

Implementasi Kontrol PID pada Kopel Motor DC dengan Menggunakan Filter Kalman

Salima Nurrahma^{1,*}, Tegar Arif Berbudi¹, Muhammad Rakha Firdaus¹, Galang Izzaulhaq¹, Imroatul Hudati¹

¹Departemen Teknik Elektro dan Informatika, Universitas Gadjah Mada; tegar.a.b@mail.ugm.ac.id, rakhafdr30@mail.ugm.ac.id, galangizzaulhaq@mail.ugm.ac.id, imroatul.hudati@ugm.ac.id

*Korespondensi: salima.nurrahma@mail.ugm.ac.id;

Abstract – This paper describes a DC motor coupling system that uses several control methods including linear quadratic regulator (LQR), linear quadratic tracking (LQT), PID and uses Kalman filter. The methods used aim to obtain stable results from the DC motor coupling system used. To get the appropriate results, several stages were carried out such as hardware design where this process was carried out by pairing all the components used including the power supply, DC motor driver, Arduino UNO, potentiometer, and the two coupled DC motors. This system uses an input voltage of 12 V and will be processed by the motor driver to drive the DC motor. Based on the results that have been obtained, it can be observed that when the system uses the Kalman filter and uses the PI control the system response results have a small steady state error (ess). This is obtained because when the Kalman filter is added, it can reduce noise in the DC motor voltage readings and produces smoother output, and is less affected by high noise. After conducting experiments with the Kalman filter, the PID control was gradually added to its usage. It was found that when the P control was used, it provided the best response that reached the set point value with an RMSE of 0.1071 and an ess of +0.0874.

Keywords – DC motor coupling, PID, Kalman filter, LQR, LQT

Intisari – Artikel ini menjelaskan mengenai sistem kopel motor DC yang menggunakan beberapa metode kendali meliputi *linear quadratic regulator* (LQR), *linear quadratic tracking* (LQT), PID serta menggunakan *Kalman filter*. Metode - metode yang digunakan bertujuan untuk mendapatkan hasil yang stabil dari sistem kopel motor DC yang digunakan. Untuk mendapatkan hasil yang sesuai maka dilakukan beberapa tahapan seperti perancangan perangkat keras di mana proses ini dilakukan dengan memasang seluruh komponen yang digunakan meliputi *power supply*, *driver motor DC*, *Arduino UNO*, *potensio* dan kedua motor DC yang telah dikopel. Sistem ini menggunakan *input* tegangan sebesar 12 V dan kemudian diolah oleh *driver motor* untuk menggerakkan motor DC. Berdasarkan hasil yang telah didapat, dapat diamati bahwa saat sistem menggunakan Kalman filter dan menggunakan kendali PI hasil respons sistem memiliki *error steady state* (ess) yang kecil. Hal ini diperoleh karena pada saat ditambahkan, hal tersebut dapat mengurangi *noise* dalam pembacaan nilai tegangan motor DC dan menghasilkan keluaran yang lebih halus dan bebas dari *noise* tinggi. Setelah dilakukan percobaan dengan filter Kalman, selanjutnya ditambahkan kendali PID secara bertahap dalam penggunaannya. Didapatkan pada saat kendali P digunakan memberikan respons paling baik yang mencapai nilai *set point* dengan RMSE sebesar 0,1071 dengan ess sebesar +0,0874.

Kata kunci – kopel motor DC, PID, filter Kalman, LQR, LQT

I. PENDAHULUAN

Kopel motor DC terdiri dari dua motor DC yang digabungkan menjadi satu di mana satu motor DC berfungsi sebagai penggerak dan yang lain sebagai sensor untuk pembacaan kecepatan dan tegangan *output* yang dihasilkan oleh sistem. Pada sistem kontrol tegangan keluaran motor DC membutuhkan perangkat keras yang terdiri dari catu daya DC, laptop, *DC motor driver*, *potensio* dan kopel untuk menggabungkan kedua motor.

Sistem kopel motor DC memiliki tegangan 12 V dengan *input* yang diatur oleh *potensio* dan menggunakan kendali untuk mendapatkan hasil *output* yang sesuai. Pembacaan hasil *output* yang telah dikontrol oleh beberapa metode akan dibaca oleh Arduino sehingga dapat dibandingkan metode mana yang sesuai dan benar.

