



**TUGAS AKHIR - TE 141599**

**IMPLEMENTASI METODE *DIFFERENTIAL EVOLUTION*  
PADA ANTENA ARRAY UNTUK MENGURANGI *SIDELobe***

Radifan Aiman Nabil Hibatulhaqqi  
NRP 2213 100 103

Dosen Pembimbing  
Eko Setijadi, ST., MT., Ph.D.

DEPARTEMEN TEKNIK ELEKTRO  
Fakultas Teknologi Elektro  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

*[Halaman ini sengaja dikosongkan]*



**FINAL PROJECT - TE 141599**

**IMPLEMENTATION OF DIFFERENTIAL EVOLUTION  
METHOD ON ANTENNA *ARRAY* TO REDUCE SIDELOBE**

Radifan Aiman Nabil Hibatulhaqqi  
NRP 2213 100 103

Supervisor  
Eko Setijadi, ST., MT., Ph.D.

ELECTRICAL ENGINEERING DEPARTMENT  
Faculty of Electrical Engineering  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

*[Halaman ini sengaja dikosongkan]*

## **PERNYATAAN KEASLIAN TUGAS AKHIR**

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “Implementasi metode *Differential Evolution* pada antena *array* untuk mengurangi *sidelobe*” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 5 Juli 2017

Radifan Aiman Nabil  
Hibatulhaqqi  
Nrp 2213 100 103

*[Halaman ini sengaja dikosongkan]*

**IMPLEMENTASI METODE *DIFFERENTIAL EVOLUTION*  
PADA ANTENA *ARRAY* UNTUK MENGURANGI *SIDELOBE***

**TUGAS AKHIR**

Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Sarjana Teknik  
Pada  
Bidang Studi Telekomunikasi Multimedia  
Departemen Teknik Elektro  
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing

  
**Eko Setijadi, ST., MT., Ph.D.**  
**NIP. 197210012003121002**



*[Halaman ini sengaja dikosongkan]*

# **IMPLEMENTASI METODE *DIFFERENTIAL EVOLUTION* PADA ANTENA ARRAY UNTUK MENGURANGI *SIDELOBE***

Radifan Aiman Nabil Hibatulhaqqi  
2213 100 103

Dosen Pembimbing : Eko Setijadi, ST., MT., Ph.D.

## **ABSTRAK**

Sistem antena *array* diperlukan pada sistem komunikasi jarak jauh seperti komunikasi satelit dan RADAR. Sistem antena *array* memiliki keunggulan gain yang tinggi, hal ini menjadikan sistem antena *array* menjadi penting pada komunikasi jarak jauh. Akan tetapi setiap penambahan jumlah elemen *array* akan menimbulkan *sidelobe* yang dapat mengganggu kinerja sistem komunikasi jarak jauh. Beberapa jenis metode optimasi untuk mengurangi *sidelobe level* telah digunakan, namun memberikan solusi yang kurang optimal. Untuk mengatasi kekurangan dari metode algoritma lain, metode algoritma *differential evolution* (DE) dikembangkan. Metode DE membandingkan selisih antar calon solusi dari permasalahan optimasi. Sehingga metode DE dapat menemukan solusi optimal dengan cepat.

Dalam penelitian ini dibandingkan nilai puncak *sidelobe* dan jumlah elemen *array* antena yang aktif pada metode optimasi *differential evolution* dan tanpa metode optimasi, diharapkan dapat membuktikan bahwa metode *differential evolution* adalah metode algoritma yang tepat untuk menurunkan nilai *sidelobe level* pada sistem antena *array*. Dari hasil simulasi, metode *differential evolution* mampu mengurangi puncak *sidelobe level* hingga -14,7442 dB dari -12 dB dengan menonaktifkan tiga elemen *array* pada *array* 15 elemen, -18,2010 dB dari -13,54 dB dengan menonaktifkan lima elemen *array* pada *array* 30 elemen, dan -28,8248 dB dari -24,59 dB dengan menonaktifkan dua elemen *array* pada *array* 60 elemen.

**Kata Kunci:** *Array Antena Linier, Optimasi, Sidelobe Level, Differential Evolution.*

*[Halaman ini sengaja dikosongkan]*

# IMPLEMENTATION OF DIFFERENTIAL EVOLUTION METHOD ON ANTENNA ARRAY TO REDUCE *SIDELOBE*

Radifan Aiman Nabil Hibatulhaqqi  
2213 100 103

Dosen Pembimbing : Eko Setijadi, ST., MT., Ph.D.

## ABSTRACT

An *array* antenna system is required on remote communications systems such as satellite communications and RADAR. Antenna *array* system has a high gain advantage, this makes the antenna *array* system becomes important in long-distance communication. But any addition of the number of *array* elements will cause *sidelobe* that can interfere with the performance of remote communication system. Several types of optimization methods to reduce *sidelobe* level have been used, but provide a less than optimal solution. To address the disadvantage of other algorithmic methods, differential evolution (DE) algorithm methods are developed. The DE method compares the difference between potential solutions from the optimization problem. So the DE method can find the optimal solution quickly.

In this study, we compared the peak *value* of *sidelobe* and the number of antenna *array* elements active with differential evolution optimization method and without optimization method, it is expected to prove that differential evolution method is the right algorithm method to decrease the *sidelobe* level *value* in the antenna *array* system. From the simulation results, the differential evolution method is able to reduce the *sidelobe* peak level to -14.7442 dB from -12 dB by disabling three *array* elements on an *array* of 15 elements, -18.2010 dB of -13.54 dB by disabling the five *array* elements on the *array* 30 elements, and -28,8248 dB from -24.59 dB by disabling two *array* elements on an *array* of 60 elements.

**Keyword:** *Array* of Linear Antenna, Optimization, *Sidelobe* Level, Differential Evolution.

*[Halaman ini sengaja dikosongkan]*

## **KATA PENGANTAR**

Alhamdulillahirabbil ‘alamin, puji syukur penulis panjatkan kepada Allah SWT, atas berkat rahmat-Nya yang telah diberikan sehingga penulis dapat menyelesaikan tugas akhir dengan judul:

### **IMPLEMENTASI METODE *DIFFERENTIAL EVOLUTION* PADA ANTENA ARRAY UNTUK MENGURANGI *SIDELobe***

Tugas akhir ini disusun sebagai salah satu persyaratan dalam menyelesaikan studi pada bidang studi Telekomunikasi Multimedia di jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember Surabaya.

Dalam kesempatan ini, penulis ingin menyampaikan rasa terima kasih kepada pihak-pihak yang telah mendukung penulis selama proses menyelesaikan tugas akhir ini, khususnya kepada:

1. Kedua orangtua penulis, yang selalu memberikan dukungan finansial maupun moral selama penulis menjalani proses perkuliahan di ITS, sampai akhirnya bisa menyelesaikan tugas akhir ini.
2. Bapak Eko Setijadi, ST., MT., Ph.D. selaku Dosen Pembimbing atas segala bimbingan selama mengerjakan Tugas Akhir ini.
3. Bapak dan Ibu dosen jurusan teknik elektro ITS, khususnya bidang studi Telekomunikasi Multimedia, atas segala ilmu yang telah diberikan selama penulis melaksanakan studi di ITS.
4. Semua rekan-rekan di lab antena dan propagasi, khususnya tim Radar, rekan-rekan E-53 dan LJ TMM atas segala motivasi yang diberikan .

Penyusunan laporan ini masih terdapat kekurangan, namun penulis berharap tugas akhir ini dapat bermanfaat bagi pembaca.

