

Original scientific paper

Received: 19 November 2022.

Revised: 21 November 2022.

Accepted: 21 November 2022.



## Algoritma Genetika Sebagai Solusi Permasalahan Persamaan Linear Matematika

Lufan Anas Zahir<sup>1\*</sup> 

<sup>1</sup> Universitas Tulungagung, e-mail: [lufananas@gmail.com](mailto:lufananas@gmail.com)

**Abstract:** The Genetic Algorithm optimization method can be used to find efficiency from real problems that can be brought to mathematical problems. This optimization method consists of solving the function by generating initial random numbers which are then carried out by crossover and mutation. The completion of this algorithm uses a rate at random numbers and a crossover rate so that each element of the initial (individual) function can be cross-scanned so that various mathematical problems such as quadratic, linear, geometric, and others can be solved. Many tools or software assistance are available nowadays, but this research was done manually, meaning that researchers did not use existing software so that the results of crossover and mutation scans can be known with certainty. The results showed that the difference in the value of the mutasu rate and the crossover rate had a major role in influencing the completion of the optimization of the Genetic Algorithm. If  $rate \geq 1$ , it can cause many initial individual elements to be lost and result in no solution being found. But in contrast to the size of the population, the larger the population of the generated functions, the higher the chances of the Genetic Algorithm in solving the problem

**Keywords:** crossover, genetic algorithm, mutation, optimization, system of linear equations

**Abstrak:** Metode optimasi Algoritma Genetika dapat digunakan untuk mencari efisiensi dari permasalahan riil yang dapat di bawa ke permasalahan matematika. Metode optimasi ini terdiri dari penyelesaian fungsi dengan generate bilangan random awal yang kemudian dilakukan *crossover* dan mutasi. Penyelesaian algoritma ini menggunakan rate pada bilangan random dan rate *crossover* sehingga setiap elemen dari fungsi (individu) awal dapat dilakukan pindai silang sehingga dengan bervariatifnya permasalahan matematika seperti kuadrat, linear, geometri, dan lain-lain dapat diselesaikan. Banyak sekali alat atau bantuan perangkat lunak yang tersedia di zaman ini, namun pada penelitian ini dilakukan secara manual, artinya peneliti tidak menggunakan perangkat lunak yang ada sehingga hasil dari pindai silang (*crossover*) dan mutasi dapat diketahui dengan pasti. Hasil penelitian menunjukkan bahwa, perbedaan nilai rate mutasi dan rate *crossover* memiliki peranan besar dalam mempengaruhi penyelesaian optimalisasi Algoritma Genetika. Jika  $rate \geq 1$ , maka dapat menyebabkan banyaknya elemen individu awal yang hilang dan mengakibatkan solusi tidak ditemukan. Namun berbeda dengan besarnya populasi, semakin besar populasi dari fungsi yang dibangkitkan, semakin tinggi pula peluang Algoritma Genetika dalam menyelesaikan permasalahan.

**Kata Kunci:** algoritma genetika, *crossover*, *mutasi*, optimasi, sistem persamaan linear.

\*Corresponding author: [lufananas@gmail.com](mailto:lufananas@gmail.com)



© 2022 by the authors. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## Pendahuluan

Generasi Z merupakan generasi yang dihadapkan dengan perubahan atau transisi teknologi modern sesuai dengan perkembangan informasi yang serba digital. Banyak sekali permasalahan-permasalahan dalam kehidupan yang dapat dibantu terselesaikan dengan perkembangan teknologi saat ini. Permasalahan yang ada tersebut juga berhubungan erat dengan kehidupan sehari-hari seperti kasus dalam pembelajaran khususnya matematika. Contoh dalam lingkungan pendidikan, yaitu permasalahan dalam matematika, yang sampai hari ini mejadi momok pada anak sekolah.

