

# ALOKASI SUMBER DAYA PADA DEVICE-TO-DEVICE UNDERLAY DOWNLINK MENGGUNAKAN ALGORITMA PARTICLE SWARM OPTIMIZATION

## (RESOURCE ALLOCATION FOR DEVICE TO DEVICE IN DOWNLINK UNDERLAY USING PARTICLE SWARM OPTIMIZATION ALGORITHM)

Ersa Rahma Aqila Majid<sup>1</sup>, Linda Meylani<sup>2</sup>, Vinsensius Sigit W.P<sup>3</sup>

<sup>1,2,3</sup> Universitas Telkom, Bandung

<sup>1</sup>ersarahma@student.telkomuniversity.ac.id,

<sup>2</sup>lindameylani@telkomuniversity.ac.id,<sup>3</sup>vinsensiusvsw@telkomuniversity.ac.id

### Abstrak

Seiring dengan perkembangan zaman, teknologi di bidang seluler semakin pesat. Peningkatan jumlah *Cellular User Equipment* (CUE) mengakibatkan permintaan kapasitas juga semakin meningkat sehingga beban trafik pada eNodeB meningkat. Sistem komunikasi *Device-to-Device* (D2D) menjadi solusi, pada sistem komunikasi ini, dua buah perangkat dapat berkomunikasi tanpa menggunakan eNodeB. Komunikasi pada D2D dilakukan dengan penggunaan bersama *resource block* yang dimiliki oleh *Cellular User Equipment* (CUE) oleh perangkat D2D. Namun penggunaan *resource* secara bersamaan ini menyebabkan terjadinya interferensi. Oleh karena itu, dibutuhkan skema alokasi pada *resource* CUE kepada pasangan D2D yang efisien.

Dalam penelitian ini, dilakukan skema pengalokasian *resource* pada *single cell* dengan arah komunikasi *downlink*. Skema pengalokasian *resource* dilakukan menggunakan algoritma *Greedy* dengan *Particle Swarm Optimization* (PSO) sebagai alokasi daya yang kemudian dibandingkan dengan algoritma *Greedy*. Kemudian dilakukan perhitungan dan analisis terhadap parameter kinerja.

Berdasarkan simulasi yang dilakukan, *Greedy* dengan *Particle Swarm Optimization* (PSO) sebagai alokasi daya pada 100 iterasi menghasilkan parameter kinerja yang paling baik dengan kenaikan *sumrate* 0.011%, *spectral efficiency* 0.011%, *power efficiency* 1.049%, *fairness* CUE 0.085%, *fairness* D2D 0.005% dan penurunan total daya 0.450%.

**Kata kunci:** *Device to Device, Greedy, Resource Block, Particle Swarm Optimization.*

### Abstract

Along with the times, technology in the cellular sector is growing rapidly. The increase in the number of Cellular User Equipment (CUE) has resulted in an increase in the demand for capacity so that the traffic load on the eNodeB will increase. Device-to-Device (D2D) communication system is the solution, in this communication system, two devices can communicate without using eNodeB. Communication in D2D is done by sharing resource blocks owned by Cellular User Equipment (CUE) by D2D devices. However, this concurrent use of resources causes interference. Therefore, an efficient allocation scheme for CUE resources to D2D pairs is needed.

In this work, a resource allocation scheme was carried out in a single cell with a downlink communication direction. The resource allocation scheme is carried out using the Greedy algorithm with Particle Swarm Optimization (PSO) as the power allocation which is then compared with the Greedy algorithm. Then performed the calculation and analysis of the performance parameters.

Based on the simulations carried out, Greedy with Particle Swarm Optimization (PSO) as power allocation in 100 iterations produces the best performance parameters with an increase in sumrate 0.011%, spectral efficiency 0.011%, power efficiency 1.049%, fairness CUE 0.085%, fairness D2D 0.005% and total power drop of 0.450%.

**Keywords:** *Kata kunci: Device to Device, Greedy, Resource Block, Particle Swarm Optimization.*

## 1. Pendahuluan

Seiring dengan perkembangan zaman, teknologi di bidang seluler semakin pesat. Semakin banyak jumlah *Cellular User Equipment* (CUE), maka permintaan akan kapasitas juga semakin meningkat sehingga beban trafik pada eNodeB meningkat pula. Maka diperlukan cara untuk mengatasi hal tersebut. Teknologi 5G memiliki fitur yaitu *Device to Device* (D2D), yang dirasa mampu memberikan solusi terhadap pengurangan beban trafik pada BTS. Komunikasi D2D merupakan komunikasi antar *device* secara langsung tanpa melibatkan eNodeB, hal ini terjadi karena komunikasi D2D memanfaatkan penggunaan kembali spektrum frekuensi pada CUE [1]. namun

