

Perancangan Algoritma *Dynamic Path Planning* pada *Autonomous Mobile Robot* Menggunakan Modifikasi *Crossover* Algoritma Genetika

Nia Saputri Utami¹, Achmad Jazidie², Rusdhianto Effendi Abdul Kadier³

¹ Program Studi Teknik Elektro, Jurusan Teknologi Produksi dan Industri, Institut Teknologi Sumatera, Jl. Terusan Ryacudu, Way Huwi, Kec. Jati Agung, Kabupaten Lampung Selatan, Lampung 35365

² Departemen Teknik Elektro, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember

³ Departemen Teknik Elektro, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember

* Corresponding email: nia.utami@el.itera.ac.id

Riwayat Artikel

Diterima
03/12/2021
Disetujui
24/01/2021
Diterbitkan
24/01/2021

Abstrak

Algoritma genetika digunakan untuk menghasilkan sistem navigasi cerdas pada *autonomous mobile robot* agar dapat menghindari tabrakan baik dengan halangan statis maupun halangan dinamis yang ada di sekitarnya. Modifikasi algoritma genetika dilakukan pada operator *crossover* yaitu dengan menambahkan syarat bahwa nilai *fitness* keturunan hasil *crossover* harus dibandingkan dengan induknya terlebih dahulu sebelum diteruskan ke proses mutasi gen. Lintasan utama dihasilkan dari GA statis sedangkan lintasan alternatif dihasilkan dari GA dinamis yang bekerja berdasarkan *moving area sensor* dan dilengkapi dengan algoritma prediksi pergerakan halangan. Hasil simulasi menunjukkan bahwa *moving area sensor* mampu mendeteksi halangan dinamis yang ada di sekitar lintasan utama dan apabila halangan dinamis tersebut berpotensi menabrak *autonomous mobile robot* maka GA dinamis akan segera membuat lintasan alternatif yang tidak jauh menyimpang dari lintasan utama sehingga *autonomous mobile robot* terhindar dari halangan dinamis tersebut. Setelah itu, *autonomous mobile robot* kembali lagi ke lintasan utama untuk kemudian menuju titik target.

Kata Kunci: Algoritma Genetika, *Path Planning*, *Autonomous Mobile Robot*

Abstract

Genetic algorithms are used to produce intelligent navigation systems on autonomous mobile robots in order to avoid collisions with both static and dynamic obstacles in the vicinity. Modification of the genetic algorithm is carried out on the crossover operator, namely by adding the condition that the fitness value of the offspring from the crossover must be compared with the parent before being forwarded to the gene mutation process. The main path is generated from a static GA, while the alternative path is generated from a dynamic GA which works based on a moving area sensor and is equipped with an obstacle movement prediction algorithm. The simulation show that the moving area sensor is able to detect dynamic obstacles around the main track. If the dynamic obstacle has the potential to crash into the autonomous mobile robot, the dynamic GA will immediately create an alternative path that does not deviate far from the main track so that the autonomous mobile robot avoids dynamic obstacles. And after that, the autonomous mobile robot returns to the main track and then goes to the target point.

Keywords: Genetic Algorithm, Path Planning, Autonomous Mobile Robot

1. Pendahuluan

Autonomous mobile robot telah banyak digunakan baik di dalam dunia industri maupun dalam kehidupan sehari-hari [1]. Salah satu manfaat dari *autonomous mobile robot* yaitu dapat digunakan sebagai alat pembersih lantai seperti pada [2]. Selain itu, para peneliti juga banyak

menggunakannya sebagai objek penelitian dengan berbagai macam tujuan. Ada tiga hal mendasar dari suatu *autonomous mobile robot* yang harus dimiliki yaitu kemampuan bergerak, persepsi, dan navigasi [3]. Agar *autonomous mobile robot* tersebut dapat bernavigasi, maka ia harus tahu posisinya saat ini, posisi tujuan, dan cara mencapainya [4]. Maka dari itu, *autonomous mobile robot* memiliki sistem yang

navigasi cerdas agar mampu mengerjakan tugasnya dan terhindar dari tabrakan dengan berbagai macam jenis halangan yang ada di lingkungan sekitarnya [5].

