

Article

Multi-Objective Stochastic Paint Optimizer for Solving Dynamic Economic Emission Dispatch with Transmission Loss Prediction Using Random Forest Machine Learning Model

Arunachalam Sundaram *  and Nasser S. Alkhalidi 

Department of Electrical Engineering, Jubail Industrial College, Al Jubail 31961, Saudi Arabia; khaldi_na@rcjy.edu.sa

* Correspondence: mailto:arunachalam@gmail.com

Abstract: Dynamic economic emission dispatch problems are complex optimization tasks in power systems that aim to simultaneously minimize both fuel costs and pollutant emissions while satisfying various system constraints. Traditional methods often involve solving intricate nonlinear load flow equations or employing approximate loss formulas to account for transmission losses. These methods can be computationally expensive and may not accurately represent the actual transmission losses, affecting the overall optimization results. To address these limitations, this study proposes a novel approach that integrates transmission loss prediction into the dynamic economic emission dispatch (DEED) problem. A Random Forest machine learning model was offline-trained to predict transmission losses accurately, eliminating the need for repeated calculations during each iteration of the optimization process. This significantly reduced the computational burden of the algorithm and improved its efficiency. The proposed method utilizes a powerful multi-objective stochastic paint optimizer to solve the highly constrained and complex dynamic economic emission dispatch problem integrated with random forest-based loss prediction. A fuzzy membership-based approach was employed to determine the best compromise Pareto-optimal solution. The proposed algorithm integrated with loss prediction was validated on widely used five and ten-unit power systems with B-loss coefficients. The results obtained using the proposed algorithm were compared with seventeen algorithms available in the literature, demonstrating that the multi-objective stochastic paint optimizer (MOSPO) outperforms most existing algorithms. Notably, for the Institute of Electrical and Electronics Engineers (IEEE) thirty bus system, the proposed algorithm achieves yearly fuel cost savings of USD 37,339.5 and USD 3423.7 compared to the existing group search optimizer algorithm with multiple producers (GSOMP) and multi-objective multi-verse optimization (MOMVO) algorithms.

Keywords: air pollution; metaheuristics; pollution control; pareto optimal solutions; power generation economics; power generation dispatch; random forest



Citation: Sundaram, A.; Alkhalidi, N.S. Multi-Objective Stochastic Paint Optimizer for Solving Dynamic Economic Emission Dispatch with Transmission Loss Prediction Using Random Forest Machine Learning Model. *Energies* **2024**, *17*, 860. <https://doi.org/10.3390/en17040860>

Academic Editor: Claus Leth Bak

Received: 8 January 2024

Revised: 1 February 2024

Accepted: 9 February 2024

Published: 12 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Using coal in power stations to produce electricity is inefficient, resulting in greenhouse gases and other harmful gas emissions to the atmosphere [1]. For sustainable development and to meet the stringent regulatory requirements, power plants that produce power using coal or other fossil fuels are forced to reduce air pollution levels and minimize fuel costs. To achieve these requirements, a multi-objective dynamic economic emission dispatch (DEED) model was formulated in the literature to minimize the conflicting objectives of the fuel cost and emission levels. When compared to the combined economic emission dispatch (CEED) problem, a DEED problem is more complicated as it incorporates a time-varying load pattern along with ramp-up and ramp-down limits of the generator. Finding a solution to a DEED problem involves solving nonlinear and non-differentiable equations. Traditional mathematical models that rely on solving derivatives will fail to solve the

DEED problem when generator valve-point effects [2] are considered. Traditional methods such as gradient-based optimization, linear programming, and heuristic methods struggle with non-convex or large-scale DEED problems, where finding globally optimal solutions becomes challenging. DEED is a bi-dimensional problem with linear and nonlinear equality and inequality constraints. Finding a non-inferior solution for a specific time horizon in a DEED problem requires a contemporary and robust multi-objective algorithm.

1.1. Solving Multi-Objective DEED Using Single-Objective Algorithm

The particle swarm optimization (PSO) algorithm using the goal attainment method in [3], the evolutionary programming (EP) algorithm [4], and the pattern search (PS) method used in [5] rely on the individual's knowledge. Variants of the PSO in [6] named standard PSO (SPSO), PSO with avoidance of worst location (PSOAWL), and PSO with gradually increasing directed neighborhood (PSO GIDN), resulted in an improved information exchange between the particles of PSO. However, they considerably increased the complexity of the algorithm. Variants of harmony search, such as NPAHS in [7] and enhanced harmony search in [8], are capable of finding Pareto-optimal (PO) solutions to the DEED problem. The efficient and enhanced differential evolution algorithm in [9] incorporates several local parameters. However, tuning these local parameters is complicated. Normalized objective functions were used in [8,9], along with an efficient strategy for constraint handling to combine two objectives into a single-objective function. The real-coded genetic algorithm in [10] combines multiple conflicting objectives using penalty factors which vary from one test system to another. Hybrid algorithms have also been used to improve the capabilities of the existing algorithms to solve the DEED problem. The hybrid algorithms in [11,12] solve the complicated DEED problem considering periodic and seasonal loads. A single-objective COOT optimization algorithm solves the DEED problem by combining conflicting objectives using weights [13]. An improved mayfly algorithm was employed to solve the DEED problem integrated with renewable energy sources [14].

The main drawback of employing single-objective algorithms to solve a bi-objective DEED problem is that the conflicting objectives must be converted into an equivalent single-objective function using penalty factors or weights. Furthermore, when PO solutions are required, the single-objective algorithm must be run several times by changing the weights during each run. Using a single-objective algorithm to solve a bi-objective DEED problem introduces limitations in terms of Pareto optimality, objective prioritization, solution diversity, and robustness. To address these drawbacks, employing multi-objective optimization techniques specifically designed for bi-objective problems is essential to ensure a more comprehensive exploration of the solution space and to facilitate informed decision-making.

1.2. Solving Multi-Objective DEED Using Multi-Objective Algorithm

The main advantage of using multi-objective algorithms to solve DEED problems is that PO solutions are obtained in a single run. The real-coded genetic algorithm (RCGA) and non-dominated sorting genetic algorithm II (NSGAI) in [15], and its improved version in [16], have been used to obtain non-dominated solutions for a DEED problem in a single run for a ten-unit system. The Pareto dominance-incorporated group search optimizer algorithm with multiple producers (GSOMP) in [17] can successfully solve the DEED problem. A multi-objective version of the single-objective differential evolution algorithm in [18] and its enhanced version in [19] have employed a constraint handling management technique that requires many computations to solve the DEED problem. A multi-objective version of the differential evolution algorithm was hybridized with a simulated annealing algorithm (MODE-SAT) [20] and employed to solve five-unit and ten-unit systems. To reduce the computational burden in solving the DEED problem, a multi-objective proximal policy optimization (MOPPO) algorithm was used in [21] along with the Markov decision process, where most of the computations were transferred to offline. Furthermore, a trained artificial neural network (ANN) and an improved fitness function were incorporated in [22]

to solve the DEED problem. The drawbacks of the methods proposed in [21,22] are that they require many hyperparameters to be tuned, and the PO solutions obtained are inferior to many existing algorithms. A multi-objective salp swarm algorithm was employed in [23] to obtain the PO solutions for 6-unit, 10-unit, and 14-unit test systems. The best compromise solution was obtained using the fuzzy membership method. Many recent works have also integrated renewable energy sources with DEED [24–26].

1.3. Modeling Transmission Loss in DEED Formulation

The two most widely used methods of transmission loss modeling are to use a single B-loss coefficient for the 24 h in a day, as in [3–9,11,12,15,17,21,22]; or to solve load flow equations, as in [16,18–20,27]. Both these models have a significant drawback when applied to the DEED formulation. As load varies from hour to hour, using a single B-loss coefficient for 24 h will result in an inaccurate transmission loss calculation. Solving the complex power flow equations during each iteration to find the power loss dramatically increases the computational efforts, increasing the convergence time. Various other approaches, such as modeling using the quadratic equation in [28], loss prediction using an ANN in [29], and other different methods are investigated in detail in [30]. This research tries to integrate a machine learning-based algorithm to predict the transmission loss without increasing the computational complexity. The literature survey indicates that the random forest algorithm has not been used for transmission loss prediction so far; hence, this research attempts to integrate transmission loss prediction using random forest in the DEED formulation.

2. Contribution

Traditionally, solving DEED problems involved incorporating losses computed using intricate nonlinear load flow equations or by using approximate loss formulas, both of which could be computationally expensive and the loss formula might not precisely capture actual transmission losses. The authors have employed powerful algorithms to solve the various economic dispatch formulations including the DEED problem in [31–38]. The integration of the machine learning model into the DEED problem is not widely explored in the literature. The authors were the first to attempt the integration of ANN into the DEED formulation using multi-objective multi-verse optimization (MOMVO) [29].

Random forest is a powerful machine learning algorithm that has been effective for a variety of prediction tasks. Thus, the innovation in this research lies in the integration of a random forest machine learning model to predict transmission losses and its integration in the dynamic economic emission dispatch (DEED) formulation. While DEED problems are not new and have been tackled using various optimization methods, the introduction of machine learning to accurately predict transmission losses is a novel approach. This is the first attempt to integrate random forest into the DEED formulation that the authors are aware of.

The use of a random forest machine learning model, offline-trained to predict these losses, is a unique and innovative solution. In doing so, the method eliminates the need for repeated calculations of transmission losses during each iteration of the optimization process, significantly reducing the computational burden and improving efficiency. As the transmission loss is predicted only once by the random forest model during a period of dispatch, the proposed method reduces the algorithm's complexity when compared to employing B-loss coefficients or power flow equations. This integration significantly reduces the computational burden of the DEED problem and improves its efficiency. The novelty of the proposed approach lies in integrating the random forest-based transmission loss prediction into the DEED formulation.

Further, the authors were motivated by the fact provided in [39] that there is always room to create potent algorithms to tackle challenging problems. This fact inspired the authors to utilize a newly developed MOSPO algorithm integrated with random forest to find the Pareto-optimal solution to a very tough and highly challenging DEED problem. The integration of a powerful multi-objective stochastic paint optimizer and a fuzzy

membership-based approach to determine the best compromise Pareto-optimal solution further adds to the innovation. These elements collectively contribute to a more effective and efficient solution to the DEED problem compared to existing algorithms in the literature.

3. Dynamic Economic Emission Dispatch Integrating Transmission Loss Prediction

In the dynamic economic emission dispatch model, the comparison of both objective functions holds significant importance in understanding the dual considerations of economic efficiency and environmental impact. The economic objective focuses on minimizing the operational costs associated with power generation, such as fuel expenses and maintenance costs. On the other hand, the emission objective aims to reduce the ecological footprint by minimizing pollutant emissions during power generation. While each objective serves a distinct purpose, the synergy between economic and emission considerations becomes crucial for achieving a balanced and sustainable solution. Combining both objectives allows decision-makers to identify a compromise that optimizes economic efficiency while simultaneously adhering to environmental sustainability goals. The joint optimization ensures that the power system operates with minimal cost and reduced emissions, contributing to a more resilient and environmentally conscious energy infrastructure. Therefore, the significance lies not only in evaluating each objective independently, but also in recognizing the intricate balance achieved when both economic and environmental factors are harmoniously integrated into the decision-making process.

3.1. Mathematical Model

DEED is a bi-objective optimization problem that tries to minimize the conflicting fuel cost objective function $f_1(x)$ given in (1) and emission level function $f_2(x)$ given in (2) during the time horizon T .

$$f_1(x) = \sum_{t=1}^T \sum_{i=1}^{Np} a_i P_{i,t}^2 + b_i P_{i,t} + c_i + \left| d_i \sin \left\{ e_i \left(P_i^{min} - P_{i,t} \right) \right\} \right| \quad (1)$$

$$f_2(x) = \sum_{t=1}^T \sum_{i=1}^{Np} \left[k_i P_{i,t}^2 + l_i P_{i,t} + m_i + n_i \exp(q_i P_{i,t}) \right] \quad (2)$$

where x is a decision vector and the elements contain the schedules of the Np thermal power generators given in (3):

$$x = [P_1, P_2, P_3, \dots, P_{Np}]^T \quad (3)$$

For each generator in the objective function (1), the quadratic coefficient of the real power is given by a_i , the linear coefficient is given by b_i , and c_i is a constant. The coefficients d_i and e_i are used in (1) to model the valve point effect. Including the valve point effect in (1) makes it nonlinear and non-convex. The coefficients k_i, l_i, m_i, n_i , and q_i are used to model the pollution created by sulfur and nitrogen oxide emissions (2) [28].