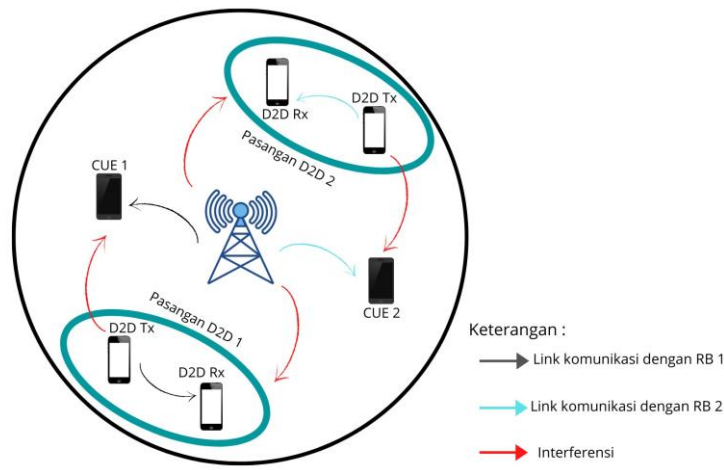
mempunyai beberapa kekurangan yaitu, belum adanya *resource block* sendiri sehingga D2D menggunakan spectrum frekuensi yang sama digunakan oleh *Cellular User Equipment* (CUE) [2], maka interferensi antara dua perangkat tersebut tidak dapat dihindari.

Salah satu upaya untuk mengurangi adanya interferensi yang terjadi adalah dengan mengoptimalkan pengalokasian *resource* dan memaksimalkan *sumrate* pada komunikasi D2D. Penelitian ini menggunakan algoritma *Particle Swarm Optimization* (PSO) sebagai alokasi daya algoritma *greedy* untuk optimasi dikarenakan jika tanpa PSO penggunaan daya akan maksimal, sedangkan jika menggunakan PSO dapat mengurangi penggunaan daya agar tidak maksimal tetapi kinerja yang dihasilkan baik. dan dibandingkan dengan algoritma *greedy*. Kemudian parameter kinerja seperti *sumrate*, *spectral efficiency*, *power efficiency*, *fairness* dan total daya yang dihasilkan akan dibandingkan dari setiap algoritma.

## 2. Metodologi dan Perancangan Simulasi

### A. Model Sistem dan Kanal

Pada penelitian ini sistem dimodelkan dengan *single cell* bertujuan untuk menghindari interferensi yang disebabkan oleh *cell* tetangga. Gambar 1 menunjukkan komunikasi D2D *underlying* pada arah *downlink*, dimana pasangan D2D memerlukan *resource* dari CUE, tiap pasangan D2D menempati 1 *resource* yang sama dengan CUE. Hal ini menyebabkan interferensi yang ditunjukkan dengan garis panah warna merah. Interferensi dirasakan oleh CUE dan D2D Receiver (D2DRx).



Gambar 2. 1 Model sistem

### B. Signal to Interference Noise Ratio (SINR)

SINR merupakan nilai perbandingan daya yang diterima *receiver* dengan jumlah *noise* dan interferensi yang terjadi. SINR yang diterima CUE dengan pengaruh interferensi dari pasangan D2D *j* dikalkulasi dengan persamaan :

$$SC_{i,j} = \frac{P_i \cdot G_{eNB,i}}{N_o + P_j \cdot G_{jTx,i}} \tag{1}$$

dimana  $G_{eNB,i}$  menunjukkan *gain* kanal eNodeB ke CUE ke-*i*,  $G_{jTx,i}$  menunjukkan *gain* kanal D2D Tx ke-*j* ke CUE,  $N_o$  menunjukkan *noise* dan  $P_i, P_j$  menunjukan daya pancar CUE ke-*i* dan daya pancar D2D ke-*j*

Adapun SINR yang didapatkan oleh pasangan D2D *j* dengan pengaruh interferensi dari eNodeB dikalkulasi dengan persamaan :

$$SD_{i,j} = \frac{P_j \cdot G_{jTx,jRx}}{N_o + P_i \cdot G_{eNB,jRx}} \tag{2}$$

dimana  $G_{jTx,jRx}$  menunjukkan *gain* kanal D2D Tx ke D2D Rx ke-*j* dan  $G_{eNB,jRx}$  menunjukkan *gain* kanal eNodeB ke D2D Rx.

Pekerjaan ini menggunakan model kanal *Urban Micro System* (UMi) dengan mempertimbangkan *pathloss* dan *shadowing* menggunakan persamaan (3).

$$G_{i,j} = PL_{i,j} + X_\sigma + \pi \tag{3}$$

Dimana  $PL_{i,j}$  yaitu *pathloss* yang dijelaskan dalam persamaan (4) dalam dB,  $X_{\sigma}$  yaitu variabel acak untuk *large scale fading*, itu adalah standar deviasi  $\sigma$  dan  $\pi$  menunjukkan *random variable* untuk *small scale fading* yang terdistribusi *rayleigh*

$$PL_{i,j} = 36.7 \log_{10}(d) + 22.7 + 26 \log_{10}(fc) \quad (4)$$

dimana  $d$  adalah jarak antara pengirim dan penerima dalam meter dan  $fc$  adalah frekuensi *carrier* dalam GHz.

### 3. Algoritma yang diajukan

#### A. Algoritma Greedy

Proses alokasi dengan algoritma *greedy* diawali dengan pasangan D2D urutan pertama dapat memilih CUE yang memiliki kapasitas SINR terbesar pada sisi eNodeB. Kapasitas SINR dari CUE terpilih akan diproyeksikan pada kapasitas SINR sisi D2D. Pasangan D2D lainnya tidak dapat memilih *resource* yang dimiliki oleh CUE yang telah dipilih oleh pasangan D2D sebelumnya. Hal ini dilakukan secara berulang hingga seluruh *user* teralokasikan.

