



## Performansi Navigasi Robot *Leader-Follower* menggunakan Algoritma Logika Fuzzy Interval Tipe 2

Gita Fadila Fitriana<sup>1</sup>, Rifki Adhitama<sup>2</sup><sup>1,2</sup>Prodi Rekayasa Perangkat Lunak, Fakultas Teknologi Industri dan Informatika, Institut Teknologi Telkom Purwokerto  
<sup>1</sup>[gita@ittelkom-pwt.ac.id](mailto:gita@ittelkom-pwt.ac.id), <sup>2</sup>[rifki@ittelkom-pwt.ac.id](mailto:rifki@ittelkom-pwt.ac.id)

### Abstract

A leader-follower robot is used to perform different tasks without continuous human assistance. The movement of robot leader-follower to environment who do not structure, avoid persecution and achieving goals is very difficult. Related to the problem, the robot leader-follower requires navigating robots independently using Interval Fuzzy Logic Type-2 (IFLT) 2 Algorithm. The IFLT 2 algorithm performance is successfully applied to this leader-follower robot, with 8 base rules less than the Fuzzy Logic Type 1 Algorithm. This simulation, the robot successfully moves to avoid obstacles and go hand in hand with the position of the follower robot always following the position of the robot leader.

Keywords: leader follower robot, interval fuzzy logic type-2 algorithm, navigating robots, fuzzy logic type 1 algorithm

### Abstrak

Robot *leader follower* digunakan untuk melakukan tugas yang berbeda tanpa bimbingan manusia secara terus menerus. Pergerakan robot *leader follower* pada lingkungan yang tidak terstruktur, menghindari halangan dan mencapai tujuan merupakan hal yang sangat sulit. Berkaitan dengan masalah tersebut, robot *leader follower* memerlukan navigasi robot secara mandiri menggunakan Algoritma Logika Fuzzy Tipe 2 (LFIT) 2. Performansi Algoritma LFIT2 berhasil diterapkan pada robot *leader follower* ini, dengan 8 basis aturan yang lebih sedikit dari algoritma Logika Fuzzy Tipe 1. Simulasi ini, robot berhasil bergerak dengan menghindari halangan serta berjalan secara beriringan dengan posisi robot *follower* selalu mengikuti posisi robot *leader*.

Kata kunci: robot *leader follower*, algoritma logika fuzzy interval tipe 2, navigasi robot, algoritma logika fuzzy tipe 1

© 2019 Jurnal RESTI

### 1. Pendahuluan

Navigasi robot secara mandiri untuk memandu pergerakan sebuah robot di lingkungan yang tidak terstruktur merupakan tugas yang sulit dan menantang [1][2][3]. Navigasi robot dapat bergerak melalui lingkungan untuk mencapai tujuan [4]. Agar robot tidak terganggu atau bertabrakan dengan objek lain, maka diperlukan navigasi dalam menghadapi masalah-masalah ketidakpastian, yaitu ketidakpastian sensor, aktuator, dan lingkungan [5].

Robot yang bergerak pada lingkungan yang tidak pasti akan memberikan informasi ke sensor dan mengarah ke berbagai perilaku robot. Beberapa pendekatan untuk kontrol dan strategi pada formasi robot *swarm* dapat dirujuk pada literatur, seperti pendekatan berbasis perilaku [6][7][8], pendekatan struktur virtual

[9][10][11], pendekatan *leader follower* [12][3]. Robot *leader follower* merupakan konsep yang mengikuti pemimpin (*leader*) didasarkan pada kebutuhan suatu kelompok untuk mencapai keteraturan. Hal terpenting dalam konsep *leader follower* adalah mengontrol formasi dari robot, sehingga robot dapat mencapai bentuk formasi yang diinginkan [13].