During every hour t of the dispatch period, the conflicting objectives are minimized to find the non-inferior solutions subject to equality and inequality constraints given in (4), (6), and (7).

The constraint given in (4) ensures that at each hour t , the real power produced by each thermal unit given by $P_{i,t}$ must exactly match with the total demand Pd_t plus the total transmission loss at time t given by Pl_t . The transmission loss can be precisely computed using complex load flow equations. To reduce the computation burden, the approximate Kron's formula given by (5) is used. The elements of the B-loss matrix of dimension $Np \times Np$ are represented using B with subscripts in (5). In this research work, a random forest-based loss prediction model replaces (5) to improve the loss modeling.

$$\sum_{i=1}^{Np} P_{i,t} - Pd_t - Pl_t(x) = 0 \quad (t = 1, 2, \dots, T) \quad (4)$$

$$Pl_t(x) = \sum_{i=1}^{Np} \sum_{j=1}^{Np} P_{i,t} B_{ij} P_{j,t} + \sum_{i=1}^{Np} B_{0i} P_{i,t} + B_{00} \quad (t = 1, 2, \dots, T) \quad (5)$$

The ramp rate limits of the thermal units for the dispatch interval Δt are given by (6). UR_i is the ramp-up limit of the i th generator and DR_i is the ramp-down limit of the i th generator.

$$\begin{cases} P_{i,t} - P_{i,t-1} \leq UR_i \Delta t \\ P_{i,t-1} - P_{i,t} \leq DR_i \Delta t \end{cases} \quad (6)$$

Each decision variable should lie between the lower bound P_i^{min} and the upper bound P_i^{max} . This constraint is given by (7):

$$P_i^{min} \leq P_{i,t} \leq P_i^{max}; i = (1, 2, \dots, Np), (t = 1, 2, \dots, T) \quad (7)$$

3.2. Constraint Handling Mechanism

At each hour of the dispatch period and during each iteration of the multi-objective algorithm, we must ensure that the schedule of the thermal generators lies within their bounds. To achieve this, we need an efficient strategy to ensure the constraints are always satisfied. Incorporating transmission losses and ramp-up and ramp-down constraints will make it even more difficult to ensure they are satisfied during the entire time horizon. The efficient constraint-handling strategy which randomly distributes the real power transmission loss to increase the quality and diversity of the solution in [29] was employed in this research work.

3.3. Fuzzy Decision-Making Approach to Select the Best Compromise Solution

Applying MOSPO to solve the DEED problem will result in multiple non-inferior solutions, and we need to provide the operator with one compromise solution. To enable this, the fuzzy decision-making approach available in [15–17,34] was employed in this research work and the steps involved are shown below.

Step 1: Determine the minimum and maximum value of each objective. Let $f_{m,min}$ and $f_{m,max}$ denote the minimum and maximum value of the m th objective.

Step 2: The membership function μ_{im} of the m th objective of the i th solution is determined using (8):

$$\mu_{im} = \begin{cases} 1 & \text{if } f_{im} \leq f_{m,min} \\ \frac{f_{m,max} - f_{im}}{f_{m,max} - f_{m,min}} & \text{if } f_{m,min} < f_{im} < f_{m,max} \\ 0 & \text{if } f_{im} \geq f_{m,max} \end{cases} \quad (8)$$

Step 3: The normalized membership function μ_i for each PO solution is calculated using (9):

$$\mu_i = \frac{\sum_{m=1}^M \mu_{im}}{\sum_{i=1}^I \sum_{m=1}^M \mu_{im}} \quad (9)$$

Step 4: The PO solution with the maximum value of μ_i is selected as the best compromise solution.

4. Multi-Objective Stochastic Paint Optimizer Algorithm

A stochastic paint optimizer (SPO) [40] algorithm to solve a single-objective problem is inspired by painting on a canvas. The idea of the SPO is to find the ideal color to be applied on the canvas (search space) by applying different color combinations using complementary, analogous, tetradic, and triadic mixing. The main advantage of SPO is that it has no internal parameters. SPO has been incorporated with an external archive, grid mechanism, and leader selection mechanism to solve the multi-objective problem. The multi-objective stochastic paint optimizer (MOSPO) [41] can solve multi-objective problems. It has excellent exploration and exploitation characteristics. The archive-handling process

and leader selection of the MOSPO algorithm are similar to the widely used multi-objective particle swarm optimization algorithm proposed in [42]. The archive-handling process involves the initialization of the archive, dominance check, archive update, and archive size control. The leader selection process involves random initialization of leaders from the archive, diversity calculation, the selection of new leaders, and replacement.

The selection of a multi-objective stochastic paint optimizer (MOSPO) to address the dynamic economic emission dispatch (DEED) model was driven by its inherent capabilities to manage the stochastic and multi-objective complexities of the optimization problem. MOSPO excels in providing robust solutions under the dynamic and uncertain operating conditions commonly found in power systems. By simultaneously considering multiple conflicting objectives and efficiently exploring the solution space, MOSPO identifies a diverse set of Pareto-optimal solutions. Its adaptability to changing conditions, capacity to handle complex system constraints, and robust performance in navigating high-dimensional optimization landscapes make MOSPO a well-suited and effective optimizer for DEED, enabling a comprehensive exploration of economic and environmental trade-offs.

MOSPO is applied to obtain the PO solutions of the complicated and strenuous DEED problem in this research work. Section 6 provides the procedure for obtaining the PO solutions of the DEED problem using MOSPO. For each time t of the dispatch period, the MOSPO algorithm is carried out, incorporating the constraint handling procedure in [29].

5. Random Forest Machine Learning Model to Predict the Transmission Loss

Transmission loss prediction is a problem where the output is a continuous variable. Decision trees are machine learning techniques that are pretty simple and very powerful for regression problems with a continuous output. Decision trees segregate the decision space into several simpler regions until a specific result is obtained. The end nodes have continuous numbers in the case of regression problems. The decision tree algorithm follows a top-down approach, and at each step, a variable from a dataset that splits the decision spaces into regions is chosen. Different metrics are used for splitting nodes, and these metrics measure the impurities of the region. Random forest algorithms are used to overcome the problem of overfitting in decision trees, as it reduces the variance and bias error. The random forest algorithm is an ensemble learning technique that consists of multitudes of decision trees. The output of the random forest regressor is the mean of all the values the different decision trees predict. Random forest regressors are very popular in many fields, such as weather prediction, stock market prediction, and medical analysis. In this research work, random forest regressors were used to predict the transmission loss of a power system using the schedules of the generators as attributes. The MOSPO algorithm and random forest-based loss prediction were carried out using MATLAB 8 on an H.P. Pavilion laptop, 1.8 GHz, Intel i7 processor, 16 GB RAM with WINDOWS 10 operating system.

5.1. Dataset for an IEEE 30 Bus System

The dataset was generated by repeatedly running a lossless DEED dispatch. To repeatedly run the DEED, existing load profiles in [17,18] and randomly generated load profiles which varied from a 3 per unit base load to 5.75 per unit base load were used. For each hour of the dispatch, 100 Pareto-optimal solutions were generated. The schedules obtained using the lossless dispatch will exactly match with the load demand Pd_t at time t . These Pareto-optimal solutions were used as the schedule for the generators while solving the load flow analysis. The demand in each bus was proportionately changed to match the demand of the DEED Pd_t at time t . The load flow analysis was solved using the Newton-Raphson algorithm 12,000 times to generate the dataset, and for each run, the transmission loss was evaluated. For each run of the load flow analysis, the schedule of the generators and the corresponding transmission loss were stored in an Excel file. During load flow analysis, the voltage profile of the generators was 1.05 per unit. The effect of the capacitor banks and tap-changing transformer were neglected. The box plot shown

in Figure 1 captures the variation in the schedules of the generators and it also indicates that for each generator the schedules lied within their respective maximum and minimum limits. The training data are available in [29].

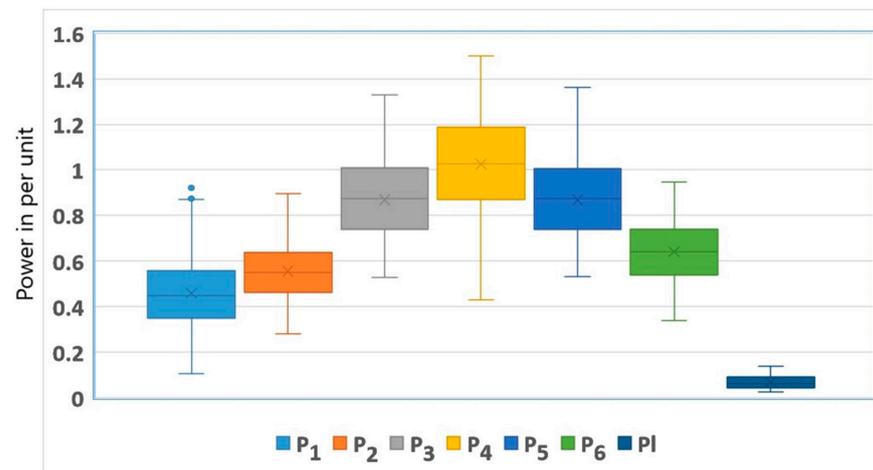


Figure 1. Box plot of the dataset used for the training and testing of random forest model.

Table 1 provides a sample of the dataset with seven attributes and 12,000 rows. The maximum and minimum value of the transmission loss P_l were 0.1363 p.u. and 0.0264 p.u., respectively.

Table 1. Sample data for training the random forest model to predict the transmission loss.

Row	P_1	P_2	P_3	P_4	P_5	P_6	P_l
100	0.257	0.399	0.636	0.904	0.586	0.466	0.032
200	0.397	0.494	0.759	0.874	0.764	0.609	0.048
300	0.309	0.434	0.690	0.894	0.688	0.482	0.037
400	0.298	0.400	0.591	0.640	0.559	0.509	0.027
500	0.270	0.400	0.653	0.916	0.643	0.466	0.034
...
8000	0.184	0.363	0.597	0.949	0.579	0.413	0.029
9000	0.329	0.484	0.792	1.015	0.846	0.580	0.059
10,000	0.549	0.613	0.836	0.795	0.850	0.712	0.064
11,000	0.592	0.673	1.134	1.346	1.136	0.781	0.112
12,000	0.389	0.487	0.850	1.087	0.863	0.554	0.056

5.2. Random Forest Machine Learning Model

A random forest is a supervised machine-learning model that consists of multiple decision trees [43]. Decision tree models are known to overfit, and hence to reduce the problem of overfitting, an ensemble of decision trees called random forests is used for prediction. The output of the random forest is the mean value of the output from all the individual learners or the decision trees. Random forests employs a technique called feature bagging, where only a subset of features is used during the growth of the individual decision trees. A random forest regression model is an ensemble learning technique that makes accurate predictions compared to any machine learning model [44].

The random forest model in MATLAB has been developed using the regression learner app, which enables automated training to choose the best ensemble model. The train-test split ratio was 70:30. In the dataset, a random split of 8600 data was used for training the random forest model and the remaining 3400 data were used for testing the model. An optimizable ensemble preset model in the application was used for model development, including the option for hyperparameter tuning using Bayesian optimization and five-fold cross-validation. Even though the K-fold cross-validation scheme needs multiple fits to

access the predictive accuracy, it is preferred as it uses the available data efficiently and works very well for a small dataset. Furthermore, Bayesian optimization techniques were preferred over the most commonly used grid search or random search method, as these techniques independently explore the whole range of possible parameter values without considering the previous outcomes. Furthermore, as the number of hyperparameters increases, the search time for the grid search and random search methods also increases exponentially. To overcome these drawbacks, Bayesian optimization techniques, which focus on promising areas of hyperparameter space, are employed, and this technique, based on the information encountered so far, predicts the next set of hyperparameters [45]. The list of hyperparameters and their values are shown in Table 2. The value of the hyperparameters during tuning using the Bayesian algorithm is shown in Table 3. The maximum objective function evaluations were set to 30. The best hyperparameter, which reduced the objective function value, is shown in bold. The mean squared error (MSE) obtained during tuning of hyperparameters is shown in Figure 2. The total time taken for hyperparameter tuning was 430 s.

