Multi-objective Slime Mold Algorithm:
A Slime Mold Approach Using Multi-objective Optimization for Parallel Hybrid Power System

Tianjun Zhu,[1*] Hegao Wan,[2**] Zhuang Ouyang,[3] Tunglung Wu,[1]
Jianguo Liang,[1] Weihao Li,[1] Bin Li,[4] and Shiting Han[1]

[1]Department of Mechanical and Automotive Engineering, Zhaoqing University, Zhaoqing 526021, China
[2]Dongfeng Dana Axle Co., Ltd., Xiangyang 441057, China
[3]Guangdong Zhaoqing Institute of Quality Inspection & Metrology, Zhaoqing 526000, China
[4]Zhaoqing University, Aptiv PLC Company, Troy 48098, USA

Owing to the importance of the fuel economy and emission performance of parallel hybrid electric vehicles (PHEVs), parallel hybrid systems are a critical research topic in the vehicle industry. However, previous research has endeavored to reflect the real situation of optimization objectives, which tends to result in low-quality solutions. To address this issue, a novel multi-objective optimization approach based on the multi-objective slime mold algorithm (MOSMA) is used to optimize the hybrid power system of a PHEV. Then, with the objectives of reducing fuel consumption, CO emission, and the sum of HC and NOx emissions, a mathematical model for three-objective optimization is established. The six parameters affecting the hybrid system performance are optimized by considering the dynamic power performance and battery charge state balance constraints. Finally, ADVISOR is used as a simulation platform to verify the optimization results. The results before and after optimization demonstrate that MOSMA can effectively address the multi-objective optimization of hybrid vehicles; concretely, the fuel consumption and the sum of HC and NOx emissions are reduced by 9.5 and 7.4%, respectively. More notably, the decrease in CO emission is as much as 34.4%.

1. Introduction

Owing to their low fuel consumption and reduced emissions, parallel hybrid electric vehicles (PHEVs) can be used to alleviate the global energy shortage and environmental pollution. The efficient operation of the hybrid system depends on the effective matching of components, and the optimization model of the hybrid system is non-differentiable, multidimensional, and nonlinear. Therefore, multi-objective optimization of power system parameters is essential. The goal of hybrid electric vehicle performance optimization is to reduce the system cost and

improve the power performance, economy, and smoothness under the premise of satisfying the performance constraints of each component.[1] Currently, an enumeration method is often used to arrange and combine the decision variables in the performance optimization of hybrid vehicles. However, the efficiency of the combination calculation is greatly reduced with increasing number of decision variables.

With the development and promotion of hybrid vehicles, various control methods and strategies have been used in research to reduce fuel consumption and air pollution. Optimization has been used to improve vehicle performance, which plays a vital role in this field in promoting the development of hybrid vehicles. Liu *et al.* established a global optimization model of the power allocation factor and applied an adaptive simulated annealing algorithm to optimize the power allocation factor offline.[2] Using a PHEV as the objective, eight energy management parameters were chosen as comprehensive optimization parameters for fuel economy and emission performance. Using the Pareto principle, Deng *et al.* modified and increased the performance of the NSGA-II algorithm and proposed a new algorithm based on multi-objective optimization.[3] Zhang *et al.* proposed an energy management strategy using the model predictive control (MPC) framework, which can obtain the optimal torque shunt and shift of a PHEV through efficient calculation.[4] Wang *et al.* explored the size of the lithium-ion battery and ultracapacitor (UC) from a new perspective, including the degree of hybridization between the UC power and battery power; in other words, they minimized fuel consumption by solving an optimization problem.[5] To optimize real-time power allocation, a new energy management strategy was then proposed. Bonfiglio *et al.* used an adaptive simulated annealing genetic algorithm to optimally match the drive components of hydrostatic hybrid vehicles, and the performance of each component was significantly improved.[6] Thus, the optimization of relevant parameters using appropriate optimization methods can effectively enhance the capability of the whole vehicle. The above-mentioned parameters also provide guidance in the control of the whole vehicle and the selection of materials for different components. These parameters can indirectly reflect how different materials should be selected for each component to meet the application requirements. For example, the real-time acquisition of the motor, engine, and battery status through power sensors and current sensors can improve the control of a vehicle to achieve better performance.

In this study, we propose a three-objective optimization model of a parallel hybrid system with constraints. We first design a multi-objective optimization algorithm for a parallel hybrid system based on the multi-objective slime mold algorithm (MOSMA). Then, the fuel consumption, the total emission of hydrocarbons and nitrogen oxides (HC + NOx), and carbon monoxide (CO) emission are selected as the optimization objectives. Furthermore, to ensure the dynamic performance and accurate fuel consumption of vehicles under various road conditions, the dynamic performance and battery charging state balance are used as constraints for optimization. The results show that the optimized system can effectively reduce fuel consumption and pollutant emissions while simultaneously ensuring the power performance. The optimized system can also improve fuel economy.

The major contributions of this paper are summarized as follows:

1. We design a fresh multi-objective optimization approach based on MOSMA to optimize the hybrid power system of a PHEV, which effectively reduces fuel consumption and pollutant emissions and ensures the power performance.
2. A comparative analysis before and after optimization of the motor efficiency, engine efficiency, and engine working point distribution is also performed. The results of comparative experiments show that the working conditions of the engine markedly improved, and the performance of the motor is maximized.

2. Hybrid Power System Optimization Model

The optimization of a parallel hybrid system involves optimizing the parameters of the power components and controllers under various constraints to reduce the fuel consumption and emissions of different pollutants of the vehicle under certain cycle conditions. In the optimization process, several inharmonious objective functions are optimized in a suitable region.[7] The optimization model is non-differentiable, regardless of whether it is discontinuous, multidimensional, constrained, or nonlinear, making this a typical multi-objective problem.[8] The discovery of solutions to multi-objective optimization problems is different from the general mathematical solution process, which is an NP-complete problem.[9]

In terms of solving problems by multi-objective optimization, the following factors must be considered. 1) In most cases, there is some kind of equilibrium among the objectives for each candidate solution, i.e., if one solution is used to improve a certain target value, it is extremely likely to make other target values worse. 2) After the optimization process, there is more than one solution, and when preference information is not considered, there is no superiority among the Pareto optimal solutions. Usually, the structure of a multi-objective optimization problem must be adjusted accordingly, that is, transformed into a maximization or minimization problem.[10] The mathematical model is expressed as follows:

$$\begin{cases} \text{min/max} & F(\vec{x}) = \left\{ f_1(\vec{x}), f_2(\vec{x}), ..., f_k(\vec{x}) \right\}, \\ \text{s. t.} & g_i(\vec{x}) \geq 0, i = 1, 2, ..., m, \\ & h_i(\vec{x}) = 0, i = 1, 2, ..., p, \\ & Lb_i \leq x_i \leq Ub_i, i = 1, 2, ..., n. \end{cases} \tag{1}$$

In this formula, n denotes the number of variables to be optimized; k denotes the number of objective functions designed; m and p denote the numbers of inequality constraints and equality constraints, respectively, g_i denotes the ith inequality constraint; h_i denotes the ith equation constraint; $[Lb_l, Ub_l]$ indicates the upper and lower bounds of the ith variable.

