

# Analisa Gerak Putar Eksentris *Continuous* Satu Massa Horisontal pada Sistem Kendali NCTF Berbasis Arduino Uno

Perwita Kurniawan<sup>1</sup>, Adi Nugroho<sup>2</sup>

<sup>1,2</sup>Program Studi Teknik Perancangan Mekanik dan Mesin  
Politeknik ATMI Surakarta.

<sup>1</sup>perwita\_k@yahoo.co.id, <sup>2</sup>adi\_nugroho@atmi.ac.id

## Ringkasan

Sistem kendali *nominal characteristic trajectory following* (NCTF) adalah sebuah sistem kendali yang terdiri dari dua buah sub sistem, yaitu *nominal characteristic trajectory* (NCT) dan kompensator. Pembuatan pengendali berdasarkan pada eksperimen *open-loop* sederhana sehingga sistem kendali NCTF sangat praktis untuk digunakan. Arduino Uno digunakan sebagai pengendali sistem putar eksentris satu massa horisontal. Perintah ke Arduino Uno diberikan oleh perangkat lunak MATLAB®, selanjutnya Arduino Uno akan menggerakkan motor dan membaca *encoder*. Performa sistem diukur dengan melakukan pengujian menggunakan variasi pembebanan dan variasi gerakan *continuous*. Variasi pembebanan yang diberikan diantaranya 0, 0,066 dan 0,156Kg. Sedangkan variasi gerakan *continuous* yang diberikan berupa gerakan bolak-balik satu kali dan gerakan bolak-balik yang berulang. Informasi yang dianalisa berupa nilai *steady state error*, *settling time*, dan *overshoot*, yang digunakan untuk menentukan performa sistem yang telah dibuat. Hasil penelitian menunjukkan bahwa sistem Kendali NCTF berbasis Arduino Uno pada Gerak Putar Eksentris *Continuous* Satu Massa Horisontal, mampu menunjukkan performa yang cukup baik. Keakuratan penunjukan posisi dapat dilihat dari kecilnya rata-rata *error*. Nilai rata-rata *error* yang terjadi sebesar  $3,85^\circ$ . Kecepatan respon yang baik dapat dilihat dari kemampuan sistem untuk menyesuaikan perintah posisi secara visual dari grafik gerakan yang dihasilkan saja. Hal ini terjadi karena nilai *settling time* dan *overshoot* tidak bisa terukur, akibat dari perintah gerak *continuous* yang selalu berubah tiap *sampling time* 0,01 detik. Sistem mampu *robust* untuk mengatasi faktor pengganggu dari luar seperti grafitasi, gesekan, inersia, dan beban untuk mekanisme putar eksentris satu massa horisontal.

**Kata kunci:** NCTF, Pengendalian posisi, Sistem putar eksentris, *Continuous*

## 1. Pendahuluan

Sistem kendali posisi dibutuhkan dalam berbagai aplikasi di bidang robotik. Berdasarkan perintah gerakannya, secara umum ada dua gerakan, yaitu perintah gerakan *point to point* dan perintah gerakan *continuous*. Kedua perintah gerakan ini menuntut keakuratan dan respon yang cepat. Sistem kendali *continuous* harus memiliki respon yang lebih cepat dibandingkan dengan sistem kendali *point to point*. Hal ini disebabkan karena perintah yang terus berubah seturut dengan *sampling time*-nya.

Sistem kendali yang banyak digunakan untuk pengendali posisi adalah sistem kendali PID (*Proportional Integral Derivative*). Dalam membangun sebuah sistem kendali PID, diperlukan sebuah model matematis yang cenderung rumit dan memerlukan

pengetahuan tentang teori sistem kendali dan rancang bangun secara lebih mendalam. Model matematis membutuhkan banyak variabel seperti nilai inersia dan beberapa variabel motor. Selain itu, parameter identifikasi pada umumnya merepotkan dan membutuhkan banyak waktu pada saat proses pengerjaan. Hal ini akan menyulitkan bagi seorang desainer mekanik yang akan merancang sebuah sistem kendali, namun tidak mempunyai keahlian yang baik dalam sistem kendali. Kemudahan desain sistem kendali dan kesederhanaan struktur pengendali sangat diperlukan dalam aplikasi praktis.

Sistem kendali NCTF (*nominal characteristic trajectory following*) telah dikembangkan sebagai pengontrol praktis pada sistem kendali posisi, untuk mengatasi permasalahan tersebut [1]. Struktur sistem kendali NCTF lebih sederhana dan *robustness*-nya lebih baik jika dibandingkan dengan sistem kendali PID [2]. Ketidaktepatan posisi akibat efek gesekan mampu diimbangi oleh sistem kendali NCTF secara efektif [3]. Sistem kendali NCTF juga memiliki kinerja posisi dan ketahanan yang baik [4].