Table 2. List of hyperparameters and their value employed during tuning using Bayesian optimization.

Hyperparameters	Values
<i>Ensemble Method</i>	<i>Bagging, LSBoost</i>
<i>Number of Learners</i>	10–500
<i>Learning Rate</i>	0.001–1
<i>Minimum Leaf size</i>	1–6000
<i>Number of predictors to sample</i>	1–6

Table 3. Values of the hyperparameters during each iteration of tuning.

Iteration	Ensemble Method	Number of Learners	Learning Rate	Minimum Leaf Size	Number of Predictors to Sample	Observed MSE
1	Bagging	38	-	2	1	1.4×10^{-6}
2	LSBoost	32	0.00105	51	4	1.4×10^{-6}
3	LSBoost	12	0.00486	2158	1	1.4×10^{-6}
4	LSBoost	13	0.99533	4	4	1.4×10^{-6}
5	Bagging	231	-	73	3	1.4×10^{-6}
6	LSBoost	118	0.0378	2	6	1.4×10^{-6}
7	Bagging	96	-	11	3	1.4×10^{-6}
8	LSBoost	373	0.3402	3233	5	1.4×10^{-6}
9	LSBoost	113	0.0627	12	4	1.4×10^{-6}
10	Bagging	174	-	2	4	1.2×10^{-6}
11	Bagging	40	-	3	4	1.2×10^{-6}
12	Bagging	35	-	2	1	1.2×10^{-6}
13	LSBoost	389	0.5535	595	5	1.2×10^{-6}
14	LSBoost	243	0.2233	2	6	7.9×10^{-7}
15	LSBoost	42	0.00352	21	5	7.9×10^{-7}
16	LSBoost	248	0.2223	1	5	7.9×10^{-7}
17	LSBoost	349	0.6527	35	3	7.9×10^{-7}
18	LSBoost	21	0.0186	3557	5	7.9×10^{-7}
19	LSBoost	123	0.37415	59	3	7.9×10^{-7}
20	LSBoost	299	0.0014	4	2	7.9×10^{-7}
21	LSBoost	23	0.95046	2031	1	7.9×10^{-7}
22	LSBoost	34	0.02568	33	2	7.9×10^{-7}
23	Bag	65	-	1	4	7.9×10^{-7}
24	LSBoost	24	0.0384	6	3	7.9×10^{-7}
25	LSBoost	55	0.001022	3	6	7.9×10^{-7}
26	LSBoost	36	0.001024	28	6	7.9×10^{-7}
27	LSBoost	14	0.21566	2	6	7.9×10^{-7}
28	LSBoost	10	0.9655	1	6	7.9×10^{-7}
29	LSBoost	11	0.06552	5203	6	7.9×10^{-7}
30	LSBoost	11	0.0189	63	6	7.9×10^{-7}

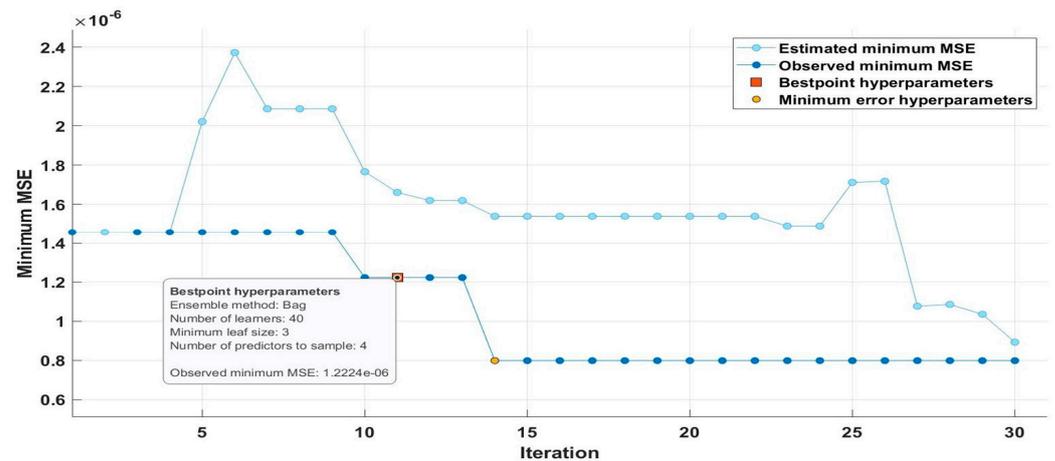


Figure 2. Mean squared error curve during tuning of hyperparameters.

The best model had a root mean square of 0.0011571, an R-squared value almost equal to one, and a mean squared error value of 1.22×10^{-6} , which was very close to the ANN model [29] whose mean squared error value was 1×10^{-8} . These statistical measures indicate that the random forest model could accurately predict transmission loss. The accuracy of the prediction of the trained model is indicated by the response plot shown in Figure 2.

A 45-degree line in Figure 3 indicates that the random forest ensemble was able to accurately predict the transmission loss. Furthermore, the variable plots shown in Figures 4–9 indicate that the error was significantly less between actual and predicted values.

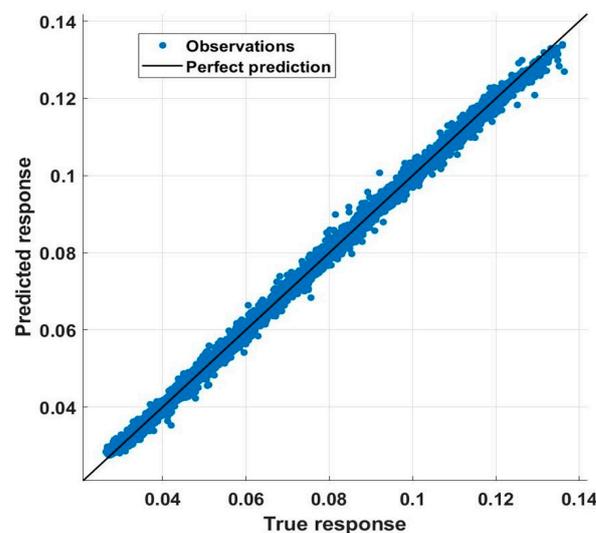


Figure 3. Response plot between the actual and predicted transmission loss.

The random forest regression model, meticulously developed and fine-tuned using MATLAB's regression learner app, emerged as a powerful tool for predicting transmission loss. Its ensemble nature, composed of multiple decision trees, effectively mitigates the risk of overfitting associated with individual tree models. Through the use of feature bagging and Bayesian optimization with five-fold cross-validation, the model demonstrated robustness and efficiency in training. The careful selection of hyperparameters detailed in Table 3 resulted in a mean squared error of 1.22×10^{-6} , showcasing the model's exceptional accuracy. Key statistical measures, including a root mean square of 0.0011571 and an R-squared value approaching unity, affirmed the precision and reliability of the model. Visual representations, such as the response plot in Figure 2 and variable plots in Figures 4–9, further underscore the proficiency of the random forest ensemble in accurately predicting

the transmission loss. The thorough evaluation from hyperparameter tuning to statistical metrics and visual representations collectively supported the assertion that the meticulously crafted and optimized random forest regression model stands as a robust and accurate tool for predicting the transmission loss, contributing both to predictive modeling advancements and practical insights into transmission loss dynamics in the studied dataset.

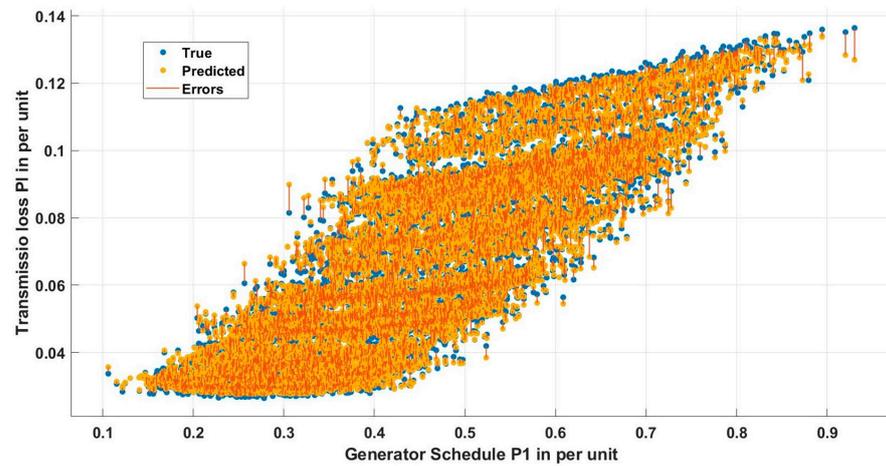


Figure 4. Variable plot between the schedule P1 and predicted transmission loss.

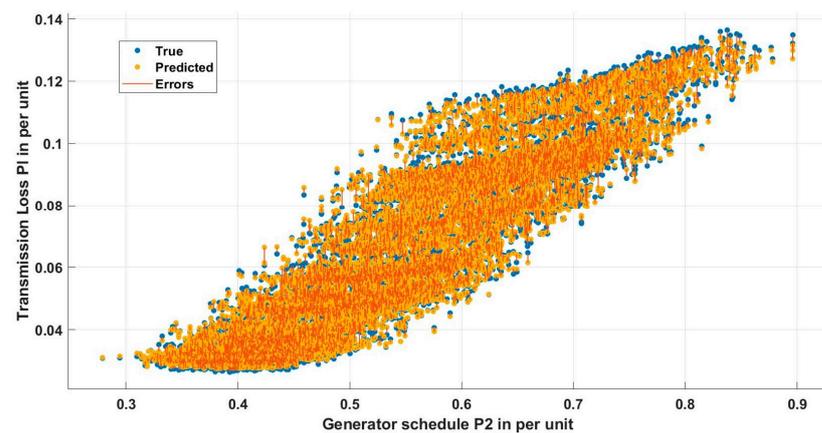


Figure 5. Variable plot between the schedule P2 and predicted transmission loss.

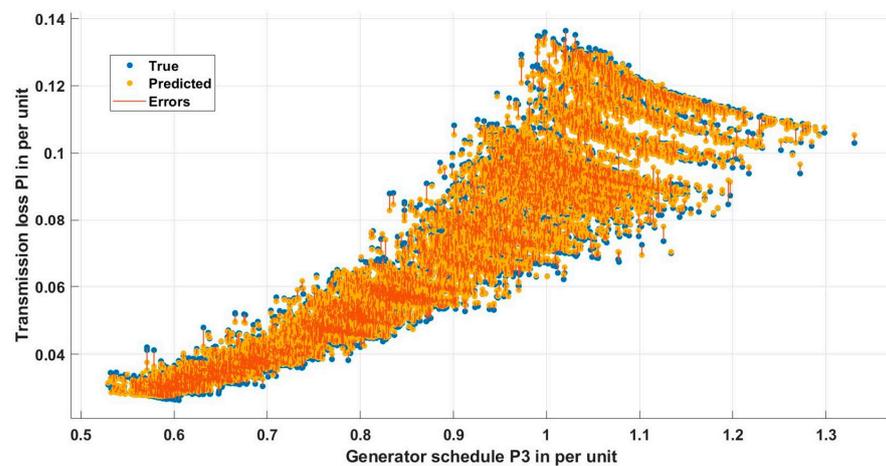


Figure 6. Variable plot between the schedule P3 and predicted transmission loss.

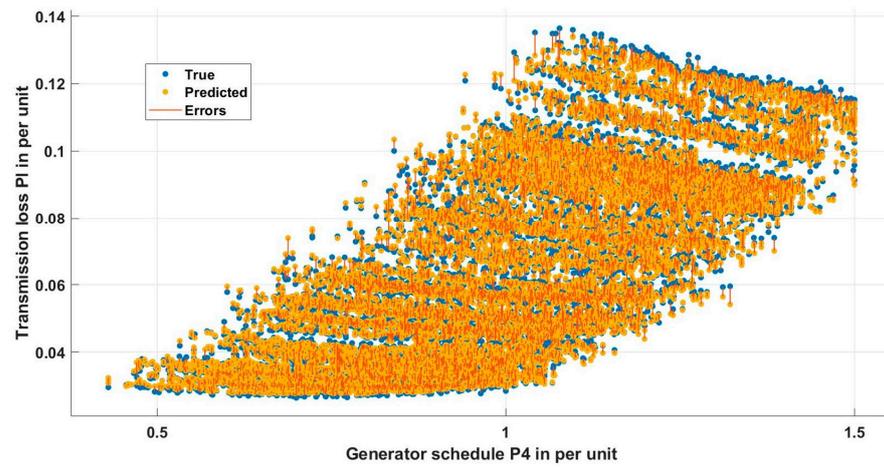


Figure 7. Variable plot between the schedule P4 and predicted transmission loss.

