

## ALGORITMA ALOKASI SUMBER DAYA RADIO PADA SISTEM KOMUNIKASI NIRKABEL MENGGUNAKAN GENETIKA DAN METODE *CLUSTERING*

1<sup>st</sup> Ilham Khairullah Utama <sup>1</sup>, Linda Meylani <sup>2</sup>, Vinsensius Sigit Widhi Prabowo <sup>3</sup>,  
<sup>1,2,3</sup> Universitas Telkom

lhamutama@telkomuniversity.student.ac.id<sup>1</sup>, lindameylani@telkomuniveristy.ac.id<sup>2</sup>, vinsensiusvsw@telkomuniversity.ac.id<sup>3</sup>

**Abstrak** — Masalah pada alokasi menjadi semakin kompleks seiring dengan peningkatan permintaan layanan data. Dengan meningkatnya permintaan layanan data maka solusi untuk masalah ini adalah melakukan alokasi sumber daya agar dapat digunakan secara bersamaan sambil mempertahankan *Quality of Service* (QoS). Penelitian ini mengusulkan sebuah metode untuk alokasi sumber daya radio pada sistem komunikasi menggunakan algoritma genetika. Metode ini menggabungkan teknik *clustering* dan penentuan posisi *relay* untuk meningkatkan efisiensi dan performa jaringan. Algoritma genetika digunakan untuk mengolah populasi solusi potensial melalui proses seleksi, *crossover*, dan mutasi, dengan tujuan menemukan solusi optimal secara bertahap. Teknik *clustering*, khususnya K-Means dan *hard clustering*, diterapkan untuk mengelompokkan pengguna berdasarkan karakteristik yang serupa seperti kualitas sinyal dan kebutuhan *bandwidth*. Hasil simulasi menunjukkan bahwa metode yang diusulkan mampu menghasilkan alokasi sumber daya yang lebih optimal dibandingkan dengan metode konvensional, terutama dalam hal *throughput* dan efisiensi spektrum. Hasil dari simulasi menunjukkan bahwa metode *clustering* K-Means *Clustering* dapat memberikan solusi yang seimbang antara *fairness* alokasi sumber daya dengan peningkatan kinerja yang substansial. Metode ini berhasil meningkatkan secara signifikan kapasitas sistem yang diukur dari *data rate* sebesar 2.5%, *sum rate* sebesar 2.64%, serta *power efficiency* sebesar 2.86%, tanpa mengorbankan prinsip keadilan yang menjadi perhatian utama dalam desain sistem komunikasi.

**Kata kunci**— *Resource block*, algoritma genetika, K-means clustering dan *hard clustering*.

### I. PENDAHULUAN

Perkembangan teknologi telekomunikasi telah meningkatkan jumlah perangkat terhubung dan kecepatan transmisi data secara signifikan. Peningkatan trafik data seluler yang pesat, bersama dengan bertambahnya jumlah pelanggan, menyebabkan penurunan kualitas layanan dan kepadatan trafik. Menurut data ITU, jumlah pelanggan telepon seluler di Indonesia meningkat dari 331 juta pada tahun 2019 menjadi 342 juta pada tahun 2022 [1]. Untuk mengatasi masalah ini, diperlukan teknologi baru yang dapat memenuhi permintaan layanan jaringan nirkabel dan kecepatan transfer data yang terus meningkat [2]. Salah satu solusi adalah komunikasi yang mengurangi peran BS dengan memungkinkan pengguna terhubung langsung melalui *relay*. Namun, model komunikasi ini dapat menyebabkan banyaknya pengguna yang ditampung dalam cakupan BS [3].

Maka dari itu, solusi untuk masalah ini adalah melakukan alokasi sumber daya agar dapat digunakan secara bersamaan sambil mempertahankan *Quality of Service* (QoS) [4]. Alokasi sumber daya menjadi aspek penting dalam pengelolaan jaringan nirkabel, sehingga diperlukan batasan dan spesifikasi yang jelas untuk memastikan kinerja sistem yang optimal [5]. Algoritma yang diusulkan adalah algoritma genetika. Algoritma genetika merupakan metode komputasi evolusioner yang terinspirasi dari proses seleksi alam. Algoritma ini bekerja dengan memanipulasi populasi solusi potensial melalui proses seleksi, *crossover*, dan mutasi untuk menemukan solusi optimal secara iteratif [6].

