ELECTRICAL ENGINEERING AND ENERGY

OPEN ACCESS

An Improved Electromagnetism-like Algorithm for Gaussian Process Hyperparameter Optimization in Remaining Useful Life Prediction of Electrolytic Capacitors

Kofi Addo Annan[1], Francis Effah Boafo[1], Emmanuel Asuming Frimpong[1], Elvis Twumasi[1], Abdul-Majid Issah Malori[1], and Mark Ampomah Otu[2]

[1] Department of Electrical and Electronic Engineering, Kwame Nkrumah University of Science and Technology. Kumasi, Ghana
[2] Department of Electrical Systems & Automatic Control, Université de Lille. Lille, France

Submitted: 9 May 2026
Accepted: 11 June 2026
Online First: 17 June 2026

Corresponding author
Kofi Addo Annan,
annanpap3@gmail.com

DOI:10.64470/elene.2026.35

Abstract: This paper introduces an innovative optimization procedure to enhance gaussian process regression (GPR) for hyperparameter tuning. The technique estimates the remaining useful life (RUL) of electrolytic capacitors under electrical overstress conditions. The traditional electromagnetism-like (EM) optimization technique suffers premature convergence and insufficient exploitation near local optimal solutions. It also suffers ineffective charge relocation within high-dimensional search spaces. To address these limitations, the adaptive chaotic opposition-based learning electromagnetism-like (ACOEM) algorithm is proposed. The new algorithm incorporates three techniques to overcome these challenges: opposition-based learning to increase initial population size and speed up the optimization process; chaotic map-based scaling of forces to strike a balance between exploration and exploitation; and lévy flights to avoid local minima. The ACOEM optimizer fine-tunes the hyperparameters of the GPR kernel by minimizing the negative log marginal likelihood (NLML) objective function. The proposed ACOEM-GPR predictor is successfully applied to National Aeronautics and Space Administration (NASA's) electrochemical capacitor electrical overstress dataset based on three (3) stress levels: 10V, 12V and 14V. Compared to the vanilla GPR predictor, ACOEM-GPR exhibited superior performance in terms of the correlation coefficient R^2, improving from 0.3015 to 0.7482 for 10V stress and from 0.5615 to 0.8921 for 14V stress. Additionally, the ACOEM-GPR method showed significant improvements in mean squared error (MSE) reducing the MSE from 9458 to 3410 μF^2 and root mean squared error (RMSE) from 97.3 to 58.4 μF at 10V. ACOEM-GPR is proven to produce accurate and robust RUL predictions allowing for effective predictive maintenance of power converter capacitors.

Keywords: Gaussian Process Regression (GPR), Electromagnetism-like Algorithm, Electrolytic Capacitors, Negative Log Marginal Likelihood, Remaining Useful Life.

1. Introduction

Capacitors made from electrolyte are an important part of power electronic convertors to help with; smooth voltage, store energy and filtering functions (Mannen & Fujita, 2016), however they have one of the highest rates of failure and provide about 30% of all failings within power electronics according to survey results done by certain associations (H. Wang et al., 2012). Aging caused by electrical overstress causes progressive degradation of these components by resulting in an overall reduction of capacitance in addition to an increase in equivalent series resistance (ESR). Hence, it is critical to predict the Remaining Useful Life (RUL) of these components as part of a predictive maintenance program to minimize unscheduled downtime and improve the reliability of the overall system (Rahimpour et al., 2022).

In data-driven forecasting algorithms, the Gaussian Process Regression (GPR) technique has shown remarkable efficiency in predicting the Remaining Useful Life (RUL) of electrolytic capacitors. This is because the algorithm is capable of capturing non-linear degradation trends and giving uncertainties (Jha & Dong, 2024). However, the forecasting performance of GPR greatly relies on the correct tuning of its kernel hyperparameters including the signal variance (σ_f^2), length scale (l), and noise variance (σ_n^2) which are a function of the Negative Log Marginal Likelihood (NLML) (Chen & Wang, 2017). Traditional hyperparameter tuning approaches like gradient-based optimizers like function minimization with constraints (fmincon) in MATLAB have been observed to yield sub-optimal results, especially when the problem has sparse and non-linear characteristics.

To overcome this issue, meta-heuristics have been used to create new options for solving problems. The electromagnetism-like (EM) algorithm, created by Birbil and Fang (Birbil & Fang, 2003), is based on the attraction and repulsion of particles that have electric charges to determine a route through the search space. The original EM algorithm has many limitations including: premature convergence; lack of sufficient benefit of search around local optima; and lack of guaranteed convergence (Lin et al., 2012). Despite many improvements made to the original EM algorithm using methods such as: split-probe-compare (Tan et al., 2016); adaptive step size (Ragb et al., 2023); and lévy flights and chaotic maps (Zhu et al., 2024), none of the EM algorithm improvements developed have addressed: the efficiency of initialization, force scaling dynamics, and lévy flights for perturbation of the particles.

The objective of this study is to fill the gaps identified in current research by proposing an Adaptive Chaotic Opposition-Based Learning Electromagnetism-Like (ACOEM) algorithm. The ACOEM algorithm is the new variation of a standard electromagnetism-like optimization algorithm for tuning GPR hyperparameters for estimating of RUL of an electrolytic capacitor. The ACOEM algorithm leverages three synergistic strategies to achieve its objectives: (i.) population diversity is improved and convergence accelerated through opposition-based learning initialization; (ii.) balancing global exploration and local exploitation by means of generating chaotic map driven force scaling based on the use of logistic maps; and (iii.) the best solution is perturbed via Lévy flights to avoid becoming trapped in a local optimum and improve intensifying solutions. The ACOEM algorithm is used to minimize the Negative Log Marginal Likelihood (NLML) which serves as an objective function for determining optimal GPR hyperparameters.

The suggested ACOEM-GPR model is evaluated using NASA's publicly published dataset for electrolytic capacitors with electrical overstress under three (3) aging voltages of 10 V, 12V and 14 V. It is found through comparative analysis that the ACOEM-GPR approach provides superior performance in comparison to vanilla GPR (trained with fmincon). This is evident through the reduction of MSE, RMSE, and MAE errors, as well as improved R^2 scores.

Over the years, there has been a proliferation of different methodologies used to estimate the Remaining Useful Life (RUL) of electrolytic capacitors including: particle filtering (PF) under thermal overstress conditions(Khorasgani et al., 2013) and variable conditions(Rigamonti et al., 2016); Kalman filtering (KF) within a model-driven context(Roman et al., 2021); relevance vector regression (RVR)(X. Wang et al., 2022); and long-short-term memory (LSTM) or bi-directional LSTM networks(Forouzandeh Shahraki et al., 2023; Kulevome et al., 2021). All of these methodologies show some utility value. However, the GPR approach used in this research has unique features that differentiate it from existing methods. For example, PF and KF require a predetermined empirical degradation model based on a hypothesized physical process. Conversely, GPR is a non-parametric Bayesian method that does not specify a fixed functional form for the degradation trend thus giving it greater flexibility in estimating nonlinear and multi-step degradation behavior, which are widely recognized as being difficult to estimate when trying to establish the RUL value of capacitors. Additionally, GPR provides predictive uncertainty estimates in the form of posterior variance which is not available with most LSTM and RVR implementations. This information is important for making maintenance decisions that are informed by risk. Recent studies have demonstrated how developed GPR systems can decrease mean square error for accelerated aging data sets. With these intrinsic qualities, that is, flexibility, robust performance when applied to datasets that have very few observations, and explicit quantification of uncertainty, positions GPR as uniquely capable of addressing the issue of capacitor prognostics, especially where degradation data are scarce and the underlying physical mechanisms are complex and/or only partially understood.

Recent studies by Jha & Dong (2024), proposed a predictive maintenance approach leveraging time series data from NASA's Prognostics Center of Excellence dataset repository on capacitors. The electrochemical impedance spectrum (EIS) data was key for pre-analysis to derive capacitance trends for 10V, 12V and 14V electrical overstress experiments based on the dataset. Two-thirds of the resampled data underwent training on the GPR model which combined the radial basis function (RBF) and WhiteKernel and validated its performance on the remaining one-third. The GPR achieved an R^2 value of 0.563 and MSE of 0.018 for predicting degradation of capacitors. Although the GPR algorithm provided a reasonable outcome in estimating the degradation of the capacitors, there is no discussion of how to optimize the GPR algorithm's parameters, such as the length-scale and noise parameter, that play an important role in modelling the nonlinear trends of degradation. This research gap can be filled by optimizing the GPR algorithm through a metaheuristic algorithm, such as the electromagnetism-like algorithm. In this way, the GPR will be able to effectively model the capacitance loss while possibly resulting in an increase in the R^2 score and decrease in the MSE value.

There have been a number of studies in the direction of improving the electromagnetism-like algorithm. Oliva et al. (2014) worked on the enhancement of the electromagnetism-like algorithm for application in template matching in image processing. They applied two key modifications to the to the traditional electromagnetism-like algorithm which are adding a memory mechanism to store previously calculated Normalized Cross-Correlation (NCC) values, preventing redundant calculations, and a selective local search (LS) that perturbs only promising particles, lowering NCC evaluations. They compared their proposed improved algorithm in template matching to other state of the art methods which some utilized Particle Swarm Optimization (PSO-TM), Chaotic Imperialist Competitive Algorithm (ICA-TM), Genetic Algorithm (GA-TM) and the original Electromagnetism-like Algorithm. Based on their results, their proposed improvement achieved superior precision. Whiles their research obtained a positive result, their study lacked a comprehensive study of the performance of their improved algorithm on standard benchmark functions in comparison with other metaheuristic algorithms.

Tan et al. (2016) worked along the lines of improving the numerical optimization of the electromagnetism-like algorithm where they introduced a novel local search strategy called Split, Probe, and Compare (SPC). Since the local search strategy of the original electromagnetism-like algorithm utilizes a random step size, which frequently leads to varying results and less-than-ideal solutions leading to imbalance in exploration and exploitation, they proposed a local search strategy based on the SPC theory. The Split divides the search into two phases or probes with the probe A towards the lower bound and probe B towards the upper bound per dimension. Probe lengths are adjusted using a non-linear Equation (1);

$$L = \frac{2}{1 + \exp(\frac{10i}{Max - LSite})} \tag{1}$$

where L is the probe length, i representing the current iteration and Max-LSite referring to the maximum local search iterations. The Equation (1) ensures their proposed improved algorithm starts with larger step size enabling wider exploration and shrinks as the number of iterations increases for finer exploitation. Lastly, the probing outcomes are evaluated shifting particles towards better solutions. They used 10 benchmark test functions to compare their proposed SPC-EM against two variants of EM, that is, the EM with smaller step size of 0.01(EMSSS) and EM with larger step size of 0.99(EMLSS). They also considered the Genetic algorithm (GA) in the comparison study. SPC-EM exhibited superior accuracy, which continuously achieved the lowest objective function values and smaller standard deviations, denoting dependability and precision. Though Tan et al. presented a more comprehensive analysis by benchmarking these algorithms, they failed to mention an important detail which is the number of iterations(i) used in their analysis. Also, the minimization capabilities of their proposed algorithm can further be improved by exploring other algorithmic components of the original EM-like algorithm.

Ragb et al. (2023) enhanced the electromagnetism-like algorithm for parameter identification in photovoltaic systems. To balance exploration and exploitation, they replaced a fixed delta value in the original EM algorithm with an adaptive local search size which decreases as the algorithm progresses in Equation (2).

$$L = \frac{2}{1 + \exp(\frac{10 \times LSITER}{MAXITER})} \tag{2}$$

where L is the proposed adaptive local search step size, LSITER is the number of iterations for the local search and MAXITER representing the maximum number of generations. Lower RMSE values in their results demonstrated their improved electromagnetism-like algorithm's quick convergence, surpassing more conventional techniques like genetic algorithms and particle swarm optimization. Their work lacked proper benchmarking with existing variants of EM-like algorithms to better validate their results.

Recent work by Zhu et al. (2024) involved the enhancement of the electromagnetism-like algorithm for optimization and validation analysis of hybrid PV-battery-diesel power systems. The key enhancements they made to the original algorithm was the integration of levy flight in the original EM's search space which introduces jumps to help avoid local optima problems. They also integrated chaotic maps to introduce unpredictable behavior in the search space to alleviate premature convergence and boost global exploration. Lastly, they modified the position update formula of the original EM algorithm by adding the levy flight and chaotic maps to the equation. They validated their improved algorithm on six multimodal benchmark functions against Biogeography-Based Optimization (BBO), Emperor Penguin Optimization (EPO) and Particle Swarm Optimization (PSO). Based on their case study, the results of their analysis showed that their improved EM algorithm optimally determines PV, diesel generator and battery sizes at lower cost of energy in comparison to the other algorithms demonstrating that their work efficiently enhances hybrid systems. Their work lacked proper benchmarking with existing variants of EM-like algorithms to better validate their results. The main contributions of this research include:

1. Proposing a novel ACOEM approach based on opposition learning, chaotic scaling, and Lévy flight techniques as solutions to the drawbacks of the conventional EM algorithm.
2. Application of the ACOEM approach to optimize the hyperparameters of GPR in relation to RUL prediction of electrolytic capacitors used in power electronics converters.
3. Benchmarking of the proposed method against the traditional GPR with quantified improvement in performance results.