Surabaya, Juni 2017

*[Halaman ini sengaja dikosongkan]*

# DAFTAR ISI

<b>PERNYATAAN KEASLIAN</b> .....	<b>i</b>
<b>LEMBAR PENGESAHAN</b> .....	<b>iii</b>
<b>ABSTRAK</b> .....	<b>v</b>
<b>ABSTRACT</b> .....	<b>vii</b>
<b>KATA PENGANTAR</b> .....	<b>ix</b>
<b>DAFTAR ISI</b> .....	<b>xi</b>
<b>TABLE OF CONTENTS</b> .....	<b>xv</b>
<b>DAFTAR GAMBAR</b> .....	<b>xix</b>
<b>DAFTAR TABEL</b> .....	<b>xxiii</b>
<b>BAB 1 PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Permasalahan .....	1
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	2
1.5 Metodologi.....	2
1.6 Sistematika Penulisan .....	2
1.7 Relevansi .....	3
<b>BAB 2 TEORI PENUNJANG</b> .....	<b>5</b>
2.1 Terminologi Antena .....	5
2.2 Geometri antena .....	6
2.3 Pola Radiasi Faktor <i>Array</i> .....	8
2.3.1 Parameter Pola Radiasi .....	10
2.3.1.1 <i>Sidelobe Level</i> .....	10
2.3.1.2 <i>Half-power Beamwidth</i> .....	12
2.3.2 Pola Isotropis, Direksional, dan Omnidireksional .....	12
2.4 Perkalian Pola ( <i>Pattern Multiplication</i> ).....	13

2.5	<i>Array Antena (Antenna Array)</i> .....	14
2.5.1	<i>Array Antena Linier</i> .....	15
2.5.2	Metode <i>Thinning</i> pada <i>Array Antena</i> .....	18
2.6	<i>Differential Evolution</i> .....	19
2.6.1	Struktur <i>Differential Evolution</i> .....	21
2.6.2	Representasi Kromosom Solusi .....	22
2.6.3	Mekanisme Kerja <i>Differential Evolution</i> .....	22
2.6.4	Pembangkitan Populasi Awal .....	23
2.6.5	Evaluasi Fungsi Kebugaran ( <i>Fitness Function</i> ) .....	24
2.6.6	Operator Genetik .....	24
2.6.6.1	Mutasi .....	25
2.6.6.2	Pindah Silang ( <i>Crossover</i> ) .....	25
2.6.7	Proses Seleksi .....	27
2.6.8	Kriteria Terminasi .....	27
<b>BAB 3</b>	<b>PEMODELAN SISTEM SIMULASI</b> .....	<b>29</b>
3.1	Parameter Sistem pada Simulasi .....	29
3.2	Pemodelan Sistem .....	30
3.3	Prosedur Simulasi Antena <i>Array Linier</i> menggunakan <i>Differential Evolution</i> .....	31
3.3.1	Pembangkitan Individu .....	31
3.3.2	Pembentukan Kromosom Individu .....	31
3.3.3	Konversi Kromosom .....	32
3.3.4	Evaluasi Individu .....	33
3.3.5	Proses Mutasi .....	34
3.3.6	Proses <i>Crossover</i> .....	36
3.3.7	Proses Seleksi .....	36
3.3.8	Penentuan Kriteria Terminasi .....	37
<b>BAB 4</b>	<b>ANALISA HASIL SIMULASI</b> .....	<b>39</b>

4.1 Hasil Simulasi Antena <i>Array</i> Linier 15 Elemen Isotropis.....	40
4.1.1 Hasil Simulasi 15 Elemen Isotropis Tanpa Metode DE .....	40
4.1.2 Hasil Simulasi 15 Elemen Isotropis dengan Metode DE .....	41
4.2 Hasil Simulasi Antena <i>Array</i> Linier 30 Elemen Isotropis.....	47
4.2.1 Hasil Simulasi 30 Elemen Isotropis Tanpa Metode DE .....	47
4.2.2 Hasil Simulasi 30 Elemen Isotropis dengan Metode DE .....	48
4.3 Hasil Simulasi Antena <i>Array</i> Linier 60 Elemen Isotropis.....	55
4.3.1 Hasil Simulasi 60 Elemen Isotropis Tanpa Metode DE .....	55
4.3.2 Hasil Simulasi 60 Elemen Isotropis dengan Metode DE .....	56
4.4 Hasil Simulasi Antena <i>Array</i> Linier 15 Elemen Mikrostrip Slot	62
4.4.1 Simulasi 15 Elemen Mikrostrip Slot Tanpa Metode DE .....	63
4.4.2 Simulasi 15 Elemen Mikrostrip Slot dengan Metode DE....	64
4.5 Hasil Simulasi Antena <i>Array</i> Linier 30 Elemen Mikrostrip Slot	70
4.5.1 Simulasi 30 Elemen Mikrostrip Slot Tanpa Metode DE .....	70
4.5.2 Simulasi 30 Elemen Mikrostrip Slot dengan Metode DE....	71
4.6 Hasil Simulasi Antena <i>Array</i> Linier 60 Elemen Mikrostrip Slot	77
4.6.1 Simulasi 60 Elemen Mikrostrip Slot Tanpa Metode DE .....	77
4.6.2 Simulasi 60 Elemen Mikrostrip Slot dengan Metode DE....	78
4.7 Sintesis .....	84
<b>BAB 5 PENUTUP.....</b>	<b>87</b>
5.1 Kesimpulan .....	87
5.2 Saran .....	88
<b>DAFTAR PUSTAKA .....</b>	<b>89</b>
<b>LAMPIRAN A.....</b>	<b>91</b>
<b>LAMPIRAN B.....</b>	<b>93</b>
<b>LAMPIRAN C.....</b>	<b>105</b>
<b>BIOGRAFI PENULIS .....</b>	<b>107</b>

*[Halaman ini sengaja dikosongkan]*

## TABLE OF CONTENTS

<b>ORIGINALITY STATEMENT .....</b>	<b>i</b>
<b>APPROVAL SHEET .....</b>	<b>iii</b>
<b>ABSTRAK .....</b>	<b>v</b>
<b>ABSTRACT.....</b>	<b>vii</b>
<b>PREFACE.....</b>	<b>ix</b>
<b>DAFTAR ISI.....</b>	<b>xi</b>
<b>TABLE OF CONTENTS.....</b>	<b>xv</b>
<b>LIST OF FIGURES .....</b>	<b>xix</b>
<b>LIST OF TABLES .....</b>	<b>xxiii</b>
<b>CHAPTER 1 INTRODUCTION .....</b>	<b>1</b>
1.1 Backgrounds .....	1
1.2 Problems .....	1
1.3 Scope of problems .....	2
1.4 Objective.....	2
1.5 Methodology.....	2
1.6 Systematics .....	2
1.7 Relevances .....	3
<b>CHAPTER 2 LITERATURE REVIEW .....</b>	<b>5</b>
2.1 Antenna terminology .....	5
2.2 Antenna geometry.....	6
2.3 Array Factor Radiation Pattern .....	8
2.3.1 Radiation Pattern Parameter .....	10
2.3.2 Isotropic, Directional, dan Omnidirectional Pattern .....	12
2.4 Pattern Multiplication .....	13
2.5 Antenna Array .....	14
2.5.1 Linear Antenna Array.....	15

2.5.2	Antenna Array with Thinning Method .....	18
2.6	Differential Evolution .....	19
2.6.1	Differential Evolution Structure .....	21
2.6.2	Solution Chromosome Representation .....	22
2.6.3	Differential Evolution Work Mechanism .....	22
2.6.4	Initial Population Generation .....	23
2.6.5	Fitness Function Evaluation .....	24
2.6.6	Genetic Operators .....	24
2.6.7	Selections Process .....	27
2.6.8	Termination Criteria .....	27
<b>CHAPTER 3 SIMULATION SYSTEM MODELLING .....</b>		<b>29</b>
3.1	Simulation System Parameters .....	29
3.2	System Modelling .....	30
3.3	Simulation Procedure for Antenna Array using Differential Evolution .....	31
3.3.1	Individual Generation .....	31
3.3.2	Individual Chromosome Forming .....	31
3.3.3	Chromosome Conversion .....	32
3.3.4	Individual Evaluation .....	33
3.3.5	Mutation Process .....	34
3.3.6	Crossover Process .....	36
3.3.7	Selection Process .....	36
3.3.8	Termination Criteria .....	37
<b>CHAPTER 4 SIMULATION RESULTS ANALYSIS .....</b>		<b>39</b>
4.1	Linear Antenna Array 15 Isotropic Elements Simulation .....	40
4.1.1	15 Isotropic Elements With DE Simulation Result .....	40
4.1.2	15 Isotropic Elements non-DE Simulation Result .....	41
4.2	Linear Antenna Array 30 Isotropic Elements Simulation .....	47

