frontiers in Energy Research

METHODS
published: 30 September 2021
doi: 10.3389/fenrg.2021.706782

A Novel Real-Coded Genetic Algorithm for Dynamic Economic Dispatch Integrating Plug-In Electric Vehicles

Wenqiang Yang[1], *Zhanlei Peng*[1], *Wei Feng*[2]* *and Muhammad Ilyas Menhas*[3]

[1]School of Mechanical and Electrical Engineering, Henan Institute of Science and Technology, Xinxiang, China, [2]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China, [3]Department of Electrical Engineering, Mirpur University of Science and Technology (MUST), Mirpur, Pakistan

OPEN ACCESS

Edited by:
Kailong Liu,
University of Warwick,
United Kingdom

Reviewed by:
Junainah Sardi,
Technical University of Malaysia
Malacca, Malaysia
Jiale Xie,
North China Electric Power University,
China

***Correspondence:**
Wei Feng
wei.feng@siat.ac.cn

Specialty section:
This article was submitted to
Smart Grids,
a section of the journal
Frontiers in Energy Research

Received: 10 May 2021
Accepted: 25 August 2021
Published: 30 September 2021

Citation:
Yang W, Peng Z, Feng W and
Menhas MI (2021) A Novel Real-Coded
Genetic Algorithm for Dynamic
Economic Dispatch Integrating Plug-In
Electric Vehicles.
Front. Energy Res. 9:706782.
doi: 10.3389/fenrg.2021.706782

Massive popularity of plug-in electric vehicles (PEVs) may bring considerable opportunities and challenges to the power grid. The scenario is highly dependent on whether PEVs can be effectively managed. Dynamic economic dispatch with PEVs (DED with PEVs) determines the optimal level of online units and PEVs, to minimize the fuel cost and grid fluctuations. Considering valve-point effects and transmission losses is a complex constrained optimization problem with non-smooth, non-linear, and non-convex characteristics. High efficient DED method provides a powerful tool in both power system scheduling and PEVs charging coordination. In this study, firstly, PEVs are integrated into the DED problem, which can carry out orderly charge and discharge management to improve the quality of the grid. To tackle this, a novel real-coded genetic algorithm (RCGA), namely, dimension-by-dimension mutation based on feature intervals (GADMFI), is proposed to enhance the exploitation and exploration of conventional RCGAs. Thirdly, a simple and efficient constraint handling method is proposed for an infeasible solution for DED. Finally, the proposed method is compared with the current literature on six cases with three scenarios, including only thermal units, units with disorderly PEVs, and units with orderly PEVs. The proposed GADMFI shows outstanding advantages on solving the DED with/without PEVs problem, obtaining the effect of cutting peaks and filling valleys on the DED with orderly PEVs problem.

Keywords: dimension-by-dimension mutation, horizontal vertical local search, collaborative optimization, constraints handling method, dynamic economic dispatch, plug-in electric vehicles, real-code genetic algorithm

INTRODUCTION

The Optimization Problem

Over the last few decades, the rapid increase in the use of fossil fuel has led to a consequential worldwide reduction of the resource; thus, its optimal utilization in power generation has become an important research topic (Niu et al., 2014; Yang et al., 2015). In addition, massive popularity of PEVs may bring opportunities or challenges to the power grid. Therefore, the DED with PEVs plays an important role in power systems operation and control. Coupling with space and time, it is a complicated optimal decision problem, and its goal is to minimize the fuel cost and fluctuation of the power grid, on the premise of satisfying a series of constraints.

TABLE 1 | The pseudo-code of conventional RCGA

Algorithm 1. Conventional RCGA
Initialize a population of N chromosomes $\boldsymbol{x_i}$, $i = 1,2,\ldots,N$.
Set p_c, p_m and maximum number of function evaluations ($MaxFEs$).
Calculate the fitness of each chromosome.
Let $FEs = N$.
While ($FEs \leq MaxFEs$)
Select parents by a selection operator.
If $rand(1,1) < p_c$
Generate children by a crossover operator.
End if
If $rand(1,1) < p_m$
Perform mutation.
End if
Calculate the fitness of each chromosome and update FEs.
Retain elite individuals
End while
Output the optimal solution and its corresponding function value.

TABLE 2 | The pseudo-code of the proposed GADMFI.

Algorithm 2. GADMFI
Initialize a population of N chromosomes $\boldsymbol{x_i}$, $i = 1,2,\ldots,N$.
Set p_c, p_m and maximum number of function evaluations ($MaxFEs$).
Calculate the fitness of each chromosome.
Let $FEs = N$.
While ($FEs \leq MaxFEs$)
Divide the population into the good and the bad populations.
Extract the genetic characteristics of excellent chromosomes by Eq. (1).
If $rand(1,1) < p_c$
Generate children by the flat crossover operator.
End if
If in the good population
Perform dimension-by-dimension mutation based on feature intervals by Eqs. (2-3).
Else
If $rand(1,1) < p_m$
Perform uniform mutation based on opposite feature intervals by Eqs. (4-5).
End if
End if
Calculate the fitness of each chromosome and update FEs.
Retain elite individuals.
End while
Output the optimal solution and its corresponding function value.

The DED problem was first introduced in 1971 by Bechert and Kwanty (Niu et al., 2014) and an extension of static economic dispatch (SED), taking into account ramp rate limits. Owing to the inertia of thermal units, the output from the units is not changed significantly from one operating hour to the next, which can avoid shortening the life of the units (Elaiw et al., 2013). The fuel cost function is a highly discontinuous, nonlinear, and non-convex curve, due to the valve-point effect (VPE) of the steam turbine (Shen et al., 2019), and the transmission loss should not be ignored on a large scale of the power system. As a mobile, distributed energy storage, PEVs charge from the grid through the grid to vehicle technology (G2V), and feedback to the grid through the vehicle to gird technology (V2G) (Wang et al., 2019a). So, VPE, transmission losses, and PEVs make the DED model more complicated, but more accurate.

Over the past few years, optimization methods based on artificial intelligence have been successfully and popularly applied to DED. Meta-heuristic algorithms do not care certain mathematical properties of the objective function such as continuous, differentiability, and convex, compared with traditional techniques. They include simulated annealing (SA) (Panigrahi et al., 2007), genetic algorithm (GA), particle swarm optimization (PSO) (Sawyerr et al., 2011), differential evolution (DE), artificial immune system (AIS) (Hemamalini and Simon, 2011a; Basu, 2011), cuckoo search (CS) (Mellal and Williams, 2020), artificial bee colony (ABC) (Hemamalini and Simon, 2011b; Tehzeeb-ul-Hassan et al., 2020), harmony search (HS) (Ravikumar Pandi and Panigrahi, 2011; Chakraborty et al., 2012; Li et al., 2019), an efficient fitness-based differential evolution (EFDE) (Shen et al., 2019), hybrid different evolution (DE), sequential quadratic programming (SQP), and hybrid PSO and SQP (Elaiw et al., 2013). However, few scholars integrate PEVs into the DED problem to perform effective management considering network loss. Panpan M. et al. manage the charging and discharging of PEVs through the scene method in the problem (Yang et al., 2014; Mei et al., 2019). Behera et al. (2019) ignored two important constraints: the power balance constraint and the ramp rate limit.

The Optimization Algorithm

Since GA was proposed by Holland in 1975 (Ali et al., 2018), which has undergone two revisions: the binary-coded genetic algorithm (BCGA) and real-coding genetic algorithm (RCGA). RCGA was first suggested by Herrera in 1998 (Akopov et al., 2019; Iyer et al., 2019). It is known that its performance depends heavily on crossover and mutation operators (Thakur et al., 2014), and then scholars mainly focused on the improvement of crossover and mutation operators and proposed many excellent variants of RCGA.

The crossover operator generates new individuals through interactive information among existing ones (Nakane et al., 2020). Arithmetic crossover (AX) (Naqvi et al., 2020) produces offspring through the linear combination of the parents. By flat crossover (FX) (Picek et al., 2013a), the parents exchange genes to produce offspring, but it does not destroy genetic information in the population. In LX (Deep and Thakur, 2007), Laplace distribution is used as the density function to generate genes near the parents. In 1993, Eshelman et al. used the concept of interval schemata to develop a blend crossover operator (BLX-α) (Wang et al., 2019b), which can do linear exploration around the parents. SPX is a multi-paternal crossover operator based on the nature of the simplex and is an extension of BLX (Chuang et al.,

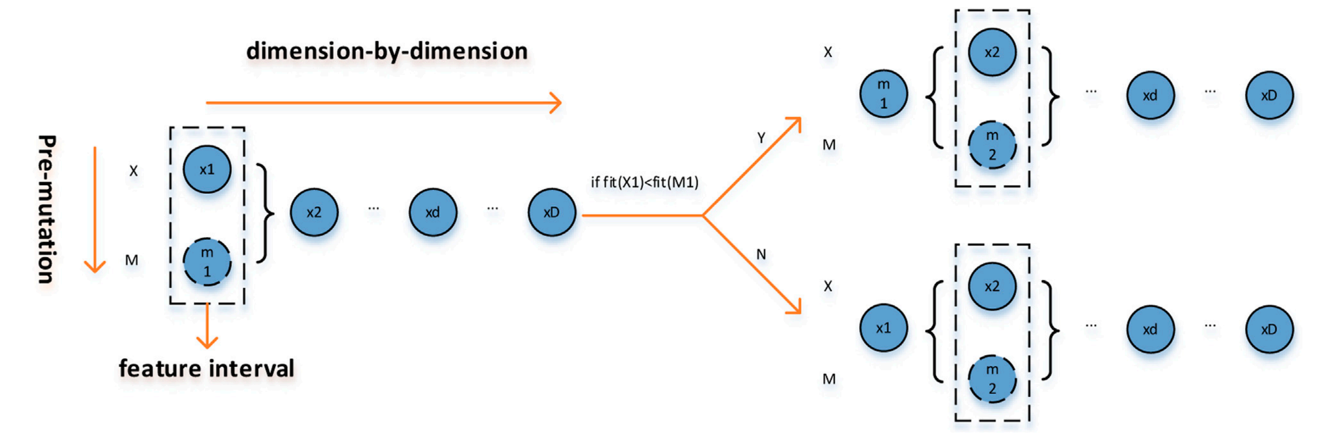

FIGURE 1 | Diagram of dimension-by-dimension mutation based on feature intervals.

2015). SBX (Naqvi et al., 2020), as its name implies, is formed by simulating binary crossover operator; UNDX can produce two or more offspring individuals from three parents (Kwak and Lee, 2016). The comparative research results have been reported in the literature (Picek et al., 2013b; Naqvi et al., 2020) for the above-mentioned crossover operators.

Mutation is the important operation responsible for exploration in RCGA (Wang et al., 2018), especially the local development. The popular and widespread mutation operators in the literature mostly use a certain distribution as a density function to generate random numbers around the gene to vary, so as to carry out the local development of the gene, which may lead to a lot of troubles in the application of the algorithm due to the introduction of extra parameters, such as the random mutation (RM), the non-uniform mutation (NUM) (Wang et al., 2019b), the power mutation (PM), the polynomial mutation (PLM), the Gaussian mutation (GM) (Wang et al., 2019b), and Cauchy mutation (CM). In addition, the direction mutation is presented (Tang and Tseng, 2013), which mutates toward a promising area by utilizing statistical population information and has achieved good results. On the basis of previous studies, two simple and efficient mutation operators are proposed. One is the dimension-by-dimension mutation based on feature intervals (DMFI), which combines horizontal search at the component level with the rule of greed to form a directional horizontal local development strategy; what is more, the genetic characteristics of outstanding individuals are extracted as the variation interval, thereby providing a directional vertical local development capability. The other is the uniform mutation based on the interval of opposing features (UMOFI); the opposing of feature intervals of outstanding individuals is employed as the variation interval for the inferior individuals, thereby introducing new information in the population.

