

Research Article

Multiobjective Optimization of PID Controller of PMSM

Qingyang Xu,¹ Chengjin Zhang,¹ Li Zhang,¹ and Chaoyang Wang²

¹ School of Mechanical, Electrical & Information Engineering, Shandong University, Weihai 264209, China

² School of Mathematics and Systems Science, Shandong University of Science and Technology, Qingdao 266590, China

Correspondence should be addressed to Qingyang Xu; xuqy1981@163.com

Received 21 April 2014; Revised 4 August 2014; Accepted 13 August 2014; Published 31 August 2014

Academic Editor: Yang Shi

Copyright © 2014 Qingyang Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

PID controller is used in most of the current-speed closed-loop control of permanent magnet synchronous motors (PMSM) servo system. However, K_p , K_i , and K_d of PID are difficult to tune due to the multiple objectives. In order to obtain the optimal PID parameters, we adopt a NSGA-II to optimize the PID parameters in this paper. According to the practical requirement, several objective functions are defined. NSGA-II can search the optimal parameters according to the objective functions with better robustness. This approach provides a more theoretical basis for the optimization of PID parameters than the aggregation function method. The simulation results indicate that the system is valid, and the NSGA-II can obtain the Pareto front of PID parameters.

1. Introduction

The high-field-strength neodymium-iron-boron (NdFeB) magnets have become commercially available with affordable prices, so the permanent magnet synchronous motors (PMSM) servo system is receiving increasing attention due to its high speed, power density, and efficiency. It is suitable for some applications of high-performance requirement, for example, robotics, aerospace, electric ship propulsion systems, and wind power generation systems [1–3]. PMSM can provide significant performance improvement in many variable speed applications [4].

PMSM is a multivariable, nonlinear, time-varying, and strongly coupled system. With the development of control theory, various alternative control methods, including feedback linearization, feedback-feedforward, sliding mode variable structure, neural network control, adaptive control, fuzzy control, H_∞, and antistep control, have been proposed [5–12]. However, some advanced control techniques are too complex to implement in practical control due to the problem of instantaneity or memory size. Therefore, PID is the most popular controller in the motor control. It provides proportional, integral, and derivative actions for the feedback control system. PID controller has the advantage of simple

structure, good stability, and high reliability [13, 14]. In the process industry, more than 90% of the controllers are PID controllers [15]. Although the number of parameters to adjust in a PID is very small, there are many tuning rules [1]. It has been experimentally checked that more than 30% of controllers are operating in manual mode and 65% of the loops operating in automatic mode are poorly tuned because of the inappropriate parameters [16]. Currently, most of the current-speed closed-loop control in the PMSM servo system adopts PID controller [17]. Nevertheless, the PID controller has poor performance in PMSM control due to the inappropriate parameters.

Over the years, many methods have been proposed for the tuning of PID controller, both in the deterministic or in the stochastic frameworks [18, 19]. The pole placement and minimum variance techniques [20], Ziegler-Nichols and Cohen-Coon methods [15], are the most commonly used conventional methods for tuning PID controllers. However, there will be large overshoot and shaking by these methods [21]. There are also other advanced methods to design the PID controller, such as H_∞ method [22] or the linear matrix inequality technique [23]. Tuning of the PID controller is not a straightforward problem especially when the plants to be controlled are nonlinear and unstable. It can be considered as

a parameter optimization process to achieve a good system response, such as a minimum rise time, overshoot, and regulating time. Thus, the tuning process of the controller has multiple objectives to be achieved, and they are conflicting with one another in most cases.

During the past decades, great attention has been paid to the stochastic approach, which has potential to solve this problem [24, 25]. GA, for instance, is a powerful search algorithm used by research to optimize PID controllers. The optimization process of GA does not require any gradient information and inherent parallelism in searching the design space. However, most of the literatures focus on a single-objective optimization while the PID tuning problem is clearly a multiobjective optimization problem.

Recently, the number of multiobjective evolutionary algorithms increases drastically due to their popularity and capability of successfully solving multiobjective optimization problems. Herreros et al. [16] proposed a MRCD (multiobjective robust control design) genetic algorithm to optimize the PID and considered some benchmark plants for PID controller design. Zhao et al. proposed two lbests multiobjective particle swarm optimization (2LB-MOPSO) to optimize PID controllers of MIMO system [15]. Gao et al. adopted multiobjective differential evolution algorithm to optimize the fractional order PID of gun control equipments [26]. In the single-objective optimization works of research, the aggression function method is always the preferred one. The performance of PID is evaluated through the weighting of some systems' performance indicators by the aggression function. However, the weighting values always have different dimensions. Therefore, it is difficult to determine the values of the weight. NSGA which was proposed by Srinivas and Deb [27] has been successfully applied to solving many problems. An improved version of NSGA, which they called NSGA-II, overcomes some disadvantages of NSGA, such as high computational complexity of nondominated sorting, lack of elitism, and need of a sharing parameter. NSGA-II is considered as the state-of-the-art multiobjective evolutionary algorithm [18]. The solution of the multiobjective is a Pareto set. The solutions of the aggression function method are on a hyperplane due to the usage of the weighting. However, the Pareto front is not limited within the hyperplane. Without setting the values of weight, the realization of the optimization based on NSGA-II is easy to be achieved [28]. And when the system structure is changed, the algorithm is effective, unlike the aggression function method where the weight values must be adjusted again. The purpose of this paper is to extend this methodology for solution of a multiobjective control problem under the framework of NSGA-II approach. The efficiency of the proposed method is illustrated by solving the tuning of a PID controller applied to a PMSM. In the present work, a multiobjective optimization was carried out to obtain the optimal PID's parameters. The simulation results have shown that the NSGA-II algorithm can evolve good control profiles and that the optimization algorithm exhibits more robustness than GA when the system structure is changed.