Pemahaman tentang matematika dan penyelesaiannya membutuhkan waktu proses yang lama. Dari hal tersebut, ada beberapa cara penyelesaian permasalahan pada matematika, salah satunya dengan menggunakan suatu metode yaitu algoritma genetika (Retta et al., 2020). Sebagai contoh apabila digunakan untuk mencari permasalahan linier kuadrat, garis ataupun kuadrat memudahkan dalam mencari penyelesaian. Algoritma genetika sendiri didasarkan pada kehidupan gen manusia yang berkembang setiap saat, dalam arti kata bawah penyelesaian menggunakan metode algoritma genetika yaitu membutuhkan nilai awal yang disebut individu, dengan pengertian lain semakin banyak individu awal yang digunakan semakin banyak punya kesempatan untuk menyelesaikan permasalahan. Dalam teori algoritma genetika, dibatasi denganya adanya batasan-batasan seperti pada kehidupan sehari-hari, dimana batasan tersebut dapat digunakan sebagai nilai atau data yang membatasi supaya individu awal tidak melebihi batas (Ardiansyah et al., 2022). Algoritma genetika disarankan pada kehidupan manusia yang berkembang biak dengan diselingi adanya gangguan pada kehidupan sehingga menjadi lebih baik, dimana gangguan tersebut disebut dalam algoritma dengan kata sederhana yaitu mutasi diterapkan di metode algoritma genetika disebut dengan mutasi (Caprio et al., 2022). Algoritma genetika menghasilkan populasi lengkap dari titik penjawab. Setiap titik diuji secara mandiri dan untuk menetapkan populasi baru, termasuk poin yang dimodifikasi, poin-poin yang ada dapat diuji (Cowdrey et al., 2018). Algoritma genetika mempertimbangkan banyak titik secara bersamaan dan karakteristik ini menyesuaikan dengan lebih paralel prosesor pada penyelesaian suatu masalah, karena mempertimbangkan setiap titik membutuhkan beberapa perhitungan termasuk fungsi target diferensiasi, dll (Anjasmara, 2019). Dalam Algoritma ini, operan dan mekanisme yang berbeda di implementasikan, yang dijelaskan di sini untuk penemuan algoritma genetika sebagai algoritma mengoptimalkan terutama berdasarkan simulasi evolusi alami dan mereka belum didasarkan pada teori matematika yang kuat. Teknik ini menggunakan genetika alami, metode statistik dengan mencoba menghasilkan suatu kesimpulan untuk mengoptimalkan fungsi pada algoritma genetika (Sugeha et al., 2019).

Saat ini, ada beberapa metode umum untuk mengoptimalkan pencarian yang meliputi: analitis enumerasi dan acak (Hanief & Jepriana, 2020). Teknik analisis telah dipelajari dengan ekstensif. Mereka terbagi menjadi kategori langsung dan tidak langsung (Haruna et al., 2015). Metode tidak langsung mendapatkan yang dioptimalkan poin dengan memecahkan satu set persamaan non-linear yang dicapai dengan menempatkan fungsi target diferensiasi sama ke nol. Metode langsung dengan mencari titik yang dioptimalkan oleh fungsi target dan bergerak masuk menuju arah gradien. Kedua teknik ini dioptimalkan dan dikembangkan tetapi ada beberapa bukti dijelaskan kemudian menunjukkan bahwa mereka tidak mampu mengoptimalkan dengan benar (Heinonen & Jusko, 2016).

Kedua teknik memiliki titik optimal yang berbeda. Titik optimal yang dicari teknik ini adalah titik terbaik lingkungan dari titik sekarang. Sebagai contoh, jika kita memiliki fungsi dengan dua puncak, salah satunya adalah salah dan nyata, kemudian mencapai puncak yang lebih kecil berakibat kita terhalang dari titik yang sebenarnya. Karena itu, sebaiknya kita gunakan metode lain, seperti restart acak dengan loop untuk mendapatkan respons terbaik (Manamping, 2015). Teknik analisis bergantung dengan adanya diferensiasi. Bahkan jika kita menggunakan angka perkiraan diferensiasi dalam perhitungan, akan menghasilkan solusi optimal dengan kekurangan yang intensif. Oleh sebab itu jelaslah bahwa teknik berdasarkan batas kelanjutan dan turunan tidak cocok untuk setiap kasus.