Beberapa penelitian terkait metode ini antara lain penelitian [7]. Pada penelitian tersebut disimpulkan bahwa penerapan algoritma genetika untuk alokasi spektrum

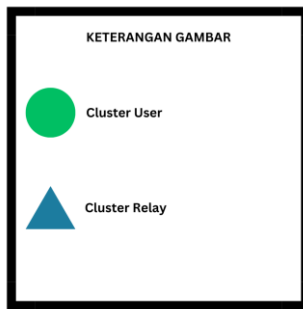
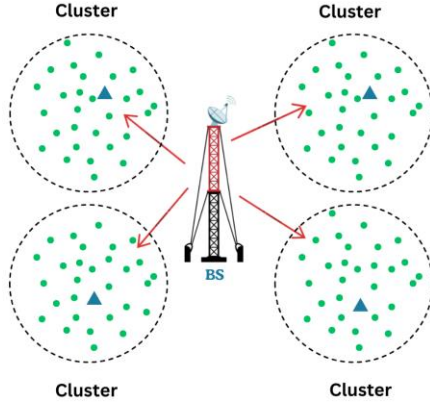
berhasil mengoptimalkan proses alokasi spektrum pada jaringan radio kognitif. Mekanisme evolusi yang diadopsi oleh algoritma ini memungkinkan pencarian solusi yang lebih baik secara iteratif. Operasi genetika seperti seleksi, *crossover*, dan mutasi berperan penting dalam menghasilkan individu-individu (solusi) yang lebih fit, sehingga meningkatkan kinerja keseluruhan sistem. Penelitian lain, yaitu [8] menunjukkan bahwa algoritma genetika dapat menjadi metode yang sangat berguna dalam merancang sistem komunikasi 5G *ultra-dense* yang lebih efisien. Dengan kemampuannya dalam mengatasi masalah alokasi *resource block* yang kompleks, algoritma genetika dapat membantu meningkatkan kualitas layanan dan kapasitas jaringan, sehingga memenuhi kebutuhan pengguna yang semakin meningkat.

Penelitian ini mengedepankan teknik *clustering*, yaitu sebuah metode dalam *machine learning* yang memungkinkan pengelompokan pengguna secara otomatis berdasarkan karakteristik serupa seperti kualitas sinyal dan kapasitas *bandwidth*. Dengan mengelompokkan pengguna berdasarkan kebutuhan yang serupa, alokasi sumber daya dapat dilakukan lebih efisien, mengurangi interferensi, dan memaksimalkan *throughput* sistem [9].

### II. MODEL SISTEM

Model sistem yang digunakan pada penelitian ini menggunakan *single cell* dengan *Base Station* (BS) yang terletak di tengah area jangkauan menggunakan skema *clustering* dan *hard clustering* dengan radius 1000 meter untuk CU yang tersebar secara acak dengan lokasi acak ditentukan menggunakan koordinat polar dengan jarak ( $d$ )

dan sudut acak ( $\theta$ ). BS hanya bertindak sebagai autentikasi akses, pengendali koneksi dan alokasi sumber daya. Sisi yang diamati yaitu pada arah *uplink* dimana *uplink* merupakan komunikasi dari CU ke BS. *Resource block* diasumsikan *orthogonal* sehingga tidak saling mengalami interferensi.



GAMBAR 1  
MODEL SISTEM

Model sistem dibangun menggunakan model *path-loss log normal*. *Path-loss* merupakan penurunan daya sinyal yang diterima dibandingkan dengan daya sinyal yang dipancarkan yang terjadi dalam jarak tertentu. Berikut merupakan rumus perhitungan *path-loss* dengan persamaan [10]:

$$pl = 10 \times n \times np \cdot \log_{10}(d0) \quad (1)$$

Dengan  $pl$  adalah *path-loss*,  $n$  adalah eksponen *path-loss*, dan  $d0$  adalah jarak referensi 0. Setelah mendapatkan nilai *path-loss*, maka dapat melakukan perhitungan *received power* dengan persamaan [11]:

$$P_{rx} = P_{tx} + g_{an} + g_{us} - pl - S \quad (2)$$

dengan  $P_{tx}$  adalah *transmit power* antena BS dalam satuan dBm,  $g_{an}$  merupakan *gain* antena BS dalam dB,  $g_{us}$  merupakan *gain* antena CU dalam dB,  $pl$  merupakan *path-loss*, dan  $S$  adalah *shadowing*.