At present, genetic algorithms have the defects of falling into local optimal and lack of exploration capabilities for large-scale and high-dimensional problems (Sawyerr et al., 2011; Fang et al., 2014; D'Angelo and Palmieri, 2021; Sawyerr et al., 2014). In the final analysis, this is a fundamental and difficult problem faced by metaheuristics: how to balance the exploration and exploitation of the algorithm. In general, researchers design or improve an algorithm based on the idea that focuses on exploration in the early stage and later on exploitation. However, if the development capability is not enough to find the global optimal area in the early stage, it will become stuck in the local optimal. Given this, this study provides a new solution, based on the idea of collaborative optimization of the superior and inferior individuals, in which excellent individuals strengthen local search. Inferior individuals are responsible for introducing new information to increase the diversity of the population. Then, interacting information between excellent individuals and inferior individuals helps to find outstanding individuals and realizing their comprehensive development carried out in each iterative.

The characteristics of GA's mechanisms and the new solutions mentioned above are consistent on the issue of balancing the global and local search capabilities. Therefore, an RCGA based on co-optimization of superior and inferior individuals is proposed: 1) through the rank selection and the flat crossover to realize the genetic interaction between superior and inferior populations, 2) using DMFI to obtain the ability of directed vertical and horizontal local development for excellent individuals and achieve in-depth local search, and 3) using UMOFI for the inferior individuals, which is controlled by the mutation probability *Pm*, thereby introducing new genetic information while maintaining the diversity of the population.

Constraint Handling Methods

For constraint optimization problems, the feasibility of the solutions is more important than the objective values. The penalty function method is common and popular for handling some constraints (Shen et al., 2019). However, it is a troublesome thing to choose a suitable penalty factor. Hence, some scholars proposed several types of repair methods to meet problem constraints. Feasibility-based rules are used to lead the search toward the feasible region to handle inequality constraints effectively (Yuan et al., 2009), which is more efficient to filter feasible solutions and ease the burden of setting the penalty factor

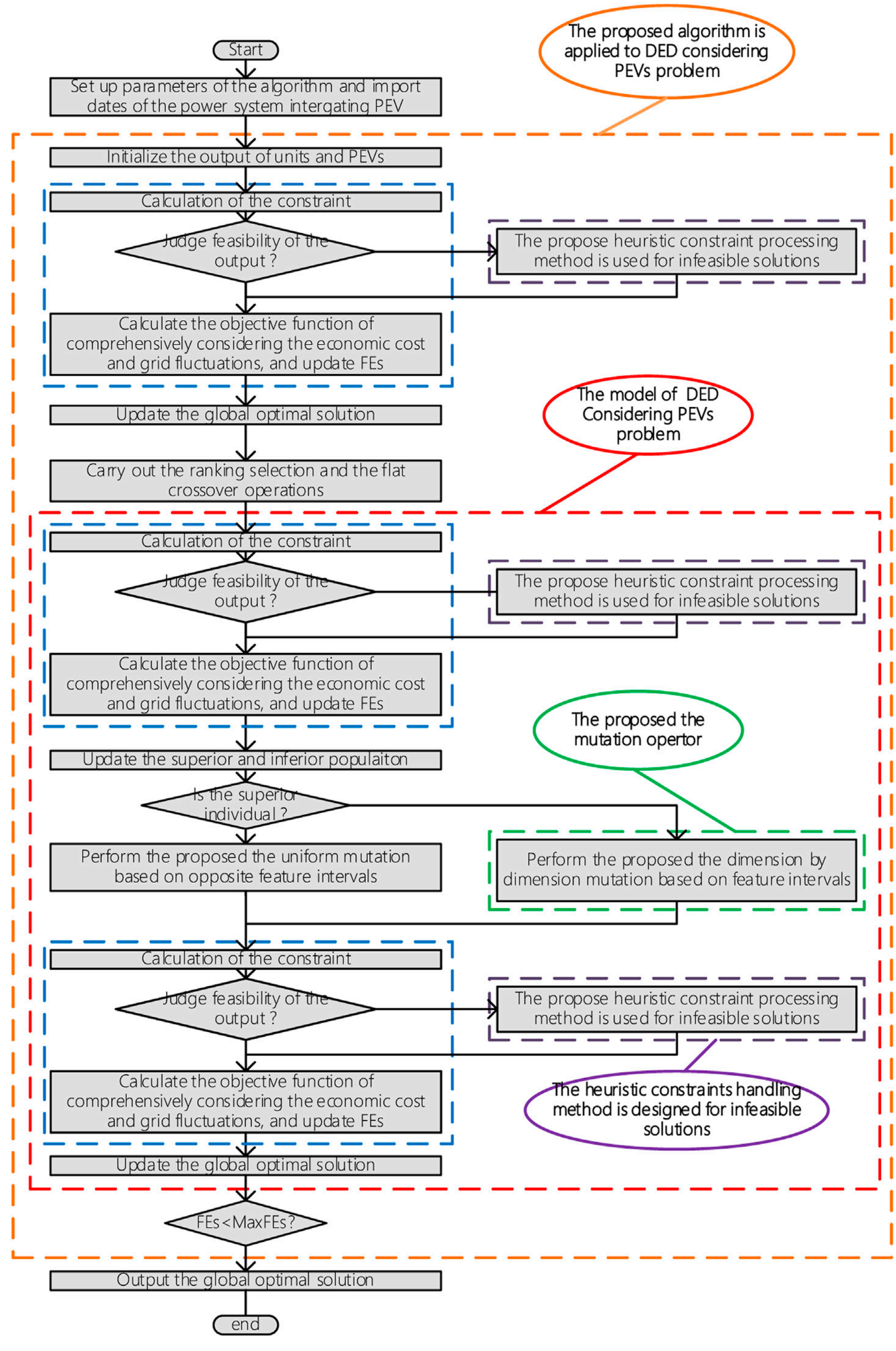

FIGURE 2 | The flowchart of the implementation of GADMFI for the DED with PEVs.

compared to the penalty function method. However, for optimization problems with equality constraints, owing to feasible solutions occupying an extremely small proportion of the solution space, it is difficult to repair by screening feasible solutions and guiding infeasible solutions. Heuristics for handling constraints perform excellently to deal with equality constraints without considering the transmission loss (Wang et al., 2011). They adjust the output power according to general experience to gradually reduce the violation of the constraint until it becomes a feasible solution, whereas this cannot efficiently solve complex equality constraints, for instance, the power balance

TABLE 3 | The basic information of the six benchmark functions.

No.	Function	Dimension (D)	Characteristics	Range	Min
F1	Step	30,50,100	US	(–100,100)	0
F2	Levy	30,50,100	MS	(–10,10)	0
F3	Rastrigin	30,50,100	MS	(–5.12,5.12)	0
F4	Schwefel	30,50,100	MS	(–500,500)	0
F5	Griwank	30,50,100	MN	(–600,600)	0
F6	Ackley	30,50,100	MN	(–32.768,32.768)	0

TABLE 4 | The private parameters of the algorithms.

Algorithms	Parameter settings
GWO	$NP = 30.$
ABC	$NP = 30, limit = 30.$
BAHTFS	$NP = 100, A(0) = 0.95, r(0) = 0.9, \alpha = 0.99, \gamma = 0.9.$
WOA	$NP = 30.$
DEWOA	$NP = 30, F = 0.5, CR = 0.9.$
GADMFI	$NP = 100,\ Pc = 0.7, Pm = 0.3.$

constraint with the transmission loss, and there may be an excessive adjustment, especially when the amount of violation is not big. The forced repair technology refers to adjusting the output strictly according to the characteristics of the equation, which greatly improves repair efficiency of equality constraints (Panigrahi et al., 2007; Zou et al., 2018). Apart from equality limits, inequality constraints also influence the feasibility, and hence the forced repair technology may spend more time solving the quadratic equation of DED with transmission losses. In Li et al. (2019) and Shen et al. (2019), the constraints handling technology combining a heuristic repair technology and the forced repair technology is proposed, which enhances the capability of repairing infeasible solutions.

In summary, the methods or techniques above-mentioned have the following shortcomings in solving DED with PEVs: 1) the algorithms are difficult to solve large-scale problems or face premature phenomena and 2) constraint processing techniques are difficult to repair infeasible solutions or do not work well with the algorithm. In view of this, this study proposed GADMFI based on RCGA. Meanwhile, a simple and efficient constraint processing technique is designed.

The remainder of this study is arranged as follows: *The Formulation of the Dynamic Economic Dispatch With Plug-In Electric Vehicles* gives the mathematical formulation of the DED with PEVs. *Genetic Algorithm Dimension Mutation Based on Feature Intervals* describes GADMFI in detail. In *Constraints Handling Method*, the constraint handling methods are proposed. *The Implementation of Genetic Algorithm Dimension Mutation Based on Feature Intervals for Dynamic Economic Dispatch With Plug-In Electric Vehicles* presents the implementation of GADMFI on the DED problem. *Experimental Results and Analysis* designs and analyzes experiments. Finally, the study ends with conclusions and further research work.

THE FORMULATION OF THE DYNAMIC ECONOMIC DISPATCH WITH PLUG-IN ELECTRIC VEHICLES

The DED integrating PEVs aims to determine the optimal generation levels of all online units and PEVs, during a specified period of time (e.g., 24 intervals a day), so as to minimize the total fuel cost and subject to a number of equality and inequality constraints.

The Optimization Objectives

There are two optimization objectives for this problem: one is to minimize fuel costs and the other is to minimize the fluctuation of the grid, that is, maximize peak shaving and valley filling. They are described in **Eqs. 1**, **2**.

$$\min f_1(P) = \sum_{t=1}^{T} \sum_{i=1}^{N} a_i + b_i P_{t,i} + c_i P_{t,i}^2 + \left| c_i sin\left[f_i \left(P_i^{min} - P_{t,i} \right) \right] \right|. \tag{1}$$

$$\min f_2(P, P_{PEV}) = \sum_{t=1}^{T-1} \left[\sum_{i=1}^{N} \left(P_{t+1,i} + P_{PEV,t+1} - P_{L,t+1} \right) - \sum_{i=1}^{N} \left(P_{t,i} + P_{PEV,t} - P_{L,t} \right) \right]^2, \tag{2}$$

where P and P_{PEV} constitute the decision variables of the problem and $P_{L,t}$ is the transmission loss at time t.

Constraints

The DED with PEVs is an optimization problem containing multiple inequalities and equality constraints, including capacity constraints, ramp rate limits, PEVs charge/discharge limits, PEVs demand limits, and power balance constraints.

Capacity Constraints

The capacity limits of the thermal unit are inequality constraints, which are determined by the physical characteristics of the unit and are given as follows:

$$P_i^{min} \le P_{t,i} \le P_i^{max}, \tag{3}$$

where P_i^{min} and P_i^{max} represent the min and max output power of the ith unit, respectively.

Ramp Rate Limits

Due to the inertia of thermal power units, the ramp rate limits are considered to extend the service life of the units; that is, the output

TABLE 5 | The statistics data of 30 runs of the benchmarks of 30 dimensions.