Among the pollutant emissions, CO emission is about 10 times greater than those of hydrocarbons and nitrogen oxides.[11] Thus, in this study, fuel consumption, total hydrocarbon and nitrogen oxide emissions (HC + NOx), and CO emission were selected as the targets as below:

$$
\begin{cases}
f_1(x) = Q_{Fuel}(x), \\
f_2(x) = Q_{CO}(x), \\
f_3(x) = Q_{HC}(x) + Q_{NO}(x),
\end{cases}
\tag{2}
$$

where $Q_{Fuel}(x)$, $Q_{CO}(x)$, and $Q_{HC}(x) + Q_{NO}(x)$ represent the fuel consumption, CO emission, and total of HC and NOx emissions, respectively.

2.1   Constraint conditions

In this study, the power performance and battery charge state of the PHEV are used as the main constraints. In general, the acceleration performance, maximum speed, and maximum climbing gradient are important indicators reflecting the power performance of a vehicle,[12] and the specific contents of the constraints are listed in Table 1.

2.1.1   Power performance constraint

Vehicle acceleration characteristics and the maximum climbing gradient of the vehicle are the main indexes used to estimate the dynamic behavior of the vehicle. The acceleration characteristics of the vehicle are classified into two types: the start-up accelerating performance and the overtaking acceleration performance. However, it is difficult to measure the acceleration value in an actual experiment; thus, the acceleration ability of the vehicle is generally expressed by the acceleration time. The constraints on the acceleration characteristics and climbing gradient are to ensure that the vehicle exhibits good dynamic performance under various driving conditions.

2.1.2   Battery charge state balance constraints

PHEVs have a generator and an engine, i.e., two independent energy supply systems. The energy consumption of a vehicle refers to not only the fuel consumption, but also the power consumption of the battery. During the simulation-based optimization process, it is necessary and feasible to reflect the actual fuel consumption of PHEVs. Generally, the electric energy consumed in the whole cycle is considered, and the consumption of the battery before and after the relevant driving condition is converted into the fuel consumption of the engine, or the simulation is considered effective when the difference in the battery charge state before and after

Table 1
PHEV powertrain constraints.

Indicator	Specific content	Condition
	Acceleration time within 100 km/h	≤14 s
Power performance	40–100 km/h acceleration time	≤14 s
	Climbing gradient	≥30%
Battery charge state balance	Difference between initial and final state of charge	≤0.5%

the restriction is less than or equal to a given small value; thus, the impact of the battery power supply on engine fuel economy is minimized. In this test, to obtain an accurate value for engine fuel consumption, we regarded a difference in the state of charge (SOC) of within 0.5% before and after the relevant driving condition as an effective working condition.

2.2   Optimization parameters

A PHEV is a highly intricate nonlinear system with multiple factors affecting its power system. When all these variables are considered, the problem becomes complicated and the corresponding optimization becomes very difficult. Thus, in this paper, we select the parameters that dominate the power system performance as the optimization object of this experiment.[13] Usually, the driving speed and the maximum speed achieved by PHEVs are influenced by the maximum power of the engine (P_e); the number of modules of the battery (N_b) determines the pure electric driving range of the vehicle; the power factor of the electric motor (M_e) and the final drive ratio (R_b) govern the hill climbing and acceleration ability of the vehicle; and the upper (H_{soc}) and lower (L_{soc}) limits of the SOC of the battery almost completely determine the working range of the battery. Not only do the limits affect the average stable working power of the battery under specific road conditions, but they also have a major impact on the charging and discharging efficiency of the battery system. Moreover, they also strongly affect the power performance, emission performance, and fuel economy performance of the hybrid power system. Thus, the above parameters are selected as the optimization object of this simulation-based test, and the optimization parameters and their corresponding ranges of values are shown in Table 2.

3.   Multi-objective Optimization Algorithm

The power system of a PHEV is a very complex nonlinear system whose performance is influenced by many control parameters, signifying a typical multi-objective problem. When dealing with complex nonlinear multi-objective engineering problems, relational operators are no longer adapted to compare problems with multiple objectives, and the new operator of Pareto optimality is applied. The principle of Pareto optimality is an important concept in game theory, as the Pareto solution set has a good distribution and convergence. Pareto optimality means an ideal state of resource allocation based on a symbolic definition in this regard,[14] and the following important concepts can be shown:

Table 2
Optimization parameters of PHEV and corresponding ranges.

Parameter	Range of values
Maximum engine power (P_e)	[35, 70]
Power factor of electric motor (M_e)	[0.6, 1.5]
Lower limit of battery charge state (L_{soc})	[0.3, 0.55]
Number of modules of battery (N_b)	{20, 21, ..., 49, 50}
Final drive ratio (R_b)	[0.5, 2.5]
Upper limit of charge state of battery (H_{soc})	[0.5, 0.9]

Definition 1: Pareto dominance

For two vectors $\vec{x} = (x_1, x_2, ..., x_k)$ and $\vec{y} = (y_1, y_2, ..., y_k)$, vector $\vec{x}$ is said to dominate $\vec{y}$ (expressed as $\vec{x} < \vec{y}$) if and only if

$$\forall i \in \{1, 2, ..., k\} : f_i(\vec{x}) \leq f_i(\vec{y}) \land \exists i \in \{1, 2, ..., k\} : f_i(\vec{x}) < f_i(\vec{y}). \tag{3}$$

Definition 2: Pareto optimal solutions

Solution a is Pareto optimal if and only if

$$\nexists \vec{y} \in X \mid F(\vec{x}) < F(\vec{y}). \tag{4}$$

Definition 3: Pareto optimal solution set

Pareto optimal solutions are often multiple instead of single, and the following constitutes the Pareto solution set:

$$P_s = \{x, y \in X \mid \exists F(\vec{y}) > F(\vec{x})\}. \tag{5}$$

Definition 4: Pareto optimal frontier (the set of all Pareto efficient allocations)

There are generally many Pareto optimal solutions, and each solution has a corresponding objective function. The corresponding objective function values of all solutions constitute the Pareto solution set:

$$P_f = \{F(\vec{x}) \mid \vec{x} \in P_s\}. \tag{6}$$

3.1 MOSMA algorithm

The slime mold algorithm (SMA), proposed by Li *et al.*, is a meta-heuristic algorithm based on a population that was inspired by observing the oscillatory behavior of slime molds.[15] The SMA uses a positive and negative feedback system combined with an optimal food path design. The slime molds modulate their search paths in real time in accordance with the food quality. The SMA simulates three basic principles, namely, search, wrap, and approach phenomena. The search phenomenon prevents the collision of slime bacteria searching for food, the wrap phenomenon indicates the flowing speed of slime bacteria, and the approach phenomenon explains how slime bacteria approach and snap the food more efficiently

3.1.1 SMA algorithm

At the beginning of the SMA, populations are randomly generated within their upper and lower boundaries, where N denotes the population size (i.e., number of slime bacteria) and *dim* denotes the dimensionality of the problem to be optimized. Then, the objective function is used to evaluate the overall population. In the next stage, the population is updated by executing the

search, wrap, and approach phenomena in each iteration.. Parameters such as the slime mold fitness weight (W) control the progression of the SMA, not only ensuring rapid convergence, but also avoiding local solutions. The vibration parameter (vb) guarantees the accuracy of the search in the early stages and the later development of individual slime bacteria.