Setelah mendapatkan nilai *received power*, maka dapat menghitung nilai *Signal To Noise Ratio* (SNR). SNR adalah perbandingan antara kekuatan sinyal yang diinginkan dengan kekuatan *noise* yang tidak diinginkan. SNR dengan nilai yang lebih tinggi menunjukkan kualitas sinyal yang lebih baik. Berikut merupakan persamaan untuk mencari SNR dengan persamaan [12]:

$$SNR = \frac{P_{rx}}{Noise} \quad (3)$$

Dimana  $P_{rx}$  merupakan *received power* dalam dBm dan *noise* dalam bentuk mW.

#### A. Parameter Sistem

Model sistem yang dibangun untuk mengelola jaringan komunikasi *user* dengan BS yang terletak di tengah-tengah pusat *cell* menggunakan frekuensi *carrier* 1.8 GHz dan *bandwidth* 180 KHz dengan radius BS sebesar 1000 meter, dengan masing-masing *user* berjarak sebesar 20 meter secara acak di sekitar BS tapi jaraknya tidak lebih besar dari radius BS. Penyebaran *user* dilakukan menggunakan jarak dan sudut yang acak. Model sistem ini menggunakan prinsip *clustering* dan *relay* menggunakan algoritma K-Means dengan membagi *cluster* sebanyak 4 bagian dengan masing-masing terdapat 1 *relay* di dalamnya untuk mengelompokan *user* dan *relay* berdasarkan karakteristik yang serupa seperti nilai SNR, jarak ke *user*, dan kebutuhan *bandwidth*. Setelah *user* dikelompokan menggunakan K-Means *cluster*, *user* yang paling sesuai dipilih salah satu untuk menjadi *relay* dan melayani *user* dalam *cluster* tersebut. Pemilihan dilakukan berdasarkan pada faktor-faktor seperti jarak, kualitas sinyal dan kapasitas *relay*. Setelah *cluster* terbentuk, sumber daya jaringan seperti *bandwidth*, *power*, atau kanal dapat dialokasikan secara dinamis ke setiap *cluster* sesuai dengan kebutuhan mereka misalnya *cluster* dengan *user* yang memiliki kebutuhan *bandwidth* tinggi dapat diberikan alokasi *bandwidth* yang lebih besar, sehingga dapat mengoptimalkan efisiensi dan kinerja jaringan seluler dengan cara yang lebih adaptif dan cerdas. Model matematis yang digunakan pada penelitian ini dilakukan dengan menggunakan transmisi *uplink* dimana membutuhkan 1 masukan yaitu SNR sehingga menghasilkan *data rate*. Sebelum dilakukan perhitungan perlu dilakukan inialisasi awal dari parameter-parameter dan variabel spesifikasi berdasarkan tabel berikut:

TABEL 1  
PARAMETER SISTEM

Parameter Sistem	Nilai Parameter
User / CU	250
Jarak minimum antar user	20 Meter
Gain antenna	5 dB
Gain user	3 dB
Frekuensi	1.8 GHz
Konstanta Cahaya (c)	$3 \times 10^8$ m/s
Received Power	27 dBm
Shadowing	4 dB
Noise density	-174 dBm
Bandwidth per rb	180 KHz

### B. Metode Clustering

*Clustering* merupakan metode *machine learning* yang digunakan untuk melakukan proses pengelompokan suatu data kedalam beberapa kelompok (*cluster*) berdasarkan kemiripan antara data-data tersebut. *Clustering* yang digunakan pada penelitian ini menggunakan metode *K-means Clustering* dan metode *Hard Clustering*. Berikut penjelasan terkait kedua metode *clustering* yang digunakan:

#### 1. K-Means Clustering

*K-Means* adalah jenis yang mengelompokkan *user* ke dalam sub-kelompok agar tidak terjadi tumpang tindih. Cara kerja dari *K-means Clustering* dengan menentukan berapa banyak *cluster* yang akan dibuat lalu inialisasi titik tengah (*centroid*). Titik *centroid* ditentukan berdasarkan koordinat  $x$  dan koordinat  $y$  dengan posisi *eNodeB* di titik pusat, proses ini dilakukan dengan perhitungan jarak Euclidean. Lakukan iterasi pada proses penentuan titik *centroid* tersebut hingga tidak ada perubahan dalam pengelompokan data atau hingga perubahan *sentroid* menjadi sangat kecil (*konvergensi*). Untuk *cluster* yang berisi nilai data  $N$  dengan kordinat  $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$  dapat dihitung dengan persamaan [13]:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad (4)$$

Dimana  $\bar{x}$  merupakan rata-rata dari semua titik  $N$  dari tiap *cluster*.