No.	Statistics	ABC	GWO	WOA	BA-HTFS	DEWOA	GADMFI
F1	Mean	0.469387	0.008336	6.63E-16	6.05E-06	8.18E-26	**0**
	Std	0.216323	0.045657	1.1E-16	2.92E-06	1.55E-26	**0**
	Best	0.235973	4.49E-08	4.35E-16	2.14E-06	4.98E-26	**0**
	Runtime (s)	1.176628	0.87896	1.440756	0.680118	1.008423	**0.279276**
	p-value	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	NA
	Winner	+	+	+	+	+	NA
F2	Mean	1.0809	0.352752	6.03E-16	0.011978	27.22283	**1.5E-32**
	Std	0.210637	0.208362	9.45E-17	0.030962	8.883094	**1.11E-47**
	Best	0.634462	1.51E-07	3.83E-16	9.7E-07	15.73242	**1.5E-32**
	Runtime (s)	3.225753	2.974774	3.628555	2.731598	3.131769	**2.42477**
	p-value	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	NA
	Winner	+	+	+	+	+	NA
F3	Mean	**0**	**0**	1.12E-13	3.22E-14	65.93248	**0**
	Std	**0**	**0**	5.87E-14	4.65E-14	15.71409	**0**
	Best	**0**	**0**	**0**	**0**	37.80841	**0**
	Runtime (s)	**0.037284**	0.234023	1.454388	0.615531	1.137729	0.204316
	p-value	1	1	2.56E-06	0.000488	1.73E-06	NA
	Winner	=	=	+	+	+	NA
F4	Mean	6261.479	1278.239	**0.000382**	11.8921	5095.71	0.636312
	Std	432.9709	1036.075	**1.01E-08**	36.19683	591.7563	1.787811
	Best	5462.455	0.003162	**0.000382**	0.000881	4012.246	**0.000382**
	Runtime (s)	1.450033	1.153483	1.703252	0.926646	1.255747	**0.595745**
	p-value	1.73E-06	2.35E-06	0.056952	0.025637	1.73E-06	NA
	Winner	+	+	=	+	+	NA
F5	Mean	0.000504	**0**	1.95E-12	0.001236	0.009351	**0**
	Std	0.001919	**0**	8.85E-12	0.00439	0.009712	**0**
	Best	**0**	**0**	1.11E-16	**0**	**0**	**0**
	Runtime (s)	0.12435	**0.080368**	1.610477	0.891262	0.960121	0.233424
	p-value	0.5	1	1.71E-06	5.6E-06	0.000192	NA
	Winner	=	=	+	+	+	NA
F6	Mean	7.4E-15	**4.2E-15**	5.12E-14	7.51E-09	4.331715	7.76E-15
	Std	1.35E-15	**9.01E-16**	5.68E-15	5.63E-09	1.039935	**9.01E-16**
	Best	4.44E-15	**8.88E-16**	4E-14	9.86E-10	2.738319	4.44E-15
	Runtime (s)	1.237275	0.954717	1.558052	0.858519	1.175921	**0.463191**
	p-value	0.453125	1.14E-07	1.4E-06	1.73E-06	1.73E-06	NA
	Winner	=	**-**	+	+	+	NA

Bold digits are the best statistical performance measures of various algorithms.

of the unit cannot be adjusted greatly in a short time, and the output of current time will affect the output of the next time:

$$\begin{cases} P_{t,i} - P_{t-1,i} \le UR_i \\ P_{t-1,i} - P_{t,i} \le DR_i \end{cases}, \tag{4}$$

where UR_i and DR_i, respectively, represent the maximum allowable rise and fall of the ith unit, which are limited to its physical characteristics.

Plug-In Electric Vehicles Charge/Discharge Limits

The maximum charging power and discharging power of PEVs should be limited to a normal range. Because different types of electric vehicles have different models, the charging and discharging power of PEVs at t time is described as a variable $P_{PEV,t}$:

$$P_{PEV,disc}^{max} \le P_{PEV,t} \le P_{PEV,char}^{max}. \tag{5}$$

The Plug-In Electric Vehicles Demand Constraint

For users' daily travel, the PEVs demand constraint should be met (Yang et al., 2017a), which is described as **Eq. 6**:

$$\sum_{t=1}^{T} P_{PEV,t} \le P_{PEV,total}, \tag{6}$$

where $P_{PEV,total}$ is the desired power for daily use.

The Power Balance Constraint

Power balance limit is the most important and complex constraint, especially considering the transmission loss, which is defined as

$$\sum_{t=1}^{T} P_{t,i} = P_{D,t} + P_{L,t} + P_{PEV,t}, \tag{7}$$

where $P_{D,t}$ presents the load demand at time t, $P_{L,t}$ is the transmission loss, and its mathematic model is expressed by Kron's loss (Abdelaziz et al., 2008) as **Eq. 8**.

$$P_{L,t} = \sum_{i=1}^{N} \sum_{j=1}^{N} P_{t,i} B_{ij} P_{t,j} + \sum_{j=1}^{N} B_{0i} P_{t,i} + B_{00}, \tag{8}$$

where B_{ij}, B_{0i}, and B_{00} represent the loss coefficients of the generation units. In addition, the model of the transmission loss is usually simplified as **Eq. 9** (Pan et al., 2018), which is adopted in this study:

$$P_{L,t} = \sum_{i=1}^{N} \sum_{j=1}^{N} P_{t,i} B_{ij} P_{t,j}. \tag{9}$$

TABLE 6 | The statistics data of 30 runs of the benchmarks of 50 dimensions.

No.	Statistics	ABC	GWO	WOA	BA-HTFS	DEWOA	GADMFI
F1	Mean	1.889057	0.025157	1.47E-15	1.61E-05	6.9E-32	**0**
	Std	0.579573	0.076187	2.14E-16	5.35E-06	2.63E-32	**0**
	Best	0.917128	1.18E-07	1.05E-15	7.52E-06	2.47E-32	**0**
	Runtime (s)	2.785521	1.756891	2.483228	1.413194	1.838462	**0.475277**
	p-value	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.71E-06	NA
	Winner	+	+	+	+	+	NA
F2	Mean	2.433452	0.820967	1.42E-15	0.003063	46.05496	**1.5E-32**
	Std	0.256288	0.294353	2.19E-16	0.016334	17.74721	**1.11E-47**
	Best	1.995291	0.358323	9.43E-16	5.46E-06	17.44572	**1.5E-32**
	Runtime (s)	8.467784	7.633276	8.540686	7.100606	7.783307	**6.28073**
	p-value	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	NA
	Winner	+	+	+	+	+	NA
F3	Mean	**0**	**0**	3.02E-12	2.65E-14	129.8418	**0**
	Std	**0**	**0**	9E-12	4.89E-14	20.13228	**0**
	Best	**0**	**0**	1.71E-13	**0**	96.51085	**0**
	Runtime (s)	**0.06699**	0.234524	2.884798	1.101694	2.308427	0.41372
	p-value	1	1	1.71E-06	0.007813	1.73E-06	NA
	Winner	=	=	+	+	+	NA
F4	Mean	11793.61	1672.455	**0.36091**	34.99552	9070.586	3.23715
	Std	790.472	1921.418	**1.358315**	123.4267	1098.779	6.859328
	Best	10093.76	0.00519	**0.000636**	0.001336	7090.506	**0.000636**
	Runtime (s)	3.492594	2.43678	3.152868	2.034369	2.513048	**1.169097**
	p-value	1.73E-06	5.22E-06	0.734325	0.585712	1.73E-06	NA
	Winner	+	+	=	=	+	NA
F5	Mean	**0**	0.000247	1.22E-14	0.00058	0.009681	**0**
	Std	**0**	0.001351	3.31E-14	0.002231	0.013748	**0**
	Best	**0**	**0**	1.11E-16	**0**	**0**	**0**
	Runtime (s)	**0.064767**	0.206464	2.976106	1.885592	2.269486	0.483447
	p-value	1	1	1.71E-06	5.69E-05	2.56E-06	NA
	Winner	=	=	+	+	+	NA
F6	Mean	8.82E-15	**4.09E-15**	1.13E-13	2.3E-09	7.357773	1.17E-14
	Std	2.02E-15	**1.08E-15**	1.41E-14	1.98E-09	1.212143	2.72E-15
	Best	7.99E-15	**8.88E-16**	8.62E-14	3.22E-11	5.051174	7.99E-15
	Runtime (s)	2.947703	1.952596	2.831282	1.891285	2.337538	**0.848766**
	p-value	0.000384	1.11E-06	1.63E-06	1.73E-06	1.73E-06	NA
	Winner	+	**-**	+	+	+	NA

Bold digits are the best statistical performance measures of various algorithms.

PROPOSED GENETIC ALGORITHM DIMENSION-BY-DIMENSION MUTATION BASED ON FEATURE INTERVALS

RCGAs generally consist of selection, crossover, mutation, and elite retention strategies, and the pseudo-code is summarized in **Table 1**. Without destroying its main structure and extra parameters, this study proposes a simple and efficient, superior and inferior population collaborative optimization algorithm, namely, a novel real-coded genetic algorithm: GADMFI. Its pseudo-code is shown in **Table 2**. It is worth noting that this study takes the minimum value of the function as the optimization objective.

In the study, two novel mutation operators are proposed: the dimension-by-dimension mutation based on feature intervals (DMFI) and the uniform mutation based on the interval of opposite features (UMOFI). DMFI and UMOFI are designed based on the idea that excellent individuals strengthen local exploitation capabilities, to improve the convergence accuracy and speed of the algorithm; low-quality individuals introduce new information, to improve population diversity; and good and bad individuals exchange information by an interactive operation. Excellent individuals perform DMFI to strengthen local development in both vertical and horizontal dimensions. Inferior individuals introduce new genes through UMOFI. Then, the information of the two is exchanged through the ranking selection (RS) and FX, so as to achieve the effect of collaborative optimization.

The Selection and Crossover Operator

The selection operator is the first operator of GA. One of the most widely used selection operators is the roulette selection. The higher the fitness, the greater the probability of being selected. However, the excellent genes of the inferior individuals may be abandoned. The other most commonly used is RS (Chuang et al., 2016). Excellent individuals are used as the parent 1 to cross, and inferior individuals are used as the parent 2. All individuals participate in the crossover with the same probability, which does not affect the diversity of the population. RS is used in the study; however, the difference between this study and the literature (Chuang et al., 2016) is random matching for the individuals, thus enhancing the population diversity.

The flat crossover (FX) is rarely used due to its poor local development ability, but it has the characteristics of not

TABLE 7 | The statistics data of 30 runs of the benchmarks of 100 dimension.

No.	Statistics	ABC	GWO	WOA	BA-HTFS	DEWOA	GADMFI
F1	Mean	8.729457	0.243518	4.48E-15	6.25E-05	5.07E-31	**0**
	Std	0.878348	0.24989	8.67E-16	1.21E-05	1.3E-31	**0**
	Best	7.13375	1.02E-06	2.99E-15	3.73E-05	2.56E-31	**0**
	Runtime (s)	9.74352	4.879711	5.330635	4.426168	4.566675	**1.025109**
	p-value	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	NA
	Winner	+	+	+	+	+	NA
F2	Mean	6.322538	1.62652	4.5E-15	0.015282	72.04177	**1.5E-32**
	Std	0.286192	0.561543	7.59E-16	0.082944	20.32359	**1.11E-47**
	Best	5.810564	0.542552	3.29E-15	1.59E-05	37.29714	**1.5E-32**
	Runtime (s)	31.50907	27.08374	27.96728	26.31595	26.88412	**22.98865**
	p-value	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	NA
	Winner	+	+	+	+	+	NA
F3	Mean	**0**	**0**	8.27E-09	4.17E-14	255.6375	5.55E-07
	Std	**0**	**0**	2.54E-08	6.99E-14	36.44933	3.04E-06
	Best	**0**	**0**	1.43E-11	**0**	184.0671	**0**
	Runtime (s)	0.186826	**0.155049**	6.498615	2.56054	5.997735	1.192726
	p-value	1	1	3.11E-05	0.097656	1.73E-06	NA
	Winner	=	=	+	=	+	NA
F4	Mean	25824.59	2476.58	174.0298	135.3648	18879.06	**25.38108**
	Std	1133.43	1961.341	100.3657	442.9999	1239.333	**60.22819**
	Best	23031.77	0.023106	0.001274	0.005186	15004.96	**0.001273**
	Runtime (s)	12.11576	7.066971	7.399292	6.467395	6.636016	**3.162116**
	p-value	1.73E-06	6.34E-06	8.92E-05	0.14704	1.73E-06	NA
	Winner	+	+	+	=	+	NA
F5	Mean	**0**	**0**	2.07E-14	0.000336	1.598701	**0**
	Std	**0**	**0**	2.59E-14	0.001841	0.615646	**0**
	Best	**0**	**0**	4.11E-15	**0**	0.893507	**0**
	Runtime (s)	**0.196128**	0.240497	7.51387	4.668474	6.900636	1.455745
	p-value	1	1	1.73E-06	0.015625	1.73E-06	NA
	Winner	=	=	+	+	+	NA
F6	Mean	1.38E-14	**3.73E-15**	3.58E-13	2.37E-10	11.61896	2.61E-14
	Std	2.18E-15	**1.45E-15**	6.19E-14	1.58E-10	0.963155	2.53E-15
	Best	7.99E-15	**8.88E-16**	2.42E-13	2.35E-11	9.291818	2.22E-14
	Runtime (s)	10.29742	5.368046	6.409801	6.116458	6.344188	**1.987388**
	p-value	1.13E-06	1.11E-06	1.72E-06	1.73E-06	1.73E-06	NA
	Winner	**-**	**-**	+	+	+	NA

Bold digits are the best statistical performance measures of various algorithms.

changing the genetic information in the population that other operators do not have. It can maintain the diversity of the population and the interaction between individuals. The role of RS and FX is to carry out information interaction between individuals without changing the genetic information of the population.