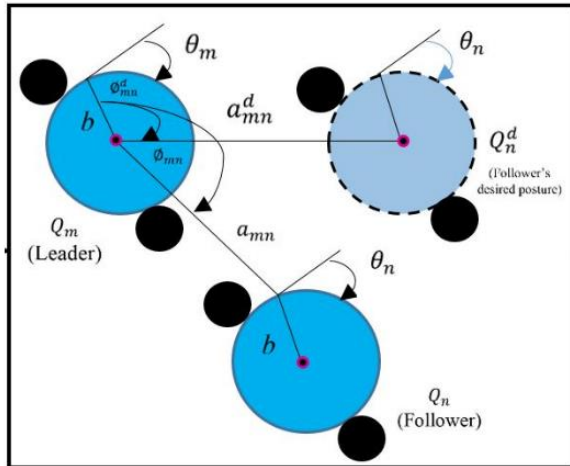
Penelitian yang dilakukan oleh [4], navigasi yang digunakan algoritma logika fuzzy. Penelitian tersebut menjelaskan dengan 16 aturan robot dapat bergerak secara bebas tanpa tujuan. Pada penelitian ini, robot bergerak dengan perilaku *leader follower* bergerak dalam menghindari halangan dan mencapai target. Algoritma Logika Fuzzy Interval Tipe (LFIT) 2 menggunakan 8 aturan yang dapat menyelesaikan

masalah menghindari halangan dan bergerak menuju target.

## 2. Metode Penelitian

### 2.1. Robot Leader Follower

*Leader follower* merupakan salah satu perilaku dari robot *swarm*. *Leader follower* merupakan robot yang berperan sebagai robot pemimpin (*leader*) dan sebagian robot pengikut (*follower*). Gerak robot *follower* mengikuti trajektori robot *leader* dengan jarak yang ditentukan dalam kecepatan yang stabil serta percepatan yang dinamis [14].



Gambar 1. Robot Leader-Follower [15]

Bentuk persamaan kinematik robot *leader follower* ditunjukkan pada Persamaan 1 sebagai berikut :

$$\dot{Q}_m = \begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos \theta_m & -b \sin \theta_m \\ \sin \theta_m & b \cos \theta_m \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_m \\ \omega_m \end{bmatrix} \quad (1)$$

Dimana  $Q_m$  merupakan posisi aktual robot *leader*,  $Q_n$  merupakan posisi aktual robot *follower*,  $Q_n^d$  merupakan posisi robot *follower* yang diharapkan,  $a_{mn}$  yaitu jarak actual antar robot *leader* dan *follower*,  $\phi_{mn}$  merupakan sudut actual pergerakan roda antara *follower* dan *leader*,  $a_{mn}^d$  merupakan jarak robot *leader* dan robot *follower* yang diharapkan, dan  $\phi_{mn}^d$  merupakan sudut pergerakan roda robot *leader* dan robot *follower* yang diharapkan. Berdasarkan Persamaan (1) maka dapat diturunkan persamaan rumus kinematik menjadi

$$\dot{a}_{mn} = -v_m \cos \phi_{mn} + v_n \cos \beta_{mn} + b \omega_n \sin \beta_{mn} \quad (2)$$

$$\dot{\phi}_{mn} = \frac{1}{a_{mn}} \{v_m \sin \phi_{mn} - v_n \sin \beta_{mn} + b \omega_n \cos \beta_{mn}\} - \omega_m \quad (3)$$

Hasil turunan rumus berupa keluaran kecepatan robot dan perubahan sudut. Nilai ini menjadi keluaran pada algoritma LFIT2.

### 2.2. Algoritma Logika Fuzzy Interval Tipe 2 (LFIT) 2

Logika fuzzy merupakan sebuah logika yang memiliki nilai keabuan antara benar dan salah [16]. Tahapan algoritma LFIT2 meliputi fuzzifikasi, basis aturan dan inferensi, reduksi tipe dan defuzzifikasi. Proses fuzzifikasi merupakan suatu tahapan untuk mengubah suatu nilai masukan dari bentuk nilai pasti menjadi *fuzzy input* (variable linguistik). Bentuk variable *linguistic* disediakan dalam bentuk himpunan-himpunan *fuzzy* dengan fungsi keanggotaannya masing-masing. Fungsi keanggotaan dalam algoritma LFIT 2 memiliki dua derajat keanggotaan, yaitu derajat keanggotaan primer dan sekunder (yang merupakan derajat keanggotaan dari derajat keanggotaan primer). Sedangkan derajat keanggotaan sekunder dalam algoritma LFT 2 *interval* adalah bernilai 1, yang disebut FOU (*Footprint Of Uncertainty*) [17], garis yang berada pada nilai-nilai tertinggi disebut sebagai *Upper Membership Function* (UMF), dan yang berada pada nilai-nilai terendah disebut sebagai *Lower Membership Function* (LMF). Fungsi keanggotaan dapat dinyatakan dalam fungsi keanggotaan segitiga, trapesium, sigmoid atau fungsi keanggotaan lainnya. Selanjutnya, tahapan basis kaidah merupakan aturan-aturan berupa pernyataan logika fuzzy yang berbentuk "Jika-Maka" (*If-Then*) yang merupakan inti dari relasi *fuzzy* [16].