2. Mathematic Model of PMSM

The mathematical model of PMSM in a d, q two-phase rotating coordinate system is shown below. The voltage equation is [29]:

$$\begin{aligned} u_q &= R_s i_q + L_q \dot{i}_q + \omega_e L_d i_d + \omega_e \psi_f, \\ u_d &= R_s i_d + L_d \dot{i}_d - \omega_e L_q i_q, \end{aligned} \quad (1)$$

where the u_d and u_q represent the stator winding shaft in a straight axis and the quadrature voltage, respectively; i_d and i_q are the direct-axis current and quadrature-axis current, respectively; R_s is the stator phase resistance; L_d is the straight axis inductance; L_q is the quadrature-axis inductance; ψ_f is the permanent-magnet fundamental excitation magnetic field and stator winding of the magnetic chain; ω_e is the electric angular speed of rotor.

The magnetic linkage equation can be expressed as follows:

$$\begin{aligned} \psi_d &= L_d i_d + \psi_f, \\ \psi_q &= L_q i_q, \end{aligned} \quad (2)$$

where the ψ_d and ψ_q represent the syntheses of the magnetic fields in space-direct and quadrature-axis stator winding of the magnetic chain, respectively.

The electromagnetic torque of PMSM in the d, q coordinate is [30]

$$T_e = p_n (\psi_f i_q - (L_d - L_q) i_d i_q), \quad (3)$$

where p_n is the number of the pole pairs.

According to the motion equation of motor,

$$\begin{aligned} J p \dot{\Omega}_r &= T_e - T_l - B \Omega_r, \\ \Omega_r &= \frac{\omega_e}{p_n}, \end{aligned} \quad (4)$$

where the Ω_r is the mechanical angular speed of rotor, B is the viscous friction coefficient, J is the total moment inertia of rotor and load, and T_l is the load torque.

Thus, the state equation can be derived from the above equations:

$$\begin{aligned} \dot{i}_q &= \frac{1}{L_q} (u_q - R_s i_q - L_d i_d \omega_e - \psi_f \omega_e), \\ \dot{i}_d &= \frac{u_d}{L_q} (u_d - R_s i_d - \omega_e L_q i_q), \\ \dot{\omega}_e &= \frac{1.5 p_n^2 (\psi_f i_q + (L_d - L_q) i_d i_q) - p_n T_m - B \omega_e}{J}. \end{aligned} \quad (5)$$

In the VC system of PMSM, $i_d = 0$. Therefore, the state space equation (5) is described as

$$\begin{aligned} \dot{i}_q &= \frac{1}{L_q} (u_q - R_s i_q - \psi_f \omega_e), \\ \dot{\omega}_e &= \frac{1.5 p_n^2 \psi_f i_q - p_n T_m - B \omega_e}{J}. \end{aligned} \quad (6)$$

```

Pop = InitPop(N) % initialization
Fronts = Fast_Nondominated_Sort (Pop) % fast nondominated sort
while gen ≤ ngens
  for i = 1: N
    p1, p2 = Parents_select (Pop(i))
    TempPop(i).x = crossover(p1, p2) % crossover operation
    TempPop(i).x = mutation(%mutation) % mutation operation
    NewPop(i) = TempPop(i) ∪ Pop(i) % new population obtaining
    Fronts = Fast_Nondominated_Sort (NewPop(i))
  end for
  k = 0
  Pop = Null
  while sizeof (Pop) < N
    Frontk = Get_Fronts(Fronts, k) % Pareto fronts obtaining
    Pop = Pop ∪ Ind_Pront(NewPop, Frontk)
    k = k + 1
  end while
  gen = gen + 1
end while

```

ALGORITHM 1: NSGA-II algorithm.

3. PID Controller Optimization

3.1. PID Controller. The continuous form of a PID controller, with input e and output u , is shown as follows:

$$u(t) = K_p e(t) + K_i \int e(t) + K_d \dot{e}(t), \quad (7)$$

where K_p is the proportional gain, K_i is the integral gain, and K_d is the derivative gain.

