

# Sistem Kendali Kalang Tertutup dengan Metode PID pada Robot Wayang

Yanuardi Ramadhan<sup>1</sup>, Sisdarmanto Adinandra<sup>2</sup>

Jurusan Teknik Elektro, Universitas Islam Indonesia  
Jl Kaliurang KM 14.5 Yogyakarta, Indonesia

<sup>1</sup>14524133@students.uui.ac.id

<sup>2</sup>s.adinandra@uui.ac.id



**Abstrak**— Pada penelitian ini dirancang sebuah sistem kendali kalang tertutup untuk robot lengan dengan 4 DOF (*Degree of Freedom*). MATLAB akan digunakan sebagai antarmuka pengendalian robot. Data perintah dari MATLAB akan dikirimkan ke Arduino. Setelah itu Arduino akan memberikan sinyal untuk menggerakkan servo pada sudut yang diinginkan. Dengan menggunakan potensiometer sebagai sensor, besar sudut yang dihasilkan oleh servo dapat dibaca untuk menciptakan sistem kalang tertutup. Target yang ingin dicapai adalah mengurangi besar *steady-state error* (SSE) yang terjadi pada setiap servo.

Dibandingkan sistem kalang terbuka, sistem ini dapat mengurangi rata-rata SSE pada servo 1, 2, 3, dan 4 berturut-turut sebesar 5,67°, 2,46°, 3,71°, dan 20,39°. Sedangkan *settling time*-nya bertambah sebesar 1,66 detik, 2,24 detik, 1,77 detik dan 1,69 detik.

**Kata kunci** : *Articulated robot, Arduino, MATLAB, steady state error, settling time*

## I. PENDAHULUAN

Robotika telah menjadi topik penelitian lebih dari puluhan tahun. Pengaplikasiannya dimulai di sektor industri, guna membebaskan pekerja dalam pekerjaan berbahaya atau melelahkan [1]. Jika berbicara tentang robot industri, hal pertama yang terpikirkan adalah articulated robot (robot lengan). Robot jenis ini dapat meningkatkan produktivitas industri dengan meningkatkan kecepatan dan akurasinya. Selain itu kualitas produk yang dibuat akan meningkat. Karena alasan tersebut, robot lengan menjadi robot yang paling sering digunakan di pabrik-pabrik di seluruh dunia [2].

Pada laboratorium Fakultas Teknologi Industri Universitas Islam Indonesia terdapat berbagai robot lengan untuk keperluan pembelajaran mahasiswa. Contohnya, pada WayBot (robot wayang) yang mana masih memakai sistem kendali *open-loop* (kalang terbuka). Hal ini membuat *steady-state error* dari sistem menjadi besar. Sistem tersebut dapat ditingkatkan menjadi sistem *closed-loop* (kalang tertutup).

Untuk menutupi kekurangan dari sistem tersebut maka dibuatlah sistem kendali kalang tertutup. Sistem yang dibuat menggunakan pengendali PID untuk meningkatkan performa dari servo. Tujuan utama dari sistem ini untuk meminimalisir *steady state error*. Pada tugas akhir ini digunakan komputer yang di dalamnya terdapat perangkat lunak yang disebut MATLAB. MATLAB berguna sebagai antarmuka dari pengendalian robot. Robot wayang akan dikendalikan oleh

servo. Sedangkan untuk menggerakkan servo digunakan Arduino.

Untuk menutupi kekurangan dari sistem tersebut maka dibuatlah sistem kendali kalang tertutup. Sistem yang dibuat menggunakan pengendali PID untuk meningkatkan performa dari servo. Tujuan utama dari sistem ini untuk meminimalisir *steady state error*. Pada tugas akhir ini digunakan komputer yang di dalamnya terdapat perangkat lunak yang disebut MATLAB. MATLAB berguna sebagai antarmuka dari pengendalian robot. Robot wayang akan dikendalikan oleh servo. Sedangkan untuk menggerakkan servo digunakan Arduino.

## II. TINJAUAN PUSTAKA

### A. Studi Literatur

Dari penelitian oleh Jokic dkk, dibahas tentang implementasi sistem kendali *closed-loop* untuk robot *single axis* dengan motor DC. MATLAB digunakan sebagai *software* untuk membuat algoritma pengendalinya. Pada MATLAB terdapat toolbox yang dirancang secara khusus untuk keperluan pengembangan struktur kendali secara *real time*[3].

