

Robot Keseimbangan Dengan Kendali Proporsional-Integral-Derivatif (PID) dan Kalman Filter

Alfian Ma'arif¹, Riky Dwi Puriyanto², dan Fadlur T. Hasan³

Program Studi Teknik Elektro, Universitas Ahmad Dahlan^{1,2,3}

alfianmaarif@ee.uad.ac.id¹

Informasi Artikel

Riwayat Artikel:

Dikirim 06 Oktober 2019

Direvisi 21 Oktober 2019

Diterima 25 Februari 2020

Kata Kunci:

Kendali PID

Kalman Filter

Robot Keseimbangan

Sensor MMA731

ATmega32

Abstrak

Robot Keseimbangan memiliki dinamika yang cepat, tidak stabil, dan *non-linear* sehingga memerlukan pengendali yang sesuai. Robot keseimbangan menggunakan sensor *accelerometer* untuk mengukur perubahan sudut saat bergerak. Sifat sensor tersebut adalah sangat sensitif dan ber-*noise* sehingga memerlukan metode untuk mengurangi noise tersebut. Pada penelitian ini digunakan pengendali Proporsional Integral Derivatif (PID) untuk mengatasi dinamika tersebut. Kelebihan Pengendali PID adalah memiliki respon yang cepat dan mudah untuk diterapkan. Sementara untuk mengurangi noise pada sensor accelerometer digunakan metode kalman filter. Hasil pengujian menunjukkan bahwa metode kalman filter mampu untuk mengurangi noise pada sensor *accelerometer*. Nilai parameter kalman filter sangat mempengaruhi hasil filter sehingga memerlukan penentuan nilai yang tepat. Nilai matriks variasi proses harus lebih besar daripada nilai matriks variasi pengukuran. Nilai parameter kalman filter yang terbaik adalah matriks variasi proses $R = 10$ dan matriks variasi pengukuran $Q = 0,01$. Pengendali PID dapat menstabilkan robot pada posisi tegak. Nilai parameter terbaik pengendali PID adalah $K_p = 20$, $K_i = 1$, dan $K_d = 20$.

© This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

Koresponden:

Alfian Maarif,

Program Studi Teknik Elektro,

Universitas Ahmad Dahlan,

Kampus 4 Universitas Ahmad Dahlan, Jln. Ring Road Selatan, D.I.Yogyakarta, Indonesia.

Email : alfianmaarif@ee.uad.ac.id

1. PENDAHULUAN

Robot keseimbangan adalah salah satu jenis robot yang mampu berdiri tegak dengan menggunakan dua buah roda di kedua sisinya. Robot keseimbangan memiliki karakteristik dengan dinamika yang cepat, tidak stabil, dan *nonlinear* [1]. Hal tersebut disebabkan karena untuk mempertahankan robot agar dapat berdiri tegak diperlukan pemberian sinyal kendali yang tepat. Selain itu perancangan sistem mekanik robot dan berat robot juga perlu diperhatikan. Penentuan pusat gravitasi yang tepat

akan sangat mempengaruhi keseimbangan robot.

Penelitian tentang robot keseimbangan telah banyak dilakukan oleh peneliti sebelumnya dengan menggunakan berbagai macam pengendali seperti LQR (Linear Quadratic Regulator) [2], PD (Proporsional Derivatif) [3], PID (Proporsional Integral Derivatif) [4], *Neural Network Control* [5], *Backstepping* [1]. Namun, penelitian tersebut masih sebatas pada simulasi. Oleh karena itu agar kendali yang dirancang dapat lebih terbukti, diperlukan implementasi pada perangkat keras.

Robot keseimbangan memiliki prinsip yang sama seperti pada sistem pendulum terbalik [6], robot bola [7], pendulum terbalik di kereta, dan roket [8]. Selain itu juga muncul alat transportasi yang menggunakan prinsip robot keseimbangan yaitu *segways* yang menggunakan energi listrik yang bebas polusi udara.

Untuk dapat mengukur perubahan sudut pada robot keseimbangan diperlukan sensor *accelerometer*. Sensor tersebut memiliki kelemahan yaitu sangat sensitif terhadap perubahan dan memiliki *noise* yang berasal dari getaran. Oleh karena itu, diperlukan metode untuk menangani masalah pada sensor tersebut. Pada penelitian ini akan menggunakan metode *Kalman Filter* untuk mengatasi masalah tersebut.

Berdasarkan pada latar belakang yang telah diuraikan, penelitian tentang robot keseimbangan sangat menarik untuk dilakukan. Karakteristik robot keseimbangan memiliki dinamika yang cepat, tidak stabil, *nonlinear*, memiliki sensor yang bernoise dan memiliki banyak penerapan, menjadi tantangan bagi peneliti untuk mendesain sistem mekanik dan kendali robot keseimbangan.