Setelah menentukan basis kaidah pada sistem ini, lalu masuk ke tahapan tipe reduksi. Tipe reduksi merupakan *output processing* terdiri dari dua langkah. Langkah pertama adalah reduksi tipe (*type reduction*) dan kedua adalah defuzzifikasi. Hal ini dikarenakan keluaran dari *inference* merupakan himpunan LFIT2. Metode reduksi tipe bertujuan untuk mengubah himpunan fuzzy tipe 2 yang dihasilkan pada proses sebelumnya menjadi himpunan fuzzy tipe 1. Himpunan yang telah direduksi tipe kemudian difuzzifikasi menjadi nilai tegas (*crisp*) untuk keluaran dari fuzzy proses [18]. Tahapan terakhir pada algoritma LFIT2 yaitu defuzzifikasi. Defuzzifikasi merupakan Nilai tegas dari suatu himpunan tereduksi dapat dihitung dengan merata-rata nilai paling kanan dan paling kiri dari himpunan tereduksi, atau dapat dinyatakan sebagai:

$$y = \frac{y_{kiri} + y_{kanan}}{2} \quad (3)$$

## 3. Hasil dan Pembahasan

Pada hasil penelitian ini untuk memvisualisasikan suatu benda kehidupan nyata, benda-benda tersebut harus dimodelkan terlebih dahulu. Model-model kemudian digunakan sebagai dasar dari pembuatan perangkat lunak.

### 3.1. Lingkungan

Lingkungan utama dari simulasi dimodelkan sebagai bidang dua dimensi, dengan sumbu koordinat x dan y.

Arah sumbu x positif dan y positif dari lingkungan dapat dilihat pada Gambar 2 berikut

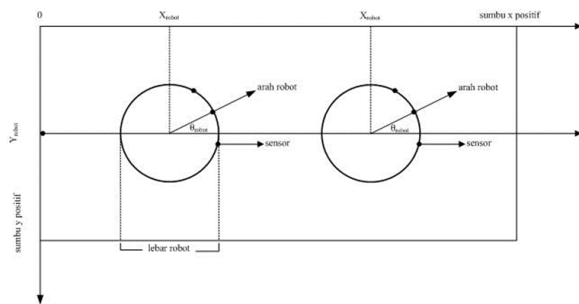


Gambar 2. Model lingkungan simulasi robot

Skala lingkungan adalah 1,4 : 1 cm, di mana 1 cm di nyata lingkungan adalah sama dengan 1,4 piksel pada aplikasi. Ukuran pemodelan lingkungan terbentuk ruang 961 x 561 piksel.

### 3.2. Pemodelan Robot Leader-Follower

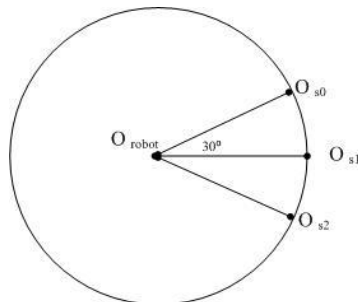
Robot leader dan robot follower divisualisasikan sebagai bidang lingkaran yang berjalan berdasarkan nilai kecepatan dan sudut yang dapat dilihat pada Persamaan (2) dan (3). Letak model robot swarm leader follower pada lingkungan ditentukan oleh koordinat titik pusatnya pada lingkungan, yaitu xrobot dan yrobot, sedangkan arah robot adalah besar sudutnya terhadap sumbu x positif berupa  $\theta_{robot}$ . Besar bidang lingkaran ditentukan oleh besar lebar robot, seperti yang diilustrasikan pada Gambar 3.