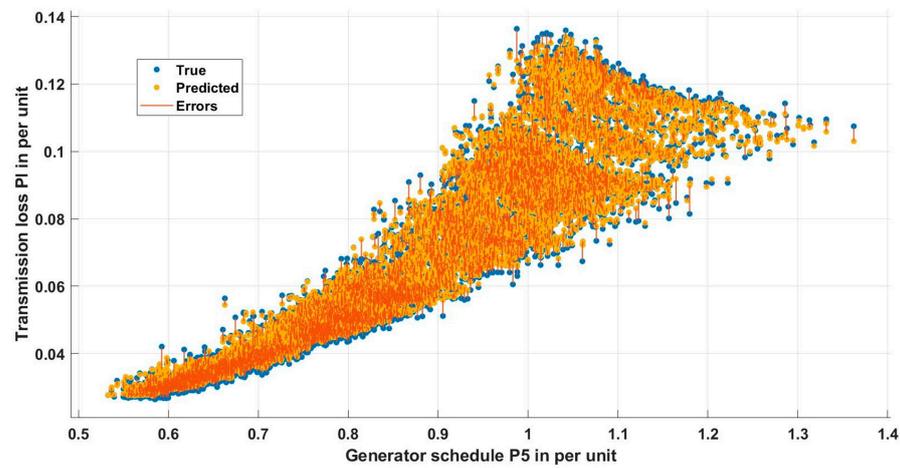


Figure 8. Variable plot between the schedule P5 and predicted transmission loss.

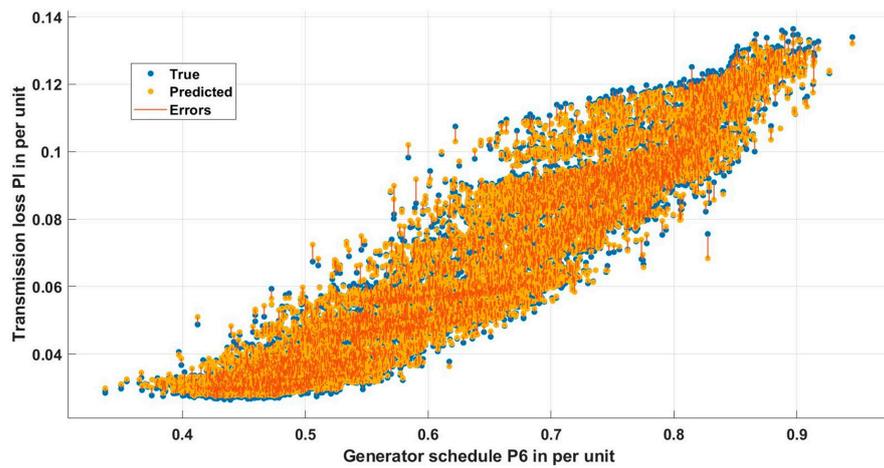


Figure 9. Variable plot between the schedule P6 and predicted transmission loss.

6. Proposed Algorithm for DEED Incorporating Random Forest Based Loss Prediction

The lossless dispatch DEED problem is easier to solve and converges rapidly compared to the DEED problem incorporated with real power transmission losses. Hence, the MOSPO algorithm was developed only to solve a lossless dispatch to satisfy the demand Pd_t at any moment t of the dispatch period until the algorithm reaches a quarter of the maximum iteration value. When the iteration counter is equal to a quarter of the maximum

iteration value, the MOSPO algorithm using the fuzzy decision approach [18–20,46] selects the best compromise solution from the list of available compromise solutions. The schedule of the generators available in the best compromise solution is used as an input to the trained random forest model to predict the transmission loss Pl_t . Until the iteration counter reaches the maximum value, the MOSPO algorithm will only run a lossless dispatch, with the demand increased to demand plus losses $Pd_t + Pl_t$. Once the MOSPO algorithm is completed, the best compromise solution is selected based on the fuzzy decision-making approach. The main advantage of the proposed algorithm lies in the fact that load flow equations are not solved to find the transmission loss, nor are B-loss coefficients used, but only an offline-trained random forest model predicts the transmission loss once during the algorithm. The algorithmic steps of the proposed algorithm using MOSPO are provided in Figure 10 with the novelty indicated in orange color. The main advantage of the MOSPO algorithm compared to widely used algorithms, such as MOPSO and hybrid NSGAI-MOPSO algorithms, is that it has the fewest parameters compared to any other algorithm. When the number of parameters increases, finding the ideal combination becomes cumbersome. MOSPO has only 6 parameters, MOPSO has 9, and the hybrid NSGAI-MOPSO algorithm has 14 parameters.

Algorithm to solve DEED Problem by using MOSPO Algorithm

Step 1: Specify the parameters for the DEED problem

- The duration of dispatch T and the dispatch interval Δt
- The total demand vector of the power system PD
- Initialize $Pdemand = PD(1)$
- Fuel cost and emission coefficients for each generating unit
- Number of decision variables $nVar$
- Lower bounds of the decision variables $VarMin$
- Upper bounds of the decision variables $VarMax$
- Current time is t

Step 2: Specify the parameters for MOSPO Algorithm

- Population number $Colors_Number = 100$
- External Archive Size $Archive_size = 100$
- Grid inflation parameter $alpha = 0.1$
- Number of grids for each dimension $nGrid = 30$
- Leader selection pressure parameter $beta = 4$
- Repository member selection pressure $gamma = 2$
- Maximum number of iterations $MaxIT$
- Current iteration is $Iter$

Step 3: Load the trained Random Forest Regressor Model

Step 4: While $t \leq T$ do

- $Pd_t = Pdemand$
- Calculate the bounds of the decision variables using
 - $P_{i,t}^{min} = \begin{cases} P_i^{min} & \text{if } t = 1 \\ \max(P_i^{min}, P_i^{t-1} - DR_i) & \text{if } t \neq 1 \end{cases}; i = 1 \text{ to } nVar$
 - $P_{i,t}^{max} = \begin{cases} P_i^{max} & \text{if } t = 1 \\ \min(P_i^{max}, P_i^{t-1} + UR_i) & \text{if } t \neq 1 \end{cases}; i = 1 \text{ to } nVar$
- Initialize population(colors) by generating a random population of size N
- Perform constraint handling mechanism for lossless dispatch
- Evaluate the fitness function for the given colors
- Sort the fitness function according to their colors
- Obtain the non-dominated solution and create the archive

While $Iter < MaxIt + 1$ do

- For each color do
 - Find leader

Figure 10. Cont.

- Divide colors into three groups as primary, secondary, and tertiary.
 - Create new colors using analogous combination, complementary combination, triadic combination, and tetradic combination
- **End For**
 - Evaluate the objective function for all colors
 - Obtain the non-dominated solutions
 - **Updating** the archive according to found non-dominated solutions
 - **If** the archive is full
 - Use the grid mechanism to delete the current archives
 - Add the new solution to the archive
 - **End If**
 - **If** any of the newly added solutions to the archive is located outside the hypercubes
 - Update the grids
 - **End If**
 - Plot the non-dominated solution in the archive
 - **If** $Iter = \text{round}\left(\frac{MaxIt}{4}\right)$ **then**
 - Find best compromise solution using fuzzy membership approach
 - **Using the best compromise solution predict the total transmission loss Pl_t using Random Forest Regressor**
 - Update $Pdemand = Pd_t + Pl_t$
 - Re-initialize the population. Generate a random N size population
 - Apply constraint handling mechanism for lossless dispatch
 - Calculate the fuel and emission objective functions
 - **End If**
 - Increment $Time$
- End While**
- Plot the pareto optimal front
 - Find best compromise solution using fuzzy membership approach.
 - Increment time t by Δt
 - $Pdemand = PD(t)$
- End While**

Figure 10. Proposed algorithm to solve the dynamic economic emission dispatch.

The proposed algorithm for dynamic economic emission dispatch (DEED) embodies a strategic approach, leveraging the multi-objective stochastic paint optimizer (MOSPO) framework and integrating a random forest-based loss prediction to efficiently address real power transmission complexities. Initially employing lossless dispatch for rapid convergence, MOSPO dynamically transitions to consider transmission losses using a trained random forest model. The algorithm's distinctive feature lies in its ability to bypass load flow equations and B-loss coefficients, relying solely on the predictive capabilities of the random forest model. This streamlined approach not only enhances computational efficiency but also minimizes the need for iterative calculations. Illustrated in Figure 10, the algorithmic steps showcase the novelty in orange, emphasizing the unique contributions. Critically, MOSPO's simplicity shines through with a minimal parameter set—only six parameters—setting it apart from more complex algorithms such as MOPSO and hybrid NSGAI-MOPSO. This simplicity expedites the search for optimal parameter combinations and positions MOSPO as an efficient and effective tool for solving the DEED problem with real-world power system constraints.

7. Case Studies

In this section, the efficacy of the MOSPO algorithm in solving the DEED problem with B-loss coefficients and using the proposed random forest regressor model is carried out. First, the effectiveness of the MOSPO and its competitive performance is established using 5 and 10 thermal units with valve point effects and a B-loss coefficient. Furthermore, MOSPO is tested with an IEEE 30 bus test system consisting of six thermal units using the proposed random forest-based loss prediction algorithm. The chosen case studies, involving 5 and 10 thermal units with valve point effects, and the IEEE 30 bus test system with 6 thermal units, are widely utilized in the literature. This selection allows for an easy comparison of results with existing techniques, ensuring a comprehensive evaluation of the MOSPO algorithm's performance in addressing the DEED problem with B-loss coefficients.

Unless specified in the case study, the parameters of the MOSPO algorithm were set as shown in Figure 10 and this was based on the recommendation in [41] demonstrating its effectiveness in complex benchmark functions and also on the complexity of the test system adopted. To ensure fairness, the parameters of the competing algorithms were obtained from their respective research work and are summarized in Table 4 along with the MOSPO algorithm. As seen from Table 4, the population of MOSPO was very reasonable and the iteration count was the least when compared to any other algorithm.

Table 4. Parameters of the competing algorithms along with MOSPO.

Algorithm	Parameters				
PSO [3]	Iterations = 300				
EP [4]	Iterations = 100	Population = 50			
SPSO [6]	Iterations = 3000	Acceleration coefficients C1 and C2 = 2.05 Constriction factor = 0.72984			
PSOAWL [6]	Iterations = 3000	Acceleration coefficients C1 and C2 = 1.845 Acceleration coefficients C3 and C4 = 0.205	Constriction factor = 0.72984	Neighborhood expansion speed $\gamma = 2$	Initial number of connections between particles $b = 3$.
NPAHS [7]	Iterations = 50,000	Harmony memory size = 5	Harmony memory consideration rate = 0.99	Pitch adjustment rate = 0.01	Width = 0.001
NEHS [8]	Iterations = 50,000 for 5 unit Iterations = 100,000 for 10 unit	Harmony memory size = 10	Harmony memory consideration rate close to 1	Pitch adjustment rate = 0.3	Band width = 0.05, 0.0125, and 0.003125
EFDE [9]	Iterations = 5000 for 5 unit Iterations = 20,000 for 10 unit	Population = 20	Adaptive mutation factor Adaptive scaling and crossover probability		
DE-SQP PSO-SQP [12]	Iterations = 20,000 Population = 60	Mutation factor = 0.423	Crossover probability = 0.885	Acceleration coefficients C1 and C2 = 2.25	Adaptive inertial and constriction factor
NSGAI [15]	Iterations = 100	Population = 20	Crossover probability = 0.9	Mutation probability = 0.2	
MODE [20]	Iterations = 500,000	Population = 100	Adaptive scaling factor between 0.3 and 0.9	Adaptive crossover rate between 0.1 and 0.9	
MOPPO [21]	Episode number = 1000	Epoch number = 10	Function coefficient = 0.5	Entropy coefficient = 0.01	Exploration standard deviation = 0.5
MOMVO [29]	Max Time = 400	Population = 100 Archive Size = 100	Exploitation accuracy = 6	Wormhole existence probability between 0.2 and 1	
MOSPO	Iterations = 40 for 5-unit system, IEEE 30 bus system Iterations = 100 for 10-unit system		Population = 100	Archive size = 100	Alpha = 0.1 Beta = 4 Gamma = 2

7.1. Case 1—Five-Unit Test System

The five-unit test system considered in [4] was employed in this case study. A dynamic economic dispatch was considered for a day with a dispatch interval of 1 h. The transmission loss was modeled using the B-loss coefficients that are available in [4]. The up-ramp limits for generators 1 to 5 were 30 MW/h, 30 MW/h, 40 MW/h, 50 MW/h, and 50 MW/h respectively. The down-ramp limits for generators 1 to 5 were 30 MW/h, 30 MW/h, 40 MW/h, 50 MW/h, and 50 MW/h respectively. The proposed algorithm was run 20 times. The time elapsed for each run, the total emission, and the total fuel cost obtained for each run are summarized in Table 5. As seen from Table 5, the average time required by the proposed algorithm to find the PO solutions for a 24 h dispatch with 1 h intervals was 933.914 s. This time includes the convergence time of the algorithm along with time for plotting the PO solutions for each hour, and the printing of results.

