

Speed and Displacement Control System of Bearingless Brushless DC Motor Based on Improved Bacterial Foraging Algorithm

Xiaoyan Diao, Huangqiu Zhu and Duo Zhang

School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, Jiangsu Province, China

Abstract. To solve the deficiencies of long optimization time and poor precision existing in conventional bacterial foraging algorithm (BFA) in the process of parameter optimization, an improved bacterial foraging algorithm (IBFA) is proposed and applied to speed and displacement control system of bearingless brushless DC (Bearingless BLDC) motors. To begin with the fundamental principle of BFA, the proposed method is introduced and the individual intelligence is efficiently used in the process of parameter optimization, and then the working principle of bearingless BLDC motors is expounded. Finally, modeling and simulation of the speed and displacement control system of bearingless BLDC motors based on the IBFA are carried out by taking the software of MATLAB/Simulink as a platform. Simulation results show that, speed overshoot, torque ripple and rotor position oscillation are dramatically reduced, thus the proposed method has good application prospects in the field of bearingless motors.

1 Introduction

Depending on its outstanding characteristics of good speed regulation, strong ability for overload, high torque of starting/braking, non-friction, high efficiency and reliability, the bearingless BLDC motor has much research values and extensive application prospects in the biomedical field such as blood pumps, high/ultra-high speed centrifugal machines, surgical electric saws and the new energy of flywheel energy storage, etc. [1]. The PID controller is one of the most common controllers with a series of advantages of simple principle, good robustness, strong adaptability and so on [2-3]. But a main problem of the PID controller is that its parameters need manual adjustment and are immutable in the entire control process after regulated [4]. At the same time, the coupling relationships among torque and suspension forces in the bearingless BLDC motor are very complex, and it is easy to cause higher torque ripple, speed overshoot and rotor position oscillation.

To reduce speed overshoot and torque ripple in the speed control system of servo motors, many international and domestic organizations and scholars have proposed many optimization methods for PID controllers, such as genetic algorithm, particle swarm optimization algorithm, and so forth [5]. And these advanced methods have been widely used in many complex engineering problems. Another random search algorithm, BFA, was proposed by American professor Kevin inspired by the foraging method of colibacillus in 2002. The theoretical basis for BFA is that it makes characteristics, concepts and

thoughts of bacteria foraging behaviors mathematization and has been shown the unique adaptive advantages in fields of function optimization, nonlinear model identification, electric system and image compression [6]. However, there are some problems in BFA: the total intelligence and random motion of coli-group are considered but the individual intelligence of a coli is ignored in the foraging process; there is still the possibility for dispersing bacteria with the highest fitness value; the swim step is constant in the entire chemotaxis process, it maybe result in too long optimization time and poor accuracy. Thus it is difficult to convergence in some complicated cases for the optimization process and the optimal solution may be placed in the worst position while swimming.

Dynamic decoupling control of bearingless BLDC motors has been widely studied by domestic and foreign scholars and already achieved many major breakthroughs. In [1], an accurate mathematical model of radial suspension force was proposed, the equivalent circuit was established by the permanent magnet equivalent method and a mathematical model with single degree of freedoms for radial suspension force was derived. Decoupling control of bearingless BLDC motors had been realized successfully and the control system has the advantage of simple and easy to realize. Whereas it has the disadvantage of high torque ripples, speed overshoots and rotor position oscillation.

To solve the deficiencies existing in conventional PI controllers of speed and displacement control system for bearingless BLDC motors and conventional bacterial

foraging algorithm itself, an IBFA is introduced to the bearingless BLDC motor based on the mathematical model of the control system proposed by [1]. This paper emphatic studies the method of reducing torque ripple, speed overshoot and rotor position oscillation. Firstly, the IBFA is proposed by proper foraging behaviour of population and individual, which has the following characteristics: the optimum bacterial is averted in the dispersal process, swim steps are defined as the fitness function and swim is stopped in chemotaxis cycles. Secondly, the IBFA is used in conventional PI controllers for parameter optimization. The speed and displacement control system is proposed based on the method and this system has the characteristics of online tracking. Finally, modeling and simulation are carried out by taking the software of MATLAB/Simulink as platform. Simulation results show that the proposed method can reduce speed overshoot, torque ripple and rotor position oscillation effectively in the speed and displacement control system of the bearingless BLDC motor.

