

# SISTEM KONTROL KECEPATAN MOTOR DC MENGGUNAKAN PID-MQPSO

Elkana Ishak<sup>1</sup>, Fachrudin Hunaini<sup>2</sup>, Faqih Rofii<sup>3</sup>

<sup>1,2,3</sup> Jurusan Teknik Elektro, Fakultas Teknik, Universitas Widyagama Malang

<sup>1</sup> zakariaelkana@yahoo.com, <sup>2</sup> fachrudin\_h@widyagama.ac.id, <sup>3</sup> faqih@widyagama.ac.id

---

## Abstrak

Proses penalaan pada pengontrolan kecepatan motor dc biasanya dilakukan secara manual dengan berdasarkan pada teori penalaan. Teori penalaan yang sering digunakan adalah metode Ziegler-Nichols disamping metode *Trial-Error*. Kendala yang dialami oleh metode penalaan ini adalah respon yang dihasilkan belum optimal. Penelitian ini menggunakan algoritma MQPSO (*Modified Quantum behaved Particle Swarm Optimization*) untuk optimasi parameter  $K_p$ ,  $K_i$ , dan  $K_d$  dari hasil penalaan metode Ziegler-Nichols. MQPSO merupakan pengembangan dari algoritma PSO (*Particle Swarm Optimization*) dan QPSO (*Quantum behaved Particle Swarm Optimization*). MQPSO memiliki perilaku pencarian partikel yang lebih detail karena memperhitungkan posisi partikel terhadap *mbest* dan *gbest*. Berdasarkan penelitian yang dilakukan, didapatkan hasil bahwa penalaan MQPSO menghasilkan parameter PID yang memiliki respon paling optimal dibandingkan penalaan QPSO, PSO, dan Ziegler-Nichols. MQPSO menghasilkan settling time sistem sebesar 41.0201ms, sedangkan QPSO sebesar 42.8276ms, PSO sebesar 43.7008ms, dan Ziegler-Nichols 61.8571ms.

**Kata kunci** : Motor DC, PID, MQPSO

---

## 1. Pendahuluan

Motor listrik DC merupakan salah satu jenis motor listrik yang sering digunakan. Hal ini dikarenakan motor DC memiliki karakteristik torsi yang linier terhadap kecepatan dan juga efisiensi yang tinggi (Zuhail, 1988). Pada beberapa aplikasi diperlukan ketepatan dan kecepatan yang presisi dari motor DC. Oleh karena itu diperlukan suatu mekanisme untuk meregulasi kecepatan dari motor DC agar sesuai dengan kebutuhan produksi.

Sistem kontrol PID merupakan mekanisme kontrol yang dapat diaplikasikan untuk pengaturan kecepatan motor DC. Kontrol PID menggunakan tiga komponen penting yaitu proporsional, integral, dan derivative. Ketiga komponen tersebut memiliki suatu konstanta ( $K_p$ ,  $K_i$ , dan  $K_d$ ) yang digunakan untuk mengatur respon dari sistem kontrol PID. (Astrom, *et al*, 1934).

Kennedy, *et al* (1995) menyatakan bahwa algoritma PSO (*Particle Swarm Optimization*) adalah algoritma optimasi berdasarkan perilaku kawanan burung dalam mencari makanan. Algoritma ini dikembangkan untuk menyelesaikan permasalahan optimasi. QPSO merupakan algoritma hasil pengembangan PSO konvensional dengan penambahan teorema mekanika kuantum (Sun *et al*, 2012). QPSO digunakan pada salah satu varian dari PSO untuk menentukan nilai konstanta parameter kontrol PID. Hal ini menyebabkan QPSO memiliki konvergensi global yang lebih baik dibandingkan PSO

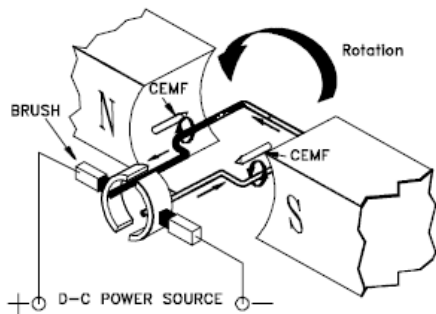
konvensional. Oleh karena itu, nilai optimum dapat diperoleh dengan proses iterasi yang lebih cepat. Kekurangan algoritma QPSO adalah nilai global optimum akan konvergen jika iterasi mendekati nilai tak hingga. Oleh karena itu diperlukan sebuah pengembangan dari QPSO yang disebut MQPSO. MQPSO merupakan algoritma optimasi hasil pengembangan QPSO konvensional dengan perubahan perhitungan *global attractor* (Jia *et al*, 2015).