Keunggulan sistem kendali NCTF dapat dilihat pada desainnya yang praktis. Sistem kendali NCTF tetap membutuhkan sebuah model, akan tetapi model yang dibangun bukanlah model yang sempurna. Identifikasi parameter dari model tersebut bisa dilakukan dengan pendekatan yang terstruktur. Dengan kemudahan ini seorang desainer mekanik bisa memiliki peluang untuk menciptakan sebuah mekanisme mekanik terlebih dahulu, dan memikirkan sistem pengendalinya dengan lebih mudah kemudian. Proses *reverse engineering* secara sistem kendali dari sebuah peralatan yang sudah ada pun bisa dilakukan dengan sistem kendali NCTF.

Peningkatan kepraktisan sistem kendali NCTF dapat dilakukan dengan penggunaan pengendali yang sederhana seperti Arduino Uno. Arduino Uno mampu meringkas sistem karena program pengendalian bisa dibangun dengan menggunakan perangkat lunak Matlab Simulink. Perangkat lunak Matlab Simulink ini menggunakan program yang berbasis diagram blok. Kemudahan penggunaan Arduino Uno didukung oleh sumber pembelajaran yang terbuka dan komunitas Arduino yang banyak.

Sistem kendali NCTF sudah banyak dikembangkan oleh para peneliti, diantaranya pengaplikasian sistem tersebut pada sistem gerak linear maupun gerak berputar, baik secara *Point to Point* (PTP) [5,6,7,8] maupun secara *Continuous Path* (CP) [9,10]. Para peneliti berusaha untuk menemukan langkah yang paling optimal dan sederhana, salah satunya menemukan kompensator yang berbasis PI. Namun sampai saat ini, kompensator berbasis PI pada sistem kendali NCTF yang diaplikasikan untuk mekanisme PTP dan CP tersebut hanya digunakan pada kendali posisi putar dengan massa yang seimbang [11]. Belum pernah dilaporkan sebuah analisa gerakan *continuous* pada sistem kendali NCTF untuk mengendalikan posisi putar dengan massa tidak seimbang. Pengendalian posisi putar dengan massa yang tidak seimbang pernah dilaporkan untuk gerakan *point to point* [12].

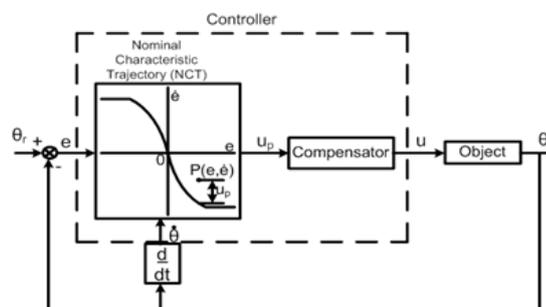
Tujuan dari penelitian ini adalah untuk melihat performa dan respon dari sistem apabila perintah gerakan secara *continuous* diberikan pada sistem kendali posisi putar horisontal dengan massa yang tidak seimbang.

## 2. Sistem Kendali NCTF

Sebuah sistem kendali NCTF terdiri dari dua buah bagian yang saling terkait satu sama lain. Kedua bagian tersebut adalah NCT dan kompensator. NCT merupakan karakteristik dasar dari sebuah obyek, sedangkan kompensator merupakan sebuah

mekanisme yang menuntut obyek mengikuti karakteristik dasarnya. Dalam sistem kendali ini, *error* merupakan masukan dari pengendali, dan sinyal  $u$  merupakan keluaran dari pengendali. Sinyal  $u$  merupakan sinyal yang digunakan untuk menggerakkan obyek.

Sistem kendali NCTF bekerja dengan prinsip dasar membandingkan masukan berupa gerak obyek dengan *error-rate* yang disajikan oleh NCT, pada nilai *error* tertentu. Perbedaan nilai *error-rate* aktual dari obyek dibandingkan dengan yang ada di NCT adalah sinyal  $u_p$ , yang merupakan keluaran NCT. Jika gerak obyek mengikuti NCT dengan sempurna, maka besarnya nilai sinyal  $u_p$  adalah nol, sehingga kompensator tidak melakukan aksi untuk mengkoreksi gerakan. Sebaliknya, jika nilai sinyal  $u_p$  tidak sama dengan nol, maka kompensator akan melakukan aksi untuk mengkoreksi gerakan agar nilai dari sinyal  $u_p$  mencapai nol.



Gambar 1. Sistem Kendali NCTF

Pembuatan sistem kendali NCTF berdasarkan dari sebuah eksperimen *open-loop* sederhana. Ada tiga tahapan dalam pembuatan sistem kendali NCTF, yaitu:

1. Menggerakkan obyek dengan sinyal masukan *stepwise*, kemudian diukur perpindahannya dan kecepatannya.
2. Membuat NCT berdasarkan data dari langkah pertama.
3. Membuat kompensator berdasar data yang diperoleh dari NCT.