Selain itu terdapat penelitian oleh Marsian, dibahas tentang kendali robot wayang dengan 4-DOF. Pada penelitian tersebut dirancang suatu pengendali robot manipulator lengan berbentuk wayang dengan konfigurasi joint RRRR. LabVIEW digunakan sebagai antarmuka pengendalian robot dan penghitungan *forward* dan *inverse kinematic*. Data hasil penghitungan tersebut akan dikirimkan ke Arduino untuk mengendalikan servo. Pada penelitian ini digunakan sistem kendali kalang terbuka.

Dengan memperhatikan beberapa literatur diputuskan untuk dibuatnya kendali robot wayang yang telah dibahas di paragraf sebelumnya. Akan tetapi, pada penelitian ini akan dibuat robot dengan sistem kalang tertutup untuk meningkatkan kekurangan dari penelitian oleh saudara Marsian.

### B. Servo

Motor servo adalah salah satu jenis motor DC. Output dari perangkat ini adalah posisi pada porosnya. Poros ini dapat diposisikan ke posisi sudut tertentu dengan mengirimkan sinyal kepada servo. Selama terdapat sinyal pada jalur input, servo akan mempertahankan posisi sudut poros. Saat terjadi perubahan sinyal, posisi sudut poros akan berubah.

Motor servo memiliki beberapa sirkuit kontrol dan potensiometer yang terhubung ke poros. Potensiometer memungkinkan sirkuit kontrol untuk memantau sudut arus motor servo. Jika poros telah di sudut yang benar, motor akan berhenti[6].

### C. MATLAB

MATLAB (*Matrix Laboratory*) adalah sebuah perangkat lunak yang berguna untuk komputasi dan analisis numerik. *Software* ini juga merupakan bahasa pemrograman matematika yang menggunakan bentuk dan sifat matriks. Selain memiliki fungsi *built-in*, pengolahan sinyal, dan kalkulasi matematis lainnya, MATLAB juga memiliki *toolbox* untuk beberapa fungsi khusus. MATLAB adalah program yang bersifat *extensible*, hal ini berarti pengguna dapat membuat fungsi tambahan yang akan dimasukkan pada *library*.

### D. Arduino

Arduino adalah salah satu perangkat kendali mikrokontroler. Arduino berasal dari keluarga ATmega. Arduino dibuat dengan upaya membuat perangkat yang terjangkau, bersifat *open-source*, dan sederhana. Penggunaan Arduino sangat mudah, sehingga dapat dipelajari oleh orang awam sekalipun[7].

Bahasa pemrograman yang digunakan oleh arduino adalah *Processing*. Bahasa ini berbasis dari bahasa C dan C++. Karena itu bahasa ini dapat dijadikan jembatan ke bahasa pemrograman AVR C jika pengguna ingin mendapatkan pemahaman yang lebih mendalam dalam hal teknis. [8].

### E. Pengendali PID

PID merupakan salah satu algoritma kontrol yang paling umum digunakan dalam industri kontrol. Pengendali ini sangat populer dikarenakan kesederhanaan dalam pengoperasiannya, sehingga dapat digunakan dengan mudah. Walaupun sederhana, pengendali PID memiliki kinerja yang bagus dan cocok untuk kebanyakan kondisi operasi. Pengendali PID terdiri dari tiga koefisien, yaitu proporsional, integral dan derivatif.

Proporsional adalah komponen yang membandingkan besar antara set point dan variabel proses. Perbedaan kedua hal tersebut disebut *error*. Nilai dari proporsional menentukan respon output terhadap error yang terjadi. Secara umum, dengan menaikkan nilai proporsional maka kecepatan dari respon akan meningkat. Namun, hal tersebut akan menimbulkan osilasi yang lebih besar pula.