Penelitian ini bertujuan untuk menerapkan algoritma MQPSO (*Modified Quantum behaved Particle Swarm Optimization*) pada proses penalaan PID untuk pengaturan kecepatan motor dc. Performa sistem hasil optimasi MQPSO akan dibandingkan dengan hasil optimasi PSO dan QPSO untuk mengukur tingkat efektivitas masing-masing algoritma optimasi. Pengukuran efektivitas algoritma ini berdasarkan performa sistem yang dihitung menggunakan ISE (*Integral Squared Error*).

## 2. Motor DC

Motor DC merupakan salah satu jenis motor yang membutuhkan suplai tegangan searah (DC) untuk diubah menjadi energi mekanik. Motor DC memiliki dua kumparan yang masing-masing menghasilkan energi magnet pada saat dialiri arus listrik (Zuhail, 1988). Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang

berputar). Skema dari motor dc yang paling sederhana dapat dilihat pada gambar berikut.



Gambar 1. Skema Motor DC sederhana

### 3. MQPSO

Algoritma MQPSO (*Modified Quantum behaved Particle Swarm Optimization*) merupakan pengembangan algoritma QPSO konvensional. MQPSO digunakan karena pada algoritma QPSO nilai global optimal hanya dapat dicari jika iterasi mendekati nilai tak terhingga. Pada aplikasinya nilai dari total iterasi merupakan bilangan real dan tidak dapat digantikan oleh bilangan tak hingga. Hal ini akan mengakibatkan kemampuan pencarian nilai global dari QPSO terbatas sesuai dengan jumlah iterasi.

Pada MQPSO digunakan penambahan variabel  $g_{best}$  pada persamaan untuk menghitung *global attractor* (Sheng *et al*, 2015). Perhitungan menggunakan rata-rata antara jumlah selisih masing-masing antara  $g_{best}$  dan  $m_{best}$  dengan posisi partikel untuk mendapatkan hasil yang optimal. Perhitungan posisi partikel dari MQPSO menggunakan rumus sebagai berikut.

$$x_i(t+1) = p_i \pm \alpha \left| \frac{(g_{best} - x_i(t)) + (m_{best} - x_i(t))}{2} * \ln(1/u) \right| u \approx U(0,1) \quad (1)$$

Persamaan (1) menunjukkan nilai laju konvergensi MQPSO dipengaruhi oleh variabel  $\alpha$  (*expansion-contraction factor*). Pada MQPSO nilai  $\alpha$  mengalami penurunan berdasarkan nilai iterasi. Hal ini mengakibatkan proses variasi antara  $x(t)$  dan  $x(t+1)$  pada iterasi awal menghasilkan *diversity* yang besar dan semakin lama semakin kecil. Proses pencarian ini mengakibatkan MQPSO melakukan pencarian secara global pada awal iterasi dan semakin banyak iterasi pencarian dilakukan semakin lokal (Jia *et al*, 2015).

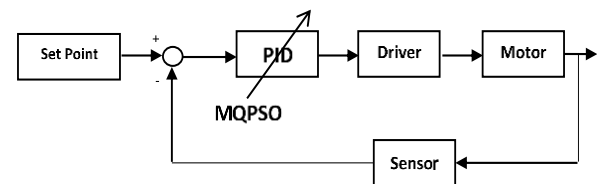
$$\alpha = (\alpha_0 - \alpha_1) \times \left( \frac{iter}{maxiter} \right) + \alpha_1 \quad (2)$$

Persamaan (2) menunjukkan perhitungan untuk nilai variabel  $\alpha$ . Pada persamaan diatas nilai awal variabel  $\alpha$  adalah sebesar  $\alpha_0$  dan pada akhir iterasi sebesar  $\alpha_1$ . Laju perubahan nilai variabel  $\alpha$

dipengaruhi oleh nilai iterasi (*iter*) saat ini dan nilai maksimum iterasi (*maxiter*) .