Slime bacteria can approach food on the basis of airborne odors, and their approach behavior is represented by[15]

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X_b}(t) + \overrightarrow{vb} \cdot \left(\vec{W} \cdot \overrightarrow{X_A}(t) - \overrightarrow{X_B}(t) \right), r < p, \\ \overrightarrow{vc} \cdot \vec{X}(t), r \geq p, \end{cases} \tag{7}$$

where $\overrightarrow{vb}$ and $\overrightarrow{vc}$ are parameters in the intervals $[-a, a]$ and $[-b, b]$, respectively, t is the number of iterations, $\overrightarrow{X_b}$ is the individual location with the greatest flavor potency found up to the current time, $\vec{X}$ is the current position of the slime, $\overrightarrow{X_A}$ and $\overrightarrow{X_B}$ are two individuals randomly selected among the slime, and $\vec{W}$ is the weighting factor of the slime.

p can be expressed as follows:

$$p = \tanh\left(S(i) - DF\right), \tag{8}$$

where $i \in 1, 2, ..., n$, $S(i)$ indicates the fitness of $\vec{X}$, and DF is used to obtain the best fitness in all iterations.

$\overrightarrow{vb}$ can be expressed by

$$\overrightarrow{vb} = [-a, a], \tag{9}$$

$$a = \operatorname{arctanh}\left(-\left(\frac{t}{\max\_t} \right) + 1 \right). \tag{10}$$

Similarly,

$$\overrightarrow{vc} = [-b, b], \tag{11}$$

$$b = 1 - \frac{i}{Max\_iter}. \tag{12}$$

The formula for $\vec{W}$ is

$$\vec{W}(SmellIndex(\tau)) = \begin{cases} 1 + r \cdot \log\left(\frac{bF - S(i)}{bF - wF} + 1 \right), & \text{condition} \\ 1 - r \cdot \log\left(\frac{bF - S(i)}{bF - wF} + 1 \right), & \text{others} \end{cases} \tag{13}$$

$$SmellIndex = sort(S), \tag{14}$$

where *condition* means that $S(i)$ is in the top half of the total, r is a random value in the interval [0, 1], *Max_t* is the maximum number of iterations, *Max_iter* is the ith iteration, *bF* is the optimal fitness obtained in the current iteration, *wF* is the lowest fitness obtained in the current iteration, and *SmellIndex* indicates the order of the fitness values (ascending in the minimum value problem).

By first searching for the position of individual $\vec{X}$, then locating the best position $\overrightarrow{X_b}$ by finely adjusting parameters $\vec{vb}$, $\vec{vc}$, and $\vec{W}$, the optimal solution of the algorithm can be found. Equation (7) ensures the optimal solution at different positions.

Equation (13) simulates the feedback mechanism of the slime mold, that is, the positive and negative feedback between the width of the leaf veins of the slime mold and the density of the food, and the components in Eq. (13) simulate the uncertainty of the vein contraction pattern. The logarithm is used to moderate the rate of change of the numerical values, which markedly slows the frequency change of vein contraction, and the condition simulates the slime adjusting its search pattern according to the density of the food. The weight of a region with a high food concentration is larger. As the food is consumed, the weight of the region decreases, and then the slime mold moves to other regions to search for food. Figure 1 shows the evaluation process of the slime mold fitness value.

According to the above principle, the mathematical formula used to update the individual positions of slime molds when searching for food is as follows:

$$\vec{X}^* = \begin{cases} rand \cdot (Ub - Lb) + Lb, (rand < 0.03), \\ \overrightarrow{X_b}(t) + \vec{vb} \cdot \left(W \cdot \overrightarrow{X_A}(t) - \overrightarrow{X_B}(t) \right), (r < p), \\ \vec{vc} \cdot \vec{X}(t), \end{cases} \tag{15}$$

where *Lb* represents the upper limit of the search range, *Ub* represents the lower limit of the search range, and *rand* and *r* denote random values in [0, 1]. The oscillation process $\vec{vb}$ imitates the status of the slime bacteria, determining whether the bacteria approach or seek other food
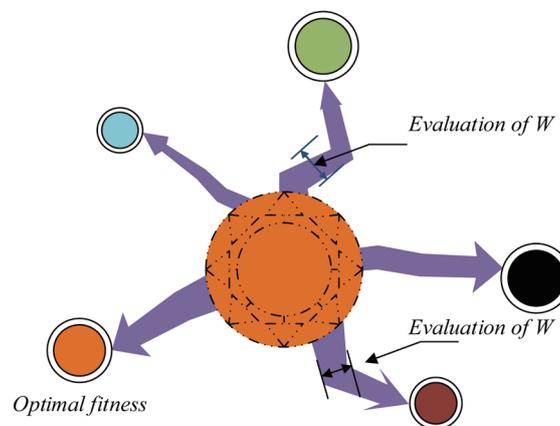
Fig. 1.   (Color online) Assessment of fitness.

sources, and it also increases the likelihood that slime bacteria search for high-quality food, preventing the algorithm from falling into a local optimum.

3.1.2 Framework of MOSMA algorithm

An elite non-dominated ranking (NDR) and a crowding distance mechanism are applied to MOSMA to maintain diversity.[16] The NDR process is given in Fig. 2, where two frontiers are given. The rank index of the solution in the first frontier is 0 because it is not dominated by any solution, while the solution in the second frontier is dominated by at least one solution in the first frontier. The NDR of these solutions equals the number of solutions dominating them. The crowding distance mechanism is shown in Fig. 3, which is used to sustain the diversity among the obtained solutions.

The mathematical expression for the crowding distance (*CD*) is

$$CD_j^i = \frac{f_j^{i+1} - f_j^{i-1}}{f_j^{max} - f_j^{min}},$$
(16)

where f_j^{min} and f_j^{max} are the minimum and maximum values of the *j*th objective function, respectively. A schematic representation of the NDR algorithm is shown in Fig. 3.