There are two types of discrete PID by discretization of continuous PID. The position type discrete PID is described as

$$u(k) = K_p e(k) + K_i \sum_{j=0}^k T_s e(k) + \frac{K_d}{T_s} (e(k) - e(k-1)), \quad (8)$$

where $u(k)$ is the controller output and $e(k)$ is the error. In practical system control, the integral part is not flexible. Therefore, another velocity-type discrete PID is described as

$$\begin{aligned} \Delta u(k) &= K_p \Delta e(k) + K_i T_s e(k) + \frac{K_d}{T_s} (\Delta e(k) - \Delta e(k-1)), \\ \Delta e(k) &= e(k) - e(k-1), \end{aligned} \quad (9)$$

where T_s is the sampling time. For the velocity-type PID, we do not need to calculate the integral part, and the controller output is the increment of PID. Therefore, it is often used in practical system control.

3.2. Single-Objective Optimization of PID Controller. The principal part of multiobjective optimization algorithm is the individuals' evaluation mechanism, such as the aggregation function method and the Pareto based method.

Aggregation function is a conventional method which can convert a multiobjective problem into a single-objective problem:

$$\text{fitness} = \sum_{i=1}^n w_i f_i, \quad (10)$$

where the fitness is the summation of fitness, w_i is the weight of i th objective, and f_i is the fitness value of i th objective.

In the optimization process, the object is to evaluate the performance of PIDs. Thus, for PID, the fitness function is written as follows:

$$\begin{aligned} f_1 &= \int_0^{\infty} |e(t)| dt, \\ f_2 &= \int_0^{\infty} u^2(t) dt, \\ f_3 &= t_r, \end{aligned} \quad (11)$$

where the $e(t)$ is the system error, $u(t)$ is the control output, and t_r is the rising time.

To avoid overshoot, a penalty value is always adopted in the fitness function. That is, once overshoot occurs, the value of overshoot is added to the fitness function. Hence, the penalty function is written as

$$f_4 = \begin{cases} \int_0^{\infty} (y(t) - y(t-1)) dt, & \text{if } e(t) < 0, \\ 0, & \text{if } e(t) \geq 0, \end{cases} \quad (12)$$

where the $y(t)$ is the control output.

Making use of the aggregation function, the fitness function is constructed as follows:

$$f = w_1 f_1 + w_2 f_2 + w_3 f_3 + w_4 f_4, \quad (13)$$

where w_1 , w_2 , w_3 , and w_4 are the weight coefficients, and $w_4 \gg w_1$.

```

for  $p$  in  $Pop(n)$ :
     $p\_dom\_q = [NULL]$ 
     $np = 0$ 
    for  $q$  in  $Pop(n)$ 
        if  $p > q$ 
             $p\_dom\_q.add(q)$  %  $p$  dominate  $q$  set
        else if  $p \leq q$ 
             $np = np + 1$  %  $p$  dominated by  $q$ , increment the counter
        end if
    end for
    if  $np = 0$  % the first rank of Pareto
         $p\_rank = 1$ 
    end if
     $F1.add(p)$ 
end for
 $F.add(F1)$ 
while  $F[i]$ 
     $Q = Null$ 
    for  $p$  in  $F[i]$ 
        for  $q$  in  $p\_dom\_q$ 
            if  $nq = 0$  % dominated by other individual
                 $q\_rank = i + 1$ 
                 $Q.add(q)$ 
            end if
        end for
    end for
     $F.add(Q)$ 

```

ALGORITHM 2: *Fast_nondominated_sort* function.

3.3. Multiobjective Optimization of PID Controller. According to Section 3.2, there are multiple objectives to be achieved for the PID parameters optimization. NSGA-II is based on Pareto solutions, measuring individual fitness according to their dominance property. The Pareto based approach is the most successful approach to realize multiobjective optimization and to search the true Pareto front [31]. The nondominated individuals in the population are regarded as the fittest, and the dominated individuals are assigned lower fitness values. This way, the number of dominated individuals will be counted as the fitness values instead of the value of objective function. To maintain the diversity in the Pareto solutions, NSGA-II introduced a measure of individual's density with respect to other individuals in the objective space and had an elitism mechanism and crowded comparison operator to preserve the diversity of population [32], which is shown in the pseudocode of *NSGA-II algorithm* (Algorithm 1).

In the *fast_nondominated_sort* function (Algorithm 2), all individuals in the first nondominated front are found first. In the first nondominated front, the domination counter is zero. We can have a traversal of the member of p_dom_q set. If the domination counter of q becomes zero, it belongs to the second nondominated front. Then, the procedure is repeated, and other nondominated fronts are found.