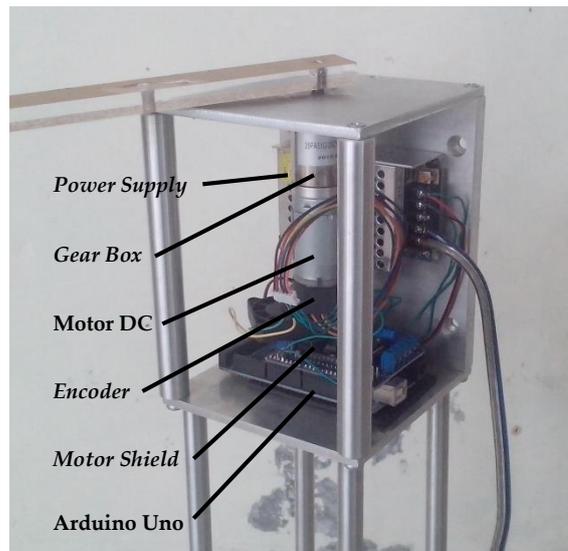
### 3. Set-up Eksperimen

Eksperimen diawali dengan membuat susunan alat uji yang terdiri dari:

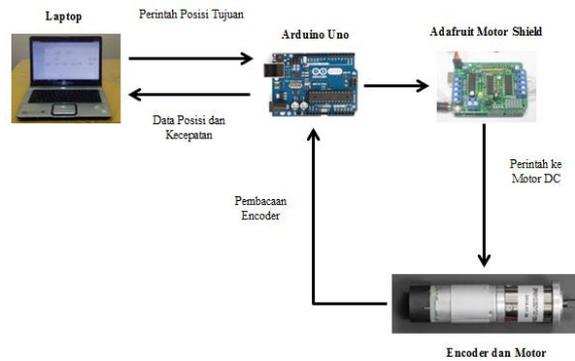
1. *Power supply* berarus 3 Ampere,
2. Arduino Uno,
3. Motor DC yang terangkai bersama transmisi dan *encoder*-nya,
4. *Adafruit motor shield*.

Semua peralatan diatas disatukan oleh sebuah rangka alumunium. Rangkaian alat uji secara keseluruhan dapat dilihat pada Gambar 2. Dua buah beban dengan massa 0,066kg, dan 0,156kg disiapkan untuk menguji *robustness* dari sistem.

*Set-up* eksperimen diawali dengan memberikan perintah pergerakan posisi sudut motor pada perangkat lunak Matlab. Dengan menggunakan perintah blok Simulink pada Matlab, data posisi tujuan akan dikirim menuju Arduino Uno. Perintah posisi sudut tujuan motor dapat dieksekusi oleh motor DC apabila sudah melewati *Adafruit motor shield*. Hasil pergerakan posisi motor DC akan dibaca oleh *encoder* dan akan dikirim menuju laptop melalui Arduino Uno. *Set-up* eksperimen dapat dilihat pada Gambar 3.



Gambar 2. Alat uji



Gambar 3. Set-up eksperimen

#### 4. Pembuatan NCT

Pembuatan NCT diawali dengan melakukan sebuah eksperimen sederhana. Dalam perangkat lunak Matlab, diberikan sebuah sinyal step untuk menggerakkan motor DC tanpa beban lengan. *Encoder* akan membaca perubahan posisi dan perubahan kecepatan pada saat yang bersamaan. Hasil eksperimen sederhana ini dapat dilihat pada Gambar 4, yang menunjukkan sinyal masukan *step*, kecepatan dan perpindahan. Eksperimen sederhana ini dilakukan pada dua arah gerakan yang berbeda, yaitu searah dan berlawanan arah jarum jam.

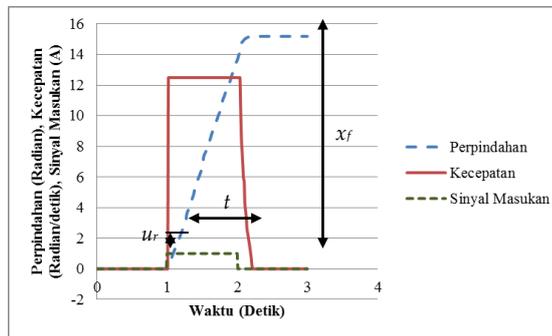
Pembuatan NCT membutuhkan beberapa parameter yang dapat dilihat pada Gambar 4, diantaranya adalah:

1.  $x_f$  yang merupakan perpindahan maksimal,
2.  $u_r$  yang merupakan tinggi sinyal input, dan
3.  $t_r$  yang merupakan waktu sinyal input.

