

Kendali Formasi Mobile Robot Berdasarkan Jarak Menggunakan Algoritma *Cosinus*

ANGGORO DWI NUR ROHMAN¹, M. AZIZ MUSLIM², BAMBANG SISWOJO³

^{1,2,3}Program Studi Magister, Teknik Elektro, Universitas Brawijaya, Indonesia
Email : anggoro_dwi@student.ub.ac.id

Received 30 November 201x | *Revised* 30 Desember 201x | *Accepted* 30 Januari 201x

ABSTRAK

Kendali formasi adalah topik penelitian kendali multi-robot, dimana sekelompok robot dapat mencapai formasi tertentu dan mempertahankannya ketika berpindah ke arah yang diinginkan. Salah satu pengembangan kendali formasi adalah kendali formasi berdasarkan jarak dimana setiap individu robot menggunakan informasi jarak antar sesamanya untuk mencapai tujuan formasi. Banyak pengembangan yang dilakukan menggunakan model yang sederhana dan membutuhkan pengembangan lebih lanjut untuk penerapan kendali ke model yang lebih nyata. Pada kondisi awal, robot tidak dapat menentukan koordinat tetangganya menggunakan informasi jarak saja. Dikembangkan algoritma cosinus untuk menemukan koordinat tetangga menggunakan informasi jarak saja sebagai solusi kondisi awal. Dari hasil simulasi percobaan, algoritma cosinus dapat menemukan koordinat tetangganya dengan waktu yang cukup singkat dengan rata-rata waktu 6.5 detik dan minimal settling time yang di capai pada kendali formasi berdasarkan jarak menggunakan model robot holonomic cukup baik yaitu mencapai 2 detik.

Kata kunci: *Kendali Formasi, Multi-Robot, Algoritma Cosinus, Mobile Robot.*

ABSTRACT

Formation control is a research topic of multi-robot control, where a group of robots can reach a certain formation and defend it when moving in the desired direction. Formation control distance-based is one of the development where each individual robot uses the distance information to achieve the formation goal, but it requires further development to a more real model. In the initial condition, the robot cannot determine the coordinates of its neighbors using only distance information. A cosine algorithm is developed to find the coordinates of neighbors using only distance information as a solution to the initial conditions. From the results of experimental simulations, the cosine algorithm can find the coordinates of its neighbors in a fairly short time with an average time of 6.5 seconds and the minimum settling time achieved in formation control based on distance using a holonomic robot model is quite good, reaching 2 seconds.

Keywords: *Formation Control, Multi-robot, Cosine Algorithm, Mobile Robot.*

1. PENDAHULUAN

Kendali formasi adalah topik penelitian kendali multi-robot untuk memecahkan permasalahan koordinasi pergerakan. Kendali formasi bertujuan untuk mengendalikan sekelompok robot dalam mencapai formasi tertentu dan dapat mempertahankan formasi tersebut ketika bermanuver menuju arah yang diinginkan. Dari penjabaran oleh Guanghua (Guanghua et al., 2013), pengembangan kendali formasi dilakukan dari beberapa algoritma strategi. Seperti yang dikembangkan oleh Wang (Wang et al., 2014) menggunakan strategi *leader-follower*, menggunakan *Fuzzy-Logic* sebagai tingkah laku robot oleh Ferik (Ferik et al., 2016) dan menggunakan *struktur virtual* dimana sekelompok robot memiliki titik referensi sebagai satu robot oleh Xuen-ren (Li et al., 2015).