The Mutation Operator

Strong local search capabilities should be possessed for each stochastic algorithm. At the same time, it is also indispensable to maintain the diversity of the population. In order to illustrate the design idea of the mutation operator, horizontal search and vertical search are firstly defined. If there is a comparison between the variants or a variant and ontology in an operator, it is called horizontal search of individuals. If only components are changed, it is called the vertical search of the individual. In previous related studies, local search often refers to vertical search. All the components are developed simultaneously. In this section, horizontal search and the greedy rule, vertical search, and feature intervals are combined to fulfill horizontal and vertical local search so as to obtain in-depth development for the superior population by DMFI. In addition, new population information is also introduced into the inferior population by UMOFI.

The Dimension-by-Dimension Mutation Based on Feature Intervals

The dimension-by-dimension mutation is a dimension-by-dimension search for outstanding individuals in the characteristic interval and is combined with the greedy rule to achieve a directed horizontal search. **Figure 1** shows the change from one dimension to the next dimension of DMFI. A gene $x1$ of the chromosome X is pre-mutated to $m1$ in the feature interval that is formed by the minimum and maximum gene values of the excellent population individuals, which can be described as **Eqs. 10**, **11**. Because the feature interval contains the genetic characteristics of excellent individuals, the mutation will search toward a promising area and is used to realize the vertical local development for the superior population, which is different from the previous uniform mutation in that it is not centered on the individual, but the center of the superior population:

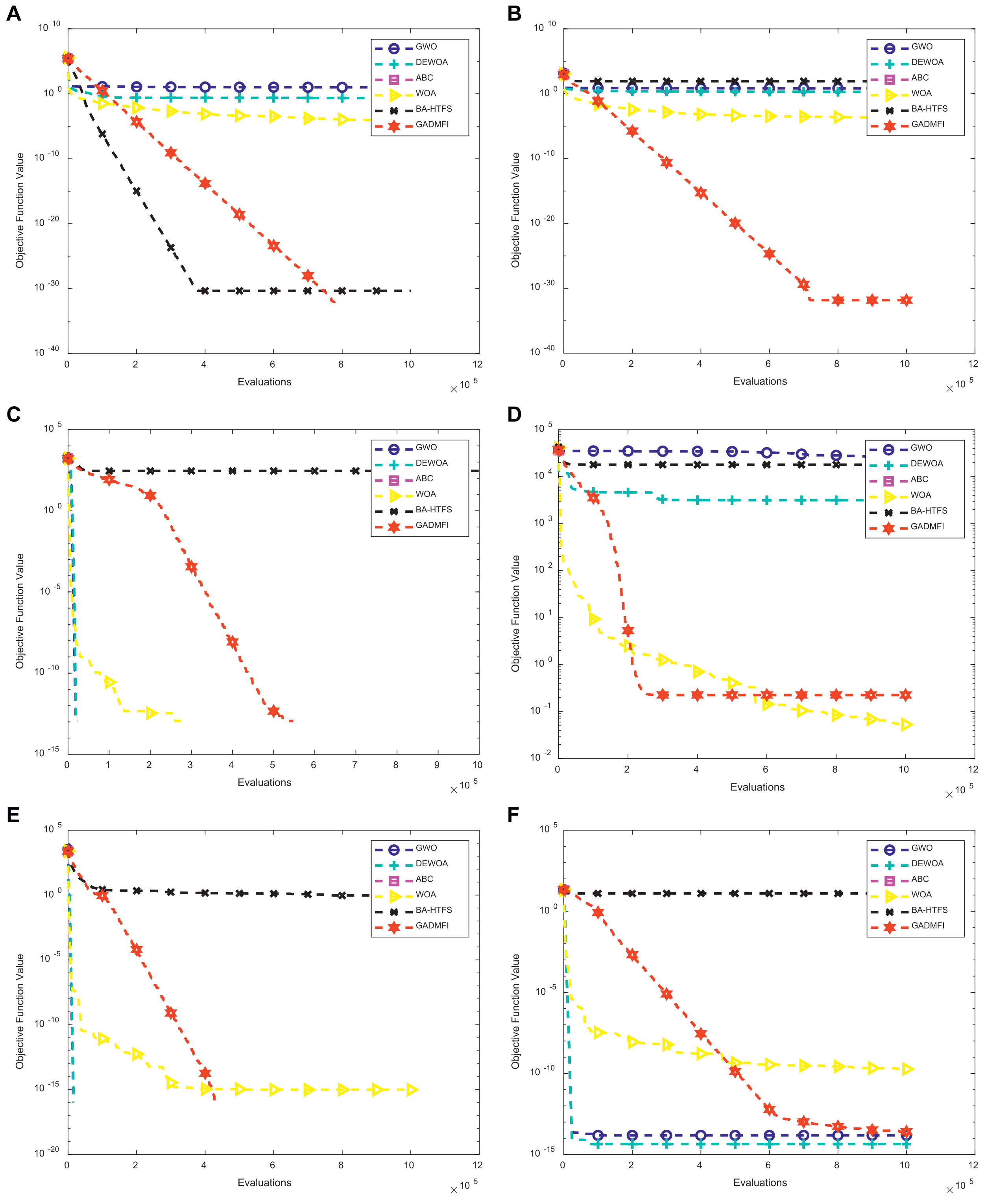

FIGURE 3 | Evolution curves of six functions of 100 dimensions are shown. As can be seen, the convergence curve of the GADMFI shows superior exploration and exploitation abilities. **(A)** The curve presents a straight line, which shows that when dealing with unimodal problems, the proposed DMFI has the potential to explore a promising area. **(B–F)** With multiple local minima, it stays at approximately constant speed until it converges, which again confirms the role of DMFI.

TABLE 8 | Six cases in three scenarios designed to verify the performance of GADMFI in DED with PEVs.

Scenario A: only units	Scenario B: units with disorderly PEVs	Scenario C: units with orderly PEVs
Case I: 5 units **Case II**: 10 units	**Case III**: 5 units with disorderly PEVs **Case IV**: 10 units with disorderly PEVs	**Case V**: 5 units with orderly PEVs **Case VI**: 10 units with orderly PEVs

TABLE 9 | Specific results of several algorithms for solving the DED without PEVs problem.

Case	Algorithms	Fuel costs				Constraints violations	
		Min	Mean	Max	Std	V_R	V_P
Case I	NEHS, Li et al. (2019)	43066.0731	43490.52344	44005.36139	182.95804	0	0.00013
	AIS, Hemamalini and Simon (2011a)	44385.43	44758.8363	45553.7707	NA	0	0.0012
	DE-SQP, Elaiw et al. (2013)	43261	NA	NA	NA	0	0.0015
	PSO-SQP, Elaiw et al. (2013)	43263	NA	NA	NA	0	0.0012
	EFDE, Shen et al (2019)	43047.851	43167.757	43325.179	64.047	0	0.000011
	SOA-SQP, Sivasubramani and Swarup (2010)	41923	NA	NA	NA	0	**184.2229**
	SA, Panigrahi et al. (2007)	47356	NA	NA	NA	0	0.1083
	GADMFI	**43030.079**	**43084.676**	**43168.454**	**36.585**	0	**9.15E-07**
Case II	HCRO, Elattar (2015)	2479931.38	2480143.473	2481367.921	NA	0	0.28626
	AIS, Basu (2011)	2.52E+06	NA	NA	NA	**4.5678**	**32.285**
	DE-SQP, Elaiw et al. (2013)	2.4659 + 06	NA	NA	NA	0	0.0083
	IBFA, Pandit et al. (2012)	2481733.25	NA	NA	NA	**37.929**	**43.36**
	PSO-SQP, Elaiw et al. (2013)	2.4668 + 06	NA	NA	NA	0	0.0102
	NEHS, Li et al. (2019)	**2463500.84**	2464881.787	2466310.211	791.781	0	**0.00015**
	GADMFI	2464270.10	**2464782.087**	**2.465280491**	**273.883**	**0**	0.0027067

Bold digits are the best statistical performance measures of various algorithms.

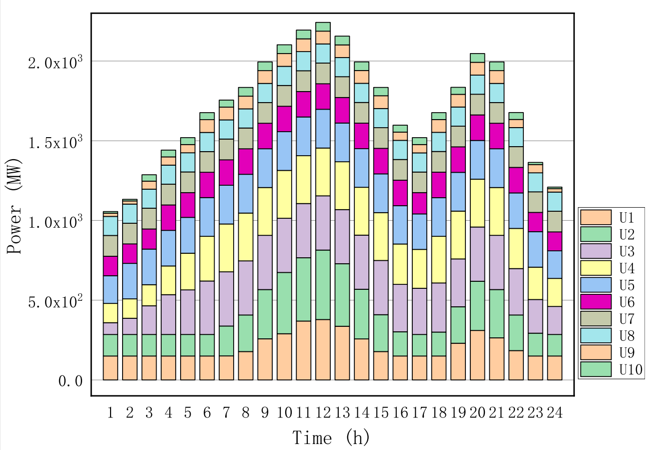

FIGURE 4 | The power output of the only 10-unit power system.

$$\left[S_{upper}(d), S_{lower}(d)\right] = \{min[S(d)], max[S(d)]\}, \tag{10}$$

$$m(d) = rand \times \left(S_{upper}(d) - S_{lower}(d)\right) + S_{lower}(d), \tag{11}$$

where $S_{upper}(d)$ and $S_{lower}(d)$ are the lower and upper limits of the characteristic interval of the dth dimension, $S(d)$ is the dth dimension of the superior population, $m(d)$is the pre-mutation gene of the d-dimension, and *rand* is a number generated randomly from 0 to 1. And then, M is the chromosome after the pre-mutation, and X are compared by the greedy rule. If M is better, then the variation is executed and the pre-mutation of the next dimension continues; otherwise, it is not mutated. Until all dimensions have performed the process, X is an excellent individual who has completed directional vertical and partial development. Its mathematical formula can be expressed as **Eq. 12**:

$$X = M, if\ fit(M) < fit(X). \tag{12}$$

The Uniform Mutation Based on the Interval of Opposite Features

In order to obtain new genetic information without destroying the diversity of the population, the opposite feature intervals are utilized as the range of variation to carry out by **Eqs. 13, 14** in UMOFI. What is more, UMOFI controls Pm:

$$\left[O_{L,lower}(d), O_{L,upper}(d)\right] = \{X_{lower}(d), min[S(d)]\}, \tag{13}$$

$$\left[O_{R,lower}(d), O_{R,upper}(d)\right] = \left\{max[S(d)], X_{upper}(d)\right\}, \tag{14}$$

where $O_{L,lower}(d)$ and $O_{R,lower}(d)$ represent the lower limits of opposite feature intervals on the left and right, respectively.

CONSTRAINTS HANDLING METHOD

Ramp Rate Limits Handling

$$P_{t,i}^{min} = \begin{cases} P_i^{min}, & if\ t = 1 \\ \max\left(P_i^{min}, P_{t-1,i} - DR_i\right), & otherwise \end{cases}, \tag{15}$$

TABLE 10 | The optimal solution information for **Case I**: only 5-unit power system.