Gambar 3. Model Robot leader follower pada lingkungan

### 3.3. Sensor Ultrasonik

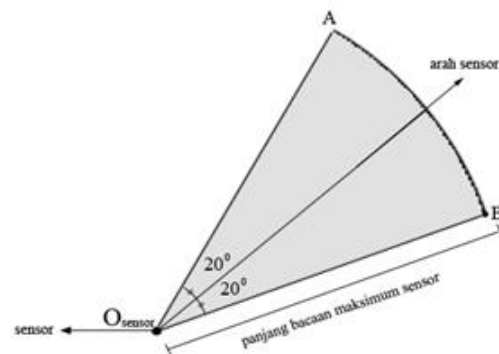
Pada simulasi, model robot swarm memiliki tiga buah model sensor ultrasonik pada bagian depan, samping kiri dan samping kanan, seperti pada Gambar 4.



Gambar 4. Letak tiga buah sensor pada robot leader follower

Sensor-sensor tersebut kemudian disebut sebagai sensor 0, sensor 1, dan sensor 2. Letak dari masing-masing sensor dipisahkan masing-masing sebesar  $30^\circ$  dari  $\theta_{robot}$ . Arah sensor 1 adalah sama dengan arah robot, sehingga besar sudutnya terhadap sumbu x positif adalah sama dengan  $\theta_{robot}$ , koordinat dari sensor 1  $O_{s1}$  ditentukan dengan mentranslasi titik dari Orobot searah dengan Orobot sejauh setengah dari lebar robot. Koordinat dari sensor 0 didapatkan dengan melakukan rotasi dari titik  $O_{s1}$  terhadap Orobot sebesar  $-30^\circ$  sedangkan nilai arahnya adalah  $-30^\circ$  dari  $\theta_{robot}$ . Koordinat dari sensor 2 didapatkan dengan melakukan rotasi dari titik  $O_{s1}$  sebesar  $30^\circ$  terhadap Orobot, dan nilai arahnya didapatkan dengan menambahkan  $30^\circ$  dari  $\theta_{robot}$ . Pada aplikasi ini robot swarm berbentuk melingkar dengan 35 piksel diameter lingkaran atau 25 cm.

Model sensor ultrasonik yang digunakan pada model robot swarm divisualisasikan sebagai titik pada lingkungan yang gerakannya mengikuti gerakan dari model robot swarm. Letak sensor ditentukan dengan merotasi titik pada bagian depan robot berdasarkan jarak sudutnya dari pusat robot. Pada aplikasi yang dirancang, besar bacaan dari model sensor didasarkan pada keadaan ideal, dimana model sensor akan membaca jarak terdekat dengan halangan yang berada pada juring AO sensor seperti diilustrasikan pada Gambar 5.



Gambar 5. Model sensor ultrasonik

Keadaan ideal pada sebuah sensor ultrasonik adalah nilai besar sudut AosenorB sebesar  $400^\circ$  cm, dan bacaan maksimum, yaitu garis OsensorA dan OsensorB adalah sebesar 300 cm. Pada perangkat lunak ini, hasil bacaan model sensor untuk mendapatkan jarak terdekat antara titik model sensor dengan bidang persegi model halangan.

### 3.4 Pengujian Algoritma LFIT2

Robot leader follower dengan algoritma LFIT2 menggunakan inferensi Mamdani. Menggunakan 2 variabel linguistik. Berikut Tabel 1 kaidah 2 variabel linguistik.

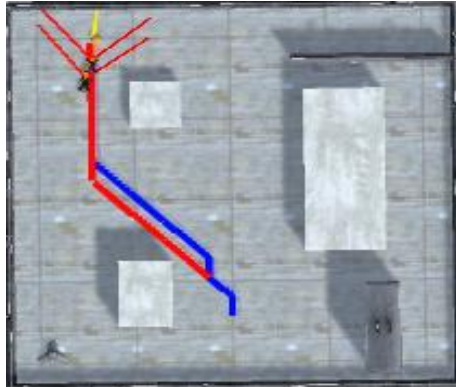
Tabel 1. Basis Kaidah 2 Variabel Linguistik

Sensor0	Sensor1	Sensor2	Kecepatan	Perubahan Sudut
Dekat	Dekat	Dekat	Lambat	Lurus
Dekat	Dekat	Jauh	Lambat	Kanan
Dekat	Jauh	Dekat	Lambat	Lurus
Dekat	Jauh	Jauh	Sedang	Kanan
Jauh	Dekat	Dekat	Lambat	Kiri
Jauh	Dekat	Jauh	Sedang	Kiri
Jauh	Jauh	Dekat	Sedang	Kiri
Jauh	Jauh	Jauh	Cepat	Lurus