Penjabaran oleh Kwang-Kyo (Oh et al., 2015) bahwa dari berbagai pengembangan kendali formasi dapat diambil garis besar menjadi 3 bagian, yaitu berdasarkan posisi, perpindahan, dan jarak. Ketiga bagian tersebut tertuju pada jawaban dari pertanyaan, "variable apa yang digunakan sebagai sensor" dan "variabel apa yang aktif dikendalikan oleh sistem multi-robot untuk mencapai formasi yang diinginkan". Dikhususkan pada kendali formasi berdasarkan jarak, variabel yang dikendalikan adalah variabel jarak antar agent yang terhubung dan koordinat yang digunakan tidak mengacu pada koordinat global. Sehingga pada penerapannya, formasi berdasarkan jarak menggunakan sensor yang lebih sedikit dibanding dengan posisi dan perpindahan. Namun salah satu permasalahan pada metode tersebut adalah penerapan model yang lebih nyata. Pengembangan formasi berdasarkan jarak telah dikembangkan menggunakan teori graph dengan model sederhana pada *single* dan *double integrator* oleh Kwang-Kyo (Oh & Ahn, 2014), menggunakan informasi jarak untuk mengendalikan model sederhana *double integrator* oleh Xiaoyu (Cai & de Queiroz, 2014), menggunakan konsensus antara robot untuk mencapai bentuk formasi pada model *single integrator* sederhana Oleh Deghat (Deghat et al., 2016), mengendalikan bentuk formasi menggunakan *adaptive control* untuk mengestimasi kecepatan tetangga dari model robot yang sederhana oleh Sung-Mo (Kang et al., 2014), menggabungkan kendali formasi berdasarkan jarak dan perpindahan untuk mengendalikan model robot yang sederhana oleh Myoung-Chul (Park & Ahn, 2015), dan menggunakan kendali *Proportional-Integral* (PI) untuk mengendalikan jarak setiap model robot yang sederhana oleh Rozenheck (Rozenheck et al., 2015).

Kendali PI pada penelitian sebelumnya tidak dapat langsung diterapkan menggunakan sensor jarak karena kendali tersebut mengambil informasi jarak menggunakan selisih dari koordinat global kartesian setiap robot. Sedangkan dalam praktiknya robot hanya bisa mengukur jarak dan tidak mengetahui koordinat dari robot tetangganya. Selain itu, informasi jarak adalah besaran yang tidak memiliki arah sehingga robot diharuskan untuk mengetahui koordinat tetangganya untuk mencapai jarak yang diinginkan.

Terinspirasi dari penelitian oleh Qiang (Qiang et al., 2017a) menggunakan sensor *Ultra Wide band* (UWB) yang dipasang pada robot sebagai sensor jarak untuk mengendalikan formasi robot. Sensor *Ultra Wide Band* kerap digunakan untuk mendeteksi lokasi seperti pada penelitian oleh Kexing (Guo et al., 2020) dimana sensor UWB menggantikan sensor berbasis *vision* dan juga penerapan penerapan UWB oleh Qiang (Qiang et al., 2017b) dapat mendeteksi lokasi dimana UWB dipasang secara statis sehingga dapat dimanfaatkan untuk mengendalikan sekelompok robot secara terpusat. Pada penelitian ini akan dikembangkan sebuah algoritma untuk menemukan koordinat tetangga menggunakan informasi jarak dan digunakan untuk mengetahui koordinat tetangga pertama kali sebelum menjalankan kendali formasi berdasarkan jarak. nilai kondisi awal pada kendali formasi berdasarkan jarak. Percobaan akan menggunakan model robot holonomic dengan harapan menjadi langkah awal untuk

mengembangkan kendali formasi berdasarkan jarak menggunakan model robot yang lebih nyata.

2. METODE

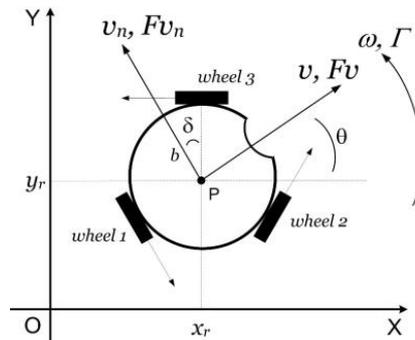
2.1 Model Robot

Robot menggunakan tiga buah motor yang dihubungkan pada *omniwheel* sehingga robot dapat bergerak kesegala arah (**Error! Reference source not found.**). Berikut adalah model dari robot *holonomic* dalam bentuk *state-space* (Correia et al., 2012).