4.2.1	30 Isotropic Elements With DE Simulation Result .....	47
4.2.2	30 Isotropic Elements non-DE Simulation Result .....	48
4.3	Linear Antenna Array 60 Isotropic Elements Simulation .....	55
4.3.1	60 Isotropic Elements With DE Simulation Result .....	55
4.3.2	60 Isotropic Elements non-DE Simulation Result .....	56
4.4	Linear Antenna Array 15 Slot Microstrip Elements Simulation	62
4.4.1	15 Slot Microstrip Elements With DE Simulation Result ...	63
4.4.2	15 Slot Microstrip Elements non-DE Simulation Result .....	64
4.5	Linear Antenna Array 30 Slot Microstrip Elements Simulation	70
4.5.1	30 Slot Microstrip Elements With DE Simulation Result ...	70
4.5.2	30 Slot Microstrip Elements non-DE Simulation Result .....	71
4.6	Linear Antenna Array 60 Slot Microstrip Elements Simulation	77
4.6.1	60 Slot Microstrip Elements With DE Simulation Result ...	77
4.6.2	60 Slot Microstrip Elements non-DE Simulation Result .....	78
4.7	Synthesis .....	84
<b>CHAPTER 5 RECOMMENDATIONS AND CONCLUSIONS.....</b>		<b>87</b>
5.1	SUMMARIES AND CONCLUSIONS .....	87
5.2	RECOMMENDATIONS FOR FUTURE STUDIES .....	88
<b>BIBLIOGRAPHY .....</b>		<b>89</b>
<b>APPENDIX A .....</b>		<b>91</b>
<b>APPENDIX B .....</b>		<b>93</b>
<b>APPENDIX C .....</b>		<b>105</b>
<b>BIOGRAPHY .....</b>		<b>107</b>

*[Halaman ini sengaja dikosongkan]*

## DAFTAR GAMBAR

<b>Gambar 2.1</b>	Geometri antena.....	7
<b>Gambar 2.2</b>	Pola radiasi antena .....	10
<b>Gambar 2.3</b>	Pola radiasi polar (a) <i>broadside</i> , (b) <i>endfire</i> .....	12
<b>Gambar 2.4</b>	<i>Array</i> dengan amplitudo dan fasa sama, jarak setengah panjang gelombang (a) <i>array</i> dipol pendek, (b) konsep perkalian pola .....	14
<b>Gambar 2.5</b>	Antena <i>array</i> linier dengan elemen berjarak sama .....	15
<b>Gambar 2.6</b>	Diagram alir <i>differential evolution</i> .....	23
<b>Gambar 2.7</b>	Ilustrasi operator mutasi.....	25
<b>Gambar 2.8</b>	Ilustrasi operator crossover .....	26
<b>Gambar 2.9</b>	Proses dan cara kerja crossover .....	26
<b>Gambar 3.1</b>	Diagram Alir Simulasi .....	30
<b>Gambar 4.1</b>	<i>Dense array</i> dari 15 elemen isotropis tanpa metode DE..	40
<b>Gambar 4.2</b>	Plot hasil 30 kali eksperimen DE 15 elemen isotropis.....	42
<b>Gambar 4.3</b>	Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen isotropis.	43
<b>Gambar 4.4</b>	Konvergensi gen individu pertama yang terpilih pada DE 15 elemen isotropis.....	44
<b>Gambar 4.5</b>	Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen isotropis.	44
<b>Gambar 4.6</b>	Konvergensi gen individu kedua yang terpilih pada DE 15 elemen isotropis.....	45
<b>Gambar 4.7</b>	Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen isotropis.	46
<b>Gambar 4.8</b>	Konvergensi gen individu ketiga yang terpilih pada DE 15 elemen isotropis.....	46
<b>Gambar 4.9</b>	<i>Dense array</i> dari 30 elemen isotropis tanpa metode DE..	48
<b>Gambar 4.10</b>	Plot hasil 30 kali eksperimen DE 30 elemen isotropis...	50
<b>Gambar 4.11</b>	Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen isotropis.	51
<b>Gambar 4.12</b>	Konvergensi gen individu pertama yang terpilih pada DE 30 elemen isotropis.....	51
<b>Gambar 4.13</b>	Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen isotropis.	52
<b>Gambar 4.14</b>	Konvergensi gen individu kedua yang terpilih pada DE 30 elemen isotropis.....	52

<b>Gambar 4.15</b>	Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen isotropis	.53
<b>Gambar 4.16</b>	Konvergensi gen individu ketiga yang terpilih pada DE 30 elemen isotropis .....	54
<b>Gambar 4.17</b>	<i>Dense array</i> dari 60 elemen isotropis tanpa metode DE	55
<b>Gambar 4.18</b>	Plot hasil 30 kali eksperimen DE 60 elemen isotropis ...	57
<b>Gambar 4.19</b>	Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen isotropis	.58
<b>Gambar 4.20</b>	Konvergensi gen individu pertama yang terpilih pada DE 60 elemen isotropis .....	59
<b>Gambar 4.21</b>	Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen isotropis	.59
<b>Gambar 4.22</b>	Konvergensi gen individu kedua yang terpilih pada DE 60 elemen isotropis .....	60
<b>Gambar 4.23</b>	Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen isotropis	.61
<b>Gambar 4.24</b>	Konvergensi gen individu ketiga yang terpilih pada DE 60 elemen isotropis .....	61
<b>Gambar 4.25</b>	Pola elemen mikrostrip slot .....	63
<b>Gambar 4.26</b>	<i>Dense array</i> dari 15 elemen mikrostrip slot tanpa metode DE .....	64
<b>Gambar 4.27</b>	Plot hasil 30 kali eksperimen DE 15 elemen mikrostrip slot .....	66
<b>Gambar 4.28</b>	Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen mikrostrip slot .....	67
<b>Gambar 4.29</b>	Konvergensi gen individu pertama yang terpilih pada DE 15 elemen mikrostrip slot .....	67
<b>Gambar 4.30</b>	Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen mikrostrip slot .....	68
<b>Gambar 4.31</b>	Konvergensi gen individu kedua yang terpilih pada DE 15 elemen mikrostrip slot .....	68
<b>Gambar 4.32</b>	Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen mikrostrip slot .....	69
<b>Gambar 4.33</b>	Konvergensi gen individu ketiga yang terpilih pada DE 15 elemen mikrostrip slot .....	70

<b>Gambar 4.34</b>	<i>Dense array</i> dari 30 elemen mikrostrip slot tanpa metode DE.....	71
<b>Gambar 4.35</b>	Plot hasil 30 kali eksperimen DE 30 elemen mikrostrip slot .....	73
<b>Gambar 4.36</b>	Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen mikrostrip slot .....	74
<b>Gambar 4.37</b>	Konvergensi gen individu pertama yang terpilih pada DE 30 elemen mikrostrip slot .....	74
<b>Gambar 4.38</b>	Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen mikrostrip slot .....	75
<b>Gambar 4.39</b>	Konvergensi gen individu kedua yang terpilih pada DE 30 elemen mikrostrip slot .....	75
<b>Gambar 4.40</b>	Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen mikrostrip slot .....	76
<b>Gambar 4.41</b>	Konvergensi gen individu ketiga yang terpilih pada DE 30 elemen mikrostrip slot .....	76
<b>Gambar 4.42</b>	<i>Dense array</i> dari 60 elemen mikrostrip slot tanpa metode DE.....	78
<b>Gambar 4.43</b>	Plot hasil 30 kali eksperimen DE 60 elemen mikrostrip slot .....	80
<b>Gambar 4.44</b>	Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen mikrostrip slot .....	81
<b>Gambar 4.45</b>	Konvergensi gen individu pertama yang terpilih pada DE 60 elemen mikrostrip slot .....	81
<b>Gambar 4.46</b>	Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen mikrostrip slot .....	82
<b>Gambar 4.47</b>	Konvergensi gen individu kedua yang terpilih pada DE 60 elemen mikrostrip slot .....	82
<b>Gambar 4.48</b>	Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen mikrostrip slot .....	83
<b>Gambar 4.49</b>	Konvergensi gen individu ketiga yang terpilih pada DE 60 elemen mikrostrip slot .....	83