Meskipun algoritma pencarian acak memiliki lebih banyak opsi dan melakukan lebih baik dari pada dua metode lainnya, tetapi hasil yang diinginkan tidak sesuai, ketika domain responsif besar. Tetapi genetika memiliki semua algoritma keuntungan dari metode yang disebutkan, serta tidak memiliki kekurangan, dan menggunakan kreativitas, yang sebagian besar terbukti dalam sikap manusia. Algoritma Genetika juga mendapat bantuan fungsi probabilitas sebagai alatnya. Karenanya perbedaan teknik lain dengan algoritma genetika dibagi menjadi beberapa kategori (Krisnandi & Agung, 2017) yaitu algoritma genetika dengan satu set pengkodean, algoritma genetika dengan sekumpulan titik, yang memulai pencarian tidak hanya dari satu titik, algoritma genetika menggunakan informasi akhir (fungsi target), bukan dari turunan atau perangkat tambahan lainnya. Algoritma genetika dalam menyelesaikan menggunakan hukum probabilitas dan bukan hukum tetap.

Dalam penelitian sebelumnya metode penyelesaian banyak menggunakan pemrograman linier yang merupakan satu model pemrograman matematika, yang dianggap sebagai alat yang diakui yang menyebabkan jutaan dolar tabungan untuk unit perdagangan besar di kota-kota maju di dunia dan penggunaannya dengan cepat maju di bagian lain dari komunitas solusi terbaik dan optimal (Kowalski et al., 2021). Dengan maksud lain, dalam kasus melakukan kegiatan terkait dengan memanfaatkan sumber terbatas, biasanya diperlukan untuk mereka, sumber daya alokasi dan karenanya penentuan tingkat aktivasi harus dipertimbangkan (Kusaik, 1989). Memang, dapat dikatakan bahwa ada puluhan buku dan artikel yang diterbitkan membahas aplikasi dan sekitar  $\frac{1}{4}$  dari perhitungan praktis total untuk komputer terkait dengan pemrograman linier dan derivasi. Singkatnya, pemrograman linear berhubungan dengan masalah pembatasan sumber daya antara kegiatan kompetitif untuk menemukan Model matematis digunakan dalam pemrograman linier untuk menjelaskan masalah yang dipertimbangkan.

Kata "linear" menunjukkan bahwa semua relasi permodelan matematis perlu menjadi fungsi linear. Hal tersebut mempunyai opini bahwa pemrograman linier digunakan mengoptimalkan supaya mempunyai hasil lebih baik daripada hasil alternatif. Spesifikasi pada model pemrograman linier adalah bahwa fungsi target dan kendala di dalamnya bersifat linier dan linearitas dari beberapa model dapat dibenarkan sesuai dengan karakteristik fisik. Seperti disebutkan, teknik ini dengan gradien fungsi untuk mencapai target dan solusi optimal, yang pada gilirannya membutuhkan turunan dan kontinuitas fungsi di semua pengoptimalan, karena jika fungsi tidak memiliki karakteristik itu, kita akan menghadapi kesulitan dalam memperoleh gradien, sedangkan teknik algoritma genetika tidak memiliki ketergantungan terhadap derivabilitas dan melanjutkan fungsi dan mulai dengan memilih beberapa titik di permasalahan, serta mencari jawaban optimal dengan menggunakan poin utama. Selain itu, teknik ini tidak perlu perhitungan tambahan, seperti diferensiasi dan memperoleh gradien dan juga mengganti nilai dalam fungsi yang rumit (Lamini et al., 2017). Sementara itu solusi masalah yang berbeda oleh kedua metode, Secara empiris, menunjukkan bahwa jawabannya diperoleh dalam waktu yang lebih singkat oleh algoritma genetika.