Table 5. Summary of the results obtained for five-unit test system.

Run	Time	Emission (lbs)	Fuel Cost (\$)
1	1049.723	18,523.9	47,786.58
2	890.4258	18,572.14	47,384.56
3	1028.585	18,672.82	47,218.9
4	1006.217	18,552.32	47,238.85
5	945.2438	18,865.26	46,298.08
6	924.1873	18,690.57	47,746.91
7	900.312	18,592.19	47,165.49
8	897.8754	18,794.08	46,871.81
9	916.1342	18,451.84	48,379.4
10	912.8549	18,857.35	46,570
11	941.0269	18,436.18	48,221.11
12	942.2903	18,599.01	47,217.86
13	932.938	18,466.96	48,203.98
14	931.0885	18,525.5	47,337.87
15	891.3465	18,772.47	47,183.07
16	884.0648	18,626.94	47,082.39
17	974.1892	18,537.08	47,498.67
18	898.1882	18,533.54	47,399.39
19	887.9814	18,607.79	47,248.88
20	923.6068	18,487.68	47,783.05
Max	1049.723	18,865.26	48,379.4
Min	884.0648	18,436.18	46,298.08
Average	933.914	18,608.28	47,391.84

The PO solutions obtained using the MOPSO algorithm for the first twenty runs are shown as red dots and the non-Pareto solutions are shown as black dots in Figure 11. The compromise solution obtained using the fuzzy membership method is shown as a blue dot in Figure 11. The generator schedules of the compromise solution are tabulated in Table 6 along with the demand, the total generation, and the transmission loss for each hour. The emission levels and fuel cost obtained for each hour are also tabulated. From the schedules obtained we can observe that the minimum and maximum generator limits for all generators were satisfied, along with the ramp limit constraints for the entire dispatch.

From Table 6, we can infer that the minimum demand of 410 MW occurred at 1 AM and the maximum demand of 740 MW occurred at 12 noon. Even though the demand pattern changed throughout the day, only one B-loss coefficient was used during the dispatch period. As the B-loss coefficients are determined for a specific load level, use of these coefficients will only result in an approximate loss calculation at all other load levels. The main idea of this research is to overcome this drawback and to replace this approximate B-loss formula with a transmission loss prediction. In this case study, the

MOSPO algorithm was used along with approximate B-loss formula only to ensure the fairness of the comparison with the existing algorithm

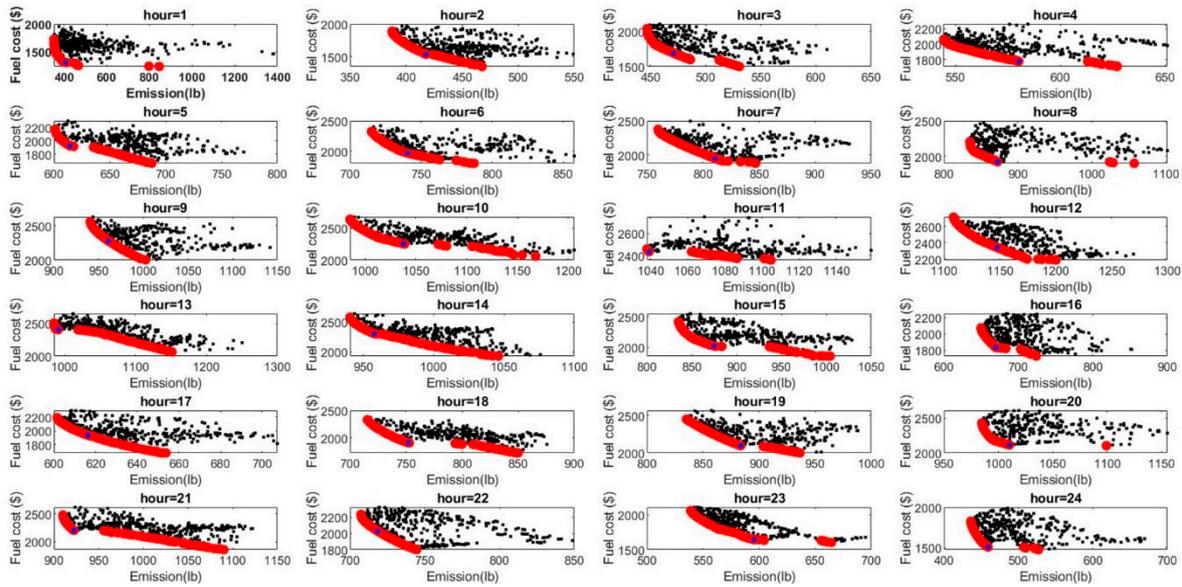


Figure 11. Non-Pareto-optimal solutions (black dot), PO Solutions (red dot), and compromise solution (blue dot) for each hour of the dispatch period for the five-unit system.

Table 6. The schedules of the generators obtained for the dynamic dispatch for the five-unit system.

t (h)	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	$\sum_{i=1}^5 P_i$	Pd_t (MW)	P_l (MW)	Emission (lbs)	Fuel Cost (\$)
1	22.07	98.39	116.35	126.77	50.00	413.58	410.00	3.58	407.62	1307.2
2	52.07	68.39	106.97	161.55	50.00	438.98	435.00	3.98	417.71	1535.1
3	75.00	87.00	146.97	111.55	59.11	479.63	475.00	4.63	470.85	1689.8
4	47.95	57.00	174.99	146.60	109.11	535.65	530.00	5.65	580.14	1778.4
5	74.99	86.98	134.99	180.19	87.23	564.38	558.00	6.38	614.37	1923.7
6	75.00	98.02	174.99	130.19	137.23	615.43	608.00	7.43	740.38	1970.9
7	75.00	85.58	134.99	165.15	173.19	633.91	626.00	7.91	810.86	1948.2
8	48.93	115.57	174.99	200.05	123.19	662.73	654.00	8.73	872.4	1926.2
9	75.00	88.42	164.73	209.06	162.42	699.63	690.00	9.63	961.05	2277.2
10	64.83	118.42	175.00	243.55	112.42	714.22	704.00	10.22	1037.2	2257.7
11	75.00	125.00	175.00	203.15	152.39	730.54	720.00	10.54	1039.2	2435.1
12	75.00	95.00	163.93	238.03	179.13	751.09	740.00	11.09	1147.5	2343.8
13	75.00	124.99	175.00	209.96	129.13	714.08	704.00	10.08	990.99	2407.1
14	75.00	100.46	175.00	180.31	168.80	699.57	690.00	9.57	957.73	2304.1
15	71.72	122.11	135.00	215.26	118.80	662.89	654.00	8.89	874.36	2024.9
16	75.00	92.11	174.98	165.26	79.53	586.87	580.00	6.87	669.66	1834.1
17	68.50	68.21	134.98	163.04	129.53	564.26	558.00	6.26	616.28	1936.5
18	65.24	98.21	174.98	197.65	79.53	615.61	608.00	7.61	751.75	1909.4
19	75.00	91.06	134.98	232.26	129.53	662.83	654.00	8.82	883.5	2093.3
20	66.38	121.06	174.98	182.26	169.36	714.04	704.00	10.04	1010.2	2114.2
21	75.00	102.51	175.00	217.54	119.36	689.41	680.00	9.41	922.46	2203
22	75.00	84.05	145.34	167.54	140.44	612.37	605.00	7.37	718.32	2029
23	75.00	59.85	105.34	202.17	90.44	532.80	527.00	5.80	595.5	1642.5
24	45.00	74.95	145.34	152.17	50.00	467.46	463.00	4.46	459.2	1507.2
Total									18,549.23	47,398.6

Figure 12 compares the total emission and total fuel cost of the dispatch obtained using the MOSPO algorithm with several existing algorithms. In Figure 12, the red color and bold indicate that the MOSPO algorithm could outperform the existing algorithms

in that specific objective. Out of the 14 algorithms compared, MOSPO could outperform five algorithms in both objectives, six algorithms in at least one objective, and it could not outperform three algorithms in any objective.

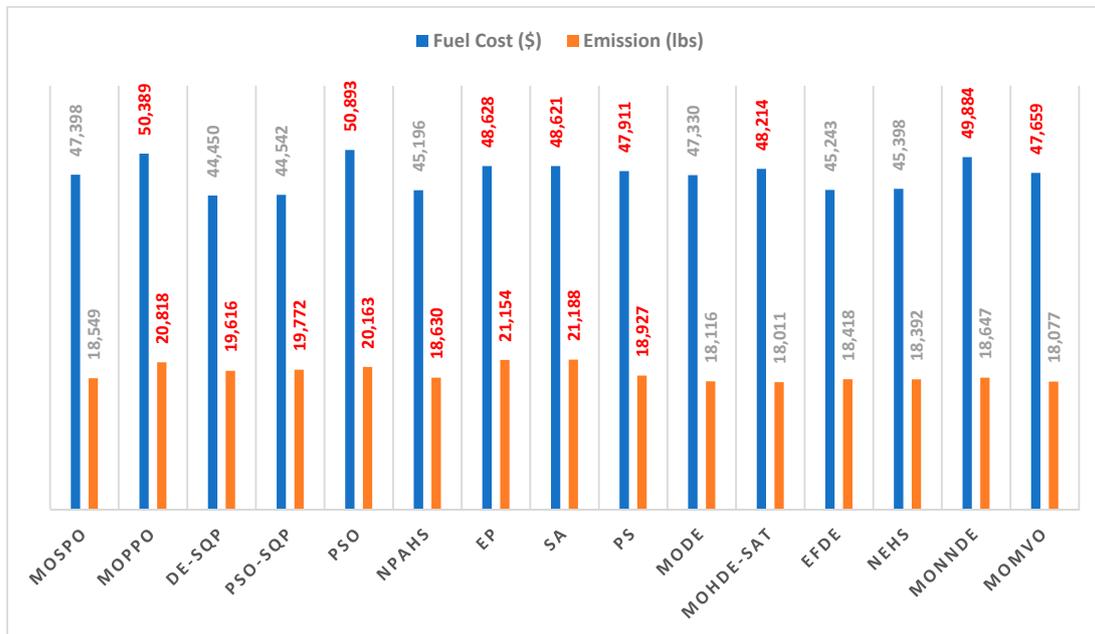


Figure 12. Comparison of the fuel costs and emissions obtained using the MOSPO algorithm with the MOPPO [21], DE-SQP [12], PSO-SQP [12], PSO [3], NPAHS [7], EP [4], SA [5], PS [5], MODE [20], MOHDE-SAT [20], EFDE [9], NEHS [8], MONNDE [22] and MOMVO [29] for the five-unit systems. The red color and bold indicate that the MOSPO algorithm could outperform the existing algorithms in that specific objective.