Comparing with the single-objective optimization, the effective space for searching is different. For example, the aggression function will restrict the fitness to a hyperplane, which is decided by coefficients w . In the multiobjective

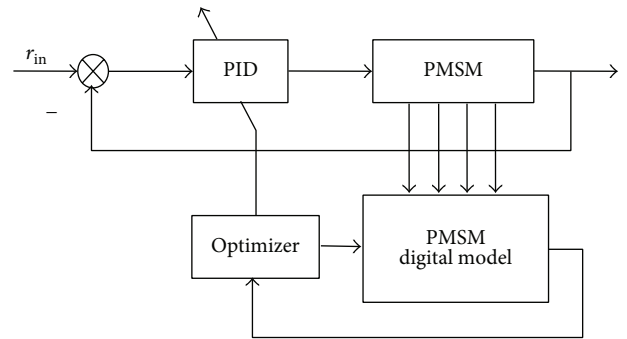


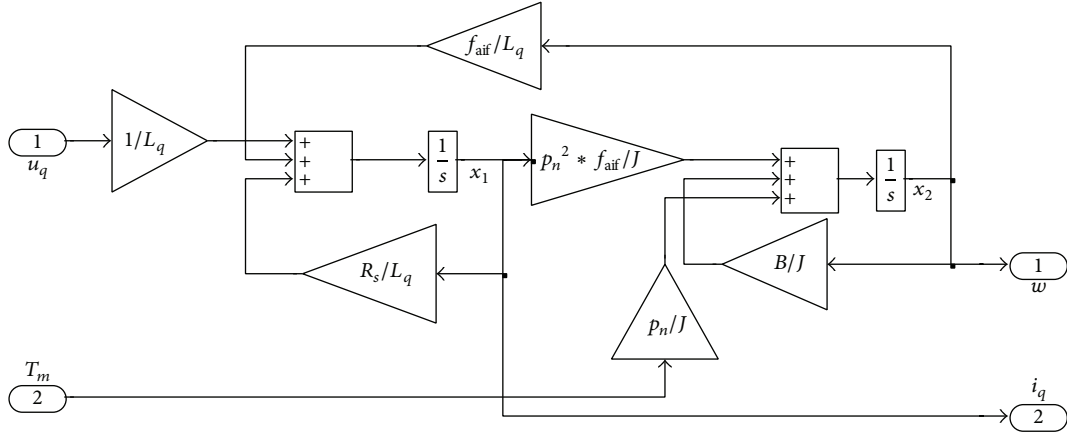
FIGURE 1: The diagram of PID controller optimization system.

optimization, the Pareto solutions can spread all over the space.

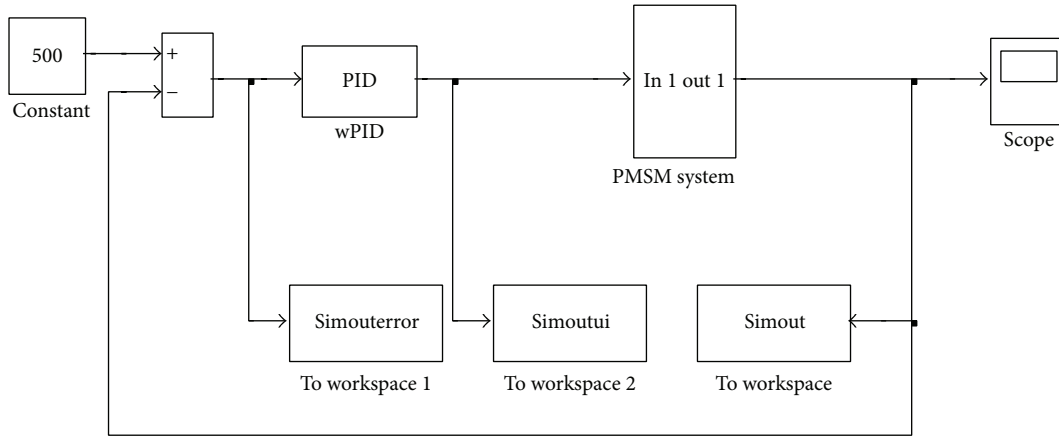
The diagram of the optimization system is shown as Figure 1. The parameters of PMSM model can be obtained by the least square estimation [14] in the practical control system. The aim of the NSGA-II in the PID controller tuning is the minimization of objectives f_i , such as overshoot, rising time, and system error.

4. Simulation

According to the state space equation (6), we can build the state space model of PMSM in MATLAB/simulink as



(a) MATLAB/simulink model of PMSM



(b) The diagram of PMSM control system

FIGURE 2: PMSM simulation model.

Figure 2(a). The parameters of PMSM are as follows; R_s is 0.9664, L_q is 0.00621, P_n is 4, J is 0.00033, B is 0.0001619, and ψ_f is 0.09382 according to motor.

The component of PMSM is encapsulated into a module. A speed controller added to the speed closed-loop. Figure 2(b) is the diagram of PMSM control system. The “simouterror,” “simoutui,” and “simout” units are used to record the simulation data for optimization.

