

Pengaturan Formasi Menggunakan Pendekatan *Leader - Follower* pada Sistem Multi Robot

Haris Tri Rahmanto dan Achmad Jazidie

Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111

e-mail: haris.tri13@mhs.ee.its.ac.id, jazidie@ee.its.ac.id

Abstrak—Sistem multi robot dapat didefinisikan sebagai kumpulan robot yang berkoordinasi melalui sistem komunikasi untuk melakukan tugas kooperatif tertentu. Penggunaan multi robot secara terkoordinasi memungkinkan pencapaian tugas yang lebih kompleks. Efisiensi penyelesaian tugas pada sistem multi robot dapat ditingkatkan dengan menggunakan formasi. Pengaturan formasi pada sistem multi robot dapat diselesaikan dengan menggunakan pendekatan *leader-follower*. *Leader* sebagai pusat koordinasi sedangkan *follower* mengikuti *leader* dengan jarak dan orientasi relatif yang telah ditentukan. Penerapan formasi pada lingkungan tidak terstruktur terdapat rintangan yang memungkinkan formasi tidak terbentuk sesuai dengan yang diinginkan. Pendekatan *leader-follower* memiliki kelemahan jika *leader* menghadapi rintangan maka keseluruhan robot dapat terganggu. Berdasarkan kelemahan tersebut, *leader* harus dapat menghindari rintangan. Penelitian ini menggunakan kontroler *tracking* dan kontroler menghindari rintangan dengan algoritma *fuzzy*. Penerapan kontroler *tracking* pada sistem multi robot bertujuan membentuk formasi. Sistem dengan kontroler yang diusulkan tersebut mampu membentuk formasi sesuai spesifikasi yang diberikan sekaligus menghindari rintangan.

Kata Kunci—Pengaturan formasi, *Leader-follower*, *Mobile robot*, Logika *fuzzy*

I. PENDAHULUAN

SISTEM multi robot dapat didefinisikan sebagai kumpulan robot yang berkoordinasi melalui sistem komunikasi untuk melakukan tugas kooperatif tertentu. Penggunaan multi robot secara terkoordinasi memungkinkan pencapaian tugas yang lebih kompleks. Setiap robot tidak harus memiliki kemampuan sangat lengkap, karena masing-masing robot dapat memfokuskan untuk menyelesaikan tugas tertentu [1].

Pengaturan formasi merupakan bagian yang penting ketika bekerja dengan sistem yang melibatkan multi robot. Sistem multi robot bergerak dalam formasi memungkinkan satu pengguna untuk mengatur seluruh kelompok tanpa menentukan secara langsung perintah untuk masing-masing robot. Dalam pengembangan pengaturan formasi, diperkenalkan pendekatan *leader-follower* dimana terdapat dua peran *leader* dan *follower*. Robot *leader* menentukan jalur yang akan dilalui. Robot *follower* mengikuti robot *leader* dengan menjaga bentuk formasi yang diinginkan. Telah dirancang algoritma mempertahankan bentuk formasi ketika menghadapi rintangan. Algoritma tersebut menggunakan dua kontroler yaitu, menghindari rintangan dan *tracking* [2].

Penggunaan pendekatan *leader-follower* dengan konsep umpan balik visual yang diperkenalkan oleh Das et al. Kontroler yang digunakan ada dua, kontroler *separation-bearing* dan kontroler *separation-separation*. Pada kontroler

separation-bearing robot *follower* mengikuti satu robot *leader* dengan jarak dan orientasi relatif tertentu. Di sisi lain, pada kontroler *separation-separation* robot *follower* mengikuti dua robot *leader* dengan jarak dan orientasi relatif tertentu [3].

Pendekatan *leader-follower* menggunakan kontroler PID telah dikerjakan oleh In-Sung Choi. Pada penelitiannya robot dilengkapi dengan sensor. Akan tetapi, informasi yang diperoleh dari sensor terganggu oleh *noise*. Oleh karena itu, formasi tidak dapat terbentuk sesuai dengan yang diinginkan. Dengan menggunakan kontroler PID kesalahan akibat *noise* dapat terselesaikan. Telah dibuktikan dengan menggunakan kontroler PID kesalahan posisi robot berkurang [4].

Kelemahan dalam penerapan *leader-follower* yaitu jika terjadi gangguan terhadap robot *leader* maka keseluruhan robot akan menerima dampak dari gangguan tersebut. Pergerakan dari robot *leader* tidak bergantung dengan *follower*. Jika robot *leader* mengalami gangguan maka *follower* akan terganggu. Tidak ada sistem umpan balik antara *follower* dengan *leader*. Oleh karena itu, *leader* tidak dapat dibiarkan mengalami gangguan [3], [5], [6]. Gangguan rintangan dapat terselesaikan dengan perubahan peran *leader* dan *follower*. Ketika *leader* mengalami gangguan, perannya akan berganti sementara menjadi *follower* dan *follower* menjadi *leader* [7]. Di sisi lain, keunggulan dalam penerapan *leader-follower* dapat mengurangi permasalahan *tracking* dimana *leader* mengikuti tujuan kelompok, sedangkan *follower* mengikuti jejak *leader* [6].