The MOSMA process is described as follows:
1) Define the initial value of the control parameters, i.e., the size of the initial population (N_{pop}), the loop termination criterion, and the maximum number of iterations (*Max_t*).
2) Randomly generate parent populations P_0 in region S of the feasible search space, and evaluate each objective function of the objective space vector F in the search space of P_0.
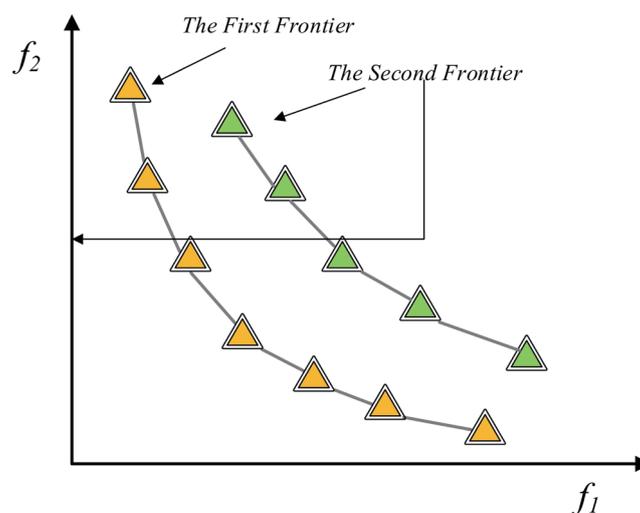
Fig. 2.   (Color online) NDR.
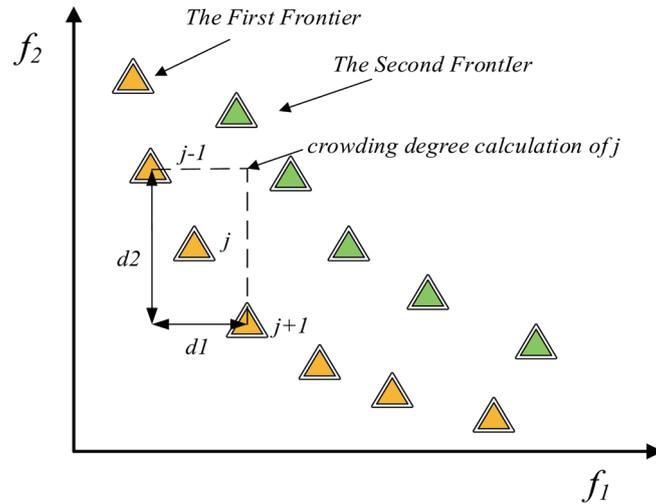
Fig. 3.　(Color online) Crowding calculation.

3) Apply the elite-based NDR sum CD to P_0, obtain the NDR and Pareto frontiers for all individuals $\vec{X}$, and calculate the CD for each frontier.

4) From the results obtained in step 3), calculate the corresponding $\vec{vb}$, $\vec{vc}$, and $\vec{W}$, and update the positions of individuals $\vec{X}$ in the population to create a new population P_j.

5) Merge P_i with P_0 to obtain the population P_i ($P_i = P_0 \cup P_j$) and evaluate each objective function of P_i in the objective space vector F, applying the elite-based NDR and CD and selecting N_{pop} individuals to replace P_0.

6) Satisfy the loop termination criterion, then output P_0. Otherwise, return to step 2).

MOSMA applies the elite-based NDR, and CD guarantees the diversity of population positions; the adaptive weight W ensures that individuals rapidly converge while maintaining a certain perturbation rate to avoid optimal local trapping in the rapid convergence process; the vibration parameter V_b allows individual positions to contract in a particular pathway, thereby ensuring the efficiency of the pre-search and the accuracy of the post-approach; the position update decision parameter p and the use of three different position updates ensure that the shape memory formed during the search process has better adaptability in different search stages to improve the scalability of the individual position update process in the population as shown in Fig. 4. The figure also illustrates the position adjustment of the searching individual in 3D space. The formula *rand in* makes the individual form a search vector at any angle, in other words, search the solution space in any orientation, to simulate the circular fan structure of the slime as it approaches the food. This concept can also be extended to hyperdimensional space.

3.2　Design based on MOSMA algorithm

The computation of the MOSMA-based parallel hybrid system optimization algorithm with constraints involves the following steps (see Fig. 5 for a flowchart):
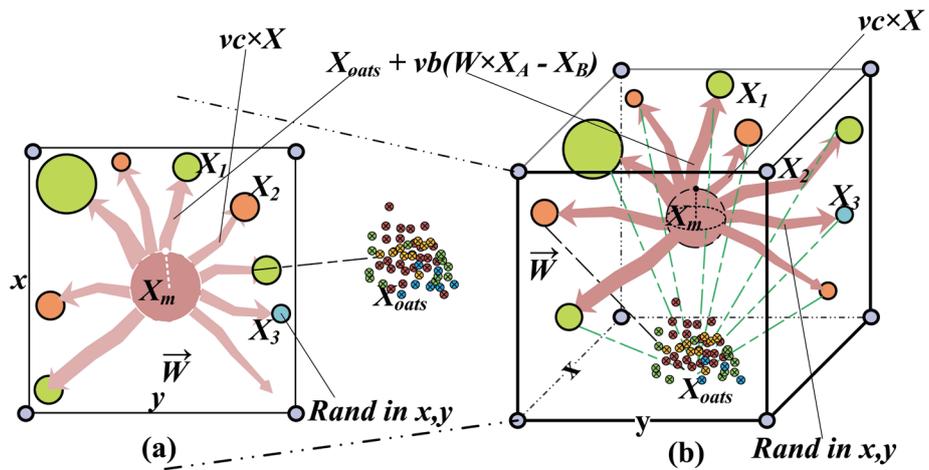
Fig. 4.　(Color online) Schematic of population location update in MOSMA.
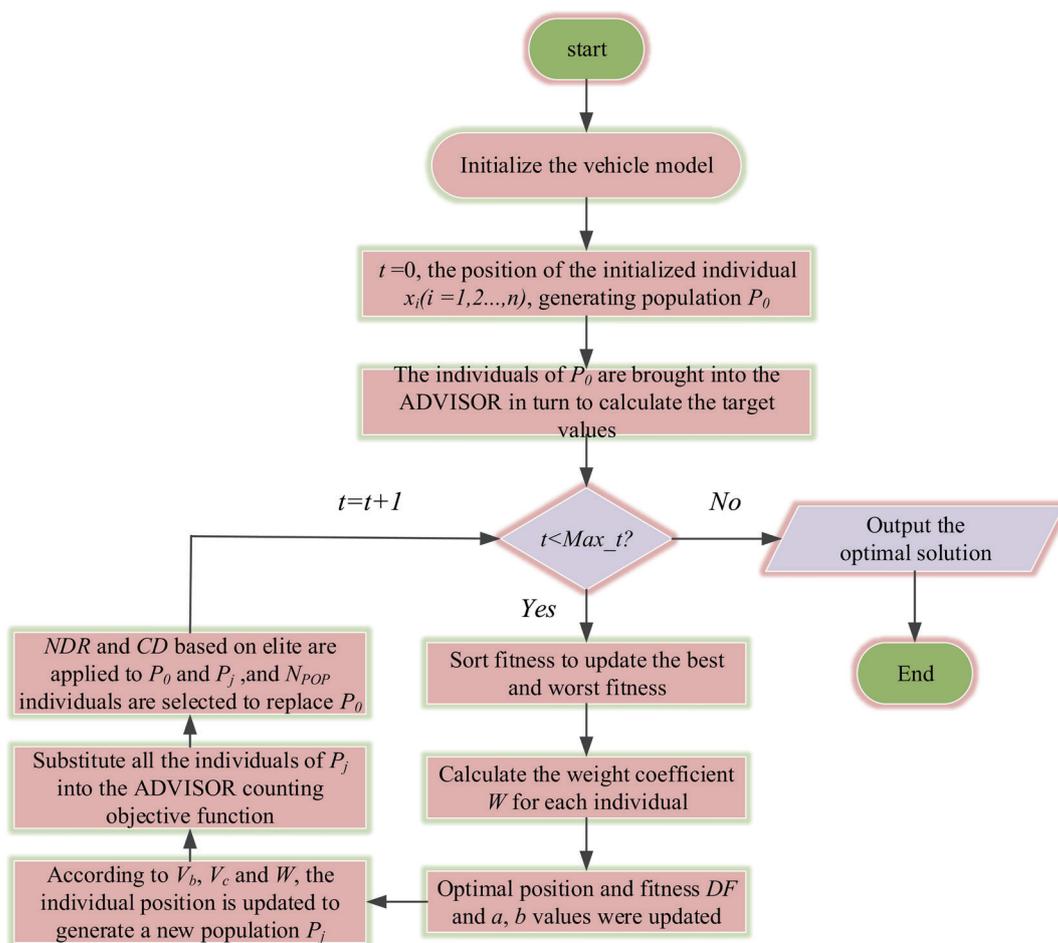
Fig. 5.　(Color online) Flowchart of MOSMA.