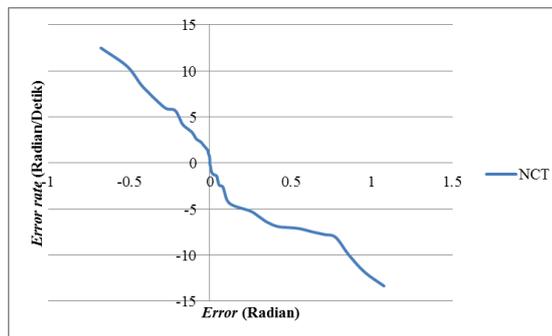
Ketiga parameter tersebut dapat digunakan untuk menghitung konstanta K, dengan menggunakan rumusan yang terdapat pada Persamaan (1). Konstanta K digunakan untuk menentukan nilai kompensator.

$$K = \frac{x_f}{u_r \cdot t_r} \quad (1)$$

Data hasil pemutaran motor DC dilakukan pada dua arah yang berbeda secara *open-loop*, dan diambil sejak sinyal masukan dihilangkan. Nilai perubahan kecepatan dan perubahan percepatan yang terjadi sejak sinyal masukan dihilangkan digabungkan. Gabungan kedua data ini merupakan karakteristik dari sistem yang disebut dengan NCT. Untuk mendapatkan NCT yang utuh, maka data pada arah putaran searah jarum jam dan berlawanan arah jarum jam digabungkan menjadi satu, seperti yang diperlihatkan pada Gambar 5.



Gambar 4. Respon *open-loop* untuk membuat NCT



Gambar 5. NCT

## 5. Pembuatan Kompensator

Kompensator terdiri dari dua buah konstanta, yaitu Konstanta Proporsional dan Konstanta Integral. Pembuatan kompensator diawali dengan pembuatan daerah batas kestabilan sistem, dengan cara menggerakkan mekanisme NCT yang sudah didapatkan sebelumnya dengan menggunakan konstanta proporsional saja. Besarnya nilai konstanta proporsional diperoleh dengan cara menggerakkan sistem dengan konstanta proporsional saja, dan mengulanginya secara bertahap dari nilai konstanta proporsional kecil, hingga diperoleh respon gerak osilasi yang tak berhingga. Konstanta proporsional ini dinamakan dengan  $K_{pu}$  atau  $K_p$  *ultimate* (bernilai 0,39). Nilai  $K_{pu}$  dapat digunakan untuk menghitung rasio redaman ( $\zeta$ ) dengan menggunakan Pers. 2.

$$\zeta = K_{pu} \frac{\alpha K}{2\omega_n} \quad (2)$$

Daerah batas kestabilan sistem merupakan daerah batas yang terbentuk dari dua buah variabel, yaitu rasio redaman ( $\zeta$ ) dan frekuensi natural ( $\omega_n$ ). Kedua variabel ini belum dapat diketahui nilainya. Sedangkan  $\alpha$  diperoleh dengan menggunakan gradien NCT pada Gambar 5, pada daerah yang mendekati titik nol. Dengan menggunakan dasar Persamaan 2, maka dapat dibuat daerah batas kestabilan sistem, dengan bantuan perangkat lunak Excel. Data hasil pengolahan data di perangkat lunak Excel digunakan untuk membuat grafik, dengan sumbu X adalah frekuensi natural dikalikan dengan *sampling time* (ditentukan oleh peneliti sebesar 0,01detik), sedangkan sumbu Y adalah rasio redaman. Grafik ini dinamakan grafik batas kestabilan, seperti yang disajikan pada Gambar 6.

Grafik batas kestabilan berfungsi untuk membatasi daerah stabil yang berada di kiri dan bawah batas, dengan daerah yang tidak stabil yang berada di kanan dan atas batas. Angka keamanan sebesar 80% dari nilai rasio redaman dan frekuensi natural, ditentukan oleh peneliti agar didapatkan kepastian bahwa sistem akan bekerja pada kondisi stabil. Dengan menggunakan data rasio redaman dan frekuensi natural pada kondisi 80%, maka perhitungan untuk menentukan Konstanta Proporsional dan Konstanta Integral dapat dilakukan dengan Pers.3 dan Pers. 4.

$$K_p = \frac{2\zeta\omega_n}{\alpha K} \tag{3}$$

$$K_i = \frac{\omega_n^2}{\alpha K} \tag{4}$$

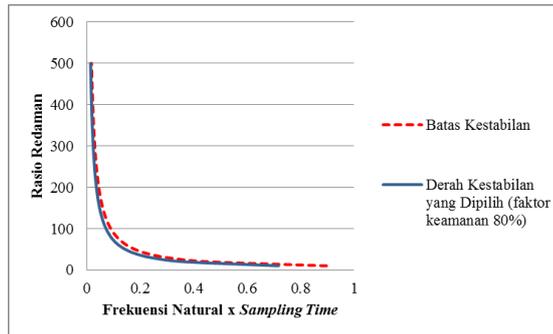
Variasi konstanta integral akan muncul sebagai akibat dari variatifnya frekuensi natural dalam daerah kestabilan yang dipilih. Dalam daerah kestabilan yang dipilih, peneliti melakukan pendekatan dengan menentukan tiga buah variasi rasio redaman, yaitu sebesar 10, 50 dan 150, seperti yang diperlihatkan pada Gambar 7.