7.2. Case 2—Ten-Unit Test System

A widely used 10-unit test system was employed in this case study. The fuel cost data for the ten thermal generators, the load profile, and a single B-loss coefficient for the entire time horizon are available in [15], and the maximum iteration count for the MOSPSO algorithm was set to 100. The stacked bar plot of the generator schedules is shown in Figure 13. The schedule of the ten generators for each hour obtained using the MOSPO algorithm is tabulated in Table 7.

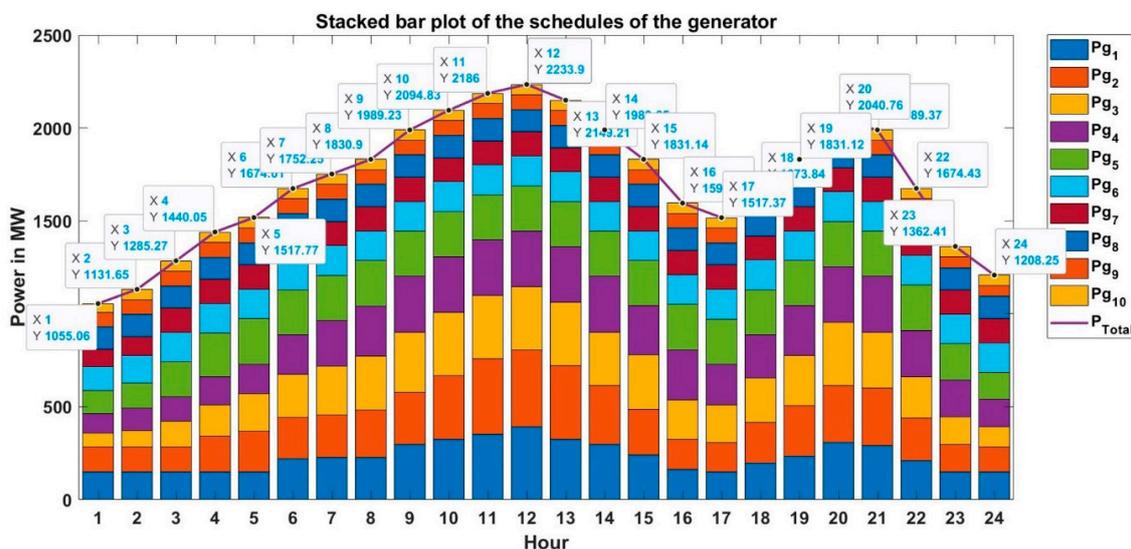


Figure 13. Stacked bar plot of the schedules for the generator obtained using MOSPO.

Table 7. The schedules of the generators obtained for the DEED for the ten-unit system.

t (h)	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	P_6 (MW)	P_7 (MW)	P_8 (MW)	P_9 (MW)	P_{10} (MW)	$\sum_{i=1}^{10} P_i$	Pd_t (MW)	P_l (MW)
1	150.00	135.00	73.61	103.96	124.22	129.81	92.53	120.00	80.00	45.93	1055.06	1036.00	19.06
2	150.00	135.00	86.67	119.75	135.46	147.10	102.68	120.00	80.00	55.00	1131.65	1110.00	21.65
3	150.00	135.00	138.54	131.34	185.46	160.00	130.00	120.00	79.93	55.00	1285.27	1258.00	27.27
4	150.00	192.57	168.70	148.32	235.46	160.00	130.00	120.00	80.00	55.00	1440.05	1406.00	34.05
5	150.19	217.50	201.50	160.59	243.00	160.00	130.00	120.00	80.00	55.00	1517.77	1480.00	37.77
6	218.88	224.81	231.74	210.59	243.00	160.00	129.99	120.00	80.00	55.00	1674.01	1628.00	46.01
7	226.13	230.30	261.42	246.40	243.00	160.00	130.00	120.00	80.00	55.00	1752.25	1702.00	50.25
8	225.95	256.09	291.47	269.39	243.00	160.00	130.00	120.00	80.00	55.00	1830.90	1776.00	54.90
9	296.83	280.69	323.70	300.00	243.00	160.00	130.00	120.00	80.00	55.00	1989.23	1924.00	65.23
10	325.68	341.14	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	2094.83	2022.00	72.83
11	350.57	407.43	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	2186.00	2106.00	80.00
12	390.77	415.12	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	2233.90	2150.00	83.90
13	325.28	395.93	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	2149.21	2072.00	77.21
14	296.97	315.93	288.76	300.00	243.00	160.00	130.00	120.00	80.00	55.00	1989.65	1924.00	65.65
15	242.11	244.03	291.89	265.11	243.00	160.00	130.00	120.00	80.00	55.00	1831.14	1776.00	55.14
16	162.11	164.03	211.89	269.39	243.00	160.00	130.00	120.00	80.00	55.00	1595.42	1554.00	41.42
17	150.00	156.65	203.33	219.39	243.00	160.00	130.00	120.00	80.00	55.00	1517.37	1480.00	37.37
18	196.19	219.10	237.51	233.03	243.00	160.00	130.00	120.00	80.00	55.00	1673.84	1628.00	45.84
19	233.04	273.82	268.06	268.20	243.00	160.00	130.00	120.00	80.00	55.00	1831.12	1776.00	55.12
20	306.37	306.39	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	2040.76	1972.00	68.76
21	291.75	307.42	302.21	300.00	243.00	160.00	130.00	120.00	80.00	55.00	1989.37	1924.00	65.37
22	211.75	227.42	222.21	250.00	243.00	160.00	130.00	120.00	55.05	55.00	1674.43	1628.00	46.43
23	150.00	147.42	147.60	200.00	193.00	160.00	130.00	120.00	59.39	55.00	1362.41	1332.00	30.41
24	150.00	135.00	106.45	150.00	143.00	160.00	129.45	120.00	59.36	55.00	1208.25	1184.00	24.25

Figure 14 compares the results obtained using the MOSPO algorithm with several existing algorithms. In Figure 14, the red color and bold indicate that the MOSPO algorithm could outperform the existing algorithms in that specific objective. The MOSPO algorithm outperformed the following nine state-of-the-art algorithms: MOPPO [21], SPSO [6], NSGA-II [15], IBFA [47], RCGA [15], MAMODE [19], CRO [11], HCRO [11], and MODE [20] in both objectives. The MOSPO algorithm can outperform the following existing algorithms MOMVO [29], PSOAWL [6], MOHDE-SAT [20], DE-SQP [12], PSO-SQP [12], EFDE [9], NEHS [8], and MONNDE [22] in at least one objective.

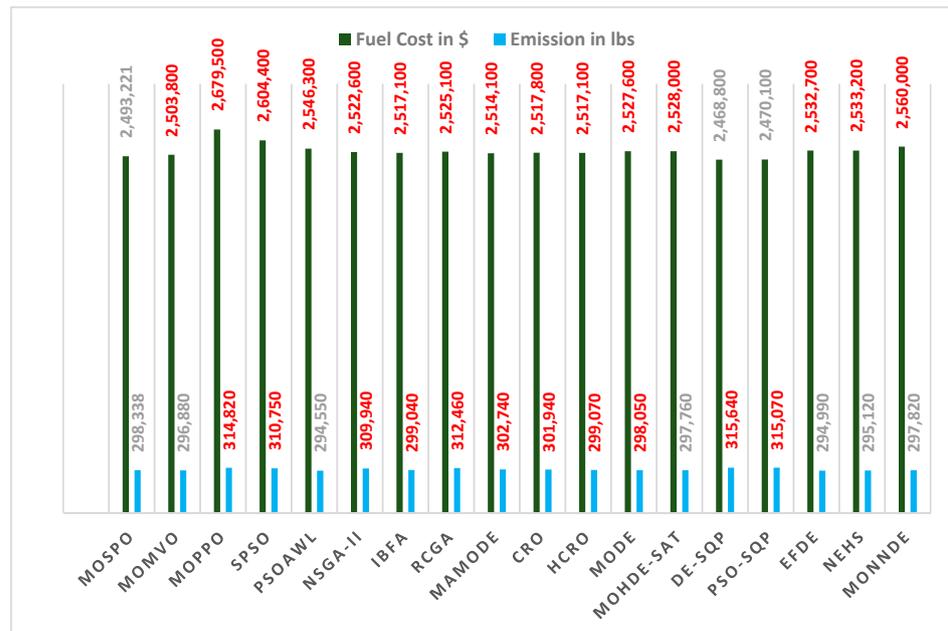


Figure 14. Comparison of the fuel costs and emissions obtained using the MOSPO algorithm with the existing algorithms for the ten-unit systems. The red color and bold indicate that the MOSPO algorithm could outperform the existing algorithms in that specific objective.

7.3. Case 3—IEEE 30 Bus Test System

An IEEE 30 bus power system with six thermal generators was used in this case study. The fuel cost and emission data are available in [17,18]. The ramp-up and ramp-down limits were set at 0.5 p.u. The offline-trained random forest regression model presented in Section 5 was used to predict the transmission loss during each hour of the dispatch period. A base value of 100 MW was used for the per-unit calculations. All existing algorithms except [29] use only one B-loss coefficient for the demand, which varies widely from a based load of 3 p.u. to the peak load of 5.75 p.u. Using one B-loss coefficient for the entire dispatch period will not provide an accurate transmission loss, but the trained random forest model will be able to predict the loss accurately.

The proposed algorithm integrated with loss prediction was run 20 times and the results obtained are summarized in Table 8. The load demand shown in column 2 of Table 9 was considered to obtain a comprehensive analysis. The average time the algorithm took to converge was 1038.847 s. This time included the convergence time of the algorithm along with loss prediction, time for plotting the PO solutions for each hour, and the printing of results. The results indicate the robustness and consistency of the proposed algorithm in obtaining reliable and feasible results. Comparing the maximum, minimum, and average values of the time, emission, and fuel cost indicated the high level of convergence obtained using the proposed algorithm in obtaining the PO solutions.

Table 8. Summary of the results obtained for IEEE 30 bus test system.

Run	Time	Emission (lbs)	Fuel Cost (\$)	Run	Time	Emission (lbs)	Fuel Cost (\$)
1	1053.325	5.994373	25,824.72	11	1025.34	5.958727	25,846.53
2	1017.908	5.950465	25,861.09	12	1138.781	6.002332	25,814.56
3	1012.752	5.965218	25,841.02	13	1102.976	5.972242	25,839.13
4	1018.667	5.974769	25,844.07	14	1037.276	6.010266	25,816.18
5	1013.71	5.955149	25,857.12	15	1040.52	6.009722	25,823.46
6	1014.724	5.965142	25,841.33	16	1034.59	5.985644	25,833.88
7	1015.476	5.963196	25,848.1	17	1035.014	5.975186	25,833.62
8	1014.505	5.986876	25,820.08	18	1038.629	5.994643	25,823.39
9	1020.026	5.978975	25,832.06	19	1059.743	5.978423	25,836.58
10	1017.071	5.98509	25,824.91	20	1065.911	5.948667	25,850.73
				Max	1138.781	6.010266	25,861.09
				Min	1012.752	5.948667	25,814.56
				Average	1038.847	5.977755	25,835.63

The results obtained for the 21st run of the proposed algorithm are shown in Table 9. The maximum iteration count for the MOSPO algorithm was set as 40. Until the iteration count reached 10 during every hour of the dispatch period, a lossless dispatch was carried out, and when the count reached 10 the compromise solution obtained using the MOSPO algorithm was used by the random forest model to predict the transmission loss. Using this transmission loss, the total demand at that hour Pd_t is increased to $Pd_t + P_l$. Starting from the iteration count 11 to 40, the MOSPO algorithm solves a lossless dispatch to satisfy the demand $Pd_t + P_l$. The fuzzy membership function is used at the end of the iteration to find the best compromise solution. For example, let us take hour 5. Until the iteration count reaches ten, the MOSPO algorithm solves a lossless dispatch to satisfy the demand $Pd_5 = 3.35$ p.u. When the count reaches 10, the random forest algorithm predicts the loss for the compromise solution as 0.0357 p.u. From iteration count 11 to 40, the MOSPO algorithm solves a lossless dispatch to satisfy a demand of 3.3857 p.u. A fuzzy membership approach is applied to find the compromise solution at the end of the iteration, and the schedules of the generators are shown in Table 8. The transmission loss obtained during each hour of the dispatch period was cross-verified using power flow analysis, and the loss obtained using power flow analysis exactly matches the loss predicted by the random

forest algorithm. The PO solutions (red dot) and compromise solution (blue dot) for each hour of the dispatch period for the IEEE 30 bus system are shown in Figure 15.