## Metode

Algoritma genetika yang diterapkan dalam penelitian bertujuan untuk mencari penyelesaian pada permasalahan matematika. Hasil dari optimasi penyelesaian tersebut berupa solusi dengan bermacam-macam nilai *crossover* dan mutasi (Oktarina & Ajjah, 2019). Menurut Mauluddin (2018), Ada beberapa langkah harus diperhatikan didalam melakukan proses algoritma genetika untuk menghasilkan sebuah solusi yang paling optimum, diantaranya:

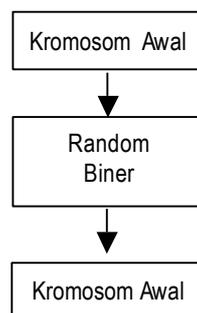
### 1. Menentukan Persamaan Fungsi dan Menentukan Fitness.

Langkah kesatu penelitian dengan membangkitkan populasi awal dengan membangkitkan beberapa individu yang bangkitkan secara acak yang memiliki prosedur tertentu. Ukuran pada populasi individu dilihat pada permasalahan yang dituju dengan beberapa jenis kromosom pada algoritma genetika.

Setelah size populasi selesai dihasilkan, kemudian dilakukan inialisasi terhadap beberapa kromosom didalam populasi tersebut. Setelah itu yang dilakukan dalam pemecahan masalah dengan algoritma genetika adalah menentukan fungsi, dimana merupakan contoh dalam permasalahan matematika. Setelah fungsi ditentukan maka selanjutnya yaitu menentukan fitness. Penentuan fitness sangat penting pada proses algoritma genetika. Nilai fitness adalah suatu ukuran dalam nilai optimum yang baik dan merupakan solusi pada satu individu, atau dengan arti lain nilai fitness dapat digunakan dalam menyatakannilai dari fungsi untuk mencapai tujuan (Permata et al., 2018). Untuk menentukan nilai fitness maka penelitian ini, menggunakan persamaan dari fungsi, dalam arti kata jika kita menghendaki nilai minimum maka nilai terkecil yang diambil, sedangkan jika menghendaki nilai maksimum, maka nilai terbesar yang diambil.

## 2. Mendefinisikan Bentuk Kromosom

Menentukan individu pada metode algoritma genetika dengan menentukan kromosom, kromosom yang digunakan adalah kromosom biner 8 bit. Model kromosom 8 bit merupakan kromosom standart dari algoritma genetika (yang terdiri dari bilangan biner yang disusun gen 0 dan 1). Skema pengkodean ditentukan oleh algoritma genetika dengan diinisialisasi pada sebuah populasi dengan beberapa kromosom.

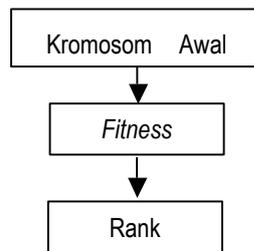


Gambar 1. Proses Pembangkitan Kromosom

Gen dari algoritma genetika akan mengisi pada masing-masing kromosom dan dibangkitkan secara random. Masing-masing kromosom mempunyai pengkodean dari individu dengan memiliki nilai fitness yang ditentukan, dan kemudian menghasilkan populasi baru hasil pembentukan dengan menggunakan mekanisme pada seleksi secara alamiah, yaitu dengan random individu-individu awal dengan proporsional terhadap nilai fitness yang telah ditentukan, dan genetika secara alamiah dengan arti lain yaitu pindah silang atau dengan kata lain yaitu crossover dan mutasi. Pada umumnya algoritma genetika mempunyai metode yaitu adalah pergantian skema pada populasi dan disebut dengan generational replacement, dengan kata lain, kromosom pada suatu generasi dengan digantikan oleh pindah silang serta mutasi. Kromosom ditentukan dengan dibatasi oleh nilai individu awal dan individu akhir dari suatu random.