Hour	Unit					PL	PD	$\sum_i^N p_{i,t} - PL(t) - PD(t)$
	U1	U2	U3	U4	U5			
1	19.90517	98.98362	30.10853	125.029	139.7919	3.818258	410	1.42E-08
2	10.00983	98.45674	65.94291	124.9306	139.7901	4.130096	435	1.81E-08
3	10.51126	98.57997	105.9382	124.923	139.8295	4.781907	475	–5.9E-08
4	10.05254	98.58816	112.6764	174.9155	139.7815	6.013984	530	–6.5E-09
5	10	93.02795	112.387	209.5877	139.7542	6.756857	558	–4.2E-08
6	10	98.54863	112.6755	209.803	184.9645	7.991613	608	7.69E-09
7	10.00743	72.68313	112.4768	209.7585	229.5348	8.460606	626	4.1E-08
8	12.5504	98.53759	112.7754	209.8724	229.5223	9.258051	654	–4.3E-09
9	42.52659	105.322	112.96	209.8684	229.5219	10.19897	690	–5.6E-08
10	64.084	98.52378	112.5764	209.7966	229.5792	10.55995	704	3.17E-08
11	74.99511	103.7247	112.9192	209.8796	229.5243	11.04292	720	–2.8E-08
12	74.97861	124.6893	112.7119	209.8145	229.5255	11.7197	740	–1.1E-07
13	64.15387	98.45896	112.6127	209.8146	229.5193	10.55947	704	7.22E-08
14	48.90734	99.15647	112.747	209.8286	229.5314	10.17071	690	1.3E-07
15	18.90759	98.4442	112.7402	203.6813	229.4421	9.215378	654	–2.3E-08
16	10	82.09235	112.1867	153.6813	229.2425	7.202832	580	1.44E-08
17	10	88.0169	112.6147	124.7447	229.3068	6.683105	558	–3.8E-08
18	10.03469	98.57471	112.7078	165.1254	229.5079	7.950548	608	1.77E-08
19	12.33464	98.58259	112.9116	209.8996	229.5301	9.258485	654	–9.5E-08
20	42.33464	120.1073	112.8268	209.8358	229.5534	10.65802	704	–1.7E-08
21	39.37357	98.50906	112.6148	209.8184	229.5862	9.901916	680	2.53E-09
22	10.02334	98.59234	112.6765	162.0365	229.5421	7.87072	605	–6.5E-08
23	10	98.56103	112.6654	124.9046	186.7749	5.905862	527	1.61E-08
24	10	80.24887	112.669	124.9133	139.6562	4.48748	463	3.77E-09

Total fuel cost is 43030.079 $; total violate is 9.1507e-7 MW.

TABLE 11 | The optimal solution information for **Case II**: only 10-unit power system.

Hour	Unit										PL	PD	$\sum_i^N p_{i,t} - PL(t) - PD(t)$
	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10			
1	150.00	135.01	74.05	120.75	173.37	122.53	129.87	119.98	20.00	10.00	19.57	1036	4.82E-8
2	150.00	135.00	101.75	121.46	222.41	122.32	129.47	120.00	20.00	10.00	22.41	1110	–2.2E-08
3	150.10	135.04	179.26	132.33	223.78	126.35	129.77	119.97	50.00	40.00	28.59	1258	2.82E-8
4	150.00	135.00	249.46	180.07	223.43	159.63	129.90	119.85	51.46	42.73	35.53	1406	8.76E-9
5	150.00	135.00	279.31	229.95	224.71	155.75	129.90	120.00	51.42	43.40	39.45	1480	5.09E-9
6	150.03	135.04	335.19	279.95	243.00	159.24	129.78	120.00	79.88	43.98	48.08	1628	3.82E-8
7	150.40	187.20	340.00	300.00	243.00	160.00	130.00	120.00	80.00	44.41	53.01	1702	–6.5E-08
8	177.76	228.93	340.00	300.00	243.00	159.98	130.00	120.00	80.00	54.73	58.41	1776	–1.9E-08
9	257.76	308.79	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	70.55	1924	–0.00062
10	289.26	384.33	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	79.58	2022	–0.00058
11	369.00	396.86	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	87.86	2106	–0.00054
12	378.55	435.90	340.00	300.00	243.00	160.00	129.99	120.00	80.00	55.00	92.44	2150	–1.2E-08
13	336.70	391.72	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	84.42	2072	–0.00097
14	256.77	311.72	339.12	300.00	242.57	160.00	130.00	120.00	80.00	54.40	70.58	1924	1.41E-7
15	177.43	231.74	340.00	300.00	243.00	159.95	129.95	119.99	80.00	52.36	58.43	1776	–2.4E-09
16	150.00	151.88	297.25	252.92	240.53	160.00	129.91	120.00	51.51	43.67	43.68	1554	–1.1E-07
17	150.03	135.00	289.92	243.32	222.60	133.67	129.38	119.99	52.28	43.45	39.63	1480	–1.7E-07
18	150.00	149.67	308.27	292.70	242.41	159.93	129.64	120.00	80.00	43.43	48.06	1628	–6.2E-08
19	229.63	229.28	299.76	299.98	242.92	160.00	129.70	120.00	79.99	43.54	58.80	1776	–4.8E-08
20	309.62	309.28	340.00	300.00	243.00	160.00	130.00	120.00	80.00	54.99	74.89	1972	1.94E-9
21	263.68	302.92	339.99	299.99	243.00	160.00	129.99	120.00	80.00	54.99	70.55	1924	1.12E-7
22	183.68	223.13	291.14	252.01	222.96	159.97	129.93	120.00	50.56	43.46	48.85	1628	3.13E-8
23	150.00	143.13	211.14	202.43	222.71	120.71	129.59	119.38	51.24	13.65	31.99	1332	–3.5E-08
24	150.09	135.04	175.56	175.83	172.75	118.98	129.73	120.00	21.24	10.00	25.22	1184	6.15E-8

Total fuel cost is 2464270.102 $; total violate is 0.0027067 MW.

TABLE 12 | Statistics of 30 trials in **Scenario B**, units with disorderly PEVs, and **Scenario C**, units without orderly PEVs with different λ.

λ	Objectives	Case III: 5 units integrating PEVs				Case IV: 10 units integrating PEVs			
		Min	Mean	Max	Std	Min	Mean	Max	Std
0	f1	4.288E+04	4.320E+04	4.350E+04	2.352E+02	2.455E+06	2.456E+06	2.457E+06	8.181E+02
	f2	2.894E+04	3.950E+04	**6.376E+04**	**1.538E+04**	3.052E+05	3.215E+05	**3.321E+05**	**1.052E+04**
1	f1	**4.328E+04**	**4.331E+04**	4.339E+04	1.402E+01	**2.463E+06**	2.464E+06	**2.465E+06**	**3.595E+02**
	f2	**3.894E+03**	4.096E+03	**4.227E+03**	**8.238E+01**	**2.623E+05**	2.705E+05	2.768E+05	**2.704E+03**
2	f1	4.329E+04	4.331E+04	**4.336E+04**	**1.376E+01**	2.463E+06	2.464E+06	2.466E+06	1.499E+03
	f2	3.900E+03	4.122E+03	4.516E+03	2.116E+02	2.625E+05	2.701E+05	2.773E+05	6.603E+03
3	f1	4.329E+04	4.331E+04	4.343E+04	1.770E+01	2.463E+06	2.464E+06	2.466E+06	9.740E+02
	f2	3.894E+03	**4.093E+03**	4.290E+03	1.983E+02	2.625E+05	**2.686E+05**	**2.758E+05**	3.670E+03

Bold digits are the best statistical performance measures of various algorithms.

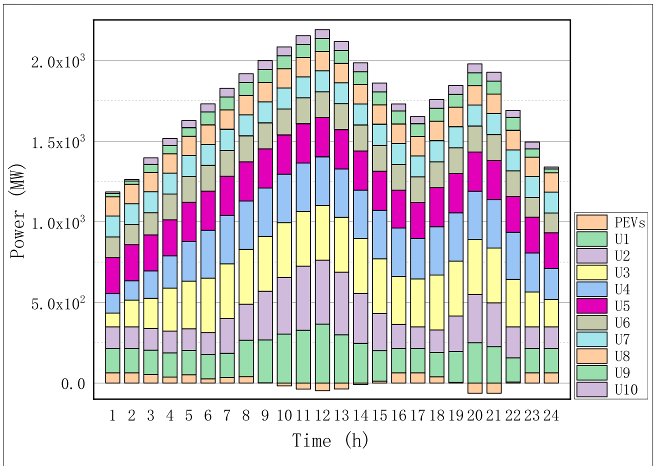

FIGURE 5 | The power output of the 10 units with an orderly PEVs power system.

$$P_{t,i}^{max} = \begin{cases} P_i^{max}, & if\ t = 1 \\ \max\left(P_i^{max}, P_{t-1,i} + UR_i\right), & otherwise \end{cases}, \tag{16}$$

where $P_{t,i}^{min}$ and $P_{t,i}^{max}$ stand for the new lower and upper bounds, considering simultaneously the ramp rate constraint and the capacity limit of the ith unit for the tth time, as **Eqs 15, 16**. Then, if $P_{t,i}$ is beyond its new bound, it will be limited to the bound. Namely, $P_{t,i}$ is repaired by **Eq. 17**. It is simpler and more efficient than the traditional penalty function method:

$$P_{t,i} = \begin{cases} P_{t,i}^{min}, & if\ P_{t,i} \le P_{t,i}^{min} \\ P_{t,i}^{max} & if\ P_{t,i} \ge P_{t,i}^{max} \end{cases}. \tag{17}$$

The Power Balance Constraint Handling

Considering the network transmission loss, the power balance constraint is the most difficult to repair among all constraints. This study proposes a simple and efficient repair technology. The overall process is designed as two stages: firstly, rough adjustment can rapidly reduce the violation and then enter the second fine adjustment stage to eliminate the violation; the detailed steps are described as **Steps 1–4**.

Step 1. Set the set A = $\{1, 2, 3, \ldots, N-1, N\}$, and select randomly a unit r from A and roughly adjust the output by **Eq. 18**:

$$P_{t,r} = P_{t,r} - Vio(t), \tag{18}$$

where $Vio(t)$ is the violation of the power balance constraint at t time. And if $P_{t,r}$ does not go beyond its new boundary, it is thought that $Vio(t)$ is so small to repair by the fine stage and thus go to the next step and reset A = $\{1, 2, 3, \ldots, N-1, N\}$. Let $k = 1$; otherwise, remove r unit from A. If A is an empty set, end repair; otherwise, repeat **Step 1**.

Step 2. Handling the power balance constraint can be converted as solving a quadratic equation, and the output of the unit is solved by **Eq. 19**. Here, two cases are discussed as follows:

Otherwise, it can be converted as solving a quadratic equation, and the output of the unit is solved by **Eq. 19**. Here, two cases are discussed as follows:

$$B_{kk}P_{t,k}^2 + \left(2\sum\nolimits_{i\in A,\neq k} B_{ki}P_{t,i} - 1\right)P_{t,k} + \left(P_{D,t} + \sum\nolimits_{i\in A,\neq k} \sum\nolimits_{j\in A,\neq k} P_{t,j}B_{ij}P_{t,j} - \sum\nolimits_{i\in A,\neq k} P_{t,i}\right) = 0. \tag{19}$$

Let a $= B_{kk}$, b $= 2\sum\limits_{i\in A,\neq k} B_{ki}p_{t,i} - 1$, c $= P_{D,t} + \sum\limits_{i\in A,\neq k}\sum\limits_{j\in A,\neq k} P_{t,i}B_{ij}P_{t,j} - \sum\limits_{i\in A,\neq k} P_{t,i}$; then, if existing, the roots are calculated by Sol_1 and Sol_2 = $\frac{-b\pm\sqrt{b^2-4ac}}{2a}$ when $a\neq 0$, or $Sol_3 = -\frac{c}{b}$ when $a = 0$ and $b \neq 0$.