#### 2. Hard Clustering

*Hard clustering* adalah salah satu teknik dalam pengelompokan data di mana setiap data hanya dapat menjadi anggota dari satu *kluster* saja. Dalam metode ini, setiap elemen data secara tegas diklasifikasikan ke dalam satu *kluster* tertentu, tanpa adanya kemungkinan menjadi bagian lebih dari satu *kluster*. Dengan demikian, hasil dari *hard clustering* adalah pembagian data yang jelas dan tidak

tumpang tindih ke dalam *kluster-kluster* yang berbeda. Cara kerja dari *Hard Clustering* dengan menentukan jumlah *cluster* yang diinginkan. Kemudian pilih pusat *cluster* awal secara acak atau dengan metode tertentu, kemudian setiap data poin dalam *dataset* diberikan ke *cluster* yang pusatnya paling dekat, berdasarkan jarak Euclidean, lalu hitung ulang pusat *cluster* berdasarkan rata-rata posisi dari semua data poin yang termasuk dalam *cluster* tersebut. Untuk memproses *hard clustering* dapat dilakukan dengan persamaan [14]:

$$N = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (5)$$

Dimana  $N$  merupakan persamaan jarak euclidean dan  $x_1, x_2, y_1$  dan  $y_2$  merupakan koordinat *user*.

### C. Relay Selection

*Relay* merupakan sebuah *base station* (BS) berdaya rendah yang berperan dalam memperluas jangkauan dan kapasitas sebuah sel. Cara kerja *relay* adalah dengan menerima sinyal dari *base station* utama, kemudian melakukan *demodulasi* dan *decoding*, serta melakukan koreksi kesalahan, sebelum akhirnya menyalurkan kembali sinyal tersebut ke *user*. Penggunaan *relay* dikenal sebagai *relaying*, di mana *relay* ditempatkan dalam sebuah grup pada *cluster* tertentu. Penempatan ini memungkinkan pengurangan daya yang dibutuhkan oleh pemancar karena jarak transmisi yang lebih pendek, sehingga akan mengurangi terjadinya interferensi. *Relay* yang digunakan pada penelitian ini menggunakan *centroid-based relay*, *centroid-based relay* merupakan jenis *relay* yang meneruskan informasi dari BS kepada *user* dengan melakukan perhitungan berdasarkan pada koordinat pada setiap CU. Dalam mencari *relay* menggunakan persamaan.

$$R = \sqrt{(X_i - C_x)^2 + (Y_i - C_y)^2} \quad (6)$$

Dimana  $X_i$  dan  $Y_i$  merupakan koordinat dari *user* dan  $C_x$  dan  $C_y$  merupakan koordinat dari *centroid*.

## III. ALGORITMA GENETIKA

Algoritma genetika adalah algoritma yang mencari keturunan individu dengan kualitas terbaik, kemudian individu yang terbaik ini akan terpilih menjadi orang tua pada generasi berikutnya. Semua proses tersebut akan diulang, sehingga pada generasi berikutnya akan menerima lebih banyak keturunan dengan *fitness* yang tinggi. Proses ini akan terus berlanjut hingga terdeteksinya keturunan yang baik. Algoritma genetika beroperasi melalui tiga operator utama, yaitu seleksi proporsional, *crossover*, dan mutasi. Seluruh proses ini dilakukan berulang untuk membentuk beberapa generasi yang akan menghasilkan keturunan terbaik.

### A. Inisiasi Populasi dan Seleksi Proporsional

Sistem operasi seleksi proporsional bekerja dengan memilih kromosom kecocokan paling tinggi untuk menjadi orang tua dalam generasi berikutnya berdasarkan probabilitas

seleksi mereka. Probabilitas seleksi kromosom ini ditentukan oleh nilai *fitness* individu mereka.

populasi={individu1,individu2,...,individuP}

dengan:

$$RB_u = \begin{cases} RB \text{ Random,} & \text{Jika available } rb \neq 0 \\ -1, & \text{Jika available } rb = 0 \end{cases} \quad (7)$$

