



JASEE

Journal of Application and Science on Electrical Engineering

<https://jurnal.widyagama.ac.id/index.php/jasee/index>



Implementation of a 3-wheeled Wall Following Robot Navigation System using Coppelia

Nurisma Zenita¹, Kurniawan Budi K.², Abdul Mujib³, Nur Laily⁴
^{1,2,3,4}Universitas Negeri Yogyakarta, Indonesia

Corresponding author, email: Nurismazenita.2019@student.uny.ac.id

Abstract

This article will design a controller for a three-wheeled wall-following robot based on Coppelia software. The problem with the wall-following robot is how to control a follower robot to move constantly along the wall in the intended direction. The robot controller uses Sugeno fuzzy logic as a rule base for stationary and moving states. This controller is created through distance and orientation navigation. Both are estimated by the robot model and corrected by the sensor measurement results. In cases where the wall is not available, for example, an open door, the robot will stop then there will be feedback to take the next step. The designed controller has been verified experimentally, where the results show an error rate of five millimeters from the actual distance.

Keywords: 3-wheel robot, Coppelia, Sugeno, wall-following, sensors, navigation.

p-ISSN : 2721-3625
e-ISSN : 2721-320X

1. PENDAHULUAN

Pada artikel ini, kami menggunakan Coppelia Simulator sebagai simulasi robot wall-following 3 roda yang sudah kami rancang. CoppeliaSim, sebelumnya dikenal sebagai V-REP, adalah simulator robot yang digunakan dalam industri, pendidikan, dan penelitian. Ini awalnya dikembangkan dalam Toshiba R&D dan saat ini sedang aktif dikembangkan dan dikelola oleh Coppelia Robotics AG, sebuah perusahaan kecil yang berlokasi di Zurich, Swiss. Itu dibangun di sekitar arsitektur kontrol terdistribusi yang memiliki skrip Lua atau plug-in C/C++ yang bertindak sebagai pengontrol sinkron individual [1]. Pengontrol asinkron tambahan dapat dijalankan dalam proses, utas, atau mesin lain melalui berbagai solusi middleware (ROS, API jarak jauh, ZeroMQ) dengan bahasa pemrograman seperti C/C++, Python, Java, dan Matlab.

CoppeliaSim menggunakan mesin kinematika untuk perhitungan kinematika maju dan mundur, dan beberapa perpustakaan simulasi fisika (Bullet, ODE, Vortex, Newton Game Dynamics) untuk melakukan simulasi benda tegar. Model dan adegan dibangun dengan merakit berbagai objek (jerat, sambungan, berbagai sensor, awan titik, pohon OC, dll.) ke dalam struktur hierarkis. Fungsionalitas tambahan, disediakan oleh plug-in, meliputi: perencanaan gerak (melalui OMPL), penglihatan sintesis dan pemrosesan pencitraan (misalnya melalui OpenCV), deteksi tabrakan, perhitungan jarak minimum, antarmuka pengguna grafis khusus dan visualisasi data (misalnya melalui

<https://doi.org/10.31328/jasee.v3i01.4>

Received: 23 December 2021

Revised: 16 March 2022

Accepted: 20 March 2022, published by ©UWG Press 2022

plot [2]. Robot wall follower merupakan jenis mobile robot yang memiliki kemampuan dalam mengikuti dinding (wall following). Robot tipe ini menjadi salah satu pilihan yang dijadikan sebagai objek penelitian baik dalam event nasional maupun internasional [3].