1) Initialize the vehicle model, the population size N_{pop}, and the maximum number of iterations *Max_t*, and set $t = 0$.

2) Initialize the locations of the slime x_i ($i = 1, 2, …, n$) in the feasible search space region S and generate a parent population P_0.

3) $\forall x_i \in P_0$, calculate the objective functions $f_1(x_i), f_2(x_i),$ and $f_3(x_i)$.

4) Apply the elite-based NDR and CD to P_0, find the NDR and Pareto frontiers for all individuals x_i, and calculate the crowding distance for each frontier.

5) Calculate the weight coefficients W for each individual based on the optimal fitness value wF and the worst fitness value bF of the current iteration.

6) Update the best fitness DF and obtain the corresponding $\vec{vc}, \vec{vb}$ in all iterations to create a new population P_j by generating new slime mold positions x_i ($i = 1, 2, …, n$).

7) $\forall x_i \in P_j$, calculate the objective functions $f_1(x_i), f_2(x_i),$ and $f_3(x_i)$, and merge P_j with P_0 to obtain the population P_i ($P_i = P_0 \cup P_j$).

8) Evaluate each objective function of the target space vector F of P_i and select N_{pop} individuals to replace P_0 based on the NDR and CD; $t = t + 1$.

9) If the loop termination criterion is satisfied, output P_0; otherwise, return to step 5).

The number of slime $\vec{X}$ represents the size of the population, and each slime position x_i ($i = 1, 2, …, N$) represents a candidate solution. When initializing the slime positions, an individual slime position N_{pop} is generated, and the initialized population P_0 is obtained. In addition, the best fitness DF is updated and the corresponding $\vec{vb}$ and $\vec{vc}$ are obtained in all iterations to generate a new population P_j, and P_j is merged with P_0 to obtain the population P_i ($P_i = P_0 \cup P_j$). The ADVISOR simulation software is used for $\forall x_i \in P_o$ and $\forall x_i \in P_i$ to determine the power performance of the given model by calculating the objective functions $f_1(x_i), f_2(x_i),$ and $f_3(x_i)$. If the performance constraints are met, then the simulated model is driven under the given road conditions and the following values for the objective functions are obtained on the basis of the returned results: $f_1(x_i)$ = fuel consumption, $f_2(x_i)$ = sum of HC and NOx emissions, and $f_3(x_i)$ = CO emission. Otherwise, a sufficiently large value is assigned to $f_1(x_i), f_2(x_i),$ and $f_3(x_i)$. According to this simple penalty method, the poorly positioned individuals have poor adaptation and reduce the frequency of oscillations.

4. Multi-objective Optimization Algorithm

4.1 Experimental setup

In this study, the proposed algorithm for multi-objective optimization of a parallel hybrid powertrain (MOSMA) is implemented on MATLAB. The vehicle model is initialized in step 1), and it is necessary to specify the range of values for each optimization parameter in the optimization variables. The ranges of values in Table 2 and the constraints in Table 1 are employed, and the population size and the maximum number of evolutionary generations are set as 30 and 100, respectively. The test model used is the "PARALLEL_defaults_in" model provided by ADVISOR, and its technical parameters are shown in Table 3.

Table 3
Technical parameters of PHEV.

PHEV part	Parameter	Value
	Vehicle weight	1350 kg
Overall vehicle parameters	Windward area	2.0 m^2
	Wind resistance coefficient	0.335
Engine	Maximum power	41 kW/5700 r/min
	Maximum output torque	81 N.m/3477 r/min
	Maximum power	75 kW
Drive motor	Maximum output torque	273 N.m
	Maximum torque	10000 r/min
Battery	Individual module metrics	12 V, 25 Ah
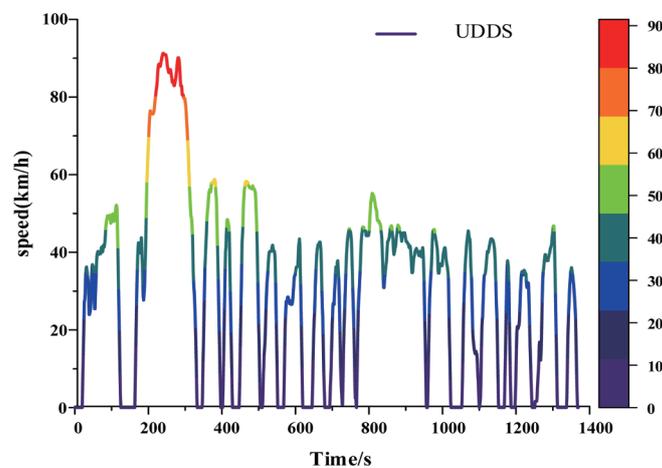
Fig. 6.   (Color online) UDDS road map.

The simulation uses the urban dynamometer driving schedule (UDDS) shown in Fig. 6. This schedule reflects the frequent acceleration, deceleration, and idling on suburban roads with good road conditions, which are very close to the actual situation.

4.2   Optimization results and analysis

The sets of Pareto solutions obtained by performing eight optimization processes are shown in Fig. 7. Observing the data distributions in Figs. 7(a)–7(h), it is evident that the eight optimizations obtain satisfactory convergence results.

