

# Research On Driving Torque Strategy For Dual Axis Drive Electric Vehicles

Xiaolin Dong\*

School of Metallurgy and Automotive Engineering, Shandong Vocational L. College of Industry,  
Zibo, 255000, China;

\*dong979173783@163.com

**Abstract.** The advancements in intelligent control have facilitated the implementation of increasingly sophisticated control algorithms in managing the energy of electric vehicles. Among them, particle swarm optimization algorithm is increasingly receiving attention from researchers due to its advantages in speed and accuracy. It starts from any solution and searches for the optimal solution of the target through multiple iterations. The quality of the search results is evaluated using a specific function. Therefore, this article proposes a dual motor dual axis drive electric vehicle driving torque allocation strategy based on the Inertial Weight Linear Decreasing Particle Swarm Optimization (IWLDP SO) algorithm.

**Keywords:** Dual motor dual axle drive electric vehicle; Vehicle controller; Drive torque distribution.

## 1. Introduction

For traditional single motor driven pure electric vehicles with VCU, under normal faultless driving conditions, the VCU usually collects the voltage value of the accelerator pedal sensor, and after a series of filtering and other processing, converts the voltage value into 0-1 accelerator pedal opening value. Based on the external characteristic torque of the motor at the current speed, the required torque is calculated, and torque commands are issued to the MCU. For the dual motor dual axle drive pure electric vehicle studied in this article, after the VCU collects the accelerator pedal values, the total required torque of the entire vehicle can be obtained based on the cumulative value of the accelerator pedal's displacement opening and the external characteristic torque of the two motors [1]. Therefore, the focus of this article is on the driving torque allocation strategy adopted by VCU, which reasonably distributes the total required torque of the entire vehicle to two motors, driving the vehicle while ensuring safety and economy. Figure 1 depicts the arrangement of a dual-motor, dual-axle drive electric vehicle.

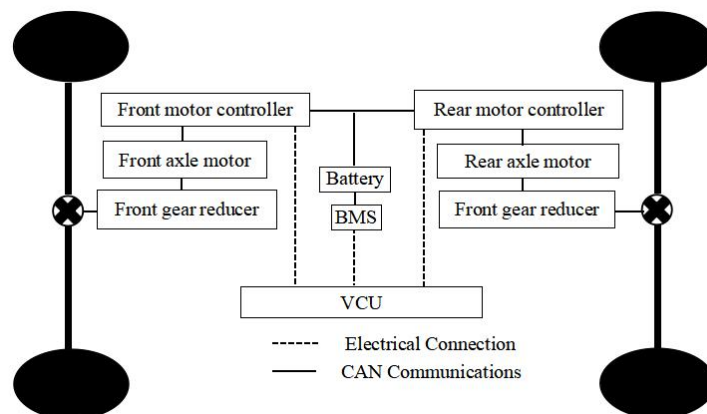


Fig.1 The configuration of dual motor dual-shaft driving electric vehicle

The main work done in this chapter to address this issue is as follows:

(1) This paper proposes an optimization model at the model level to determine the driving torque distribution for dual motor dual axis drive electric vehicles. The objective of the model is to minimize power consumption from the battery and address the economic challenges associated with this electric vehicle configuration.

(2) This study introduces a driving torque allocation strategy that utilizes an inertia weight linearly decreasing particle swarm optimization algorithm. The purpose is to mitigate the negative impact of inertia weight on the search performance of particle swarm optimization.

## 2. Description and Modeling of Torque Allocation Problems

The efficiency of the AC asynchronous motor installed on the vehicle studied in this article is shown in Figure 2. How to find the optimal operating point within the achievable operating range of the motor shown in Figure 2 at any time during vehicle operation, and meet the minimum total electrical power demand of the dual motor system, will be the key problem to be solved by the torque allocation strategy studied in this paper.