#### B. Particle Swarm Optimization (PSO)

Algoritma *Particle Swarm Optimization* (PSO) dapat mengatasi permasalahan dengan cara menggerakkan partikel menggunakan fungsi tertentu untuk posisi dan kecepatan dari sebuah partikel. Proses alokasi daya *resource* kepada CUE dan pasangan D2D dengan menggunakan algoritma PSO.

Proses algoritma PSO dimulai dengan mengamsumsikan jumlah partikel ( $N$ ), kecepatan ( $v$ ), dan posisi ( $x$ ). Kemudian posisi awal pada tiap partikel dalam  $N$  disebar secara acak. Kemudian nilai *fitness* setiap partikel ditafsir menurut fungsi (*objective function*) yang ditetapkan. Jika nilai *fitness* setiap partikel pada iterasi pertama lebih baik dari  $P_{best}$  dan  $G_{best}$ , maka  $P_{best}$  dan  $G_{best}$  atur untuk posisi saat ini. Menghitung kecepatan pada iterasi berikutnya dengan persamaan 3.6

$$Velocity = C1 \times R1 \times (P_{best} \text{ partikel} - \text{Posisi partikel}) \quad (5)$$

Iterasi dilakukan sebanyak 3 kali dengan nilai 10, 50, dan 100 sampai menemukan solusi terbaik. Untuk data yang dialokasikan adalah  $P_{cu}$  dan  $P_{d2d}$ . Parameter yang digunakan pada penelitian ini seperti  $P_{best}$  adalah kapasitas personal setiap partikel,  $G_{best}$  adalah kapasitas sistem yang paling baik,  $C$  adalah konstanta, dan nilai *fitness function* dinyatakan dalam  $f(x, y)$  yang dimana  $f$  adalah *fitness function* (*sumrate* sistem),  $x$  adalah power D2D dan  $y$  adalah *power eNodeB*. Bertujuan untuk memaksimalkan *sumrate* dengan daya yang efisien.

#### C. Parameter Simulasi

Pada pekerjaan ini proses alokasi *resource* dilakukan menggunakan algoritma *greedy* yang kemudian dibandingkan dengan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya. Proses alokasi dilakukan dengan memvariasikan jumlah iterasi pada algoritma PSO. Tabel 1 menunjukkan parameter simulasi yang digunakan.

Selanjutnya dihitung parameter kinerja setiap algoritma yang digunakan. Parameter kinerja yang dihitung yaitu *sumrate*, *spectral efficiency*, *power efficiency*, *fairness* dan total daya. *Sumrate* menunjukkan banyak data yang ditransmisikan dalam satu detik. Persamaan (5) digunakan untuk menghitung *sumrate*[4].

$$\mu_{i,j} = B \cdot \log_2 (1 + SINR_{i,j}) \quad (6)$$

dimana SINR adalah *Signal to Interference Noise Ratio* dan  $B$  adalah *bandwidth* dalam satuan hertz. *Spectral Efficiency* menunjukkan kecepatan data yang dikirimkan dalam setiap hertz. Persamaan (6) digunakan untuk mengetahui nilai *spectral efficiency* pada sistem[5]

$$SE_{i,j} = \frac{\sum_{i=1}^C \mu_{i,j} + \sum_{i=1}^D \mu_{i,j}}{B \cdot RB} \quad (7)$$

dimana  $SE_{i,j}$  adalah *sumrate* dan  $RB$  adalah jumlah *resource block* yang digunakan. *Power Efficiency* menunjukkan seberapa efisien energi yang digunakan dalam mentransmisikan sejumlah bit dalam satu watt. Persamaan (7) digunakan untuk mengetahui nilai *power efficiency* pada sistem[6]

$$\eta_{i,j} = \frac{\sum_{i=1}^C \mu_{i,j} + \sum_{i=1}^D \mu_{i,j}}{C \cdot \sum_{i=1}^C P_c + D \cdot \sum_{i=1}^D P_d} \quad (8)$$

dimana  $C$  adalah jumlah *user*,  $D$  adalah jumlah pasangan D2D. *Fairness* merupakan keadilan *sumrate* yang dirasakan oleh *user*. Persamaan (8) digunakan untuk mengetahui nilai *fairness* menggunakan *Jain's Fairness Index*[7].

$$FI = \frac{\sum_{i=1}^C \mu_{i,j} + \sum_{i=1}^D \mu_{i,j}^2}{D \cdot \sum_{i=1}^C \mu_{i,j} + \sum_{i=1}^D \mu_{i,j}^2} \quad (9)$$

**Tabel 3. 1** Parameter Simulasi

Parameter	Value
Radius Cell	500 m
Radius D2D	20 m
Frekuensi kerja	1800 Mhz
Bandwidth RB	180 Khz
Jumlah resource blocks	50
Jumlah CUE	50
Jumlah pasangan D2D	5-50
Daya pancar CUE	3,98 Watt
Daya pancar D2DTx	1 Watt
Model kanal	Rayleigh (mean : 1, standar deviasi : 1)
Model shadowing	Lognormal (mean :0, standar deviasi :1)
Model pathloss	UMi pathloss
TTI	500
Jumlah Partikel	10
Congnative Constant (C1)	1
Social Constant (C2)	1
Inertion Constant	0.8
Iterasi	10, 50, 100