Pada penelitian ini, pengendali yang digunakan adalah pengendali PID (Proporsional-Integral-Derivatif). Pengendali PID telah banyak diterapkan pada berbagai jenis robot yaitu *line follower*, *line maze solving robot* [9]. Hal tersebut disebabkan karena pengendali PID mudah untuk dimengerti dan mudah untuk diterapkan baik secara simulasi ataupun implementasi. Oleh karena itu, pengendali PID menjadi pengendali yang paling banyak digunakan di industri dan robotika.

Selain pengendali PID, terdapat pengendali lain seperti pengendali *State Feedback* [10] yang bisa digunakan. Pengendali *state feedback* adalah pengendali yang termasuk dalam sistem kendali modern [11]. Pengendali *state feedback* memiliki kekurangan yaitu semua variabel yang digunakan untuk mengendalikan sistem harus diketahui atau dapat diukur [12]. Sementara, pengendali PID tidak memerlukan semua variabel yang dikendalikan harus dapat diukur, sehingga pengendali PID lebih baik daripada *State Feedback*.

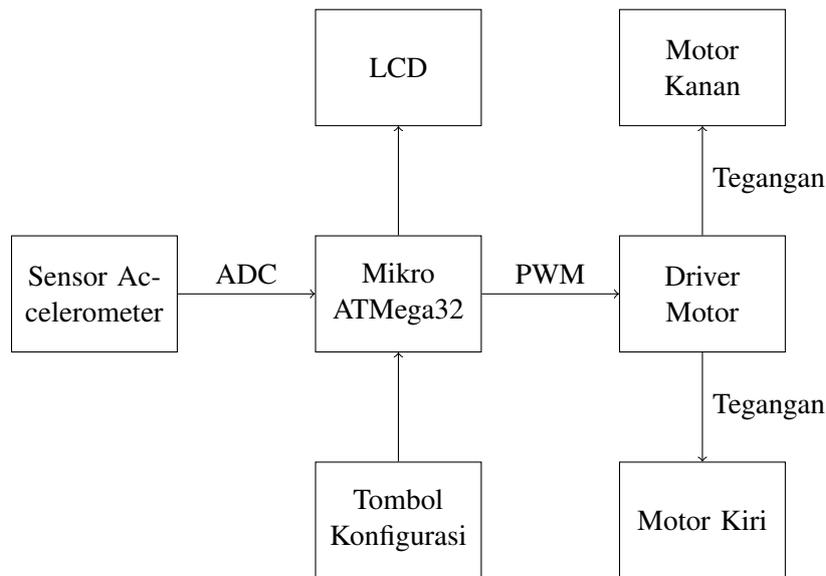
Pengendali yang lain adalah pengendali nonlinear seperti *Sliding Mode Control* (SMC) [13]. Namun untuk menggunakan pengendali SMC, diperlukan model sistem yang sesuai, tidak semua model sistem dapat dikendalikan pengendali SMC. Oleh karena itu, jika model sistem tidak sesuai maka pengendali SMC tidak dapat mengendalikan sistem. Pengendali SMC juga memiliki kekurangan pada sinyal kendalinya yang memiliki fenomena *chattering* sehingga tidak cocok diterapkan pada balancing robot yang memerlukan pengendali yang stabil dan cepat dalam menanggapi respons.

Bagian pertama penelitian ini adalah Pendahuluan. Bagian kedua adalah metode yang diusulkan. Bagian ketiga adalah metode penelitian. Bagian Keempat adalah hasil dan pembahasan. Bagian terakhir adalah kesimpulan.

2. METODE YANG DIUSULKAN

Terdapat beberapa desain dan metode yang digunakan dalam penelitian. Diagram blok sistem dan sistem kendali secara lengkap ditunjukkan pada Gambar 1 dan Gambar 2. Keterangan Simbol pada Gambar 1 adalah sebagai berikut. Bagian-bagian robot keseimbangan adalah Sensor Accelerometer, Mikrokontroler ATmega32, *Liquid Crystal Display* (LCD), Tombol konfigurasi, Driver motor bertipe *H-Bridge* dan Motor DC (*Direct Current*).

Data sensor accelerometer berupa nilai tegangan yang merupakan nilai sudut dan menjadi masukan untuk Mikrokontroler ATmega32. Data sensor Accelerometer dibaca dengan menggunakan fitur ADC (*Analog to Digital Converter*) 10 bit dengan tegangan referensi yang berasal dari PIN VREF. Lalu, keluaran Mikrokontroler ATmega32 adalah sinyal *Pulse Width Modulation*(PWM) yang dikirim ke Driver Motor *H-Bridge* untuk menggerakkan Motor DC kanan dan Motor DC kiri.