2 Bacterial foraging algorithm

Recently, the search and optimization algorithm based on bacteria foraging behaviour has been used to solve optimization problems. The group can gang-up on larger preys and individuals can obtain protection from predators due to the group characteristics. In the group, each bacterial enjoys the individual intelligence and nutrient source for existence. The bacteria gotten enough food survives in the end. The solution to the problem is corresponding to the bacteria state in the search space, means the fitness value. Bacteria foraging behaviour embodies four elements: chemotaxis, swarming, reproduction and dispersal.

2.1 Chemotaxis

Chemotaxis consists of two basic motion, swim and tumbling, by rotation of flagellums. Bacterial direction of forward motion is dependent on the rotation direction of flagellums. Bacteria swims continue along the prior direction as it rotates in a counter-clockwise direction. Similarly, the direction is changed as in a clockwise direction. The position function of bacteria in the process of swim and tumbling can be expressed as

$$\alpha^i(j+1, k, l) = \alpha^i(j, k, l) + C(i)\varphi(i) \quad (1)$$

where $\alpha^i(j, k, l)$ specifies the position of i th bacterium of flora at j th chemotactic, k th reproductive and l th dispersal step. $C(i)$ is the size of the step that is taken in the random direction only at the beginning of the algorithm and after that, it is specified automatically. $\varphi(i)$ is the new direction after tumbling and can be represented as

$$\varphi(i) = \beta(i) \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad (2)$$

where $\Delta(i)$ is the random variable ranging from 0 to 1 in the process of turning direction. Possible values of $\beta(i)$

can be -1 or 1. At the beginning of the algorithm, the initial value of $\beta(i)$ is selected at random and after that, it changes smartly according to the fitness function of the optimization goal. In particular, $\beta(i)$ is always following the pre-defined direction as long as bacteria swims, but cannot get the minimum value of the fitness function in that direction. Provided that we make $J(\alpha)$ as a fitness function for a bacteria at the position α , then $C(i)$ can be calculated based on the fitness function $J(\alpha^i)$ for i th bacteria

$$C(i) = \max\left(C_{\min}, \frac{|J(i) - J_0|}{K_1}\right) \quad (3)$$

where J_0 is the global optimum solution and $J(\alpha^i)$ is the fitness value for i th bacteria at the position α , and there is $K_1 \gg |J|$. $J(\alpha^i)$ can vary automatically as the optimization process, and its positive constant coefficients should be adjusted properly. From (3), it is observed that $C(i)$ can be adjusted by changing the fitness value and decreases continuously as chemotaxis number increases in the whole process. Thus bacterium has the smaller step with the closer to the global optimum solution.

2.2 Swarming

Swarming bacteria can be realized by cell-to-cell signal via an attractant in the foraging process and the fitness function can be expressed as [7]

$$J_{cc}(\alpha_m(j, k, l), \alpha(j, k, l)) = k \cdot \left(\sum_{i=1}^S \left[-d_a \exp\left(-\omega_a \sum_{m=1}^p (\alpha_m - \alpha_m^i)^2\right) \right] \right) + \sum_{i=1}^S \left[h_r \exp\left(-\omega_r \sum_{m=1}^p (\alpha_m - \alpha_m^i)^2\right) \right] \quad (4)$$

where $J_{cc}(\alpha_m, \alpha(j, k, l))$ is another fitness function combined with the influence of other bacteria based on the real fitness function $J(j, k, l)$. d_a denotes dosage of the repellent released by bacteria, ω_a the test value of the repellent signal width, ω_r the depth of the rejection and h_r the longitudinal rejection value.

2.3 Reproduction and dispersal

After entering the stage of reproduction, the superior can be reproduced and the inferior can be eliminated selected by health values. The health value can be represented by the sum of the fitness values in the chemotaxis process,

$$J_{\text{health}}^i = \sum_{j=1}^{N_c+1} J(i, j, k, l) \quad (5)$$

where J_{health}^i is the health value, that is the sum of the optimal fitness values in the chemotaxis cycles. N_c is the number of chemotaxis. The smaller health value of bacteria means the more nutrition achieved, so it has the greater chance of reproduction. On the other hand, the bigger health value means the bigger possibility of being

eliminated. After reaching the chemotaxis number N_c , the health values obtained from (6) are sorted and half of the bacteria with a bigger health value are eliminated. Simultaneously, half of the bacteria with a smaller health value are reproduced to two bacteria. Where one of them is placed in the same location and another is put around the location. It makes the population of bacteria constant in the reproduction process.