2. Materials and Method

2.1 Electromagnetism-Like Algorithm

The electromagnetism-like optimization algorithm was originally developed by S. Ilker Birbil and Shu-Cherng Fang from the University of North Carolina State, Raleigh in 2003. EM-like algorithm is a heuristic for global optimization which depends on the principle of attraction-repulsion in moving sample points towards optimality (Birbil & Fang, 2003). The EM-like algorithm has four major algorithmic phases. They include initialization, calculating total force on each particle, moving in the force's direction, and using neighborhood search to find local minima. The EM-like algorithm deals with functions with bounded variables having the form in Equation (3) below;

$$\min f(x) \quad s.t. \qquad x \in \quad [l, u] \tag{3}$$

where [l, u] are all real numbers such that $l_k \leq x_k \leq u_k$ where k ranges from [k = 1, 2…n]. The function described above have the following parameters which will be used to describe the algorithm's characteristics in the following paragraphs.

n = number of dimensions of the function
u_k = the upper bound at k where [k = 1,2…n]
l_k = the lower bound at k where [k = 1,2…n]
f(x) = references the objective function

2.1.1 Initialization of Charges

At the initialization phase of the EM-like algorithm, there is uniform random distribution of *m* charges within the bounds of the specified objective function, l_k and u_k. Based on the objective function for the minimization problem, the uniformly distributed *m* charges are assigned values for each dimension. For a defined *m* number of charges, the charge with the best value with respect to the function's global minimum is assigned the global best at the instance (Ragb et al., 2023). Each charge's position in each dimension is initialized as shown in Equation (4);

$$x_{i,k} = l_k + \lambda\,(u_k - l_k\,), \qquad where\ \lambda\ is\ U(0,1) \tag{4}$$

for number of particles *i* and dimension *k*, l_k and u_k are the lower and upper bounds respectively.

2.1.2 Local Search

The local information for a point x_i is gathered using local search. The number of iterations and the multiplier for the neighborhood search are represented by the parameters LSITER and δ, respectively. The following is how the process iterates: First, using the parameter δ, the maximum practical step length is determined. Secondly, coordinate by coordinate, improvement in x_i is sought for a given *i*. The initial information is stored by assigning the point x_i to a temporary point *y* for a specified position as shown in Equation (5) below. The point *y* is then moved along the chosen step length, which is chosen at random. The neighborhood search for point *i* is terminated if the point *y* observes a better point within LSITER iterations, and y replaces the point x_i (Lin et al., 2012). The λ value is a random number uniformly distributed in [0,1].

$$y_k = \begin{cases} x_{i,k} + \lambda_1 \,.\, \delta(u_k - l_k) & if\ \lambda_1 > 0.5, \\ x_{i,k} - \lambda_2 \,.\, \delta(u_k - l_k) & otherwise, \end{cases} \tag{5}$$

where $\lambda_1, \lambda_2 \sim U(0,1)$, δ is the step-size parameter, $x_{i,k}$ is a charge point in the search space, and l_k and u_k are the lower and upper bounds respectively.

2.1.3 Total Force Calculation

The electromagnetic force between two points is directly proportional to the product of their charges and inversely proportional to the distance between them, by the superposition principle of electromagnetic theory. In each iteration, the values of the objective function are used to calculate the charges q_i in Equation (6) below. The calculated charge q_i represents a potential solution to the optimization problem. The charges are given according to their fitness values. Unlike electrical charges, no signs are attached to the charge of an individual point, rather the charges with better solutions are given higher charges. The exponential function below is used to compute the charge q_i (Birbil & Fang, 2003).

$$q_i = \exp\left(-n \, \frac{f(x^i) - f(x^{best})}{\sum_{k=1}^{m}\left(f(x^k) - f(x^{best})\right)}\right), \quad \forall\, i \tag{6}$$

where i is an index of a particle in a population, m is the population size, x^i is a candidate solution, f (x^i) is the objective function value at x^i, x^{best} is the best solution found, f (x^{best}) is the objective function value at x^{best}, and n representing the number of dimensions.

The direction of the force between two charges is determined after weighing the charges' objective function values. In the calculation of the force on a point charge q_i, particles with better solutions applies attractive force to charges with worse solutions. Conversely, the worse charges repel the charges with relatively better solutions. Therefore, the Equation (7) is used to calculate the total force F_i applied to point *i* (Birbil & Fang, 2003).

$$F_i = \sum_{j \neq i}^{m} \begin{cases} (x^j - x^i)\dfrac{q^i q^j}{||x^j - x^i||^2} & if\ f(x^j) < f(x^i), \\ (x^i - x^j)\dfrac{q^i q^j}{||x^j - x^i||^2} & if\ f(x^j) \geq f(x^i), \end{cases} \quad \forall\ i \tag{7}$$

where F_i is the net force, *m* is the population size, i and j represents particles in the population where i $\neq j$, x^i and x^j also represent position vectors for i^{th} and j^{th} particles respectively and q represents the computed charges.

2.1.4 Movement of Charges based on force calculation

Equation (8) states that the point *i* is pushed in the force's direction by a random step length following the evaluation of the total force vector F_i. In order to update the positions of the charges and balance exploration and exploitation, the random step length, λ, is assumed to be evenly distributed between 0 and 1 (Birbil & Fang, 2003).

$$x_{i,k} = \begin{cases} x_{i,k} + \lambda \,.\, F_{i,k} \,.\, (u_k - x_{i,k}) & if\ \ F_{i,k} > 0, \\ x_{i,k} + \lambda \,.\, F_{i,k} \,.\, (x_{i,k} - l_k) & otherwise. \end{cases} \tag{8}$$

$F_{i,k}$ is the normalized force that determines the movement, step length λ is between U (0, 1), l_k and u_k are the lower and upper bounds respectively and $x_{i,k}$ is a charge point in the search space.

2.1.5 Termination Criteria

The EM-like optimization algorithm ends after a declared fixed maximum number of iterations (MAXITER) is reached. When there is no discernible increase in fitness over iterations, convergence is presumed (Table 1).

Table 1 A pseudocode for the original Electromagnetism-like algorithm

Pseudocode of the EM-like Algorithm
The initial population of search solutions x_i (i = 1,2...n) is generated randomly according to Eq. (4).
Calculate fitness function f (x_i) for each search solution.
X* = the best search solution in terms of minimum f value
while (t < maximum number of iterations MAXITER)
for each search solution i
// Local search (optional – can be used to all solutions or just the best one)
Make local search according to Eq. (5), where LSITER and δ are parameters
end for
Update X* after local search
// Charge calculation
Calculate total charge qi for each solution according to Eq. (6)
// Force calculation
for each search solution i
F_i = 0 (vector)
for each other solution j ≠ i do
if f (x_j) < f (x_i) then
// attraction: calculate by Eq. (7), first case
else
// repulsion: calculate by Eq. (7), second case
end if
end for
end for
// Move solution by force
for each search solution i (except for the best solution X*)
Normalize the vector: $F_i = F_i / \|\|F_i\|\|$
Random step length λ = U (0,1)
Calculate x_i according to Eq. (8), coordinate-wise
end for
t = t+1
end while
return X*

2.2 Proposed Adaptive Chaotic-Opposition Based Learning Electromagnetism-Like Algorithm

The original EM-like algorithm proposed by Birbil and Cherng Fang in 2003 has no convergence guarantee and based on the study by (Lin et al., 2012), EM-like algorithm suffers premature convergence and can miss the global minimum of a function in minimization problems. Additionally, the original charge computation that determines either an attractive or repulsive forces between charges practically is unreliable for some types of optimization problems, rendering the movement of the charges within the search space inefficient, ultimately leading to slower convergence. Lastly, the findings from the study done by (Lin et al., 2012) suggests that the vanilla EM-like algorithm has poor performance on standard test functions. In this study, modifications to the original EM-like algorithm were done by stitching together three ideas used to create hybrid metaheuristics from (Rahnamayan et al., 2006; Yang & Deb, 2010; Y. Zhang et al., 2022) to address the major drawbacks of the vanilla EM-like algorithm.

2.2.1 Initialization Based on Opposition-Based Learning

(Rahnamayan et al., 2006) proposed a novel opposition-based learning concept which improved the efficiency of the Differential Evolution (DE) algorithm, outperforming it in terms of convergence speed. Opposition based learning is now the heart of most numerical optimization problems (Kullampalayam Murugaiyan et al., 2024; Long et al., 2019; Yu et al., 2024). Opposition based initialization and regeneration is integrated to the vanilla EM-like initialization principle such that, for every charge position $x_{i,k}$ generated, its opposite is also evaluated with Equation (9).

$$x_i^{opp} = \alpha_i + b_i - x_i \tag{9}$$

where α_i and b_i refers the upper and lower boundaries, i represents the dimension and x_i in the initial particle position.

Unlike the vanilla EM-like algorithm with initial particle positions for only $x_{i,k}$, the search space coverage increases to $x_{i,k}$ and x_i^{opp} with the potential of increasing convergence speed of the EM-like algorithm. Furthermore, after every iteration, there is regeneration of opposites of current worse *p*% charges, again this potentially increases the convergence speed of the algorithm.

2.2.2 Force Scaling Driven by Chaotic Maps

Numerous Genetic-algorithm and PSO variations have included chaotic scheduling of parameters, such as adjusting the gravitational constant using a logistic map (Dai et al., 2015; Y. Zhang et al., 2022). In the original EM-like algorithm, the force calculation formula relies on (x^j- x^i) or (x^i- x^j). This parameter is usually fixed, leading to premature convergence and poor exploration. Incorporation of the chaotic maps based on Equation (10) below replaces this parameter with G(t) such that;

$$\begin{cases} u_{t+1} = \mu u_t(1 - u_t) \\ G(t) = G_{max} u_t \\ with\ \mu\ \approx 3.9\ and\ u_o\ \in (0,1) \end{cases} \tag{10}$$

where u_o is the initial value of the chaotic map, u_t is the chaotic map at time t, μ is a control parameter of logistic map, u_{t+1} is the chaotic map at time t+1, G_{max} is the maximum gravitational constant as a scaling factor. Critical limitations in striking a balance between exploration (global search) and exploitation (local refinement) are addressed by incorporating these parameters into the standard EM algorithm

2.2.3 Lévy-Flight Local Moves for Intensification

The Lévy flight was introduced by Xin-She Yang and Suash Deb in 2010 (Yang & Deb, 2010) when they proposed a new metaheuristic algorithm well known as the Cuckoo Search (CS). The lévy flight perturbations can be seen in some recent optimization algorithm modifications (Asigri et al., 2024; Z. Wang et al., 2022; B. Zhang et al., 2024). After charges are moved in the direction of the calculated force in the standard EM algorithm in Equation (8), a lévy flight perturbation is applied to the current best solution using the Equation (11) below;

$$x_{best}(t+1) = x_{best}(t) + \alpha \,.\, Levy(\beta) \tag{11}$$

where α is a scale factor that decreases with time, and $\beta \approx 1.5$ regulates the tail's fatness. The lévy flight is implemented to intensify the local movement of the charges ($x_{best}(t)$) enhancing them to explore uncovered regions within the search space to charge $x_{best}(t+1)$.

2.3 Flowchart to the Proposed ACO-EM-like Algorithm

ACOEM begins by generating a population of *N* candidate solutions and their opposites, which it scores and discards to leave the top *N*. It sets a chaotic seed for the logistic map and begins its main loop of at most T_{max} iterations. At each iteration it computes particle "charges" based on their fitness scores, updates a time varying force constant with the logistic map, and calculates attraction or repulsion forces among particles. The position of each particle is shifted accordingly, the best current solution can be perturbed by a Lévy -flight jump with some probability, and out-of-bounds positions are mapped back within the feasible area. Every *T*obl generations, the worst *p*% particles are re-generated via opposition-based sampling if their opposites are better. After re-computing fitness, the algorithm logs its progress and iterates again; upon termination, it returns the best solution discovered. The flowchart illustrating the proposed ACOEM-like algorithm is shown in Figure 1 below.