Selain pendekatan *leader follower*, diperkenalkan juga pendekatan *virtual structure* yang mengasumsikan titik *virtual* sebagai *virtual leader*. Pada pendekatan ini, informasi dari masing-masing robot yang berupa posisi dan orientasi akan diolah terus-menerus untuk mengetahui performa titik *virtual*. Implementasi robot dalam jumlah besar membutuhkan *bandwidth* yang besar [1].

Pertandingan robot sepak bola merupakan salah satu aplikasi sistem multi robot. Setiap robot sepak bola membutuhkan kemampuan berkoordinasi untuk menyelesaikan tugasnya. Dengan menggunakan formasi robot dapat meningkatkan peluang untuk menang dan mengurangi gangguan yang mungkin terjadi antar robot [8].

Lingkungan yang tidak terstruktur dimana terdapat rintangan statis maupun dinamis. Oleh karena itu robot harus memiliki kemampuan mempertahankan bentuk formasi. Pada kondisi tersebut, pada penelitian ini dilakukan perancangan kontroler *tracking* dan menghindari rintangan untuk mengatur formasi robot agar saat menghadapi rintangan robot dapat menghindari rintangan dan membentuk formasi sesuai yang diinginkan.

Pembahasan dalam makalah ini meliputi beberapa bagian. Bagian II membahas mengenai model kinematik robot.

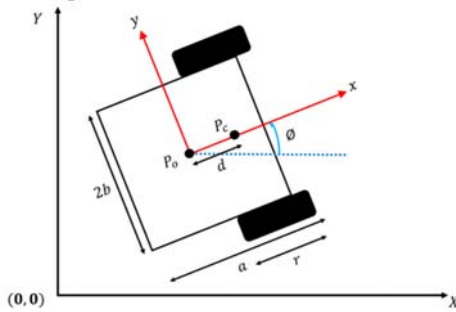
Bagian III membahas proses perancangan kontroler *tracking* dan kontroler menghindari rintangan. Pada bagian IV dibahas hasil pengujian sistem dan analisisnya. Bagian V memaparkan kesimpulan yang diperoleh dari penelitian yang telah dilakukan.

II. MODEL KINEMATIK

A. Model Kinematik Robot

Kinematik dalam robotika adalah suatu bentuk pernyataan geometris yang berisi deskripsi matematik dari suatu struktur robot. Model kinematik dapat diperoleh hubungan antara konsep koordinat sendi dengan koordinat kartesian [9].

Penelitian ini menggunakan *mobile robot* yang memiliki dua roda kanan-kiri dengan kemudi terpisah (*differential driven mobile robot*, disingkat DDMR), seperti diilustrasikan pada Gambar 1.



Gambar 1. DDMR pada medan 2D Cartesian

Berdasarkan pada Gambar 1 lebar *mobile robot* adalah $2b$ dan r adalah jari-jari roda. Koordinat global dinotasikan dengan (X, Y) , dan koordinat lokal yang terpusat pada robot dinotasikan (x, y) . Titik pusat koordinat lokal berada pada titik pusat masa robot (P_o) dengan koordinat (x, y) . Titik pusat massa robot adalah P_c . Jarak antara P_o dengan P_c adalah d . Model kinematik robot diperoleh sesuai dengan pada [10]

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos \phi & 0 \\ \sin \phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

Dari Persamaan (1) diasumsikan bahwa ω_L dan ω_R menyatakan kecepatan angular roda kiri dan roda kanan. Hubungan antara kecepatan angular dan kecepatan linier robot dapat diketahui dengan Persamaan (2) dan (3).

$$\begin{aligned} v &= \frac{v_R + v_L}{2} \\ &= \frac{\omega_R r_R + \omega_L r_L}{2} \\ &= \frac{\omega_R + \omega_L}{2} r \end{aligned} \quad (2)$$

Dan

$$\begin{aligned} \omega &= \frac{v_R - v_L}{b} \\ &= \frac{\omega_R r_R - \omega_L r_L}{b} \end{aligned}$$

$$= \frac{\omega_R - \omega_L}{b} r \quad (3)$$

Kecepatan angular roda robot dapat diperoleh dengan Persamaan (4) dan (5)

$$\omega_L = \frac{v - \left(\frac{b}{2}\right)\omega}{r} \quad (4)$$

$$\omega_R = \frac{v + \left(\frac{b}{2}\right)\omega}{r} \quad (5)$$

Berdasarkan Persamaan (2) dan (3), substitusi Persamaan (4) dan (5) ke Persamaan (1) diperoleh

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \frac{r_R}{2} \cos \phi & \frac{r_L}{2} \cos \phi \\ \frac{r_R}{2} \sin \phi & \frac{r_L}{2} \sin \phi \\ -\frac{r_R}{b} & \frac{r_L}{b} \end{bmatrix} \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix} \quad (6)$$