Motor DC memiliki kekurangan dalam hal pengaturan keluaran kecepatan atau tegangan dari motor DC. Maka dari itu untuk menekan tingginya *overshoot* perlu dilakukan pengaturan sehingga kecepatan dan tegangan keluaran motor DC sesuai dengan nilai *set point* yang diinginkan dengan *rise*

time yang tinggi dan *overshoot* yang kecil. Pengaturan yang diperlukan berupa kendali *linear quadratic regulator* (LQR), *linear quadratic tracking* (LQT), *proportional-integral-derivative* (PID) dan filter Kalman. Motor DC bekerja secara non linier terutama pada saat diberi variasi beban. Terdapat beberapa penelitian pengendalian motor DC yang telah dilakukan, seperti pada penelitian [1]. Pengendalian kecepatan motor DC dilakukan dengan menggunakan metode kendali PID. Pengendali konvensional seperti PID umum digunakan karena memiliki performa yang baik dalam mengendalikan sistem linier. Kelemahan pada metode tersebut adalah sulit disesuaikan dengan perubahan kondisi sistem yang cepat atau tugas kendali yang kompleks. Penelitian hanya dilakukan dengan menggunakan simulasi sehingga tidak memperhatikan kondisi sesungguhnya dari kecepatan motor DC. Pada penelitian [2], dilakukan pengendalian kecepatan motor DC dengan menggunakan kendali adaptif. Pada penelitian ini, hanya dilakukan pada simulasi dengan menganalisis model motor DC. Kelemahan pada metode ini adalah pengujian dilakukan dengan menggunakan analisis motor DC sehingga membuat model referensi yang diberikan tidak sesuai dengan kondisi

sesungguhnya dari motor DC. Penelitian [3] tentang kendali motor DC, dengan menggunakan metode *fuzzy logic*, di mana data yang digunakan diambil dari PCI 6221. Kelemahan dari metode ini adalah penentuan *membership function* yang cukup rumit. Peneliti harus cukup berpengalaman agar dapat menentukan *membership function* yang sesuai dengan sistem yang akan dikendalikan.

Pada penelitian ini, dilakukan kopel motor DC menggunakan kendali PID untuk mengatur mengendalikan kecepatan motor DC dengan masukan tegangan yang dikendalikan dengan potensiometer. Selain itu, penelitian kali ini juga menggunakan filter Kalman untuk mendapatkan hasil respons yang sesuai.

II. DASAR TEORI

A. Discrete PID Controller

Kontrol PID merupakan kontrol dengan mekanisme umpan balik yang sering digunakan di dunia industri. PID berfungsi untuk mengatur putaran motor yaitu kontrol proporsional yang memiliki kelebihan *rise time* yang cepat, kontrol integral untuk memperkecil nilai *error* dan kontrol derivatif untuk meredam *overshoot* [4].

Kendali PID digunakan dalam sistem *loop* tertutup yang memiliki umpan balik dari *output* sistem untuk mencapai respon yang diinginkan. Kontrol PID terdiri dari proporsional, integral dan derivatif dan saling memiliki pengaruh terhadap yang lain [4].

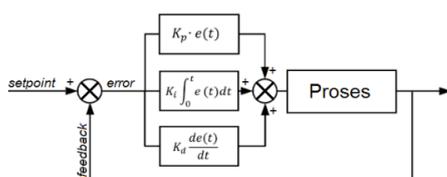
Pada Gambar 1 ditunjukkan bahwa jika ketiga parameter tersebut dikombinasikan secara optimal akan menghasilkan respons keluaran yang baik. PID memiliki masukan berupa sinyal kesalahan $e(t)$ dan keluaran pengendali merupakan sinyal kontrol $u(t)$. Hubungan antara masukan dan keluaran pengendali PID pada (1).

$$u(t) = K_p \left(e(t) + \frac{1}{\tau_i} \int_0^t e(t) dt + \tau_d \frac{de(t)}{dt} \right) \quad (1)$$

Hasil Laplace pada (1) menjadi (2).

$$U(s) = K_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right) E(s) \quad (2)$$

Pada (2), K_p merupakan penguatan proporsional dan τ_i merupakan waktu integral dan τ_d adalah waktu diferensial. Ketiga parameter tersebut dapat ditentukan nilainya. Sehingga fungsi alih pengendali PID ditunjukkan oleh (3).