Sistem navigasi cerdas tersebut dikenal sebagai *path planning*, yaitu suatu jalan yang berasal dari *waypoint-waypoint* yang terdapat diantara titik awal dan tujuannya. Selama ini para peneliti mengkategorikan metode *path planning* berdasarkan kondisi dari *environment*, yaitu statis dan dinamis [6]. Statis *environment* merupakan suatu tempat yang berisi objek tidak bergerak (diam), dimana posisi seluruh objek tetap dan diketahui. Sedangkan dinamis *environment* merupakan kebalikannya, dimana terdapat objek bergerak yang posisinya berubah dari waktu ke waktu [7]. Permasalahan *path planning* ini dapat diselesaikan dengan berbagai macam metode, baik metode klasik maupun heuristic [8]. Salah satu metode heuristic yang banyak digunakan yaitu algoritma genetika (GA). Hal ini dikarenakan kemampuannya yang dapat mencari solusi optimal. Adapun optimal yang dimaksud yaitu memenuhi beberapa kriteria seperti jarak, energi, keselamatan, dan waktu [9]. Semakin banyak kriteria yang digunakan semakin baik lintasan yang diperoleh, namun hal ini berimbas ke waktu komputasi algoritma yang akan semakin lama. Oleh karena itu, agar mendapatkan hasil yang optimal dengan memenuhi banyak kriteria tersebut, berbagai macam penelitian telah dilakukan dengan memodifikasi ataupun menggabungkan GA dengan metode lain [10].

Pada [11], GA digunakan untuk menyelesaikan permasalahan *path planning* pada statis *environment*. Modifikasi pada GA dilakukan dengan memberikan syarat tambahan pada proses *crossover*, dimana nilai *fitness* dari keturunan yang dihasilkan akan dibandingkan terlebih dahulu dengan nilai *fitness* induknya. Jika nilai *fitness* keturunan tidak lebih baik daripada induknya, maka keturunan akan diabaikan dan induk yang akan digunakan pada proses selanjutnya yaitu mutasi gen. Hal ini dilakukan dengan tujuan agar mendapatkan individu baru yang berkualitas. Sedangkan kriteria optimal yang harus dicapai dari penelitian ini yaitu lintasan terpendek dan bebas tabrakan dengan halangan. Akan tetapi, *path planning* penelitian ini hanya bertujuan untuk menghindari halangan statis saja, sehingga metode GA yang diajukan kurang teruji ketangguhannya. Sementara itu, [12] juga menggunakan GA untuk menyelesaikan permasalahan *path planning* pada statis *environment* dengan menggunakan dua pengaturan kondisi awal yang berbeda. Hasil simulasi menunjukkan bahwa meskipun dengan pengaturan kondisi awal yang berbeda, GA mampu menghasilkan lintasan yang optimal.

Untuk permasalahan mengenai penghindaran halangan dinamis diselesaikan pada [13] dengan menggunakan tiga kriteria optimal yaitu jalur bebas tabrakan dengan halangan, jalur terpendek, dan *smoothness* jalur. Penelitian ini juga menggunakan GA untuk menyelesaikan permasalahannya dengan menerapkan teknik *coding* tertentu yaitu membagi area pencarian menjadi beberapa *node*. Tujuannya untuk mempersempit ruang pencarian dan mempercepat waktu komputasi GA. Namun sayangnya meskipun sudah membagi ruang pencarian menjadi beberapa titik, penelitian ini masih memperhitungkan semua kemungkinan munculnya halangan dinamis mulai dari titik awal sampai ke titik akhir (titik target). Sedangkan pada [14] menggunakan mobil sebagai *plant* untuk menghindari halangan dinamis. Penelitian ini fokus pada perilaku mobil di jalan raya untuk menghindari tabrakan dengan mobil lainnya ketika melakukan alternatif demi mencapai kinerja yang optimal dan kenyamanan pengemudi. Manuver mobil terdiri dari beberapa tipe antara lain *lane changing*, *lane keeping*, *overtaking*, *intersection crossing*, serta *turn left* ataupun *turn right*. Lintasan dinamis dibuat dengan tujuan agar dapat melakukan manuver tipe tertentu untuk menghindari tabrakan.

Oleh karena itu, pada penelitian ini akan menggunakan metode GA termodifikasi *crossover* untuk membuat algoritma *dynamic path planning* pada *autonomous mobile robot* untuk menghindari halangan dinamis yang mungkin muncul pada *moving area sensor* agar dapat menghemat waktu komputasi algoritma dan membatasi ruang pencarian berdasarkan algoritma pendeteksian sensor dan prediksi trayektori halangan dinamis.