## 3. Menentukan Seleksi.

Seleksi dilakukan setelah masing-masing fitness dari kromosom didapatkan. Fitness yang memiliki nilai paling besar memiliki probabilitas paling besar untuk menjadi individu terbaik. Hasil individu dari seleksi digunakan untuk menghitung probabilitas individu mana yang keluar sebagai individu crossover. Seleksi menggunakan roulette wheel. Metode roulette wheel adalah metode yang sederhana serta paling sering digunakan di algoritma genetika serta banyak juga dikenal mempunyai nama stochastic dengan sampling. Pada metode roulette wheel, individu-individu awal dipetakan di suatu segmen garis dan beraturan sehingga tiap segmen pada individu awal mempunyai ukuran yang sama dengan ukuran fitnessnya.

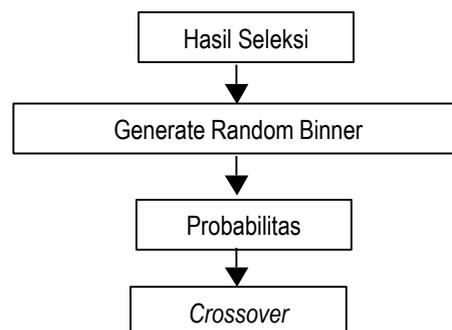


Gambar 2. Flowcart Seleksi

Sebuah bilangan pada random individu awal akan seleksi dan dibangkitkan serta individu awal yang memiliki fitness dalam kawasan yang ditentukan pada bilangan random akan diseleksi. Proses seleksi ini diulang sehingga memperoleh sejumlah individu seperti yang diharapkan sesuai fitness. Beberapa langkah dalam melakukan seleksi yaitu hitung peluang probabilitas untuk tiap individu, bangkitkan data random sebanyak individu dan yang ketiga, tukar posisi individu dengan nilai random terdekat. Seleksi yang digunakan menggunakan persentasi terbanyak individu yang keluar pada data.

#### 4. Menentukan Crossover.

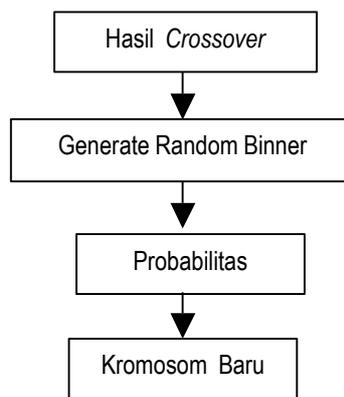
Hasil individu dari kromosom seleksi akan dilakukan *cross-over*, dimana proses yang akan dilakukan adalah mengkombinasikan dua individu hasil seleksi, yang diharapkan dari hasil tersebut mendapatkan individu yang mempunyai fitness lebih tinggi dari sebelumnya. Tidak semua pasangan individu hasil seleksi mengalami proses crossover, jumlah pasangan individu hasil seleksi yang akan diproses melalui crossover ditentukan pada parameter probabilitas crossover. Proses *crossover* membutuhkan dari dua induk individu hasil seleksi untuk membentuk kromosom sebagai individu baru. Pindah silang dari hasil crossover menghasilkan individu, yang nanti akan diteruskan pada proses mutasi di algoritma genetika. Probabilitas crossover menentukan pada proses persilangan, yang dengan kata lain apabila nilai probabilitas lebih dari satu, maka akan menyebabkan hilangnya individu baru yang mempunyai gen lebih baik (Septyanto et al., 2018).