4.1. PID Optimization Based on GA. In the simulation, the single-objective optimization methods were chosen to be compared to the multiobjective optimization approach to exhibit the robustness of multiobjective way. In the single-objective optimization method, GA is selected as the searching algorithm. The parameters of GA are as follows; the population size is 30, crossover probability is 0.9, and mutation probability is adaptive to individual fitness. The variable domain of K_p is $[0, 20]$; K_i and K_d are $[0, 1]$. The iteration number is 50. Higher fitness has lower mutation probability and lower fitness has higher mutation probability. w_1 , w_2 ,

w_3 , and w_4 of f_i are set according to the requirement of control system. w_1 is corresponding to the system error, w_2 is a weight coefficient of controller output, w_3 is for the system rising time, and w_4 is the penalty of overshoot. If we want a system without overshoot and have a small rising time, w_1 , w_3 , and w_4 will be set bigger, and w_2 smaller. If the controller output is limited, w_2 will be set bigger. Therefore, these parameters can be set according to the practical requirement. However, the coefficients are very difficult to decide due to dimensional problems. They are always experiential. In the simulation, w_1 is 1, w_2 is 0.1, w_3 is 2, and w_4 is 200. The processes of PID parameters optimization are shown as follows. Figure 3 shows the system output for each individual of GA. From Figure 3, we can see the system responses are different due to the different PID parameters at the beginning (Figure 3(a)). According to the fitness, GA will generate next generation with better fitness and have some mutation individuals (Figure 3(b)). Figure 4 shows the optimal system response of each iteration. From Figure 4, we can see that these system outputs have small overshoot and short rising

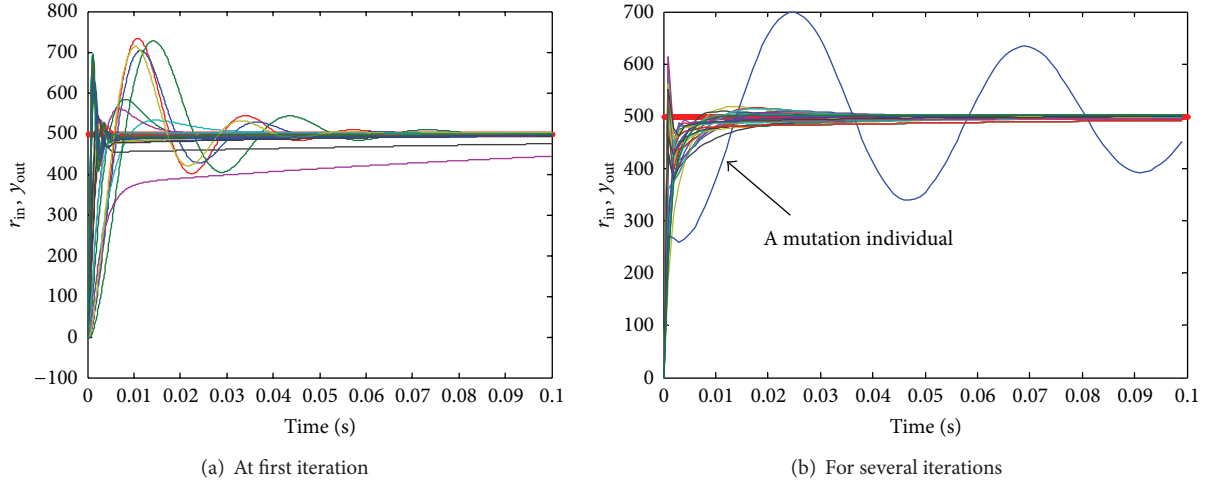


FIGURE 3: System output for each individual of GA.

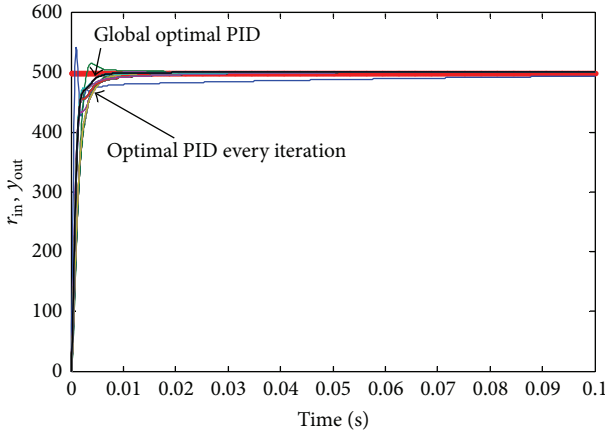


FIGURE 4: The optimal output of each iteration.

time. The optimal system output is the black line in Figure 4 $(K_p, K_i, K_d) = (0.1419, 10.3592, 0.0)$.

However, there is always a small lag in the speed loop due to characteristics of mechanical system, as shown in Figure 5. GA cannot search effective parameters of PID to make the system stable using the same coefficients w_i , though the lag time is very small. We cannot get the optimal result by GA, and the optimization results are not convergent. The adaptivity and robustness of GA is not very good. We should adjust the coefficients w_i again when the status of the system is changed. Nevertheless, it is difficult to select proper weights.