2. Metode

Penelitian ini menggunakan model matematis *autonomous mobile robot* pada [15] serta metode GA yang memodifikasi proses *crossover* dimana nilai *fitness* keturunan hasil dari *crossover* akan dibandingkan dengan induknya. Adapun aturan yang dibuat adalah sebagai berikut:

1. Jika nilai *fitness* keturunan hasil dari *crossover* lebih baik daripada induknya, maka keturunan tersebut akan digunakan pada proses selanjutnya yaitu mutasi.
2. Jika nilai *fitness* keturunan tidak lebih baik daripada induknya, maka keturunan tersebut diabaikan dan induknya akan digunakan pada proses mutasi

Metode GA dibagi menjadi dua yaitu GA statis [11] dan GA dinamis, dimana algoritma GA statis digunakan untuk menghasilkan lintasan utama sedangkan GA dinamis dirancang untuk

menghasilkan lintasan alternatif ketika terdeteksi halangan dinamis oleh *moving area sensor*.

Lintasan alternatif atau lintasan dinamis yang dihasilkan berdasarkan eror yang terdeteksi antara lintasan utama dan halangan dinamis. Ketika halangan dinamis muncul disepanjang lintasan utama dan berpeluang menabrak *autonomous mobile robot*, maka *autonomous mobile robot* yang telah diberi algoritma genetika akan men-*generate* lintasan baru dalam rangka menghindari halangan tersebut. Lintasan baru yang dihasilkan tidak boleh jauh menyimpang dari lintasan utama karena diharapkan setelah menghindari halangan dinamis tersebut, *autonomous mobile robot* kembali mengikuti lintasan utama untuk menuju ke titik target. Data masukan yang digunakan untuk membuat lintasan alternatif yaitu informasi lintasan utama yang berupa *waypoints*, titik awal dan target, serta posisi halangan statis dan dinamis. Adapun tahapan-tahapan dalam merancang lintasan alternatif menggunakan metode GA dinamis adalah seperti berikut ini,

1. Inisialisasi Populasi
Populasi awal yang dibutuhkan tidak sebanyak pada GA statis. Hal ini dikarenakan dibutuhkan waktu eksekusi algoritma yang cepat sehingga *autonomous mobile robot* dapat menghindari tabrakan dengan halangan dinamis secepat mungkin. Oleh karena itu jumlah populasi yang digunakan kali ini yaitu sebanyak dua kromosom.
2. Verifikasi *Obstacles Avoidance*
Kromosom hasil pembangkitan populasi awal tersebut kemudian diverifikasi apakah menabrak halangan statis atau tidak.
3. Seleksi dan Fungsi *Fitness*
Setelah itu kromosom yang sudah dipastikan tidak menabrak halangan statis maupun dinamis akan melalui proses *crossover*.
4. Modifikasi *Crossover* dan *Roulette Wheel*
Proses *crossover* pada GA dinamis ini sama dengan pada GA statis, hanya saja kromosom yang digunakan lebih sedikit karena untuk mempersingkat waktu eksekusi algoritma.
5. Mutasi
Hasil *crossover* selanjutnya melewati proses mutasi untuk memberikan variasi kromosom dan diharapkan mendapatkan kromosom yang lebih baik lagi.
6. Verifikasi *Obstacles Avoidance*
Kromosom hasil mutasi selanjutnya diverifikasi apakah menabrak halangan atau tidak. Jika kromosom tersebut tidak menabrak halangan maka dilanjutkan ke proses selanjutnya, namun jika kromosom tersebut masih menabrak halangan maka kembali lagi ke proses *crossover*.
7. Evaluasi Fungsi *Fitness*

Setelah mendapatkan kromosom generasi baru hasil dari operator genetika *crossover* dan mutasi, selanjutnya hitung nilai fungsi *fitness* kromosom tersebut, lalu diurutkan sesuai dengan nilai *fitness*-nya. Pilih kromosom dengan nilai *fitness* maksimal sebagai solusi terbaik dari permasalahan.