B. Model Kinematik Pendekatan Leader-Follower

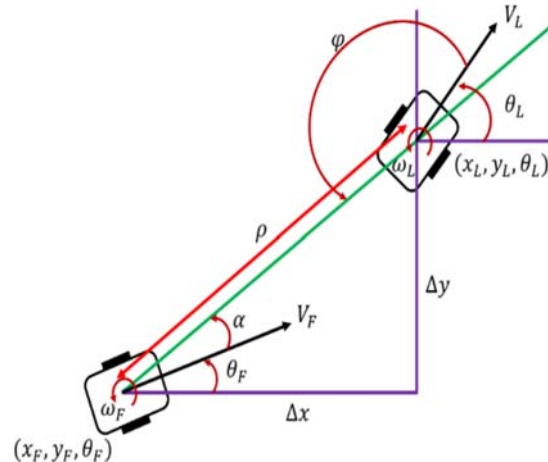
Leader-follower pada Gambar 2 menggunakan DDMR. Pada koordinat kartesian, *leader* memiliki koordinat (x_L, y_L, θ_L) dan *follower* (x_F, y_F, θ_F) di belakang *leader*.

Leader dan *follower* terpisah dengan jarak ρ , yang didefinisikan garis lurus antara titik pusat robot *leader* dengan *follower*. Nilai ρ dapat diperoleh dengan Persamaan (7)

$$\rho = \sqrt{(x_L - x_F)^2 + (y_L - y_F)^2} \quad (7)$$

Dan, sudut antara pergerakan *follower* dengan robot *leader* didefinisikan dengan α yang dapat diperoleh dengan Persamaan (8).

$$\alpha = \tan^{-1} \left(\frac{y_L - y_F}{x_L - x_F} \right) - \theta_F \quad (8)$$



Gambar 2. Diagram *leader-follower*

Sistem memerlukan variabel sudut yang dapat menghubungkan orientasi *leader* dengan pergerakan *follower* dan garis ρ , yaitu

$$\phi = \alpha + \theta_F - \theta_L + \pi \quad (9)$$

Model kinematik

$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \alpha & 0 \\ \sin \alpha & -1 \\ \rho & 0 \\ \sin \alpha & 0 \\ \rho & 0 \end{bmatrix} \begin{bmatrix} v_F \\ \omega_F \end{bmatrix} + \begin{bmatrix} -\cos \varphi & 0 \\ \sin \varphi & 0 \\ \rho & 0 \\ \sin \varphi & -1 \\ \rho & -1 \end{bmatrix} \begin{bmatrix} v_L \\ \omega_L \end{bmatrix} \quad (10)$$

ρ, α dan φ merepresentasikan model kinematik untuk merancang sistem pengaturan robot.

Tujuan akhir dari kontroler adalah untuk memperoleh kecepatan linier (v_F) dan kecepatan sudut (ω_F) follower sehingga dapat mengikuti leader dan menjaga jarak dan orientasi relatif yang konstan. Robot *follower tracking* robot leader memerlukan dua variabel, jarak leader dengan follower ρ dan α . Apabila ρ_d dinotasikan sebagai jarak yang diinginkan antara kedua robot dan α_d sudut antara pergerakan follower dengan ρ yang diinginkan. Spesifikasi yang diinginkan adalah kesalahan sistem mendekati 0, sehingga sinyal kontrol dapat diperoleh dengan Persamaan (11).

$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} k & 0 \\ 0 & k \end{bmatrix} \begin{bmatrix} \rho_d - \rho \\ \alpha_d - \alpha \end{bmatrix} \quad (11)$$

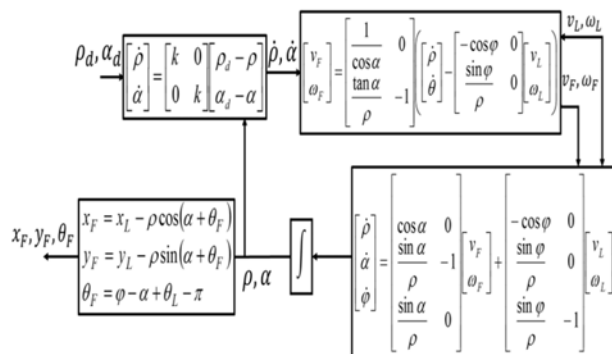
k adalah parameter penguat positif yang dapat membuat sistem stabil. Tahap selanjutnya adalah memperoleh nilai v_F dan ω_F follower dengan menggunakan Persamaan (11). Substitusi Persamaan (11) ke Persamaan (10) dan menghilangkan bagian persamaan $\dot{\varphi}$ akan diperoleh v_F dan ω_F sesuai dengan yang ada pada [12].

$$\begin{bmatrix} v_F \\ \omega_F \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \cos \alpha & \tan \alpha \\ \rho & -1 \end{bmatrix} \left(\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \end{bmatrix} - \begin{bmatrix} -\cos \varphi & 0 \\ \sin \varphi & 0 \\ \rho & 0 \end{bmatrix} \begin{bmatrix} v_L \\ \omega_L \end{bmatrix} \right) \quad (12)$$

Tahap terakhir adalah konversi variabel (ρ, α, φ) ke dalam koordinat kartesian (x, y, θ) . Nilai (x_F, y_F, θ_F) dapat diperoleh dengan

$$\begin{aligned} x_F &= x_L - \rho \cos(\alpha + \theta_F) \\ y_F &= y_L - \rho \sin(\alpha + \theta_F) \\ \theta_F &= \varphi - \alpha + \theta_L - \pi \end{aligned} \quad (13)$$

Gambar menunjukkan diagram blok pendekatan *leader-follower* dengan satu follower sesuai dengan [11].