Pada beberapa kasus, permasalahan yang muncul adalah bagaimana membuat suatu sistem navigasi pada robot agar robot mampu menyelesaikan tantangan yang memiliki rintangan berupa dinding dalam suatu lintasan. Salah satu metode yang dapat digunakan untuk menyelesaikan permasalahan tersebut adalah dengan mengikuti kontur dinding. Navigasi wall following adalah suatu algoritma untuk menyediakan orientasi navigasi kepada robot dengan menyusuri dinding [4]. Salah satu keuntungannya adalah tidak memerlukan garis penuntun ataupun tanda khusus sebagai arahan bagi robot. Cara kerjanya adalah dengan mengatur jarak dinding dengan robot tetap konstan. Jika terjadi perubahan, maka robot akan bergerak untuk kemudian menyesuaikan jarak kembali. Selain memiliki sistem navigasi yang baik, robot wall follower juga harus memiliki sistem kendali yang tepat. Untuk memuluskan pergerakan robot saat menelusuri lintasan, diperlukan suatu pengendali yang membuat pergerakan robot saat dioperasikan dapat sesuai dengan harapan.

Robot Wall Follower adalah salah satu jenis dari mobile robot yang memiliki kemampuan untuk mengikuti kontur dinding. Aktuator dari robot wall follower dapat berupa roda maupun berkaki. Keunggulan dari robot wall follower dengan aktuator berupa kaki yaitu perpindahan gerak dari satu posisi ke posisi lain lebih cepat dibandingkan dengan robot wall follower dengan aktuator berupa kaki. Untuk dapat melakukan kombinasi gerakan tertentu, perlu dilakukan beberapa pengaturan supaya pergerakan robot dapat sesuai dengan keinginan. Navigasi wall following merupakan suatu algoritma untuk menyediakan orientasi navigasi kepada robot dengan menyusuri dinding [5]. Salah satu keuntungannya adalah tidak perlu adanya garis penuntun atau suatu benda khusus sebagai arahan bagi robot. Cara kerjanya adalah dengan mengatur jarak dinding dengan robot agar tetap konstan. Bila terjadi perubahan, maka robot akan bergerak untuk kemudian menyesuaikan jarak lagi.

2. STUDI PUSTAKA

Coppelia adalah software simulator robot dengan bahasa pemrograman seperti C/C++, Python, Java, dan Matlab. CoppeliaSim menggunakan mesin kinematika untuk perhitungan kinematika maju dan mundur, dan beberapa perpustakaan simulasi untuk melakukan simulasi benda tegar. Model dan adegan dibangun dengan merakit berbagai objek ke dalam struktur hierarkis. Fungsionalitas tambahan, disediakan oleh plug-in, meliputi: perencanaan gerak, penglihatan sintesis dan pemrosesan pencitraan, deteksi tabrakan, perhitungan jarak minimum, antarmuka pengguna grafis khusus dan visualisasi data [2].

2.1 Fuzzy Logic

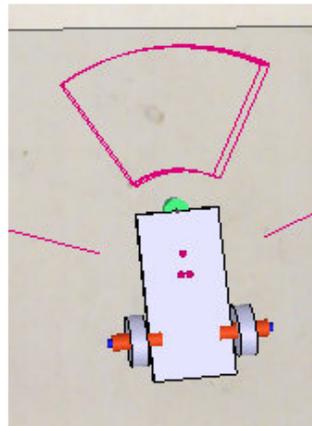
Fuzzy logic adalah suatu logika yang memungkinkan nilai keanggotaan antara 0 dan 1. Logika fuzzy di gunakan untuk menerjemahkan suatu besaran yang diekspresikan menggunakan linguistik karena mirip dengan cara berpikir manusia. Sistem fuzzy logic dapat mempresentasikan pengetahuan manusia dalam bentuk matematis dengan menyerupai cara berfikir manusia [6][4]. Kelebihan metode fuzzy tidak bergantung pada model matematis sistem yang dikendalikan namun bergantung pada kegiatan percobaan dan juga pengalaman yang dimiliki perancang. Metode kontrol berbasis fuzzy juga menjadi salah satu alternatif dalam menghadapi sistem yang tidak linier. Dalam memproses setiap masukan dari lingkungan, metode fuzzy menggunakan range-to-point control. Setiap parameter (input/output) diklasifikasikan dalam variabel linguistik [7].