#### B. Crossover

Operator *crossover* memiliki fungsi untuk mengkombinasikan dua individu yang menjadi orang tua, serta menghasilkan keturunan untuk generasi selanjutnya. Awalnya, nilai acak dialokasikan untuk setiap blok sumber daya dalam keturunannya, dimana Panjang setiap blok harus sesuai dengan jumlah total komunikasi. Panjang ini mencerminkan jumlah potensial komunikasi yang dapat diakomodasi dalam blok sumber daya tersebut. persamaan untuk mencari *crossover* dan mutasi dalam penelitian ini. Proses *crossover* didapatkan dengan persamaan []:

$$C1 = \begin{cases} U1 & \text{jika } r \geq U_c \\ U2 & \text{jika } r < U_c \end{cases} \\ C2 = \begin{cases} U1 & \text{jika } r \geq U_c \\ U2 & \text{jika } r < U_c \end{cases} \quad (6)$$

Dimana C1 dan C2 merupakan kromosom, U1 dan U2 merupakan individu yang menjadi orang tua dan  $U_c$  merupakan probabilitas *crossover*.

#### C. Mutasi

Operator mutasi umumnya diterapkan pada satu individu dengan tujuan memodifikasi solusi secara acak untuk menghindari solusi lokal. Dalam algoritma ini, untuk keluar dari solusi optimal lokal, penerapan perubahan acak dalam kromosom melibatkan pemilihan dua pengguna seluler dari keturunan secara acak dan pertukaran di antara mereka. Hal ini dilakukan untuk meningkatkan variasi dalam solusi. Proses mutasi didapatkan dengan persamaan:

$$C[i], C[j] = C[j], C[i] \quad (7)$$

Dimana C merupakan kromosom, [i] dan [j] merupakan dua posisi acak.

TABEL 2

PSEUDECODE PROSES ALGORITMA GENETIKA

1. **FUNCTION** main\_algorithm\_genetic():
2. population=initialize\_population(pop\_size,user, num\_resource\_blocks)
3. **for** gen ∈ {0,1,...,num\_generations-1}

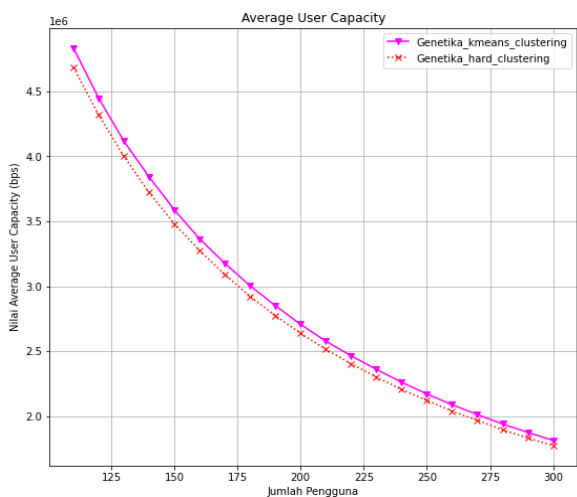
**do:**

4. fitness\_scores=[]
5. **for each** individual ∈ population **do:**
6. fitness = user, SNR
7. fitness\_scores.append(fitness)
8. C=[]
9. **while** length(C)<pop\_size **do:**
10. (U1,U2) = select\_parents(user, fitness\_scores)
11. (C1,C2) = crossover(U1,U2)
12. combined\_population = population + C1,C2
13. combined\_fitness\_scores = []
14. **for each** individual ∈ combined\_population **do:**
15. fitness = user, SNR
16. combined\_fitness\_scores.append(fitness)
17. **best\_individual**=individual ∈ population with highest fitness score
18. allocation\_matrix\_genetika=zeros(user\_grafik,num\_resource\_blocks)
19. **for** u ∈ {0,1,...,len(best\_individual)-1} **do:**
20. rb=best\_individual[u]
21. **if** rb ≠ -1 **then:**
22. **RETURN** allocation\_matrix\_genetika