When environmental conditions such as nutrient source and temperature variation in the local region, it is possible that the region is no longer suitable for survival of bacteria. Thus bacteria is compelled to migrate to a new location. To simulate this phenomenon, let P_{ed} denote the probability of bacteria dispersed from one location to another location. Because the outside conditions are stable under the laboratory environment that just the unhealthy bacteria could be dispersed, so the faster and more appropriate rate of convergence are favorable to be obtained.

3 Improved bacterial foraging algorithm

This section will simulate the behaviors of chemotaxis, swarming, reproduction and dispersal and focus on providing detail realization process of the proposed method.

The proposed algorithm has two main characteristics: According to (2), The operator leads each bacterium to the optimum solution, so that if swimming in a pre-defined direction makes a better solution ($\beta(i)=1$), it continues in the same direction, otherwise the direction is changed in an inverse way by tumbling ($\beta(i)=-1$), thus realizes the intelligent foraging; According to (3), a variable step is introduced by each bacteria's fitness function, thus realizes the adaptive step size. It is beneficial to increase the size of the step for bacterium with higher fitness values. What's more, the step decreases and thereby the accuracy increases simultaneously with the convergence process.

The proposed method is used to search the minimum value of the fitness function $J(\alpha)$. Before presenting the detail the implementation process of the proposed algorithm, the necessary variables are defined as follows: p is the latitude of search range, namely number of parameters to be optimized, S the bacterial number in the entire search region, N_s the maximum step size of swimming, N_c the chemotaxis number, N_{er} the number of reproduction steps, N_{ed} the number of dispersal operation, S_r the number of reproduction in each generation and P_{ed} the probability of dispersal. $P(i, j, k) = \{\alpha^i(j, k, l) | i=1, 2, \dots, S\}$ is the initial position of each bacterium, respectively.

Step 1:

(a) Initialize parameters $j=k=l=0$ and select the position for i th bacteria randomly, random variable $\Delta(i)$ in the process of changing direction and the new direction $\beta_0(i)$, where $i=1, 2, \dots, S$.

(b) The initial variables $j=k=l=0$ is used to calculate the fitness values for each bacteria, and then the sum J_{sw} of all bacteria's fitness values is

$$J_{sw}(i, j, k, l) = J(i, j, k, l) + J_{cc}(\alpha_m(j, k, l), \alpha(j, k, l)) \quad (6)$$

Let $J_{last}^i = J_{sw}$ and $J_{last} = \min_{i=1,2,\dots,S}(J_{last}^i)$ as the current fitness values. And save these until a better fitness value is found to replace it via a run.

Step 2: Dispersal loop count, $l++$.

Step 3: Reproduction loop count, $k++$.

Step 4: Chemotaxis loop count, $j++$. Bacterium swims and tumbles in the following rules:

(a)Tumble: Generate a vector $\beta(i)$ and a random vector $\Delta(i)$.

(b)Swim: if m is less than p , let $m++$;

• Let the initial value of the counter for swim length N equal zero;

•while($N < N_s$), let $\alpha_m^i(j, k, l)_N = \alpha_m^i(j, k, l)_{N-1} + C(i)\varphi(i, m)$ and $N=N+1$, where $C(i)$ is the direction of tumbling for each bacteria by (2) and (3).

• Compute $J_{sw}(i, j, k, l)$ by (6). If $J_{sw}(i, j, k, l) < J_{last}^i$, let $\alpha_m^i(j, k, l)_{N+1} = \alpha_m^i(j, k, l)_N + C(i)\varphi(i, m)$ and $J_{last}^i = J_{sw}(i, j, k, l)$. And use $\alpha^i(j, k, l)_{N+1}$ to compute the new $J_{sw}(i, j, k, l)$; otherwise, if $N < 2$, let $\alpha_m^i(j, k, l)_N = \alpha_m^i(j, k, l)_{N-1} + C(i)\varphi(i, m)$ $\beta(i, m) = -\beta(i, m)$ and recompute the new $J_{sw}(i, j, k, l)$. If $J_{sw}(i, j, k, l) < J_{last}^i$, let $\alpha_m^i(j, k, l)_{N+1} = \alpha_m^i(j, k, l)_N + C(i)\varphi(i, m)$ and $J_{last}^i = J_{sw}(i, j, k, l)$, recompute the new $J_{sw}(i, j, k, l)$; otherwise, let $N=N_s$. The while statement is over and the next m of while loop begins.