The distribution of the final Pareto optimal solution obtained after optimization is shown in Fig. 8. The values of the variables corresponding to the positions of the Pareto optimal solution x_i, i.e., the maximum power of the engine (P_e), the upper limit (H_{soc}) and lower limit (L_{soc}) of the battery charge state, the number of battery modules (N_b), the main drive ratio R_b, and the power factor of the motor M_e, are shown in Table 4. f_1 (fuel consumption), f_2 (sum of HC and NOx emissions), f_3 (CO emission), and the optimized percentage are shown in Table 5.
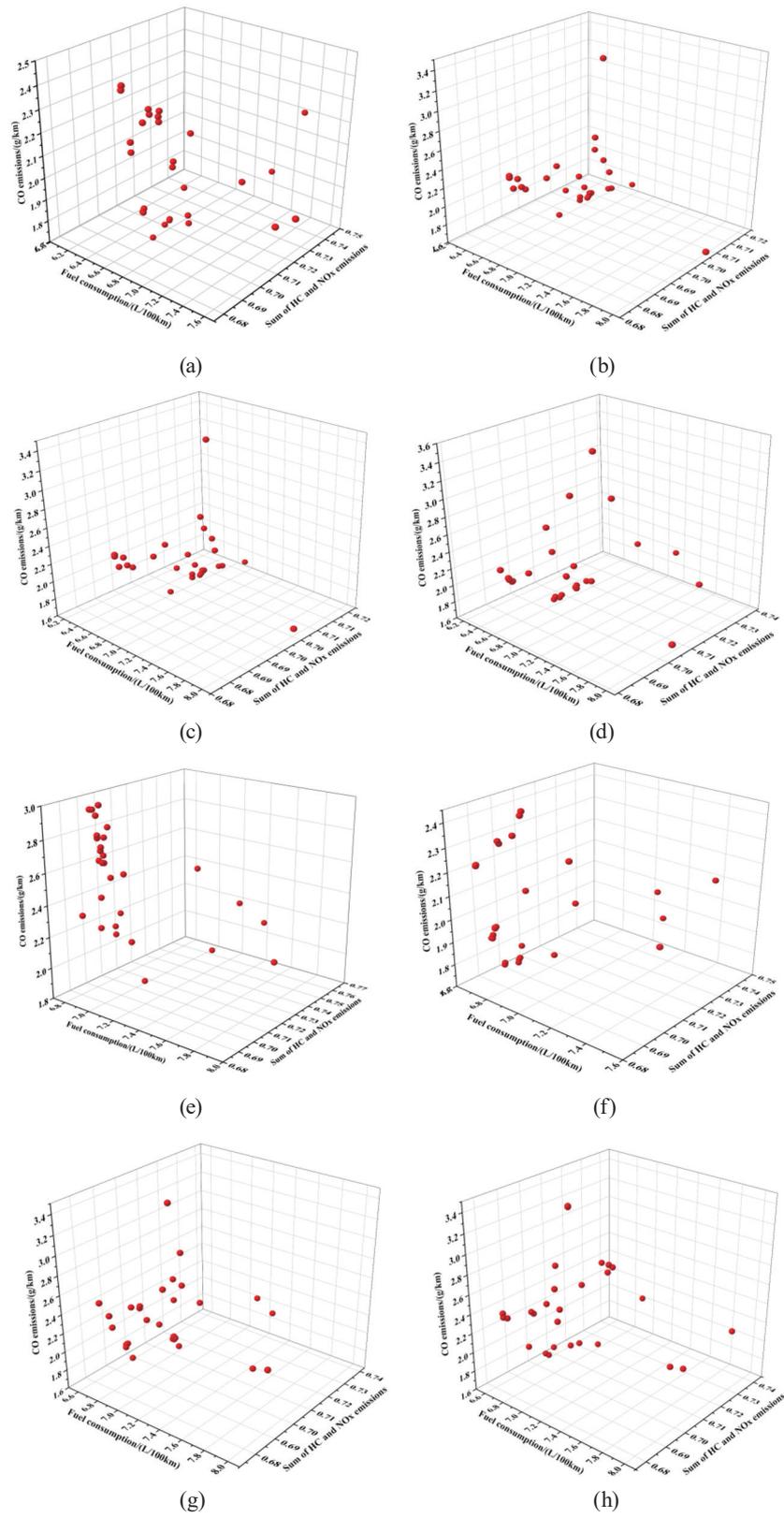
Fig. 7.   (Color online) Eight optimization processes: (a) 1st optimization, (b) 2nd optimization, (c) 3rd optimization, (d) 4th optimization, (e) 5th optimization, (f) 6th optimization, (g) 7th optimization, and (h) 8th optimization..
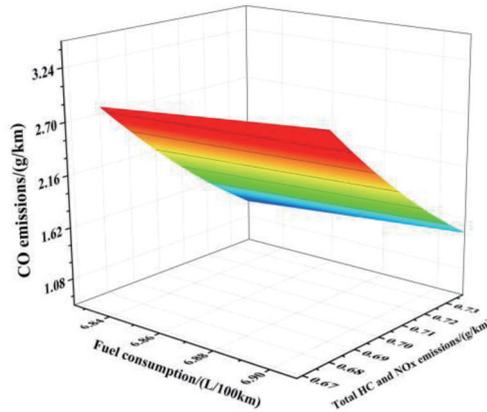
Fig. 8.   (Color online) Pareto optimal solution distribution.

Table 4
Detailed parameters of optimal Pareto solution sets (eight optimizations).

Number	P_e (kW)	H_{soc} (%)	L_{soc} (%)	N_b	R_b	M_e
0	41.007	0.70	0.60	25	1.00	1.00
1	35.000	0.55	0.42	20	1.14	0.600
2	35.000	0.80	0.42	20	1.09	0.600
3	35.000	0.50	0.30	20	1.12	0.600
4	35.000	0.80	0.42	20	1.09	0.600
5	35.000	0.80	0.20	20	1.09	0.600
6	40.057	0.64	0.20	20	1.06	0.600
7	35.000	0.80	0.20	20	1.08	0.600
8	70.000	0.80	0.20	20	1.05	0.610
9	41.762	0.55	0.27	20	1.14	0.600
10	35.000	0.80	0.55	20	0.97	0.600
11	35.000	0.80	0.55	20	0.97	0.600
12	45.200	0.62	0.37	23	1.08	0.862
13	35.000	0.50	0.30	20	1.09	0.600
14	35.000	0.50	0.30	20	1.09	0.600
15	35.000	0.50	0.30	20	1.05	0.600
16	35.000	0.50	0.30	20	1.12	0.734
17	70.000	0.69	0.20	20	0.94	0.600
18	70.000	0.80	0.20	20	1.17	0.600
19	49.884	0.63	0.34	22	1.35	0.685
20	35.000	0.50	0.30	26	1.36	0.600
21	35.000	0.55	0.38	20	1.26	0.600
22	35.000	0.50	0.30	20	1.11	0.600
23	35.000	0.50	0.30	20	1.25	0.600
24	47.414	0.50	0.30	20	1.19	0.600
25	45.339	0.50	0.30	20	1.16	0.600

Among these results from group 0 to group 25, the data of group 0 comprise the system parameter values before optimization and the corresponding fuel consumption, sum of HC and NOx emissions, and CO emission per 100 km. According to the data in Table 4, the optimized results provide different possible sets of parameters according to the requirements. For example,

Table 5
Corresponding objective values and optimization percentages of optimal Pareto solutions (eight optimizations).