Nilai maksimum *fairnes* yaitu satu. Nilai *fairness* akan semakin baik jika mendekati satu. Jika nilai *fairness* adalah satu.maka setiap *user* mendapatkan *sumrate* yang adil. Total daya adalah parameter yang menunjukkan keseluruhan daya yang dipakai pada sistem. Total Daya memiliki satuan watt. Persamaan (9) digunakan untuk mengetahui total daya pada sistem.

$$TotalDaya = (Pc \times I) + (Pd \times J) \quad (10)$$

$P_{cu}$  adalah *power* dari CUE dan  $P_d$  adalah *power* dari D2D, I untuk total *user* CUE dan J untuk total *user* D2D.

#### 4. Hasil dan Analisis Simulasi

##### A. Sumrate

Pada gambar 2 menunjukkan hasil *sumrate* setiap algoritma yang digunakan berdasarkan penambahan jumlah pasangan D2D. Gambar 2 menunjukkan bahwa terjadi peningkatan nilai *sumrate* ketika terjadi peningkatan jumlah D2D baik menggunakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya atau algoritma *greedy*. Hal ini karena penambahan jumlah pasangan D2D dapat meningkatkan kapasitas, sehingga mempengaruhi *sumrate* yang dihasilkan.

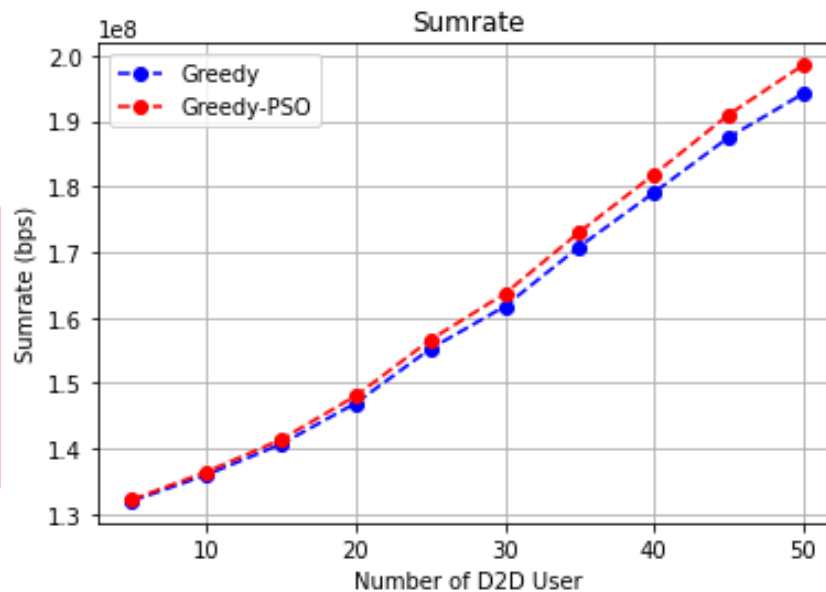
Tabel 2 menunjukkan nilai *sumrate* setiap algoritma. Berdasarkan data tersebut terlihat bahwa *sumrate* lebih tinggi terjadi pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya, Hal ini dikarenakan pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya mengalokasikan daya yang dimana mendapatkan kombinasi daya terbaik sehingga *sumrate* yang dihasilkan baik. Pada algoritma *greedy* dilakukan dalam satu fase saja sehingga *resource* yang terpilih adalah *resource* dengan nilai kapasitas terbesar. yaitu sebesar. Pada tabel 2 pada nilai *sumrate* algoritma *greedy* dengan algoritma PSO sebagai alokasi daya sebesar 162.2 Mbps. Perbedaan nilai *sumrate* dengan algoritma *greedy* tidak terlalu jauh.

**Tabel 4. 2** Perbandingan *sumrate* rata-rata tiap algoritma pada iterasi 100

Algoritma	Sumrate (Mbps)
<i>Greedy-Particle Swarm Optimization</i>	162.2
<i>Greedy</i>	160.4