If $m=p$ and $i < S$, then go to the next bacterium $i=i+1$ to process the next bacterium.

Step 5: if $j < N_c$, go to step 4.

Step 6: if $l > N_{ed}$, the algorithm stops.

Step 7: Reproduction.

(a) For the given k and l , health values of bacterium calculated by (6) are sorted in ascending order, and the greater the health value means the healthier.

(b) Let $S_r = S/2$, eliminate S_r bacterium with the bigger health values and reproduce other S_r bacterium. Copies of that are placed in proximity of the same location as their parent.

Step 8: If $k < N_r$, go to step 3. The next generation will be started in the chemotactic loop in the case of not reaching the number of specified steps.

Step 9: After the health values J_{health} sorted in ascending order, bacterium i ($N_{ed}, N_{ed}+1, \dots, S$) is dispersed to a new location, and then go to step 2. After bacterium reaches a new location, it is need to recompute $J_{sw}(i, j, k, l)$ and sort them by health values.

4 Mathematical model of bearingless BLDC motor

This paper will model the mathematical model of bearingless brushless DC motors according to [1]. The bearingless BLDC motor has two windings: torque windings and suspension force windings. Where the former provides the torque for the motor rotating and drives external loads; the latter is used to sustain the rotor and realize the purpose of the non-friction and high speed.

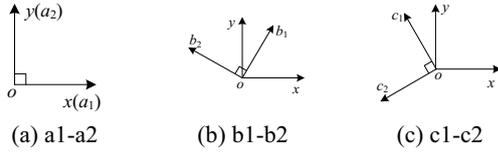


Figure 1. Three coordinate systems of suspension forces exist.

Suppose that the rotor position $\theta \in [0, 30^\circ)$, the rotor suspension is controlled by the a1-a2 coordinates defined by Figure 1(a), so the mathematical model of radial suspension force in any position of the bearingless BLDC motor can be expressed as

$$\begin{bmatrix} F_{sa1} \\ F_{sa2} \end{bmatrix} = \begin{bmatrix} k_i i_{a1} + k_x a_1 \\ k_i i_{a2} + k_x a_2 \end{bmatrix} = k_i \begin{bmatrix} i_{a1} \\ i_{a2} \end{bmatrix} + k_x \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \quad (7)$$

where i_{a1} and i_{a2} denote currents in suspension force windings a₁ and a₂, respectively; a_1 and a_2 denote radial displacement along axis a₁ and axis a₂, respectively; F_{sa1} and F_{sa2} denote radial suspension force along axis a₁ and axis a₂, respectively; k_i is stiffness coefficient of current and k_x is stiffness coefficient of displacement, there are

$$k_i = \frac{2SB_r l_m N}{(l_m + g)^2} \quad (8)$$

$$k_x = \frac{6B_r^2 l_m^2 S}{\mu_0 (l_m + g)^3} \quad (9)$$

Transform (7) to the coordinates xoy , and there is

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} F_{sa1} \\ F_{sa2} \end{bmatrix} = k_i \begin{bmatrix} i_{a1} \\ i_{a2} \end{bmatrix} + k_x \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \quad (10)$$

where F_x and F_y denote radial suspension force along axis x and axis y, respectively.

In a similar way, mathematical models of radial suspension force under the b1-b2 coordinates and the c1-c2 coordinates can be derived.

Suppose that m denotes rotor quality, without regard to disturbing force from outside (because disturbing force is mainly gravity of rotor), then the motion equations are [8] [9]

$$k_i i_x + k_x x = m \frac{d^2 x}{dt^2} \quad (11)$$

$$k_i i_y + k_x y = m \frac{d^2 y}{dt^2} \quad (12)$$

where Ω is rotor velocity, T_L is load torque, D is viscous damping coefficient and J is rotational inertia.

5 Modeling and simulation verification

To verify the feasibility and reliability of the proposed method, parameters of K_{ps} , K_{is} and K_{pd} , K_{id} of the speed PI controller and the rotor radial displacement PI controller are carried out with online optimization respectively[10].