$$\dot{x}(t) = A_r x(t) + B_r u(t) + K_r \text{sgn}(x(t)) \quad (1)$$

$$y(t) = Cx(t) \quad (2)$$

Dimana vector $u(t) = [u_1(t) \ u_2(t) \ u_3(t)]^T$ adalah masukan model bersatuan *volt* yang memiliki tegangan motor $-6 \leq u_i \leq 6$ *volt*. Vektor $y(t) = x(t) = [v(t) \ v_n(t) \ w(t)]^T$ adalah kecepatan robot yang akan diperoleh dari sensor percepatan, dimana v adalah kecepatan robot bergerak maju/mundur atau koordinat x berdasarkan kerangka robot, v_n adalah kecepatan robot untuk menyamping kiri/kanan atau koordinat sumbu y berdasarkan kerangka robot, dan $w(t)$ adalah kecepatan rotasi dari kerangka robot. Matrix $A_r \in R^{3 \times 3}$ dan $B_r \in R^{3 \times 3}$ adalah parameter fisik robot yang diperoleh dari persamaan fisika. Matrix $K_r \in R^{3 \times 3}$ adalah parameter *friction* dari robot yang diestimasi dari hasil identifikasi sistem dengan cara percobaan.



Gambar 1. Geometri Rangka Robot (Correia et al., 2012)

2.2 Kendali Robot Holonomic

Kendali robot holonomic akan dibagi menjadi dua mode. Perbedaan dari kedua mode tersebut adalah setpoint dari kendalinya dimana mode satu akan memiliki setpoint kecepatan koordinat kerangka robot dan koordinat kerangka robot. Lebih lengkap kegunaan kedua kendali tersebut akan dijelaskan pada bab metode algoritma *cosinus*.

Mode Satu. Kendali robot mode satu bertujuan untuk mencapai kecepatan robot yang diinginkan. Untuk mencapai tujuan tersebut akan menggunakan persamaan kendali sebagai berikut

$$u_{c1}(t) = -K_1^c x(t) + N_1^c r_1^c \quad (3)$$

$$N_1^c = -[C(A_r - B_r K_1^c)^{-1} B_r]^{-1} \quad (4)$$

Dimana vector $r_1^c = [v^* \ v_n^* \ w^*]$ adalah setpoint kendali mode satu, $K_1^c \in R^{3 \times 3}$ dan $N_1^c \in R^{3 \times 3}$ konstanta yang diperoleh dari optimasi persamaan *Riccati* terhadap matriks A_r dan B_r , dan konstanta yang diperoleh dari *inverse state-space* pada keadaan *steady state*. Dengan menggabungkan persamaan (1) dan persamaan (3) akan mendapatkan persamaan *state-space* robot yang baru

$$\dot{x}(t) = A_c x(t) + B_c r_1^c + K_r \text{sgn}(x(t)) \quad (5)$$

$$A_c = (A_r - B_r K_1^c) \quad (6)$$

$$B_c = B_r N_1^c \quad (7)$$

Mode Dua. Kendali robot mode dua bertujuan untuk mencapai titik koordinat tertentu berdasarkan koordinat frame robot. Untuk mencapai tujuan tersebut akan dimodifikasi persamaan (1) dengan menambah koordinat *frame* robot pada *state* nya, $x_{c2}(t) = [x_r(t) \ y_r(t) \ \theta_r(t) \ v(t) \ v_n(t) \ w(t)]^T$

$$\dot{x}_{c2}(t) = A_{c2} x_{c2}(t) + B_{c2} u_{c2}(t) + K_{c2}(x_{c2}(t)) \quad (8)$$

$$y_{c2}(t) = C_{c2} x_{c2}(t) \quad (9)$$

$$A_{c2} = \begin{bmatrix} 0 & I \\ 0 & A_r \end{bmatrix} \in R^{6 \times 6} \quad (10)$$

$$B_{c2} = [0 B_r] \in R^{6 \times 3} \quad (11)$$

$$K_{c2}(x) = \begin{bmatrix} 0 & 0 \\ 0 & K_r \text{sgn}(x) \end{bmatrix} \in R^{6 \times 3} \quad (12)$$

Menggunakan persamaan (8) dapat ditentukan persamaan $u_{c2}(t)$ sebagai persamaan kendalinya.

$$u_{c2}(t) = -K_2^c x_{c2}(t) + N_2^c r_2^c \quad (13)$$

Dimana $r_2^c = [x_r^* \ y_r^* \ \theta_r^* \ v^* \ v_n^* \ w^*]^T$, $K_2^c \in R^{3 \times 6}$ dan $N_2^c \in R^{3 \times 6}$ adalah setpoint kendali mode dua, konstanta yang diperoleh dari optimasi persamaan *Riccati* terhadap matriks A_{c2} dan B_{c2} dan konstanta yang diperoleh dari *inverse state-space* pada keadaan *steady state*.