Nilai input dari sensor dalam bentuk variabel linguistik ini akan di proses di dalam aturan kontrol yang dibuat perancang untuk menentukan nilai output yang sesuai. Aturan kontrol inilah yang menjadi

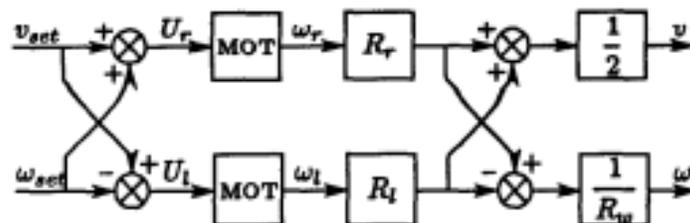
kekuatan metode fuzzy dalam menghadapi sistem yang tidak linier. Kontroler fuzzy dirancang di bawah dua asumsi, bahwa pertama, tidak ada dinding yang bergerak atau hambatan, dan kedua, tidak ada sektor buta antara sensor. Ini berarti bahwa dinding selalu dirasakan oleh setidaknya salah satu sensor tidak peduli apa orientasi dari roboti. Asumsi pertama tidak mewakili keterbatasan yang disebabkan oleh konsep persepsi umum. Dia dibuat untuk menguji konsep ini dalam kombinasi dengan pengontrol fuzzy pada prinsipnya. Asumsi kedua bisa dipenuhi oleh sejumlah besar sensor [8].

2.2 Model Robot

Robot Maze ialah robot dengan 3 roda sebagai alat geraknya yang bergerak menyusuri ruang di arena lapangan berbentuk rumah [6]. Robot Maze ini merupakan jenis robot Wall Follower, yaitu suatu jenis robot beroda yang memiliki sensor untuk mendeteksi suatu dinding dengan sirkuit tertentu. Ia bergerak menelusuri dinding tersebut dengan tujuan tertentu sebagaimana pada Gambar 1 dan 2. Robot Wall Follower bisa disebut robot pintar, sebab merupakan gabungan dari pengolah, sensor, dan pengontrol atau aktuator [7]. Tidak hanya pada Kontes Robot Pemadam Api, dalam dunia sebenarnya robot juga mulai digunakan untuk memadamkan api dikarenakan pada sektor ini berbahaya untuk manusia. Banyak ruangan yang harus dimasuki untuk menemukan titik api ketika robot memadamkan api dalam suatu gedung dan robot harus menemukan jalan terpendek untuk segera kembali ketempat awal robot berjalan yang menandakan bahwa api sudah dipadamkan atau sebaliknya [9]. Supaya robot tidak kembali lagi ke ruangan yang sudah dimasuki dan dapat menemukan jalan terpendek untuk segera kembali ke homingnya maka robot perlu suatu kemampuan untuk menyelesaikan misinya.



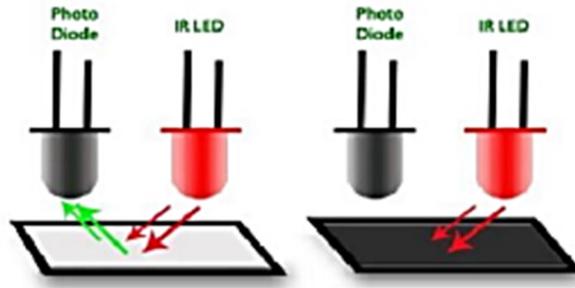
Gambar 1 . Robot Maze 3-roda



Gambar 2. Model Robot Maze

2.3 Sensor

Sensor adalah sesuatu yang digunakan untuk mendeteksi adanya perubahan lingkungan fisik atau kimia, sedangkan transduser adalah pengubah variable keluaran dari sensor menjadi besaran listrik [8]. Sensor sering digunakan untuk pendeteksian pada saat melakukan pengukuran atau pengendalian. Sensor garis atau sensor proximity adalah sensor yang berfungsi mendeteksi warna garis hitam atau putih. Biasanya sensor garis ini terdapat pada robot line follower atau line tracking. Perhatikan gambar 3. berikut.