Case 1. If no solution, let = $k + 1$; if $k < N$, repeat **Step 2**; and otherwise, end repair.

Case 2. If there are solutions, checking whether they satisfy the other constraints. If satisfied, let $P_{t,k}$ be equal to any, and end; if only a solution is satisfied, let $P_{t,k}$ be equal to the solution. Otherwise, let $k = k + 1$; if $k < N$, repeat **Step 2**; and otherwise, end repair.

Step 1 can rapidly decrease violation of the equality constraint associated with power balances, and Step 2 further decrease or eliminate the violation by fine adjustment as well as solving. And finally, if still infeasible, the feasible-rule (Yuan et al., 2009) is used to strictly screen the feasible solutions of the population.

THE IMPLEMENTATION OF GENETIC ALGORITHM DIMENSION MUTATION BASED ON FEATURE INTERVALS FOR DYNAMIC ECONOMIC DISPATCH WITH PLUG-IN ELECTRIC VEHICLES

The Implementation of GADMFI on DED integrating PEVs is a process that effectively combines heuristic algorithms and

TABLE 13 | The optimal solution information for **Case V**: the 5 units with orderly PEVs.

Hour	λ = 1						PL	PD	$\sum_i^N p_{i,t} - PL(t) - PD(t)$
	U1	U2	U3	U4	U5	PEVs			
1	10.00763	84.77189	30	124.7421	229.4654	63.75	5.236962	410	–3.2E-08
2	18.65022	98.75426	30.00133	124.9503	229.519	61.15883	5.716354	435	1.45E-08
3	10.05842	98.51895	69.26687	124.9237	229.5208	51.13098	6.157715	475	–3.4E-08
4	10.00006	96.48811	109.2515	124.7849	229.2666	32.96012	6.831012	530	–5.2E-08
5	26.84449	98.45897	112.7252	125.2689	229.5367	27.51259	7.321632	558	–1.2E-08
6	10.88256	98.61089	112.6623	175.2255	229.511	10.6552	8.237037	608	–7.3E-08
7	10.00281	82.6314	112.5842	209.8485	229.4198	9.75066	8.735997	626	8.33E-08
8	14.4919	98.65622	113.1529	209.8373	229.5681	2.391691	9.314721	654	2.54E-08
9	39.74695	98.64638	112.6786	209.7913	229.5965	-9.45676	9.916527	690	1.68E-08
10	52.80744	98.52219	112.6839	209.7079	229.5185	-11.0095	10.2495	704	3.49E-08
11	64.41219	98.54479	112.6956	209.8169	229.4505	-15.6493	10.56924	720	–2.2E-08
12	74.99786	99.40575	112.8564	209.9872	229.594	-24.0698	10.91099	740	–1.4E-08
13	58.15658	98.48497	112.6774	209.8286	229.5078	-5.74099	10.39632	704	–3.2E-08
14	49.96416	98.50633	112.6383	209.8122	229.5655	0.309664	10.17684	690	–6.6E-08
15	29.52558	98.59672	112.789	209.8117	229.5462	16.60822	9.660985	654	–2.6E-08
16	10.0064	78.63041	112.6348	209.7877	229.4994	51.93233	8.62642	580	8.31E-08
17	10.0067	64.18732	112.5355	209.8336	229.4745	59.80215	8.235522	558	–6.8E-08
18	10	85.69395	112.5494	209.5248	229.3593	30.31734	8.810133	608	1.26E-08
19	13.67358	98.79633	112.7959	209.8289	229.5454	1.348831	9.291236	654	7.73E-08
20	33.30203	98.55797	112.6687	209.8001	229.5211	-29.8994	9.749276	704	–1.7E-08
21	14.46368	98.69659	112.7496	209.8921	229.5302	-23.9748	9.306891	680	5.86E-09
22	10.00007	98.53408	112.6287	169.9267	229.5327	7.548168	8.074086	605	1.82E-08
23	10.00286	98.09282	111.3995	124.8193	229.4811	39.87367	6.921881	527	–5E-09
24	10.00138	97.04352	71.60713	124.8649	229.3884	63.75	6.155354	463	1.7E-08

Total fuel cost is 43279.905 $

TABLE 14 | The optimal solution information for **Case VI**: the 10 units with orderly PEVs.

Hour	λ = 1											PL	PD	$\sum_i^N p_{i,t} - PL(t) - PD(t)$
	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	PEVs			
1	150.00	135.00	85.22	120.79	222.83	128.65	129.43	119.86	20.00	10.00	63.75	22.04	1036.00	3.39E-10
2	150.10	135.01	165.22	120.50	223.57	124.32	129.98	119.95	20.01	10.00	63.75	24.93	1110.00	–1.3E-07
3	150.00	135.00	186.47	170.50	223.01	137.74	130.00	119.90	50.01	40.00	53.71	30.93	1258.00	–1.9E-08
4	150.00	135.00	266.43	200.75	223.44	159.76	129.69	119.93	51.95	43.34	36.85	37.44	1406.00	–4.6E-08
5	150.00	135.06	295.13	246.47	242.15	159.88	129.99	119.65	52.44	43.38	51.84	42.32	1480.00	–6.2E-08
6	150.08	136.51	337.95	296.47	243.00	160.00	129.87	120.00	80.00	49.93	26.11	49.70	1628.00	–4.3E-08
7	150.09	215.08	340.00	300.00	243.00	160.00	129.96	120.00	80.00	53.69	34.35	55.46	1702.00	–1.4E-08
8	226.15	223.24	340.00	300.00	243.00	160.00	130.00	120.00	80.00	54.64	39.53	61.50	1776.00	7.56E-09
9	265.89	302.17	340.00	300.00	243.00	160.00	130.00	120.00	80.00	54.97	1.36	70.67	1924.00	2.23E-08
10	303.14	351.76	339.86	300.00	243.00	160.00	129.96	119.99	80.00	54.75	-17.42	77.89	2022.00	3.45E-08
11	327.11	397.34	340.00	299.98	243.00	160.00	130.00	120.00	80.00	55.00	-37.65	84.06	2106.00	–2.3E-08
12	365.14	396.67	340.00	300.00	242.99	160.00	130.00	120.00	80.00	55.00	-47.68	87.48	2150.00	4.05E-09
13	299.49	388.25	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	-37.08	80.81	2072.00	4.04E-08
14	246.54	309.50	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	-9.67	69.71	1924.00	–2.4E-08
15	189.33	229.80	340.00	299.94	243.00	160.00	130.00	120.00	80.00	55.00	11.77	59.30	1776.00	–8.5E-09
16	150.11	150.00	297.37	300.00	234.75	159.81	129.82	119.99	80.00	43.31	63.73	47.43	1554.00	–2.8E-08
17	150.01	135.00	297.03	251.23	222.21	160.00	129.62	119.74	78.44	43.47	63.75	43.01	1480.00	4.44E-08
18	150.24	140.06	339.81	299.84	243.00	160.00	130.00	120.00	80.00	54.79	39.20	50.55	1628.00	5.71E-08
19	191.22	220.06	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	4.53	58.74	1776.00	–1.2E-09
20	250.38	299.22	339.90	300.00	243.00	160.00	129.95	119.99	79.99	55.00	-63.74	69.18	1972.00	–1.8E-08
21	225.48	271.92	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	-63.72	65.11	1924.00	2.32E-08
22	150.00	192.17	294.84	291.29	222.82	158.69	129.90	120.00	80.00	43.28	6.27	48.71	1628.00	1.03E-07
23	150.04	135.00	216.12	241.29	222.60	122.22	129.59	120.00	52.01	41.99	63.73	35.14	1332.00	4.18E-08
24	150.00	135.03	169.77	191.29	222.40	122.38	129.71	120.00	23.22	11.99	63.75	28.03	1184.00	1.2E-08

Total fuel cost is 2462765.9108 $

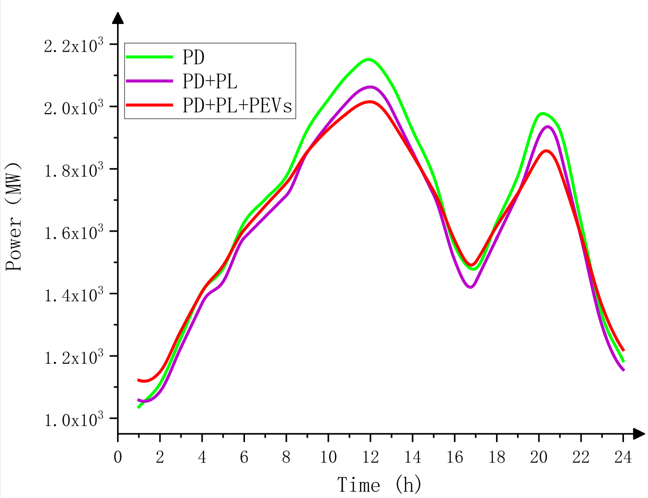

FIGURE 6 | The demand considering the loss and PEVs for **Case VI**.

constraints handling methods and optimal mathematical model. The overall framework is described in **Figure 2**.

Step 1. The initialization of the population consisted of NP individuals. Each is expressed as a matrix of T rows and (N+1) columns by **Eq. 20**:

$$[P, P_{PEV}] = \begin{bmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,N} & P_{PEV,1} \\ P_{2,1} & P_{2,2} & \cdots & P_{2,N} & P_{PEV,2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{T,1} & P_{T,2} & \cdots & P_{T,N} & P_{PEV,T} \end{bmatrix}, \tag{20}$$

and all variables of $P_{t,i}$ are initialized by **Eq. 21**:

$$\begin{cases} P_{t,i} = P_{t,i}^{min} + rand \times \left(P_{t,i}^{max} - P_{t,i}^{min}\right), & t = 1, 2, \ldots, T, n = 1, 2, \ldots, N \\ P_{PEV,t} = P_{PEV,disc}^{max} + rand \times \left(P_{PEV,char}^{max} - P_{PEV,disc}^{max}\right), & t = 1, 2, \ldots, T \end{cases}. \tag{21}$$

Step 2. Checking individuals' feasibility. If feasible, go to the next step; otherwise, repair by constraint handling technology and go to the next step.

Step 3. Evaluate their fitness by **Eq. 22**, in which the objective functions $f1$ and $f2$ are combined into f by a weighting factor λ and update the optimal individual. If FEs are equal to MaxFEs, output the best solution; otherwise, go to the next step:

$$f_1 = f_1 + \lambda f_2. \tag{22}$$

Step 4. Update individuals in the population via GADMFI, and go to Step 2

EXPERIMENTAL RESULTS AND ANALYSIS

Validation of the Performance of Genetic Algorithm Dimension Mutation Based on Feature Intervals

In order to validate the performance of the algorithm, a set of benchmark functions are selected from Civicioglu (2013), which are shown in **Table 3**, including low-dimensional, multi-dimensional, unimodal (U), multimodal (M), separable (S), and non-separable (N) functions. Advanced meta-heuristics are employed for qualitative and quantitative comparison using the benchmark problems. They are ABC, the grey wolf optimizer (GWO) (Mirjalili et al., 2014), the whale optimization algorithm (WOA) (Mirjalili and Lewis, 2016), the bat algorithm with triangle-flipping strategy (BA-HTFS) (Cai et al., 2017), and Hybrid DE-WOA Algorithm (DEWOA) (Wang et al., 2019c). In addition, the Wilcoxon Signed-Rank Test was used for pairwise comparisons, with the statistical significance value $\alpha = 0.05$. The null hypothesis H_0 for this test is as follows: there is no difference between the median of the solution between the two algorithms. The experimental computer is Intel(R) Core (TM) i9-10900F CPU @ 3.7GHZ and its RAM is 16.0 GHz.

Parameters Setting

MaXFEs for 30, 50, and 100-dimensional benchmarks are set as $D{*}10000$, which means that when FEs reach MaxFEs, the optimization algorithms will be terminated. In addition, private parameters of methods taking part in the comparison from the corresponding references are set as in **Table 4**.