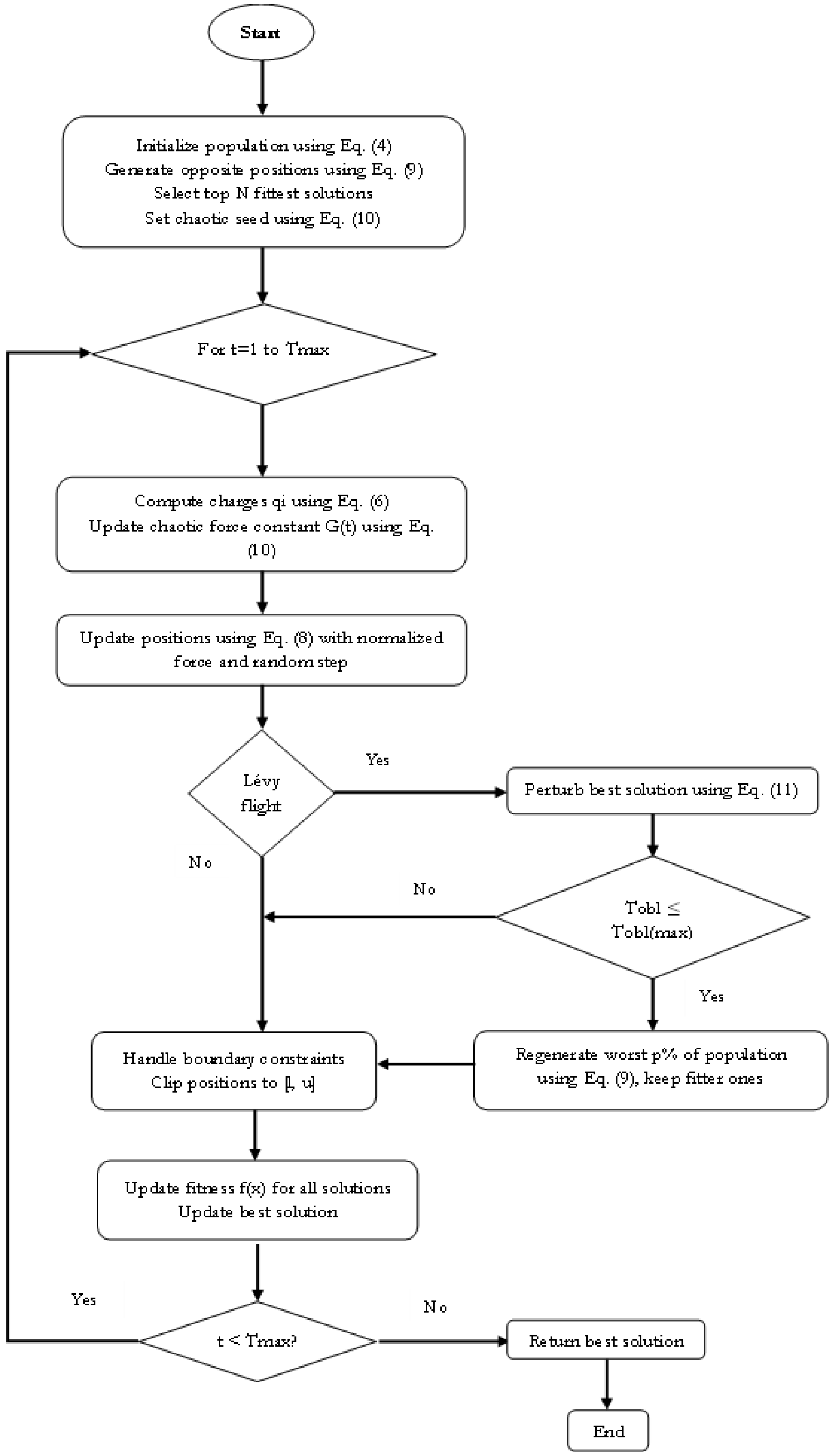

Figure 1 Flowchart of proposed ACOEM-like optimization algorithm

2.3.1 Data Acquisition and Pre-processing

2.3.1.1 NASA's Capacitor Electrical Overstress Dataset

The electrolytic capacitor degradation under electrical overstress dataset contains a total of 24 electrolytic capacitors in batches of 8 capacitors of 2200 μF capacitance, rated 10 V, each corresponding to the different voltage levels the experiment was performed, that is, 10 V,12 V and 14 V. The capacitors were connected to a circuit that replicated actual operating circumstances by applying the specified voltage for a specific period of time (Renwick et al., 2015). Data on the electrolytic capacitor electrical overstress experiment was downloaded from NASA's Prognostics Centre of Excellence (PCoE) dataset repository. The dataset has three MATLAB files ES10.mat, ES12.mat and ES14.mat. Each has a struct format containing "Transient Data", "EIS Data" and "Initial Date".

1. Transient Data: The transient data consists of data on:
   - V_o (the output voltage of the signal amplifier hardware) measured directly at the amplifier's output terminal representing the input voltage to the RC circuit during charge/discharge cycles.
   - V_L is the voltage across the load resistor of 100Ω to analyze the discharge behavior of the capacitor.

2. EIS Data: The electrolytic capacitors' Electrochemical Impedance Spectroscopy (EIS) data shows detailed electrical behavior at various frequencies. The following are the columns of the NASA's capacitor EIS dataset;
   - Frequency (freq/Hz) of the square wave amplified signal.
   - Re(Z)/Ohm is the resistive component of the complex impedance.
   - -Im(Z)/Ohm is the capacitive reactance representing the imaginary part of the impedance.
   - |Z|/Ohm representing the total impedance magnitude.
   - Phase(Z)/deg is the phase angle between current and voltage.
   - Series Capacitance (Cs/F)
   - Parallel Capacitance (Cp/F)
   - Other parameters include: the admittance parameters (Re (Y)/Ohm^{-1}, Im (Y)/ Ohm^{-1}, |Y|/Ohm^{-1} and Phase(Y)/deg) and electrical measurement parameters (time/s, < Ewe > /V, < I > /mA, Range, |Ewe|/V, and |I|/A), where < Ewe > /V is the potential of the working electrode.
   - Lastly, the "EIS Reference Table" tracks the dates with corresponding timestamps in hours spanning the entire duration of the experiment.

3. Initial Date: Contains the commencement date of the experiment '10/01/2014 10:00:00 AM'

Since there are many null data points in the transient data, only the EIS data was taken into account when analyzing the data for predicting the electrolytic capacitors RUL.

2.3.1.2 NASA-PCoE-Electrolytic Capacitor Aging Data Preprocessing

The following procedure was used for preprocessing the NASA Capacitor EIS data;

1. The frequency (freq/Hz), Re(Z)/Ohm, -Im(Z)/Ohm and |Z|/Ohm columns of the EIS data were extracted into excel files for each capacitor.
2. With each excel file corresponding to a particular timestamp in EIS Reference Table, ES10 C1 at 0 hours, ES10 C1 at 105 hours and ES10 C1 at 296 hours of capacitor 1 under 10 V electrical overstress was read into Panda's data frames and the EIS of the capacitor was visualized with the help of the seaborn library at the different time intervals that is 0, 105 and 296 hours. The electrochemical

impedance spectrum is displayed in Figure 2 at 0, 105, and 296 hours, respectively. It can be observed that the Re(Z)/Ohm increases from left to right, that is from 0 to 296 hours as the experiment progresses indicating changes in the capacitor's properties due to aging effects owing to increase in internal resistance. Also, increase in the Im(Z)/Ohm implies the decrease in capacitance as the experiment progresses.

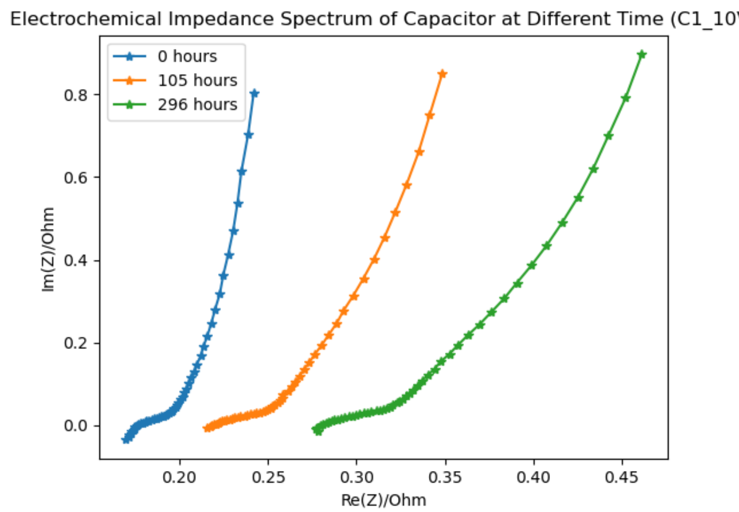

Figure 2 Electrochemical Impedance Spectrum of Capacitor at Different Time (C1 at 10V)

3. From the scipy.optimize library in Python, the curve fit function was imported. Based on the electrical overstress experiment described in (Renwick et al., 2015), the capacitor was modelled with the Equation (12) below as;

$$Z = R_{ESR} + \frac{1}{j\omega C} \tag{12}$$

where Z is the total impedance, R_{ESR} is the equivalent series resistance, C is the capacitance, *j* is the imaginary unit and ω is 2πf representing angular frequency. The above Equation (12) was defined as a function in a python script which extracts frequency (freq/Hz) and $|Z|/Ohm$ columns from each excel file. An initial guess for ESR was set to 0.1Ω assuming it has the lowest value during the start of the experiment and Capacitance (C) set to 2200μF as per the rating for the experiment (Renwick et al., 2015).

4. By using the function "curve_fit" for the impedance Equation (12), the ESR and capacitance values for every timestamp was extrapolated and the values were extracted. Hence, the aging data for capacitor 1 under 10 V, 12 V and 14 V electrical stress is visualized in Figure 3 below.

From Figure 3, it can be observed that the higher the voltage stress value, the greater the decrease in the capacitance value. At the three voltage levels, two significant phases can be observed;

- Phase I: There is a steady decrease in the capacitance values to a threshold of 20% below the actual capacitance rating. At this point, the capacitor is now deemed unsuitable for usage (Renwick et al., 2015).
- Phase II: There appears to be a capacitance recovery as the experiment progresses which can be attributed to a scientific phenomenon (Jha & Dong, 2024).

- The apparent capacitance recovery that can be seen later in electrical overstress aging in Figure 3 is due to electrolytic capacitors' dynamic self-healing behaviors. When a prolonged voltage stress occurs, there will be some area which will generate enough heat and that part of the liquid electrolyte will evaporate. This will temporarily reduce the effective capacitance of that electrolytic capacitor. Intermittent stress periods, or during a reduction in the electric field, will allow the residual liquid electrolyte to redistribute and re-wet over the etched aluminum oxide (Al_2O_3) dielectric layer. In addition, while this is taking place, the oxide layer can become electrochemically reformed and repair its microscopic defects which result in a partial restoration of the dielectric constant of the dielectric layer. These two mechanisms will create a transient measurement increase in the measured capacitance value which is well documented in (TDK Electronics Co., 2026) even though overall the capacitor is still permanently degraded.

5. The extracted features for the RUL analysis of the capacitors will be based on the phase I of the experiment and they include: Aging time in hours, R_{ESR}/Ω and Capacitance/F. The Figure 4 below shows how well they are correlated.

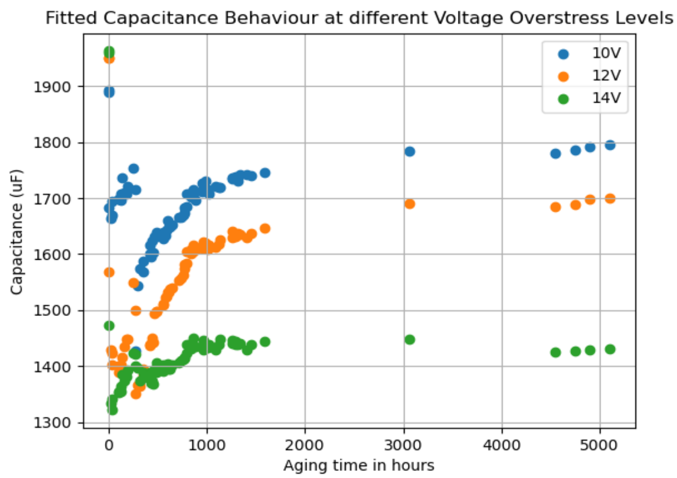

Figure 3 Fitted Capacitance Behavior at different Voltage Overstress Levels

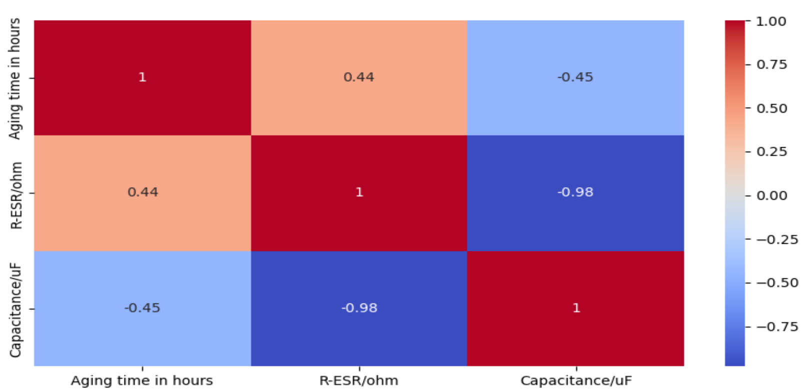

Figure 4 Correlation map of precursors for the Capacitor's RUL prediction

2.3.1.3 Electrolytic Capacitor Aging Data Preprocessing Workflow

Figure 5 below shows how data cleaning and feature engineering was done.