8. Jumlah iterasi
Solusi optimal dari permasalahan diperoleh setelah jumlah iterasi terpenuhi.

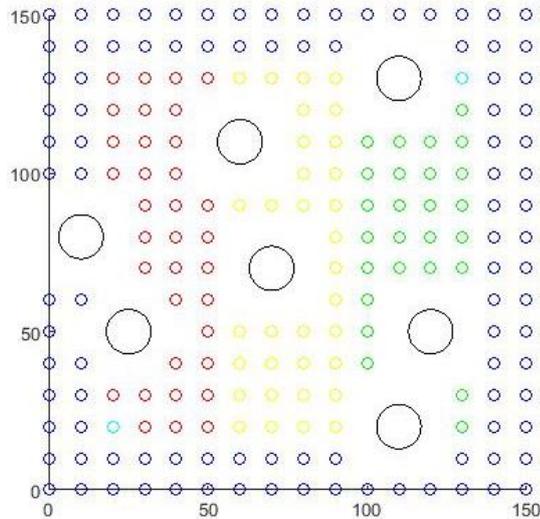
Pada penelitian ini digunakan suatu mekanisme sistem yang seolah-olah berfungsi sebagai sensor yang akan mendeteksi halangan dinamis yang memiliki kemungkinan bertabrakan dengan *autonomous mobile robot* di sepanjang lintasan utama. Langkah pertama yaitu membangkitkan halangan dinamis dengan memberikan nilai kecepatan konstan pada halangan tersebut. Selanjutnya setelah melakukan pembangkitan halangan dinamis, maka dirancang algoritma pendeteksian halangan oleh sensor. Masukan pada sensor berupa koordinat halangan dinamis, koordinat *autonomous mobile robot*, dan sudut yang dibentuk antara *autonomous mobile robot* dan halangan dinamis. Sensor hanya akan mendeteksi halangan sesuai area tertentu saja, yang mana disebut dengan *moving area sensor*. Sudut *beam* pada *moving area sensor* yang didesain yaitu 45° dengan jarak jangkauan *beam* 2dm.

Halangan dinamis akan terdeteksi oleh *moving area sensor* jika jarak *autonomous mobile robot* dengan halangan tersebut lebih kecil dan sama dengan jangkauan *beam* serta sudut yang terbentuk antara *autonomous mobile robot* dan halangan dinamis lebih kecil dan sama dengan sudut *beam*. Akan tetapi, meskipun masuk ke area cakupan *moving area sensor*, halangan dinamis tersebut belum tentu menabrak *autonomous mobile robot*. Oleh karena itu harus dihitung titik perpotongan antara lintasan *autonomous mobile robot* dan lintasan halangan dinamis untuk mengetahui titik kemungkinan tabrakan antara *autonomous mobile robot* dan halangan tersebut.

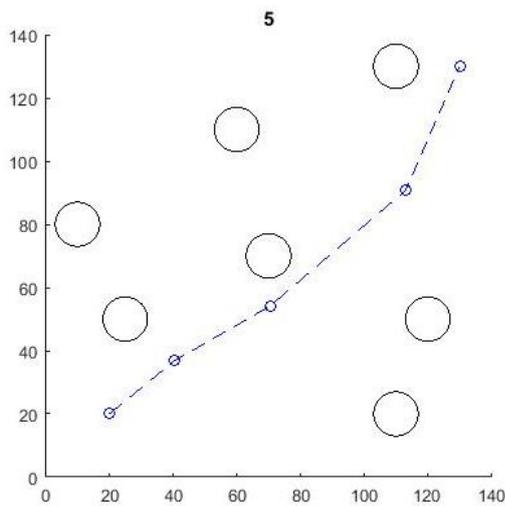
3. Hasil dan Pembahasan

Lintasan utama yang diperoleh dari GA statis dijadikan masukan pada simulasi GA dinamis. Simulasi dilakukan dengan memberikan halangan dinamis yang melewati lintasan utama. Kemudian halangan dinamis tersebut dideteksi oleh *moving area sensor autonomous mobile robot*. Simulasi dilakukan dengan tujuan untuk membuktikan bahwa *moving area sensor* memberikan penyimpangan lintasan *minimum*. Gambar 1 merupakan *environment* yang digunakan pada simulasi dimana di dalamnya terdapat 7 halangan statis dengan koordinat bidang XY yaitu (10,80), (25,50), (60,110), (70,70), (110,20), (110,130), dan (120,50). Lintasan utama kemudian di-*generate* pada *environment*

tersebut melalui proses GA statis sehingga diperoleh lintasan utama. *Autonomous mobile robot* akan melewati lintasan tersebut dan ketika *moving area sensor* mendeteksi adanya halangan dinamis di depannya, maka akan di-*generate* lintasan alternatif atau lintasan dinamis untuk menghindari tabrakan dengan halangan dinamis tersebut.



Gambar 1. Statis Environment.