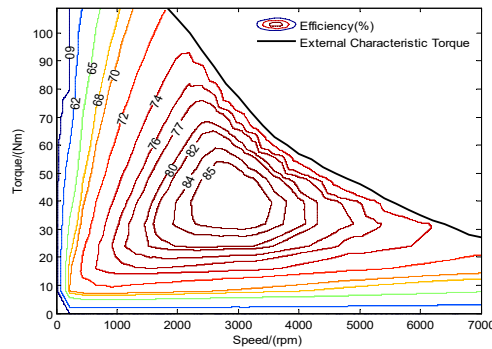


Fig.2 The efficiency map of AC asynchronous motor

This article focuses on the research object and proposes a rule-based torque allocation strategy based on the efficiency diagram of AC asynchronous motors. This strategy includes two working modes: dual motor mode and single motor mode, which are determined by the motor speed  $n$  and the required torque  $T_d$  of the entire vehicle. In single motor mode, the front motor independently drives the entire vehicle by default; In the dual motor mode, a torque sharing strategy is implemented [2]. The specific strategy is as follows:

(1)  $n \leq 1200\text{rpm}$ : When  $0 < T_d \leq 40\text{Nm}$ , use single motor mode; When  $40\text{Nm} < T_d < 220\text{Nm}$ , dual motor mode is used.

(2)  $1200\text{rpm} < n \leq 4500\text{rpm}$ : When  $0 < T_d \leq 50\text{Nm}$ , use single motor mode; When  $50 < T_d \leq T_{\max}(n)$ , the dual motor mode is adopted, and  $T_{\max}(n)$  is the external characteristic torque value of the motor at the current speed.

(3)  $4500\text{rpm} < n \leq 7000\text{rpm}$ : When  $T_d < T_{\max}(n)$ , use single motor mode; When  $T_d > T_{\max}(n)$ , dual motor mode is used.

## 3. Algorithm Design of Drive Torque Allocation Strategy

### 3.1 Key Technologies for Implementing Algorithms

The standard particle swarm optimization algorithm is based on the concept of initializing a group of particles and subsequently adjusting their search speed using global and individual extremum values, iterate, search in the solution space, evaluate the iteration quality using fitness function, and finally find the optimal solution. The initialized quantities include particle position, particle velocity, individual extremum, and global extremum.

In this article, we investigate a problem related to the positioning of particles representing the torque values of a dual motor system. Specifically, the torque values of the  $i$ -th particle during iteration are denoted as  $X_i = (x_{i1}, x_{i2})$ , where  $x_{i1}$  represents the torque value of the front motor and  $x_{i2}$  represents the torque value of the rear motor. Similarly, the particle speed represents the variation value of torque, denoted as  $V = (v_{i1}, v_{i2})$ , where  $v_{i1}$  represents the variation value of the front motor torque and  $v_{i2}$  represents the variation value of the rear motor torque [3].

Individual extremum refers to the optimal solution found by each particle itself as of this iteration, represented by  $P_i=(p_{i1}, p_{i2})$  as the individual extremum of the  $i$ -th particle. The global extremum refers to the optimal solution discovered by the entire population as of this iteration, represented by  $P_g=(P_{g1}, P_{g2})$  to represent the global extremum of the population.

During each iteration, every particle is able to update both its position and velocity by applying the formula provided below.

$$\begin{cases} V_i^{k+1} = \omega V_i^k + c_1 r_1 (P_i - X_i^k) + c_2 r_2 (P_g - X_i^k) \\ X_i^{k+1} = X_i^k + V_i^{k+1} \end{cases} \quad (1)$$

The formula includes  $k$ , representing the number of iterations;  $c_1$  and  $c_2$ , known as learning factors;  $r_1$  and  $r_2$ , randomly distributed numbers between 0 and 1; and  $\omega$ , the inertia weight.

### 3.2 Evaluation of solutions

The selection of fitness function is also an important point in particle swarm optimization. The primary focus of the torque allocation strategy investigated in this study is to minimize the energy consumption of the dual motor system. At each sampling point, the velocity remains constant. Hence, we employ Equation 2 as the fitness function in this research article.

$$f = \frac{T_f}{\eta_f} + \frac{T_r}{\eta_r} \quad (2)$$

Although the standard particle swarm optimization algorithm is known for its easy implementation and high efficiency, experience has revealed that as the number of iterations increases, the algorithm tends to suffer from limitations in global search capabilities, such as local optima and premature convergence. By selecting suitable methods to adjust the value of the inertia weight, this phenomenon can be mitigated or even avoided, thereby improving the performance of the particle swarm algorithm to a certain extent.