*[Halaman ini sengaja dikosongkan]*

## DAFTAR TABEL

<b>Tabel 2.1</b> Terminologi yang digunakan dalam <i>differential evolution</i> ...	20
<b>Tabel 3.1</b> Sebagian individu hasil dari proses pembangkitan individu.	32
<b>Tabel 4.1</b> Parameter <i>differential evolution</i> .....	39
<b>Tabel 4.2</b> Persebaran data hasil eksperimen DE 15 elemen isotropis...	41
<b>Tabel 4.3</b> Hasil 30 eksperimen DE 15 elemen isotropis.....	41
<b>Tabel 4.4</b> Kombinasi gen yang terpilih pada DE 15 elemen isotropis..	47
<b>Tabel 4.5</b> Persebaran data hasil eksperimen DE 30 elemen isotropis...	48
<b>Tabel 4.6</b> Hasil 30 eksperimen DE 30 elemen isotropis.....	49
<b>Tabel 4.7</b> Kombinasi gen yang terpilih pada DE 30 elemen isotropis..	54
<b>Tabel 4.8</b> Persebaran data hasil eksperimen DE 60 elemen isotropis...	56
<b>Tabel 4.9</b> Hasil 30 eksperimen DE 60 elemen isotropis.....	56
<b>Tabel 4.10</b> Kombinasi gen yang terpilih pada DE 60 elemen isotropis	62
<b>Tabel 4.11</b> Persebaran data hasil eksperimen DE 15 elemen mikrostrip slot.....	64
<b>Tabel 4.12</b> Hasil 30 eksperimen DE 15 elemen mikrostrip slot .....	65
<b>Tabel 4.13</b> Kombinasi gen yang terpilih pada DE 15 elemen mikrostrip slot.....	70
<b>Tabel 4.14</b> Persebaran data hasil eksperimen DE 30 elemen mikrostrip slot.....	71
<b>Tabel 4.15</b> Hasil 30 eksperimen DE 30 elemen mikrostrip slot .....	72
<b>Tabel 4.16</b> Kombinasi gen yang terpilih pada DE 30 elemen mikrostrip slot.....	77
<b>Tabel 4.17</b> Persebaran data hasil eksperimen DE 60 elemen mikrostrip slot.....	78
<b>Tabel 4.18</b> Hasil 30 eksperimen DE 60 elemen mikrostrip slot .....	79
<b>Tabel 4.19</b> Kombinasi gen yang terpilih pada DE 60 elemen mikrostrip slot.....	84

*[Halaman ini sengaja dikosongkan]*

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Pola radiasi satu elemen antena relatif lebar dan setiap elemen memberikan gain yang rendah. Banyak aplikasi yang membutuhkan karakteristik antena yang terarah untuk memenuhi kebutuhan komunikasi jarak jauh, seperti pada komunikasi satelit, *mobile*, dan radar. Hal ini hanya dapat dipenuhi dengan memperbesar ukuran antena, akan tetapi ada cara lain yaitu dengan membentuk elemen sesuai dengan konfigurasi geometri tertentu. Antena dengan banyak elemen ini disebut *array*.

Ketika mendesain elemen antena *array*, kita dapat mengubah jenis elemen, orientasi, lokasi dan arus transmisi. Pola radiasi antena *array* ditentukan dengan masukan parameter desain dan delay relatif tiap elemen.

Antena *array* memiliki nilai gain yang tinggi dan memiliki pola radiasi terarah pada penerimaan sinyal datang di sudut tertentu, hal ini yang mendasari penggunaan antena *array* pada sistem RADAR. Setiap penambahan *array* pada antena maka semakin kecil pola radiasi *sidelobe* yang terjadi namun arus eksitasi transmisi yang diperlukan menjadi lebih banyak.

*Main lobe* dapat membawa informasi utama tentang jarak, kecepatan, dan sudut, akan tetapi *sidelobe* adalah radiasi *lobe* yang diluar arah sudut yang diinginkan pada pola radiasi yang terarah. *Sidelobe* juga membawa informasi utama yang sama dengan *main lobe*. Sehingga pola radiasi *main lobe* dapat terganggu oleh interferensi dari pola radiasi *sidelobe*. Maka dari itu, dibutuhkan metode untuk meminimalisir *sidelobe* yang terbentuk oleh *array* antena sehingga informasi pada *main lobe* dapat diperoleh secara maksimal.

### 1.2 Permasalahan

Adapun permasalahan yang telah dirumuskan pada tugas akhir ini adalah :

1. Bagaimana teknik pengurangan *sidelobe* level pada *array* antena isotropis?
2. Bagaimana penggunaan metode *differential evolution* (DE) untuk optimasi elemen *array* isotropis pada parameter arus eksitasi dan teknik *thinning*?

### **1.3 Batasan Masalah**

Adapun batasan masalah dari tugas akhir ini yaitu:

1. Antena yang digunakan adalah antena *array* linier dengan elemen isotropis.
2. Algoritma optimasi yang digunakan adalah *differential evolution* (DE).
3. Desain keseluruhan sistem hanya dilakukan dengan simulasi menggunakan *software*.

### **1.4 Tujuan**

Adapun tujuan penulisan tugas akhir ini adalah:

1. Meminimalkan level *sidelobe* pada *array* antena isotropis linier.
2. Menggunakan metode DE untuk optimasi elemen *array* antena isotropis linier.
3. Menerapkan hasil metode DE pada *array* antena isotropis linier.

### **1.5 Metodologi**

Adapun metodologi yang digunakan pada penelitian tugas akhir ini adalah sebagai berikut:

1. Studi literatur *array* antena isotropis linier dan *differential evolution*.
2. Mensimulasikan desain geometri *array* antena isotropis linier dengan proses simulasi *differential evolution*.
3. Menganalisa performansi antena hasil simulasi sebelum dan sesudah *differential evolution* diterapkan.
4. Menyusun tugas akhir.

### **1.6 Sistematika Penulisan**

Laporan tugas akhir ini disusun dengan sistematika sebagai berikut:

#### **BAB I: PENDAHULUAN**

Bagian pendahuluan berisi latar belakang, maksud dan tujuan penulisan, perumusan masalah, batasan masalah, metode yang digunakan, sistematika penulisan, dan relevansi yang dibahas pada tugas akhir ini.

#### **BAB II: DASAR TEORI**

Pada bab ini akan dijelaskan teori-teori yang mendasari desain antena dan *differential evolution*.

### BAB III: METODE SIMULASI

Bab ini menjelaskan tentang pemodelan yang dilakukan pada tugas akhir ini.

### BAB IV: HASIL SIMULASI DAN ANALISA

Bab ini berisi hasil dari model simulasi yang telah dijelaskan pada BAB III dan analisisnya. Analisa yang dilakukan fokus pada perbandingan performansi *array* antena isotropis linier sebelum dan sesudah *differential evolution* diterapkan.

### BAB V: PENUTUP

Bab ini merupakan kesimpulan dari keseluruhan uraian pada bab-bab sebelumnya dan juga berisi saran.

#### **1.7 Relevansi**

Dengan terselesaikannya tugas akhir ini, dapat diperoleh kombinasi optimal antar elemen *array* antena isotropis linier dengan *sidelobe* level minimal.

*[Halaman ini sengaja dikosongkan]*

## **BAB 2**

### **TEORI PENUNJANG**

Pada bab ini akan diberikan teori dasar yang melandasi permasalahan dan penyelesaiannya yang diangkat dalam tugas akhir ini, meliputi terminologi antena, geometri antena, pola radiasi antena, antena *array*, dan *differential evolution*.

#### **2.1 Terminologi Antena**

Antena didefinisikan sebagai suatu perangkat yang dapat digunakan untuk meradiasikan dan/atau menerima gelombang radio. Di sistem komunikasi *wireless*, antena dipakai untuk proses pentransmisi dan pendeteksian. Antena juga didefinisikan sebagai peralatan untuk melakukan proses *coupling* frekuensi radio pada transmisi kabel menuju ruang bebas, meradiasikan daya ke ruang bebas, dan berlaku sebaliknya. Antena dapat berbentuk sangat sederhana seperti sebuah kabel atau sangat rumit dengan komponen elektronik aktif. Oleh karena merupakan perangkat perantara antara saluran transmisi dan udara, maka antena harus mempunyai sifat yang sesuai dengan saluran atau jaringan transmisinya.

