2024. doi: 10.15407/srenergy2024.03.046.

[20] G. Kronberger, F. de Franca, H. Desmond, D. Bartlett, and L. Kammerer, "The Inefficiency of Genetic Programming for Symbolic Regression – Extended Version," *GECCO '18: Proceedings of the Genetic and Evolutionary Computation Conference*, 2024. [Online]. Available: http://arxiv.org/abs/2404.17292.

[21] P. Orzechowski, W. La Cava, and J. Moore, "Where are we now? A large benchmark study of recent symbolic regression methods," in *Proceedings of the 2018 Genetic and Evolutionary Computation Conference*, 2018. doi: 10.1145/3205455.3205539.

[22] M. Taha, M. Tousizadeh, A. Deb, O. Alatise, L. Ran, and P. Mawby, "Real Time Estimation of Power Transistor Junction Temperature for Motor Drive Application," in *IET Conference Proceedings*, 2022. doi: 10.1049/icp.2022.1153.

[23] G. Xu, H. Wu, Z. Pan, W. Wei, and Y. Zhang, "A Novel Online Method for Monitoring the Junction Temperature Based on the Input and Output of the Inverter," in *8th International Conference on Electrical and Electronics Engineering, ICEEE 2021*, 2021. doi: 10.1109/ICEEE52452.2021.9415952.

[24] Q. Niu and L. Wang, "Real-time Inverter Semiconductor Die Temperature Estimation Using Inverter Operating Condition-based Gate Recurrent Unit," in *9th IEEE International Symposium on Power Electronics for Distributed Generation Systems, PEDG 2018*, 2018. doi: 10.1109/PEDG.2018.8447642.

[25] D. Angelis, F. Sofos, and T. Karakasidis, "Artificial Intelligence in Physical Sciences: Symbolic Regression Trends and Perspectives," 2023. doi: 10.1007/s11831-023-09922-z.

[26] A. Ziogas et al., "Productivity, portability, performance: data-centric Python," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, in SC '21. New York, NY, USA: Association for Computing Machinery, 2021. doi: 10.1145/3458817.3476176.

[27] J. Jiang, "Multiple Linear Regression," in *Applied Medical Statistics*, John Wiley & Sons, Ltd, 2022, ch. 15, pp. 345–367. doi: https://doi.org/10.1002/9781119716822.ch15.

[28] Y. Radwan, G. Kronberger, and S. Winkler, "A Comparison of Recent Algorithms for Symbolic Regression to Genetic Programming," 2024. [Online]. Available: http://arxiv.org/abs/2406.03585.

[29] B. Supriet al., "Asian Stock Index Price Prediction Analysis Using Comparison of Split Data Training and Data Testing," *Jurnal Ekonomi, Manajemen, dan Akuntansi*, vol. 9, no. 4, 2023. doi: 10.35870/jemsi.v9i4.1339.

[30] T. Hodson, "Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not," 2022. doi: 10.5194/gmd-15-5481-2022.

[31] Z. Dayev, A. Kairakbaev, K. Yetilmezsoy, M. Bahramian, P. Sihag, and E. Kıyan, "Approximation of the discharge coefficient of differential pressure flowmeters using different soft computing strategies," *Flow Measurement and Instrumentation*, vol. 79, 2021. doi: 10.1016/j.flowmeasinst.2021.101913.

[32] M. Montes-Rivera, C. Guerrero-Mendez, D. Lopez-Betancur, and T. Saucedo-Anaya, "Dynamical Sphere Regrouping Particle Swarm Optimization Programming: An Automatic Programming Algorithm Avoiding Premature Convergence," *Mathematics*, vol. 12, no. 19, pp. 3021, 2024. doi: 10.3390/math12193021.