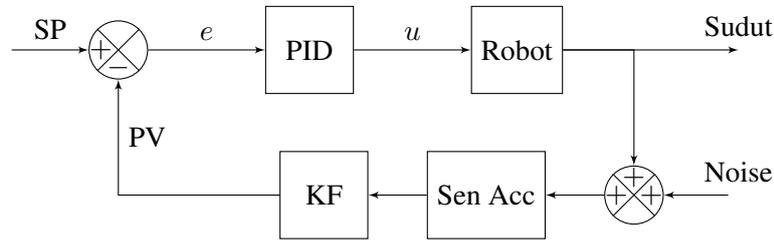


Gambar 1. Diagram Blok Sistem

Robot keseimbangan menggunakan LCD berukuran 2×16 , dua baris dan enam belas kolom, yang berfungsi untuk menampilkan data sensor, menampilkan data parameter robot (parameter kendali dan parameter kalman filter), dan melakukan kalibrasi (pengambilan data sudut referensi). Sementara, tombol konfigurasi berfungsi untuk melakukan pengaturan parameter kalman filter, pengaturan parameter kendali, dan tombol konfigurasi untuk berpindah antar menu.

Robot keseimbangan menggunakan *Lithium-Polymer* (Li-Po) sebagai sumber daya dengan tegangan kerja antara 11-12,6 volt. Tegangan kerja yang digunakan adalah 5 volt dan 12 Volt. Tegangan 5 volt berfungsi sebagai sumber tegangan mikrokontroler dan sensor accelerometer. Tegangan 12 volt berfungsi sebagai sumber tegangan motor DC. Untuk mendapatkan tegangan 5 volt dari tegangan 12 volt, digunakan komponen LM7805.

Keterangan simbol pada Gambar 2 adalah sebagai berikut. Blok SP adalah *set point* sistem yaitu nilai referensi sudut yang diinginkan. Untuk dapat berdiri tegak, posisi tegak robot dianggap dan disimpan sebagai sudut referensi. Blok PID adalah bagian kendali PID yang berfungsi untuk mengendalikan dan mengkoreksi posisi robot agar dapat tepat berdiri tegak. Blok robot adalah robot keseimbangan yang merupakan *plant* atau sistem yang dikendalikan. Blok Sen Acc adalah sensor accelerometer yang mengukur data perubahan sudut. Blok KF adalah kalman filter yang berfungsi untuk mem-filter data keluaran sensor accelerometer. Blok *noise* adalah data tambahan yang tidak diinginkan yang terukur oleh sensor dan tercampur dengan data sensor.



Gambar 2. Diagram Blok Kendali Sistem Kendali Kalang Tertutup

3. METODE PENELITIAN

3.1. Kalman Filter

Kalman filter berfungsi untuk mengestimasi suatu nilai berdasarkan pada data yang diketahui [14]. Kalman filter memiliki banyak penerapan atau aplikasi, salah satunya adalah untuk memfilter *noise*. Kalman filter berbeda dengan filter pada umumnya seperti *high pass filter* (HPF), *low pas filter* (LPF) ataupun *band pass filter* (BPF) yang hanya memfilter pada frekuensi tertentu. Semua frekuensi akan difilter oleh kalman filter sehingga kalman filter lebih baik dari pada LPF, HPF dan BPF dalam menghilangkan *noise*.

Fungsi utama kalman filter bukanlah menyaring atau memfilter sinyal yang masuk, akan tetapi mengestimasi sesuai dengan masukan yang diberikan. Oleh karena itu hasil sinyal setelah melalui proses kalman filter seperti terlihat terfilter. Terdapat beberapa macam pengembangan dari kalman filter seperti Extended Kalman Filter (EKF) [15], Unscented Kalman Filter (UKF) [16], Ensemble Kalman Filter (EnKF) [17]. Kelebihan lain Kalman filter adalah telah banyak diterapkan pada suatu sistem seperti pada Quadrotor [18] [19], Kesehatan [20], AUV [21], Magnetic Levitation Ball [22] dan Robot [23].

Pada penelitian ini, kalman filter berfungsi untuk memfilter data sudut dari sensor *accelerometer*. Data tersebut kemudian akan diolah oleh suatu Mikrokontroler. Persamaan kalman filter terdiri atas dua bagian yaitu bagian prediksi dan bagian *update* sebagai berikut.

Prediksi:

$$\hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{B}_t \mathbf{u}_t \quad (1)$$

$$\hat{\mathbf{P}}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^T + \mathbf{Q}_t \quad (2)$$

Update:

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{y}_t - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}) \quad (3)$$

$$\hat{\mathbf{K}}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^T (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^T + \mathbf{R}_t)^{-1} \quad (4)$$

$$\hat{\mathbf{P}}_{t|t} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \quad (5)$$

dengan $\hat{\mathbf{x}}$ adalah estimasi state, \mathbf{F} adalah matriks transisi, \mathbf{u} adalah variabel kendali, \mathbf{B} adalah matriks kendali, \mathbf{P} matriks variasi state, \mathbf{Q} adalah matriks variasi proses, \mathbf{y} adalah variabel pengukuran, \mathbf{H} adalah matriks perhitungan, \mathbf{K} adalah penguatan kalman, \mathbf{R} adalah matriks variasi pengukuran.