### 4. Sistem Kontrol PID-MQPSO

PID merupakan skema kontrol yang terdiri dari tiga aksi koreksi yaitu proportional (P), integral (I), dan derivative (D) (Zhong, 2006). Masing-masing aksi koreksi memiliki karakteristik tersendiri terhadap error. Karakteristik kontrol proportional ialah memiliki stabilitas terhadap gangguan yang kecil, namun tidak dapat menghilangkan error pada keadaan tunak. Keadaan ini dapat ditanggulangi dengan menambahkan kontrol integral yang memiliki karakteristik menghilangkan error pada keadaan tunak. Penambahan kontrol integral akan mengakibatkan settling time dari kontroler semakin lambat karena kontroler integral menghasilkan aksi koreksi terhadap akumulasi error terhadap waktu. Untuk mempercepat settling time dari kontroler maka perlu ditambahkan kontrol derivative. Kontrol ini bereaksi terhadap perubahan error yang terjadi sehingga respon dari kontroler dapat dipercepat (Hunaini *et al*, 2016). Skema sistem kontrol PID-MQPSO dapat dilihat pada Gambar 2.



Gambar 2. Sistem Kontrol PID-MQPSO

Kontroler PID akan menghasilkan keluaran berupa sinyal MV (*manipulated variable*) yang merupakan hasil kalkulasi keluaran tiap aksi kontrol. Persamaan ini dapat ditulis pada bidang waktu sebagai berikut.

$$u(t) = MV(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (3)$$

Variabel  $K_p$ ,  $K_i$ , dan  $K_d$  pada persamaan diatas akan ditentukan oleh algoritma MQPSO. Hal ini bertujuan untuk mencari kombinasi nilai  $K_p$ ,  $K_i$ , dan  $K_d$  yang menghasilkan performa sistem kontrol yang paling optimal.

### 5. Pemodelan Sistem

Pada proses perencanaan sistem kontrol perlu diketahui karakteristik dari fungsi alih sistem yang akan dikontrol. Sistem yang ada pada penelitian ini terdiri dari input berupa sinyal PWM, rangkaian driver motor, motor dc yang terhubung dengan generator dc, dan rangkaian rotary encoder.

Berdasarkan komponen penyusunnya maka sistem memiliki karakteristik yang merupakan

gabungan dari karakteristik masing-masing. Oleh karena itu digunakan pemodelan berdasarkan software *System Identification Toolbox* pada MATLAB (Ljung, 2012).

Pada proses identifikasi fungsi alih ini digunakan *2-pole* dan *2-zero*. Sehingga akan dihasilkan persamaan orde dua sebagai berikut.

$$F(s) = \frac{0.285s^2 + 96.41s + 384.2}{s^2 + 65.17s + 246.4} \quad (4)$$

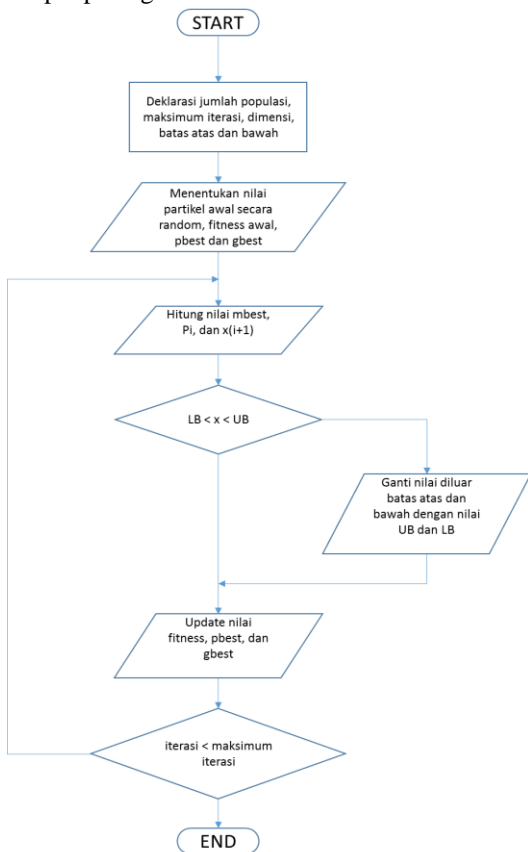
Fungsi alih diatas memiliki nilai FTE (*Fit To Estimation*) sebesar 91,71% berdasarkan hubungan antara input dan output sistem. Oleh karena itu fungsi alih diatas dapat digunakan sebagai fungsi alih *plant* dalam proses optimasi MQPSO.