Gambar 3. Flowcart Cross Over

#### 5. Menentukan Mutasi.

Langkah selanjutnya di algoritma genetika yaitu proses mutasi. Dimana mutasi sangat berperan untuk menggantikan nilai dari hasil individu dari crossover dengan nilai yang dibangkitkan dari nilai random, tidak semua individu hasil crossover terkena mutasi, letak rotasi sudut yang akan dilakukan mutasi dipilih secara acak dan ditentukan oleh nilai dari probabilitas mutasi. Nilai mutasi dapat diisi sesuai yang diinginkan. Mutasi adalah bagian proses digunakan untuk menggantikan isi nilai pada satu individu atau beberapa dari gen pada suatu kromosom individu hasil *crossover*. Optimasi proses mutasi yang akan dilakukan di kromosom memiliki tujuan untuk mendapatkan kromosom-kromosom baru dari individu *crossover* sebagai kandidat individu baru sebagai solusi pada generasi individu baru dengan fitness yang mempunyai individu lebih baik, dan proses optimasi sehingga memiliki solusi optimum seperti yang ditentukan. Akan tetapi, untuk mendapatkan hasil ini, penekanan selektif memiliki peranan penting.



Gambar 4. Flowcart Mutasi

### Hasil dan Pembahasan

Penelitian pertama dilakukan pada contoh soal matematika yang memiliki persamaan linier seperti persamaan 1.

$$F(x) = 5x_1 + 4x_2 + 8x_3 - 100 \dots\dots\dots (1)$$

$F(x)$  adalah fitness yang dicari dengan variable yang berbeda. Individu awal yang dibangkitkan secara acak. Dengan menggunakan 3000 populasi, ratemutasi 0,6 dan rate crossover 0,7 dengan pengulangan gen sebanyak 7 kali, didapatkan hasil seperti pada tabel 1:

Tabel 1. Generate Populasi Awal

$F(x)$	$x_1$	$x_2$	$x_3$	Individu Ke-	Gen Ke -
...	...	...	...	...	...
101	1	8	7	11826	4
218	13	21	8	12807	5
101	12	5	4	12808	5
776	27	31	64	12809	5
464	43	23	21	12810	5
1132	69	63	64	12811	5
436	42	36	10	12814	5
617	56	27	28	12816	5
679	96	52	1	12817	5
...	...	...	...	...	...

Dari hasil pada tabel 2.1,  $f(x)$  adalah hasil yang di inginkan dengan nilai 100, sedangkan jumlah individu awal berjumlah 3000 dengan pengulangan sebanyak 7 kali, maka individu yang akan dihasilkan untuk menyelesaikan problem di matematika tersebut sebanyak 21000 dengan random yang berbeda. Apabila individu ke 12809 di substitusika ke persamaan, maka  $5(12) + 4(4) + 3(3) = 100$ . Hasil tersebut membuktikan bahwa dengan algoritma genetika problem matematika dapat diselesaikan dan juga dapat memberikan solusi lebih dari satu. Proses penyelesaian tergantung pada besarnya individu awal yang disebut popsize dan probabilitas gen crossover dan gen mutasi. Semakin banyak populasi maka peluang untuk penyelesaian juga semakin banyak, dan juga probabilitas pada gen crossover dan gen mutasi tidak boleh lebih dari satu, apabila semakin banyak probabilitas, maka individu awal hasil seleksi, yang merupakan individu yang baik akan hilang, yang menyebabkan penyelesain tidak dapat

ditemukan. Berikut beberapa soal dari penyelesaian algoritma genetika dengan berbagai macam populasi, dengan probabilitas yang berbeda-beda seperti pada table.

**Tabel 2.**  
Penyelesaian Fungsi Linear

Populasi	Crossover	Mutasi	a	b	c
3000	0,6	0,7	12	8	1
			4	12	4
			3	3	6
			4	10	5
			8	11	2
			4	18	1
2000	0,4	0,5	0	25	0
			12	0	5
			0	3	11
			4	16	2

Beberapa hasil percobaan yang dilakukan seperti pada tabel 2. menjelaskan bahwa perbedaan popsize mempengaruhi hasil yang ingin diselesaikan. Peluang untuk mendapatkan hasil lebih besar jika random individu awal lebih banyak, rate mutasi dan rate crossover juga mempengaruhi hasil yang diinginkan. Penelitian kedua dilakukan pada contoh soal matematika yang memiliki persamaan kuadrat seperti persamaan 3.