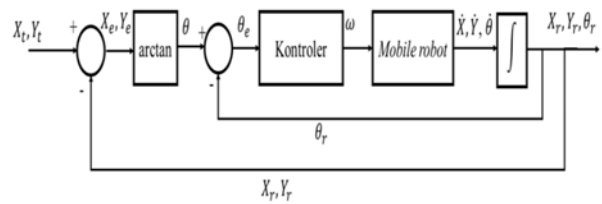
Gambar 3. Diagram blok pendekatan *leader-follower*

III. PERANCANGAN KONTROLER

Pada subbab ini, dirancang dua kontroler yaitu kontroler *tracking* dan kontroler menghindari rintangan. Kontroler *tracking* menggunakan kontroler proporsional (P) dimana parameter penguat proporsional (K_p) ditentukan menggunakan logika *fuzzy*. Dirancangan dua kontroler *tracking* yang memiliki dua tujuan yaitu, kontroler *tracking* GTT digunakan pada robot leader untuk menentukan trayektori menuju *target*. Kedua, kontroler *tracking leader* digunakan pada pendekatan *leader-follower* untuk robot follower.

A. Kontroler Tracking GTT

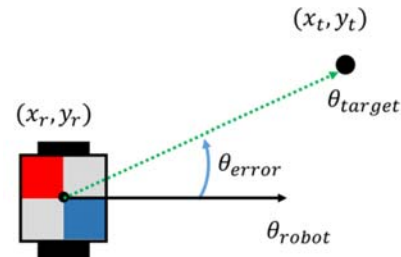
Kontroler ini bertujuan agar robot dapat menghampiri posisi *target* yang telah ditentukan. Blok diagram algoritma GTT dapat dilihat pada Gambar 4. Ilustrasi algoritma GTT pada robot sebenarnya dapat dilihat pada Gambar 5.



Gambar 4. Blok diagram algoritma GTT

Posisi *target* dapat tercapai dengan mengatur orientasi robot (θ_{robot}) mengarah ke *target* sesuai dengan yang diilustrasikan Gambar 5. Dengan diketahui posisi robot (x_r, y_r) , kesalahan sudut antara posisi robot dengan posisi *target* (x_t, y_t) dapat diperoleh dengan

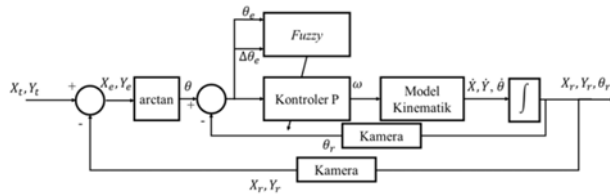
$$\theta_{error} = \theta_{target} - \theta_{robot} = \arctan\left(\frac{y_t - y_r}{x_t - x_r}\right) - \theta_{robot} \quad (14)$$



Gambar 5. Ilustrasi GTT

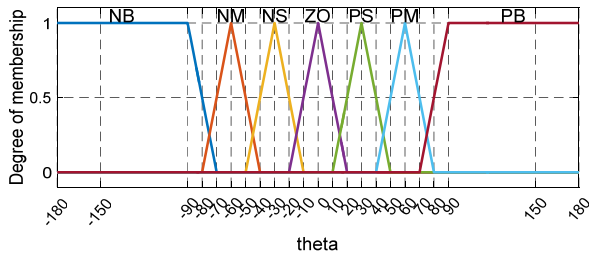
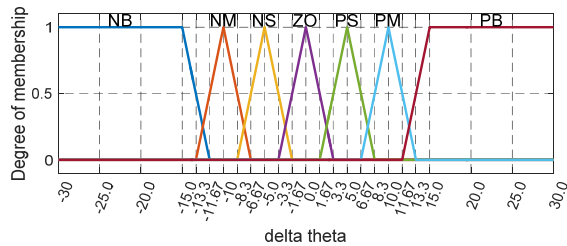
Kontroler *tracking* menggunakan kontroler P. Penentuan parameter kontroler P dapat diperoleh secara analitis atau menggunakan aturan Ziegler-Nichols, Cohen-Coon, logika *fuzzy* dll. Pada penelitian ini parameter kontroler P diperoleh menggunakan logika *fuzzy*.