Gambar 1. Blok diagram pengendali PID

Tabel 1. Pengaruh tiap parameter PID

Respons CL	Rise Time	Overshoot	Settling Time	Error Steady State
Kp	Naik	Meningkat	Perubahan Kecil	Menurun
Ki	Menurun	Meningkat	Meningkat	Hilang
Kd	Perubahan Kecil	Menurun	Menurun	Perubahan Kecil

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right) \quad (3)$$

Tipe pengendali proporsional K_p akan memberikan pengaruh pengurangan nilai *rise time* akan tetapi tidak menghapus nilai *error steady state* [5]. Kontrol integral akan memberikan pengaruh menghilangkan *error steady state* akan tetapi berakibat menurunkan respons *transient*. Sedangkan pengendali derivatif Kd akan memberikan pengaruh meningkatkan stabilitas sistem dan mengurangi nilai *overshoot*. Efek dari setiap pengendali dalam sistem *closed loop* hal ini dapat dilihat pada Tabel 1.

Pada pengendali PID memiliki tiga parameter yang mempengaruhi hasil dari kontrol yang dirancang. Berdasarkan Tabel 1 didapatkan hasil pengaruh beberapa parameter seperti parameter Kp ditambahkan nilainya atau diperbesar maka akan berpengaruh meningkatkan nilai *rise time* dan *overshoot* pada hasil kontrol. Sebaliknya pada nilai *settling time* dan nilai *error steady state* akan menurun. Jika menambahkan nilai parameter Ki maka nilai *overshoot* dan *settling time* akan meningkat, sedangkan nilai *rise time* akan menurun dan nilai *error steady state* akan hilang. Terakhir jika menambahkan nilai Kd jika ditambahkan akan memberikan pengaruh pada nilai *overshoot* dan *settling time* akan menurun, sedangkan nilai *rise time* dan *error steady state* akan berubah sedikit.

B. Filter Kalman

Filter Kalman merupakan metode yang digunakan untuk melakukan estimasi suatu nilai [6]. Kalman Filter ini digunakan untuk meminimalisasi *error* estimasi *mean squared* pada linier stokastik menggunakan *noisy linear sensors*. Metode ini sering disebut dengan *linear quadratic estimator (LQE)* karena meminimalisasi fungsi kuadrat dari eror estimasi pada sebuah sistem dinamis linear dengan *white measurement* dan *disturbance noise*.

Filter Kalman lebih dipergunakan bagi permasalahan umum estimasi pada suatu keadaan dari proses kontrol diskrit yang dibangun dari penurunan persamaan stokastik linier [7]. Filter Kalman membahas masalah umum untuk memperkirakan keadaan proses waktu diskrit terkontrol yang diatur oleh persamaan linier stokastik (4).

$$x_k = Cx_{k-1} + Du_k + w_k \quad (4)$$

Pada waktu ke- k dengan x_{k-1} menyatakan vector keadaan dan u_k sebagai vektor dengan model pengukuran $z \in R$ dengan,

$$z_k = H x_k + v_k \tag{5}$$

Variabel acak w_k dan v_k menyatakan proses dan pengukuran *noise*. Masing-masing diasumsikan independen dan memiliki probabilitas distribusi normal. Matriks *covariant noise* dapat berubah pada setiap kali pengukuran, namun dalam hal ini *noise* diasumsikan konstan. C, D dan H masing-masing merupakan matriks yang elemennya adalah koefisien dari komponen masing-masing vektor.

Filter Kalman mengestimasi suatu proses dengan menggunakan suatu bentuk kontrol *feedback* berulang. Persamaan pada filter Kalman dibagi dalam dua tahap persamaan pembaruan waktu (*time update*) dan persamaan pembaruan pengukuran (*measurement update*) [8].