Persamaan kalman filter bagian prediksi dan *update* tersebut, dalam kasus untuk memfilter data sensor perlu dilakukan penyesuaian dan modifikasi. Hal tersebut dilakukan karena tidak semua variabel diperlukan dalam suatu kasus tertentu. Akan tetapi dimungkinkan untuk memakai seluruh variabel dalam kasus tertentu. Hasil modifikasi persamaan kalman filter adalah sebagai berikut.

Prediksi:

$$x_{t|t-1} = x_{t-1|t-1} \quad (6)$$

$$p_{t|t-1} = p_{t-1|t-1} + Q \quad (7)$$

Update:

$$x_{t|t} = x_{t|t-1} + K_t(y_t - x_{t|t-1}) \quad (8)$$

$$K_t = p_{t|t-1}(p_{t|t-1} + R)^{-1} \quad (9)$$

$$p_{t|t} = (1 - K_t)p_{t|t-1} \quad (10)$$

Setelah diperoleh persamaan kalman filter yang diperlukan, persamaan tersebut dikonversi menjadi program pada *Listing Program 1*.

```

Dataadc = Getadc (7)
Yt = Dataadc

Xt_update = Xt_prev
Pt_update = Pt_prev + Q

Kt = Pt_update + R
Kt = Pt_update / Kt

Xt = Yt - Xt_update
Xt = Kt * Xt
Xt = Xt_update + Xt

Pt = 1 - Kt
Pt = Pt * Pt_update

Xt_prev = Xt
Pt_prev = Pt

Accelx = Xt
Error = Vreff - Accelx

```

Listing 1. Program Kalman Filter

Keterangan listing program tersebut adalah sebagai berikut, *dataadc* adalah variabel untuk menyimpan data keluaran sensor, *R* dan *Q* adalah konstanta kalman filter yang nilainya menentukan tingkat penghalusan hasil filter. Nilai *R* dan *Q* tidak boleh dipilih secara sembarangan karena kesalahan dalam penentuan kedua parameter tersebut membuat hasil filter menjadi tidak baik.

3.2. Kendali Proporsional Integral Derivatif (PID)

Kendali PID adalah kendali yang terdiri atas Kendali Proporsional, Kendali Integral, dan Kendali Derivatif [8]. Kendali PID berfungsi untuk mengendalikan robot mencapai nilai referensi yang diinginkan yaitu membuat robot keseimbangan dapat berdiri tegak seimbang. Kelebihan pengendali PID adalah mudah untuk dimengerti, mudah untuk diterapkan baik secara *hardware* maupun simulasi [24]. Persamaan kendali PID dalam domain waktu (t) adalah

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (11)$$

dengan

$$K_i = \frac{K_p}{T_i} \quad K_d = K_p T_d \quad (12)$$

atau dalam domain (s) adalah

$$U(s) = K_p E(s) + K_i \frac{1}{s} E(s) + K_d s E(s) \quad (13)$$

Kendali PID dalam bentuk program adalah

```

Prop = P * Error
Sum_error = Sum_error + Error
Integral = I * Sum_error
Diff = Error - Last_error
Deriv = D * Diff
Last_error = Error

```

Listing 2. Program Kendali PID

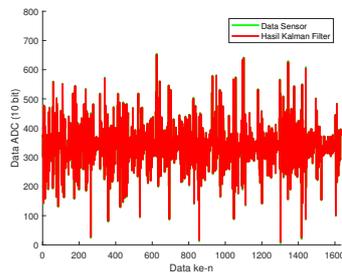
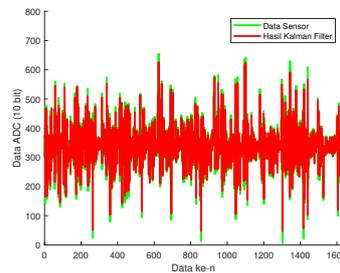
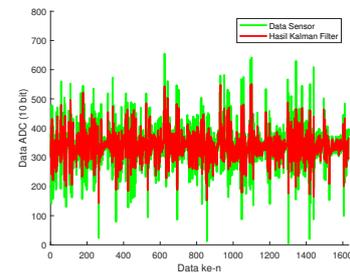
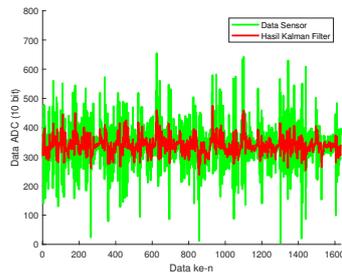
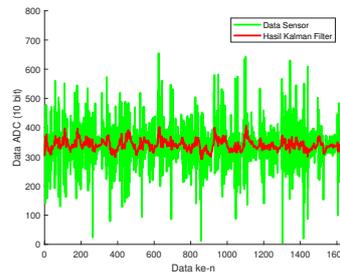
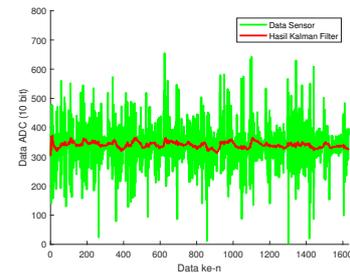
Keterangan program pada *listing 2* adalah sebagai berikut. Variabel P adalah konstanta proporsional atau penguatan proporsional, Variabel I adalah konstanta integral atau penguatan integral, dan variabel D adalah konstanta derivative atau penguatan derivative. Nilai *Error* merupakan selisih antara sudut referensi (*set point*) dan sudut saat ini (*present value*). Nilai *Error* sebelumnya disimpan dalam variabel *Last_Error*. Variabel *Sum_Error* adalah hasil penjumlahan dari seluruh nilai error dari yang pertama kali sampai waktu tertentu. Variabel *Diff* adalah selisih antara error yang sekarang dikurangi dengan *error* yang sebelumnya. Sinyal kendali PID diperoleh dengan menjumlahkan variabel *Prop*, *Deriv* dan *Integral* sebagai sinyal kendali kecepatan motor DC dalam bentuk sinyal PWM (*Pulse Width Modulation*).