Performance Analysis

The performance of an optimization method should be evaluated in convergence accuracy, speed, and robustness. Therefore, the mean, best, and standard deviation of the objective values based on 30 independent runs from three dimensions 30, 50, and 100 are utilized in the quantitative analysis in **Tables 5–7**, and several typical qualitative graphs are shown in **Figure 3**.

In **Tables 5–7**, the optimal values of the indicators including "mean", "Std", and "best" are bold among the six comparative algorithms. Winner = 1, 0, and −1 mean GADMFI is obviously superior, equivalent, and inferior to other methods with $\alpha = 0.05$. "NA" refers to not available. First of all, it can be seen that, as the dimension size increases, the performance of the algorithm does not change greatly. It can be observed from best that except for F6, none performs better than the proposed GADMFI in terms of the global search. This is attributed to the collaboration of mechanisms of the proposed algorithm. From mean and Std, it can be seen that GADMFI is the most stable and GWO, secondly, which benefits from DMFI has the ability to be directed fine-grained to develop near the current optimal population. From Runtime, the running time of GADMFI is the shortest in most problems, and it comes from the algorithm maintaining the traditional framework of RCGAs. As can be obtained from Winner, only in F6, problem is inferior to ABC, and F6 of 100 dimensions is inferior to GWO, but the difference is very small.

In **Figure 3**, evolution curves of six functions of 100 dimensions are shown. As can be seen, the convergence curve of the GADMFI shows superior exploration and exploitation abilities. In F1, the curve presents a straight line, which shows that when dealing with unimodal problems, the proposed DMFI has the potential to explore a promising area. In F2–F6 with multiple local minima, it stays at approximately constant speed until it converges, which again confirms the role of DMFI.

In summary, the proposed GADMFI has outstanding performance for the benchmark problems.

Simulation Results and Discussion on Dynamic Economic Dispatch With Plug-In Electric Vehicles Problem

In this section, in order to verify the reliability of the proposed algorithm and constraint handling method, three scenarios and six cases are considered, as described in **Table 8**, as follows: **Scenario A:** only units, **Scenario B:** units with disorderly PEVs, **Scenario C:** units with orderly PEVs. **Case I:** only 5 units, **Case II:** only 10 units, **Case III:** 5 units with disorderly PEVs, **Case IV:**10 units with disorderly PEVs, **Case V:** 5 units with orderly PEVs, and **Case VI:** 10 units with orderly PEVs. It is worth noting that PEV and transmission loss are all considered in all cases. The population size of the algorithm is 100, P_c is 0.7, and P_m is 0.3. MaxFEs is set to D*10000, D represents the dimension of the decision variable; that is, for **Cases I-II**; D = N*24, for **Cases III-VI**; D = (N + 1)*24, and in the study, T is set as 24. In addition, in order to avoid contingency, each case is run independently 30 times. The optimization is implemented in the MATLAB ® 2019b on an Intel(R) Core (TM) i9-10900K CPU @ 3.70 GHz with RAM is 16.0 GHz personal computer.

Scenario A: Only Units Without Plug-In Electric Vehicles

The data of **Cases I-II** are derived from (Basu, 2008; Mohammadi-ivatloo et al., 2012; Qian et al., 2020), including predicted power demand (PD), unit information, and B coefficients in transmission loss. Fuel costs and constraint violations are counted in **Table 9** and are compared with the current popular literature, including the new enhanced harmony search (NEHS), the artificial immune system (AIS), the hybrid DE and sequential quadratic programming (DE-SQP), the hybrid PSO and sequential quadratic programming (PSO-SQP), the efficient fitness-based differential evolution algorithm (EFDE), the hybrid seeker optimization algorithm (SOA) and sequential quadratic programming method (SOA-SQP), the simulated annealing (SA), a hybrid genetic algorithm and bacterial foraging approach (HCRO), and the improved bacterial foraging algorithm (IBFA).

In **Table 9**, the minimum, average, maximum, and standard deviation of fuel cost of 30 independent trials are presented, as well as the number of violations of unit ramp rate limits and the power balance constraint. Here, the minimum value of each statistic is bold in black font. the constraint violation amount is greater than one and is bold in red font, indicating that the solution is not feasible. For constrained optimization problems, judging the quality of a solution must first meet the conditions of a feasible solution and then evaluate the value of the objective function. Obviously, in the two cases, the proposed constraint processing technology can efficiently repair infeasible solutions. In addition, compared with other algorithms in the literature, except for the minimum value in **Case II**, it is slightly inferior to NEHS, and GADMFI shows extremely high superiority.

In order to clearly show the output of each unit, the stacked histogram of 10 units is drawn. The impact of the ramp rate limit on the output of the unit can be clearly seen in **Figure 4**; that is, the power difference between two adjacent moments within the smaller ranges and the optimal solution of 30 trails for **Cases I-II** are shown in **Tables 10-11**, as well as the transmission loss (PL) and the amount of violation.

Scenarios B and C: Units With Plug-In Electric Vehicles

A total of 50,000 PEVs are assumed to be integrated into the 5- and 10-unit power systems, and the daily average traveling distance and expected power demand of a PEV are 32.88 miles and 8.22 kWh, respectively (Saber and Venayagamoorthy, 2011). The total power necessity for PEVs is 411 MW and is expected to be met by power generation. The state of charge SOC is 50%; the number of PEVs that can provide V2G/G2V service is 50000/36125; the average battery capacity is 15 kWh; the charging efficiency and the discharge efficiency are both 85%; and available PEVs are 20% (Yang et al., 2017b). In that way, the maximum discharge power $P_{PEV,disc}^{max}$ = −50, 000*15KWh*85%*20%*50% = −63.75KW, and the maximum charge power $P_{PEV,char}^{max}$ = 36125*15KWh/85%*20%*50% = +63.75KW. λ = 0 means that electric vehicles are not managed; that is, for **Cases III-IV**, λ> 0 means that electric vehicles are orderly managed, that is, **Cases V-VI**. Comparing **scenarios B** and **C** and determining λ, f_1, and f_2 in 30 trials are counted in **Table 12**.

In the column of Max and Std, the larger value is marked in red, indicating that the quality of the solution is poor and unstable, which reflects that the grid fluctuates greatly when electric vehicles are not managed, and further, in λ = 1, 2, 3, when the balance effect is best for f_1 and f_2, λ = 1; therefore, in **Scenario C**, λ is set to 1, and the decision variables are listed in **Tables 13** and **14** for **Cases V-VI**. The output of units and PEVs for **Case VI** is drawn in **Figure 5**.

From **Eqs 15**, **16**, plug-in electric vehicles are effectively managed by the proposed strategy f_2, that is, during peak demand periods, discharge through G2V and charge during trough periods by V2G. In addition, from the perspective of PL at various times, it is larger than or close to P_{PEV}, so transmission loss should be considered in DED; otherwise, a few decisions may cause mistakes. In order to describe this effect more clearly, PD and PL and P_{PEV} are plotted in **Figure 6**. It can be seen that the magnitude of the loss is close to the maximum output power of electric vehicles, and the management strategy of electric vehicles has been proved to be effective; that is, it plays the role of cutting peaks and filling valleys.

CONCLUSION

In view of the impact of plug-in electric vehicles on the power grid, and the complexity of dynamic economic dispatch considering the valve-point effect and transmission loss, this study integrates PEVs into DED and proposes a novel genetic algorithm: GADMFI, a simple and yet efficient constraint handling method aiming at power balance constraints. In three scenarios with two scales, only units, units with disorderly PEVs, and units with orderly PEVs, a horizontal and vertical comparison was carried out. The results show that GADMFI has an excellent performance in dealing with multi-modal, high-dimensional, and large-scale problems such as DED. At the same time, the proposed constraint handling method guarantees the feasibility of solutions and the design of target f_2 had achieved the effect of adaptive peak clipping and valley filling.

DATA AVAILABILITY STATEMENT

The author selected the following statement: the data analyzed in this study is subject to the following licenses/restrictions: commercial data. Requests to access these datasets should be directed to zl.yang@siat.ac.cn.

AUTHOR CONTRIBUTIONS

WY: writing-original draft preparation, data curation. ZP: writing-original draft preparation, investigation. WF: writing-original draft preparation. MM: supervision, visualization, writing-reviewing and Editing.

FUNDING

This work was financially supported by the National Natural Science Foundation of China (nos. 52077213, 620033326), the Scientific and Technological Project of Henan Province (nos. 202102110281, 202102110282), the Program for Innovative Research Team (in Science and Technology) in University of Henan Province (no. 20IRTSTHN016).

REFERENCES

Abdelaziz, A. Y., Kamh, M. Z., Mekhamer, S. F., and Badr, M. A. L. (2008). A Hybrid HNN-QP Approach for Dynamic Economic Dispatch Problem. *Electric Power Syst. Res.* 78, 1784–1788. doi:10.1016/j.epsr.2008.03.011

Akopov, A. S., Beklaryan, L. A., Thakur, M., and Verma, B. D. (2019). Parallel Multi-Agent Real-Coded Genetic Algorithm for Large-Scale Black-Box Single-Objective Optimisation. *Knowledge-Based Syst.* 174, 103–122. doi:10.1016/j.knosys.2019.03.003

Ali, M. Z., Awad, N. H., Suganthan, P. N., Shatnawi, A. M., and Reynolds, R. G. (2018). An Improved Class of Real-Coded Genetic Algorithms for Numerical Optimization☆. *Neurocomputing* 275, 155–166. doi:10.1016/j.neucom.2017.05.054

Basu, M. (2011). Artificial Immune System for Dynamic Economic Dispatch. *Int. J. Electr. Power Energ. Syst.* 33, 131–136. doi:10.1016/j.ijepes.2010.06.019

Basu, M. (2008). Dynamic Economic Emission Dispatch Using Nondominated Sorting Genetic Algorithm-II. *Int. J. Electr. Power Energ. Syst.* 30, 140–149. doi:10.1016/j.ijepes.2007.06.009

Behera, S., Behera, S., and Barisal, A. K. (2019). "Ieee. Dynamic Economic Load Dispatch with Plug-In Electric Vehicles Using Social Spider Algorithm," in Proceedings of the 2019 3rd International Conference on Computing Methodologies and Communication (New York: IEEE), 489–494.