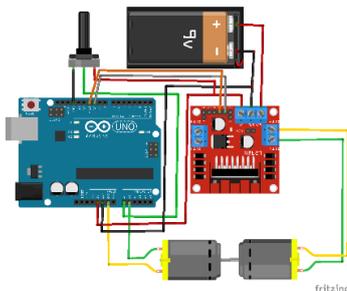
Persamaan pembaharuan waktu yang ada akan digunakan untuk memproyeksikan keadaan saat ini dan estimasi *covariant error*-nya untuk mendapatkan estimasi *priori* untuk langkah selanjutnya.

III. METODOLOGI

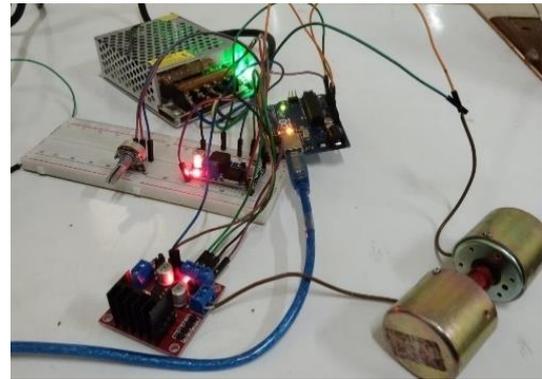
A. Perancangan Perangkat Keras

Perancangan perangkat keras dilakukan dengan membuat *wiring diagram* kopel motor DC menggunakan bantuan dari *software* Fritzing. Gambar 2 menunjukkan *wiring diagram* dari sistem kopel motor yang menggunakan Arduino UNO sebagai pengontrol sistem. Gambar 3 menunjukkan rancangan perangkat keras dari kopel motor DC.

Arduino dan *driver motor* akan mendapatkan suplai tegangan dari sumber tegangan sebesar 12 V. *Driver motor* digunakan untuk menggerakkan motor DC pertama, di mana besar PWM untuk *driver* motor tersebut akan diatur menggunakan potensiometer. Pada saat motor DC pertama berputar, maka *shaft* motor DC kedua akan ikut berputar dikarenakan kedua motor sudah dalam keadaan terkopel sehingga pada saat motor DC kedua bergerak, akan dihasilkan tegangan pada kedua terminalnya dan tegangan tersebut akan dibaca oleh Arduino. Data tegangan motor DC kedua tersebut akan berfungsi sebagai data *output* yang akan digunakan untuk keperluan identifikasi sistem.



Gambar 2. *Wiring diagram* motor DC



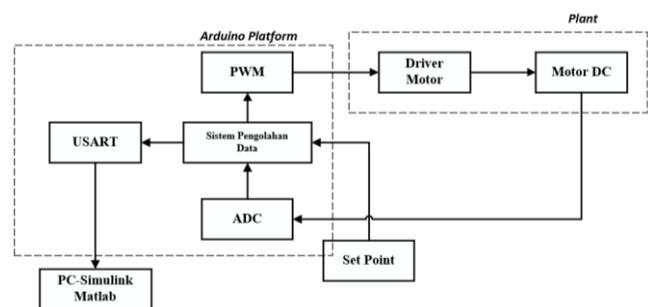
Gambar 3. Perangkat keras kopel motor DC

B. Perancangan Sistem Akuisisi Data

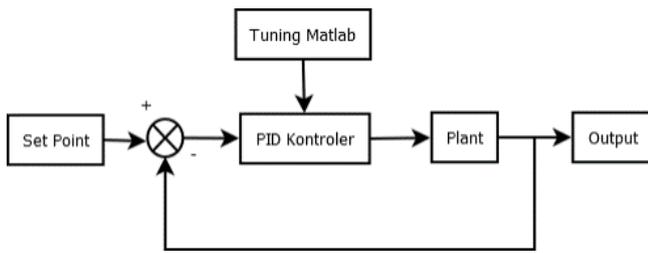
Dalam penelitian ini dibutuhkan suatu sistem akuisisi data yang diperlukan untuk melakukan pengambilan data dari sistem menuju ke komputer. Hal ini diperlukan karena untuk melakukan identifikasi sistem diperlukan data masukan dan keluaran dari sistem tersebut, sehingga diperlukan rancangan sistem akuisisi data yang mana diagram blok rancangan sistem akuisisi data dapat dilihat pada Gambar 4.