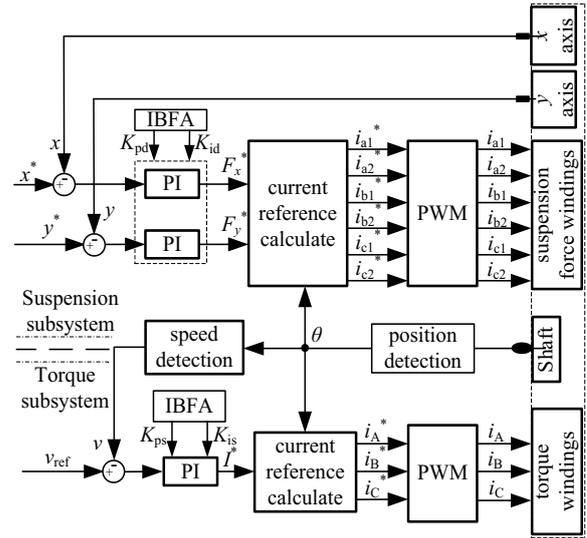


Figure 2. Block diagram of the control system for bearingless BLDC motors.

Figure 2 is block diagram of the control system for the bearingless BLDC motor based on the IBFA. The control system consists of position detection, speed detection, current reference calculation, PWM controller, speed PI controller, radial displacement PI controller and the bearingless BLDC motor ontology. Where parameters of K_{ps} , K_{is} and K_{pd} , K_{id} of the speed PI controller and the rotor radial displacement PI controller are carried out with online optimization by the proposed IBFA, respectively. The current reference signal is generated after the speed error via the PI controller, the error obtained by comparison of the current reference and the actual current of the stator is as input of the PWM controller. The suspension force signal is generated after the radial displacement errors via the PI controllers, and then the current reference of radial displacement is obtained by the rotor position[11].

Main parameters are as follows: equivalent impedance of each phase is 0.05 Ω , load torque is constant 5Nm. The initial speed is 850 r/min and the speed jumps to 1350r/min. Simulation time is 0.2 s. The appropriate choice of parameters N_{er} , N_c , N_s and P_{ed} is beneficial to the balance between the algorithm speed and the accuracy, so these values are concerned with the algorithm performance. In this paper, let bacterium number $S=90$, $N_{ed}=3$, $N_{er}=2$, $N_c=1$, $N_s=5$ and $P_{ed}=15\%$. And then, simulation is carried out and the dynamic characteristics of speed, torque and rotor position when the speed changes.

Figure 3 is the response curve of speed. It can be observed that speed rise to the initial speed setpoint 850r/min rapidly during motor starting process and speed overshoot is nearly zero. At time 0.1 s, speed reaches to speed setpoint in such a short time 5ms, and then it is rapidly entering a new steady state.

Figure 4 is the response curve of electromagnetic torque. It can be observed that electromagnetic torque rise to load torque during motor starting process. In order to quickly adapt the speed variation at time 0.1s, torque has a small fluctuation about 5ms, and then it stabilizes

quickly. Thus torque has excellent dynamic and static performance, and the torque ripple is small within the entire simulation time except at time of speed mutation.

Figure 5 and Figure 6 are the response curve of radial displacement along the axis x and the axis y , respectively. We could see that the rotor position return to zero rapidly when motor starts because the initial status of rotor position is unknown. The rotor position is almost no turbulence throughout the entire process.

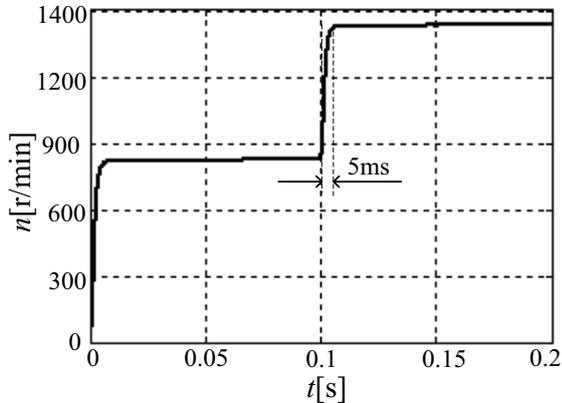


Figure 3. The response curve of speed.