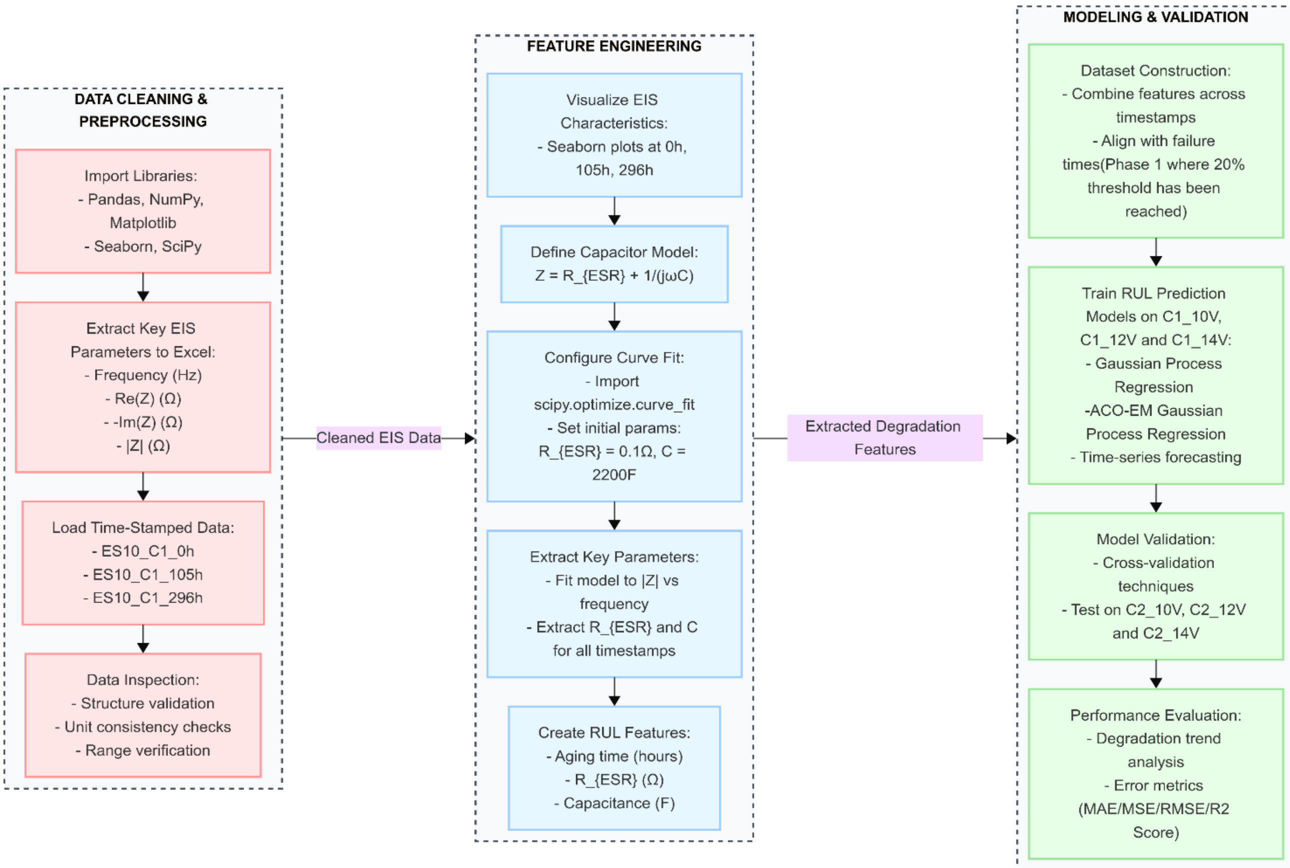

Figure 5 Capacitor EIS data preprocessing procedure

2.3.1.4 Testing of the Improved Algorithm

The ACOEM-like algorithm's efficacy was assessed using 10 classical benchmark functions obtained from (Tan et al., 2016) research work on EM-like algorithm improvement. The test functions include both uni-modal and multi-modal functions, which assess an algorithm's rate of convergence and its ability to locate global optima as the number of local optima increases, respectively. A description of the 10 test functions is shown in Table 2 and Table 3 below.

Table 2 Uni-modal and multi-modal test functions.

Function	Range	Dimension	FGO	Type
F1 Himmelblau	[-5.12, 5.12]	2	0	Multimodal
F2 Schaffer N2	[-100, 100]	2	0	Multimodal
F3 Rosenbrock	[-5, 10]	d = variable	0	Unimodal
F4 Booth	[-10, 10]	2	0	Unimodal
F5 Beale	[-4, 5, 4.5]	2	0	Multimodal
F6 Rastrigin	[-5, 12, 5.12]	d	0	Multimodal
F7 Six-Hump Camel	$x_1 \in$ [-3, 3], $x_2 \in$ [-2, 2]	2	≈ -1.0316	Multimodal
F8 Ackley	[-32.768, 32.768]	d	0	Multimodal
F9 Sphere	[-5.12, 5.12]	d	0	Unimodal
F10 Shubert	[-10, 10]	2	-186.7309	Multimodal

Table 3 Description of benchmark functions

Function	Description
F1	$\min f(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$
F2	$\min f(x) = 0.5 + \frac{sin^2(x_1^2 - x_2^2) - 0.5}{(1 + 0.001(x_1^2 + x_2^2))^2}$
F3	$\min f(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$
F4	$\min f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$
F5	$\min f(x) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 - x_1x_2^3)^2$
F6	$\min f(x) = 10d + \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i)]$
F7	$\min f(x) = (4 - 2.1x_1^2 + \frac{x_1^4}{3})x_1^2 + x_1x_2 + (4x_2^2 - 4)x_2^2$
F8	$\min f(x) = -20\exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(2\pi x_i)\right) + 20 + e$
F9	$\min f(x) = \sum_{i=1}^{d} x_i^2$
F10	$\min f(x) = [\sum_{i=1}^{5} i\cos((i+1)x_1 + i)][\sum_{i=1}^{5} i\cos((i+1)x_2 + i)]$

The proposed ACOEM-like algorithm is bench-marked with traditional algorithms such as the Particle Swarm Optimization (PSO), the Genetic Algorithm, the original EM-like (Birbil & Fang, 2003) and existing EM-like variants (Ragb et al., 2023; Tan et al., 2016; Zhu et al., 2024) with these metrics;

- Minimum or best value: This the lowest objective function value obtained across all runs.
- Maximum or worst value: This is the highest objective function value obtained among all runs.
- Mean Value: This is the average objective function value over all runs.
- Standard deviation: This measures the consistency of the algorithm's results over all runs.

The simulations were carried out in MATLAB R2022b on a computer with the following specifications; HP ENVY dv6, with 2.30 GHz AMD A10-4600M APU with Radeon (tm) HD Graphics with 8 GB ram size (7.46 GB usable) in Windows 10 operating system.

The proposed ACOEM-like algorithm performance was compared with the PSO and GA based on the test functions described in Table 2 and Table 3 for a maximum number of iterations of 100, set at 20 runs to avoid stochastic discrepancy and particle size of 50.

The ACOEM parameters were set to local search iterations (LSITER) of 1000, beta levy (β_{lev}) of 1.5, p levy (p_{lev}) of 0.1, initial step size (α_o) was 0.1, the logistic map parameter (μ) was set to 3.9, the maximum gravitational constant (G_{max}) was 1 with initial chaotic seed (μ_o) set at 0.5. The regeneration interval of the

Opposition-Based Learning (T_{obl}) was set to 50 and worse replacement percentage (p%) of 20% was used. The proposed ACOEM-like algorithm performance was compared with the original EM-like and variants based on the test functions described in Table 2 and Table 3 for a maximum number of iterations of 200, set at 20 runs to avoid stochastic discrepancy and particle size of 10. Lastly, the Friedman's rank test was used to assign statistical rankings to the EM-like variants and the proposed ACO-EM algorithm.

2.4 Model Development and Optimization

The Gaussian Process Regression was implemented using MATLAB programming. The GPR model was created through specifying the Radial Basis Function (RBF) as the kernel function and this was used to characterize the relationship between aging time, R-ESR/ohm and capacitance. The GPR model was trained using data from capacitor 1 subjected to electrical overstress at 10 V, 12 V, and 14 V, and was validated with data from capacitor 2 under the same voltage conditions. This was performed using the pre-processed NASA dataset focused on electrical overstress in capacitors. Negative log marginal likelihood (NLML) is used in the Gaussian Process Regression to model the observed data y as an expression of a Gaussian Process, which is a function of the covariance function with hyperparameters. The Radial Basis Function (RBF) kernel was the covariance function used in the training process and is expressed in Equation (13) as;

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_{i=1}^{d}\frac{(x_i - x_i')^2}{\ell_i^2}\right) \tag{13}$$

where σ_f^2 is the signal variance representing the square of the standard deviation and ℓ_i^2 is length scale. x and x' are two input points from the dataset. These are the hyperparameters of the RBF. The NLML is also expressed in Equation (14) as;

$$NLML\ (\theta) = -\log p(y|X,\theta) = \frac{1}{2}\, y^T(K + \sigma_n^2 I)^{-1}y + \frac{1}{2}\log\det(K + \sigma_n^2 I) + \frac{n}{2}\log(2\pi) \tag{14}$$

where K refers to the RBF kernel, I is $n \times n$ identity matrix where n is the number of training samples and σ_n^2 is the noise variance (another hyperparameter). The parameter y is a vector of observed output values and y^T is the transpose of the vector. The NLML served as the objective function minimized by the ACOEM algorithm for optimal results. The hyperparameters which were tuned by the ACOEM are σ_f^2, ℓ_i^2 and σ_n^2 representing signal variance, length scale and noise variance respectively. The Figure 6 below shows the workflow of the hyperparameter tuning with the proposed ACO-EM.

3. Results and Discussion

3.1 Comparison of the Proposed ACOEM-like Algorithm to PSO and GA

ACOEM, GA and PSO performance on each of the ten (10) optimization test functions were analyzed by comparing its best (minimum value) and worst (maximum value) solutions, as well as its mean values and standard deviations across 20 runs as shown in Table 4. The ACOEM outperformed the standard PSO and GA on all the 10 test functions in terms of solutions to find the function's global optimum and convergence. The figures below show the convergence behavior of the ACOEM-like algorithm, GA and PSO on the selected benchmark functions. It can be observed that for all the 10 functions, ACOEM-like algorithm rapidly decreases with an increase in the number of iterations, achieving the most optimal results. This reflects the impressive exploitation capability of ACOEM-like algorithm, its proficiency in evading local minima, and its enhanced optimization precision.

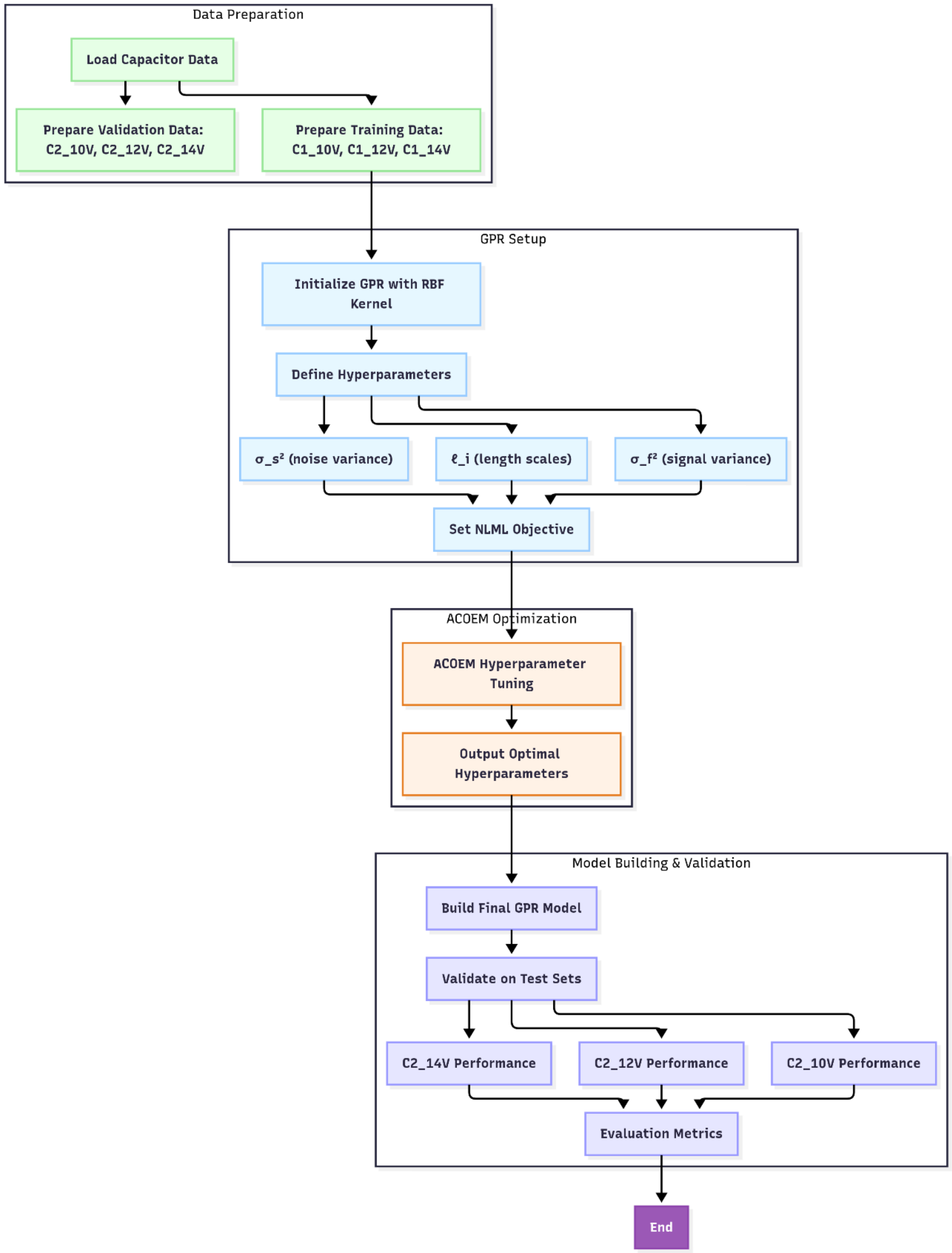

Figure 6 The GPR optimization process

Table 4 Performance benchmarking of ACOEM, PSO and GA for particle size of 50, 100 iterations and 20 runs.