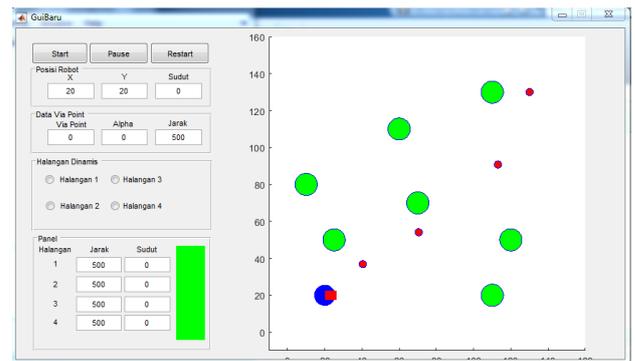


Gambar 2. Lintasan utama dari hasil GA statis.

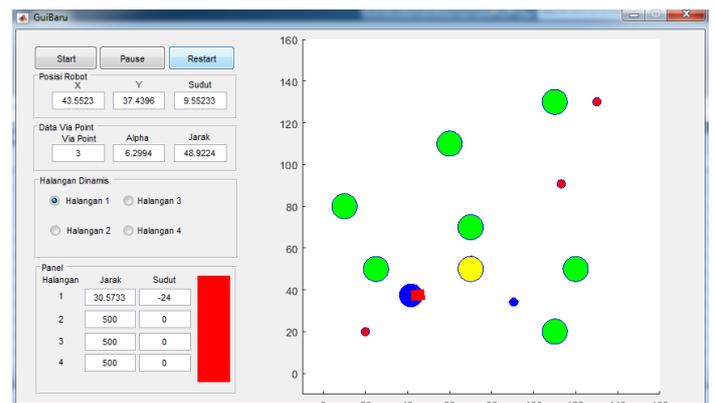
Sedangkan Gambar 2 merupakan lintasan utama yang telah di-*generate* oleh GA statis yang telah didesain berdasarkan informasi data halangan statis. Pada lintasan utama tersebut terdapat lima *waypoint* yaitu pada titik koordinat awal (20,20), koordinat kedua (40.44,36.82), koordinat ketiga (70.45,54.11), koordinat keempat (113.06,90.78), dan koordinat akhir atau tujuan (130,130). Gambar 3 menunjukkan simulasi yang dibuat pada GUI Matlab dengan spesifikasi *environment* yang sama seperti pada Gambar 2. Halangan statis merupakan lingkaran yang berwarna hijau, sedangkan lingkaran berwarna biru-merah merupakan *autonomous*

mobile robot, dan lima titik yang berwarna merah merupakan *waypoint* yang harus dilewati. Disebelah kiri GUI terdapat keterangan mengenai posisi robot, data *waypoint* dan data halangan dinamis. Informasi posisi robot berupa koordinat XY yang dilewati. Data *viapoint* menampilkan *waypoint* mana saja yang dilewati, *heading error* dari *beam autonomous mobile robot* serta jarak antar *waypoint*. Informasi mengenai halangan dinamis menampilkan jarak antara halangan dinamis terhadap robot. Warna hijau pada data halangan dinamis akan berwarna merah jika *moving area sensor robot* mendeteksi halangan dinamis berdasarkan jangkauannya seperti pada Gambar 4.

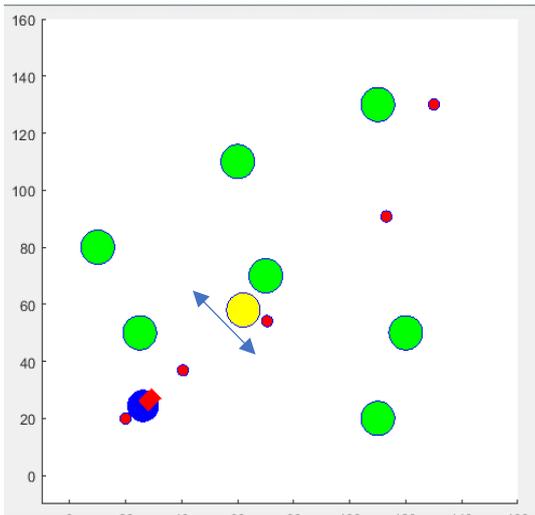
Terdapat sebuah halangan dinamis yang berwarna kuning di dalam *environment* yang selalu bergerak bolak balik seperti yang ditunjukkan arah panah pada Gambar 5. Ketika halangan dinamis tersebut melewati lintasan utama dan terdeteksi oleh *moving area sensor*, maka akan di-*generate* *waypoint* baru yang membentuk lintasan alternatif untuk menghindari tabrakan dengan halangan dinamis tersebut seperti ilustrasi pada Gambar 6. *Moving area sensor* pada *autonomous mobile robot* mampu mendeteksi halangan dinamis tersebut berdasarkan perhitungan jarak halangan dinamis terhadap robot seperti data grafik jarak halangan dinamis terhadap *autonomous mobile robot* yang dapat dilihat pada Gambar 7.