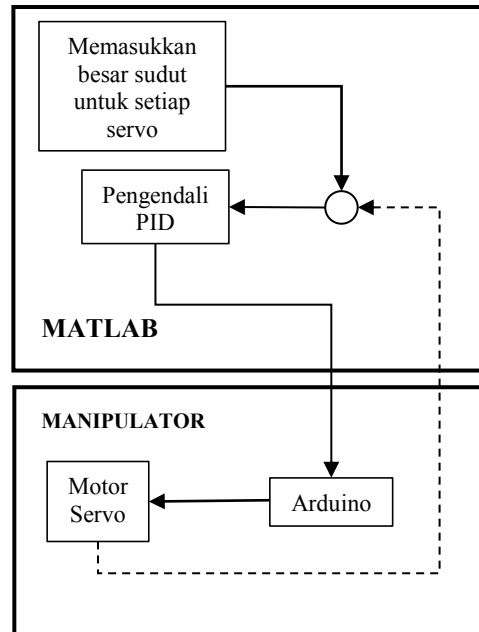
Pada komponen integral error pada setiap waktu akan terus diakumulasikan. Hal ini menyebabkan respon output akan terus meningkat seiringnya waktu berjalan. Integral juga bisa disebut sebagai penjumlahan *error*. Respon integral akan terus meningkat sampai nilai *error* mengarah ke nilai nol. Secara umum, integral berguna untuk mengurangi nilai *steady-state error* atau yang disebut dengan nilai error pada kondisi tetap.

Derivatif merupakan komponen yang berguna untuk memprediksi besar error di masa mendatang. Respon dari

derivatif bergantung pada perubahan dari error pada setiap waktu yang dikalikan dengan nilai konstan derivatif yang telah dipilih. Dengan menggunakan derivatif, respon output akan menjadi lebih terkontrol karena error dari output lebih teredam.

## III. METODE PENELITIAN

### A. Perancangan Sistem



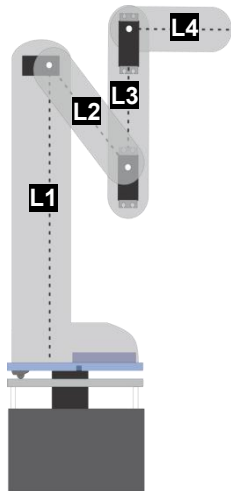
Gambar 1. Blok Diagram Sistem Kendali Robot Wayang

Proses sistem kendali berawal dengan menggunakan komputer. Pada kasus ini digunakan software MATLAB. Pada MATLAB dipilih sudut-sudut tiap joint. Data tersebut akan dikirimkan ke Arduino. Setelah data sampai ke perangkat arduino, data tersebut akan diproses dan dikirimkan ke motor servo untuk menggerakkan servo.

Setelah servo telah bergerak, maka posisi setiap lengan akan dibaca dengan menggunakan potensiometer. Data tersebut akan dikirimkan ke MATLAB melalui perantara Arduino. Data dari sudut lengan yang diinginkan dan sudut yang terbaca oleh potensiometer akan dibandingkan. Data perbandingan (*error*) tersebut akan dikirimkan ke pengendali PID untuk diproses. Pada pengendali PID terjadi proses untuk mengoreksi posisi dari robot berdasarkan nilai error yang dihasilkan. Data tersebut akan dikirimkan kembali kepada motor servo melalui Arduino. Proses ini akan diulang sampai posisi lengan robot dianggap benar.

### B. Sistem Mekanik

Lengan robot ini merupakan robot 4R dengan menggunakan mikrokontroler Arduino Uno. Pada setiap joint terdapat 3 jenis servo, yaitu servo Tower Pro MG 995 sebagai penggerak base, servo Tower Pro MG946R sebagai penggerak lengan 1 dan 2, dan servo Tower Pro MG90 sebagai penggerak lengan 3. Panjang lengan dari L1, L2, L3, dan L4 berturut-turut adalah 30 cm, 12 cm, 12 cm, dan 10cm.



Gambar 2. Desain Kerangka Robot

### C. Sistem Elektronik

Berikut adalah port yang digunakan pada mikrokontroler pada sistem skematik robot :

1. *Pin* 9, 8, 7, dan 6 digunakan sebagai pemberi sinyal kepada servo.
2. *Pin* analog A1, A2, A3, dan A4 digunakan untuk membaca nilai posisi dari servo. Sensor yang digunakan untuk membaca posisi dari servo adalah potensiometer yang telah tertanam pada servo.
3. Port serial sebagai port komunikasi antara komputer dan Arduino Uno.

Robot ini menggunakan *power supply* komputer sebagai suplai daya tambahan. Jika hal ini tidak digunakan maka servo akan kekurangan arus untuk bekerja.