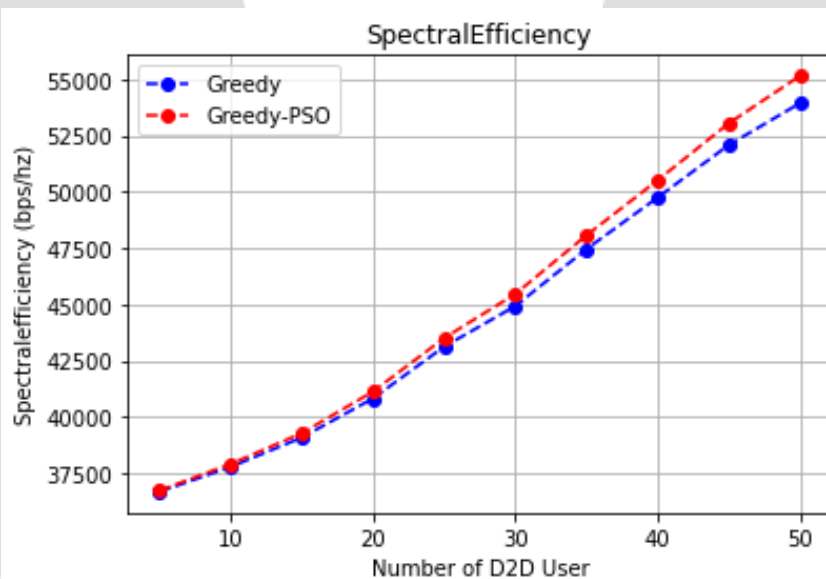
##### B. Spectral Efficiency

Pada gambar 3 menunjukkan hasil *spectral efficiency* setiap algoritma yang digunakan berdasarkan penambahan jumlah pasangan D2D. Gambar 3 menunjukkan bahwa terjadi peningkatan nilai *spectral efficiency* ketika terjadi



Gambar 4.2 Hasil *sumrate* tiap algoritma pada iterasi 100

peningkatan jumlah D2D baik menggunakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya atau algoritma *greedy*. Hal ini dikarenakan penambahan jumlah pasangan D2D dapat meningkatkan kapasitas sehingga mempengaruhi *sumrate* yang mengakibatkan peningkatan *spectral efficiency*.



Gambar 4.3 Hasil *spectral efficiency* tiap algoritma pada iterasi 100

Berdasarkan gambar 3 algoritma *greedy* dengan algoritma PSO sebagai alokasi daya lebih unggul dibandingkan dengan algoritma *greedy*, hal ini dikarenakan pada algoritma *greedy* hanya dilakukan dalam satu fase saja sehingga *resource* yang terpilih adalah *resource* dengan nilai kapasitas terbesar. Pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya, daya yang didapatkan adalah yang terbaik sehingga *sumrate* yang dihasilkan baik yang mempengaruhi juga nilai *spectral efficiency*.

Tabel 3 menunjukkan nilai rata-rata *spectral efficiency* dari setiap algoritma. Berdasarkan data tersebut terlihat bahwa *spectral efficiency* tertinggi terjadi pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya, yaitu sebesar 0.0450 Mbps/Hz sedangkan algoritma *greedy* sebesar 0.0445 Mbps/Hz. Hal ini dikarenakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya memiliki nilai rata-rata *sumrate* sistem yang lebih tinggi dari pada algoritma *greedy* sehingga nilai *spectral efficiency* sistem meningkat pula. Dan pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya yang dimana mendapatkan kombinasi dayang terbaik sehingga *spectral*

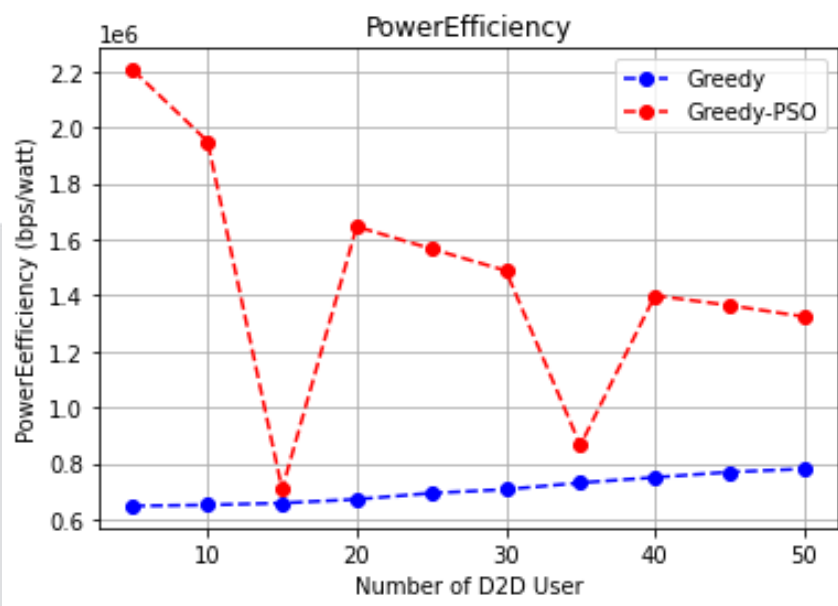
**Tabel 4. 3** Perbandingan *spectral efficiency* rata-rata tiap algoritma pada iterasi 100

Algoritma	<i>Spectral Efficiency</i> (Mbps/Hz)
<i>Greedy-Particle Swarm Optimization</i>	0.0450
<i>Greedy</i>	0.0445

*efficiency* naik. Pada algoritma *greedy* dilakukan dalam satu fase saja sehingga *resource* yang terpilih adalah *resource* yang memiliki nilai kapasitas terbesar.

### C. Power Efficiency

Gambar 4 menunjukkan perbandingan hasil *power efficiency* setiap algoritma yang digunakan berdasarkan perubahan jumlah pasangan D2D. Dari grafik tersebut dapat terlihat bahwa pada algoritma *greedy* penambahan jumlah pasangan D2D dapat meningkatkan *power efficiency* pada sistem. Hal ini dikarenakan penambahan jumlah pasangan D2D dapat meningkatkan kapasitas dan mempengaruhi *sumrate* sehingga menyebabkan peningkatan *power efficiency*.