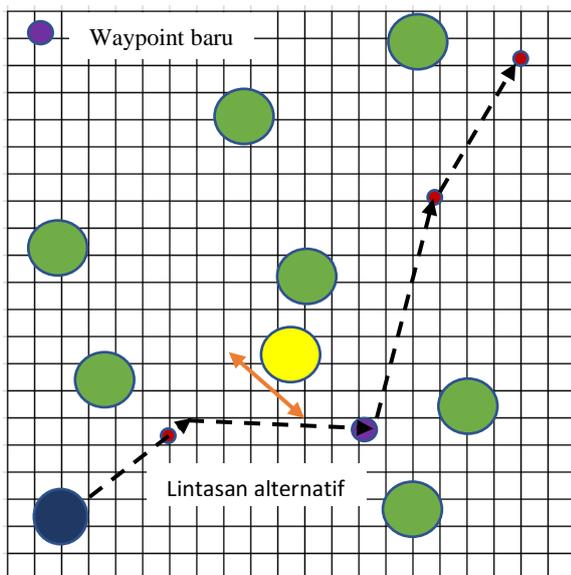
Gambar 3. Tampilan simulasi GUI Matlab.



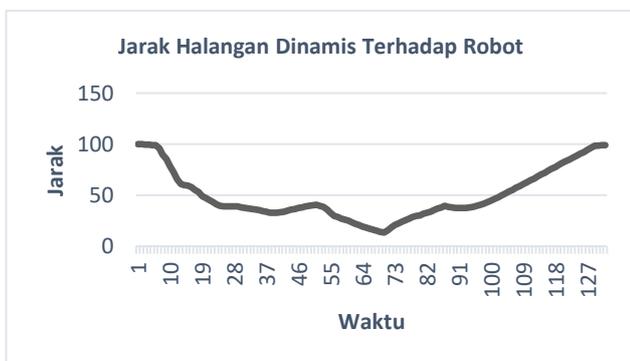
Gambar 4. *Moving area sensor* mendeteksi halangan statis.



Gambar 5. Environment diberi halangan statis.



Gambar 6. Environment yang dilengkapi dengan halangan dinamis.



Gambar 7. Jarak halangan dinamis terhadap autonomous mobile robot.

Kromosom yang di-generate hanya berjumlah dua sehingga lebih cepat dalam mengambil keputusan untuk menghindari tabrakan dengan

halangan dinamis. Berdasarkan Tabel 1, gen yang berubah adalah gen ketiga (WpX3 dan WpY3) karena halangan dinamis berada di dekat *waypoint* ketiga. Sedangkan *waypoint* pertama dan kelima tetap sama yaitu titik *start* (20, 20) dan titik *goal* (130,130).

Tabel 1. Kromosom baru untuk lintasan dinamis.

	Kromosom 1	Kromosom 2
WpX2	40.44	40.44
WpY2	36.82	36.82
WpX3	50.45	90.45
WpY3	74.11	34.11
WpX4	113.06	113.06
WpY4	90.78	90.78
Nilai <i>Fitness</i>	0.8144	0.8750

Sementara itu, gen ketiga yang awalnya (70.45,54.11) berubah menjadi (90.45,34.11) karena berdasarkan nilai *fitness* dari kedua kromosom diperoleh bahwa kromosom terbaik adalah kromosom kedua dengan nilai *fitness* 0.8750 dengan gen ketiga pada koordinat *waypoint* (90.45,34.11).