### D. Sistem Kalibrasi

Kalibrasi diperlukan untuk mendapatkan nilai sudut dari sensor yang sesuai dengan nilai aslinya. Pada setiap servo terdapat sebuah potensiometer yang dapat berguna untuk mengetahui sudut dari servo tersebut. Dengan menyambungkan potensiometer ke Arduino menggunakan kabel maka sudut dari tiap servo dapat diketahui. Nilai yang terbaca pada potensiometer adalah berupa tegangan listrik. Nilai tegangan listrik tersebut akan diteruskan oleh Arduino ke PC di mana akan diproses oleh MATLAB. Pada MATLAB nilai dari tegangan listrik akan diubah oleh ADC dengan resolusi 10 bit. Karena nilai yang dibutuhkan adalah nilai dalam bentuk derajat maka diperlukan konversi untuk mengubah nilai tersebut menjadi nilai dengan besaran derajat. Kalibrasi perlu dilakukann untuk tiap sensor karena terdapat perbedaan karakteristik pada tiap servo walaupun untuk tipe yang sama. Cara kerja dari kalibrasi adalah dengan memberikan input sudut minimal dan sudut maksimal dari tiap servo. Dengan menggunakan persamaan regresi linier maka akan didapatkan persamaan untuk tiap servo yang dapat digunakan untuk

$$y = mx + c, \quad (1)$$

proses konversi dari nilai biner ke besaran derajat.

Pada hal ini  $x$  adalah nilai sudut yang diukur pada servo, sedangkan  $y$  adalah nilai dari biner yang dihasilkan oleh pengukuran pada sensor.

## IV. HASIL DAN ANALISIS

### A. Hasil Kalibrasi

Tabel 1. Nilai Maksimum dan Minimum dari Tiap Servo

Servo	Persamaan yang didapat
1	$y = \frac{306}{170}x + 118$
2	$y = \frac{306}{160}x + 105$
3	$y = \frac{302}{160}x + 113$
4	$y = \frac{315}{150}x + 111$

Kalibrasi dilakukan dengan mengambil nilai minimum dan maksimum dari servo. Servo akan digerakkan ke sudut minimum dan maksimum, lalu diukur nilai sudut tersebut dengan menggunakan busur. Sensor pun akan membaca nilai dari sudut servo dengan bentuk nilai biner. Setelah nilai tersebut telah didapat, maka akan dihitung persamaan dari konversi servo dengan menggunakan persamaan garis lurus. Pada pengukuran pertama (nilai minimum), semua servo akan digerakkan hingga mencapai sudut  $0^\circ$ . Setelah itu nilai biner yang terbaca oleh sistem akan dicatat. Pada pengukuran kedua (nilai maksimum), servo 1, 2, 3, dan 4 berturut-turut akan digerakkan hingga mencapai sudut  $170^\circ$ ,  $160^\circ$ ,  $160^\circ$ , dan  $150^\circ$ . Nilai yang didapat pada pengukuran dapat dilihat pada Tabel 4.1.

Dengan menggunakan persamaan regresi linier, maka persamaan konversi yang didapat adalah

Tabel 2. Nilai Maksimum dan Minimum dari Tiap Servo

	Nilai minimum		Nilai maksimum	
	Sudut terukur ( $^\circ$ )	Nilai biner yang dibaca sensor	Sudut terukur ( $^\circ$ )	Nilai biner yang dibaca sensor
Servo 1	0	118	170	424
Servo 2	0	105	160	411
Servo 3	0	113	160	415
Servo 4	0	111	150	426

## B. Pengujian pada Sistem Kalang Terbuka

Tabel 3. Hasil Percobaan pada Sistem Kalang Terbuka

Servo	Rata-rata steady-state (derajat)	Rata-rata settling time (detik)
1	6,226	0,819
2	3,054	0,652
3	3,822	0,719
4	21,33	1,066

Dengan menggunakan sistem kalang terbuka rata-rata *steady state error* untuk tiap servo melebihi 3°. Bahkan untuk servo keempat *steady state error* yang didapat melebihi 20°. Dapat disimpulkan bahwa sistem kalang terbuka cukup buruk dalam hal akurasi dari sudut servo. Walaupun demikian, rata-rata *settling time* untuk setiap servo dapat dikatakan cukup cepat. Servo 1, 2, dan 3 dapat mencapai *steady state* dalam waktu kurang dari satu detik, sedangkan servo 4 hanya membutuhkan ± 1 detik.