Konstanta Proporsional ( $K_p$ ) dihitung dengan menggunakan Pers.3, sedangkan Konstanta Integral ( $K_i$ ) dihitung dengan menggunakan Pers.4. Kombinasi variabel pada kondisi X, Y dan Z, digunakan sebagai dasar perhitungan. Pada Tabel 1 ditampilkan hasil perhitungan  $K_p$  dan  $K_i$ . Besarnya nilai  $K_i$  berbanding lurus dengan nilai frekuensi natural. Jika nilai frekuensi natural semakin besar maka nilai  $K_i$  juga akan semakin besar.

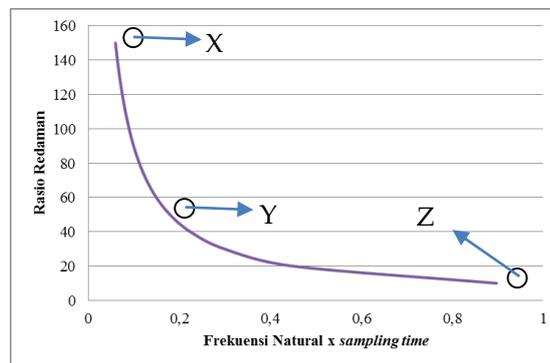
Setelah mendapatkan variasi perhitungan konstanta proporsional dan konstanta integral pada ketiga titik, maka dilakukan pengujian pada alat eksperimen. Hasil pengujian tersebut dibandingkan satu dengan yang lain, kemudian diambil respon yang terbaik. Respon yang terbaik dihasilkan oleh kompensator Y. Dengan demikian dapat disimpulkan bahwa kombinasi konstanta proporsional dan konstanta integral sistem menggunakan variasi pada kompensator Y.

Tabel 1. Karakteristik Kompensator

Keterangan	Kompensator		
	X	Y	Z
Rasio redaman	150	50	10
Frekuensi natural [Hz]	5,98	17,95	89,75
$K_p$	0,31	0,31	0,31
$K_i$	0,01	0,07	1,75



Gambar 6. Grafik kestabilan sistem

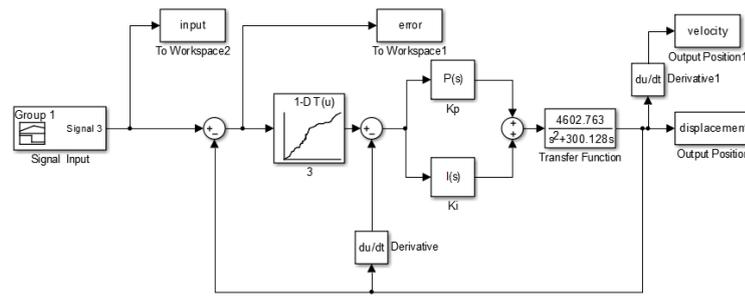


Gambar 7. Variasi Kompensator

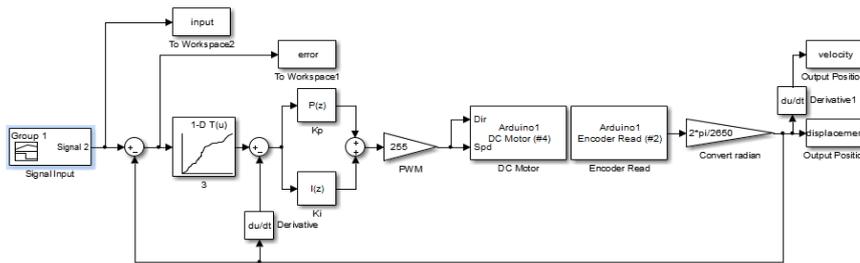
## 6. Evaluasi Performa Sistem

Tahapan selanjutnya adalah pengujian sistem dengan memasukkan konstanta proporsional dan konstanta integral pada blok  $K_p$  dan  $K_i$  di program blok Simulink Matlab. Dua buah metode yang digunakan dalam pengujian sistem secara umum adalah metode simulasi dan metode eksperimen. Metode simulasi dilakukan dengan menjalankan simulasi pada program Matlab saja (program hanya melakukan perhitungan matematis gerakan) dengan menggunakan *transfer function*-nya. Sedangkan metode eksperimen dilakukan dengan cara memerintahkan alat uji secara langsung untuk melakukan gerakan. Gambar 8 adalah gambar program blok Simulink Matlab untuk pengujian secara simulasi, sedangkan Gambar 9 adalah program blok Simulink Matlab untuk pengujian secara eksperimen.