Dari pengertian sebelumnya antena dapat digunakan dalam beberapa fungsi antara lain:

1. Antena komunikasi *point-to-point* (p2p) untuk mentransfer sinyal dari satu tempat menuju tempat yang lain, misalnya antena sistem terestrial, antena sistem satelit, dan lain-lain.
2. Antena penerima yang difungsikan untuk menerima sinyal, baik dari pemancar buatan manusia (dalam kasus *broadcast ataupun point-to-point*) atau menerima sinyal bebas dari alam semesta.
3. Antena pemancar *broadcast* untuk memancarkan sinyal ke area yang sangat luas, misalnya antena pemancar radio modulasi frekuensi, antena pemancar televisi, antena sistem posisi global dan lain-lain.

Secara umum, antena digolongkan menjadi antena isotropis, antena *omnidirectional*, antena *phase array*, antena optimal dan antena adaptif. Antena isotropis (*isotropic*) merupakan sumber titik yang memancarkan daya ke segala arah dengan intensitas yang sama, seperti permukaan bola. Antena ini tidak dapat diwujudkan dalam dunia nyata

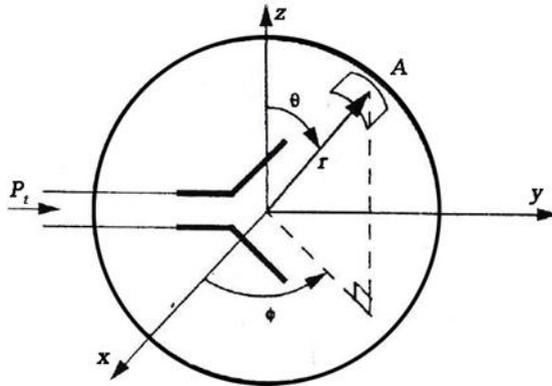
dan hanya digunakan sebagai dasar untuk merancang serta menganalisa geometri antena yang lebih kompleks. Antena *omnidirectional* adalah antena yang memancarkan daya ke segala arah, dan bentuk pola radiasinya digambarkan seperti bentuk donat dengan pusat tidak berlubang. Antena ini dapat dibuat, dan dalam pengukuran dipergunakan sebagai pembandingan terhadap antena yang lebih rumit. Contoh antena ini adalah antena *dipole* setengah panjang gelombang. Antena *directional* merupakan antena yang memancarkan daya lebih besar ke arah tertentu. Gain antena ini relatif lebih besar dari antena *omnidirectional*. Contoh, suatu antena dengan gain 10 dBi (atau “dB” saja). Artinya antena ini pada suatu arah tertentu memancarkan daya 10 dB lebih besar dibandingkan dengan antena isotropis. Ketiga jenis antena di atas merupakan antena tunggal, dan bentuk pola radiasinya tidak berubah tanpa merubah geometri antena atau diputar secara mekanis dari geometri antena.

Antena *phase array* merupakan pergabungan *array* dari beberapa antena lebih simpel dan sinyal digabungkan menginduksi masing-masing elemen tersebut untuk membentuk pola radiasi khusus pada *output array*. Setiap antena sederhana yang menyusun konfigurasi *array* disebut dengan elemen *array*. Arah gain maksimum dari antena *phase array* dapat menentukan dengan diaturnya fase antar elemen-elemen *array*.

Antena optimal merupakan suatu antena yang penguatan (*gain*) dan relatif fase setiap elemen antenanya diubah sedemikian rupa agar didapatkan kerja yang outputnya paling optimal. Optimasi kerja dan performa dapat dilakukan dengan dihilangkan atau diminimalkan sinyal-sinyal tidak diinginkan dan dimaksimalkan sinyal diterima yang diinginkan.

## 2.2 Geometri antena

Geometri dasar antena dapat digambarkan pada gambar 2.1. Dapat digambarkan bahwa masukan antena berupa sinyal dari jaringan/transmisi kabel. Dianggap sinyal tersebut mempunyai rata-rata daya  $p_i$ . Antena memancarkan daya ke segala arah [1].



**Gambar 2.1** Geometri antena

Agar menyederhanakan proses analisa, arah digambarkan dalam koordinat polar,  $\phi$  dan  $\theta$ . Jika vektor  $\mathbf{r}$  adalah koordinat polar, maka  $\phi$  merupakan sudut yang diapit oleh sumbu-x dan proyeksi  $\mathbf{r}$  terhadap bidang x-y, dan  $\theta$  merupakan sudut antara sumbu-z dan  $\mathbf{r}$ .

Kerapatan daya memiliki unit  $w/(\text{rad})^2$  pada arah  $(\phi, \theta)$  dinyatakan sebagai  $U(\phi, \theta)$ . Jika dianggap antena bersifat ideal, dan menangkap secara optimal, maka semua daya dari  $P_t$ , yang ditransmisikan ke antena via *waveguide* akan diradiasikan. Pada asumsi ini, daya total yang diradiasikan dinyatakan sebagai

$$P_t = \int_{\phi=0}^{2\pi} \int_{\theta=0}^{\pi} U(\phi, \theta) \sin\theta d\theta d\phi \quad (2.1)$$

Kerapatan daya rata  $U_{\text{ave}}$  sebagai

$$U_{\text{ave}} = \frac{P_t}{4\pi} = \frac{1}{4\pi} \int_{\phi=0}^{2\pi} \int_{\theta=0}^{\pi} U(\phi, \theta) \sin\theta d\theta d\phi \quad (2.2)$$

Jika antena meradiasikan daya yang serupa ke semua arah, maka nilai  $U(\phi, \theta)$  akan selaras dengan  $U_{\text{ave}}$ , dan antena dinyatakan seperti isotropis. Antena isotropis berfungsi untuk analisa kinerja. Pada

pengaplikasiannya, antena tentu memiliki daya yang lebih banyak pada arah khusus. Nilai maksimal pada kerapatan daya antena sebagai

$$U_m = \max_{\phi, \theta} \{U(\phi, \theta)\} \quad (2.3)$$

### 2.3 Pola Radiasi Faktor Array

Pola radiasi suatu antena diartikan menjadi ekspresi secara ilustratif yang mengilustrasikan karakter radiasi antena tertentu (pada keadaan medan jauh) pada fungsi arah atau sudut suatu koordinat [1]. Pola radiasi dapat diartikan sebagai pola medan (*field pattern*) jika kuat radiasi yang diilustrasikan adalah kuat medannya dan diartikan pola daya (*power pattern*) jika kuat radiasi yang diilustrasikan merupakan vektor *poyniting*-nya. Pola radiasi dihitung pada medan jauh yang pendistribusian daya angular yang ditransmisikan tak bergantung jarak.

Kondisi medan jauh terwujud jika:  $r > \frac{2D^2}{\lambda}$

$$\begin{aligned} r &\gg D \\ r &\gg \lambda \end{aligned} \quad (2.4)$$

dengan

D= panjang antena

r= jarak titik pusat ke suatu titik observasi

$\lambda$  = panjang gelombang

Pada koordinat bola, medan listrik  $E$  dan medan magnet  $H$  telah didapat, keduanya memiliki unit vektor  $\theta$  dan  $\phi$  yang masing-masing menggambarkan unit sudut *elevasi* dan sudut *azimuth*. Sedangkan vektor *poyniting* dalam koordinat ini hanya memiliki unit radial saja dan dirumuskan sebagai:

$$P_r = \frac{1}{2} \frac{|E|^2}{\eta} \quad (2.5)$$

dengan:

$$|E| = \sqrt{E_\theta^2 + E_\phi^2}, \text{ resultan magnitud gelombang listrik.}$$

$E_\theta$  = unit medan listrik pada arah  $\theta$ .

$E_\varphi$  = unit medan listrik pada arah  $\varphi$ .  
 $\eta$  = impedansi intrinsik ruang bebas ( $377\Omega$ ).