## IV. HASIL DAN PEMBAHASAN

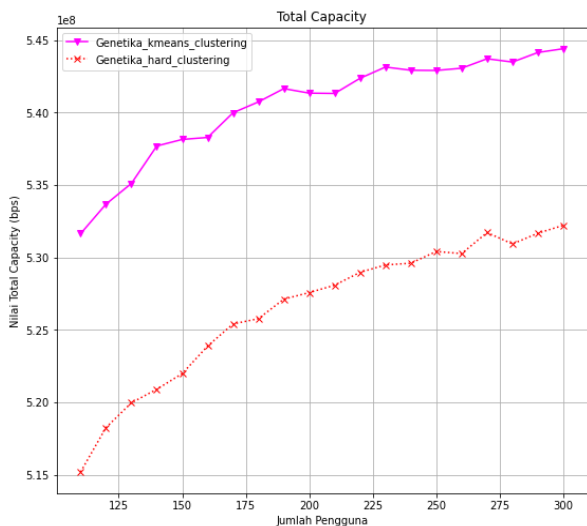
Pengujian dilakukan dengan melakukan variasi jumlah pengguna 100-250 *user* dengan kenaikan setiap 10 *user* diuji dengan radius 1000 meter. Setiap pengguna ditempatkan secara acak dengan jarak minimal 20 meter dan dialokasikan 100 RB. Proses *clustering* dilakukan untuk mengelompokkan pengguna dan menentukan *relay*. Nilai SNR dari setiap pengguna digunakan sebagai input untuk algoritma. Simulasi dijalankan sebanyak 500 iterasi untuk memperoleh hasil yang lebih representatif. Kinerja sistem dinilai berdasarkan parameter performansi *sum rate*, *data rate*, *efisiensi daya*, dan *fairness*.

Nilai *data rate* pada algoritma genetika metode *hard clustering* menghasilkan nilai  $2.80 \times 10^6$  bps, sedangkan pada algoritma genetika metode K-Means *clustering* menghasilkan nilai  $2.87 \times 10^6$  bps dengan selisih sebesar  $7 \times 10^4$  bps. Hal ini menunjukkan peningkatan sebesar 2,5% pada nilai *data rate* ketika menggunakan metode K-Means *clustering* dibandingkan dengan metode *hard clustering*. Hasil penelitian ini menunjukkan bahwa dalam pengujian, sinergi antara algoritma genetika dan K-Means *clustering* menghasilkan nilai *data rate* yang optimal, mengindikasikan efisiensi alokasi *resource block* yang lebih baik dibandingkan dengan metode *hard clustering*.



GAMBAR 3  
GRAFIK DATA RATE

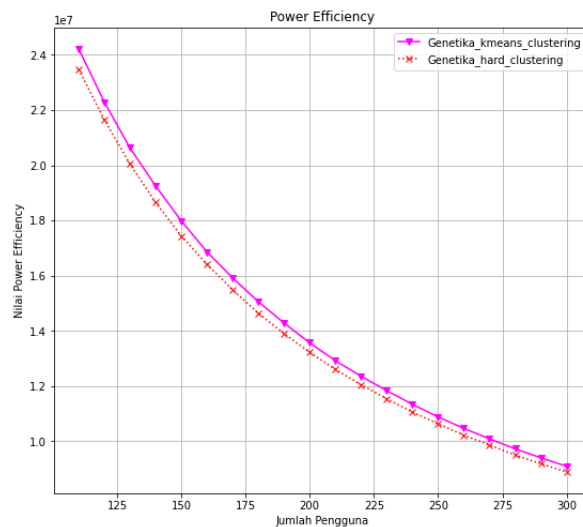
Nilai *sum rate* pada algoritma genetika metode *hard clustering* menghasilkan nilai  $5.26 \times 10^8$  bps, sedangkan pada algoritma genetika metode *K-Means clustering* menghasilkan nilai  $5.40 \times 10^8$  bps dengan selisih sebesar  $14 \times 10^6$  bps. Hal ini menunjukkan peningkatan sebesar 2,64% pada nilai *sum rate* ketika menggunakan metode *K-Means clustering* dibandingkan dengan metode *hard clustering*. Hasil penelitian ini mengindikasikan bahwa menggabungkan algoritma genetika dan *K-Means clustering* merupakan strategi yang paling efektif untuk memaksimalkan kapasitas sistem, sebagaimana tercermin pada nilai *sum rate* yang optimal.



GAMBAR 4  
GRAFIK SUM RATE

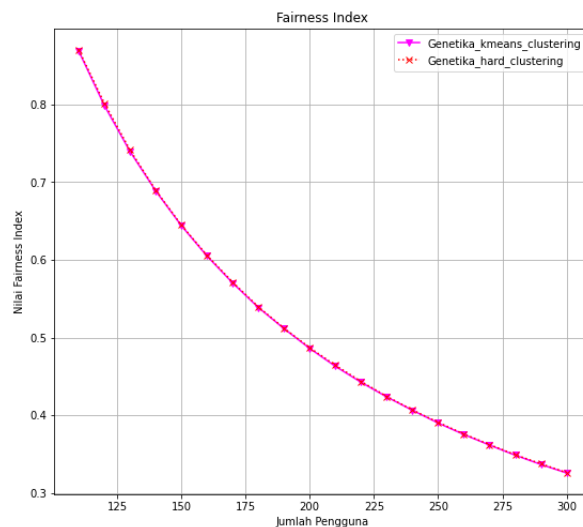
Nilai *power efficiency* pada algoritma genetika metode *hard clustering* menghasilkan nilai  $1.40 \times 10^7$  bps/w, sedangkan pada algoritma genetika metode *K-Means clustering* menghasilkan nilai  $1.44 \times 10^7$  bps/w. Algoritma genetika menggunakan metode *K-Means* memiliki performa lebih baik dengan nilai *power efficiency*  $1.44 \times 10^7$  bps/w dengan selisih sebesar  $4 \times 10^5$  bps. Hal ini menunjukkan peningkatan sebesar 2,86% pada nilai *power efficiency* ketika