### 6. Optimasi PID dengan MQPSO

Parameter yang PID yang dioptimasi oleh MQPSO adalah  $K_p$ ,  $K_i$ , dan  $K_d$ . Parameter *fitness* yang digunakan untuk mengetahui performa sistem adalah fungsi *Integral Squared Error* (ISE).

$$ISE = \int e^2 dt \quad (5)$$

Proses optimasi pada penelitian ini dilakukan pada MATLAB berdasarkan diagram alir yang terdapat pada gambar 3.



Gambar 3. Diagram Alir PID-MQPSO

Nilai awal  $K_p$ ,  $K_i$ , dan  $K_d$  ditentukan menggunakan metode penalaan kurva reaksi (Ziegler *et al*, 1942). Nilai awal ini digunakan sebagai dasar untuk menentukan variabel optimasi yang akan dilakukan. Untuk mengetahui performa dari MQPSO pada penelitian ini digunakan penalaan Ziegler-Nichols, PSO, dan QPSO sebagai pembanding. Parameter masing-masing metode yang digunakan adalah sebagai berikut.

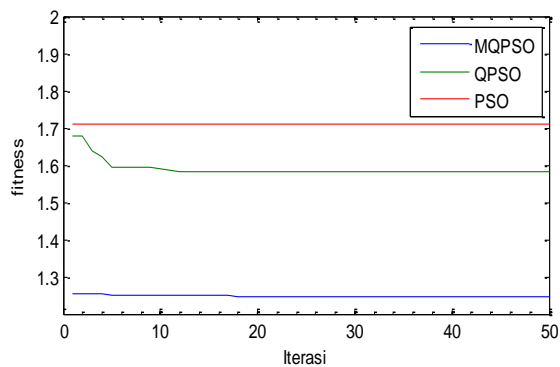
Tabel 1. Parameter Berbagai Penalaan PID

Metode	Jumlah Partikel	Jumlah Iterasi
Z-N		
PSO	10	50
QPSO	10	50
MPSO	10	50

Setelah proses optimasi dilakukan maka dapat diketahui nilai masing-masing fitness dan parameter PID hasil optimasi.

Tabel 2. Parameter PID Hasil Optimasi

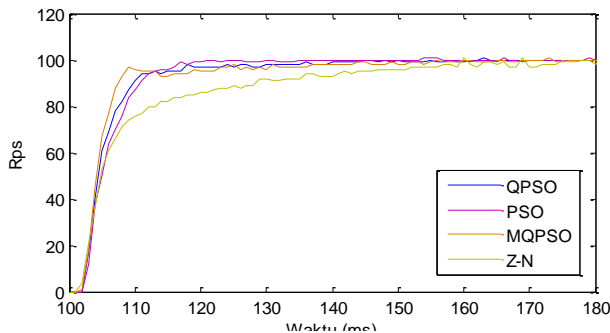
Metode	Fitness	$K_p$	$K_i$	$K_d$	Settling Time (ms)
Z-N	5.4467	1.29	11.34	0.11	61.8571
PSO	1.7083	1.2460	20.7640	0.0868	43.7008
QPSO	1.5807	1.4731	20.3947	0.0496	42.8276
MPSO	1.2477	1.9989	20.3046	0.0960	41.0201



Gambar 4. Laju perubahan fitness PSO, QPSO, dan MQPSO

Gambar 4 menunjukkan perbandingan laju perubahan *fitness* antara masing-masing algoritma optimasi. Hasil yang diperoleh yaitu nilai *fitness* akhir pada MQPSO paling kecil dibandingkan algoritma lainnya. Pada MQPSO diperoleh nilai *fitness* akhir sebesar 1.2477, sedangkan pada QPSO sebesar 1.5807, PSO sebesar 1.7083, dan Ziegler-Nichols sebesar 5.4467. Hal ini menunjukkan bahwa secara simulasi, algoritma MQPSO memiliki kemampuan optimasi yang lebih baik dibandingkan QPSO, dan PSO.

Nilai parameter hasil simulasi akan diaplikasikan pada *plant* sistem kontrol kecepatan motor dc untuk mengetahui efektivitas algoritma optimasi. Pengujian nilai parameter PID pada *plant* menggunakan variabel settling time sebagai indikator performa sistem.



**Gambar 5.** Perbandingan Respon Step

Gambar 5 menunjukkan grafik perbandingan respon step dari masing-masing metode penalaan. Berdasarkan grafik tersebut dapat diketahui bahwa MQPSO memiliki respon yang paling cepat dibandingkan QPSO, PSO, dan Ziegler-Nichols.