**Gambar 4. 4** Hasil *power efficiency* tiap algoritma pada iterasi 100

Pada gambar 4 terlihat hasil *power efficiency* yang menggunakan algoritma *greedy* lebih linier dibandingkan dengan hasil *power efficiency* menggunakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya. Hal ini dikarenakan algoritma *greedy* dilakukan dalam satu fase saja sehingga *resource* yang terpilih adalah *resource* yang memiliki nilai kapasitas terbesar. Dalam grafik terlihat hasil *power efficiency* menggunakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya pada pasangan D2D ke-15 dan ke-35 menurun dikarenakan iterasi yang dipakai masih kurang banyak sehingga nilai belum menemukan nilai optimal.

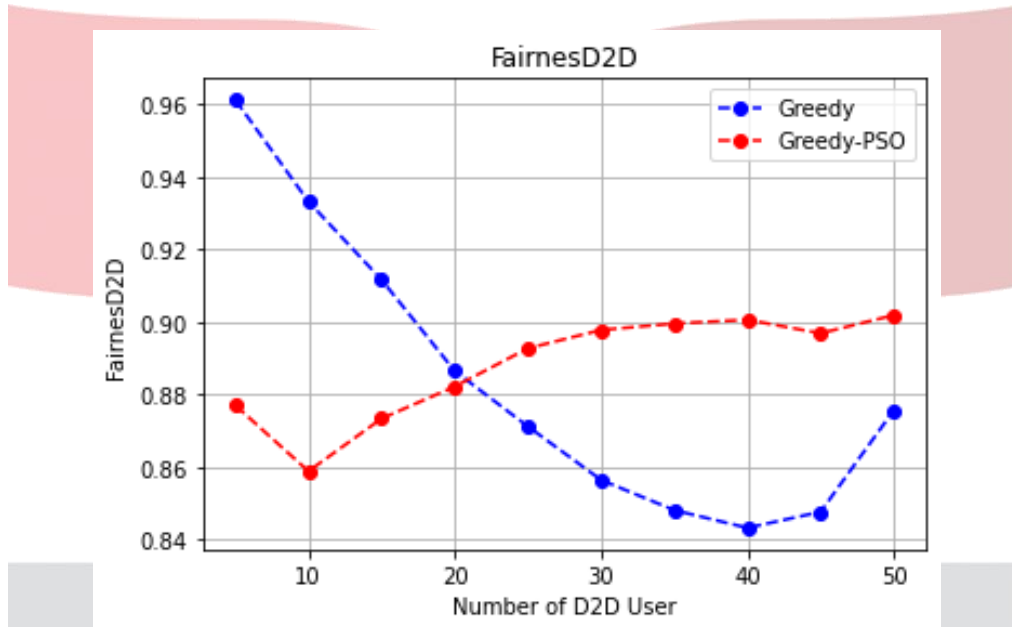
**Tabel 4. 4** Perbandingan *power efficiency* rata-rata tiap algoritma pada iterasi 100

Algoritma	<i>Power Efficiency</i> (Mbps/watt)
<i>Greedy-Particle Swarm Optimization</i>	1.445
<i>Greedy</i>	0.705

Tabel 4 menunjukkan nilai rata-rata *power efficiency* dari setiap algoritma. Berdasarkan data tersebut terlihat bahwa *power efficiency* lebih tinggi terjadi pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya, yaitu sebesar 1.445 Mbps/watt sedangkan algoritma *greedy* sebesar 0.705 Mbps/watt. Hal ini dikarenakan pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya mengalokasikan daya secara optimal yang dimana akan mendapatkan daya terbaik yang akan menyebabkan peningkatan *sumrate* maka nilai *power efficiency* juga naik. Pada algoritma *greedy* dilakukan dalam satu fase saja sehingga *resource* yang terpilih adalah *resource* dengan nilai kapasitas terbesar.

#### D. Fairness D2D

Pada gambar 5 menunjukkan perbandingan hasil *fairness* D2D dari setiap algoritma yang digunakan terhadap jumlah pasangan D2D. Gambar 5 menunjukkan hasil *fairness* D2D pada saat menggunakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya mengalami penurunan pada pasangan D2D ke-10 tetapi menaik lagi di pasangan D2D ke-15 dan terlihat lebih konstan sampai pasangan D2D ke-50 yang dimana pasangan D2D mendapatkan kesetaraan *dara rate* yang adil Tabel 5 menunjukkan nilai rata rata *fairness* D2D dari kedua algoritma.



**Gambar 4. 5** Hasil *fairness* D2D tiap algoritma pada iterasi 100

Berdasarkan data tersebut terlihat bahwa *fairness* D2D lebih tinggi terjadi pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya adalah 0.888 sedangkan algoritma *greedy* sebesar 0.883. Hal ini dikarenakan pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya pada saat alokasi daya diatur dengan baik yang dimana mempertimbangkan performansi sistem. Namun semua algoritma yang digunakan pada skenario ini menghasilkan *fairness* D2D yang cukup baik karena *index fairness* yang dihasilkan mendekati 1.