Variabel masukan yang digunakan pada algoritma ini adalah selisih orientasi *target* (θ_e) dan perubahan selisih orientasi ($\Delta\theta_e$) antara robot dengan sudut *target*. Fungsi keanggotaan (*membership function*) dari masukan selisih orientasi *target* dan perubahan selisih orientasi dibagi menjadi 7 himpunan yaitu, NB (*negative big*), NM (*negative middle*), NS (*negative small*), ZO (*zero*), PS (*positive small*), PM (*positive middle*), dan PB (*positive big*). Setting algoritma penalaan parameter K_p dengan menggunakan *fuzzy* adalah seperti pada Gambar 6.

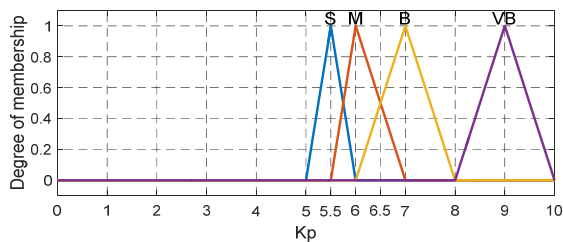


Gambar 6. Diagram blok penalaan parameter Kp dengan logika fuzzy

Fungsi keanggotaan dari θ_e dan $\Delta\theta_e$ memiliki rentang nilai $\{-180,180\}$ dan $\{-30,30\}$ yang dapat dilihat pada Gambar 7 dan Gambar 8.

Gambar 7. Fungsi keanggotaan masukan θ_e Gambar 8. Fungsi keanggotaan masukan $\Delta\theta_e$

Sebagai keluaran sistem fuzzy yaitu berupa parameter Kp yang ditunjukkan pada Gambar 9. Fungsi keanggotaan dari keluaran parameter Kp dibagi menjadi 5 himpunan yaitu, S (small), M (medium), B (Big) dan VB (Very Big).



Gambar 9. Fungsi keanggotaan keluaran Kp

Dari kedua masukan terdapat 49 rule yang direpresentasikan pada Tabel 1 sesuai dengan makalah [11].

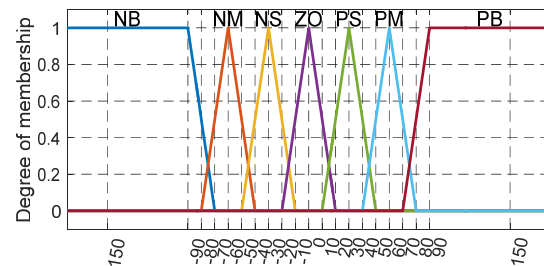
Tabel 1. Rule base penalaan parameter Kp untuk kontroler tracking GTT

		$\Delta\theta_e$						
		NB	NM	NS	ZO	PS	PM	PB
θ_e	NB	VB	VB	B	B	M	S	S
	NM	VB	VB	B	M	M	S	M
	NS	VB	VB	M	M	S	M	M
	ZO	B	B	M	S	M	M	B
	PS	M	M	S	M	M	B	B
	PM	M	S	M	B	B	B	VB
	PB	S	S	B	B	B	VB	VB

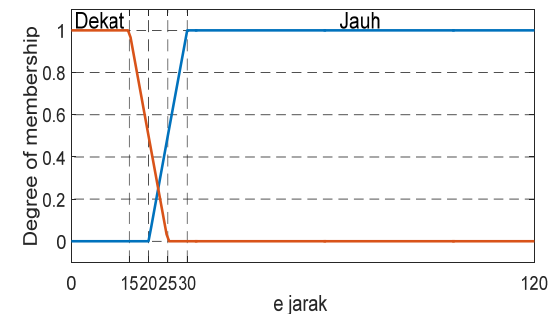
Dari Tabel diperoleh if e bernilai NB dan Δe bernilai NB, then Kp bernilai PB, dan seterusnya.

B. Kontroler Tracking Leader

Variabel masukan yang digunakan pada algoritma ini adalah kesalahan sudut dan kesalahan jarak antara *follower* dengan *leader*. Fungsi keanggotaan (*membership function*) dari masukan kesalahan sudut dibagi menjadi 7 himpunan yaitu, NB (*negative big*), NM (*negative middle*), NS (*negative small*), ZO (*zero*), PS (*positive small*), PM (*positive middle*), dan PB (*positive big*). Fungsi keanggotaan kesalahan sudut dibagi menjadi himpunan Jauh dan Dekat. Pada Gambar 10 dan Gambar 11 dapat dilihat fungsi keanggotaan masukan kesalahan sudut dan kesalahan jarak yang memiliki rentang nilai $\{-180,180\}$ dan $\{0,120\}$.