Function		ACOEM	PSO	GA
F1	Best	$\mathbf{9.9243 \times 10^{-10}}$	1.8507×10^{-4}	5.2722×10^{-2}
	Worst	$\mathbf{4.9495 \times 10^{-8}}$	2.6356×10^{-2}	7.4819
	Mean	$\mathbf{9.0936 \times 10^{-9}}$	3.5000×10^{-3}	1.9792
	SD	$\mathbf{1.1846 \times 10^{-8}}$	5.7279×10^{-3}	1.9697
F2	Best	$\mathbf{3.3956 \times 10^{-11}}$	2.0931×10^{-7}	3.3924×10^{-2}
	Worst	$\mathbf{5.3764 \times 10^{-5}}$	3.7533×10^{-3}	3.9669×10^{-1}
	Mean	$\mathbf{5.4508 \times 10^{-6}}$	7.3950×10^{-4}	1.5089×10^{-1}
	SD	$\mathbf{1.2745 \times 10^{-5}}$	1.0821×10^{-3}	1.0640×10^{-1}
F3	Best	$\mathbf{1.0579 \times 10^{-7}}$	1.0728×10^{-4}	1.8576×10^{-1}
	Worst	5.2246	$\mathbf{5.5007 \times 10^{-3}}$	23.3770
	Mean	2.6125×10^{-1}	$\mathbf{2.1528 \times 10^{-3}}$	6.3617
	SD	1.1683	$\mathbf{1.6591 \times 10^{-3}}$	6.2930
F4	Best	$\mathbf{5.6133 \times 10^{-10}}$	1.4522×10^{-5}	2.3316×10^{-2}
	Worst	2.8087	$\mathbf{4.4728 \times 10^{-3}}$	6.3535
	Mean	2.3782×10^{-1}	$\mathbf{1.1500 \times 10^{-3}}$	2.0022
	SD	7.4519×10^{-1}	$\mathbf{1.0170 \times 10^{-3}}$	1.8834
F5	Best	$\mathbf{1.3896 \times 10^{-7}}$	3.4653×10^{-6}	2.0416×10^{-3}
	Worst	$\mathbf{2.8702 \times 10^{-6}}$	7.6308×10^{-1}	2.0216
	Mean	$\mathbf{1.1969 \times 10^{-6}}$	3.8296×10^{-2}	3.9680×10^{-1}
	SD	$\mathbf{6.9918 \times 10^{-7}}$	1.7060×10^{-1}	5.0069×10^{-1}
F6	Best	$\mathbf{1.0040 \times 10^{-7}}$	27.5630	49.6090
	Worst	$\mathbf{1.3384 \times 10^{-6}}$	47.2680	93.8350
	Mean	$\mathbf{4.6561 \times 10^{-7}}$	36.0930	65.3870
	SD	$\mathbf{2.7894 \times 10^{-7}}$	4.9728	11.1290
F7	Best	**-1.0316**	**-1.0316**	-1.0311
	Worst	**-1.0316**	-1.0313	-8.8940×10^{-1}
	Mean	**-1.0316**	-1.0315	-9.6827×10^{-1}
	SD	$\mathbf{5.1907 \times 10^{-10}}$	9.3939×10^{-5}	4.4839×10^{-2}
F8	Best	$\mathbf{2.1761 \times 10^{-4}}$	3.9304	13.4330
	Worst	$\mathbf{5.6636 \times 10^{-4}}$	6.7410	18.7930
	Mean	$\mathbf{3.7760 \times 10^{-4}}$	5.4937	16.9140
	SD	$\mathbf{9.4639 \times 10^{-5}}$	6.8854×10^{-1}	15.0480
F9	Best	$\mathbf{1.0098 \times 10^{-9}}$	1.3793×10^{-1}	9.1582
	Worst	$\mathbf{1.1428 \times 10^{-8}}$	5.0128×10^{-1}	26.1340
	Mean	$\mathbf{2.5697 \times 10^{-9}}$	3.5006×10^{-1}	15.1830
	SD	$\mathbf{2.2840 \times 10^{-9}}$	1.1029×10^{-1}	4.5420
F10	Best	$\mathbf{-1.8673 \times 10^{2}}$	-1.8670×10^{2}	-1.8010×10^{2}
	Worst	$\mathbf{-1.8673 \times 10^{2}}$	-1.8551×10^{2}	-7.9068×10^{1}
	Mean	$\mathbf{-1.8673 \times 10^{2}}$	-1.8626×10^{2}	-1.3531×10^{2}
	SD	$\mathbf{1.7309 \times 10^{-5}}$	3.8183×10^{-1}	3.4775×10^{1}

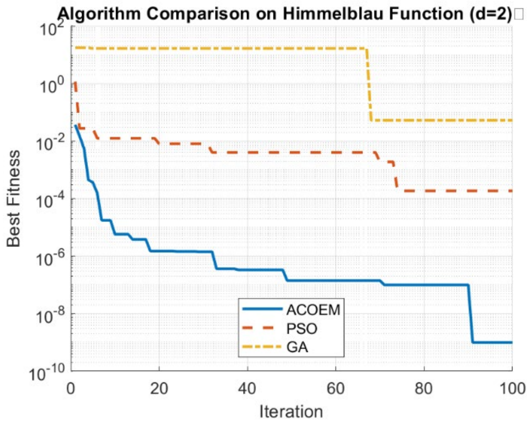

Figure 7 Convergence curves for Himmelblau function (F1)

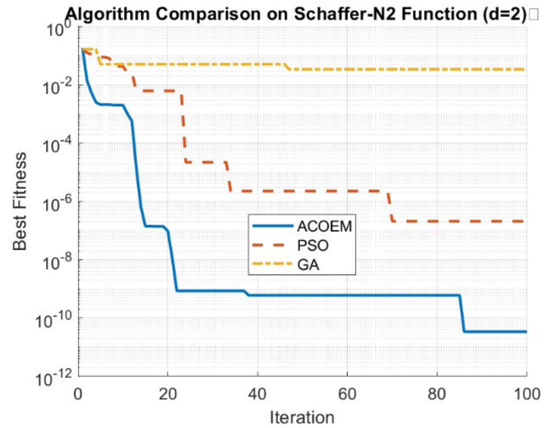

Figure 8 Convergence curves for Schaffer N2 function (F2)

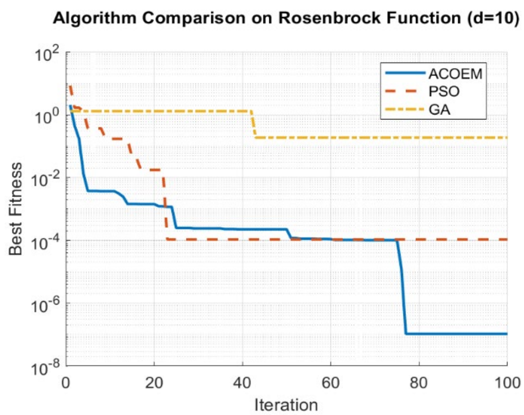

Figure 9 Convergence curves for Rosenbrock function (F3)

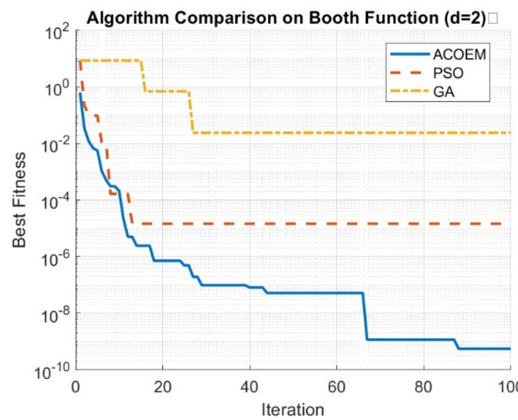

Figure 10 Convergence curves for Booth function (F4)

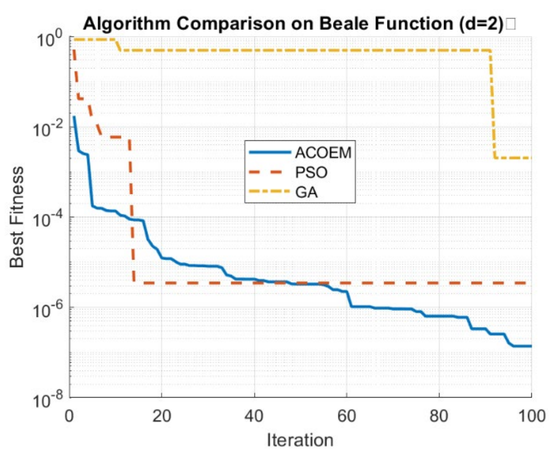

Figure 11 Convergence curves for Beale function (F5)

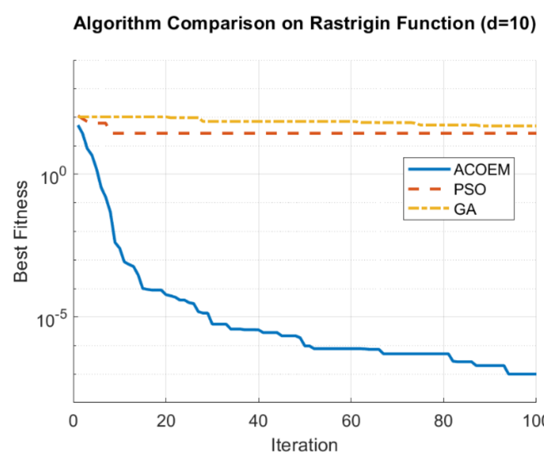

Figure 12 Convergence curves for Rastrigin function (F6)

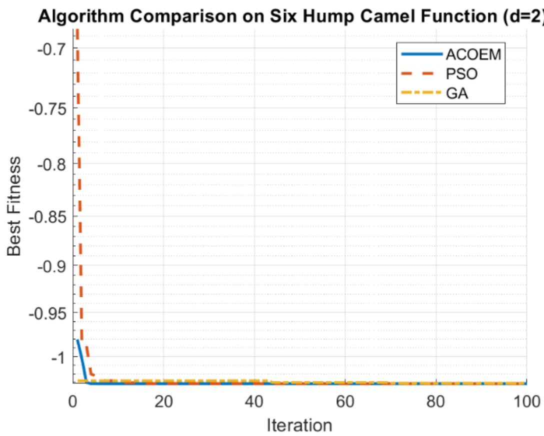

Figure 13 Convergence curves for Six-Hump Camel function (F7)

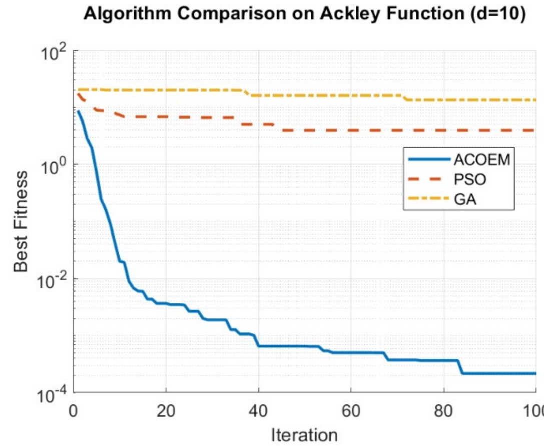

Figure 14 Convergence curves for Ackley function (F8)

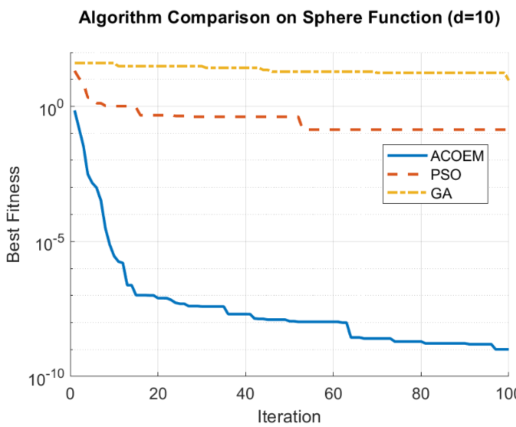

Figure 15: Convergence curves for Sphere function (F9)

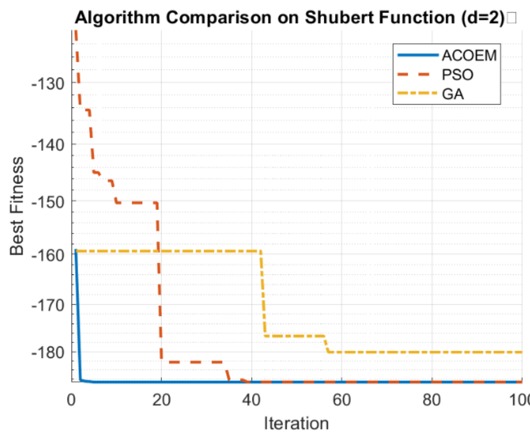

Figure 16 Convergence curves for Shubert function (F10)