This article proposes IWLDPSO to address the practical problem, where the inertia weight linearly changes with the number of iterations (Equation 3). In the early stages of iteration, a larger inertia weight benefits particle quick convergence towards the global optimal value, enhancing the search speed of the particle swarm algorithm while maintaining global search capability. In the later stages of iteration, a smaller inertia weight enhances the accuracy of particle local search. Additionally, linearly decreasing inertia weights offer advantages in programming implementation [4].

$$\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \times \frac{k}{k_{\max}} \quad (3)$$

In the equation, the symbol  $\omega$  represents the current inertia weight,  $\omega_{\max}$  denotes the maximum inertia weight, and  $\omega_{\min}$  denotes the minimum inertia weight. The variable  $k$  represents the current number of iterations, while  $k_{\max}$  represents the maximum number of iterations. It is important to continuously verify and modify the specific values of inertia weight during the debugging process in order to achieve the optimal search capability.

### 3.3 Algorithm steps

Step 1: Population initialization is a crucial step as it provides the initial conditions for algorithm execution. To do this, the population size, particle dimension, and initial position and velocity of each particle need to be determined. Additionally, the learning factor, maximum and minimum inertia weights, and maximum number of iterations should be set.

Step 2: Calculate the initial fitness value of each particle and compare them to determine the individual extreme and global extreme. The rotational speed is obtained from the CAN bus by the

VCU, and the torque is initialized in step 1. To obtain the efficiency value, refer to the motor efficiency chart. Finally, use formula 2 to calculate the fitness value of each particle and determine both the individual and global extreme values.

Step 3: Begin the main loop for iteration. In each iteration, calculate the inertia weight using formula 3. Update the position of each particle (representing the torque value of the dual motor to be solved) using formula 1. Calculate the fitness value of each particle using the method described in step 2. Finally, update the individual extreme value and the global extreme value.

Step 4: After reaching the maximum number of iterations, the algorithm stops iterating and outputs the optimal torque value. Once the maximum number of iterations is reached, the algorithm stops iterating and calculates the final global extreme value. The particle position corresponding to the global extreme value is the sought-after global optimal value of the dual motor torque.

Considering practical engineering applications, it is necessary to impose certain constraints on specific aspects of the particle swarm optimization algorithm. This results in the development of a comprehensive particle swarm algorithm specifically tailored for the problem examined in this paper. The constraints encompassed in this algorithm are as follows [5]:

(1) Limitations on particle position. On the one hand, the torque allocated to each motor at each speed point must not exceed the external characteristic torque value of the motor at that speed. The particle position should be judged during initialization and iteration, and those that do not meet the conditions should be dealt with accordingly; On the other hand, the sum of the dual motor torque to be allocated must be equal to the total required torque of the entire vehicle, so the particle position must also be determined during initialization and iteration updates, and this condition must be met. The initial dual motor torques in this article are each half of the required torque.

(2) The limitation of particle velocity during iteration. On one hand, to enhance search accuracy and prevent the omission of the optimal torque point, and also consider the search speed of the algorithm; On the other hand, in order to prevent excessive changes in particle position during iteration updates, reduce the number of times the particle position exceeds the limit, and thus increase unnecessary computation, this article sets an upper limit of 5 for the maximum particle speed during iteration.

(3) Due to the previous strategy where the motor was the main motor, the torque value of the front motor should not be less than the torque value of the rear motor during the search. If the torque value of the rear motor is greater than that of the front motor, the search result should be rounded.