**Tabel 3.**  
Penyelesaian Fungsi Kuadrat

Populasi	Crossover	Mutasi	$x_1$	$x_2$	$x_3$
3000	0,5	0,7	4	9	16
			3	7	21
			2	12	16
			2	3	28
			7	12	1
			7	0	17
2000	0,4	0,5	6	4	16
			3	10	17
			1	18	9
			5	12	9

Algoritma genetika pun dapat digunakan untuk menyelesaikan fungsi yang lebih kompleks seperti berikut tabel 3. Semakin banyak popsize maka semakin banyak pula individu awal yang dirandom, sehingga menyebabkan hasil dari penyelesaian fungsi mempunyai peluang lebih banyak. Begitu pula dengan crossover dan mutasi, nilai dari rate kedua fungsi tidak boleh lebih dari 1, agar dari pertukaran individu tidak semua yang mengalami proses crossover dan mutasi yang menyebabkan hilangnya individu baik. Proses pengulanganpun dapat mempengaruhi dari hasil yang diinginkan, sebagai contoh popsize 3000 dengan popsize 2000, hasil individu mempunyai popsize lebih sedikit cenderung menghasilkan angka nol pada penyelesaiannya.

## Kesimpulan

Algoritma Genetika dapat membantu menyelesaikan permasalahan matematika pada sistem persamaan linear. Hasil dari optimasi penyelesaian tersebut berupa solusi dengan bermacam-macam nilai *crossover* dan mutasi. Perbedaan nilai dalam rate mutasi dan rate *crossover* sangat berpengaruh dalam penyelesaian pada algoritma genetika, jika nilai rate lebih dari 1, maka dapat menyebabkan banyak individu awal hilang yang berakibat tidak adanya solusi yang ditemukan. Penelitian selanjutnya diharapkan dapat menggunakan hybrid algoritma dengan berbagai algoritma optimasi untuk menyelesaikan beberapa kasus lain seperti quadratic, geometri, dll.

## Acknowledgements

Terimakasih kepada Universitas Tulungagung yang telah memberikan dukungan dalam penelitian ini.

## Conflict of interests

The authors declare no conflict of interest.