Gambar 3. Ilustrasi sensor garis Proximity

2.4 Rule Base

Di dalam artikel ini, yang disebut pengkodean produk korelasi dalam ketiga sensor yang digunakan, yang akan diilustrasikan secara singkat dengan cara berikut ini [9]:

```
noseSensor1=sim.getObjectHandle("FrontUS1")
noseSensor2=sim.getObjectHandle("FrontUS2")
noseSensor3=sim.getObjectHandle("FrontUS3")
end
```

Berdasarkan script diatas, robot wall-following yang sudah dirancang menggunakan 3 sensor Proximity.

```
function sysCall_actuation()
result1, jarak1=sim.readProximitySensor(noseSensor1)
result2=sim.readProximitySensor(noseSensor2)
result3=sim.readProximitySensor(noseSensor3)
--print(jarak1)
```

Result 1,2,3 diatas merupakan suatu kondisi dimana robot akan otomatis menyusuri dinding dan bisa membaca sensor.

```
if (result2==0)and(result3==0)then
sim.setJointTargetVelocity(leftMotor,5)
sim.setJointTargetVelocity(rightMotor,3.5)
end
```

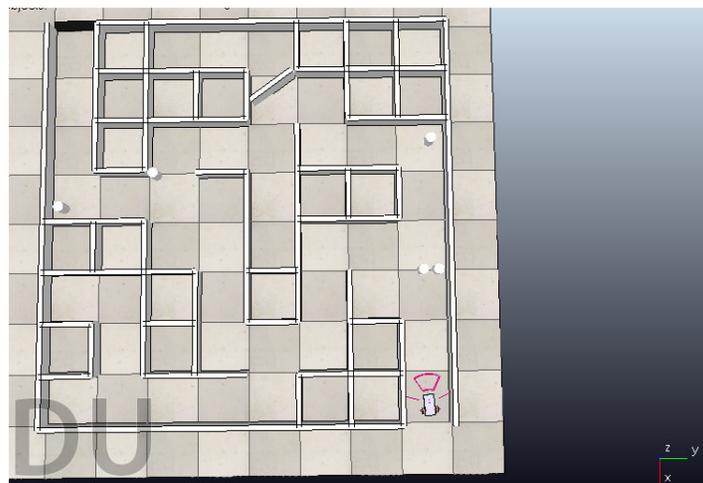
Kondisi diatas menjelaskan suatu kondisi 2 dan kondisi 3 diam maka motor kiri ada pada kecepatan 5 dan motor kanan ada pada kecepatan 3.5. begitupun dengan pergerakan robot selanjutnya. Yang perbandingan kecepatannya lebih besar di motor kanan. Sedangkan, saat kondisi kedua, robot akan membaca sensor sesuai kecepatan yang dimilikinya seperti script dibawah ini.

```

if (result2==1)then
    sim.setJointTargetVelocity(leftMotor,3.5)
    sim.setJointTargetVelocity(rightMotor,4.5)
end
    
```

2.5 Arena Robot

Arena robot ini, kami menggunakan model Labirin yang bersekat-sekat karena dirancang untuk wall-following dan robot Maze ini bisa menyusuri dinding dengan bantuan pembacaan sensor Proximity



Gambar 4. Arena Robot

3. METODE

Dalam system control logika fuzzy terdapat beberapa tahapan operasional yang meliputi fuzzyfikasi, aturan dasar, dan defuzzyfikasi . Gambar 5 adalah blok diagram dari controller logika fuzzy. Kami menggunakan metode Sugeno, aturan linguistik untuk menentukan aksi kontrol apa yang harus dilakukan dalam merespon nilai masukan yang diberikan. Evaluasi rule terdiri dari sejumlah aturan yang biasanya dinyatakan secara linguistic. Evaluasi rule juga mengacu pada fuzzy interface, mengaplikasikan aturan pada masukan fuzzy yang dihasilkan dalam proses fuzzyfikasi, kemudian mengevaluasi tiap aturan dengan masukan yang dihasilkan dari proses fuzzyfikasi. Aturan fuzzy dinyatakan dengan “IF...THEN...”. Pada sistem fuzzy alat ini menggunakan aturan-aturan dalam bentuk bahasa alami yangn dibatasi oleh istilah linguistik, serta sintaksis yang baku. Sintaksi tersebut adalah:

If antecedent 1 And antecedent 2 Then consequent 1 And consequent 2

Dimana:

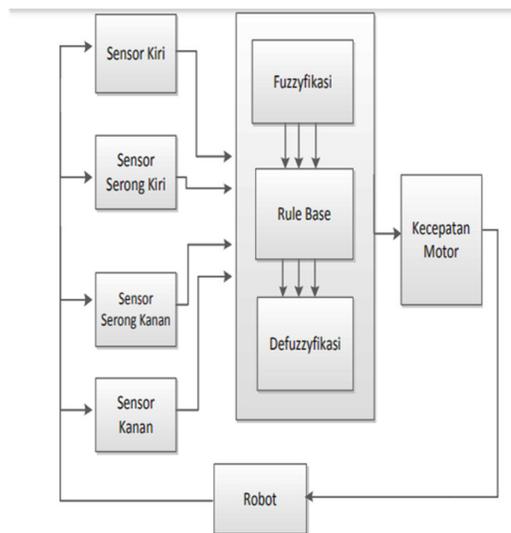
And adalah salah satu operator logika fuzzy yang diizinkan

Antecedent adalah bentuk dari: Variabel masukan = Label (contohnya: jarak sensor kiri = dekat, dimana jarak sensor kiri adalah variabel masukan dan dekat adalah salah satu label fungsi keanggotaan yang berhubungan dengan jarak sensor kiri). *Consequent* adalah bentuk: variabel keluaran = label (contohnya: kecepatan motor = cepat). Pada simulasi wall-following robot ini, menggunakan 2 peng-kondisian sebagai berikut :

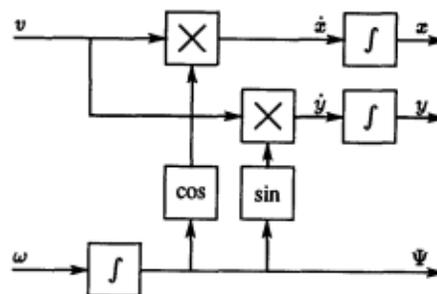
```

if (result2==0)and(result3==0)then
sim.setJointTargetVelocity(leftMotor,5)
sim.setJointTargetVelocity(rightMotor,3.5)
end
    
```

Jadi, ketika kondisi 2 dan 3 berada posisi 0. Maka kecepatan motor kiri 5 dan kecepatan motor 3.5.