4.2. PID Optimization Based on NSGA-II. Aggression function method being based on single-objective optimization, the solution space is a hyperplane based on the weight values $(w_1, w_2, w_3, \text{ and } w_4)$. However, this is a multiple-objective problem. The aggression function method limits the domain of the solution. For NSGA-II method, we choose several typical system parameters, such as overshoot δ , peak time t_p , and stabilization time t_s . NSGA-II will minimize the selected parameters and search optimal PID parameters. We do not

need penalty values anymore such as f_4 of Section 4.1. We redefine the fitness function as follows:

$$\text{Minimize } f(x) = [f_1(x), f_2(x), f_3(x)],$$

$$f_1 = \delta,$$

$$f_2 = t_p,$$

$$f_3 = t_s.$$

$$x = (K_p, K_d, K_i).$$

(14)

The Pareto front is the solution space, which is not a plane. The Pareto front is shown in Figure 6. It is not a plane but a curved surface. For NSGAII, a parameter set is obtained. We can select a group of parameters according to our requirement.

When the system has structural change as in Figure 5, the NSGA-II also can obtain the Pareto front shown in Figure 7 as well as Figure 6. According to simulation, we can see that the multiobjective algorithm has a better robustness to realize the system optimization.

5. Conclusion

PID is a popular controller in industry. However, there are only few parameters, the parameters are difficult to determine due to the uncertain model of industry. The tuning of PID is always a multiobjective problem. Although GA provides a means to find optimal parameters of PID by aggression method, it has shortcomings of too much being experience-based, having poor robustness, and being sensitive to parameters. In this paper, we use a multiobjective approach to optimize the K_p , K_i , and K_d of PMSM controller, which provides more theoretical basis than aggregation function method. The simulations exhibit the advantages of the method. Although the work is promising, we have a lot of work to do, such as obtaining the accuracy model of PMSM and the instantaneity improvement of NSGA-II.

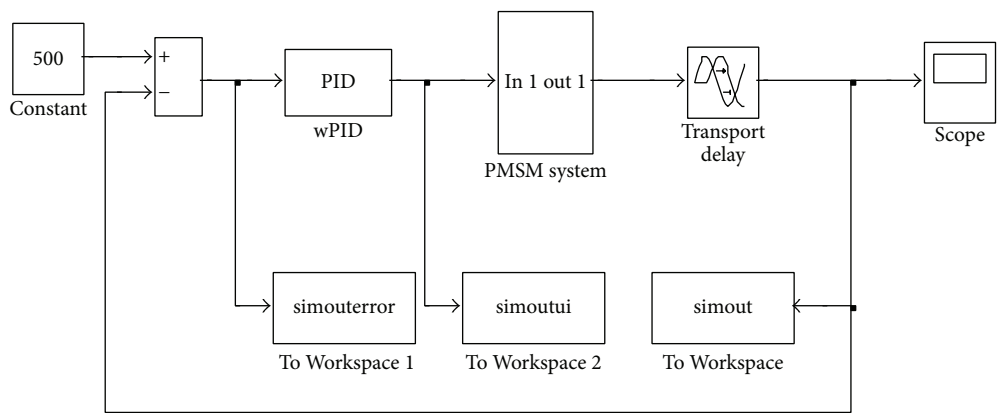


FIGURE 5: The diagram of PMSM control system with lag unit.