Pada sistem navigasi robot *leader follower*, ditanamkan algoritma LFT 2. Sebuah algoritma Logika Fuzzy dengan keluaran berupa nilai perubahan sudut, dan sebuah lainnya dengan keluaran berupa nilai kecepatan dari robot. Pada pengujian ini robot bergerak secara beriringan dan menghindari halangan berupa lima dinding serta bergerak untuk pencapaian target. Berdasarkan Table 1, seluruh kombinasi kaidah akan dilakukan dengan arah robot adalah 0, yaitu searah dengan sumbu x dari model lingkungan. Pengujian algoritma LFIT2 menggunakan aplikasi Unity versi 5. Berikut pengujian berdasarkan kombinasi pada Tabel 1.

a. Kombinasi I

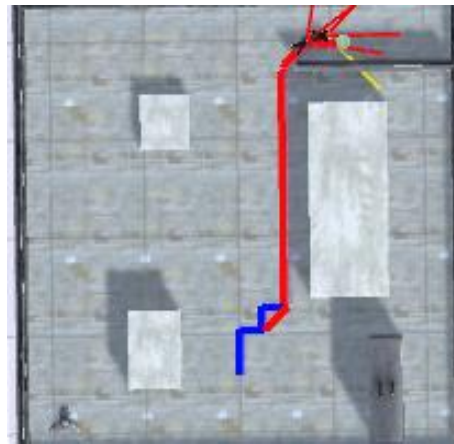
Pada kombinasi I, sensor 0 bernilai dekat, sensor 1 bernilai dekat dan sensor 2 bernilai dekat, sedangkan hasil yang diharapkan adalah model robot berbelok ke kiri. Hasil jalannya model robot pada lingkungan dengan kombinasi ini dapat dilihat pada Gambar 6.



Gambar 6. Lintasan Robot Kombinasi I

b. Kombinasi II

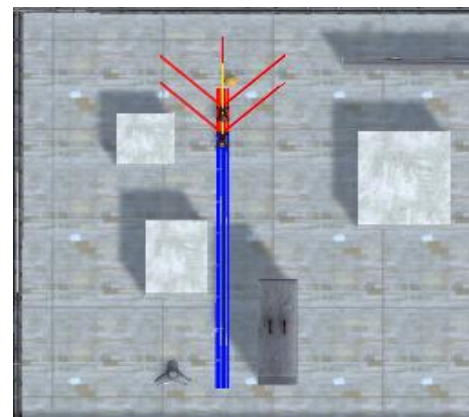
Pada kombinasi II, sensor 0 bernilai dekat, sensor 1 bernilai dekat dan sensor 2 bernilai dekat, sedangkan hasil yang diharapkan adalah model robot berbelok ke kanan. Hasil jalannya model robot pada lingkungan dengan kombinasi ini dapat dilihat pada Gambar 7.



Gambar 7. Lintasan Robot Kombinasi II

c. Kombinasi III

Pada kombinasi III, sensor 0 bernilai dekat, sensor 1 bernilai jauh dan sensor 2 bernilai dekat, sedangkan hasil yang diharapkan adalah model robot berjalan lurus. Hasil jalannya model robot pada lingkungan

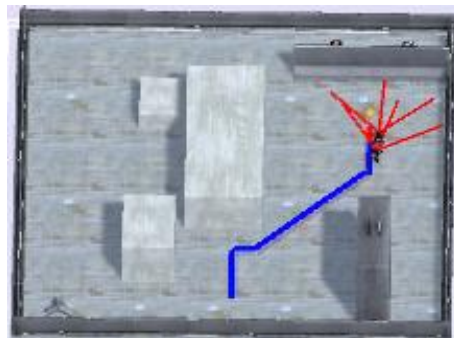


Gambar 8. Lintasan Robot Kombinasi III

dengan kombinasi ini dapat dilihat pada Gambar 8.

d. Kombinasi IV

Pada kombinasi IV, sensor 0 bernilai dekat, sensor 1 bernilai jauh dan sensor 2 bernilai jauh, sedangkan hasil yang diharapkan adalah model robot berbelok ke kanan. Hasil jalannya model robot pada lingkungan dengan kombinasi ini dapat dilihat pada Gambar 9.