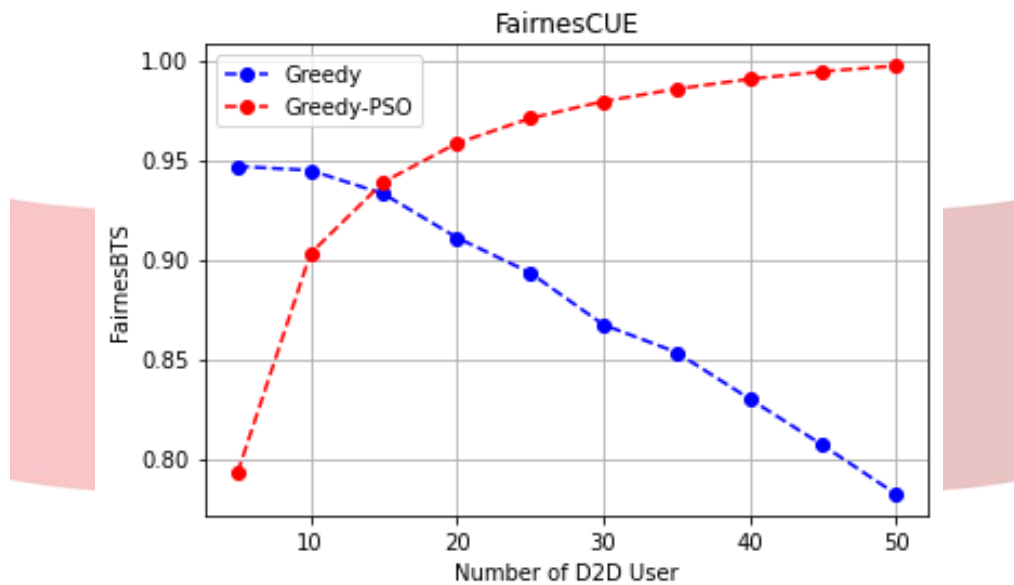
**Tabel 4. 5** Perbandingan *fairness* D2D rata-rata tiap algoritma pada iterasi 100

Algoritma	<i>Fairness</i> D2D
<i>Greedy-Particle Swarm Optimization</i>	0.888
<i>Greedy</i>	0.883

#### E. Fairness CUE

Pada gambar 6 menunjukkan perbandingan hasil *fairness* CUE dari setiap algoritma yang digunakan terhadap penambahan jumlah pasangan D2D. Gambar 6 menunjukkan hasil *fairness* CUE menaik dengan menggunakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya karena ketika jumlah CUE sama dengan jumlah pasangan D2D, *resource* yang dimiliki CUE dialokasikan kepada seluruh pasangan D2D dan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya mengatur daya CUE sehingga daya yang dipakai akan lebih adil. Pada algoritma *greedy* semakin banyak pasangan D2D menjadikan nilai *fairness* menurun karena *resource* yang dialokasikan pada pasangan D2D semakin akhir bukan yang terbaik.

Tabel 6 menunjukkan nilai rata rata *fairness* CUE dari setiap algoritma. Berdasarkan data tersebut terlihat bahwa *fairness* CUE lebih tinggi terjadi pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya adalah 0.951 dibandingkan dengan algoritma *greedy* sebesar 0.876. Hal ini dikarenakan algoritma *greedy* memilih kapasitas SINR CUE dan D2D paling besar sehingga didapat nilai SINR total yang besar. Sehingga *fairness* CUE lebih baik dengan menggunakan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya. Namun semua algoritma yang digunakan pada skenario ini menghasilkan *fairness* CUE yang cukup baik karena *index fairness* yang dihasilkan mendekati 1.



Gambar 4. 6 Hasil *fairness* CUE tiap algoritma pada iterasi 100

Tabel 4. 6 Perbandingan *fairness* CUE rata-rata tiap algoritma pada iterasi 100

Algoritma	Fairness CUE
<i>Greedy-Particle Swarm Optimization</i>	0.951
<i>Greedy</i>	0.876

### 5. Total Daya

Gambar 7 menunjukkan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya berwarna merah, dan algoritma *greedy* berwarna biru. Pada grafik total daya yang lebih banyak digunakan yaitu algoritma *greedy* dengan algoritma PSO sebagai alokasi daya, karena algoritma *greedy* dengan algoritma PSO sebagai alokasi daya pada setiap nilai rata-rata parameter kinerja selalu lebih tinggi dari algoritma *greedy*.

Tabel 7 menunjukkan nilai rata rata *total daya* dari kedua algoritma. Berdasarkan data tersebut terlihat bahwa *total daya* pada algoritma *greedy* sebesar 226.5 watt sedangkan algoritma *greedy* dengan algoritma PSO sebagai alokasi daya, yaitu sebesar 124.350 watt. Hal ini dikarenakan pada algoritma *greedy* dengan algoritma PSO sebagai alokasi daya mengatur besar kecilnya daya pada CUE dan D2D sehingga tidak selalu membuat daya menjadi maksimal semakin nilai parameter kinerja naik maka total daya akan menurun. Pada grafik saat menggunakan algoritma *greedy* menaik hasil total dayanya dikarenakan daya pada algoritma tersebut maksimal.