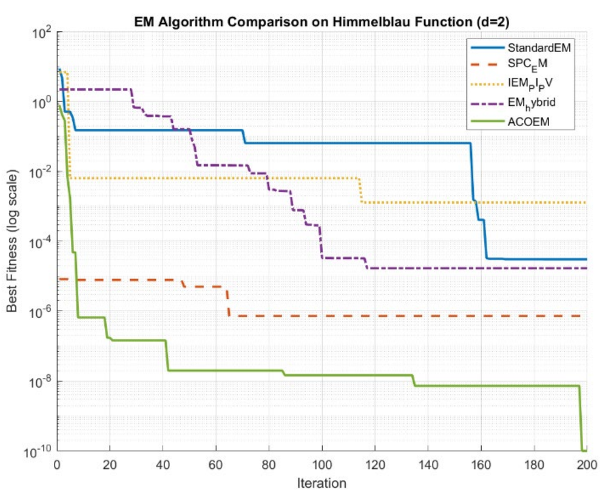

Figure 17 Convergence curves for F1

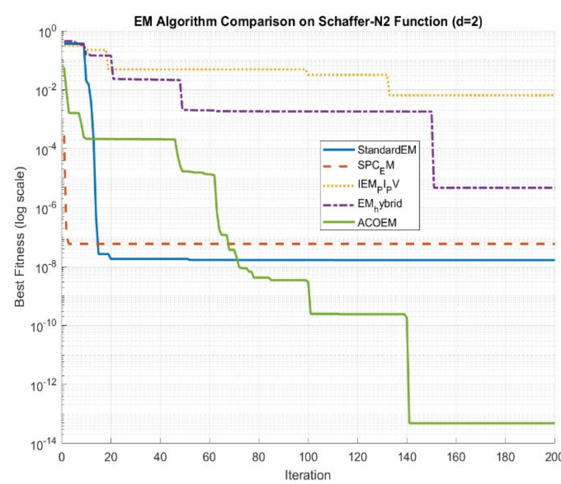

Figure 18 Convergence curves for F2

3.2 Comparison of the Proposed ACOEM-like algorithm to Existing EM-like variants

The original EM-like (Birbil & Fang, 2003), split-probe and compare (SPC-EM) (Tan et al., 2016), adaptive step-size (IEM-PI-PV) (Ragb et al., 2023) and lévy flights and chaotic maps(EM-hybrid) (Zhu et al., 2024) performances on each of the ten (10) optimization test functions were analyzed by comparing its best (minimum value) and worst (maximum value) solutions, as well as its mean values and standard deviations across 20 runs in Table 5. The figures below reveal the proposed ACOEM-like algorithm demonstrates superior optimization outperforming the other algorithms in obtaining the optimum solutions for the test functions (F1, F2, F3, F4, F8 and F9) due to its ability to balance exploration and exploitation.

Table 5 Performance benchmarking of ACOEM, Standard-EM, SPC-EM, IEM-PI-PV and EM-hybrid for population size of 10 particles, 200 iterations for 20 runs

Function		Original EM	SPC-EM	IEM-PI-PV	EM-Hybrid	ACOEM
F1	Best	3.04×10^{-5}	7.25×10^{-7}	1.28×10^{-3}	1.68×10^{-5}	**1.01×10^{-10}**
	Worst	8.48×10^{-2}	**8.09×10^{-6}**	6.53×10^{-2}	1.03×10^{-1}	2.13
	Mean	2.41×10^{-2}	**5.70×10^{-6}**	2.88×10^{-2}	1.81×10^{-2}	1.06×10^{-1}
	SD	2.80×10^{-2}	**2.41×10^{-6}**	2.30×10^{-2}	3.06×10^{-2}	4.76×10^{-1}
F2	Best	1.66×10^{-8}	5.88×10^{-8}	6.49×10^{-3}	4.71×10^{-6}	**4.84×10^{-14}**
	Worst	5.04×10^{-2}	**5.38×10^{-4}**	5.86×10^{-2}	1.69×10^{-2}	2.08×10^{-3}
	Mean	1.57×10^{-2}	**7.63×10^{-5}**	3.05×10^{-2}	2.11×10^{-3}	1.34×10^{-4}
	SD	1.59×10^{-2}	**1.65×10^{-4}**	1.46×10^{-2}	4.32×10^{-3}	4.62×10^{-4}
F3	Best	6.31×10^{-9}	5.60×10^{-5}	1.48×10^{-14}	3.85×10^{-6}	**1.05×10^{-19}**
	Worst	3.25×10^{-3}	8.72×10^{-1}	1.67×10^{-2}	2.87×10^{-1}	**2.84×10^{-5}**
	Mean	5.54×10^{-4}	1.88×10^{-1}	2.26×10^{-3}	6.25×10^{-2}	**1.73×10^{-6}**
	SD	1.05×10^{-3}	3.10×10^{-1}	4.47×10^{-3}	8.80×10^{-2}	**6.36×10^{-6}**
F4	Best	2.56×10^{-5}	1.93×10^{-6}	1.33×10^{-3}	7.05×10^{-4}	**7.11×10^{-10}**
	Worst	4.76×10^{-2}	**4.33×10^{-6}**	7.67×10^{-2}	24.8	3.78
	Mean	9.91×10^{-3}	**3.50×10^{-6}**	2.10×10^{-2}	1.98	1.89×10^{-1}
	SD	1.18×10^{-2}	**7.76×10^{-7}**	1.76×10^{-2}	5.44	8.44×10^{-1}
F5	Best	6.98×10^{-5}	2.58×10^{-7}	7.91×10^{-5}	1.41×10^{-5}	**3.52×10^{-8}**
	Worst	1.37×10^{-2}	**6.22×10^{-7}**	2.25×10^{-2}	1.81	8.03×10^{-1}
	Mean	5.00×10^{-3}	**3.24×10^{-7}**	5.16×10^{-3}	6.83×10^{-1}	4.02×10^{-2}
	SD	4.17×10^{-3}	**1.04×10^{-7}**	5.50×10^{-3}	5.79×10^{-1}	1.80×10^{-1}
F6	Best	5.47	1.83×10^{-4}	8.50	**7.40×10^{-12}**	3.72×10^{-8}
	Worst	37.40	3.57×10^{-4}	40.70	18.80	**3.68×10^{-7}**
	Mean	17.90	2.84×10^{-4}	15.00	3.56	**1.28×10^{-7}**
	SD	9.27	5.84×10^{-5}	7.21	4.75	**9.10×10^{-8}**
F7	Best	-1.031619	**-1.031628**	-1.031141	**-1.031628**	**-1.031628**
	Worst	-1.024022	**-1.031628**	-1.020894	-8.56×10^{-1}	**-1.031628**
	Mean	-1.030134	**-1.031628**	-1.028538	-1.015151	**-1.031628**
	SD	2.23×10^{-3}	6.61×10^{-8}	3.11×10^{-3}	3.93×10^{-2}	**2.26×10^{-10}**
F8	Best	1.75	5.81×10^{-3}	2.14	2.27×10^{-4}	**1.16×10^{-4}**
	Worst	4.37	1.12×10^{-2}	4.57	7.71	**3.70×10^{-4}**
	Mean	3.10	9.81×10^{-3}	3.00	1.81	**1.91×10^{-4}**
	SD	6.14×10^{-1}	1.50×10^{-3}	6.26×10^{-1}	1.98	**6.56×10^{-5}**
F9	Best	2.08×10^{-2}	2.62×10^{-7}	1.78×10^{-2}	7.86×10^{-10}	**8.61×10^{-11}**
	Worst	1.16×10^{-1}	1.92×10^{-6}	7.58×10^{-2}	2.11×10^{-1}	**1.34×10^{-9}**
	Mean	4.54×10^{-2}	1.33×10^{-6}	4.18×10^{-2}	2.58×10^{-2}	**5.65×10^{-10}**
	SD	2.32×10^{-2}	3.83×10^{-7}	1.68×10^{-2}	5.22×10^{-2}	**3.30×10^{-10}**
F10	Best	-186.72	**-186.73**	-186.26	**-186.73**	**-186.73**
	Worst	-163.59	**-186.73**	-166.86	-150.67	**-186.73**
	Mean	-183.20	**-186.73**	-181.10	-179.91	**-186.73**
	SD	5.02	5.95×10^{-4}	5.44	11.27	**1.42×10^{-6}**