4. HASIL DAN ANALISIS

Terdapat dua pengujian pada bagian ini yaitu pengujian nilai konstanta kalman filter dan pengujian sistem kendali robot. Pengujian konstanta kalman filter berfungsi untuk mengamati pengaruh penambahan dan pengurangan nilai konstanta kalman filter. Pengujian sistem kendali robot berfungsi untuk menguji hasil perancangan sistem kendali robot dan perancangan kalman filter atau kinerja robot.

4.1. Pengujian Konstanta Kalman Filter

Hasil pengujian nilai konstanta kalman filter, R dan Q , ditunjukkan pada Gambar 3 - 8. Pada pengujian ini, data sensor accelerometer diambil dengan menggunakan arduino kemudian diolah dengan menggunakan software Matlab. Pada pengujian tersebut, dilakukan variasi nilai R dan Q untuk mendapatkan hasil filter yang paling baik. Berdasarkan pada pengujian Gambar 3 dan Gambar 4 dengan nilai konstanta Q lebih besar daripada R menunjukkan bahwa hasil filter tidak berkurang secara signifikan. Oleh karena itu, nilai konstanta tersebut kurang baik untuk memfilter sinyal *noise* pada sensor.

Gambar 3. $R=1$ $Q=100$ Gambar 4. $R=1$ $Q=10$ Gambar 5. $R=1$ $Q=1$ Gambar 6. $R=1$ $Q=0,1$ Gambar 7. $R=10$ $Q=0,1$ Gambar 8. $R=10$ $Q=0,01$

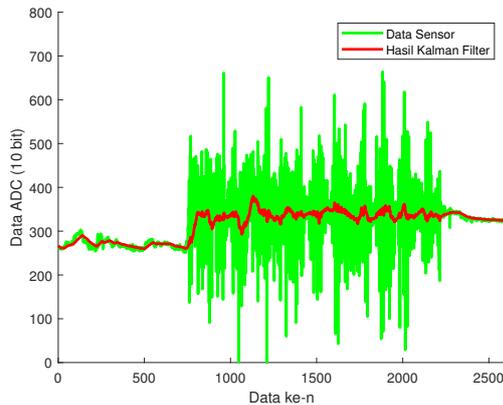
Berdasarkan pada pengujian Gambar 5, dengan nilai $R = 1$ dan $Q = 1$ menunjukkan bahwa sinyal mulai terfilter dengan baik. Namun nilai tersebut masih kurang baik untuk digunakan sebagai filter pada robot. Hasil filter yang baik ditunjukkan pada Gambar 6, Gambar 7, Gambar 8. Berdasarkan pada hasil pengujian tersebut dapat diketahui bahwa data sensor mulai terfilter dengan baik, semakin besar jarak antara nilai R dan Q , hasil filter semakin halus dan berkurang nilai noise-nya.

Berdasarkan hasil pengujian ini dapat diketahui bahwa parameter kalman filter yang terbaik adalah nilai konstanta R harus lebih besar daripada nilai konstanta Q . Jika nilai Q lebih besar dari nilai R , maka data sensor tidak terfilter dengan baik. Namun perlu diperhatikan Jika jarak antara nilai R dan Q semakin besar maka hasil filter akan semakin halus bahkan tidak terfilter. Nilai konstanta terbaik yang diperoleh adalah $R = 10$ dan $Q = 0,01$.