Tahapan sesudah program blok dibuat adalah dengan mengeksekusi program blok tersebut. Hasil eksekusi gerakan inilah yang menjadi hasil pengujian. Hasil pengujian antara pengujian simulasi dan eksperimen dibandingkan untuk dapat dianalisa. Perbandingan hasil ini dapat kita lihat pada Gambar 10, dan Gambar 11. Hasil pengujian simulasi secara umum sedikit bersolisasi pada kondisi *steady*, namun respon pengujian simulasi tampak lebih cepat. Pada Tabel 2 tampak bahwa pengujian secara eksperimen tidak menunjukkan hasil yang jauh berbeda dengan pengujian secara simulasi. Dengan mengacu pada data tersebut, maka dapat disimpulkan bahwa respon sistem masih cukup baik.



Gambar 8. Program Blok Simulink untuk Pengujian Simulasi



Gambar 9. Program Blok Simulink untuk Pengujian Eksperimen

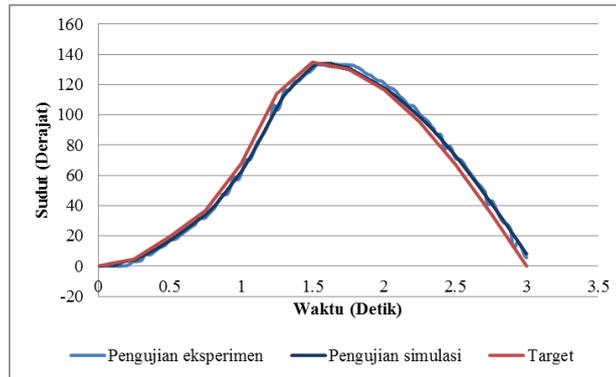
Tabel 2. Perbandingan hasil simulasi dan eksperimen

Sinyal Masukan [Pola]	Metode Pengujian	Rata-rata Error [°]
Bolak-balik	Simulasi	0,07
	Eksperimen	0,08
Berulang	Simulasi	0,03
	Eksperimen	0,06

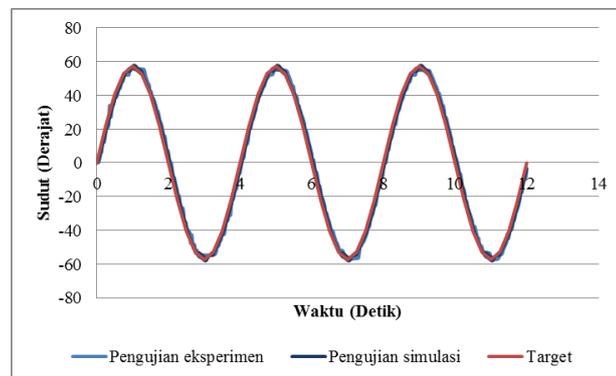
Pengujian secara eksperimen dengan menggunakan berbagai variasi perintah *continuous* dan variasi massa pembebanan dapat digunakan untuk melihat *Robustness* sebuah sistem. Ketika dilakukan berbagai pembebanan (*robust*), sebuah sistem kendali dapat dikatakan baik apabila memiliki respon yang hampir sama dengan simulasi.

Respon yang relatif sama biasanya akan muncul pada pengujian yang menggunakan pembebanan yang sama juga, namun ketika pembebanan yang berbeda diberikan, maka respon yang terjadi pasti akan berbeda. Semakin besar beban diberikan maka kestabilan gerakan *continuous* semakin berkurang. Pada Gambar 12 respon sistem masih mendekati hasil pengujian tanpa beban, namun pada Gambar 13 dan Gambar 14, respon sistem semakin menjauhi pengujian tanpa beban, namun masih normal. Hal ini juga berlaku pada pengujian eksperimen dengan gerakan berulang. Pada Gambar 15 respon sistem masih mendekati hasil pengujian tanpa beban, namun pada Gambar 16 dan Gambar 17, respon sistem semakin menjauhi pengujian tanpa beban, namun masih normal.

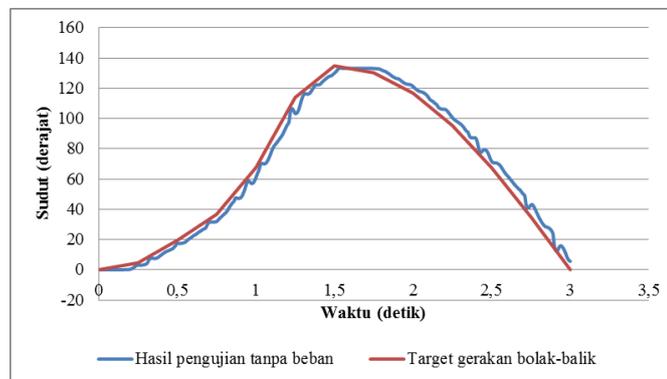
Pada Gambar 12, Gambar 13, dan Gambar 14, tersaji grafik respon sistem pada pengujian eksperimen secara bolak-balik, sedangkan pada Gambar 15, Gambar 16 dan Gambar 17, tersaji grafik respon sistem pada pengujian eksperimen secara berulang. Data grafik hasil pengujian secara eksperimen tersebut terangkum pada Tabel 3. Pada Tabel 3 tampak bahwa nilai rata-rata *error* cukup bervariasi, dari nilai terendah sebesar 2,75 ° hingga 5,27 °. Untuk sebuah sistem yang *continuous*, hal ini tampak normal dari grafik, karena lebih disebabkan oleh keterlambatan sinyal.