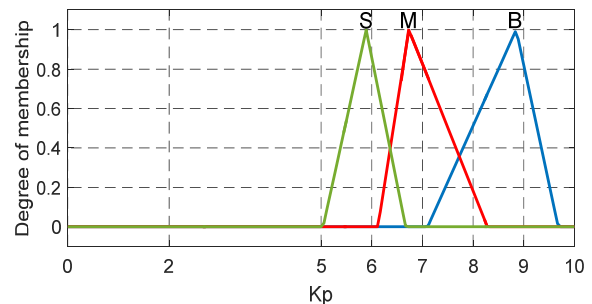


Gambar 10. Fungsi keanggotaan masukan kesalahan sudut

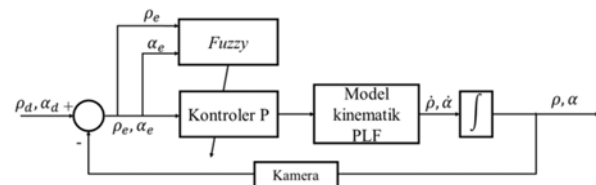


Gambar 11. Fungsi keanggotaan masukan kesalahan jarak

Keluaran logika fuzzy yaitu berupa parameter Kp yang ditunjukkan pada Gambar 12. Fungsi keanggotaan keluaran parameter Kp dibagi menjadi 3 himpunan yaitu, S (small), dan B (Big) Setting algoritma penalaan parameter Kp dengan menggunakan fuzzy adalah seperti pada Gambar 13.



Gambar 12. Fungsi keanggotaan keluaran Kp



Gambar 13. Diagram blok penalaan parameter Kp dengan logika fuzzy

Dari kedua masukan terdapat 14 rule yang direpresentasikan pada Tabel 2.

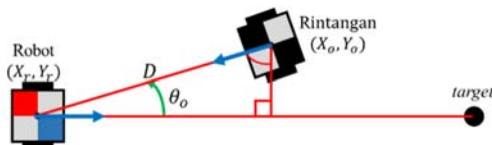
Tabel 2. Rule base penalaan parameter Kp untuk kontroler tracking

		e sudut						
		NB	NM	NS	ZO	PS	PM	PB
Jarak	Jauh	B	B	M	S	M	B	B
	Dekat	B	B	B	M	B	B	B

C. Kontroler Menghindari Rintangan

Perancangan kontroler menghindari rintangan menggunakan logika fuzzy yang ditanam ke seluruh robot. Apabila robot *leader* mendeteksi rintangan, robot akan menghindarinya kemudian membentuk kembali formasi yang sudah ditentukan. Begitu pula robot dengan *follower*.

Pada lingkungan tidak terstruktur terdapat rintangan yang memungkinkan mengganggu *leader*. Oleh karena itu, dengan menggunakan algoritma fuzzy untuk memperkirakan kecepatan angular sehingga rintangan dapat dihindari. Ilustrasi posisi robot, rintangan dan titik tujuan dapat dilihat pada Gambar 14.



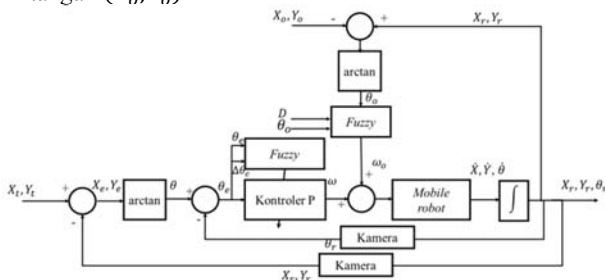
Gambar 14. Ilustrasi posisi robot, rintangan, dan tujuan

Jarak antara robot dengan rintangan dapat diperoleh berdasarkan koordinat masing-masing. Nilai jarak dan posisi rintangan dapat diperoleh dengan menggunakan Persamaan (15) dan Persamaan (16).

$$D = \sqrt{(X_o - X_r)^2 + (Y_o - Y_r)^2} \quad (15)$$

$$\theta_o = \tan^{-1} \left(\frac{Y_o - Y_r}{X_o - X_r} \right) - \theta_{robot} \quad (16)$$

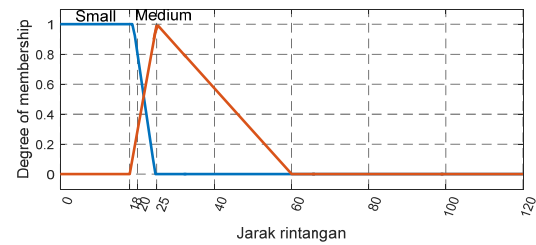
D adalah jarak antara posisi robot (X,Y) dengan posisi rintangan (X_o,Y_o).



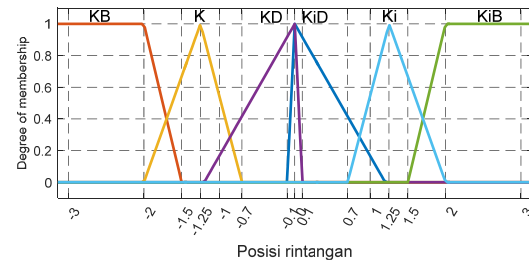
Gambar 15. Diagram blok kontroler menghindari rintangan

Pada penelitian ini, jarak terjauh yang diperkirakan adalah 60 cm. Fungsi keanggotaan untuk masukan jarak rintangan adalah *small*, dan *medium* yang dapat dilihat pada Gambar 16. Posisi rintangan direpresentasikan bernilai antara π sampai dengan $-\pi$ dengan fungsi keanggotaan kanan besar (KB), kanan (K), kanan dekat (KD), kiri dekat (KiD), kiri (Ki) dan kiri besar (KiB) yang dapat dilihat pada Gambar 17. Omega adalah fungsi keanggotaan keluaran logika fuzzy, yang merepresentasikan seberapa besar sudut untuk merubah arah gerak robot untuk menghindari rintangan.