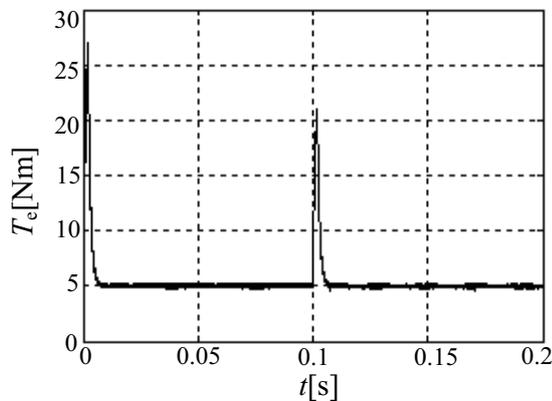


Figure 4. The response curve of electromagnetic torque.

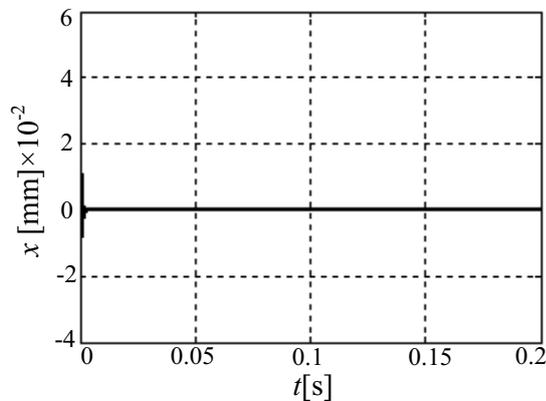


Figure 5. The response curve of radial displacement along the axis x .

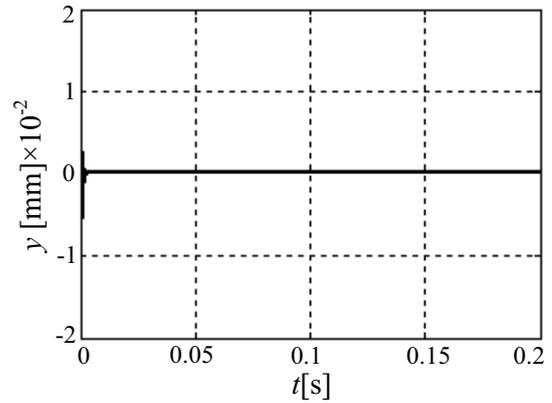


Figure 6. The response curve of radial displacement along the axis y .

6 Conclusion

An improved bacterial foraging algorithm is proposed and applied to the speed PI controller and the radial displacement PI controller for parameter optimization. The principle of bacterial foraging algorithm is described and the proposed bacterial foraging algorithm is concrete introduced. Simulation results show that speed overshoot decreases, torque ripple and radial displacement oscillation are reduced significantly when using the proposed method for parameter optimization of the speed PI controller and the radial displacement PI controller in the bearingless BLDC motor. And the system has good dynamic and static performance. Therefore, the proposed method has a good reference significance and application prospect for improving the speed and displacement control system of bearingless BLDC motors.

References

1. C.L. Xia, H.W. Fang, *Trans. CES*, **27**, 25 (2012)
2. L.G. Chen, H.Q. Zhu, *Proc. CSEE*, **32**, 75 (2012)
3. Z.B. Qin, F.F. Chen, B. Jin, *J. ZJ. Univ.* **49**, 1503 (2015)
4. G.F. Li, S.F. Pan, X. Wang, *Ind. Control. Comput.* **27**, 64 (2014)
5. R. Hooshmand, S. Soltani, *IEEE Trans. P. S.* **27**, 47 (2012)
6. W.F. Huang, W.X. Lin, H.K. Fan, *Comput. Eng. Appl.* **47**, 82 (2011)
7. E. Daryabeigi, B.M. Dehkordi, *Elec. Power Energy Syst.* **62**, 364 (2014)
8. H.Q. Zhu, L.G. Chen, X.Y. Diao, *Chinese Control Conference* (Chinese Association of Automation, Hefei, 2012)
9. L.G. Chen, X.D. Sun, H.Q. Zhu, *Springer*, **97**, 273 (2011)
10. H. Li, F. Wang, *Mach. T. Hyd.* **36**, 337 (2008)
11. H.Q. Zhu, L. Fang, *IEEE International Power Electronics & Motion Control Conference* (IEEE, Shanghai, 2006)