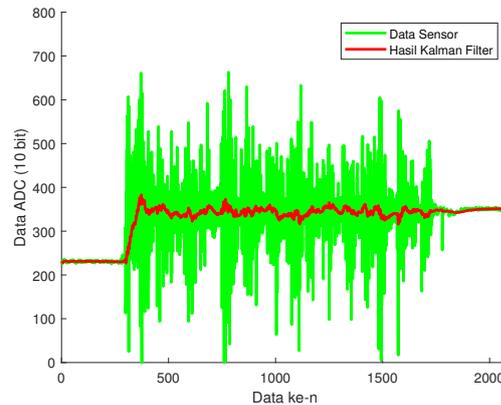
4.2. Pengujian Sistem Kendali Robot

Hasil pengujian sistem kendali robot ditunjukkan pada Gambar 9 - Gambar 11. Grafik berwarna hijau adalah data adc dari sensor dan grafik berwarna merah adalah data hasil filter. Pengujian dilakukan dengan cara meletakkan robot pada posisi sudut tertentu, kemudian menyalakan robot. Se-

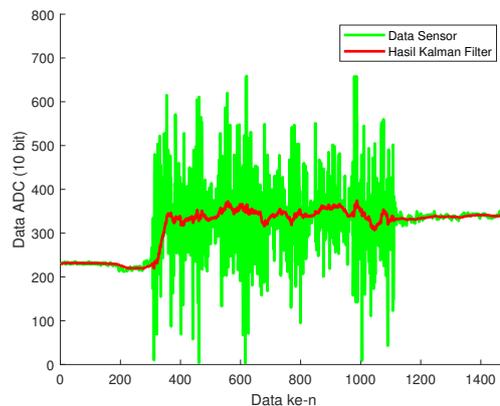
belum pengujian, dilakukan kalibrasi untuk mengambil data adc pada posisi robot tegak. Posisi tersebut menjadi referensi bagi pengendali robot untuk dapat membuat robot berdiri seimbang dan diset sebagai sudut 0 derajat. Nilai referensi yang diperoleh dari proses kalibrasi adalah 340.



Gambar 9. Pengujian 1



Gambar 10. Pengujian 2



Gambar 11. Pengujian 3

Berdasarkan pada hasil pengujian Gambar 9 - Gambar 11, dapat diketahui bahwa robot dapat berdiri tegak. Hal tersebut dapat dilihat pada grafik berwarna merah yang berada di sekitar nilai referensi (nilai 340). Nilai variabel k_p yang terbaik bernilai 20, Nilai variabel k_i yang terbaik bernilai 1, dan Nilai variabel k_d yang terbaik bernilai 20.

Dalam proses pencarian nilai parameter pengendali PID yang terbaik, dilakukan dengan cara sebagai berikut. Nilai pertama yang dituning adalah variabel k_p , variabel k_i dan k_d diberi nilai 0. Tuning dilakukan dengan cara menaikkan nilai k_p mulai dari nilai 2 sampai nilai tertentu. Berdasarkan pengujian, penambahan nilai k_p membuat robot bergerak semakin sensitif terhadap perubahan sudut. Nilai k_p berpengaruh terhadap kecepatan respon robot terhadap perubahan sudut, jika nilai k_p kecil maka robot memiliki respon yang lambat. Sementara, jika nilai k_p besar, maka robot memiliki respon yang cepat namun membuat robot berosilasi.

Nilai kedua yang dituning adalah variabel k_d yang berfungsi untuk mengurangi osilasi pada robot. Dengan menggunakan kombinasi nilai k_p dan penambahan nilai k_d membuat robot memiliki

kinerja respon yang cepat namun tidak terlalu berosilasi. Dibandingkan dengan kinerja robot ketika hanya menggunakan nilai k_p , kinerja robot cenderung lebih lambat dalam menanggapi perubahan sudut. Oleh karena itu, untuk mengembalikan kinerja robot dilakukan dengan penambahan nilai k_i .

Penambahan nilai k_i harus dilakukan dengan dimulai dari nilai sangat kecil (0.05). Hal tersebut disebabkan karena penambahan nilai k_i membuat robot memiliki respon yang sangat cepat. Respon tersebut disebabkan karena pengendali integral menjumlahkan nilai semua error dari detik pertama sampai terakhir sehingga sinyal kendali nya menjadi sangat besar atau sangat kecil. Oleh karena itu, jika penambahan nilai k_i terlalu besar, maka robot dapat bergerak dengan liar bahkan bisa menabrak lantai (sinyal kendalinya sangat besar). Nilai k_i terbaik yang diperoleh adalah 1.

5. KESIMPULAN

Penelitian ini mengusulkan tentang kendali PID (Proporsional-Integral-Derivatif) pada robot keseimbangan dan kalman filter pada sensor accelerometer. Berdasarkan pengujian, pengendali PID dapat membuat robot keseimbangan mampu untuk dapat berdiri tegak pada posisi seimbang. Nilai penguatan kendali proporsional, integral, dan derivatif yang terbaik adalah $k_p = 20$, $k_i = 1$, & $k_d = 20$. Hasil lain adalah kalman filter mampu untuk mem-filter noise pada sensor accelerometer. Nilai konstanta kalman filter yang terbaik adalah matriks variasi proses $R = 10$ dan matriks variasi pengukuran $Q = 0,01$.