Dari rancangan blok diagram di atas dapat diketahui bahwa akuisisi data pada penelitian ini menggunakan Arduino sebagai sistem untuk pengontrol dan pengolah data pada sistem. Arduino akan membaca tegangan pada potensiometer sebagai nilai *set point*. Nilai *set point* tersebut digunakan untuk mengatur besarnya nilai PWM yang akan diberikan ke *driver* motor untuk mengatur kecepatan putaran motor DC pertama.

Dengan konfigurasi kopel antara kedua motor, Arduino akan membaca tegangan yang dihasilkan oleh motor DC kedua secara serial. Untuk menyimpan data serial tersebut, digunakan *software* MATLAB yang telah terhubung secara serial dengan Arduino sehingga hasil data yang telah didapatkan oleh Arduino akan dicatat oleh MATLAB sehingga MATLAB akan menyimpan data, baik data masukan potensiometer atau data keluaran tegangan motor kedua sehingga data-data tersebut dapat diolah selanjutnya untuk keperluan identifikasi sistem.



Gambar 4. Diagram blok rancangan sistem akuisisi data [9]



Gambar 5. Diagram blok kontrol PID

C. Perancangan Pengendali PID

Sebelum menentukan nilai parameter pengontrol PID, sebuah sistem terlebih dahulu harus ditinjau bagaimana karakteristik dari sistem tersebut [10]. Nilai-nilai parameter dari pengendali PID akan dicari secara *trial and error*. Perubahan pada setiap nilai parameter akan memengaruhi parameter yang lainnya sehingga pada penelitian ini digunakan bantuan dari MATLAB untuk melakukan *tuning* nilai [11] PID secara otomatis. Gambar 5 menunjukkan rancangan blok diagram kontrol PID.

Dari blok diagram di atas dapat diketahui bahwa nilai parameter PID akan diatur menggunakan fitur pengaturan PID pada MATLAB. Blok *Plant* merupakan hasil identifikasi sistem yang telah dimodelkan dalam bentuk transfer fungsi. Data *output* akan menjadi nilai *feedback* dan akan dibandingkan dengan nilai *set point*. Pengendali PID akan menghitung nilai *error* dari beda antara nilai *set point* yang telah ditetapkan dengan nilai *output* pada sistem.

IV. HASIL DAN PEMBAHASAN

A. Filter Kalman

Sebelum diberikan filter Kalman, tegangan keluaran yang diukur pada motor DC kedua memiliki banyak *noise*. Adanya *noise* ini dapat mengganggu proses umpan balik dalam perhitungan sinyal kontrol. Gambar 6 menunjukkan grafik tegangan keluaran dari motor DC kedua selama 1000 kali *sampling*.

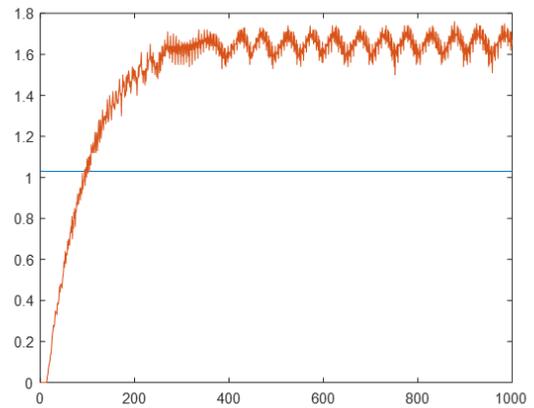
Filter Kalman dapat mengurangi *noise* pada pembacaan nilai tegangan motor DC [8]. Filter Kalman mengurangi *noise* dengan melakukan estimasi nilai (*state*) selanjutnya berdasarkan nilai-nilai sebelumnya. Filter Kalman memprediksi *state* dari nilai sebelumnya kemudian memperbarui *state* berdasarkan prediksi serta nilai saat ini. Setelah tegangan keluaran diberi Filter Kalman, hasil keluarannya lebih halus dan bebas dari *noise* tinggi. Grafik tegangan keluaran dari motor DC kedua setelah diberi Filter Kalman ditunjukkan pada Gambar 7.