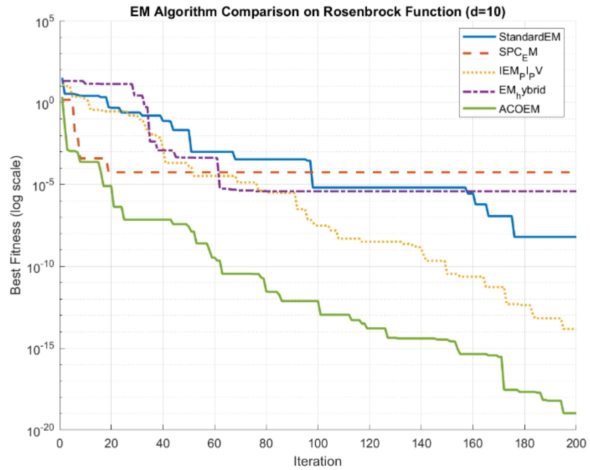

Figure 19 Convergence curve for F3

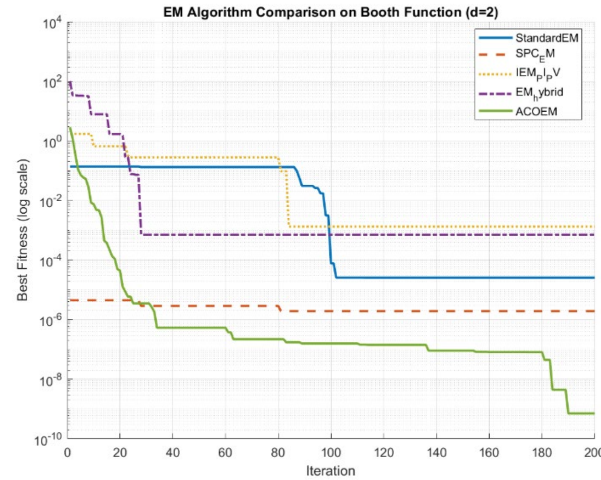

Figure 20 Convergence curves for F4

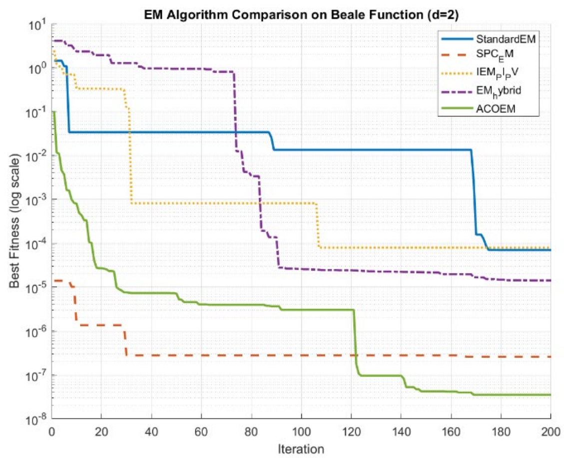

Figure 21 Convergence curves for F5

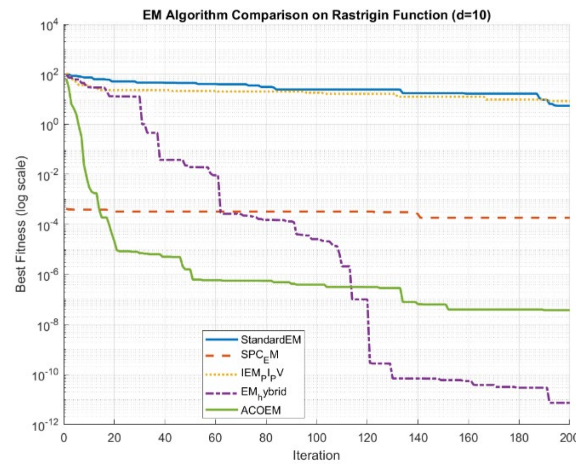

Figure 22 Convergence curves for F6

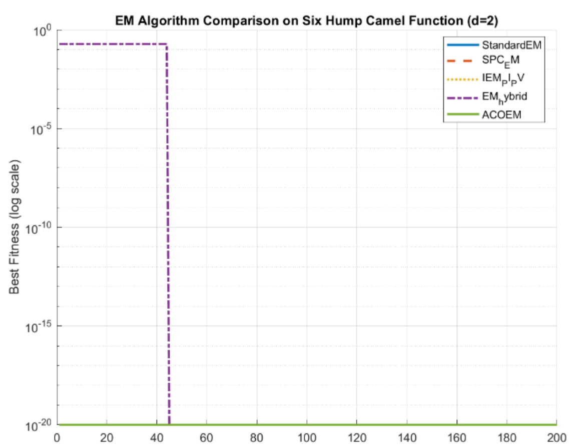

Figure 23 Convergence curves for F7

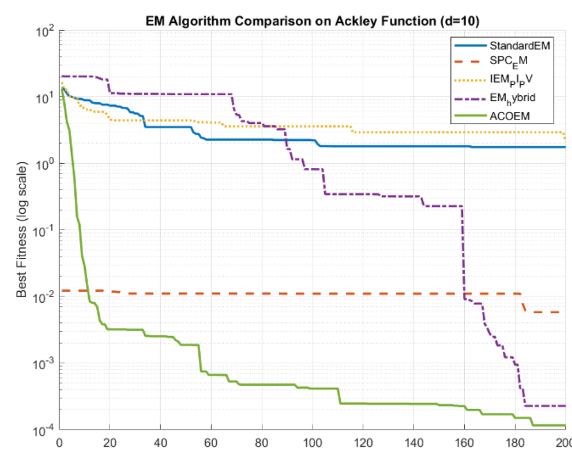

Figure 24 Convergence curves for F8

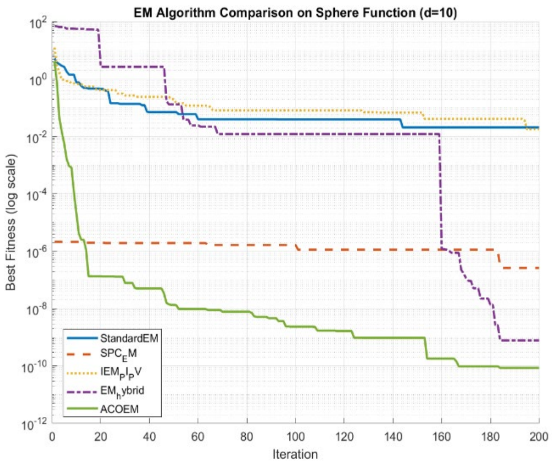

Figure 25 Convergence curves for F9

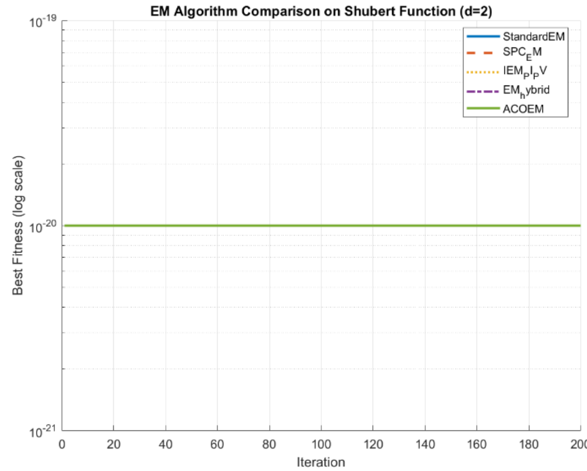

Figure 26 Convergence curves for F10

3.3 Friedman Ranking Test

In developing the proposed ACO-EM algorithm, the focus was on enhancing the minimization capabilities of the algorithm for optimal finetuning of the Gaussian Process Regressor hyperparameters. Therefore, the Friedman ranking test was used to compare the optimization algorithms based on the lowest values (best) for each of the 10 functions (F1 to F10) in Table 6.

Table 6 Friedman ranking test results for algorithms

Function	Original EM	SPC-EM	IEM-PI-PV	EM-Hybrid	ACOEM
F1	4	2	5	3	1
F2	2	3	5	4	1
F3	3	5	2	4	1
F4	3	2	5	4	1
F5	4	2	5	3	1
F6	4	3	5	1	2
F7	4	2	5	2	2
F8	4	3	5	2	1
F9	5	3	4	2	1
F10	4	2	5	2	2
Average Rank	3.7	2.7	4.6	2.7	**1.3**

ACOEM-like algorithm, with the lowest average rank of 1.3, surpassed SPC-EM and EM-LF, both at 2.7, Standard EM at 3.7, and IEM-PI-PV at 4.6, as indicated by the aforementioned average ranks.

3.4 RUL Prediction with Vanilla GPR and ACOEM-GPR on Pre-processed NASA Capacitor Data

Gaussian Process Regression (GPR) base model was developed using MATLAB software through “fmincon” optimization technique. Subsequently, GPR hyperparameters were optimized using a proposed ACOEM framework. The visualizations in Figure 27, Figure 28, and Figure 29 illustrate the standard GPR predictions for capacitor#2 subjected to 10 V, 12 V, and 14 V electrical stress, respectively.

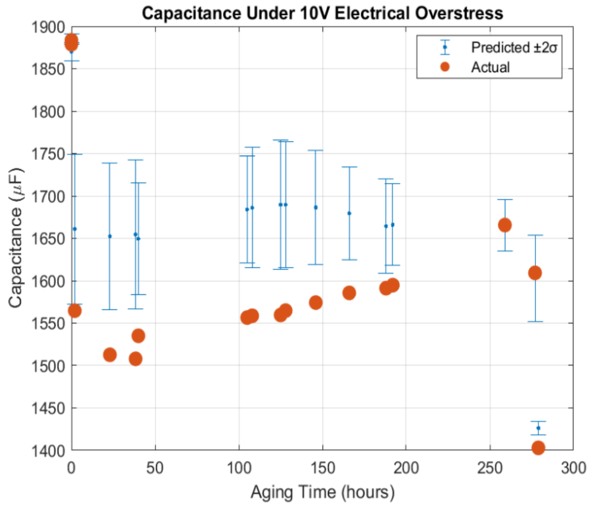

Figure 27 Vanilla GPR model predictions on the C2 at 10V (Trained with C1 at 10V)

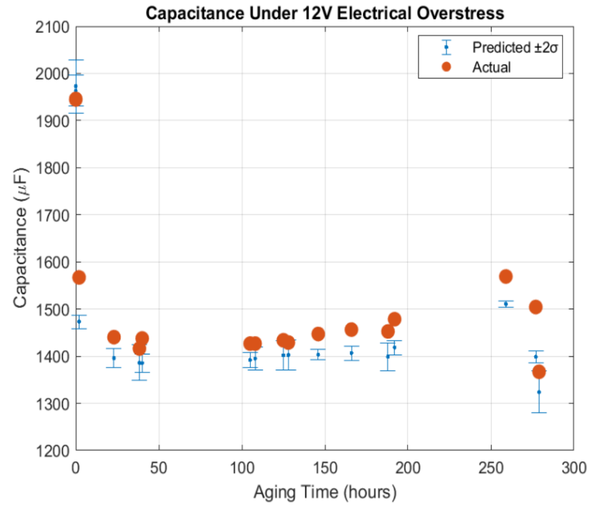

Figure 28 Vanilla GPR model predictions on the C2 at 12V (Trained with C1 at 12V)

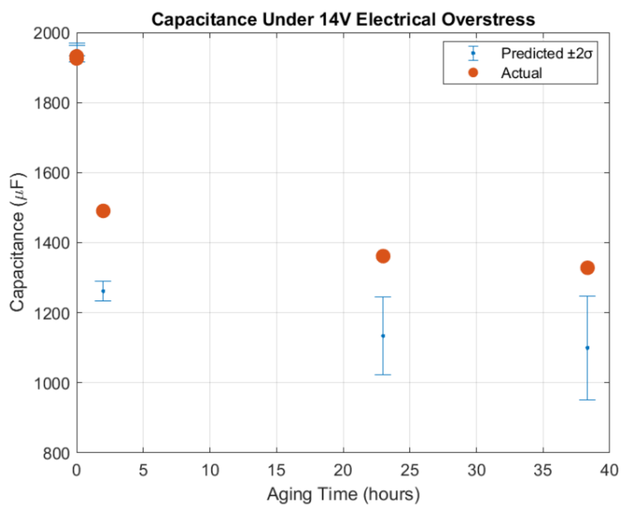

Figure 29 Vanilla GPR model predictions on the C2 at 14V (Trained with C1 at 14V)

Table 7 below shows the performance of the vanilla GPR predictions on the test data.

Table 7 Vanilla GPR Model performance evaluation for various voltage levels

Test Data	MSE	RMSE	MAE	R2Score	NLML
C2 at 10V	9457.81	97.25	82.18	0.3015	1.2
C2 at 12V	2749.89	52.44	47.37	0.8989	-0.79
C2 at 14V	31448.24	177.34	143.80	0.5615	-0.1

The above results in Table 7 show that the vanilla GPR model is excellent at generalization on the 12 V degradation data but exhibits large deviations in prediction accuracy on the 10 V and 14 V data indicating scope for improvement at 10V and 14V voltage stress levels. Hence, Adaptive Chaotic Opposition-based learning Electromagnetism-like Optimization is used as a replacement for the default optimizer fmincon in

MATLAB to optimize kernel hyperparameters more efficiently. The Figure 30 shows the convergence curve of the proposed ACO-EM algorithm in minimizing the NLML objective function for optimal hyperparameter values for C1 at 10V training. The Figure 31 shows the improved predictions of the ACOEM-GPR model on the same capacitor#2 under 10 V electrical stress data

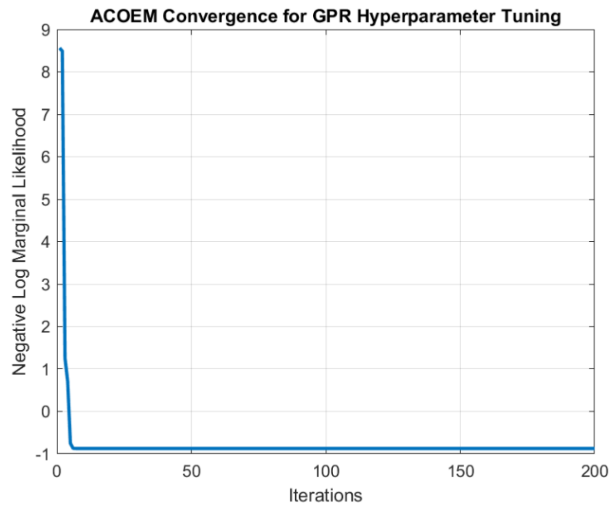

Figure 30 Negative Log Marginal Likelihood (NLML) convergence curve on the 10V dataset minimized with ACOEM

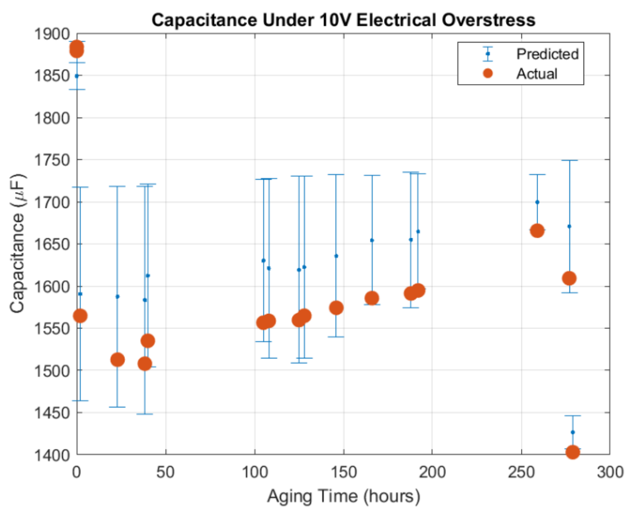

Figure 31 ACOEM-GPR model predictions on the C2 at 10V (Trained with C1 at 10V)

The Table 8 below shows the performance of the ACOEM-GPR model predictions on the test data.

Table 8 ACOEM-GPR Model performance evaluation for 10V capacitor data

Test Data	MSE	RMSE	MAE	R2Score
C2 at 10V	3409.68	58.39	54.42	0.7482

At the end of the optimization, the optimal Negative log likelihood (NLML) value was found to be -0.8741. This is a more negative value indicating a superior fit by the ACOEM-GPR model compared to the vanilla GPR case where the fmincon converged to suboptimal hyperparameter configurations. There was

substantial increase in the model's *R*2 value and reduction in the MSE, RMSE and MAE.
The Figure 32 shows the convergence curve of the proposed ACO-EM algorithm in minimizing the NLML objective function for optimal hyperparameter values for C1 at 14V training. The Figure 33 shows the improved predictions of the ACOEM-GPR model on the same capacitor#2 under 14 V electrical stress data.