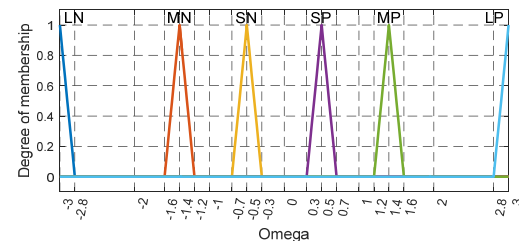
Fungsi keanggotaan arah robot dinotasikan dengan *large negatif* (LN), *medium negatif* (MN), *small negatif* (SN), *small positif* (SP), *medium positif* (MP) dan *large positif* (LP) yang dapat dilihat pada Gambar 18.



Gambar 16. Fungsi keanggotaan masukan jarak



Gambar 17. Fungsi keanggotaan masukan orientasi robot



Gambar 18. Fungsi keanggotaan keluaran arah robot

Dari kedua masukan terdapat 12 rule yang direpresentasikan pada Tabel 3.

Tabel 3. Rule base menghindari rintangan

		Posisi rintangan					
		KB	K	KD	KiD	Ki	KiB
Jarak	Small	SP	MP	LP	LN	MN	SN
	Medium	SP	SP	MP	MN	SN	SN

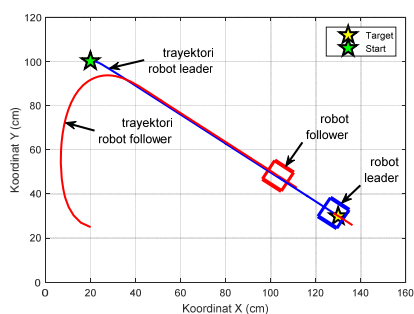
Pada Tabel menunjukkan *if* jarak rintangan *small* and posisi rintangan KB *than* omega SP dan seterusnya.

IV. PENGUJIAN DAN ANALISIS

Pengujian dilakukan dengan memberi *target* pada koordinat (130,30) dengan posisi *start* robot *leader* (20,100) dan arah hadap awal robot *leader* 0°. Keadaan awal robot *follower* diberikan dan posisi awal (25,25) dan arah hadap awal 180°. Pengujian ini bertujuan untuk mengetahui kemampuan robot dalam mempertahankan formasi sekaligus menghindari rintangan.

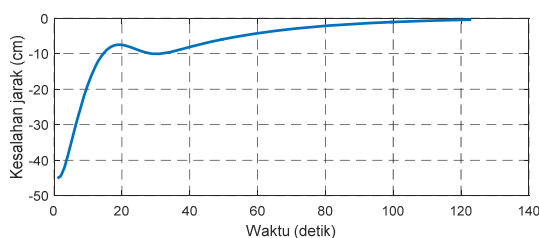
A. Simulasi Leader-Follower

Pengujian dilakukan dengan memberi spesifikasi formasi $\rho_d = 30 \text{ cm}$ dan $\alpha_d = 0 \text{ rad}$ yang membentuk formasi berbanjar. Setting pengujian sistem dengan kontroler P-fuzzy adalah seperti pada Gambar 13. Trayektori robot dapat dilihat pada Gambar 19.

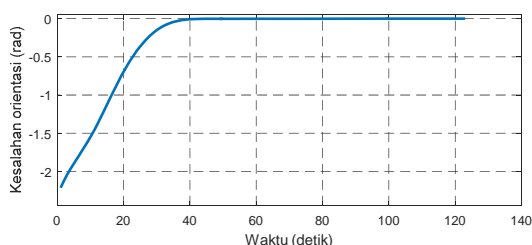


Gambar 19. Trayektori hasil pengujian GTT-PLF dengan Rintangan

Hasil yang diperoleh formasi dapat dibentuk dengan kesalahan menuju nol sesuai pada Gambar 20 dan Gambar 21.