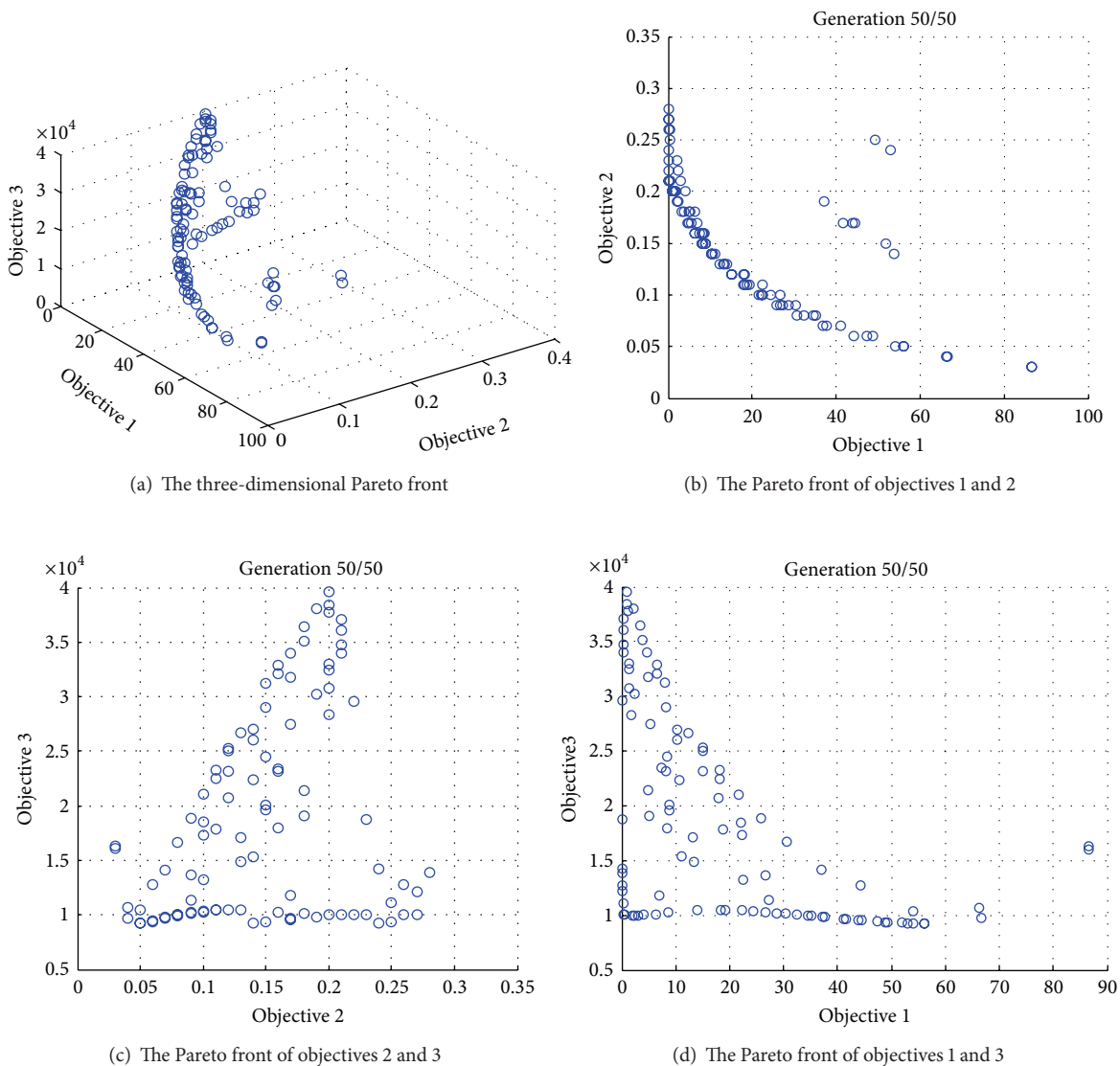


FIGURE 6: The optimization results of system 1.

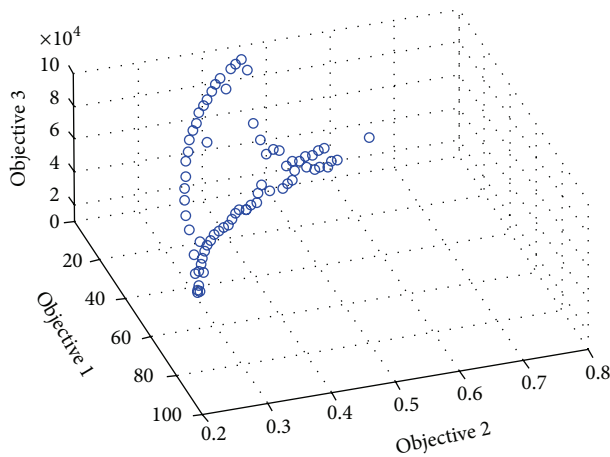


FIGURE 7: The optimization results of system with lag unit.

Conflict of Interests

The authors of the paper do not have a direct financial relation with the commercial identity mentioned in this paper that might lead to a conflict of interest for any of the authors.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 61174044.