Table 9. The schedules of the generators obtained for the DEED using the MOSPO algorithm for the IEEE 30 bus test system.

t (h)	Pd_t (p.u.)	P_1 (p.u.)	P_2 (p.u.)	P_3 (p.u.)	P_4 (p.u.)	P_5 (p.u.)	P_6 (p.u.)	$\sum_{i=1}^6 P_i$	Emission (ton)	Fuel Cost (\$)	
1	3.25	0.0332	0.2584	0.4633	0.6321	0.8005	0.6685	0.4604	3.2832	0.204574	710.313
2	3.9	0.0508	0.3811	0.5459	0.7808	0.8694	0.8058	0.5678	3.9508	0.211471	874.0063
3	3.5	0.0382	0.3352	0.4276	0.6987	0.8198	0.7326	0.5243	3.5382	0.205828	771.4103
4	3	0.0299	0.2533	0.4219	0.5394	0.7462	0.6075	0.4616	3.0299	0.201993	653.1154
5	3.35	0.0357	0.3006	0.4553	0.6967	0.7761	0.6549	0.5021	3.3857	0.203341	736.4617
6	4	0.049	0.425	0.5022	0.8152	0.9131	0.8075	0.586	4.049	0.214282	896.9822
7	4.75	0.0742	0.4919	0.5836	0.9437	1.1061	0.9829	0.716	4.8242	0.242413	1094.734
8	5.05	0.0821	0.5132	0.604	1.0204	1.2682	1.0245	0.7018	5.1321	0.26429	1172.912
9	5.45	0.1072	0.5878	0.6588	1.1086	1.3227	1.0941	0.7852	5.5572	0.28898	1293.243
10	5.2	0.0908	0.6152	0.6861	1.0208	1.2238	1.0195	0.7254	5.2908	0.264559	1224.479
11	5.5	0.1067	0.622	0.6602	1.0992	1.3169	1.1	0.8084	5.6067	0.289811	1309.338
12	5.75	0.1142	0.5922	0.7005	1.1992	1.4232	1.1176	0.8315	5.8642	0.319458	1379.691
13	5.25	0.0939	0.5444	0.5955	1.1011	1.2695	1.0676	0.7658	5.3439	0.276959	1231.984
14	5.15	0.0931	0.5868	0.6452	1.0503	1.1898	1.0199	0.7511	5.2431	0.262543	1209.895
15	4.75	0.0746	0.5177	0.5604	0.9733	1.121	0.9695	0.6827	4.8246	0.243563	1093.792
16	5.3	0.0985	0.5455	0.6467	1.042	1.3024	1.0978	0.7641	5.3985	0.279045	1247.837
17	5.15	0.0891	0.5574	0.6923	1.027	1.2146	1.0548	0.693	5.2391	0.264878	1207.298
18	5.75	0.1153	0.5982	0.713	1.1976	1.3999	1.1086	0.848	5.8653	0.316635	1381.709
19	5.25	0.0996	0.5382	0.665	1.0782	1.2492	1.0192	0.7998	5.3496	0.271746	1237.494
20	5.25	0.0968	0.5777	0.6798	1.0529	1.1919	1.0612	0.7833	5.3468	0.268065	1239.555
21	4.55	0.0677	0.4272	0.556	0.9101	1.1051	0.9212	0.6981	4.6177	0.236224	1038.153
22	4.25	0.058	0.4458	0.5249	0.9281	0.9673	0.836	0.6059	4.308	0.222762	961.2564
23	4.25	0.055	0.4314	0.5151	0.8731	0.9723	0.8996	0.6135	4.305	0.223021	959.5195
24	4	0.0503	0.4301	0.5327	0.763	0.9523	0.7995	0.5727	4.0503	0.214938	896.9213
Total									5.991378	25,822.1	

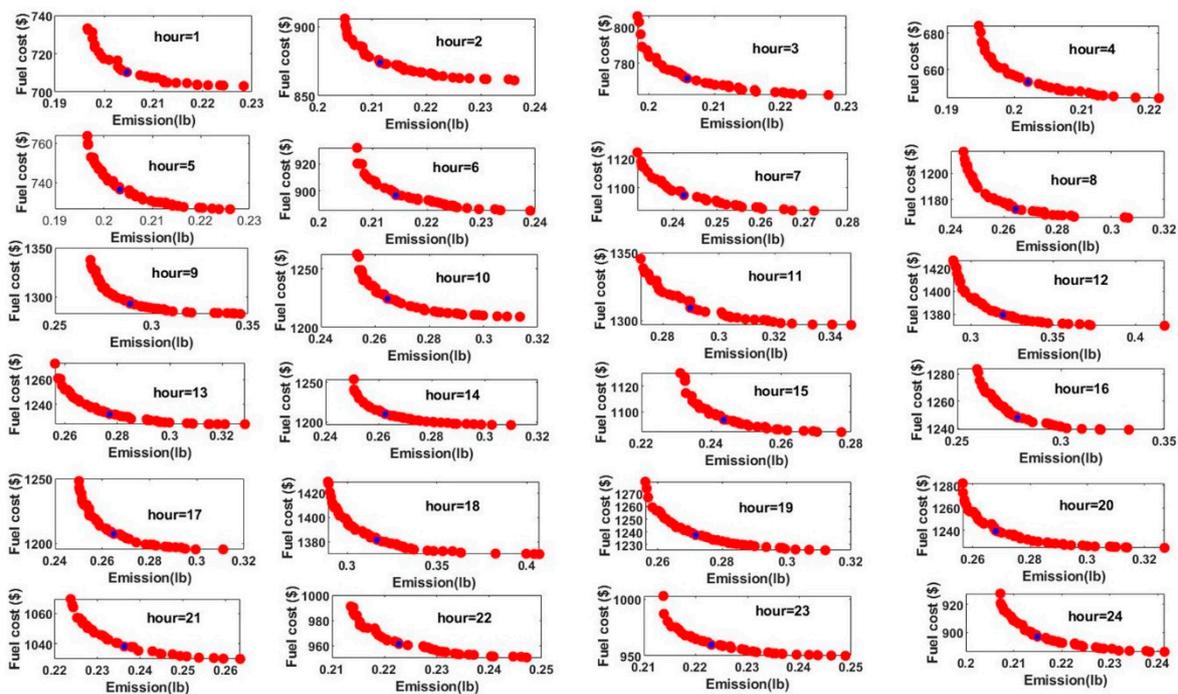


Figure 15. PO Solutions (red dot) and compromise solution (blue dot) for each hour of the dispatch period for the IEEE 30 bus system.

The results obtained using the MOSPO algorithm were compared with the GSOMP algorithm in [17] and the MOMVO algorithm in [29]. The fuel cost and emission levels obtained for the best compromise solution using the GSOMP algorithm and MOMVO algorithm were USD 25,924.4 and 6.0041 ton, and USD 25,831.48 and 5.9666 ton, respectively. The emission level obtained using the MOSPO algorithm was 5.9913 ton, which was almost the same as the other existing algorithms. The savings in the fuel cost obtained using the MOSPO algorithm are compared in Table 10. Table 10 shows the yearly savings obtained using the MOSPO algorithm compared to the GSOMP algorithm and MOMVO algorithm, and these were USD 37,339.5 and USD 3423.7, respectively. The time taken by the GSOMP algorithm was 1262.93 s, and this algorithm could outperform the NSGAI and MOSPO algorithms, as shown in [17]. The average time taken by the MOSPO algorithm was 1038.47 s, and it was much faster when compared to the time taken by the GSOMP algorithm.

Table 10. Fuel cost savings obtained using the MOSPO algorithm for the IEEE 30 bus system.

Duration	Compared Algorithms	Fuel Cost Savings in \$
One day	Proposed MOSPO with GSOMP [17]	102.3
One day	Proposed MOSPO with MOMVO [29]	9.38
One Month	Proposed MOSPO with GSOMP [17]	3069
One Month	Proposed MOSPO with MOMVO [29]	281.4
One Year	Proposed MOSPO with GSOMP [17]	37,339.5
One Year	Proposed MOSPO with MOMVO [29]	3423.7

8. Conclusions and Future Research Direction

In conclusion, this study presents a pioneering approach to tackling the highly strenuous and challenging bi-objective dynamic economic emission dispatch (DEED) model. The integration of a potent multi-objective stochastic paint optimizer (MOSPO) algorithm proved effective in addressing the complexities of the DEED problem, showcasing its superiority in solving both the fuel cost minimization and the pollutant emission reduction objectives. A key innovation lies in the incorporation of a random forest regression model, trained to predict the transmission loss specifically for the IEEE 30 bus system. This novel integration of machine learning into the DEED model overcomes the challenges associated with solving complex B-loss coefficients or conducting load flow analysis during each iteration. The elimination of these computational burdens significantly enhances the efficiency of the MOSPO algorithm.

This study initially establishes the effectiveness and superiority of the MOSPO algorithm by comparing its performance on five-unit and ten-unit systems. The results demonstrated that MOSPO outperforms most algorithms in at least one objective, affirming its robustness in solving the DEED problem. Notably, the MOSPO algorithm exhibited a faster convergence compared to the existing algorithms, further highlighting its efficiency. Subsequently, the random forest model, incorporated into MOSPO, was applied to solve the DEED for the IEEE 30 bus system. The achieved yearly fuel cost savings of USD 37,339.5 (0.396%) and USD 3423.7 (0.036%) when compared to existing GSOMP and MOMVO algorithms, respectively, underscoring the economic benefits of the MOSPO algorithm. Beyond cost savings, the algorithm's faster convergence adds another dimension to its practical applicability in real-time power system operations.

Looking ahead, future research endeavors could delve into expanding the algorithm's applicability to even larger and more complex power systems. The integration of additional machine learning techniques or optimization algorithms could further enhance the algorithm's performance and broaden its scope. Improving the predictive capabilities of the random forest model by incorporating real-time data and accommodating dynamic changes in the power system remains an avenue worth exploring. Additionally, extending the application of the proposed approach to address other optimization challenges within

power systems, such as unit commitment or economic dispatch with renewable energy sources, could contribute to a more comprehensive understanding of its capabilities.

In summary, future research should aim at advancing the algorithm's adaptability, refining its predictive capabilities, and exploring new domains within power system optimization. These efforts would further solidify the MOSPO algorithm's position as a robust and efficient solution for dynamic economic emission dispatch problems, fostering sustainability and efficiency in power system operations.

Author Contributions: Conceptualization, A.S. and N.S.A.; methodology, A.S.; software, A.S.; validation, N.S.A. formal analysis, A.S., resources, N.S.A.; writing—original draft preparation, A.S.; writing—review and editing, N.S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: A link to download the data has been provided: <https://doi.org/10.24433/CO.5812681.v1> accessed on 7 January 2024.