2.3 Kendali Formasi Berdasarkan Jarak

Didefinisi sebuah *undirect graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ terdiri dari himpunan simpul \mathcal{V} dan himpunan sisi $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Didefinisi $n \triangleq |\mathcal{V}|$ sebagai jumlah dari node dan $m \triangleq |\mathcal{E}|$ sebagai jumlah dari sisinya. Didefinisi $p = [x_1^T \ \dots \ x_n^T]^T \in R^{3n}$, dimana $x_i \in R^3$ dan $x_i \neq x_j$ untuk semua $i \neq j$. Dinotasikan vektor posisi relatif $e_k \triangleq x_j - x_i$ dan semua vector sisi $e = [e_1^t \ \dots \ e_m^t] \in R^{3m}$. Didefinisikan fungsi *Jacobian* sisi (Rozenheck et al., 2015)

$$R \triangleq \text{diag}(e_i^T)(E^T \otimes I_2) \in R^{m \times 3n} \quad (7)$$

Dimana $\text{diag}(A_i)$, I_2 , dan $E \in R^{n \times m}$ adalah $\text{diag}(A_i) \triangleq \text{blkdiag}\{A_1, \dots, A_n\} \in R^{np \times nq}$, matrik identitas 2×2 dan matrik *incidence* $\{0, \pm 1\}$ yang baris dan kolom matrik nya mengindikasikan simpulnya dan sisinya. Kendali formasi mengadopsi dari penelitian sebelumnya (Oh & Ahn, 2014) dimana persamaan potensial didefinisikan sebagai berikut.

$$\Phi(e) = \frac{1}{2} \sum_{i \in \mathcal{V}} \|v_i\|^2 + \frac{1}{2} \sum_{k=1}^m (\|e_k\|^2 - d_k^2)^2 \quad (8)$$

Lalu, *Gradient negative* dari persamaan (8) akan digunakan untuk masukan dari persamaan (4) dan diberikan konstanta *proportional* (Rozenheck et al., 2015). Berikut adalah persamaan model yang dijalankan secara simulasi.

$$\dot{p} = A_f p(t) + B_f \frac{\partial \Phi(e)}{\partial v} = A_f p(t) + B_f v(t) \quad (9)$$

$$\dot{v} = -C \left(\frac{\partial \Phi(e)}{\partial v} + \frac{\partial \Phi(e)}{\partial p} \right) = -k_{p1} v(t) + R^T(p(t)) k_{p2} (R(p(t))p(t) - d) \quad (10)$$

Dimana $A_f \in R^{3n \times 3n}$ dan $B_f \in R^{3n \times 3n}$ adalah matrix diagonal dari A_c dan B_c .

2.4 Algoritma Cosinus

Dinotasikan \mathcal{N}_A adalah himpunan tetangga dari robot A . Dinotasikan $B_i \in \mathcal{N}_A$ adalah robot i sebagai tetangga dari robot A . Dinotasikan $d_i[k]$ adalah informasi jarak antara robot A ke robot B_i . Langkah pertama (Gambar 2a) yang dilakukan oleh robot A adalah menyimpan jarak $d_i[k]$ dan menggunakan kendali mode dua pada persamaan (6) untuk berpindah sepanjang l_a atau ke koordinat $A' = (0, l_a)$ sehingga akan mendapatkan jarak $d_i[k+1]$. Dari perbedaan d_i sesudah dan sebelum berpindah tersebut akan digunakan untuk menentukan sudut α_i^a menggunakan rumus segitiga *cosinus* pada persamaan (11).

$$\zeta_i^a = \cos^{-1} \left(\frac{l_a^2 + d_i[k+1]^2 - d_i[k]^2}{2d_i[k+1]l_a} \right) \quad (11)$$

$$\alpha_i^a = 180^\circ \pm \zeta_i^a \quad (12)$$

Dari persamaan (12) akan didapat koordinat tetangga menggunakan vector berikut.