## Referensi

- Anjasmara, D. R. (2019). *optimasi rute dan waktu distribusi menggunakan metode clark and wright saving heuristic di Coca Cola official distributor WARINGIN*. Diploma thesis, Politeknik APP Jakarta.
- Ardiansyah, H., & Junianto, M. B. S., (2022). Penerapan algoritma genetika untuk penjadwalan mata pelajaran. *Jurnal Media Informatika Budidarma*, 6(1), 329-336. <http://dx.doi.org/10.30865/mib.v6i1.3418>
- Caprio, D. D., Ebrahimnejad, A., Alrezaamiri, H., & Santos-Arteaga, F. J. (2022). A novel ant colony algorithm for solving shortest path problems with fuzzy arc weights. *Alexandria Engineering Journal*, 61(5), 3403-3415. <https://doi.org/10.1016/j.aej.2021.08.058>
- Cowdrey, K. W., Lange, J. D., Malekian, R., Wanneburg, J., & Jose, A. C., (2018). Applying queueing theory for the optimization of a banking model. *Journal of Internet Technology*, 19(2), 381-389. Retrieved from <https://jit.ndhu.edu.tw/article/view/1658>
- Hanief, S., & Jepriana, I. W. (2020). *Konsep algoritma dan aplikasinya dalam bahasa pemrograman C++*. Yogyakarta: Andi.
- Haruna, A. A., Jung, L. T., & Zakaria, N. (2015). Design and development of hybrid integrated thermal aware job scheduling on computational grid environment. *2015 International Symposium on Mathematical Sciences and Computing Research (ISMSC)* (pp. 13-17). Ipoh, Malaysia: IEEE.
- Heinonen, P., & Jusko, E. K. (2016). Development of a genetic algorithms optimization algorithm. *Proceedings of the 9th EUROSIM & the 57th SIMS*. Oulu, Finland: EUROSIM 2016 & SIMS
- Kowalski, M., Izdebski, M., Żak, J., Golda, P., & Manerowski, J. (2021). Planning and management of aircraft maintenance using a genetic algorithm. *Eksplatacja I Niezawodnosc – Maintenance and Reliability*, 23(1), 143–153. <http://dx.doi.org/10.17531/ein.2021.1.15>
- Krisnandi, K., & Agung, H. (2017). Implementasi algoritma genetika untuk memprediksi waktu dan biaya pengerjaan proyek konstruksi. *Jurnal Ilmiah FIFO*, 9(2), 90-97. <http://dx.doi.org/10.22441/fifo.2017.v9i2.001>
- Kristanto, E. U. (2016). Desain jembatan kayu dengan menggunakan kayu merbau di kabupaten sorong provinsi papua barat. Skripsi Arsit. UAJY, pp. 6–7.
- Kusaik, A., (1988). EXGT-S: A knowledge based system for group technology. *International Journal of Production Research*, 26(5), 887-904. <https://doi.org/10.1080/00207548808947908>
- Lamini, C., Benhlime, S., & Elbekri, A. (2018). Genetic algorithm based approach for autonomous mobile robot path planning. *Procedia Computer Science*, 127, 180-189. <https://doi.org/10.1016/j.procs.2018.01.113>
- Manampiring, R. R. (2015). Pengaruh perkembangan industri mebel terhadap pola pemanfaatan lahan di Desa Leilem Kecamatan Sonder. *Spasial: Perencanaan Wilayah dan Kota*, 1(1), 1-10. <https://doi.org/10.35793/sp.v1i1.8239>
- Mauluddin, S., Ikkal, I., & Nursikuwagus, A. (2018). Optimasi aplikasi penjadwalan kuliah menggunakan algoritma genetik. *Jurnal RESTI*, 2(3), 792-799.

- Oktarina, D., & Hajjah, A. (2019). Perancangan sistem penjadwalan seminar proposal dan sidang skripsi dengan metode algoritma genetika. *JOISIE (Journal of Information Systems and Informatics Engineering)*, 3(1), 32-40. <https://doi.org/10.35145/joisie.v3i1.421>
- Permata, R. A., Triyanto, D., & Ilhamsyah. (2016). Aplikasi penyusun menu makanan untuk pencegahan hiperkolesterolemia menggunakan algoritma genetika. *Coding Jurnal Komputer dan Aplikasi*, 4(2), 96-106. <http://dx.doi.org/10.26418/coding.v4i2.14763>
- Pratiwi, M. I. (2014). *implementasi algoritma genetika pada optimasi biaya pemenuhan kebutuhan gizi*. Sarjana thesis, Universitas Brawijaya.
- Retta, A. M., Isroqmi, A., & Nopriyanti, T. D. (2020). Pengaruh penerapan algoritma terhadap pembelajaran pemrograman komputer. *Indiktika: Jurnal Inovasi Pendidikan Matematika*, 2(2), 126-135. <http://dx.doi.org/10.31851/indiktika.v2i2.4125>
- Septyanto, R. B., Setyaningsih, E., & Bacharuddin, F. (2018). Analisis penempatan evolved node B Area DKI Jakarta dengan menggunakan algoritma genetika dan evolutionary programming. *TESLA: Jurnal Teknik Elektro*, 19(2), 108–123. <https://doi.org/10.24912/tesla.v19i2.2694>
- Sugeha, I. H., Inkiriwang, R. L., & Pratasih, P. A. K. (2019). Optimasi penjadwalan menggunakan metode algoritma genetika pada proyek rehabilitasi puskesmas minanga. *Jurnal Sipil Statik*, 7(12), 1669-1680.