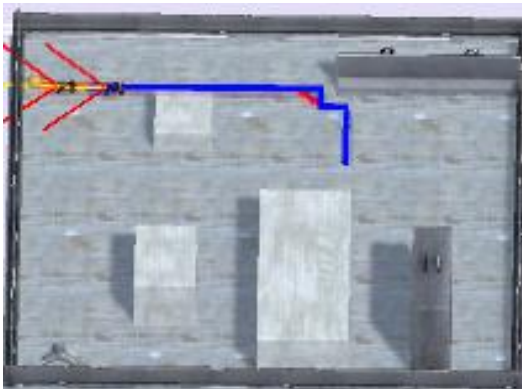


Gambar 9. Lintasan Robot Kombinasi IV

e. Kombinasi V

Pada kombinasi V, sensor 0 bernilai jauh, sensor 1 bernilai dekat dan sensor 2 bernilai dekat, sedangkan hasil yang diharapkan adalah model robot berbelok ke

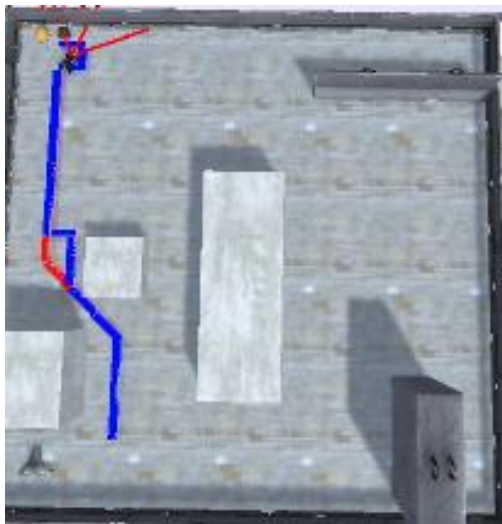
kiri. Hasil jalannya model robot pada lingkungan dengan kombinasi ini dapat dilihat pada Gambar 10.



Gambar 10. Lintasan Robot Kombinasi V

f. Kombinasi VI

Pada kombinasi VI, sensor 0 bernilai jauh, sensor 1 bernilai dekat dan sensor 2 bernilai jauh, sedangkan hasil yang diharapkan adalah model robot berbelok ke kiri. Hasil jalannya model robot pada lingkungan dengan kombinasi ini dapat dilihat pada Gambar 11.



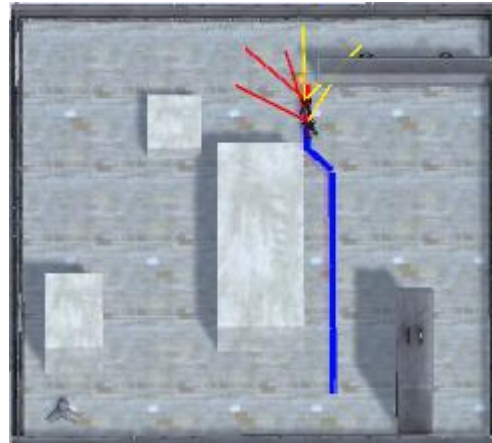
Gambar 11. Lintasan Robot Kombinasi VI

g. Kombinasi VII

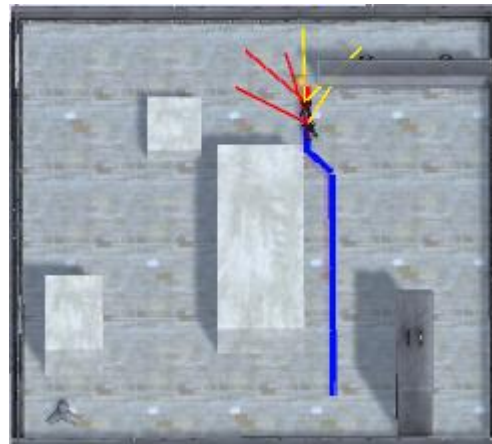
Pada kombinasi VII, sensor 0 bernilai jauh, sensor 1 bernilai jauh dan sensor 2 bernilai dekat, sedangkan hasil yang diharapkan adalah model robot berbelok ke kiri. Hasil jalannya model robot pada lingkungan dengan kombinasi ini dapat dilihat pada Gambar 12.

h. Kombinasi VIII

Pada kombinasi I, sensor 0 bernilai jauh, sensor 1 bernilai jauh dan sensor 2 bernilai jauh, sedangkan hasil yang diharapkan adalah model robot berjalan lurus. Hasil jalannya model robot pada lingkungan dengan kombinasi ini dapat dilihat pada Gambar 13.