$$x_{B_i}^A = \begin{bmatrix} x_{B_i} = d_i[k+1] \cos \alpha_i^a \\ y_{B_i} = d_i[k+1] \sin \alpha_i^a \end{bmatrix} \quad (13)$$

Hasil koordinat yang dihasilkan pada persamaan (13) akan menghasilkan bias dikarenakan letak kuadran sudut yang dihasilkan pada persamaan (12). Sudut ζ_i^a adalah sudut segitiga (Gambar 2a) $\angle AA'B_1$ atau $\angle AA'B_2$. Oleh karena itu pada persamaan (12) terdapat operasi \pm dimana operasi tersebut akan dilakukan berdasarkan letak kuadran B_i .

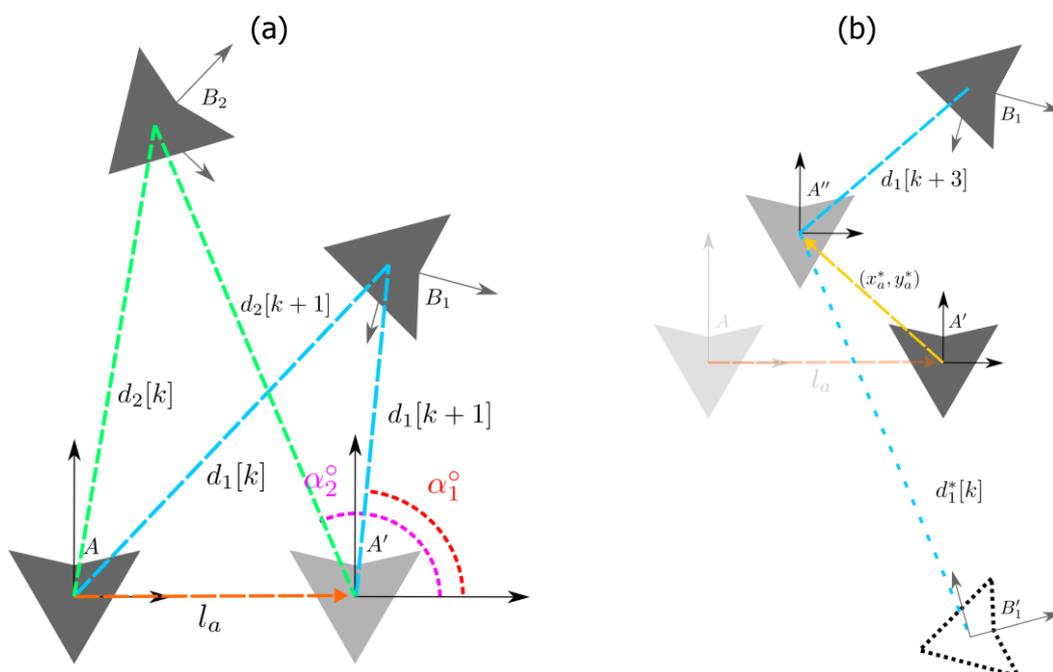
$$\alpha_i^a = \begin{cases} 180^\circ - \zeta_i^a, & \text{Kuadran 1 dan 2} \\ 180^\circ + \zeta_i^a, & \text{Kuadran 3 dan 4} \end{cases} \quad (14)$$

Maka perlu satu Langkah lagi untuk menentukan kejadian pada persamaan (14). Langkah tersebut adalah memindah robot lagi ke koordinat $A'' = (x_a^*, y_a^*)$ (Gambar 2b). Setelah dilakukan perpindahan maka akan dibandingkan jarak yang diperoleh dari sensor dengan jarak yang dihasilkan dari koordinat persamaan (13). Apabila terdapat perbedaan maka kejadian pada persamaan (14) diubah ke kejadian selanjutnya.