References

- [1] M. Azizur Rahman and P. Zhou, "Analysis of brushless permanent magnet synchronous motors," *IEEE Transactions on Industrial Electronics*, vol. 43, no. 2, pp. 256–267, 1996.
- [2] M. Ooshima, A. Chiba, A. Rahman, and T. Fukao, "An improved control method of buried-type IPM bearingless motors considering magnetic saturation and magnetic pull variation," *IEEE Transactions on Energy Conversion*, vol. 19, no. 3, pp. 569–575, 2004.
- [3] K. Liu, Z. Q. Zhu, Q. Zhang, and J. Zhang, "Influence of nonideal voltage measurement on parameter estimation in permanent-magnet synchronous machines," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 6, pp. 2438–2447, 2012.
- [4] F. Caricchi, F. Crescimbeni, and O. Honorati, "Low-cost compact permanent magnet machine for adjustable-speed pump application," *IEEE Transactions on Industry Applications*, vol. 34, no. 1, pp. 109–116, 1998.
- [5] O. Wallmark, L. Harnefors, and O. Carlson, "Control algorithms for a fault-tolerant PMSM drive," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1973–1980, 2007.
- [6] A. Caponio, G. L. Cascella, F. Neri, N. Salvatore, and M. Sumner, "A fast adaptive memetic algorithm for online and offline control design of PMSM drives," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 37, no. 1, pp. 28–41, 2007.
- [7] A. Flah and L. Sbita, "A novel IMC controller based on bacterial foraging optimization algorithm applied to a high speed range PMSM drive," *Applied Intelligence*, vol. 38, no. 1, pp. 114–129, 2013.
- [8] F. Aymen, H. Kraiem, and S. Lassaad, "Robust high speed control algorithm for PMSM sensorless drives," in *Proceedings of the 9th International Multi-Conference on Systems, Signals and Devices (SSD '12)*, pp. 1–6, March 2012.
- [9] A. M. Kassem and A. A. Hassan, "Performance improvements of a permanent magnet synchronous machine via functional model predictive control," *Journal of Control Science and Engineering*, vol. 2012, Article ID 319708, 8 pages, 2012.
- [10] H. Zhang and J. Wang, "Combined feedback-feedforward tracking control for networked control systems with probabilistic delays," *Journal of the Franklin Institute*, vol. 351, no. 6, pp. 3477–3489, 2014.
- [11] H. Zhang, Y. Shi, and B. Mu, "Optimal H_∞ -based linear-quadratic regulator tracking control for discrete-time takagi-sugeno fuzzy systems with preview actions," *Journal of Dynamic Systems, Measurement and Control*, vol. 135, no. 4, Article ID 044501, 2013.
- [12] H. Zhang, Y. Shi, and M. Liu, " H_∞ step tracking control for networked discrete-time nonlinear systems with integral and predictive actions," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 337–345, 2013.
- [13] J. Zhang, J. Zhuang, H. Du, and S. Wang, "Self-organizing genetic algorithm based tuning of PID controllers," *Information Sciences*, vol. 179, no. 7, pp. 1007–1018, 2009.
- [14] A. Chouchaine, E. Feki, and A. Mami, "Stabilization using a discrete fuzzy PDC control with PID controllers and pole placement: application to an experimental greenhouse," *Journal of Control Science and Engineering*, vol. 2011, Article ID 537491, 9 pages, 2011.
- [15] S.-Z. Zhao, M. W. Iruthayarajan, S. Baskar, and P. N. Suganthan, "Multi-objective robust PID controller tuning using two lbests multi-objective particle swarm optimization," *Information Sciences*, vol. 181, no. 16, pp. 3323–3335, 2011.
- [16] A. Herreros, E. Baeyens, and J. R. Perán, "Design of PID-type controllers using multiobjective genetic algorithms," *ISA Transactions*, vol. 41, no. 4, pp. 457–472, 2002.
- [17] M. Marufuzzaman, M. B. I. Reaz, L. F. Rahman, and T. G. Chang, "High-speed current dq PI controller for vector controlled PMSM drive," *The Scientific World Journal*, vol. 2014, Article ID 709635, 9 pages, 2014.
- [18] M. R. Rani, H. Selamat, H. Zamzuri, and Z. Ibrahim, "Multi-objective optimization for PID controller tuning using the global ranking genetic algorithm," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 1, pp. 269–284, 2012.
- [19] M. A. Sahib, B. S. Ahmed, and M. Y. Potrus, "Application of combinatorial interaction design for DC servomotor PID controller tuning," *Journal of Control Science and Engineering*, vol. 2014, Article ID 576868, 7 pages, 2014.
- [20] I. Pan and S. Das, "Chaotic multi-objective optimization based design of fractional order PIAD μ controller in AVR system," *International Journal of Electrical Power & Energy Systems*, vol. 43, no. 1, pp. 393–407, 2012.
- [21] Q.-G. Chen, N. Wang, and S.-F. Huang, "Distribution population-based genetic algorithm for parameter optimization PID controller," *Acta Automatica Sinica*, vol. 31, no. 4, pp. 646–650, 2005.
- [22] H. Zhang, Y. Shi, and A. S. Mehr, "Robust H_∞ PID control for multivariable networked control systems with disturbance/noise attenuation," *International Journal of Robust and Nonlinear Control*, vol. 22, no. 2, pp. 183–204, 2012.

- [23] H. Zhang, Y. Shi, and A. S. Mehr, "Robust static output feedback control and remote PID design for networked motor systems," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 12, pp. 5396–5405, 2011.
- [24] L. Huang, N. Wang, and J.-H. Zhao, "Multiobjective optimization for controller design," *Acta Automatica Sinica*, vol. 34, no. 4, pp. 472–477, 2008.
- [25] W. Wojsznis, A. Mehta, P. Wojsznis, D. Thiele, and T. Blevins, "Multi-objective optimization for model predictive control," *ISA Transactions*, vol. 46, no. 3, pp. 351–361, 2007.
- [26] Q. Gao, J. Chen, L. Wang, S. Xu, and Y. Hou, "Multiobjective optimization design of a fractional order PID controller for a gun control system," *The Scientific World Journal*, vol. 2013, Article ID 907256, 8 pages, 2013.
- [27] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1994.
- [28] H. V. Hultmann Ayala and L. Dos Santos Coelho, "Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator," *Expert Systems with Applications*, vol. 39, no. 10, pp. 8968–8974, 2012.
- [29] S. Wang, "Windowed least square algorithm based PMSM parameters estimation," *Mathematical Problems in Engineering*, vol. 2013, Article ID 131268, 11 pages, 2013.
- [30] S. Wang, "ADRC and feedforward hybrid control system of PMSM," *Mathematical Problems in Engineering*, vol. 2013, Article ID 180179, 12 pages, 2013.
- [31] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: a survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 32–49, 2011.
- [32] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