Agar dapat mengilustrasikan pola radiasi sebagai grafis, pola radiasi akan diilustrasikan dalam bentuk relatif. Artinya profil pola yang sudah ternormalisasi, dimana setiap nilai pola radiasi yang terbentuk sudah dibagikan dengan harga maksimumnya. Pola medan, apabila digambarkan dengan pola ternormalisasi dapat digunakan rumus sebagai berikut:

$$P(\phi, \theta) = \frac{E(\phi, \theta)}{E(\phi, \theta)_{max}} \quad (2.6)$$

Pada vektor *poynting* hanya memiliki unit radial dan menurut persamaan (2.1) berbanding lurus atau sebanding dengan hasil dari kuadrat magnitud kuat medannya, maka pola daya apabila digambarkan dalam pola normalisasi, akan serupa dengan kuadrat dari pola medan yang telah ternormalisasi.

$$P(\phi, \theta) = |F(\phi, \theta)|^2 \quad (2.7)$$

Acapkali pula pola radiasi suatu antenna diilustrasikan dan dinyatakan dengan unit *decibel* (dB). Besar medan yang dinyatakan pada unit *decibel* didefinisikan sebagai:

$$F(\phi, \theta)_{dB} = 20 \log |F(\phi, \theta)| \quad (2.8)$$

Sedang pola daya pada *decibel* dirumuskan sebagai:

$$P(\phi, \theta)_{dB} = 10 \log |P(\phi, \theta)|$$

$$P(\phi, \theta)_{dB} = 20 \log |F(\phi, \theta)| \quad (2.9)$$

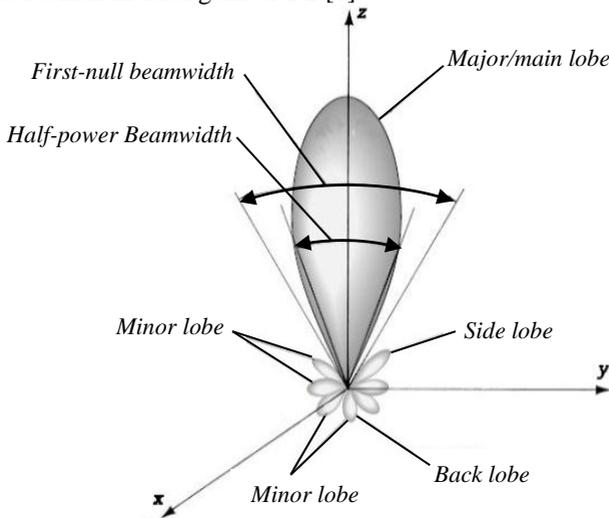
Oleh karena itu, pada satuan *decibel*, pola daya serupa dengan pola medannya. Di gambar 2.2 menggambarkan contoh faktor *array* pada antenna *array* linier pada arah pancaran utama (*main beam*)  $\theta=0^\circ$ .

Agar dapat mengilustrasikan sifat radiasi dari satu antenna pada komposisi *array*, pola radiasi satu antenna selalu dirumuskan dengan

faktor *array* (*array factor*) dengan pola serta sifat radiasi hasil konstruksi antenna *array*, pada elemen-elemen antenna *array* menggunakan elemen isotropis.

### 2.3.1 Parameter Pola Radiasi

Bagian-bagian yang beragam dari pola radiasi dinamakan dengan *lobe*. *Lobe* digolongkan menjadi beberapa fragmen, yakni: *major* atau *main*, *minor*, *sidelobe*, dan *back lobe*. *Radiation lobe* merupakan fragmen dari pola radiasi yang diapit oleh bagian dengan besaran radiasi yang lebih rendah. Ilustrasi tiga dimensi pola radiasi dalam unit polar diilustrasikan melalui gambar 2.2 [1].



**Gambar 2.2** Pola radiasi antenna

#### 2.3.1.1 Sidelobe Level

*Major lobe* (*main beam*) diartikan menjadi *lobe* yang mempunyai sudut dengan radiasi maksimal. Umumnya dengan sebagian *lobe* yang lebih rendah dirasiokan dengan *main lobe*. Semua *lobe* kecuali *main lobe* dinamakan *minor lobe*. *Sidelobe* adalah *lobe* di arah apapun yang tidak sama dari arah *mainlobe* yang ada. Sedangkan *back lobe* adalah *lobe* dengan sumbu z memiliki perbedaan sudut sebesar  $180^\circ$  dan bertolak belakang dengan *main beam* antenna. *Minor lobe* selalu menunjukkan radiasi dari arah yang tidak diinginkan sehingga *minor*

*lobe* harus diminimalkan. *Sidelobe* umumnya adalah minor lobe yang paling besar.

Besaran yang menyatakan tingkat besar daya terkandung pada *sidelobe* disebut dengan *sidelobe level* dimana adalah perbandingan dari besar puncak *sidelobe* dengan besarnya *main lobe*. Level *sidelobe* paling besar untuk seluruh pola adalah puncak *sidelobe level*. Dinyatakan dengan SLL dan satuan dalam desibel. Rumus besarnya SLL dapat dirumuskan sebagai berikut:

$$SLL_{dB} = 20 \log \frac{|F(SLL)|}{|F(max)|} \quad (2.10)$$

dimana:

$|F(SLL)|$  adalah nilai maksimal pada *sidelobe* paling besar

$|F(max)|$  adalah nilai maksimal pada *main lobe*

Pola radiasi sebagian besar antenna menunjukkan pola *lobe* pada berbagai sudut, arah di mana kekuatan sinyal terpancar mencapai maksimum, dipisahkan oleh *nulls*, yaitu sudut di mana kekuatan sinyal terpancar bernilai nol. Dalam antenna direksional dimana tujuannya adalah untuk memancarkan gelombang radio ke satu arah, lobe ke arah tertentu memiliki kekuatan medan yang lebih besar daripada yang lain; hal ini disebut *main lobe*. Lobus lainnya disebut *sidelobe*, dan biasanya mewakili radiasi yang tidak diinginkan dengan arah yang tidak diinginkan. *Sidelobe* ke arah yang berlawanan ( $180^\circ$ ) dari lobus utama disebut *backlobe*.

Dalam mentransmisikan antenna, radiasi *sidelobe* yang berlebihan menghabiskan energi dan dapat menyebabkan gangguan pada peralatan lainnya. Informasi rahasia dapat diambil oleh penerima yang tidak diinginkan. Pada antenna penerima, *sidelobe* dapat mengambil sinyal yang mengganggu, dan meningkatkan tingkat interferensi pada penerima.

Karena pola radiasi medan jauh antenna merupakan transformasi fourier dari ukuran aperturnya, kebanyakan antenna secara umum akan memiliki *sidelobe*, kecuali distribusi aperturennya berupa *gaussian* atau dimensi dan ukuran antenna sangat kecil. Antenna yang lebih besar memiliki *main beam* yang lebih sempit serta *sidelobe* yang lebih sempit. Semakin besar dimensi dan ukuran antenna semakin terlihat *sidelobe* yang terjadi.

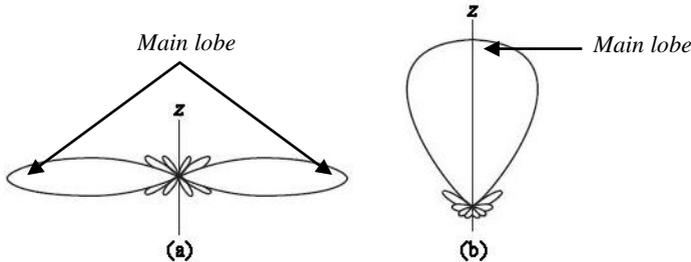
### 2.3.1.2 Half-power Beamwidth

*Half-power Beamwidth* (HPBW) merupakan sudut antar dua poin dari pola radiasi bila daya turun 3 dB atau menjadi setengah daya maksimalnya. HPBW dirumuskan sebagai:

$$HPBW = |\theta_{HPBWleft} - \theta_{HPBWright}| \quad (2.11)$$

Dengan  $\theta_{HPBWleft}$  dan  $\theta_{HPBWright}$  adalah poin-poin pada kiri dan kanan dari beam utama yang pola daya memiliki nilai setengah daya maksimumnya.

Acapkali diinginkan antenna yang memiliki pola radiasi *broadside* atau *endfire*. Satu antenna disebut *broadside*, jika arah maksimal pada beam utamanya berada di arah normal (tegak lurus) bidang antenna. Sedangkan satu antenna dinamakan sebagai *endfire antenna*, jika arah maksimal beam antenanya terdapat pada sepanjang arah bidang antenna. Jika antenna *array* linier yang berada di sepanjang sumbu  $z$ , *broadside* mempunyai arah maksimum pada  $\theta=90^\circ$  dan *endfire* mempunyai arah maksimum pada sudut  $0^\circ$  atau  $180^\circ$ . Gambar 2.3 mengilustrasikan pola radiasi pada antenna *broadside* serta antenna *endfire* [1].