B. Pengendali PID

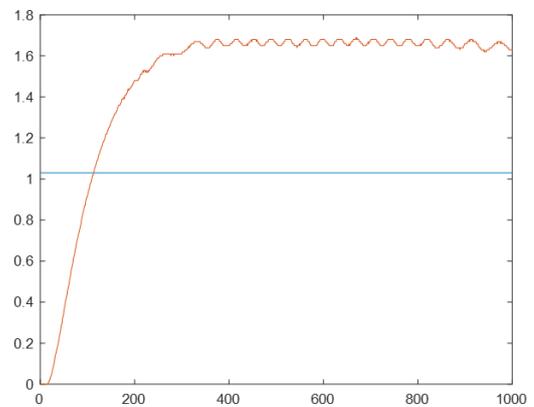
Sebelum diberikan kendali, tegangan keluaran yang diukur dari motor DC kedua memiliki *rise time* yang cukup lama dan memiliki *error steady state* yang sangat besar seperti pada Gambar 6 dan Gambar 7. Untuk mempercepat *rise time* dan juga mengurangi nilai *error steady state* tersebut,

dibutuhkan suatu kendali. Kendali yang diimplementasikan pada sistem ini adalah kendali *proportional* (P), kendali *proportional-integral* (PI), dan kendali PID.

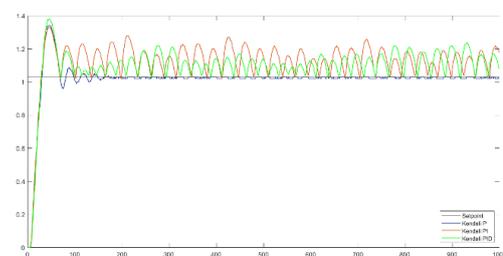
1) *Kendali P*: Nilai konstanta proporsional (K_p) yang digunakan pada kendali ini adalah 5. Kendali P memberikan respon keluaran yang mencapai set point dengan cepat. Nilai *rise time*-nya adalah 27 waktu *sampling*. Keluaran mengalami *overshoot* sebesar 30,09% dan *settling time*-nya sekitar 90 waktu *sampling*. Namun, masih terdapat *error steady state* sebesar -0,0042 V dan RMSE sebesar 0,0087.



Gambar 6. Grafik tegangan keluaran sebelum adanya filter Kalman



Gambar 7. Grafik tegangan keluaran setelah adanya filter Kalman



Gambar 8. Grafik respons kendali P, PI, dan PID dengan filter Kalman

Tabel 2. Perbandingan nilai *error* kendali P, PI, dan PID

Kendali	e_{ss} (V)	RMSE
P	-0,0042	0,0087
PI	+0,1121	0,1299
PID	+0,0874	0,1071

2) *Kendali PI*: Kendali PI ini menggunakan nilai konstanta proporsional (K_p) sebesar 5 dan konstanta integral (K_i) sebesar 1. Kendali PI pada motor DC tidak bisa menghasilkan keluaran yang stabil sesuai dengan set point namun memiliki nilai *rise time* yang cepat yakni 25 waktu sampling. Terdapat *overshoot* sebesar 29,13%, di mana keluarannya mengalami *ripple* secara terus-menerus dengan nilai rata-rata yang relatif besar. Nilai rata-rata *error steady state*-nya paling besar dari kendali lain yaitu +0,1121 V dengan nilai RMSE sebesar 0,1299.

3) *Kendali PI*: Nilai konstanta proporsional (K_p), integral (K_i), dan derivatif (K_d) yang digunakan berturut-turut adalah 5, 15, dan 10. Kendali PID pada motor DC juga tidak dapat memberikan respon keluaran yang stabil, tetapi *rise time*-nya tetap cepat sebesar 26 waktu sampling. Keluaran mengalami *overshoot* yang paling tinggi dari kendali sebelumnya yakni sebesar 33,98%. Keluarannya tetap mengalami *ripple* tinggi di atas nilai *set point*. *Ripple* ini mengakibatkan besarnya nilai rata rata eror yakni +0,0874 V dengan nilai RMSE sebesar 0,1071.