menggunakan metode *K-Means clustering* dibandingkan dengan metode *hard clustering*. Hasil penelitian ini dapat disebabkan oleh reduksi kompleksitas komputasi yang diakibatkan oleh proses segmentasi awal menggunakan *K-Means clustering*, sehingga mempercepat konvergensi algoritma genetika menuju solusi optimal.



GAMBAR 5  
GRAFIK POWER EFFICIENCY

Nilai *fairness* pada algoritma genetika metode *hard clustering* menghasilkan nilai 51.72 %, sedangkan pada algoritma genetika metode *K-Means clustering* menghasilkan nilai 51.62 %. Hal ini menunjukkan penurunan sebesar 0.11% pada nilai *power efficiency* ketika menggunakan metode *hard clustering* dibandingkan dengan metode *K-Means clustering*. Meskipun perbedaan ini relatif kecil, algoritma genetika yang menerapkan *hard clustering* menunjukkan kinerja yang lebih baik dalam mempertahankan nilai *fairness*. Hasil penelitian ini disebabkan oleh kemampuan *hard clustering* dalam mengelompokkan pengguna dengan lebih jelas dan tegas, memungkinkan distribusi *resource block* yang lebih konsisten dan merata.



GAMBAR 6  
GRAFIK FAIRNESS

TABEL 3  
PERBANDINGAN NILAI PERFORMANSI KESELURUHAN

Jenis Parameter Performansi (Satuan)	Nilai Performansi metode K-Means Clustering	Nilai Performansi metode Hard Clustering
Data Rate (bps)	$2.87 \times 10^6$	$2.80 \times 10^6$
Sum Rate (bps)	$5.40 \times 10^8$	$5.26 \times 10^8$
Power Efficiency (bps/w)	$1.44 \times 10^7$	$1.40 \times 10^7$
Fairness (%)	51.62%	51.72%

## V. KESIMPULAN

Secara keseluruhan, nilai *data rate* tertinggi dicapai oleh algoritma genetika menggunakan metode K-Means clustering, yang mencapai  $2.87 \times 10^6$  bps. Hal ini menunjukkan bahwa dalam pengujian, kombinasi algoritma genetika dan K-Means clustering menghasilkan kinerja terbaik dalam hal kapasitas rata-rata pengguna dibandingkan dengan algoritma genetika menggunakan metode *hard clustering*. Keunggulan metode K-Means clustering secara efektif mengelompokkan pengguna berdasarkan kesamaan karakteristik, seperti posisi atau kondisi jaringan, sehingga alokasi sumber daya menjadi lebih terfokus dan efisien. Selanjutnya nilai *sum rate* tertinggi dicapai oleh algoritma genetika metode K-Means clustering yang mencapai  $5.40 \times 10^8$  bps. Nilai ini lebih tinggi dibandingkan dengan nilai *sum rate* pada metode *hard clustering*. Hasil ini menunjukkan bahwa kombinasi algoritma genetika dengan K-Means clustering memberikan performa terbaik dalam hal optimasi *sumrate*, menjadikannya metode yang paling efektif untuk mencapai *throughput* maksimal dalam pengujian yang dianalisis. Lalu algoritma genetika menggunakan metode K-Means memiliki Performa lebih baik dengan nilai *power efficiency*  $1.44 \times 10^7$  bps/w ini disebabkan oleh kompleksitas algoritma yang mencari solusi optimal melalui proses evolusi yang berkurang karena penggunaan metode K-Means. Dalam hal performa *fairness*, algoritma genetika dengan metode *hard clustering* sedikit lebih unggul, dengan nilai *fairness* mencapai 51,7189%, dibandingkan dengan 51,6164% yang dicapai oleh algoritma genetika dengan metode K-Means clustering. Meskipun perbedaan ini relatif kecil, algoritma genetika yang menerapkan *hard clustering* menunjukkan kinerja yang lebih baik dalam mempertahankan nilai *fairness*. Hal ini disebabkan oleh kemampuan *hard clustering* dalam mengelompokkan pengguna dengan lebih jelas dan tegas, memungkinkan distribusi sumber daya yang lebih konsisten dan merata. Meskipun demikian, metode K-Means clustering tetap mampu menjaga tingkat *fairness* yang hampir setara dengan *hard clustering*, sambil memberikan peningkatan signifikan dalam hal *data rate*, *sum rate*, dan *power efficiency*.