3. HASIL DAN PEMBAHASAN

Kendali formasi berdasarkan jarak akan dijalankan secara simulasi menggunakan persamaan (9) dan (10). Simulasi akan menggunakan 3 robot sehingga himpunan simpul $\mathcal{V} = R_1, R_2, R_3$. Menggunakan himpunan sisi $\mathcal{E} = \{(R_1, R_2), (R_3, R_2), (R_3, R_1)\}$ sebagai konfigurasi jaringan antar robot sehingga variable jarak $d_1 = \|x_1 - x_2\|$, $d_2 = \|x_3 - x_2\|$ dan $d_3 = \|x_3 - x_1\|$. Dapat diperhatikan pada **Error! Reference source not found.** adalah grafik jarak d_i terhadap waktu dan grafik pergerakan robot terhadap koordinat global robot R_1 . Gambar grafik tersebut adalah hasil dari kendali formasi yang tidak diimplementasi algoritma *cosinus* dengan asumsi nilai state persamaan (9) dan (10) ditentukan secara langsung. **Error! Reference source not found.** adalah grafik jarak d_i dan pergerakan robot terhadap koordinat global robot R_1 yang diimplementasi dengan algoritma *cosinus*. Dapat diperhatikan pada **Error! Reference source not found.**, Robot R_1 menjalankan algoritma *cosinus* langkah 1 dimana robot berpindah sepanjang $l_a = 1$ sehingga *setpoint* kendali persamaan (6) adalah $r_2^c = [1 \ 0 \ 0 \ 0 \ 0 \ 0]^T$. Setelah *setpoint* tercapai, robot R_1 mendapatkan jarak yang dibutuhkan untuk menyelesaikan persamaan (12) dan (13) lalu dilanjutkan ke langkah ke 2 dengan berpindah ke koordinat (1,1) sehingga *setpoint* kendali persamaan (6) adalah $r_2^c = [1 \ 1 \ 0 \ 0 \ 0 \ 0]^T$. Setelah *setpoint* tercapai maka dilakukan pengecekan kejadian persamaan (14) dengan membandingkan jarak dari sensor dengan jarak dari koordinat estimasi. Robot R_3 berada pada kuadran 4 maka case yang digunakan adalah $\alpha_i^\circ = 180^\circ + \zeta_i^a$ dan robot R_2 berada pada kuadran 4 maka case yang digunakan adalah $\alpha_i^\circ = 180^\circ - \zeta_i^a$. Setelah koordinat ditemukan, maka robot menggunakan kendali formasi berdasarkan jarak pada persamaan (9) dan (10).

Dilakukan percobaan *settling time* dengan beberapa koordinat yang mewakili kombinasi dua robot dan kejadian pada persamaan (14). Serta diberikan nilai K_{p1} dan K_{p2} pada persamaan (10) yang berbeda juga. Dapat diperhatikan pada Tabel 1 bahwa dari selisih penggunaan algoritma dan tidak, memiliki rata-rata 6.5 detik. Artinya bahwa algoritma membutuhkan waktu 5-7 detik untuk menemukan tetangganya. Konstanta K_p mempengaruhi



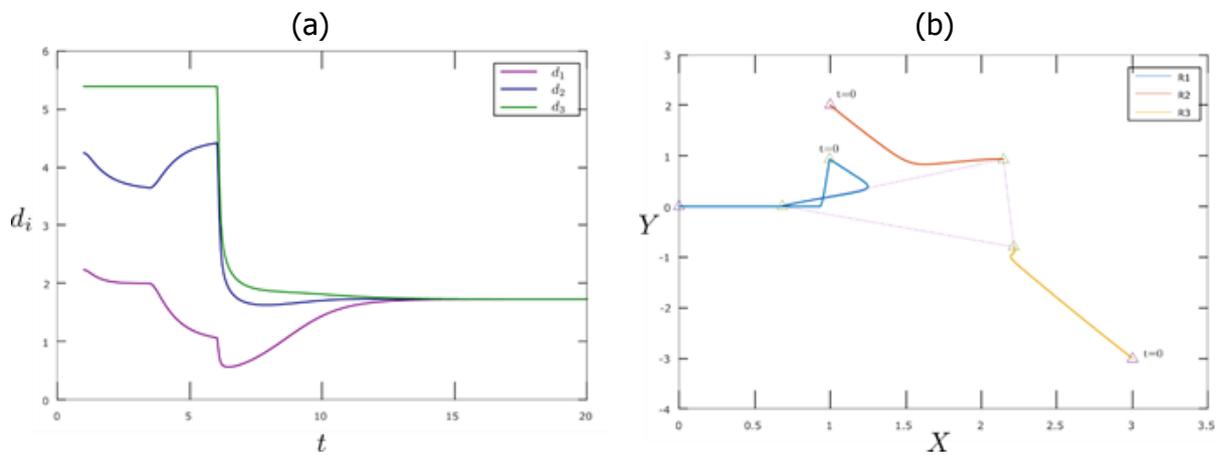
Gambar 2. Ilustrasi Strategi Algoritma *cosinus*
(a) Pencarian Koordinat Tetangga; (b) Pencarian Koordinat Tetangga,

settling time robot untuk menuju jarak yang sama dan apabila dilihat dari baris kedua mengalami saturasi pada nilai 2 detik dengan nilai K_p yang lebih besar.