Gambar 8 menunjukkan grafik tegangan keluaran motor DC dengan kendali P, PI, dan PID yang telah disaring dengan filter Kalman. Dari grafik tersebut, terlihat bahwa kendali P memiliki respons yang paling baik karena dapat mencapai *set point* tanpa adanya *ripple* yang signifikan.

Tabel 2 menunjukkan perbandingan respons *transient* dari kendali P, PI, dan PID dengan Filter Kalman. Dari nilai *error steady state* dan RMSE masing-masing kendali, diperoleh bahwa kendali P merupakan metode kendali terbaik untuk sistem ini yang mampu menghasilkan respons yang mencapai nilai *set point*. Berbeda dengan kendali P, kendali PI dan PID tidak berhasil membuat respons sistem sesuai dengan nilai *set point*. Hal ini dapat dilihat dari RMSE kendali PI dan PID yang nilainya sekitar 0,1 lebih besar dari RMSE kendali P.

Kendali P merupakan satu-satunya kendali yang mampu menghasilkan respons yang mencapai nilai *set point* tanpa adanya osilasi terus-menerus. Akibat ini juga, kendali P memiliki nilai *error steady state* yang kecil jika dibandingkan dengan kendali PI maupun PID, yakni 3% lebih kecil. Dengan menggunakan kendali proporsional, respons sistem dapat dengan stabil mencapai *set point* yang artinya kecepatan motor kopel dapat dikontrol melalui masukan potensiometer.

V. SIMPULAN

Pada percobaan sistem kopel motor DC yang telah dilakukan menggunakan beberapa metode kendali meliputi LQR, LQT, PID serta menggunakan filter Kalman. Metode-

metode yang digunakan ini bertujuan untuk mendapatkan hasil yang stabil dari sistem kopel motor DC yang digunakan. Berikut merupakan paparan hasil percobaan yang didapatkan:

1. Proses penggunaan filter Kalman berpengaruh dalam mengurangi *noise* pada pembacaan nilai tegangan motor DC. Filter Kalman memprediksi *state* dari nilai sebelumnya kemudian memperbarui *state* berdasarkan prediksi serta nilai *state* saat ini. Setelah tegangan keluaran diberi filter Kalman, hasil keluarannya lebih stabil dan *noise* yang diakibatkan oleh *noise* proses ataupun *noise* sensor berkurang.
2. Memasuki perancangan pengendali PID, nilai-nilai parameter yang digunakan dari pengendali PID ini kami dapatkan secara *trial and error*. Perubahan pada setiap nilai parameter akan memengaruhi parameter yang lainnya sehingga pada penelitian ini digunakan bantuan dari MATLAB untuk melakukan penyetelan nilai PID secara otomatis.
3. Sebelum menerapkan kendali LQR, terlebih dahulu dilakukan pengamatan terhadap respons dari *open loop* sistem dalam bentuk fungsi alih yang didapatkan dari hasil identifikasi sistem. Hal ini dilakukan untuk mengetahui respons sistem sebelum diberikan kendali.
4. Selanjutnya dilakukan pengendalian respons sistem menggunakan kendali LQR dan LQT. Didapatkan hasil dari kedua kendali tersebut yang tidak jauh berbeda di mana keluaran yang dihasilkan tetap memiliki *delay* dan memiliki batas atas maksimum 2,1 V.
5. Berdasarkan hasil yang telah didapat dan respons sistem setelah diberikan kendali, dapat diamati bahwa saat sistem menggunakan filter Kalman dan menggunakan kendali P, hasil respons sistem memiliki *error steady state* paling kecil, sebesar -0,0042 V, 3% lebih kecil dari *error steady state* kendali PI dan PID.