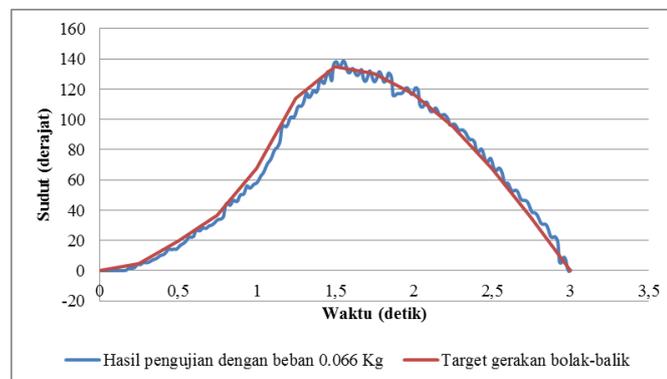
Gambar 10. Perbandingan hasil simulasi dan eksperimen pada pola target gerakan bolak-balik



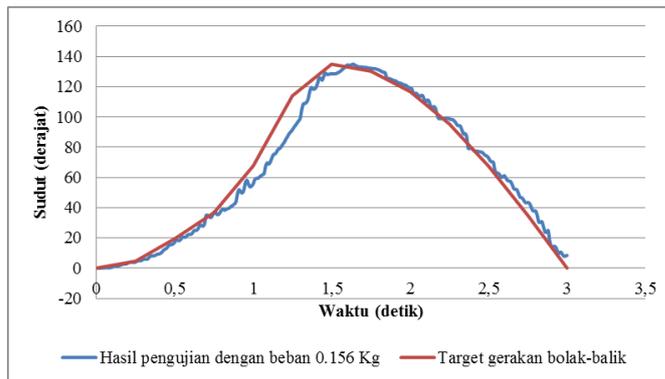
Gambar 11. Perbandingan hasil simulasi dan eksperimen pada pola target gerakan berulang



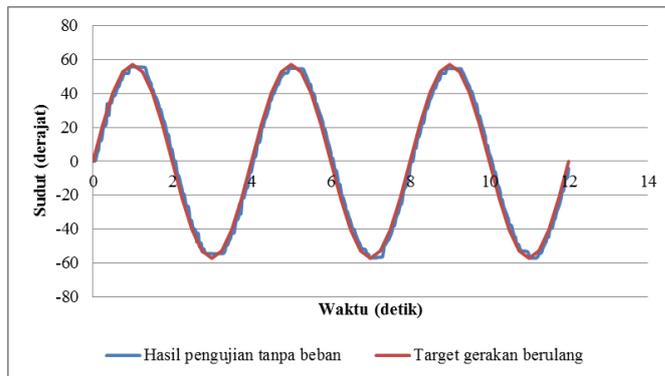
Gambar 12. Respon sistem secara eksperimen tanpa beban uji



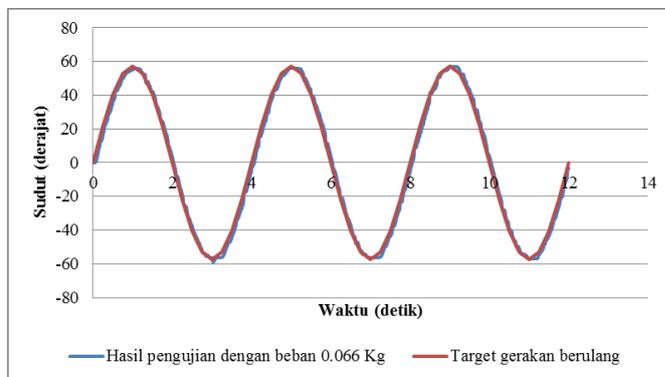
Gambar 13. Respon sistem secara eksperimen dengan beban uji 0,066kg



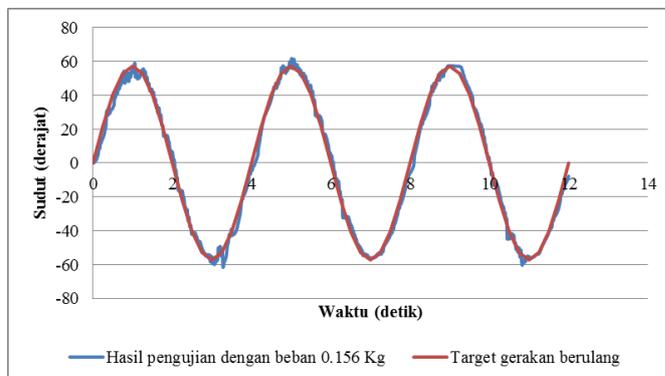
Gambar 14. Respon sistem secara eksperimen dengan beban uji 0,156kg