**Tabel 3. Pengujian Parameter Hasil Optimasi**

Metode	Fitness	Kp	Ki	Kd	Settling Time (ms)
Z-N	5.4467	1.29	11.34	0.11	6.1871
PSO	1.7083	1.2460	20.7640	0.0868	4.3708
QPSO	1.5807	1.4731	20.3947	0.0496	4.2876
MPQSO	1.2477	1.9989	20.3046	0.0960	4.1201

Tabel 3 menunjukkan parameter kontrol dan performa dari masing-masing metode penalaan. Parameter kontrol terdiri dari tiga variable yaitu :Kp, Ki, dan Kd. Sedangkan performa sistem terdiri dari nilai *fitness* dan *settling time*. Berdasarkan tabel diatas diketahui bahwa MQPSO dapat mengoptimasi parameter PID untuk menghasilkan respon yang paling optimal dibandingkan dengan metode optimasi lainnya. Hal ini ditandai dengan nilai *fitness* yang paling kecil yaitu sebesar 1.2477 dan *settling time* yang cepat sebesar 4.1201ms.

**7. Kesimpulan dan Saran**

Berdasarkan simulasi optimasi MQPSO yang telah dilakukan maka diperoleh nilai parameter PID yang optimal yaitu Kp sebesar 1.9989, Ki sebesar 20.3046, dan Kd sebesar 0.0960 dengan nilai fitness 1.2477 dan settling time 41.0201ms.

Pengujian pada *plant* menunjukkan bahwa algoritma MQPSO memiliki efektifitas yang paling baik. MQPSO menghasilkan parameter PID yang memiliki respon step yang paling cepat dibandingkan dengan penalaan Ziegler-Nichols, PSO, dan QPSO. Settling time yang dihasilkan MQPSO sebesar 4.1201ms, yaitu lebih cepat 33.69% dibandingkan penalaan manual sebesar 6.1871ms. Sedangkan PSO dan QPSO masing-masing sebesar 4.378ms dan

4.2876ms. Hal ini berarti settling time PSO dan QPSO hanya 29.36% dan 30.77% lebih cepat dibandingkan penalaan manual.

Untuk meningkatkan performa sistem dapat digunakan peralatan akuisisi data yang memiliki waktu sampling yang lebih cepat, dan dapat digunakan filter untuk menghilangkan noise pembacaan sensor. Penentuan parameter populasi dan jumlah iterasi mempengaruhi hasil akhir optimasi dari sistem ini.

**Daftar Pustaka**

Astrom, K. & Hagglund, T. (1934): *PID Controller : Theory, Design, and Tuning 2nd edition*.  
 Hunaini,F., Robandi, I., Sutantra, N. (2014): *Optimization Control System using the Quantum Behaved Particle Swarm Optimization on Vehicle Steering Control System with Steer-by-Wire System*, Jurnal Teknologi, 71:2, 2014, 91–98.  
 Hunaini,F., Robandi, I., Sutantra, N. (2016): *Optimization of automatic steering control on vehicle with steer by wire system using particle swarm optimization*, Turk J Elec Eng & Comp Sci,24; 2: 541 – 557.  
 Jia, P., Duan, S. & Yan, J. (2015): *An Enhanced Quantum-Behaved Particle Swarm Optimization Based on a Novel Computing Way of Local Attractor*, Information, pp.633-649.  
 Kennedy, J. & Eberhart, R. (1995): *Particle swarm optimization*, Proceedings of the IEEE International Conference on Neural Networks vol. 4, pp. 1942-1948.  
 Ljung, L. & Singh, R. (2012): Version 8 of the MATLAB system identification toolbox. *IFAC Proceedings Volumes*, 45(16), pp. 1826-1831  
 Sheng, X., Xi, M., Sun, J. & Xu, W. (2015): *Quantum-behaved Particle Swarm Optimization with Novel Adaptive Strategies*, Journal of Algorithms & Computational Technology pp.143-162.  
 Sun, J., Bin, F., & Wenbo, X. (2004): *Particle Swarm Optimization With Particles Having Quantum behavior*, In Proceedings of IEEE Congress on Evolutionary Computation, vol. 1, pp. 325-331.  
 Ziegler, J.G. & Nichols, N.B. (1942): *Optimum settings for automatic controllers*, transactions of the ASME 64, pp. 759–768.  
 Zuhail. (1988): *Dasar Teknik Tenaga Listrik dan Elektronika Daya*.Jakarta.