#### 4. Simulation verification

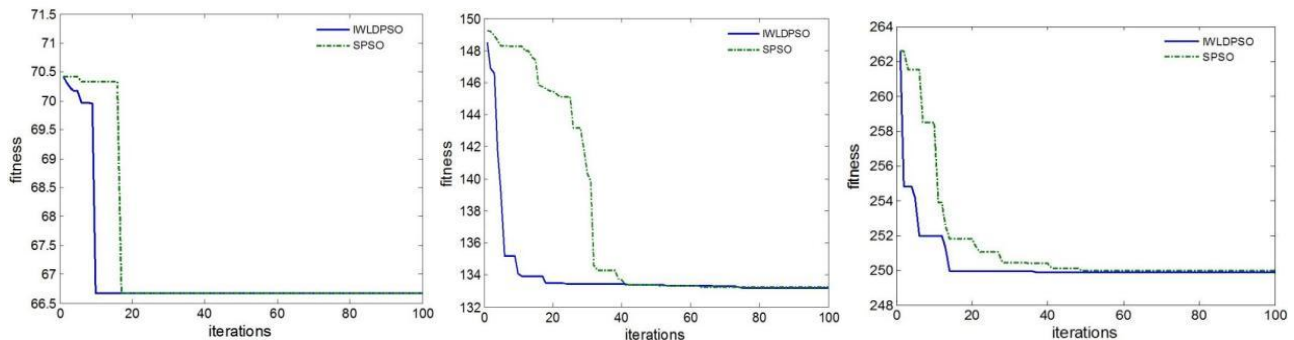
In order to verify the torque distribution strategy for dual motor electric vehicles developed earlier, this article first conducted simulation verification with IWLDPSO. Simulation is mainly used to verify the convergence of the algorithm at a single torque point, and to compare it with the Standard Particle Swarm Optimization (SPSO) control strategy. By analyzing the changes in fitness values, the performance of the two strategies is compared. The parameter settings of IWLDPSO are shown in Table 1.

Tab.1 The parameter settings of IWLDPSO

Parameters	Value
Population size: N	20
Particle dimension	2
Maximum inertia weight: $\omega_{\max}$	0.87
Maximum inertia weight: $\omega_{\min}$	0.39
Learning factor: $c_1$	1.8
Learning factor: $c_2$	1.8
Number of iterations	100

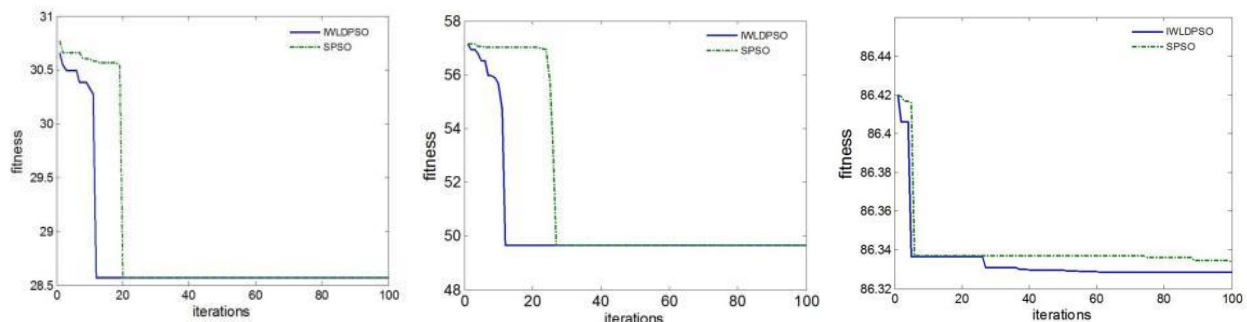
Among them, the inertia weight value of the SPSO algorithm is 0.5, and other parameters are set the same as the IWLDPSO algorithm.

Due to the fact that both SPSO and IWLDPSO belong to particle swarm optimization algorithms, SPSO adopts the same constraints as IWLDPSO. Considering the complexity of car driving conditions, this article conducted simulation validation at low load, medium load, and high load at 1000rpm and 5000rpm speed points, and plotted the fitness values of SPSO and IWLDPSO algorithms into curves. The simulation results of the 1000rpm speed point are shown in Figure 3, and the simulation results of the 5000rpm speed point are shown in Figure 4.



(a)  $T_d = 50\text{Nm}, n = 1000\text{rpm}$  (b)  $T_d = 100\text{Nm}, n = 1000\text{rpm}$  (c)  $T_d = 180\text{Nm}, n = 1000\text{rpm}$

Fig. 3 Simulation result of 1000rpm



(a)  $T_d = 20\text{Nm}, n = 5000\text{rpm}$  (b)  $T_d = 40\text{Nm}, n = 5000\text{rpm}$  (c)  $T_d = 70\text{Nm}, n = 5000\text{rpm}$

Fig. 4 Simulation result of 5000rpm

From Figures 3 and 4, it can be seen that:

(1) Under different speeds and load conditions, rule-based control strategies can directly find the target value without multiple iterations, resulting in higher real-time performance. However, compared to SPSO and IWLDPSO, the target value found by this control strategy differs significantly from the optimal solution, making it unable to leverage the configuration advantages of dual motor electric vehicles.