Gambar 15. Respon sistem secara eksperimen tanpa beban uji



Gambar 16. Respon sistem secara eksperimen dengan beban uji 0,066kg



Gambar 17. Respon sistem secara eksperimen dengan beban uji 0,156kg

Tabel 3. Data hasil uji secara eksperimen

Massa Uji [Kg]	Sinyal Masukan [Pola]	Rata-rata Error [°]
0	Gerakan bolak-balik	4,82
	Gerakan berulang	3,53
0.066	Gerakan bolak-balik	3,92
	Gerakan berulang	2,75
0.156	Gerakan bolak-balik	5,27
	Gerakan berulang	2,81

## 7. Kesimpulan

Sistem Kendali NCTF Berbasis Arduino Uno pada Gerak Putar Eksentris *Continuous* Satu Massa Horizontal, mampu menunjukkan performa yang cukup baik. Hal ini tampak dari Gambar 12, Gambar 13, Gambar 14, Gambar 15, Gambar 16, dan Gambar 17. Kecepatan respon yang baik dapat dilihat dari kemampuan sistem menyesuaikan dengan perintah posisi, namun nilai *settling time* dan *overshoot* tidak bisa terukur karena perintah gerak *continuous* selalu berubah tiap *sampling time* 0,01 detik. Keakuratan penunjukan posisi dapat dilihat dari kecilnya rata-rata nilai *rata-rata error* yang terjadi sebesar 3,85 °. Sehingga dapat disimpulkan bahwa sistem mampu *robust* untuk mengatasi faktor luar pengganggu seperti grafitasi, gesekan, inersia, dan beban untuk mekanisme putar eksentris satu massa horizontal. Hal ini tampak dari hasil pengujian secara eksperimen.

## Daftar Pustaka

- [1] Wahyudi. *New practical control of PTP positioning systems*. Ph.D. Thesis. Tokyo Institute of Technology, 2002.
- [2] Chong and Sato, "Practical Control of Non-Friction Mechanism for Precision Positioning," *Int. Conf. on Control, Automation, and System*, Seoul, Korea, 2008.
- [3] Wahyudi, K. Sato and A. Shimokohbe, "Robustness Evaluation of Three Friction Compensation Methods for Point-to-point (PTP) Positioning Systems," *Robotics and Autonomous System*, Vol. 52, Issues 2-3, pp. 247-256, 2005.
- [4] Wahyudi, K. Sato and A. Shimokohbe, "Characteristics of Practical Control for Point-to-point Positioning Systems: Effect of design parameters and actuator saturation," *Precision Engineering*, Vol. 27, No. 2, pp. 157-169, 2003.
- [5] Purtojo, R. Akmeliawati, and Wahyudi, "Fuzzy-based NCTF Controller for PTP Positioning: Fuzzy Membership and Rule Based Modifications," *3rd International Conference on Mechatronics (ICOM'08)*. Kuala Lumpur, Malaysia, 2008.
- [6] Purtojo, R. Akmeliawati R., and Wahyudi, "Intelligent Fuzzy-based NCTF Controller for Point-to-point Positioning System: Reduced Controller Structure," *The First International Seminar on Science and Technology (ISSTEC 2009)*, Yogyakarta, Indonesia, 2009.
- [7] Purtojo, Wahyudi, R. Akmeliawati & A.A. Shafie, "Fuzzy-based NCTF Control of PTP Positioning System, Experimental Results," *4th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 1741 – 1747, 2009.
- [8] Purtojo. *Fuzzy-based NCTF control of point-to-point positioning systems*. Master Dissertation. International Islamic University Malaysia, 2010.
- [9] K. Sato and A. Maeda, "Practical control method for precision motion—Improvement of NCTF control method fo CM control," *Precision Engineering*, Vol. 33, pp. 173-186, 2009.

- [10] Fitri and Rini, "Study Review on NCTF Controller for Point-To-Point Positioning System," *11th International Conference on Control, Automation, and System*, Gyeonggi-do, Korea, 2011.
- [11] G.J. Maeda and K. Sato, "Practical control method for ultra-precision positioning using a ballscrew mechanism," *Precision Engineering*, Vol. 32, No. 4, pp. 309-318, 2008.
- [12] P.Kurniawan, Purtojo, Herianto and G. Nugroho, "Perancangan Sistem Kendali NCTF Berbasis Arduino Uno untuk Sistem Putar Eksentris Satu Massa Horisontal," *Proceeding Seminar Nasional Tahunan Teknik Mesin XIV*, Banjarmasin, 2015.