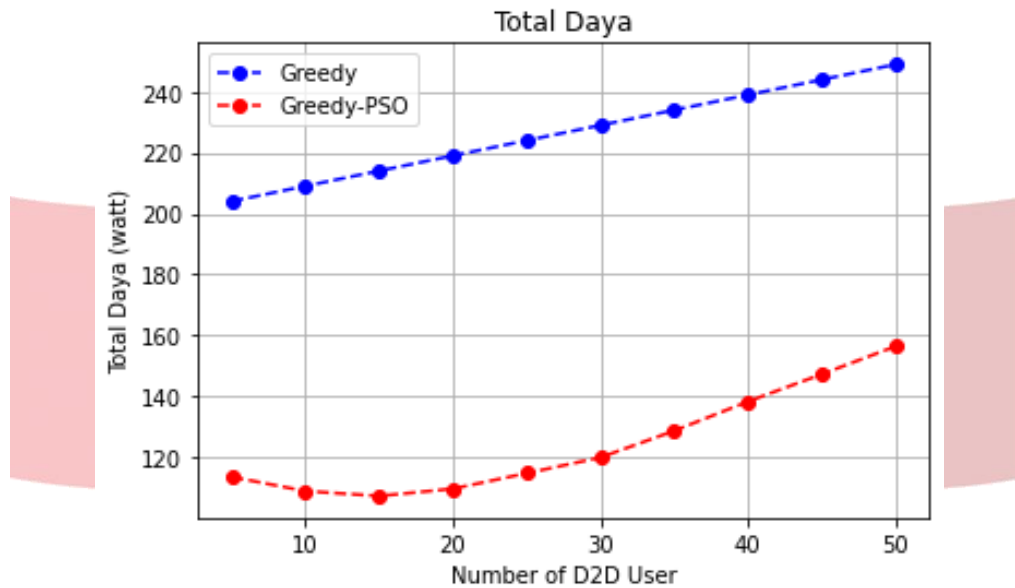
Tabel 5. 7 Perbandingan total daya rata-rata tiap algoritma pada iterasi 100

Algoritma	Total Daya (watt)
<i>Greedy-Particle Swarm Optimization</i>	124.350
<i>Greedy</i>	226.5

### 6. Kesimpulan

1. Memvariasikan iterasi yang dilakukan dalam penelitian ini memberi dampak yang begitu signifikan pada parameter kinerja, karena semakin banyak jumlah iterasi maka akan mendapatkan solusi terbaik.
2. Hasil simulasi menggunakan algoritma *Greedy* dengan *Particle Swarm Optimization* (PSO) sebagai algoritma alokasi daya menghasilkan parameter kinerja yang paling baik. Hal tersebut dikarenakan algoritma PSO mencari solusi dengan mendapatkan kombinasi daya terbaik, untuk mencapai sumrate yang baik.
3. Pada iterasi 100 algoritma *greedy* dengan PSO sebagai algoritma alokasi daya menghasilkan parameter kinerja yang paling baik yaitu 162.2 Mbps pada *sumrate*, 0.0450 Mbps/Hz pada *spectral efficiency*, dan 1.445 Mbps/watt pada *power efficiency*, 0,888 pada *fairness* D2D, 0,951 pada *fairness* CUE dan 124,350 watt pada total daya.





**Gambar 5. 7** Hasil total daya tiap algoritma pada iterasi 100

### Referensi

- [1] D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, G. Feng, and S. Li, "Device-to-Device Communications Underlying Cellular Networks," *IEEE Transactions on communications*, vol. 61, no. 8, pp. 3541–3551, 2013.
- [2] C. Wang, J. Qin, X. Yang, and W. Wen, "Energy-Efficient Offloading Policy in D2D Underlay Communication Integrated with MEC service," in *Proceedings of the 3rd International Conference on High Performance Compilation, Computing and Communications*, 2019, pp. 159–164.
- [3] A. ABADI *et al.*, "Manajemen interferensi dengan menggunakan power control untuk komunikasi device-to-device (d2d) dalam jaringan komunikasi seluler," 2017.
- [4] I. G. Fraimis and S. A. Kotsopoulos, "QoS-Based Proportional Fair Allocation Algorithm for OFDMA Wireless Cellular Systems," *IEEE Communications Letters*, vol. 15, no. 10, pp. 1091–1093, 2011.
- [5] J. Iqbal, M. A. Iqbal, A. Ahmad, M. Khan, A. Qamar, and K. Han, "Comparison of spectral efficiency techniques in device-to-device communication for 5g," *IEEE Access*, vol. 7, pp. 57 440–57 449, 2019.
- [6] D.-T. Huynh, X. Wang, T. Q. Duong, N.-S. Vo, and M. Chen, "Social-aware energy efficiency optimization for device-to-device communications in 5g networks," *Computer Communications*, vol. 120, pp. 102–111, 2018.
- [7] M. Y. Ramadhan, V. Sigit, and A. Fahmi, "Radio resource allocation for device to device network using auction algorithm," *Jurnal TIARSIE*, vol. 16, no. 2, pp. 53–58, 2019.