Dalam proses penentuan parameter pengendali PID, masih menggunakan metode *trial and error* atau coba-coba yang memerlukan banyak waktu dan tenaga untuk menentukan nilai variabel yang menghasilkan kinerja terbaik. Oleh karena itu diperlukan penelitian lanjutan untuk menentukan nilai parameter pengendali yang lebih baik daripada metode coba-coba. Penentuan nilai konstanta kalman filter juga masih menggunakan cara coba-coba sehingga perlu metode khusus agar penentuan nilai parameter dapat lebih efektif.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada editor yang telah membantu dalam proses penerbitan naskah paper. Ucapan terima kasih juga diberikan kepada mitra bestari yang telah memberikan saran dan masukan untuk meningkatkan kualitas artikel ini sehingga layak untuk dipublikasikan. Selain itu penulis juga mengucapkan terima kasih kepada penerbit yang tidak memungut biaya dalam proses pengiriman sampai penerbitan. Ke depan Semoga jurnal ITJRD dapat meningkatkan indeksasi Sinta ke Sinta 1/2 dan dapat menjadi jurnal internasional yang terindeks Scopus.

Daftar Pustaka

- [1] Nguyen Gia Minh Thao, Duong Hoai Nghia, and Nguyen Huu Phuc, "A pid backstepping controller for two-wheeled self-balancing robot," in *International Forum on Strategic Technology 2010*, Oct 2010, pp. 76–81, doi: 10.1109/IFOST.2010.5668001.
- [2] F. Sun, Z. Yu, and H. Yang, "A design for two-wheeled self-balancing robot based on kalman filter and lqr," in *2014 International Conference on Mechatronics and Control (ICMC)*, July 2014, pp. 612–616, doi: 10.1109/ICMC.2014.7231628.
- [3] C. Iwendi, M. A. Alqarni, J. H. Anajemba, A. S. Alfakeeh, Z. Zhang, and A. K. Bashir, "Robust navigational control of a two-wheeled self-balancing robot in a sensed environment," *IEEE Access*, vol. 7, pp. 82 337–82 348, 2019, doi: 10.1109/ACCESS.2019.2923916.
- [4] M. I. Ali and M. M. Hossen, "A two-wheeled self-balancing robot with dynamics model," in *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, Sep. 2017,

- pp. 271–275, doi: 10.1109/ICAEE.2017.8255365.
- [5] C. Sun, T. Lu, and K. Yuan, “Balance control of two-wheeled self-balancing robot based on linear quadratic regulator and neural network,” in *2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP)*, June 2013, pp. 862–867.
- [6] H. F. Murcia and A. E. González, “Performance comparison between pid and lqr control on a 2-wheel inverted pendulum robot,” in *2016 IEEE Colombian Conference on Robotics and Automation (CCRA)*, Sep. 2016, pp. 1–6, doi: 10.1109/CCRA.2016.7811420.
- [7] You Yunong, Hwi-Myung Ha, Yun-Ki Kim, and Jang-myung Lee, “Balancing and driving control of a ball robot using fuzzy control,” in *2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Oct 2015, pp. 492–494.
- [8] K. Ogata, *Modern Control Engineering*. Prentice Hall, 2010.
- [9] A. Maarif, S. Iskandar, and I. Iswanto, “New design of line maze solving robot with speed controller and short path finder algorithm,” *International Review of Automatic Control (IREACO)*, vol. 12, no. 3, pp. 154–162, 2019, doi: 10.15866/ireaco.v12i3.16501.
- [10] A. Ma’arif, A. I. Cahyadi, O. Wahyunggoro, and Herianto, “Servo state feedback based on coefficient diagram method in magnetic levitation system with feedback linearization,” in *2017 3rd International Conference on Science and Technology - Computer (ICST)*, July 2017, pp. 22–27, doi: <https://doi.org/10.1109/ICSTC.2017.8011846>.
- [11] A. Maarif, A. I. Cahyadi, S. Herdjunto, Y. Yamamoto *et al.*, “Tracking control of higher order reference signal using integrators and state feedback.” *IAENG International Journal of Computer Science*, vol. 46, no. 2, 2019.
- [12] A. Ma’arif, A. imam Cahyadi, and O. Wahyunggoro, “Cdm based servo state feedback controller with feedback linearization for magnetic levitation ball system,” *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 3, pp. 930–937, 2018, doi: <http://dx.doi.org/10.18517/ijaseit.8.3.1218>.
- [13] R. Uswarman, S. Istiqphara, and D. H. T. Nugroho, “Sliding mode control with gain scheduled for magnetic levitation system,” *Jurnal Ilmiah Teknik Elektro Komputer Informatika*, vol. 5, no. 1, pp. 36–43, 2019, doi: <http://dx.doi.org/10.26555/jiteki.v5i1.13223>.
- [14] M. Grewal and A. Andrews, *Kalman Filtering: Theory and Practice with MATLAB*, ser. Wiley - IEEE. Wiley, 2015.
- [15] S. Nurmaini and S. Pangidoan, “Localization of leader-follower robot using extended kalman filter,” *International Review of Automatic Control (IREACO)*, vol. 7, no. 2, pp. 95–108, 2018, doi: 10.18495/COMENGAPP.V7I2.253.
- [16] M. Han, K. Kim, D. Y. Kim, and J. Lee, “Implementation of unicycle segway using unscented kalman filter in lqr control,” in *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Oct 2013, pp. 695–698, doi: 10.1109/URAI.2013.6677427.
- [17] W. K. Barlian Henryranu Prasetio, “Uji performansi ensemble kalman filter untuk mengurangi noise pengukuran sensor pada robot,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 2, no. 2, pp. 96–101, 2015, doi: <http://dx.doi.org/10.25126/jtiik.201522139>.
- [18] M. Safitri and N. H. Wijaya, “Sistem penapisan derau pada sensor inersia wahana tanpa awak quadrotor,” *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, vol. 7, no. 2, pp. 753–760, 2016, doi: 10.24176/simet.v7i2.791.
- [19] S. H. R. H. Lasmadi Lasmadi, Adha Imam Cahyadi, “Inertial navigation for quadrotor using kalman filter with drift compensation,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 5, pp. 2596–2604, 2017, doi: 10.11591/ijece.v7i5.pp2596-2604.