**Gambar 2.3** Pola radiasi polar (a) *broadside*, (b) *endfire*

### 2.3.2 Pola Isotropis, Direksional, dan Omnidireksional

Pola *isotropis* merupakan pola suatu antenna yang mentransmisikan atau menangkap energi elektromagnetik setara ke segala arah. Pola *isotropis* merupakan rancangan ideal yang secara fabrikasi tidak mungkin dilakukan, dapat diilustrasikan sebagai bola dengan pemancar isotropis pada inti bola. Antena *direksional* merupakan antenna yang mentransmisikan atau menangkap energi ke atau dari sudut tertentu lebih banyak dari pada sudut yang lain. Antena *omnidireksional* merupakan antenna yang mempunyai pola tidak

direksional pada permukaan tertentu dan pola direksional pada permukaan lain yang tegak lurus dengan permukaan tersebut.

#### 2.4 Perkalian Pola (*Pattern Multiplication*)

Pada saat merancang suatu antena *array*, elemen-elemen penyusun diperkirakan sebagai sumber titik atau isotropis. Pada dunia nyata, elemen-elemen sebenarnya dari antena *array* adalah antena yang dapat difabrikasi/dibuat contohnya antena *dipole*, antena *helix*, antena parabola, dan lain-lain. Agar didapatkan pola radiasi utuh dari satu konfigurasi *array* pada antena sebenarnya, maka digunakan konsep perkalian pola.

Sudah dipaparkan pada bagian diatas, hasil rancangan suatu *array* menggunakan elemen isotropis, akan didapat kriteria yang disebut dengan faktor *array* (*array factor*) [1]. Serta pola yang didapatkan dari sifat radiasi dari elemen individu yang merupakan antena sebenarnya, dinamakan sebagai faktor elemen (*elemen factor*).

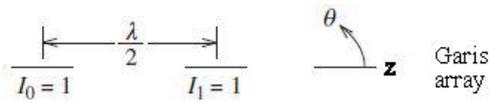
Pola radiasi yang telah dinormalisasi dari suatu antena *array* dirumuskan sebagai:

$$F(\phi, \varphi) = g_a(\phi, \varphi) \times f(\phi, \varphi) \quad (2.12)$$

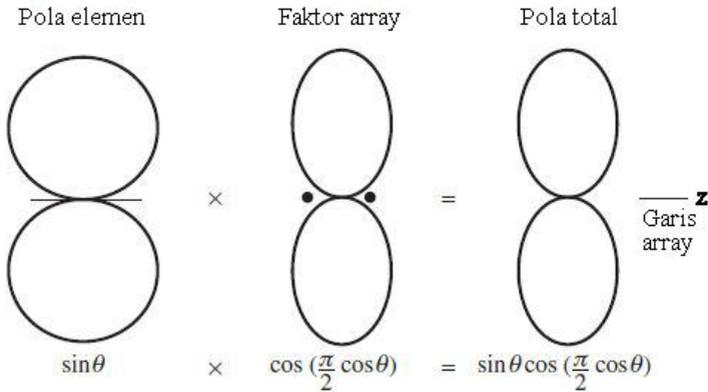
dimana:

$$\begin{aligned} g_a(\phi, \varphi) &= \text{pola elemen} \\ f(\phi, \varphi) &= \text{faktor array} \end{aligned}$$

Untuk mengilustrasikan konsep perkalian pola dijelaskan definisi sebagai berikut, dua buah dipol pendek dipisah sepanjang  $\lambda/2$  dan dicatu daya dengan arus yang sama. Pola elemen untuk suatu elemen pada sepanjang sumbu *array* dinyatakan dengan  $\sin \theta$ , sedangkan faktor *array* dari hasil perancangan desain dua sumber isotropis tadi didapatkan  $\cos[(\pi/2)\cos \theta]$ . Pola total dari susunan antena *array* yang demikian ini adalah  $\sin \theta \cos[(\pi/2)\cos \theta]$ . Konsep perkalian pola dari ilustrasi ini diilustrasikan melalui gambar 2.4 [1].



(a)



(b)

**Gambar 2.4** Array dengan amplitudo dan fasa sama, jarak setengah panjang gelombang (a) array dipol pendek, (b) konsep perkalian pola

### 2.5 Array Antena (Antenna Array)

Penataan beberapa elemen antena yang saling berdekatan di medium udara dapat memberikan hasil pola radiasi yang lebih direksional dibandingkan pola radiasi dari satu elemen antena [1]. Beberapa antena dapat diatur dalam jarak tertentu dan dihubungkan dengan satu sama lain untuk dapat suatu pola radiasi yang diinginkan. Konfigurasi antena seperti ini disebut *antena array*.

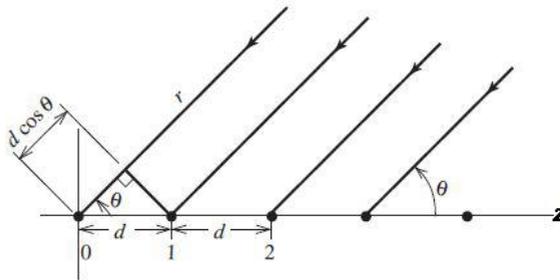
Tata letak elemen dari antena *array* dapat dikonfigurasi dalam berbagai cara. Pada tata letak yang terletak pada satu garis lurus dinamakan *array linier (linear array)*, tata letak yang terletak pada bidang datar dinamakan *array planar*, dan tata letak yang berbentuk lingkaran dinamakan *array lingkaran (circular array)*. Sedangkan jenis *array* yang lain adalah *array konformal (conformal)*, dengan elemen-elemennya terletak pada bidang tak datar.

Faktor *array* dari suatu antena *array* selain diatur oleh tata letak elemen yang dipakai, juga dapat berubah dengan jumlah dan jarak antar elemen, amplitudo daya, dan fasa relatif pada tiap elemen.

Pada elemen-elemen dari antena *array* tanpa melihat dari sisi jenis antena tersebut, dalam persambungannya dengan saluran transmisi menghasilkan pola radiasi penerima atau pancaran yang tepat, tanpa melihat sisi arah pancaran/*beam* tersebut terdapat sumber sinyal maupun tidak. Dengan melaksanakan pengolahan sinyal dari elemen-elemennya, suatu antena *array* dapat menjadi aktif dan bekerja secara optimal terhadap keadaan, menyelaraskan pancaran utama (*main beam*) ke arah sumber yang diperlukan, sementara itu mengarahkan *null-null* ke arah sinyal yang tidak diperlukan. Sehingga memberikan kerja paling optimum pada *output array*. Jenis antena *array* ini dinamakan dengan antena optimal, dan metode ini juga diaplikasikan pada antena adaptif (*adaptive antenna*).

### 2.5.1 Array Antena Linier

Antena *array* linier terdiri dari struktur beberapa elemen antena sehingga membentuk garis lurus. Jarak antar elemen dapat diatur sama atau tidak sama, bergantung permasalahan yang akan diselesaikan.



**Gambar 2.5** Antena *array* linier dengan elemen berjarak sama

Pada gambar 2.5 mengilustrasikan bentuk antena *array* linier yang terdiri dari beberapa elemen, dari elemen ke-0 sampai  $N-1$  [1]. Jarak antar elemen dianggap sama panjang,  $d$ . Sudut  $\theta$  adalah sudut yang dibentuk dari gelombang datang menuju elemen *array* sepanjang sumbu  $z$ . Pancaran dari sumber isotropis sama di segala arah, tetapi ketika keluaran ditambahkan (masing-masing diberi bobot berdasarkan  $I_a$ ), diperoleh respon arah. Fase dari gelombang datang pada *array* dimulai dari 0 jadi  $\zeta_0=0$ . Gelombang datang pada elemen 1 datang sebelum fase

gelombang datang pada sumbu dengan jarak yang lebih pendek  $d \cos\theta$ . Selisih fase antar elemen berdekatan dirumuskan dengan  $\zeta_1 = \beta d \cos\theta$ , dengan  $\zeta_0, \zeta_1, \dots$  merupakan fase gelombang datang di elemen antenna yang ditandai oleh  $0, 1, \dots$ .