Number	f_1 (L/100 km)	f_1 Optimized (%)	f_2 (g/km)	f_2 Optimized (%)	f_3 (g/km)	f_3 Optimized (%)
0	7.47	0.0	0.758	0.0	2.6	0.0
1	6.660	10.8	0.690	9.0	1.684	35.2
2	6.758	9.5	0.710	6.3	1.790	31.2
3	6.680	10.6	0.695	8.3	1.707	34.3
4	6.758	9.5	0.710	6.3	1.790	31.2
5	6.764	9.5	0.710	6.3	1.793	31.0
6	6.792	9.1	0.714	5.8	1.812	30.3
7	6.780	9.2	0.710	6.3	1.829	29.7
8	6.811	8.8	0.715	5.7	1.843	29.1
9	6.692	10.4	0.693	8.6	1.919	26.2
10	6.688	10.5	0.705	7.0	1.998	23.2
11	6.688	10.5	0.705	7.0	1.998	23.2
12	7.127	4.6	0.739	2.5	2.009	22.7
13	6.678	10.6	0.702	7.4	1.764	32.2
14	6.687	10.5	0.703	7.3	1.774	31.8
15	6.700	10.3	0.704	7.1	1.784	31.4
16	6.872	8.0	0.713	5.9	1.784	31.4
17	6.794	9.0	0.713	5.9	2.228	14.3
18	6.787	9.1	0.691	8.8	2.261	13.0
19	6.928	7.3	0.696	8.2	2.324	10.6
20	6.859	8.2	0.689	9.1	2.337	10.1
21	6.760	9.5	0.683	9.9	2.455	5.6
22	6.674	10.7	0.697	8.0	1.739	33.1
23	6.730	9.9	0.680	10.3	2.463	5.3
24	6.710	10.2	0.685	9.6	2.200	15.4
25	6.699	10.3	0.691	8.8	1.894	27.2

when lower fuel consumption is required, the parameter values corresponding to lower values (f_1 column) can be selected, such as groups 3, 14, and 22; when lower HC and NOx emissions are required, the parameter values corresponding to lower f_2 values can be selected, such as groups 21, 23, and 24; when lower CO emission is needed, the parameter values corresponding to lower f_3 values can be selected, such as groups 2, 3, and 5. The values of f_1 to f_3 in groups 1 to 25 in Table 5 are all less than the values of 7.47, 0.758, and 2.6, respectively, before optimization in group 0, indicating that the optimized system can guarantee the dynamic performance. The fuel consumption and pollutant emission are reduced by the optimization. Thus, the fuel economy of the optimized system has been improved. To be specific, the fuel consumption per 100 km has been reduced by 9.5% on average and by a maximum of 10.6%. In addition, in terms of pollutant emissions, the CO emission is markedly decreased by an average of 24.3% and a maximum of 34.4%. Moreover, the total HC and NOx emission is decreased by an average of 7.4% and a maximum of 10.3%.

The optimized solution of group 1 was selected and employed in the simulation system. The motor efficiency, engine efficiency, and the corresponding operating point distributions of group 1 and group 0 data are shown in Figs. 9–14, respectively, to compare the system before and after optimization.
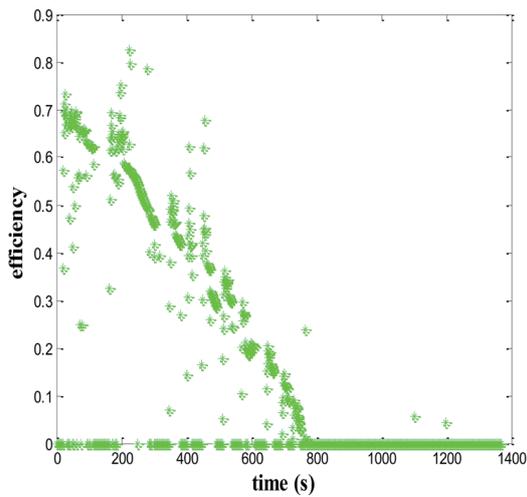
Fig. 9. (Color online) Motor efficiency before optimization.
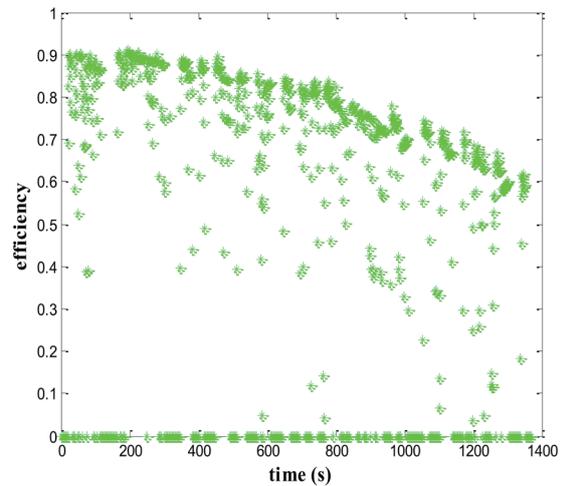
Fig. 10. (Color online) Motor efficiency after optimization.
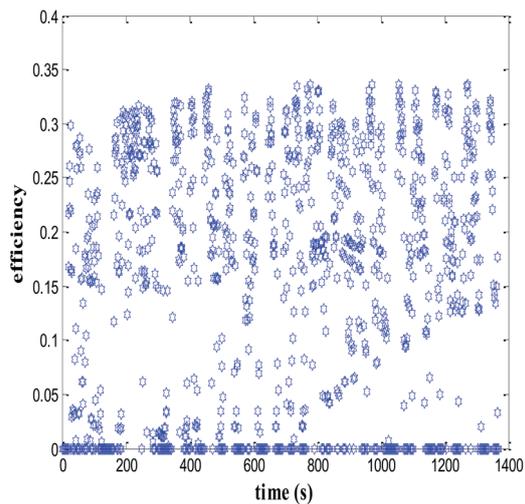
Fig. 11. (Color online) Engine efficiency before optimization.
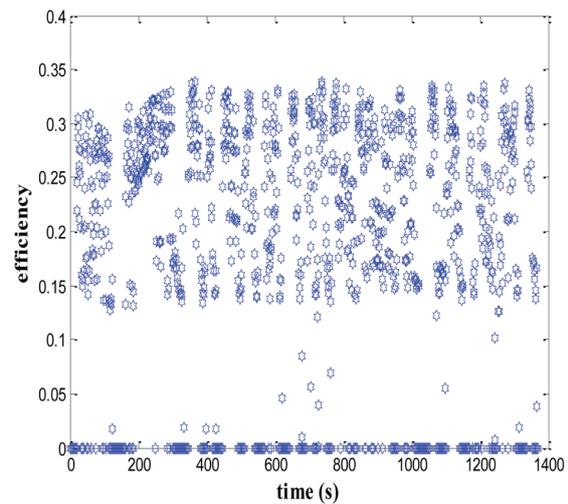
Fig. 12. (Color online) Engine efficiency after optimization.