Gambar 12. Lintasan Robot Kombinasi VII



Gambar 13. Lintasan Robot Kombinasi VIII

#### 4. Kesimpulan

Penelitian robot *leader follower* menggunakan navigasi algoritma LFIT2 menghasilkan 8 basis aturan yang dapat berjalan sesuai dengan target tujuan. Pada simulasi ini, robot berhasil bergerak dengan menghindari halangan serta berjalan secara beriringan dengan posisi robot *follower* selalu mengikuti posisi robot *leader*. Pada penelitian selanjutnya robot akan ditambahkan 1 robot dengan implementasi secara nyata.

#### Daftar Rujukan

- [1] A. Saffiotti, "The uses of fuzzy logic in autonomous robot navigation," vol. 1, 1997.
- [2] C. Wagner and H. Hagra, "A Genetic Algorithm Based Architecture for Evolving Type-2 Fuzzy Logic Controllers for Real World Autonomous Mobile Robots," *2007 IEEE Int. Fuzzy Syst. Conf.*, pp. 1–6, 2007.
- [3] Z. Li and X. Liu, "Distributed Tracking Control for Linear Multiagent Systems With a Leader of Bounded Unknown Input," vol. 58, no. 2, pp. 518–523, 2013.
- [4] S. K. Hussein and M. A. Al-Mutairi, "A Novel Prototype Model for Swarm Mobile Robot Navigation Based Fuzzy Logic Controller," *Int. J. Comput. Sci. Inf. Technol.*, vol. 11, no. 02, pp. 63–77, 2019.
- [5] G. F. Fitriana and S. Nurmaini, "Interval Type 2 Fuzzy Logic Algorithm for Leader-Follower Robot."

- [6] T. Balch, R. C. Arkin, and S. Member, "Behavior-Based Formation Control for Multirobot Teams," *IEEE Trans. Robot. Autom.*, vol. 14, no. 6, pp. 926–939, 1998.
- [7] J. R. T. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *IEEE Trans. Robot. Autom.*, vol. 19, no. 6, pp. 933–941, 2003.
- [8] A. Brunete, M. Hernando, E. Gamba, and J. E. Torres, "A behaviour-based control architecture for heterogeneous modular," *Rob. Auton. Syst.*, vol. 60, pp. 1607–1624, 2012.
- [9] K. Tan and M. A. Lewis, "Virtual Structures for High-Precision Cooperative Mobile Robotic Control \*," *Proc IROS 96 IEEE*, pp. 132–139, 1996.
- [10] M. A. Lewis and K. Tan, "High Precision Formation Control of Mobile Robots Using Virtual Structures," *Auton. Robot. 4*, vol. 403, pp. 387–403, 1997.
- [11] J. P. Desai, J. P. Ostrowski, and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots," *Robot. Autom. IEEE Trans.*, vol. 17, no. 6, pp. 905–908, 2001.
- [12] J. L. Whittington and R. G. Bell, "Leader – member exchange , enriched jobs , and goal-setting: Applying fuzzy set methodology," *J. Bus. Res.*, vol. 69, no. 4, pp. 1401–1406, 2016.
- [13] J. D. McLurkin, "Analysis and Implementation of Distributed Algorithms for Multi-Robot Systems," *Theses Diss.*, 2008.
- [14] P. Chandak, "Study and Implementation of ‘ Follow the Leader," 2002.
- [15] B. Tutuko, S. Nurmaini, Saparudin, and G. F. Fitriana, "Enhancement of Non-Holonomic leader-follower formation using Interval Type-2 Fuzzy Logic Controller," *Int. J. Online Eng.*, vol. 14, no. 9, pp. 124–142, 2018.
- [16] L. Hakim and V. Yonatan, "Deteksi Kebocoran Gas LPG menggunakan Detektor Arduino dengan algoritma Fuzzy Logic Mandani," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 1, no. 2, p. 114, 2017.
- [17] Q. Liang and J. M. Mendel, "Interval Type-2 Fuzzy Logic Systems : Theory and Design," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5 october, pp. 535–550, 2000.
- [18] N. H. Seng, "Implementation of Various Types of Fuzzy Controls on a Mobile Robot Using Sonar Sensors," 2008.