Cai, X., Wang, H., Cui, Z., Cai, J., Xue, Y., and Wang, L. (2017). Bat Algorithm with triangle-flipping Strategy for Numerical Optimization. *Int. J. Mach. Learn. Cyber.* 9, 199–215. doi:10.1007/s13042-017-0739-8

Chakraborty, P., Roy, G. G., Panigrahi, B. K., Bansal, R. C., and Mohapatra, A. (2012). Dynamic Economic Dispatch Using harmony Search Algorithm with Modified Differential Mutation Operator. *Electr. Eng.* 94, 197–205. doi:10.1007/s00202-011-0230-6

Chuang, Y.-C., Chen, C.-T., and Hwang, C. (2015). A Real-Coded Genetic Algorithm with a Direction-Based Crossover Operator. *Inf. Sci.* 305, 320–348. doi:10.1016/j.ins.2015.01.026

Chuang, Y.-C., Chen, C.-T., and Hwang, C. (2016). A Simple and Efficient Real-Coded Genetic Algorithm for Constrained Optimization. *Appl. Soft Comput.* 38, 87–105. doi:10.1016/j.asoc.2015.09.036

Civicioglu, P. (2013). Backtracking Search Optimization Algorithm for Numerical Optimization Problems. *Appl. Maths. Comput.* 219, 8121–8144. doi:10.1016/j.amc.2013.02.017

D'Angelo, G., and Palmieri, F. G. G. A. (2021). A Modified Genetic Algorithm with Gradient-Based Local Search for Solving Constrained Optimization Problems. *Inf. Sci.* 547, 136–162. doi:10.1016/j.ins.2020.08.040

Deep, K., and Thakur, M. (2007). A New Crossover Operator for Real Coded Genetic Algorithms. *Appl. Maths. Comput.* 188, 895–911. doi:10.1016/j.amc.2006.10.047

Elaiw, A. M., Xia, X., and Shehata, A. M. (2013). Hybrid DE-SQP and Hybrid PSO-SQP Methods for Solving Dynamic Economic Emission Dispatch Problem with Valve-point Effects. *Electric Power Syst. Res.* 103, 192–200. doi:10.1016/j.epsr.2013.05.015

Elattar, E. E. (2015). A Hybrid Genetic Algorithm and Bacterial Foraging Approach for Dynamic Economic Dispatch Problem. *Int. J. Electr. Power Energ. Syst.* 69, 18–26. doi:10.1016/j.ijepes.2014.12.091

Fang, N., Zhou, J., Zhang, R., Liu, Y., and Zhang, Y. (2014). A Hybrid of Real Coded Genetic Algorithm and Artificial Fish Swarm Algorithm for Short-Term Optimal Hydrothermal Scheduling. *Int. J. Electr. Power Energ. Syst.* 62, 617–629. doi:10.1016/j.ijepes.2014.05.017

Hemamalini, S., and Simon, S. P. (2011). Dynamic Economic Dispatch Using Artificial Bee colony Algorithm for Units with Valve-point Effect. *Euro. Trans. Electr. Power* 21, 70–81. doi:10.1002/etep.413

Hemamalini, S., and Simon, S. P. (2011). Dynamic Economic Dispatch Using Artificial Immune System for Units with Valve-point Effect. *Int. J. Electr. Power Energ. Syst.* 33, 868–874. doi:10.1016/j.ijepes.2010.12.017

Iyer, V. H., Mahesh, S., Malpani, R., Sapre, M., and Kulkarni, A. J. (2019). Adaptive Range Genetic Algorithm: A Hybrid Optimization Approach and its Application in the Design and Economic Optimization of Shell-And-Tube Heat Exchanger. *Eng. Appl. Artif. Intelligence* 85, 444–461. doi:10.1016/j.engappai.2019.07.001

Kwak, N. S., and Lee, J. (2016). An Enhancement of Selection and Crossover Operations in Real-Coded Genetic Algorithm for Large-Dimensionality Optimization. *J. Mech. Sci. Technol.* 30, 237–247. doi:10.1007/s12206-015-1227-2

Li, Z., Zou, D., and Kong, Z. (2019). A harmony Search Variant and a Useful Constraint Handling Method for the Dynamic Economic Emission Dispatch Problems Considering Transmission Loss. *Eng. Appl. Artif. Intelligence* 84, 18–40. doi:10.1016/j.engappai.2019.05.005

Mei, P., Wu, L., Zhang, H., and Liu, Z. (2019). A Hybrid Multi-Objective Crisscross Optimization for Dynamic Economic/Emission Dispatch Considering Plug-In Electric Vehicles Penetration. *Energies* 12, 3847. doi:10.3390/en12203847

Mellal, M. A., and Williams, E. J. (2020). Cuckoo Optimization Algorithm with Penalty Function and Binary Approach for Combined Heat and Power Economic Dispatch Problem. *Energ. Rep.* 6, 2720–2723. doi:10.1016/j.egyr.2020.10.004

Mirjalili, S., and Lewis, A. (2016). The Whale Optimization Algorithm. *Adv. Eng. Softw.* 95, 51–67. doi:10.1016/j.advengsoft.2016.01.008

Mirjalili, S., Mirjalili, S. M., and Lewis, A. (2014). Grey Wolf Optimizer. *Adv. Eng. Softw.* 69, 46–61. doi:10.1016/j.advengsoft.2013.12.007

Mohammadi-ivatloo, B., Rabiee, A., and Ehsan, M. (2012). Time-varying Acceleration Coefficients IPSO for Solving Dynamic Economic Dispatch with Non-smooth Cost Function. *Energ. Convers. Manage.* 56, 175–183. doi:10.1016/j.enconman.2011.12.004

Nakane, T., Lu, X., and Zhang, C. (2020). A Search History-Driven Offspring Generation Method for the Real-Coded Genetic Algorithm. *Comput. Intell. Neurosci.* 2020, 8835852. doi:10.1155/2020/8835852

Naqvi, F. B., Yousaf Shad, M., and Khan, S. (2020). A New Logistic Distribution Based Crossover Operator for Real-Coded Genetic Algorithm. *J. Stat. Comput. Simulation* 91, 817–835. doi:10.1080/00949655.2020.1832093

Niu, Q., Zhang, H., Li, K., and Irwin, G. W. (2014). An Efficient harmony Search with New Pitch Adjustment for Dynamic Economic Dispatch. *Energy* 65, 25–43. doi:10.1016/j.energy.2013.10.085

Pan, S., Jian, J., and Yang, L. (2018). A Hybrid MILP and IPM Approach for Dynamic Economic Dispatch with Valve-point Effects. *Int. J. Electr. Power Energ. Syst.* 97, 290–298. doi:10.1016/j.ijepes.2017.11.004

Pandit, N., Tripathi, A., Tapaswi, S., and Pandit, M. (2012). An Improved Bacterial Foraging Algorithm for Combined Static/dynamic Environmental Economic Dispatch. *Appl. Soft Comput.* 12, 3500–3513. doi:10.1016/j.asoc.2012.06.011

Panigrahi, C. K., Chattopadhyay, P. K., Chakrabarti, R. N., and Basu, M. (2007). Simulated Annealing Technique for Dynamic Economic Dispatch. *Electric Power Components Syst.* 34, 577–586. doi:10.1080/15325000500360843

Picek, S., Jakobovic, D., and Golub, M. (2013). Ieee. On the Recombination Operator in the Real-Coded Genetic Algorithms. *Ieee Congress Evol. Comput. 2013*, 3103–3110.

Picek, S., Jakobovic, D., and Golub, M. (2013). On the Recombination Operator in the Real-Coded Genetic Algorithms. *IEEE Congress Evol. Comput. 2013*, 3103–3110. doi:10.1109/CEC.2013.6557948

Qian, S., Wu, H., and Xu, G. (2020). An Improved Particle Swarm Optimization with Clone Selection Principle for Dynamic Economic Emission Dispatch. *Soft Comput.* 24, 15249–15271. doi:10.1007/s00500-020-04861-4

Ravikumar Pandi, V., and Panigrahi, B. K. (2011). Dynamic Economic Load Dispatch Using Hybrid Swarm Intelligence Based harmony Search Algorithm. *Expert Syst. Appl.* 38, 8509–8514. doi:10.1016/j.eswa.2011.01.050

Saber, A. Y., and Venayagamoorthy, G. K. (2011). Plug-in Vehicles and Renewable Energy Sources for Cost and Emission Reductions. *IEEE Trans. Ind. Electron.* 58, 1229–1238. doi:10.1109/tie.2010.2047828

Sawyerr, B. A., Adewumi, A. O., and Ali, M. M. (2014). Real-coded Genetic Algorithm with Uniform Random Local Search. *Appl. Maths. Comput.* 228, 589–597. doi:10.1016/j.amc.2013.11.097

Sawyerr, B. A., Ali, M. M., and Adewumi, A. O. (2011). A Comparative Study of Some Real-Coded Genetic Algorithms for Unconstrained Global Optimization. *Optimization Methods Softw.* 26, 945–970. doi:10.1080/10556788.2010.491865

Shen, X., Zou, D., Duan, N., and Zhang, Q. (2019). An Efficient Fitness-Based Differential Evolution Algorithm and a Constraint Handling Technique for Dynamic Economic Emission Dispatch. *Energy*, 186. doi:10.1016/j.energy.2019.07.131

Sivasubramani, S., and Swarup, K. S. (2010). Hybrid SOA-SQP Algorithm for Dynamic Economic Dispatch with Valve-point Effects. *Energy* 35, 5031–5036. doi:10.1016/j.energy.2010.08.018

Tang, P.-H., and Tseng, M.-H. (2013). Adaptive Directed Mutation for Real-Coded Genetic Algorithms. *Appl. Soft Comput.* 13, 600–614. doi:10.1016/j.asoc.2012.08.035

Tehzeeb-ul-Hassan, H., Alquthami, T., Butt, S. E., Tahir, M. F., and Mehmood, K. (2020). Short-term Optimal Scheduling of Hydro-thermal Power Plants Using Artificial Bee colony Algorithm. *Energ. Rep.* 6, 984–992. doi:10.1016/j.egyr.2020.04.003

Thakur, M., Meghwani, S. S., and Jalota, H. (2014). A Modified Real Coded Genetic Algorithm for Constrained Optimization. *Appl. Maths. Comput.* 235, 292–317. doi:10.1016/j.amc.2014.02.093

Wang, J., Zhang, M., Ersoy, O. K., Sun, K., and Bi, Y. (2019). An Improved Real-Coded Genetic Algorithm Using the Heuristical Normal Distribution and Direction-Based Crossover. *Comput. Intell. Neurosci.* 2019, 4243853. doi:10.1155/2019/4243853

Wang, J., Cheng, Z., Ersoy, O. K., Zhang, P., Dai, W., and Dong, Z. (2018). Improvement Analysis and Application of Real-Coded Genetic Algorithm for Solving Constrained Optimization Problems. *Math. Probl. Eng.* 2018, 1–16. doi:10.1155/2018/5760841

Wang, Y., Zhou, J., Lu, Y., Qin, H., and Wang, Y. (2011). Chaotic Self-Adaptive Particle Swarm Optimization Algorithm for Dynamic Economic Dispatch Problem with Valve-point Effects. *Expert Syst. Appl.* doi:10.1016/j.eswa.2011.04.236

Wang, Y., Yang, Z., Mourshed, M., Guo, Y., Niu, Q., and Zhu, X. (2019). Demand Side Management of Plug-In Electric Vehicles and Coordinated Unit Commitment: A Novel Parallel Competitive Swarm Optimization Method. *Energ. Convers. Manage.* 196, 935–949. doi:10.1016/j.enconman.2019.06.012

Wang, Z. Y., Li, Y. R., and Tang, Y. Q. (2019). "Ieee. An Efficient Hybrid DE-WOA Algorithm for Numerical Function Optimization," in Ieee 28th International Symposium on Industrial Electronics (New York: Ieee; 2019), 2629–2634.

Yang, Z., Li, K., and Foley, A. (2015). Computational Scheduling Methods for Integrating Plug-In Electric Vehicles with Power Systems: A Review. *Renew. Sustain. Energ. Rev.* 51, 396–416. doi:10.1016/j.rser.2015.06.007

Yang, Z., Li, K., Niu, Q., and Xue, Y. (2017). A Comprehensive Study of Economic Unit Commitment of Power Systems Integrating Various Renewable Generations and Plug-In Electric Vehicles. *Energ. Convers. Manage.* 132, 460–481. doi:10.1016/j.enconman.2016.11.050

Yang, Z., Li, K., Niu, Q., and Xue, Y. (2017). A Novel Parallel-Series Hybrid Meta-Heuristic Method for Solving a Hybrid Unit Commitment Problem. *Knowledge-Based Syst.* 134, 13–30. doi:10.1016/j.knosys.2017.07.013

Yang, Z., Li, K., Niu, Q., Xue, Y., and Foley, A. (2014). A Self-Learning TLBO Based Dynamic Economic/environmental Dispatch Considering Multiple Plug-In Electric Vehicle Loads. *J. Mod. Power Syst. Clean. Energ.* 2, 298–307. doi:10.1007/s40565-014-0087-6

Yuan, X., Wang, L., Zhang, Y., and Yuan, Y. (2009). A Hybrid Differential Evolution Method for Dynamic Economic Dispatch with Valve-point Effects. *Expert Syst. Appl.* 36, 4042–4048. doi:10.1016/j.eswa.2008.03.006

Zou, D., Li, S., Kong, X., Ouyang, H., and Li, Z. (2018). Solving the Dynamic Economic Dispatch by a Memory-Based Global Differential Evolution and a Repair Technique of Constraint Handling. *Energy* 147, 59–80. doi:10.1016/j.energy.2018.01.029

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.