Gambar 5. Skematik Robot Wall-Following

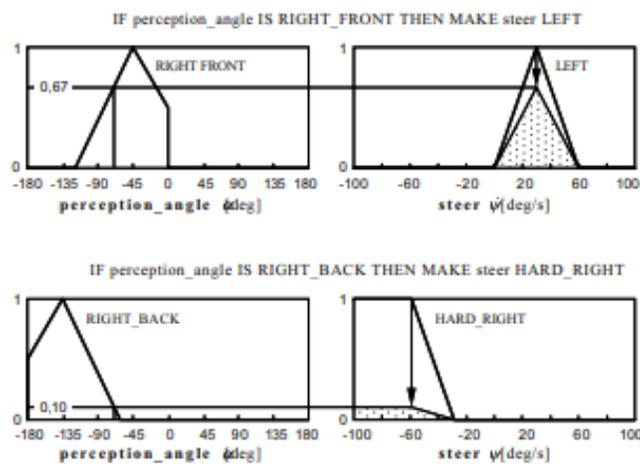


Gambar 6. Model Robot Wall Following

Model robot diringkas pada gambar 7. Kedua bagian dapat digabungkan untuk membentuk model lengkap. Namun, beberapa penyederhanaan dapat dibuat. Pertama, kopling silang ganda di bagian pertama menyebabkan istilah kopling antara terjemahan dan rotasi kecepatan. Dengan asumsi jari-jari roda sama dan motor identik unit kontrol, istilah ini dapat diabaikan. Transfer fungsi antara V_{set} dan v dan antara U_{set} dan w didekati oleh sistem orde pertama dengan keuntungan yang sama untuk satu dan konstanta waktu T_v dan T_w masing-masing [10]. Kedua, karena tugas kontrol untuk menjaga robot tetap pada garis lurus, orientasi ψ dapat dianggap kecil. Sebagai akibatnya, $\cos \psi \approx 1$

dan $\sin \theta$. Menjaga orientasi lebih kecil dari lebar balok maksimum sensor ultrasonik juga diperlukan untuk mencegah suara gelombang dari yang dipantulkan dan karenanya membuat mustahil untuk mengukur jarak. Ketiga, terjemahan kecepatan dapat diambil sebagai konstan saat mengemudi sepanjang a garis lurus. Ini menyiratkan bahwa perubahan arah ψ sebanding dengan orientasi: $\psi = v \theta$, di mana v dapat dianggap sebagai faktor keuntungan.

Dalam artikel ini, sudut -70° dilihat sebagai 67% RIGHT_FRONT dan oleh karena itu himpunan fuzzy atau fungsi keanggotaan dari keluaran aturan KIRI dikalikan dengan faktor 0,67. Hal yang sama dilakukan dengan output dari aturan kedua HARD_RIGHT, yang dikalikan dengan 0,1 karena nilai -70° pada saat yang sama milik 10% ke fuzzy-set RIGHT_BACK[8][11]. Kontrol yang tegas perintah dihasilkan dari dua kontradiksi perintah mengarahkan KIRI dan mengarahkan HARD_RIGHT terlebih dahulu menjumlahkan dua himpunan fuzzy berskala (kombinasi penjumlahan). Kemudian koordinat untuk pusat gravitasi dari area yang dihasilkan mendefinisikan apa yang disebut nilai defuzzified untuk mengarahkan (centroid defuzzifikasi), sebagaimana pada Gambar 7.



Gambar 7. Grafik Sudut

Kecepatan robot, ditunjukkan pada tabel 1, denganmasukan p dan θ dipandu oleh gagasan bahwa robotmembutuhkan pengereman minimum dan percepatan maksimumsemakin jauh dari dinding dan semakin kecil perubahannyapersepsi umum dalam waktu[6]. Perubahan cepat secara umum persepsi di sisi lain pasti menghasilkan pengereman darurat.