## C. Pengujian pada Sistem Kalang Tertutup

Pada sistem kalang tertutup, nilai komponen PID Controller yang dipakai adalah:

- Servo 1 : P = 0,1      I = 3      D = 0
- Servo 2 : P = 0,1      I = 2,5      D = 0
- Servo 3 : P = 0,1      I = 2,55      D = 0
- Servo 4 : P = 0,1      I = 2,1      D = 0

Tabel 4. Hasil Percobaan pada Sistem Kalang Tertutup

Servo	Rata-rata steady-state (derajat)	Rata-rata settling time (detik)
1	0,549	2,484
2	0,586	2,894
3	0,451	2,494
4	0,9326	2,76

Pada servo pertama terlihat bahwa nilai rata-rata *steady-state error* (SSE) berkurang dari 6,226° menjadi 0,549°. Dapat dikatakan bahwa dengan menggunakan sistem kalang tertutup, nilai SSE berkurang dari sebanyak 5,67°. Hal ini menunjukkan bahwa sistem dari kalang tertutup mampu untuk meningkatkan akurasi dari sudut servo. Akan tetapi nilai dari *settling time* pada sistem kalang tertutup lebih lama dibandingkan sistem kalang terbuka, yaitu lebih lama 1,66 detik.

Pada servo kedua dan ketiga, rata-rata SSE-nya lebih kecil dibandingkan servo. Ini menunjukkan bahwa servo 2 dan 3 lebih akurat dibandingkan servo 1. Pada servo 2 dan 3 berturut-turut, sistem kalang tertutup mampu mengurangi SSE sebanyak 2,46° dan 3,37° dibandingkan sistem kalang terbuka. Sedangkan *settling time* mengalami penurunan sebanyak 2,24 detik dan 1,77 detik.

Servo keempat dapat dibilang sebagai servo yang paling tidak akurat karena mempunyai SSE yang lebih dari 1 derajat. Dengan sistem kalang tertutup yang dirancang, nilai SSE dapat berkurang sebanyak 20,39° dan mengalami peningkatan *settling time* sebanyak 1,6 detik.

Walaupun sistem kalang tertutup dapat mengurangi nilai SSE pada servo, waktu yang diperlukan untuk mencapai *steady state* menjadi lebih lama. Tentu saja hal tersebut dapat menjadi kelemahan dalam sistem ini karena kecepatan sistem berkurang lebih dari dua kali lipat. Selain itu terkadang pembacaan posisi servo dapat terganggu jika goyangan dari gerakan servo terlalu kuat, sehingga dapat mengurangi akurasi dari pembacaan posisi servo itu sendiri dan servo lainnya.

## V. KESIMPULAN

- Rata-rata *steady state error* pada servo 1, 2, 3, dan 4 berturut-turut dapat berkurang sebanyak 5,67°, 2,46°, 3,71°, dan 20,39°.
- Rata-rata *settling time* pada servo 1, 2, 3, dan 4 berturut-turut dapat bertambah sebanyak 1,66 detik, 2,24 detik, 1,77 detik, dan 1,69 detik.

## DAFTAR PUSTAKA

- M. Keller, S. Muller, dan N. Parspour, "Design of a transverse flux machine as joint drive for an articulated six-axis robot arm," *2016 Int. Symp. Power Electron. Electr. Drives, Autom. Motion, SPEEDAM 2016*, hal. 849–854, 2016.
- D. Z. Jokic, S. D. Lubura, dan M. M. Soja, "Closed control loop implementation for single robot axis on FPGA platform," *IFAC Proc.*, vol. 11, no. PART 1, hal. 174–179, 2012.
- H. A. Hendarto, Munadi, dan J. D. Setiawan, "ANFIS application for calculating inverse kinematics of programmable universal machine for assembly (PUMA) robot," *2014 1st Int. Conf. Inf. Technol. Comput. Electr. Eng. Green Technol. Its Appl. a Better Futur. ICITACEE 2014 - Proc.*, hal. 35–40, 2015.
- L. Aggarwal, K. Aggarwal, dan R. J. Urbanic, "Use of artificial neural networks for the development of an inverse kinematic solution and visual identification of singularity zone(s)," *Procedia CIRP*, vol. 17, hal. 812–817, 2014.
- H. S. Juang dan K. Y. Lurr, "Design and control of a two-wheel self-balancing robot using the arduino microcontroller board," *IEEE Int. Conf. Control Autom. ICCA, April 2017*, hal. 634–639, 2013.
- M. Spong, S. Hutchinson, dan M. Vidyasagar, "Robot modeling and control (Vol. 3)," hal. 299–327, 2006.
- D. Tolani, A. Goswami, dan N. I. Badler, "Real-Time Inverse Kinematics Techniques for Anthropomorphic Limbs," *Graph. Models*, vol. 62, no. 5, hal. 353–388, 2000.