It is observed that the optimized engine efficiency is mainly distributed in the interval [0.1, 0.85] before optimization in Fig. 9, whereas the optimized motor efficiency is mainly distributed in the interval [0.3, 0.9] in Fig. 10. The number of effective working points has increased significantly, especially the number in the high-efficiency area; thus, the overall motor efficiency has been significantly improved, which is conducive to the efficient use of energy. In addition, the increased number of effective working points of the motor reflects the increased degree of participation of the motor in the vehicle drive. This phenomenon indirectly indicates the improved working frequency of the battery. This increase is conducive to the energy conversion of the parallel hybrid electric motor auxiliary drive, the recovery of energy during braking, and the reduction of energy loss during the driving of the vehicle.
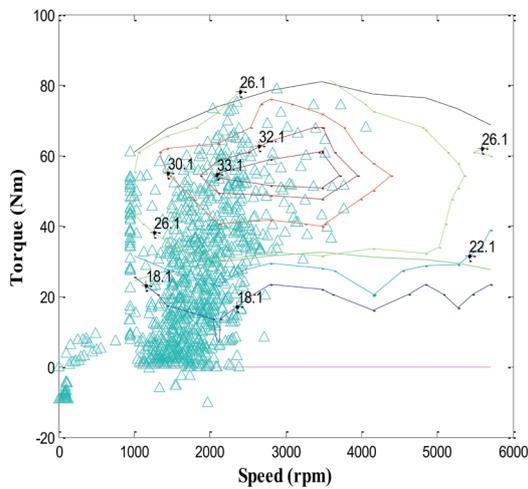
Fig. 13. (Color online) Engine working point distribution before optimization.
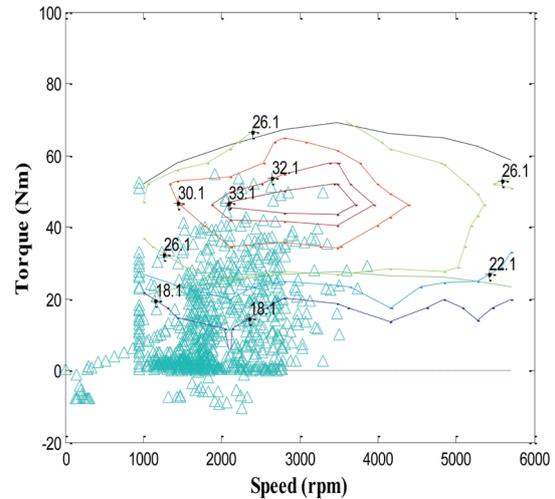
Fig. 14. (Color online) Engine working point distribution after optimization.

According to Figs. 11 and 12, the working efficiency of the engine before optimization is mainly distributed in the interval [0.05, 0.35], whereas after optimization, it is mainly distributed in the interval of [0.1, 0.35]. Moreover, by comparing the working point distributions before and after engine optimization (Figs. 13 and 14), we find that the number of effective working points of the engine is markedly reduced. These data show that the engine efficiency is significantly improved after optimization. The number of engine operating points is clearly reduced, and the points are concentrated in the high-efficiency zone, which coincides with the increase in the number of motor operating points. In addition, the working conditions of the engine are not only improved, but also the performance of the motor is maximized, thus improving its efficiency. Furthermore, the performance of the whole parallel hybrid power system is improved and pollutant emission is effectively reduced.

5.   Conclusions

(1) A multi-objective optimization algorithm based on MOSMA for a parallel hybrid system was proposed, and six parameters that influence the performance of a hybrid system (the number of modules of the battery, the maximum power of the engine, the power factor of the electric motor, the final drive ratio, and the upper and lower limits of the SOC of the battery) were optimized while considering the dynamic power performance, battery charge state balance constraints, and the maximum power of the engine. From the results, we concluded that the parallel hybrid system had significantly improved fuel consumption and lower pollutant emissions after optimization.
(2) A three-objective mathematical model of the hybrid power system was established through the design of a multi-objective optimization algorithm for a parallel hybrid power system based on MOSMA. This model can provide a variety of optional schemes, enabling the

diversification of parallel hybrid system designs. It not only provides personalized product requirements for manufacturers, but also has practical significance for engineering. Furthermore, this algorithm does not set the weight of each objective function, thus effectively avoiding the influence of the introduction of weighting coefficients in the empirical design and more accurately reflecting the change in each objective value.

(3) A comparative analysis of the motor efficiency, engine efficiency, and engine working point distribution before and after optimization was also carried out. The results demonstrate that the total efficiency of the motor, engine, and system were improved significantly, illustrating that multi-objective optimization based on MOSMA has a major advantage in improving the performance of parallel hybrid systems.

Acknowledgments

References

1   F. Q. Zhang, X. S. Hu, K. H. Xu, X. L. Tang, and Y. H. Cui: J. Mech. Eng. **10** (2019) 100. https://doi.org/CNKI: SUN: JXXB.0.2019–10–011

2   H. Liu, X. M. Li, W. D. Wang, L. J. Han, and Z. J. Yan: J. Mech. Eng. **4** (2019) 91. https://doi.org/10.3901/JME.2019.04.09

3   T. Deng, C. S. Lin, Y. N. Li, and R. Z. Lu: Energy Eng. **5** (2016) 531. https://doi.org/10.3969/j.issn.1000–680X.2016.05.003

4   F. Q. Zhang, X. S. Hu, T. Liu, K. H. Xu, and H. Pang: IEEE Trans. Veh. Technol. **70** (IEEE, 2020) 237. https://doi.org/10.1109/TVT.2020.3045271

5   L. Wang, J. Collins, G. Emmanuel, and H. Li: IEEE Trans. Vehicular Technol. **60** (IEEE, 2011) 1419. https://doi.org/10.1109/TVT.2011.2122272

6   A. Bonfiglio, D. Lanzarotto, M. Marchesoni, M. Passalacqua, R. Procopio, and M. Repetto: Energies **4** (2017) 1. https://doi.org/10.3390/en10122142

7   B. Cao, J. W. Zhao, Y. Gu, Y. B. Ling, and X. L. Ma: Swarm Evol. Comput. **3** (2020) 100626. https://doi.org/10.1016/j.swevo.2019.100626

8   H. Zille, H. Ishibuchi, S. Mostaghim, and Y. Nojima: Proc. Genetic & Evolutionary Computation Conf. Companion 2016 (GECCO, 2016) 128.

9   J. Knowles and D. Corne: Proc. 1999 Congr. Evolutionary Computation 1999 (CEC99, 1999) 123–127.

10 S. B. Li, J. L. Qu, and G. C. Yang: Proc. 17th Int. Conf. Automation and Computing 2011 (ICAC, 2011) 127–132.

11 J. K. Peng, H. W. He, and R. Xiong: Appl. Energy **2** (2017) 1633. https://doi.org/10.1016/j.apenergy.2015.12.031

12 I. Rahman, P. M. Vasant, B. Singh, and M. Abdullah-Al-Wadud: Alexandria Eng. J. **2** (2016) 419. https://doi.org/10.1016/j.aej.2015.11.002

13 M. P. Christopher, W. Eli, A. R. Jesse, K. Jonathan, and D. J. Nelson: SAE Int. J. Alt. Power **2** (2012) 503. https://doi.org/10.4271/2012–01–1774

14 C. Y. Wei, X. X. Sun, Y. Chen, L. B. Zang, and S. J. Bai: Energy **230** (2021) 1633. https://doi.org/10.1016/j.energy.2021.120858
15 S. Y. Zhou, Z. Q. Chen, D. Y. Huang, and T. T. Lin: Proc. IEEE Trans. Power Electronics **36** (IEEE, 2020) 1. https://doi.org/10.1109/TPEL.2020.3028154
16 P. Manoharan, P. Jangir, S. Ravichandran, H. H. Alhelou, and H. Chen: IEEE Access **9** (2021) 3229. https://doi.org/10.1109/ACCESS.2020.3047936