## REFERENSI

- [1] A. R. P. Octasylya and J. Rurianto. (2020, Sep). "Analisis Industri Telekomunikasi Seluler di Indonesia: Pendekatan SCP (*Structure Conduct Performance*)."
- [2] B. Setho, K. Sakti, A. Fahmi, V. Sigit, and W. Prabowo. (2019, Nov). "Analisis Performansi Alokasi Sumber Daya Radio Berbasis Algoritma Greedy pada Sistem Komunikasi D2D *Underlying Analysis Performance Radio Resource Allocation with Greedy Algorithm In D2D Underlying Communication*."
- [3] R. A. Mulyadi and U. K. Usman. (2020, Feb). "Komunikasi *Device-to-Device* pada Jaringan Seluler 5G menggunakan mmWave."
- [4] A. Maulana, A. Fahmi, I. Uke, and K. Usman. (2018, Feb). "Perencanaan Jaringan Heterogen LTE-Advanced Dengan Pico Cell Menggunakan Range Expansion Di Kota Cimahi *LTE-Advanced Heterogeneous Network Planning With Pico Cell Uses Range Expansion in Cimahi City*."
- [5] A. Alwarafy, M. Abdallah, B. S. Ciftler, A. Al-Fuqaha, and M. Hamdi. (2021, May). "Deep Reinforcement Learning for Radio Resource Allocation and Management in Next Generation Heterogeneous Wireless Networks: A Survey."
- [6] I. Brahma and H. Koubaa. (2021, October). "Genetic Algorithm based Resource Allocation for V2X Communications."
- [7] W. N. Singh and N. Marchang. (2022, January). "Spectrum Allocation in Cognitive Radio Networks Using Gene Therapy-Based Evolutionary Algorithms."
- [8] Reyna-Orta, A., & Andrade, Á. G. (2021, Jan). "Dimensionality reduction to solve resource allocation problem in 5G UDN using genetic algorithm."
- [9] M. Elsayed and M. Erol-Kantarci. (2020, Dec). "Radio Resource and Beam Management in 5G mmWave Using Clustering and Deep Reinforcement Learning."

- [10] L. Cao, S. Roy, and H. Yin. (2021, Jan). “*Resource Allocation in 5G Platoon Communication: Modeling, Analysis and Optimization.*” *IEEE Transactions on Vehicular Technology*. [Journal]. Available: <https://ieeexplore.ieee.org/document/9956752>.
- [11] L. Zhao, D. Wu, L. Zhou, and Y. Qian. (2022, April). “*Radio Resource Allocation for Integrated Sensing, Communication, and Computation Networks.*” *IEEE Transactions on Wireless Communications*. [Journal]. Available: <https://ieeexplore.ieee.org/abstract/document/9763441>
- [12] M. A. Khan, M. A. Jan, M. Alam, and M. Usman. (2021, April). “*Radio QoS-Aware Optimal Radio Resource Allocation Method for Machine-Type Communications in 5G LTE and Beyond Cellular Networks.*” *International Journal of Distributed Sensor Networks*. [Journal]. Available: <https://onlinelibrary.wiley.com/doi/full/10.1155/2021/9966366>.
- [13] P. Ajay, B. Nagaraj, and J. Jaya. (2022, April). “*Algorithm for Energy Resource Allocation and Sensor-Based Clustering in M2M Communication Systems.*” *Wireless Communications and Mobile Computing*. [Journal]. Available: <https://onlinelibrary.wiley.com/doi/full/10.1155/2022/7815916>.
- [14] M. Elsayed and M. Erol-Kantarci. (2021, April). “*A Framework for Radio Resource Allocation and SDMA Grouping in Massive MIMO Systems.*” *IEEE Access*. [Journal]. Vol. 9, pp. 61680-61696. Available: <https://ieeexplore.ieee.org/abstract/document/9409076>