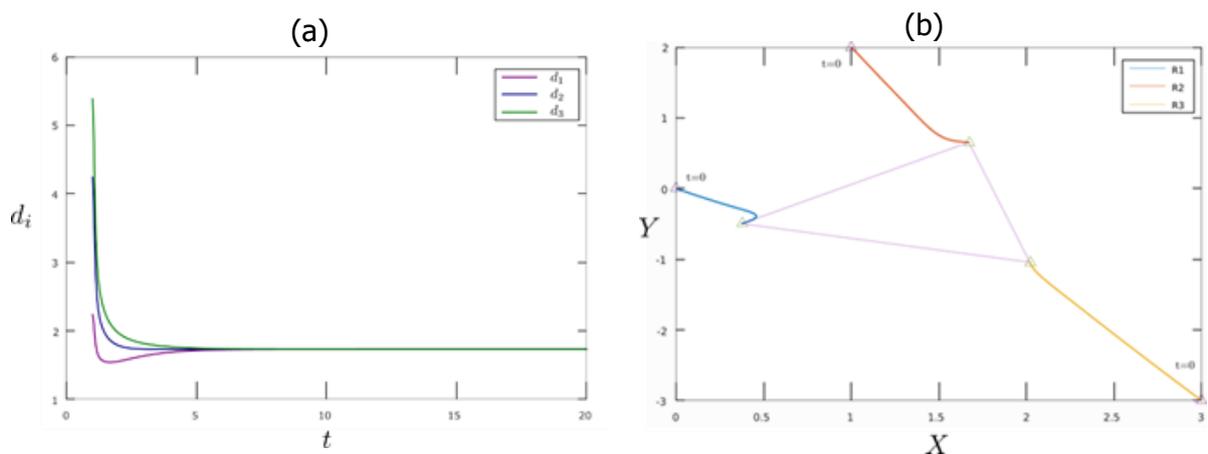
Tabel 1. *Settling Time* Dengan Konstanta K_p Yang Berbeda

Koordinat			<i>Settling Time</i> Tanpa Algoritma <i>cosinus</i> (detik)			<i>Settling Time</i> Dengan Algoritma <i>cosinus</i> (detik)			Selisih (detik)		
R1	R2	R3	A	B	C	A	B	C	A	B	C
(0, 0)	(1, 2)	(-2, 3)	6	5	3	14	11	9	8	6	6
(0, 0)	(-2, -4)	(3, -2)	4	2	2	9	8	7	5	6	5
(0, 0)	(1, 2)	(3, -3)	7	4	2	16	13	11	9	9	9
(0, 0)	(-2, -3)	(3, 2)	10	7	4	16	11	9	6	4	5

Konstanta:
 (A) $K_{p1} = 50$; $K_{p2} = 3$; (B) $K_{p1} = 80$; $K_{p2} = 7$; (C) $K_{p1} = 100$; $K_{p2} = 15$



Gambar 4. Grafik Jarak dan Pergerakan Robot Menggunakan Algoritma cosinus
 (a) Grafik jarak setiap tetangga robot, (b) Grafik jarak setiap tetangga robot.



Gambar 3. Grafik Jarak dan Pergerakan Robot Tanpa Menggunakan Algoritma cosinus
 (a) Pencarian Koordinat Tetangga, (b) Grafik jarak setiap tetangga robot.

4. KESIMPULAN

Berdasarkan hasil percobaan, rata-rata waktu yang dibutuhkan untuk algoritma cosinus menemukan koordinat tetangganya adalah 6.5 detik. Settling time minimal yang bisa dicapai oleh kendali formasi berdasarkan jarak menggunakan model robot holonomic adalah 2 detik.