(2) At different speeds and load conditions, The fitness value of the Inertia Weight Linear Decreasing Particle Swarm Optimization (IWLDPSO) decreases rapidly, resulting in the quick identification of the global optimal solution; The SPSO fitness value decreases slowly and ultimately requires a lot of iterations to find the global optimal solution. Simulation has shown that the inertia weight linearly decreasing particle swarm optimization algorithm developed in this article performs better than conventional particle swarm optimization algorithms, and has a faster speed in finding the optimal torque.

(3) At different speeds and load conditions, there are a certain number of invalid iterations during each algorithm search process, whether it is IWLDPSO or SPSO. This is because the particle swarm optimization algorithm updates at random speeds, resulting in updated torque values that do not meet practical engineering needs and need to be discarded. This phenomenon is inevitable. Due to the fact that the algorithm needs to run in an embedded system, its real-time requirements are

relatively high, and the occurrence of invalid iterations undoubtedly increases the running time of the algorithm. Therefore, further verification is needed to determine whether the algorithm can meet the actual running time requirements.

(4) In terms of finding the optimal solution, IWLDPSO also has certain advantages compared to SPSO, but this advantage is only reflected under high loads, and the two methods find the same optimal solution under medium and low loads.

In summary, compared to SPSO at a single torque point, the IWLDPSO proposed in this paper has better convergence and a shorter required time to complete torque allocation; Compared to rule-based control strategies, the target solution discovered by the Improved Weighted Linear Dynamic Particle Swarm Optimization (IWLDPSO) algorithm proposed in this study consumes less electrical power, which is more in line with our expectations.

## 5. Summary of this chapter

This article first analyzes the configuration characteristics of dual motor electric vehicles, This paper then presents an optimization model for the distribution of driving torque in dual motor electric vehicles, considering the minimum energy consumption of the entire vehicle as the baseline, without taking into account transmission losses and accessory consumption. Based on the efficiency diagram of the motor, the objective of optimization is to minimize the energy consumption of the dual motor system. After referring to the energy management strategy of ordinary multi power source vehicles, a simple original torque distribution strategy was proposed. An in-depth analysis of the torque optimization model was conducted, followed by the proposal of a particle swarm optimization algorithm that utilizes linear descent of inertia weight. Considering practical engineering applications, certain search constraints were added during the search process. This strategy maximizes the utilization of the particle swarm optimization algorithm's global search capability. The algorithm exhibits fast search speed and high accuracy. Consequently, it rapidly identifies the optimal dual motor working point, minimizing the entire vehicle's energy consumption. Offline simulation verification was conducted on this algorithm, which showed that it can search for the optimal torque value at different speeds and loads. Its performance is superior to standard particle swarm optimization algorithm and rule-based torque control strategy, and it has good convergence and fast convergence speed.

## References

- [1] Pedro Daniel Urbina Coronado, Horacio Ahuett-Garza, "Control Strategy for Power Distribution in Dual Motor Propulsion System for Electric Vehicles", *Mathematical Problems in Engineering*, vol. 2015, Article ID 814307, 10 pages, 2015.
- [2] Qingxing Zheng, Shaopeng Tian, Qian Zhang, "Optimal Torque Split Strategy of Dual-Motor Electric Vehicle Using Adaptive Nonlinear Particle Swarm Optimization", *Mathematical Problems in Engineering*, vol. 2020, Article ID 1204260, 21 pages, 2020.
- [3] Xiao-gang Wu, Dian-yu Zheng, "Contrastive Study on Torque Distribution of Distributed Drive Electric Vehicle under Different Control Methods", *Journal of Control Science and Engineering*, vol. 2017, Article ID 2494712, 12 pages, 2017.
- [4] Song Zhenbin, Li Junwei, Sun Binbin, et al. Development of a Model Based Dual Motor Drive Electric Vehicle Control System [J]. *Journal of Shandong University of Technology (Natural Science Edition)*, 2020,34 (02): 31-36.
- [5] Zhang Heng Research on Hardware in the Loop Simulation System for Pure Electric Vehicle Vehicle Controller [D]. Jilin University, 2013.