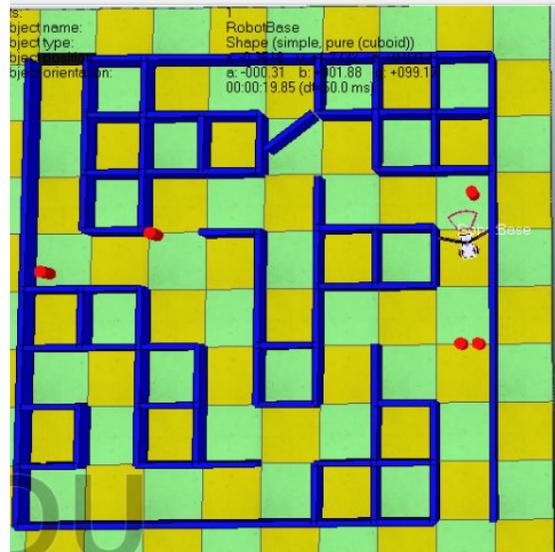
Tabel 1. Kecepatan Robot

WAKTU	KECEPATAN	SENSOR 1	SENSOR 2	SENSOR 3

4. HASIL DAN PEMBAHASAN

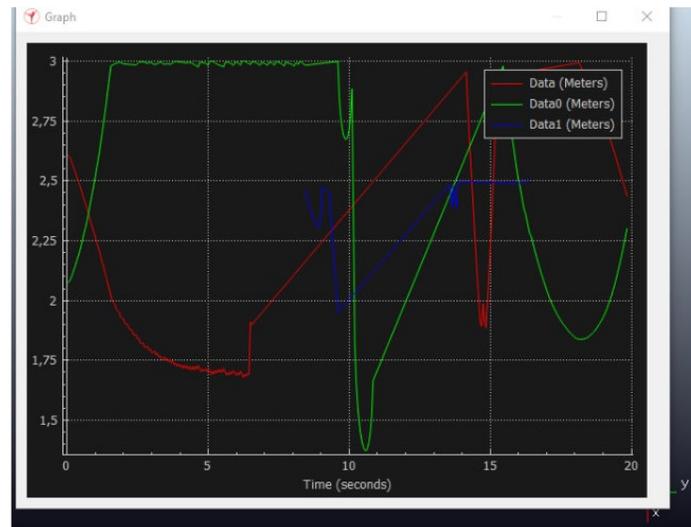
4.1 Simulasi Wall-Following Robot

Simulasi robot wall following menggunakan Coppelia dapat dilihat pada Gambar 8 berikut:



Gambar 8. Simulasi Wall-Following Robot

Pada simulasi diatas, terlihat bahwa ketika robot mengetahui posisi objek, ia akan otomatis menghindari objek tersebut dan robot akan berbelok ke kiri menggunakan susur kiri untuk menyusuri dinding dan menghindari objek didepannya. Hasil simulasi diatas dapat dilihat pada grafik di Gambar 9 berikut.



Gambar 9. Grafik pergerakan robot

Grafik pada Gambar 9 merupakan keluaran dari pergerakan robot. Dimana garis merah (Sensor kiri), garis hijau (Sensor kanan), dan garis biru (Sensor depan). Dibagian kiri garis grafik merupakan jarak yang ditempuh oleh robot. Dan dibagian bawah garis grafik merupakan waktu yang

ditempuh robot pada saat simulasi. Dari grafik diatas, dapat dikonfigurasi pada Tabel 2 dibawah ini.

Tabel 2. Perbandingan Ketiga Sensor

WAKTU	KECEPATAN	SENSOR 1	SENSOR 2	SENSOR 3
5	3.5	1	1	0
10	4	1	1	1
15	4.5	1	1	1
20	5	1	1	0

Keterangan,

Sensor 1 = sensor kiri (Merah)

Sensor 2 = sensor kanan (Hijau)

Sensor 3 = sensor depan (Biru)

Pada tabel 2, dapat diartikan bahwa ketika robot menyusuri dinding dalam waktu 5 detik dan kecepatan 3.5 m, maka yang akan membaca adalah sensor 1 dan sensor 2, sedangkan sensor 3 tidak membaca. Pada waktu 10 detik dengan kecepatan 4 m, semua sensor bekerja dan terdeteksi pada grafik. Pada waktu 15 detik dan kecepatan 4.5 m, semua sensor bekerja dan terdeteksi pada grafik. Pada waktu 20 detik dan kecepatan 5m sensor yang membaca adalah sensor 1 dan sensor 2, sedangkan sensor 3 tidak membaca seperti yang ada pada gambar 10 diatas.

5. SIMPULAN

Dalam artikel ini, kami membuat sebuah robot Wall-Following roda 3 dengan model robot maze. Dengan adanya CoppeliaSim ini kami menggunakan sensor Proximity dan sensor Kamera sebagai pergerakan robot ini. Robot ini disusun dengan Fuzzy-Logic Controller yang berarti peng-kondisian dengan rule base if-then dan sistem Navigasi. Yang mana, sensor Proximity digunakan dalam Fuzzy-logic controller yang menyatakan kondisi/posisi robot akan menyusuri dinding. Robot akan mudah dalam mendeteksi suatu objek didepannya dan akan menghindari objek sebagaimana mestinya. Pada tabel pembahasan, dapat diartikan bahwa ketika robot menyusuri dinding dalam waktu 5 second dan kecepatan 3.5 m, maka yang akan membaca adalah sensor 1 dan sensor 2, sedangkan sensor 3 tidak membaca. Dengan menunda koreksi pada orientasi dan meningkatkan keuntungan pengamat, kesalahan ini dapat sangat dikurangi. Pada percobaan akhir robot mengalami error sebesar beberapa millimeter dari jarak yang diinginkan ke dinding. Penelitian selanjutnya meliputi perbaikan observer dengan model robot yang lebih baik dan lebih detail (crossistilah kopling). Kemungkinan lain adalah penggunaan algoritma dead-reckoning sebagai pengganti model robot.