DAFTAR RUJUKAN

- Cai, X., & de Queiroz, M. (2014). Multi-agent formation maneuvering and target interception with double-integrator model. *2014 American Control Conference*, 287–292. <https://doi.org/10.1109/ACC.2014.6858603>
- Correia, M. D., Gustavo, A., & Conceição, S. (2012). Modeling of a Three Wheeled Omnidirectional Robot Including Friction Models. *IFAC Proceedings Volumes*, 45(22), 7–12. <https://doi.org/https://doi.org/10.3182/20120905-3-HR-2030.00002>
- Deghat, M., Anderson, B. D. O., & Lin, Z. (2016). Combined Flocking and Distance-Based Shape Control of Multi-Agent Formations. *IEEE Transactions on Automatic Control*, 61(7), 1824–1837. <https://doi.org/10.1109/TAC.2015.2480217>
- Ferik, S. el, Nasir, M. T., & Baroudi, U. (2016). A Behavioral Adaptive Fuzzy controller of multi robots in a cluster space. *Applied Soft Computing*, 44, 117–127. <https://doi.org/https://doi.org/10.1016/j.asoc.2016.03.018>
- Guanghua, W., Deyi, L., Wenyan, G., & Peng, J. (2013). *Study on Formation Control of Multi-Robot Systems*. 1335–1339. <https://doi.org/10.1109/ISDEA.2012.316>
- Guo, K., Li, X., & Xie, L. (2020). Ultra-Wideband and Odometry-Based Cooperative Relative Localization With Application to Multi-UAV Formation Control. *IEEE Transactions on Cybernetics*, 50(6), 2590–2603. <https://doi.org/10.1109/TCYB.2019.2905570>
- Kang, S.-M., Park, M.-C., Lee, B.-H., & Ahn, H.-S. (2014). Distance-based formation control with a single moving leader. *2014 American Control Conference*, 305–310. <https://doi.org/10.1109/ACC.2014.6858587>
- Li, X., Li, B., Zhang, P., Zhang, J., & Zhang, S. (2015). Large-Scale Unmanned Aerial Vehicle Formation Maintenance Algorithm Based on Virtual Structure. *2015 8th International Symposium on Computational Intelligence and Design (ISCID)*, 1, 417–421. <https://doi.org/10.1109/ISCID.2015.5>
- Oh, K.-K., & Ahn, H.-S. (2014). Distance-based undirected formations of single-integrator and double-integrator modeled agents in n-dimensional space. *International Journal of Robust and Nonlinear Control*, 24(12), 1809–1820. <https://doi.org/10.1002/rnc.2967>

- Oh, K.-K., Park, M.-C., & Ahn, H.-S. (2015). A survey of multi-agent formation control. *Automatica*, *53*, 424–440. <https://doi.org/https://doi.org/10.1016/j.automatica.2014.10.022>
- Park, M.-C., & Ahn, H.-S. (2015). Distance-based control of formations with orientation control. *2015 54th IEEE Conference on Decision and Control (CDC)*, 2199–2204. <https://doi.org/10.1109/CDC.2015.7402533>
- Qiang, L., Heng, W., Huican, L., & Ying, Z. (2017a). Design and implementation of multi robot research platform based on UWB. *2017 29th Chinese Control And Decision Conference (CCDC)*, 7246–7251. <https://doi.org/10.1109/CCDC.2017.7978492>
- Qiang, L., Heng, W., Huican, L., & Ying, Z. (2017b). Design and implementation of multi robot research platform based on UWB. *2017 29th Chinese Control And Decision Conference (CCDC)*, 7246–7251. <https://doi.org/10.1109/CCDC.2017.7978492>
- Rozenheck, O., Zhao, S., & Zelazo, D. (2015). A proportional-integral controller for distance-based formation tracking. *2015 European Control Conference (ECC)*, 1693–1698. <https://doi.org/10.1109/ECC.2015.7330781>
- Wang, X., Yan, Z., & Wang, J. (2014). Model predictive control of multi-robot formation based on the simplified dual neural network. *2014 International Joint Conference on Neural Networks (IJCNN)*, 3161–3166. <https://doi.org/10.1109/IJCNN.2014.6889491>