4. Kesimpulan

Sistem navigasi cerdas berupa *path planning* yang didesain bertujuan untuk menghindari tabrakan, baik dengan halangan statis maupun dengan halangan dinamis. Metode yang digunakan yaitu modifikasi *crossover* algoritma genetika. Modifikasi operator genetika *crossover* dengan membuat peraturan tertentu yaitu nilai *fitness* keturunan hasil *crossover* harus dibandingkan terlebih dahulu dengan nilai *fitness* induknya. Apabila nilai *fitness* keturunan lebih baik dari nilai *fitness* induk maka keturunan digunakan pada proses selanjutnya. Namun jika nilai *fitness* induk lebih baik daripada nilai *fitness* keturunan, maka keturunan diabaikan. Berdasarkan simulasi yang telah dilakukan, GA statis berhasil membuat lintasan utama sedangkan GA dinamis mampu membuat lintasan alternatif. Kromosom terbaik yang dihasilkan oleh GA dinamis yaitu kromosom kedua dengan nilai *fitness* 0.8750 dengan gen ketiga yang mengalami perubahan koordinat (90.45, 34.11). Hal ini menunjukkan *moving area sensor* mampu mendeteksi halangan dinamis pada jarak tertentu sehingga *autonomous mobile robot* mampu *generate waypoint* baru untuk membuat lintasan alternatif agar tidak menabrak halangan dinamis.

Daftar Pustaka

- [1] Mohammad Ali H. Eljinini, Ahmad Tayyar, "Collision-free Random Paths between Two Points", *International Journal of Intelligent Systems and Applications(IJISA)*, Vol.12, No.3, pp.27-34, 2020.
- [2] J. Arpit, K. Wojciech, "Design of Autonomous Mobile Robot for Cleaning in the Environment

- with Obstacles". *Applied Sciences*. Vol. 11, pp. 8076. 2021.
- [3] M. B. Alatis and G. P. Hancke, "A Review on Challenges of Autonomous Mobile Robot and Sensor Fusion Methods," in *IEEE Access*, vol. 8, pp. 39830-39846, 2020.
- [4] L. Tang, S. Yuta, "Indoor navigation for mobile robots using memorized omni-directional images and robot's motion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 1, Oct. 2002, pp. 269–274 vol.1.
- [5] M. Gao, J. Xu, J. Tian and H. Wu, "Path Planning for Mobile Robot Based on Chaos Genetic Algorithm," *2008 Fourth International Conference on Natural Computation*, 2008, pp. 409-413, 2008.
- [6] K. H. Sedighi, K. Ashenayi, T. W. Manikas, R. L. Wainwright and Heng-Ming Tai, "Autonomous local path planning for a mobile robot using a genetic algorithm," *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, 2004, pp. 1338-1345 Vol.2, 2004.
- [7] H. L. A. Ramirez-Serrano and G.C. Pettinaro, "Mobile robot localization in quasi-dynamic environments," *Industrial Robot: An International Journal*, vol. 35, pp. 246-258, 2008.
- [8] D. R. P. Prases K. Mohanty, "Controlling the motion of an autonomous mobile robot using various techniques: a review," *Journal of Advanced Mechanical Engineering*, vol. 1, pp. 24-39, 2013.
- [9] O. Hachour, "Path planning of autonomous mobile robot," *International journal of systems applications, engineering & development*, vol. 2, no. 4, pp. 178–190, 2008.
- [10] S. Katoch, S.S. Chauhan, & V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimed Tools Appl.*, vol. 80, pp. 8091–8126, 2021.
- [11] N. S. Utami, A. Jazidie and R. E. A. Kadier, "Path Planning for Differential Drive Mobile Robot to Avoid Static Obstacles Collision using Modified Crossover Genetic Algorithm," *2019 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2019, pp. 282-287, 2019.
- [12] S. Choueiry, M. Owayjan, H. Diab and R. Achkar, "Mobile Robot Path Planning Using Genetic Algorithm in a Static Environment," *2019 Fourth International Conference on Advances in Computational Tools for Engineering Applications (ACTEA)*, 2019, pp. 1-6, doi: 10.1109/ACTEA.2019.8851100.
- [13] Pu Shi, Yujie Cui, "Dynamic path planning for mobile robot based on Genetic Algorithm in unknown environment," in *Chinese Control and Decision Conference*. IEEE, pp. 4325-4329, 2010.
- [14] Z. Sumin, W. Deng, Z. Qingrong, S. Hao, L. Bakhtiar, "Dynamic trajectory planning for vehicle autonomous driving" in *International Conference on System, Man, and Cybernetics*. IEEE, pp. 4161-4166, 2016.
- [15] R. D. A. A. Hatab, "Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework," *Adv. Robot. Autom.*, vol. 02, no. 02, 2013.