DAFTAR PUSTAKA

- [1] C. Fernando Joventino, A. S. De Oliveira, J. Alberto Fabro, and J. H. M. Pereira, "Application of a ROS / CoppeliaSim Integration in a Practical 'OBR' Competition Scenario," *2020 Latin American Robotics Symposium, 2020 Brazilian Symposium on Robotics and 2020 Workshop on Robotics in*

- Education, LARS-SBR-WRE 2020*, pp. 1–6, 2020, doi: 10.1109/LARS/SBR/WRE51543.2020.9306988.
- [2] I. Tursynbek and A. Shintemirov, "Modeling and Simulation of Spherical Parallel Manipulators in CoppeliaSim (V-REP) Robot Simulator Software," *2020 International Conference Nonlinearity, Information and Robotics, NIR 2020*, 2020, doi: 10.1109/NIR50484.2020.9290227.
- [3] P. van Turenout, G. Honderd, and L. J. van Schelven, "Wall-following control of a mobile robot," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 1, no. May, pp. 280–285, 1992, doi: 10.1109/robot.1992.220250.
- [4] T. Dash, R. S. Soumya, T. Nayak, and G. Mishra, "Neural network approach to control wall-following robot navigation," *Proceedings of 2014 IEEE International Conference on Advanced Communication, Control and Computing Technologies, ICACCCT 2014*, no. 978, pp. 1072–1076, 2015, doi: 10.1109/ICACCCT.2014.7019262.
- [5] T. Yata and L. Kleeman, "Wall Following Using Angle Information Measured," no. May, pp. 1590–1596, 1998.
- [6] C. H. Hsu and C. F. Juang, "Evolutionary robot wall-following control using type-2 fuzzy controller with species-DE-activated continuous ACO," *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 1, pp. 100–112, 2013, doi: 10.1109/TFUZZ.2012.2202665.
- [7] U. Farooq, A. Khalid, M. Amar, A. Habiba, S. Shafique, and R. Noor, "Design and low cost implementation of a fuzzy logic controller for wall following behavior of a mobile robot," *ICSPS 2010 - Proceedings of the 2010 2nd International Conference on Signal Processing Systems*, vol. 2, pp. 740–746, 2010, doi: 10.1109/ICSPS.2010.5555781.
- [8] C. F. Scatambulo Costa and E. Alexandre Franciscon, "Data Mining applied to the navigation task in autonomous robots," *Proceedings - IEEE Symposium on Computers and Communications*, vol. 2020-July, pp. 0–5, 2020, doi: 10.1109/ISCC50000.2020.9219731.
- [9] S. Oswal and D. Saravanakumar, "Line following robots on factory floors: Significance and Simulation study using CoppeliaSim," *IOP Conference Series: Materials Science and Engineering*, vol. 1012, p. 012008, 2021, doi: 10.1088/1757-899x/1012/1/012008.
- [10] M. Katsev, A. Yershova, B. Tovar, R. Ghrist, and S. M. Lavalle, "Mapping and Pursuit-Evasion strategies for a simple wall-following robot," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 113–128, 2011, doi: 10.1109/TRO.2010.2095570.
- [11] C. F. Juang and C. H. Hsu, "Reinforcement ant optimized fuzzy controller for mobile-robot wall-following control," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 3931–3940, 2009, doi: 10.1109/TIE.2009.2017557.