- [20] E. A. Agus N.A. Syarifuddin, Dian A. Merdekawati, “Perbandingan metode kalman filter, extended kalman filter, dan ensemble kalman filter pada model penyebaran virus hiv/aids,” *Journal Mathematics and its Applications*, vol. 15, no. 1, pp. 17–29, Maret 2018, doi: <http://dx.doi.org/10.12962/limits.v15i1.3344>.
- [21] N. Ngatini and H. Nurhadi, “Estimasi lintasan auv 3 dimensi (3d) dengan ensemble kalman filter,” *Journal Mathematics and its Applications*, vol. 4, no. 1, pp. 12–21, 2019, <https://doi.org/10.35314/isi.v4i1.774>.
- [22] T. Herlambang, “Estimasi posisi magnetic levitation ball menggunakan metode akar kuadrat ensemble kalman filter (ak-enkf),” *R.E.M. (Rekayasa Energi Manufaktur) Jurnal*, vol. 2, no. 1, pp. 45–49, 2017, <https://doi.org/10.21070/r.e.m.v2i1.768>.
- [23] M. S. R. Hamzah Ahmad, Nur Aqilah Othman, “A solution to partial observability in extended kalman filter mobile robot navigation),” *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, vol. 16, no. 1, pp. 134–141, 2018, <http://dx.doi.org/10.12928/telkomnika.v16i1.9025>.
- [24] A. Ma’arif, H. Nabila, Iswanto, and O. Wahyunggoro, “Application of intelligent search algorithms in proportional-integral-derivative control of direct-current motor system,” *Journal of Physics: Conference Series*, vol. 1373, p. 012039, nov 2019, doi: 10.1088/1742-6596/1373/1/012039.

BIOGRAFI PENULIS



Alfian Ma’arif Penulis pertama memperoleh gelar sarjana teknik elektro dari universitas Islam Indonesia pada tahun 2014 dan memperoleh gelar master of engineering dari universitas gadjah mada pada tahun 2017. Saat ini, penulis adalah staf pengajar di program studi teknik elektro, universitas ahmad dahlan sejak 2018. Topik penelitiannya meliputi bidang robotika dan sistem kendali. Penulis juga merupakan anggota IAENG, Pembimbing tim robot, editor di jurnal internasional Telkomnika, mitra bestari di jurnal nasional JITEKI, dan editor di jurnal nasional BISTE. Selain menguasai bidang instrumentasi dan elektronika, penulis juga menguasai bahasa pemrograman seperti Matlab dan Latex.



Riky Dwi Puriyanto Penulis kedua memperoleh gelar sarjana teknik elektro dan magister of engineering dari universitas Gadjah Mada pada tahun 2009 dan 2013. Saat ini, penulis adalah staf pengajar di program studi teknik elektro, universitas ahmad dahlan sejak 2015. Topik penelitiannya meliputi bidang otomasi industri. Penulis juga merupakan sekretaris program studi teknik elektro UAD, Pembimbing tim robot KRSBI, dan editor di jurnal nasional BISTE. Saat ini penulis kedua sedang melaksanakan pendidikan doktor dengan topik robotika dan sistem kendali.



Fadlur Rahman T. Hasan Penulis ketiga adalah mahasiswa program studi teknik elektro universitas Ahmad dahlan angkatan 2016. Topik penelitiannya meliputi robotika dan sistem kendali.