Acknowledgments: The authors would like to thank the facilities provided by the authorities of the Royal Commission of Jubail and Yanbu, and the management of Jubail Industrial College to carry out this research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Liu, C.; Shahidehpour, M.; Li, Z.; Fotuhi-Firuzabad, M. Component and Mode Models for the Short-Term Scheduling of Combined-Cycle Units. *IEEE Trans. Power Syst.* **2009**, *24*, 976–990. [CrossRef]
- Chiang, C.L. Improved Genetic Algorithm for Power Economic Dispatch of Units with Valve-Point Effects and Multiple Fuels. *IEEE Trans. Power Syst.* **2005**, *20*, 1690–1699. [CrossRef]
- Basu, M. Particle Swarm Optimization Based Goal-Attainment Method for Dynamic Economic Emission Dispatch. *Electr. Power Compon. Syst.* **2006**, *34*, 1015–1025. [CrossRef]
- Basu, M. Dynamic Economic Emission Dispatch Using Evolutionary Programming and Fuzzy Satisfying Method. *Int. J. Emerg. Electr. Power Syst.* **2007**, *8*. [CrossRef]
- Alsumait, J.S.; Qasem, M.; Sykulski, J.K.; Al-Othman, A.K. An Improved Pattern Search Based Algorithm to Solve the Dynamic Economic Dispatch Problem with Valve-Point Effect. *Energy Convers. Manag.* **2010**, *51*, 2062–2067. [CrossRef]
- Mason, K.; Duggan, J.; Howley, E. Multi-Objective Dynamic Economic Emission Dispatch Using Particle Swarm Optimisation Variants. *Neurocomputing* **2017**, *270*, 188–197. [CrossRef]
- Niu, Q.; Zhang, H.; Li, K.; Irwin, G.W. An Efficient Harmony Search with New Pitch Adjustment for Dynamic Economic Dispatch. *Energy* **2014**, *65*, 25–43. [CrossRef]
- Li, Z.; Zou, D.; Kong, Z. A Harmony Search Variant and a Useful Constraint Handling Method for the Dynamic Economic Emission Dispatch Problems Considering Transmission Loss. *Eng. Appl. Artif. Intell.* **2019**, *84*, 18–40. [CrossRef]
- Shen, X.; Zou, D.; Duan, N.; Zhang, Q. An Efficient Fitness-Based Differential Evolution Algorithm and a Constraint Handling Technique for Dynamic Economic Emission Dispatch. *Energy* **2019**, *186*, 115801. [CrossRef]
- Parihar, S.S.; Malik, N. Fuzzy-Based Real-Coded Genetic Algorithm for Optimizing Non-Convex Environmental Economic Loss Dispatch. *Facta Univ. Ser. Electron. Energetics* **2022**, *35*, 495–512. [CrossRef]
- Roy, P.K.; Bhui, S. A Multi-Objective Hybrid Evolutionary Algorithm for Dynamic Economic Emission Load Dispatch. *Int. Trans. Electr. Energy Syst.* **2016**, *26*, 49–78. [CrossRef]
- Elaiw, A.M.; Xia, X.; Shehata, A.M. Hybrid DE-SQP and Hybrid PSO-SQP Methods for Solving Dynamic Economic Emission Dispatch Problem with Valve-Point Effects. *Electr. Power Syst. Res.* **2013**, *103*, 192–200. [CrossRef]
- Sheng, W.; Li, R.; Yan, T.; Tseng, M.L.; Lou, J.; Li, L. A Hybrid Dynamic Economics Emissions Dispatch Model: Distributed Renewable Power Systems Based on Improved COOT Optimization Algorithm. *Renew. Energy* **2023**, *204*, 493–506. [CrossRef]
- Nagarajan, K.; Rajagopalan, A.; Angalaeswari, S.; Natrayan, L.; Mammo, W.D. Combined Economic Emission Dispatch of Microgrid with the Incorporation of Renewable Energy Sources Using Improved Mayfly Optimization Algorithm. *Comput. Intell. Neurosci.* **2022**, *2022*, 6461690. [CrossRef]
- Basu, M. Dynamic Economic Emission Dispatch Using Nondominated Sorting Genetic Algorithm-II. *Int. J. Electr. Power Energy Syst.* **2008**, *30*, 140–149. [CrossRef]
- Dhanalakshmi, S.; Kannan, S.; Mahadevan, K.; Baskar, S. Application of Modified NSGA-II Algorithm to Combined Economic and Emission Dispatch Problem. *Int. J. Electr. Power Energy Syst.* **2011**, *33*, 992–1002. [CrossRef]
- Guo, C.X.; Zhan, J.P.; Wu, Q.H. Dynamic Economic Emission Dispatch Based on Group Search Optimizer with Multiple Producers. *Electr. Power Syst. Res.* **2012**, *86*, 8–16. [CrossRef]

18. Wu, L.H.; Wang, Y.N.; Yuan, X.F.; Zhou, S.W. Environmental/Economic Power Dispatch Problem Using Multi-Objective Differential Evolution Algorithm. *Electr. Power Syst. Res.* **2010**, *80*, 1171–1181. [[CrossRef](#)]
19. Jiang, X.; Zhou, J.; Wang, H.; Zhang, Y. Dynamic Environmental Economic Dispatch Using Multiobjective Differential Evolution Algorithm with Expanded Double Selection and Adaptive Random Restart. *Int. J. Electr. Power Energy Syst.* **2013**, *49*, 399–407. [[CrossRef](#)]
20. Zhang, H.; Yue, D.; Xie, X.; Hu, S.; Weng, S. Multi-Elite Guide Hybrid Differential Evolution with Simulated Annealing Technique for Dynamic Economic Emission Dispatch. *Appl. Soft Comput. J.* **2015**, *34*, 312–323. [[CrossRef](#)]
21. Shao, Z.; Si, F.; Wu, H.; Tong, X. An Agile and Intelligent Dynamic Economic Emission Dispatcher Based on Multi-Objective Proximal Policy Optimization. *Appl. Soft Comput.* **2021**, *102*, 107047. [[CrossRef](#)]
22. Mason, K.; Duggan, J.; Howley, E. A Multi-Objective Neural Network Trained with Differential Evolution for Dynamic Economic Emission Dispatch. *Int. J. Electr. Power Energy Syst.* **2018**, *100*, 201–221. [[CrossRef](#)]
23. Hassan, M.H.; Kamel, S.; Domínguez-García, J.L.; El-Naggar, M.F. MSSA-DEED: A Multi-Objective Salp Swarm Algorithm for Solving Dynamic Economic Emission Dispatch Problems. *Sustainability* **2022**, *14*, 9785. [[CrossRef](#)]
24. Yan, L.; Zhu, Z.; Kang, X.; Qu, B.; Qiao, B.; Huan, J.; Chai, X. Multi-Objective Dynamic Economic Emission Dispatch with Electric Vehicle–Wind Power Interaction Based on a Self-Adaptive Multiple-Learning Harmony-Search Algorithm. *Energies* **2022**, *15*, 4942. [[CrossRef](#)]
25. Lokeshgupta, B.; Sivasubramani, S. Dynamic Economic and Emission Dispatch with Renewable Energy Integration Under Uncertainties and Demand-Side Management. *Electr. Eng.* **2022**, *104*, 2237–2248. [[CrossRef](#)]
26. Zou, D.; Gong, D.; Ouyang, H. The Dynamic Economic Emission Dispatch of the Combined Heat and Power System Integrated with a Wind Farm and a Photovoltaic Plant. *Appl. Energy* **2023**, *351*, 121890. [[CrossRef](#)]
27. Arul, R.; Velusami, S.; Ravi, G. A New Algorithm for Combined Dynamic Economic Emission Dispatch with Security Constraints. *Energy* **2015**, *79*, 496–511. [[CrossRef](#)]
28. Zou, D.; Li, S.; Li, Z.; Kong, X. A New Global Particle Swarm Optimization for the Economic Emission Dispatch with or without Transmission Losses. *Energy Convers. Manag.* **2017**, *139*, 45–70. [[CrossRef](#)]
29. Sundaram, A. Multiobjective Multi Verse Optimization Algorithm to Solve Dynamic Economic Emission Dispatch Problem with Transmission Loss Prediction by an Artificial Neural Network. *Appl. Soft Comput.* **2022**, *124*, 109021. [[CrossRef](#)]
30. Sundaram, A. Fair and Transparent Loss Allocation and Pricing in Primary Electricity Market Using the Concept of Market Center. Ph.D. Thesis, Anna University, Chennai, India, 2014. [[CrossRef](#)]
31. Arunachalam, S.; Saranya, R.; Sangeetha, N. Hybrid Artificial Bee Colony Algorithm and Simulated Annealing Algorithm for Combined Economic and Emission Dispatch Including Valve Point Effect. In *Swarm, Evolutionary, and Memetic Computing, Proceedings of the 4th International Conference, SEMCCO 2013, Chennai, India, 19–21 December 2013*; Springer International Publishing: Cham, Switzerland, 2013; Volume 8297, pp. 354–365.
32. Arunachalam, S.; AgnesBhomila, T.; Ramesh Babu, M. Hybrid Particle Swarm Optimization Algorithm and Firefly Algorithm Based Combined Economic and Emission Dispatch Including Valve Point Effect. In *Swarm, Evolutionary, and Memetic Computing, Proceedings of the 5th International Conference, SEMCCO 2014, Bhubaneswar, India, 18–20 December 2014*; Springer International Publishing: Cham, Switzerland, 2015; Volume 8947, pp. 647–660.
33. Sundaram, A. Solution of Combined Economic Emission Dispatch Problem with Valve-Point Effect Using Hybrid NSGA II-MOPSO. In *Particle Swarm Optimization with Applications*; Erdogmus, P., Ed.; IntechOpen: London, UK, 2018; pp. 81–100, ISBN 978-1-78923-149-6.
34. Murugan, R.; Mohan, M.R.; Asir Rajan, C.C.; Sundari, P.D.; Arunachalam, S. Hybridizing Bat Algorithm with Artificial Bee Colony for Combined Heat and Power Economic Dispatch. *Appl. Soft Comput.* **2018**, *72*, 189–217. [[CrossRef](#)]
35. Sundaram, A. Multiobjective Multi-Verse Optimization Algorithm to Solve Combined Economic, Heat and Power Emission Dispatch Problems. *Appl. Soft Comput. J.* **2020**, *91*, 106195. [[CrossRef](#)]
36. Sundaram, A. Combined Heat and Power Economic Emission Dispatch Using Hybrid NSGA II-MOPSO Algorithm Incorporating an Effective Constraint Handling Mechanism. *IEEE Access* **2020**, *8*, 13748–13768. [[CrossRef](#)]
37. Ramachandran, M.; Mirjalili, S.; Malli Ramalingam, M.; Charles Gnanakkan, C.A.R.; Parvathysankar, D.S.; Sundaram, A. A Ranking-Based Fuzzy Adaptive Hybrid Crow Search Algorithm for Combined Heat and Power Economic Dispatch. *Expert. Syst. Appl.* **2022**, *197*, 116625. [[CrossRef](#)]
38. Ramachandran, M.; Mirjalili, S.; Nazari-Heris, M.; Parvathysankar, D.S.; Sundaram, A.; Charles Gnanakkan, C.A.R. A Hybrid Grasshopper Optimization Algorithm and Harris Hawks Optimizer for Combined Heat and Power Economic Dispatch Problem. *Eng. Appl. Artif. Intell.* **2022**, *111*, 104753. [[CrossRef](#)]
39. Wolpert, D.H.; Macready, W.G. No Free Lunch Theorems for Optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67. [[CrossRef](#)]
40. Kaveh, A.; Talatahari, S.; Khodadadi, N. Stochastic Paint Optimizer: Theory and Application in Civil Engineering. *Eng. Comput.* **2020**, *38*, 1921–1952. [[CrossRef](#)]
41. Khodadadi, N.; Abualigah, L.; Mirjalili, S. Multi-Objective Stochastic Paint Optimizer (MOSPO). *Neural Comput. Appl.* **2022**, *134*, 18035–18058. [[CrossRef](#)]
42. Coello Coello, C.A.; Lechuga, M.S. MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, Honolulu, HI, USA, 12–17 May 2002; IEEE: Piscataway, NJ, USA, 2002; Volume 2, pp. 1051–1056.

43. Liu, Y.; Wang, Y.; Zhang, J. New Machine Learning Algorithm: Random Forest. In *Information Computing and Applications, Proceedings of the Third International Conference, ICICA 2012, Chengde, China, 14–16 September 2012*; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7473, pp. 246–252. [[CrossRef](#)]
44. Segal, M.R. *Machine Learning Benchmarks and Random Forest Regression*; University of California: San Francisco, CA, USA, 2004.
45. Wu, J.; Chen, X.Y.; Zhang, H.; Xiong, L.D.; Lei, H.; Deng, S.H. Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization. *J. Electron. Sci. Technol.* **2019**, *17*, 26–40. [[CrossRef](#)]
46. Qiao, B.; Liu, J.; Hao, X. A Multi-Objective Differential Evolution Algorithm and a Constraint Handling Mechanism Based on Variables Proportion for Dynamic Economic Emission Dispatch Problems. *Appl. Soft Comput.* **2021**, *108*, 107419. [[CrossRef](#)]
47. Pandit, N.; Tripathi, A.; Tapaswi, S.; Pandit, M. An Improved Bacterial Foraging Algorithm for Combined Static/Dynamic Environmental Economic Dispatch. *Appl. Soft Comput. J.* **2012**, *12*, 3500–3513. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.