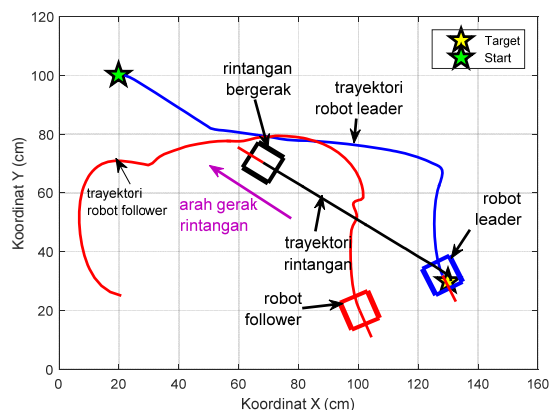
Gambar 20. Kesalahan jarak antar robot



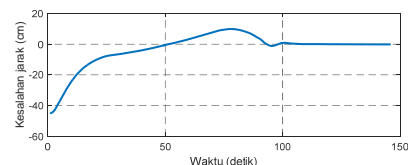
Gambar 21. Kesalahan orientasi relatif antar robot

B. Simulasi Leader-Follower dengan Rintangan Bergerak

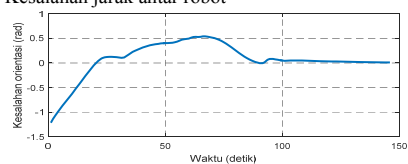
Pada pengujian kali ini kontroler menghindari rintangan dikombinasikan dengan kontroler *tracking* untuk masing-masing robot dengan harapan, formasi dapat dipertahankan meskipun terdapat rintangan yang dihadapi. Pengujian dilakukan dengan memberi spesifikasi formasi $\rho_d = 30$ cm dan $\alpha_d = \pi/3$ rad dan rintangan bergerak berpasangan dengan robot *leader*. *Setting* pengujian dengan kontroler P-fuzzy dan kontroler menghindari rintangan seperti pada Gambar 15. Hasil pengujian menunjukkan formasi dapat dipertahankan yang dapat dilihat pada Gambar 20.



Gambar 22. Trayektori hasil pengujian GTT-PLF dengan Rintangan



Gambar 23. Kesalahan jarak antar robot



Gambar 22. Kesalahan orientasi relatif antar robot

V. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dapat disimpulkan bahwa sistem pengaturan formasi lingkungan tidak terstruktur, dimana berubah sewaktu-waktu tanpa dapat diprediksi. Kondisi tersebut dapat diatasi dengan pendekatan *leader-follower*. Kontroler *tracking* dan kontroler menghindari rintangan yang diterapkan pada penelitian ini dapat bekerja sesuai tujuan perancangan yaitu pada saat terdapat rintangan, robot dapat mempertahankan bentuk formasi. Proses menghindari rintangan disini ditunjukkan dengan robot yang bergerak menjauhi rintangan hingga kemudian kembali menuju ke *target* yang telah ditentukan.

DAFTAR PUSTAKA

- [1] Y. Q. Chen and Z. Wang, "Formation Control: A Review and A New Consideration," in *Proceedings of International Conference on Intelligent Robots and Systems*, 2005.
- [2] J. Shao, G. Xie, J. Yu and L. Wang, "Leader-following Formation Control of Multiple Mobile Robots," in *IEEE International Symposium on Intelligent Control*, Limassol, 2005.
- [3] A. K. Das, R. F. V. K. J. P. Ostrowski, J. Spletzer and C. J. Taylor, "A Vision-Based Formation Control Framework," in *IEEE Transactions on Robotics and Automation*, 2002.
- [4] I.-S. Choi and J.-S. Choi, "Leader-Follower Formation Control Using PID Controller," in *ICIRA 2012*, 2012.
- [5] D. Neculescu, E. Pruner, J. Sasiadek and B. Kim, "Control of Nonholonomic Autonomous Vehicles and Their Formations," in *IFAC/IEEE 15th Int Conf. Methods and Models in Automation and Robotics MMAR*, Miedzydroje, 2010.
- [6] K. Kanjanawanishkul, Formation Control of Mobile Robots: Survey, eng.ubu.ac.th, 2005, pp. 50-64.
- [7] R. M. K. Chetty, M. Singaperumal and T. Nagarajan, "Distributed Formation Planning and Navigation Framework for Wheeled Mobile Robots," *Journal of Applied Sciences*, vol. 11, no. 9, pp. 1501-1509, 2011.
- [8] T.-Y. Tsou, C.-H. Liu and Y.-T. Wang, "Term Formation Control of Soccer Robot Systems," in *International Conference on Networking, Sensing & Control*, Taipei, Taiwan, 2004.
- [9] E. Pitowarno, Robotika, Desain, Kontrol, dan Kecerdasan Buatan, Yogyakarta: Andi Offset, 2006.
- [10] T. Fukao, H. Nakagawa and N. Adachi, "Adaptive Tracking Control of a Nonholonomic Mobile Robot," in *IEEE Transactions on Robotics and Automation*, 2000.
- [11] E. Pruner, D. Neculescu, J. Sasiadek and B. Kim, "Control of Decentralized Geometric Formations of Mobile Robots," in *IFAC/IEEE 15th Int Conference on Methods and Models in Automation and Robotics MMAR*, Miedzydroje, 2010.
- [12] A. Ćosić, M. Šušić, S. Graovac and D. Katić, "An Algorithm for Formation Control of Mobile Robots," *Serbian Journal of Electrical Engineering*, vol. 10, no. 1, pp. 59-72, February 2013.