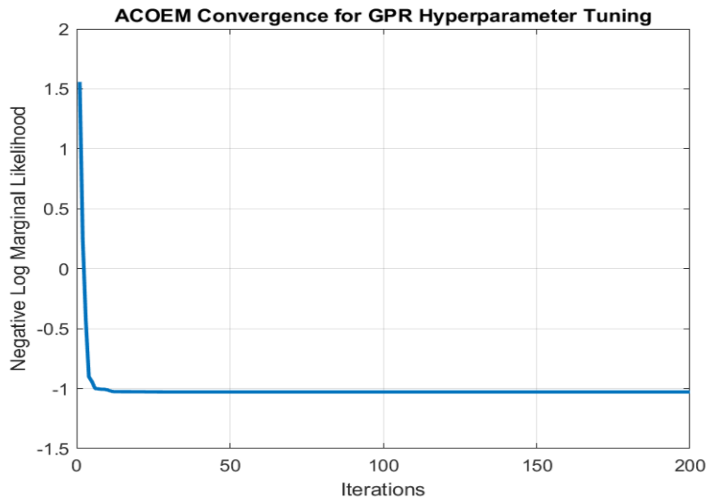

Figure 32 Negative Log Marginal Likelihood (NLML) convergence curve on the 14V dataset minimized with ACOEM

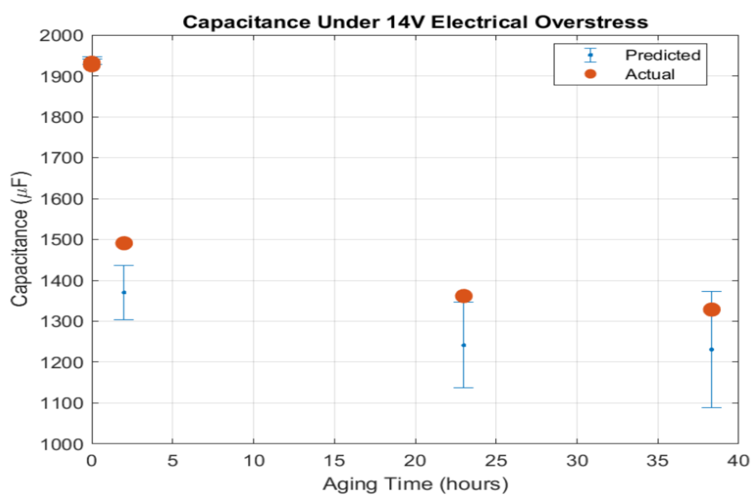

Figure 33 ACOEM-GPR model predictions on the C2 at 14V (Trained with C1 at 14V)

Table 9 below shows the performance of the ACOEM-GPR model predictions on the test data.

Table 9 ACOEM-GPR Model performance evaluation for 14V capacitor data

Test Data	MSE	RMSE	MAE	R2Score
C2 at 14V	7740.93	87.98	71.69	0.8921

At the end of the optimization, the optimal Negative log likelihood (NLML) value was found to be -1.0280, again, demonstrating a better fit of the ACOEM-GPR model compared to the base case scenario. There was substantial increase in the model's *R*2 value and reduction in the MSE, RMSE and MAE.

4. Conclusion

This study presents an enhanced Electromagnetism-Like Algorithm via Adaptive Chaotic Opposition-Based Learning (ACOEM). The method mitigates the key limitations of EM including premature convergence, insufficient exploitation of areas in proximity to local optima. It also deals with ineffective charge movement in higher-dimensional spaces. The ACOEM algorithm produces improved convergence rate, and greater diversity. It utilizes opposition-based learning during its initial phase and chaotic map-based force scaling to balance exploration and exploitation. In addition, the ACOEM algorithm incorporates a Lévy flight operator to aid the avoidance of local optima and increase the search time near promising solutions. The performance of the proposed algorithm is compared with the classical EM, Particle Swarm Optimization (PSO), and Genetic Algorithm (GA). The comparison analysis is also performed on other EM variants like SPC-EM, IEM-PI-PV, and EM hybrid algorithms on ten benchmark functions. The experimental results show that ACOEM outperforms PSO, GA, and other EM variants with respect to solution accuracy and convergence rate. This claim is further supported by the Friedman rank test with the lowest mean rank of ACOEM equals to 1.3. The ACOEM-GPR framework significantly improves the predictive performance of the GPR hyperparameter tuning method. With respect to the 10V of electrical overstress, using the ACOEM-GPR framework resulted in an increase in the R^2 score of GPR from 0.3015 to 0.7482. The mean squared error (MSE) was reduced from $9458\mu F^2$ to $3410\mu F^2$ and the root mean squared error (RMSE) was reduced from 97.3μF to 58.4μF. In addition, when assessed using the 14V of electrical overstress, R^2 increased from 0.5615 to 0.8921 and again MSE and RMSE significantly reduced. The results of this research show that ACOEM-GPR is capable of delivering scalable results for predictive maintenance of electrical power converters. Overall, the ACOEM optimization framework represents a valuable and powerful technique for solving real-world problems that involve precise hyperparameter tuning and will lead to more accurate predictive maintenance and more reliable systems.

It is important to recognize some limitations of this study. First, the model (GPR) training and testing was performed on capacitors 1 and 2 for each voltage (10V, 12V, and 14V), respectively. Therefore, while the ability of the model to generalize across different devices was shown, additional k-fold cross-validation across all 8 capacitors at each of the three voltage levels should be performed to determine robustness.

Secondly, this study only utilized the PCoE dataset from NASA, therefore it is not possible to confirm generalizability across capacitors that are of different types, made by different manufacturers, or are aged differently (i.e., through methods such as thermal cycling or application of ripple currents).

Again, the proposed approach has not been compared with deep learning methods such as the LSTM or transformer which may perform this type of complex degradation modeling differently to capture long term degradation patterns, especially in instances of large datasets.

Finally, this study examined only three discrete voltage stress combinations (i.e. 10V, 12V, 14V) for the proposed method; further studies should examine additional intermediate stress levels along with continuous voltage profiling to provide more representative real world operating conditions.

Declaration of Ethical Standards

As the authors of this study, we declare that this study complies with all ethical standards.

Credit Authorship Contribution Statement

Kofi Addo Annan: Conceptualization, Validation, Formal Analysis, Visualization, Methodology, Writing and Software implementation.

E. A. Frimpong: Conceptualization, Editing and Supervision

F. B. Effah: Conceptualization, Editing and Supervision

E. Twumasi: Supervised to ensure technical accuracy of the results

A.-M. I. Malori: Review, Writing and Editing

M. A. Otu: Software implementation and Editing

Declaration of Competing Interest

The authors declare that they have no conflict of interest.

Funding / Acknowledgements

The authors declare that they did not receive any funding for this research.

Data Availability

The data used are publicly available datasets obtained from published literature. Data for this research can be accessed from NASA's Prognostics Center of Excellence Dataset Repository. https://www.nasa.gov/intelligent-systems-division/discovery-and-systems-health/pcoe/pcoe-data-set-repository/

References

Asigri, P., Frimpong, E. A., Anto, E. K., Kwegyir, D., & Effah, F. B. (2024). Enhanced Multi-Objective Grey Wolf Optimization using Adaptive Diversity Tuning and Levy Flights. *Carpathian Journal of Electrical Engineering*, 7–35. https://doi.org/10.34302/cjee/tkyd3692

Birbil, Ş. İ., & Fang, S.-C. (2003). An Electromagnetism-like Mechanism for Global Optimization. *Journal of Global Optimization*, *25*(3), 263–282. https://doi.org/10.1023/A:1022452626305

Dai, X., Long, Z., & Zhang, J. (2015). PSO based on chaotic map and its application to PID controller self-tuning. *2015 16th International Conference on Electronic Packaging Technology (ICEPT)*, 1470–1476. https://doi.org/10.1109/ICEPT.2015.7236860

TDK Electronics Co., L. (2026). *Electrolytic Capacitors (Part 7, Vol. 2): Key Features - TDK | Electronics ABC | Learn about Technology with TDK*. https://www.tdk.com/en/tech-mag/electronics_primer/10

Forouzandeh Shahraki, Ameneh, Al-Dahidi, Sameer, Rahim Taleqani, Ali, & Yadav, Om Prakash. (2023). Using LSTM neural network to predict remaining useful life of electrolytic capacitors in dynamic operating conditions. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, *237*(1), 16–28. https://doi.org/10.1177/1748006X221087503

Jha, B., & Dong, L. (2024). Lifetime Improvement With Predictive Maintenance of Power Electronics Based on Remaining Useful Life Prediction. *2024 IEEE Texas Power and Energy Conference, TPEC 2024*. https://doi.org/10.1109/TPEC60005.2024.10472254

Khorasgani, H., Kulkarni, C., Biswas, G., Celaya, J. R., & Goebel, K. (2013). *Degredation Modeling and Remaining Useful Life Prediction of Electrolytic Capacitors under Thermal Overstress Condition Using Particle Filters*. https://doi.org/https://doi.org/10.36001/phmconf.2013.v5i1.2277

Kulevome, D. K. B., Wang, H., & Wang, X. (2021). A Bidirectional LSTM-Based Prognostication of Electrolytic Capacitor. In *Progress In Electromagnetics Research C* (Vol. 109). https://doi.org/http://dx.doi.org/10.2528/PIERC20120201

Kullampalayam Murugaiyan, N., Chandrasekaran, K., Manoharan, P., & Derebew, B. (2024). Leveraging opposition-based learning for solar photovoltaic model parameter estimation with exponential distribution optimization algorithm. *Scientific Reports*, *14*(1). https://doi.org/10.1038/s41598-023-50890-y

Lin, J.-L., Wu, C.-H., & Chung, H.-Y. (2012). Performance Comparison of Electromagnetism-Like Algorithms for Global Optimization. *Applied Mathematics*, *03*(10), 1265–1275. https://doi.org/10.4236/am.2012.330183

Long, W., Jiao, J., Liang, X., Cai, S., & Xu, M. (2019). A Random Opposition-Based Learning Grey Wolf Optimizer. *IEEE Access*, *7*, 113810–113825. https://doi.org/10.1109/ACCESS.2019.2934994

Oliva, D., Cuevas, E., Pajares, G., & Zaldivar, D. (2014). Template matching using an improved electromagnetism-like algorithm. *Applied Intelligence*, *41*(3), 791–807. https://doi.org/10.1007/s10489-014-0552-y

Ragb, O., Bakr, H., & Civalek, O. (2023). Parameters identification for photovoltaic system via improved electromagnetism-like approach and quadrature technique. *International Journal of Energy and Environmental Engineering*, *14*(3), 353–377. https://doi.org/10.1007/s40095-022-00523-3

Rahnamayan, S., Tizhoosh, H. R., & Salama, M. M. A. (2006). Opposition-Based Differential Evolution for Optimization of Noisy Problems. *2006 IEEE International Conference on Evolutionary Computation*, 1865–1872. https://doi.org/10.1109/CEC.2006.1688534

Renwick, J., Kulkarni, C. S., & Celaya, J. R. (2015). Analysis of electrolytic capacitor degradation under electrical overstress for prognostic studies. *Vol. 7 No. 1 (2015): Proceedings of the Annual Conference of the PHM Society 2015*, *6*.

https://doi.org/https://doi.org/10.36001/phmconf.2015.v7i1.2713

Rigamonti, M., Baraldi, P., Zio, E., Astigarraga, D., & Galarza, A. (2016). Particle Filter-Based Prognostics for an Electrolytic Capacitor Working in Variable Operating Conditions. *IEEE Transactions on Power Electronics*, *31*(2), 1567–1575. https://doi.org/10.1109/TPEL.2015.2418198

Roman, D., Saxena, S., Bruns, J., Valentin, R., Pecht, M., & Flynn, D. (2021). A Machine Learning Degradation Model for Electrochemical Capacitors Operated at High Temperature. *IEEE Access*, *9*, 25544–25553. https://doi.org/10.1109/ACCESS.2021.3057959

Tan, J. D., Dahari, M., Koh, S. P., Koay, Y. Y., & Abed, I. A. (2016). An improved electromagnetism-like algorithm for numerical optimization. *Theoretical Computer Science*, *641*, 75–84. https://doi.org/10.1016/j.tcs.2016.05.045

Wang, X., Jiang, B., Wu, S., Lu, N., & Ding, S. X. (2022). Multivariate Relevance Vector Regression Based Degradation Modeling and Remaining Useful Life Prediction. *IEEE Transactions on Industrial Electronics*, *69*(9), 9514–9523. https://doi.org/10.1109/TIE.2021.3114724

Wang, Z., Chen, Y., Ding, S., Liang, D., & He, H. (2022). A novel particle swarm optimization algorithm with Lévy flight and orthogonal learning. *Swarm and Evolutionary Computation*, *75*, 101207. https://doi.org/https://doi.org/10.1016/j.swevo.2022.101207

Yang, X.-S., & Deb, S. (2010). *Engineering Optimisation by Cuckoo Search*. https://doi.org/https://doi.org/10.48550/arXiv.1005.2908

Yu, F., Guan, J., Wu, H., Chen, Y., & Xia, X. (2024). Lens imaging opposition-based learning for differential evolution with cauchy perturbation. *Applied Soft Computing*, *152*, 111211. https://doi.org/https://doi.org/10.1016/j.asoc.2023.111211

Zhang, B., Liu, W., Cai, Y., Zhou, Z., Wang, L., Liao, Q., Fu, Z., & Cheng, Z. (2024). State of health prediction of lithium-ion batteries using particle swarm optimization with Levy flight and generalized opposition-based learning. *Journal of Energy Storage*, *84*, 110816. https://doi.org/https://doi.org/10.1016/j.est.2024.110816

Zhang, Y., Zhang, C., & Cheng, Z. (2022). Parameter identification based on chaotic map simulated annealing genetic algorithm for PMSWG. *Prog. Electromagn. Res. M*, *113*, 59–71. https://doi.org/doi:10.2528/PIERM22070101

Zhu, C., Zhang, Y., Wang, M., Deng, J., Cai, Y., Wei, W., & Guo, M. (2024). Optimization, validation and analyses of a hybrid PV-battery-diesel power system using enhanced electromagnetic field optimization algorithm and ε-constraint. *Energy Reports*, *11*, 5335–5349. https://doi.org/10.1016/j.egyr.2024.04.043