DAFTAR PUSTAKA

- [1] N. Mishra, S. Bajpai, and E. Narayan, "Speed Control of DC Motor using PID Controller FED H-Bridge," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 10, pp. 274–282, 2020, doi: 10.35940/ijitee.f3967.0891020.
- [2] A. Khanna and T. Gaur, "Model Predictive Control Of Dc Motor Model In Matlab," *Int. J. Sci. Eng. Res.*, vol. 8, no. 4, pp. 82–85, 2017, [Online]. Available: <http://www.ijser.org>.
- [3] A. K. Rajagiri, S. R. Mn, S. S. Nawaz, and T. Suresh Kumar, "Speed control of DC motor using fuzzy logic controller by PCI 6221 with MATLAB," *E3S Web Conf.*, vol. 87, no. 201 9, pp. 1–6, 2019, doi: 10.1051/e3sconf/20198701004.
- [4] A. Ma'Arif, Iswanto, N. M. Raharja, P. A. Rosyady, A. R. C. Baswara, and A. A. Nuryono, "Control of DC Motor Using Proportional Integral Derivative (PID): Arduino Hardware Implementation," *Proceeding - 2020 2nd Int. Conf. Ind. Electr. Electron. ICIEE 2020*, pp. 74–78, 2020, doi: 10.1109/ICIEE49813.2020.9277258.
- [5] V. Yadav and V. K. Tayal, "Tuner," pp. 442–445, 2018.
- [6] A. N. A. Syarifuddin, D. A. Merdekawati, and E. Apriliani, "Perbandingan Metode Kalman Filter, Extended Kalman Filter, dan Ensambl Kalman Filter pada Model penyebaran virus HIV/AIDS," *Limits J. Math. Its Appl.*, vol. 15, no. 1, p. 17, 2018, doi: 10.12962/limits.v15i1.3344.
- [7] P. S. Madhukar and L. B. Prasad, "State Estimation using Extended Kalman Filter and Unscented Kalman Filter," *Proc. - 2020 Int.*

- [8] Y. Xu, K. Xu, J. Wan, Z. Xiong, and Y. Li, "Research on Particle Filter Tracking Method Based on Kalman Filter," *Proc. 2018 2nd IEEE Adv. Inf. Manag. Commun. Electron. Autom. Control Conf. IMCEC 2018*, no. Imcec, pp. 1564–1568, 2018, doi: 10.1109/IMCEC.2018.8469578.
- [9] F. Fahmizal, F. Fathuddin, and R. Susanto, "Identifikasi Sistem Motor DC dan Kendali Linear Quadratic Regulator Berbasis Arduino-Simulink Matlab," *Maj. Ilm. Teknol. Elektro*, vol. 17, no. 2, p. 399, 2018, doi: 10.24843/mite.2018.v17i02.p20.
- [10] E. Apriaskar, F. Fahmizal, N. A. Salim, and D. Prastiyanto, "Performance Evaluation of Balancing Bicopter using P, PI, and PID Controller," *J. Tek. Elektro*, vol. 11, no. 2, pp. 44–49, 2019, doi: 10.15294/jte.v11i2.23032.
- [11] S. Balamurugan and A. Umarani, "Study of Discrete PID Controller for DC Motor Speed Control Using MATLAB," *2020 Int. Conf. Comput. Inf. Technol. ICCIT 2020*, no. 1, pp. 358–363, 2020, doi: 10.1109/ICCIT-144147971.2020.9213780.
- [12] C. Xiangxian, Z. Yue, and H. Hai, "Train Speed Control Algorithm Based on PID Controller and Single-Neuron PID Controller," in 2010 Second WRI Global Congress on Intelligent Systems, 2010, pp. 107–110. doi: 10.1109/GCIS.2010.41.
- [13] C. Zhao and Z. Hua, "Design of Motor Speed Control System Based on STM32 Microcontroller," in 2022 International Conference on Computation, Big-Data and Engineering (ICCBE), 2022, pp. 274–276. doi: 10.1109/ICCBE56101.2022.9888225.
- [14] M. E. A. Mohamed, G. Yanling, B. A. rahman Osman, and A. A. Mohamed, "Design of Speed Control System of DC motor Based on PID tuning with fuzzy define weighting point," in 2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), 2019, pp. 1–6. doi: 10.1109/ICCCEEE46830.2019.9071374.
- [15] S. Sahoo, B. Subudhi, and G. Panda, "Optimal speed control of DC motor using linear quadratic regulator and model predictive control," in 2015 International Conference on Energy, Power and Environment: Towards Sustainable Growth (ICEPE), 2015, pp. 1–5. doi: 10.1109/EPETSG.2015.7510130.