*Array factor* pada suatu antenna *array* N-elemen didapatkan dengan mengikutsertakan elemen sebagai titik sumber. Jika setiap elemennya tidak merupakan sumber isotropis, maka total medan yang terbentuk merupakan hasil perkalian antara *array factor* dari sebuah sumber isotropis dengan medan dari elemen tunggal. Untuk *array* linier N-elemen *array factor* yang dapat terbentuk dirumuskan sebagai:

$$AF = I_0 + I_1 e^{j\beta 2d \cos\theta} + \dots = \sum_{n=0}^{N-1} I_n e^{j\beta n d \cos\theta} \quad (2.13)$$

Jika beda fase antar elemen dinyatakan dalam  $\alpha$ , sehingga,

$$I_n = A_n e^{jn\alpha} \quad (2.14)$$

maka:

$$AF = \sum_{n=0}^{N-1} A_n e^{jn(\beta d \cos\theta + \alpha)} \quad (2.15)$$

Dirumuskan  $\psi = \beta d \cos\theta + \alpha$ , maka:

$$AF = \sum_{n=0}^{N-1} A_n e^{jn\psi} \quad (2.16)$$

Apabila eksitasi sama:  $A_0 = A_1 = A_2$ , maka persamaan sebelumnya dapat dirumuskan sebagai berikut:

$$AF = A_0 \sum_{n=0}^{N-1} e^{jn\psi} = A_0 (1 + e^{j\psi} + \dots + e^{j(N-1)\psi}) \quad (2.17)$$

Pengalihan kedua sisi dengan  $e^{j\psi}$  maka persamaan sebelumnya dapat dinyatakan sebagai:

$$AF e^{j\psi} = A_0(e^{j\psi} + e^{j2\psi} + \dots + e^{jN\psi}) \quad (2.18)$$

lalu

$$AF (1 - e^{j\psi}) = A_0(1 - e^{jN\psi})$$

$$AF = \frac{1 - e^{jN\psi}}{1 - e^{j\psi}} A_0 \quad (2.19)$$

$$AF = A_0 \frac{e^{jN\psi} - 1}{e^{j\psi} - 1} = A_0 \frac{e^{jN\psi/2} e^{jN\psi/2} - e^{-jN\psi/2} e^{-jN\psi/2}}{e^{j\psi/2} e^{j\psi/2} - e^{-j\psi/2} e^{-j\psi/2}}$$

$$AF = A_0 e^{j(N-1)\psi/2} \frac{\sin(N\psi/2)}{\sin(\psi/2)} \quad (2.20)$$

Faktor fase:  $e^{j(N-1)\psi/2}$  akan diabaikan, maka *array factor* dapat dirumuskan sebagai:

$$AF = A_0 \frac{\sin(N\psi/2)}{\sin(\psi/2)} \quad (2.21)$$

Maksimum  $AF = AF(\psi = 0) = A_0(1 + 1 + \dots + 1) = A_0N$ .

Maka rumusnya dapat dirumuskan sebagai:

$$f(\psi) = \frac{\sin(N\psi/2)}{N \sin(\psi/2)} \quad (2.22)$$

Pada persamaan (2.22) adalah fungsi dari faktor *array* yang dinormalisasi pada N-elemen dari linear *array* dengan arus amplitudo satu daya yang serba sama, atau sering disebut *Uniform Linear Array* (ULA).

Empat catatan yang dapat dicatat untuk faktor *array* pada berbagai jumlah elemen *array* sebagai berikut:

1. Jika jumlah  $N$  meningkat maka lebar *main beam* dan *main lobe* akan menjadi rapat
2. Penambahan jumlah  $N$  maka akan meningkatkan jumlah *sidelobe* dalam satu periode pada fungsi  $f(\psi)$ . Jumlah semua *lobe*, satu *main lobe* dan sejumlah *sidelobe*, dalam satu periode adalah sejumlah  $N-1$ , maka akan terdapat  $N-2$  *sidelobe* dan satu *main lobe* dalam setiap periode.
3. Lebar *lobe minor* mempunyai lebar  $2\pi/N$ , akan tetapi *lobe* utama dan *grating lobe* mempunyai lebar 2 kali lipat lebar *lobe minor*.
4. Fungsi  $f(\psi)$  akan simetri pada sudut  $\pi$ .

### 2.5.2 Metode *Thinning* pada *Array Antena*

Salah satu metode paling awal untuk mengoptimalkan geometri *array* dikenal sebagai *thinning arrays* atau *array thinning*. Cara ini relatif sederhana. Sebuah antena dengan banyak *array* berukuran seragam (linier atau planar) digunakan sebagai masalah awal. Antena dengan banyak *array* memiliki kerumitan untuk dibangun, memiliki biaya fabrikasi dan pemasangan yang lebih tinggi, serta lebih berat. Oleh karena itu, menghilangkan sejumlah elemen antena dari *array* secara sistematis sangat dibutuhkan terutama jika kinerja dan performa antena *array* tidak berkurang secara signifikan, hal ini disebut *array thinning*.

Sebagai contoh, jika terdapat *array* dengan susunan linier 20 elemen secara seragam dengan jarak setengah panjang gelombang, dengan posisi yang dinyatakan oleh:

$$d = \left[ \pm \frac{\lambda}{4}, \pm \frac{3\lambda}{4}, \dots, \pm \frac{19\lambda}{4}, \right] \quad (2.23)$$

Untuk mengilustrasikan konsep *thinned array*, hapus beberapa elemen dan sedikit ubah posisi elemen. Sebagai catatan *array* awal dan *array* hasil metode *thinning* berukuran sama. Sehingga didapatkan *array* 12 elemen linier dengan posisi dinyatakan oleh:

$$d = \left[ \pm \frac{\lambda}{4}, \pm \frac{3\lambda}{4}, \pm \frac{3\lambda}{2}, \pm \frac{5\lambda}{2}, \pm \frac{7\lambda}{2}, \pm \frac{21\lambda}{5}, \right] \quad (2.24)$$

Beberapa metode *thinning* yang lebih umum adalah sebagai berikut:

1. Metode *thinning* berdasarkan jarak atau kerapatan, metode ini mengurangi *sidelobe* dengan mengurangi magnitud pada tengah *array*.
2. Metode *thinning* secara statistik, untuk *array* sangat banyak. Pada pendekatan ini, probabilitas elemen untuk berada pada posisi tertentu adalah proporsional untuk mendapatkan *thinned array* yang diinginkan, pendekatan ini dapat menghasilkan *array* dengan *sidelobe* rendah.
3. Algoritma optimasi, metode *thinning* dan optimasi posisi seringkali dilakukan menggunakan algoritma optimasi. Contoh algoritma optimasi adalah *Genetic Algorithm* (GA), *Particle Swarm Optimization* (PSO), *Differential Evolution* (DE), serta *Simulated Annealing* (SA). Semua algoritma ini menggunakan pendekatan statistik optimasi untuk memperoleh elemen yang tepat untuk dihilangkan, lalu menghilangkan beberapa elemen jika dapat meningkatkan performa dari *array*. Hal ini sering dilakukan karena mudah diaplikasikan dan dapat menghasilkan hasil yang baik.

## 2.6 *Differential Evolution*

*Differential evolution* adalah satu bentuk teknik atau metode pencarian solusi secara stokastik, bekerja berdasarkan mekanisme yang berkaitan pada proses seleksi alam dan genetika secara alami [2]. Setiap makhluk hidup tersusun atas gen-gen, merupakan bagian dari kromosom yang menentukan atau mempengaruhi karakter atau sifat tiap individu. Mekanisme genetika memproyeksikan kemampuan individu untuk dapat melangsungkan/melakukan perkawinan dan menghasilkan keturunan yang memiliki karakteristik perpaduan dari orang tuanya. Sedangkan konsep dan prinsip seleksi alam dinyatakan bahwa tiap makhluk hidup dapat mempertahankan dirinya sendiri jika mampu menyesuaikan dengan lingkungannya. Dengan demikian, diinginkan keturunan yang dihasilkan mempunyai kombinasi karakteristik yang unik dan terbaik dari orang tuanya, dan dapat mendukung generasi-generasi berikutnya.

*Differential evolution* sering digunakan untuk menyelesaikan masalah yang tidak sederhana, fungsi tujuannya tidak memiliki sifat-sifat yang sederhana seperti sifat tidak kontinu, tidak dapat diturunkan ataupun saat klasifikasi tentang daerah asal jawaban (domain) sangat

sulit atau bahkan tidak ada informasi yang berkaitan sehingga akan meningkatkan kompleksitas baik desain ataupun analisa, pada saat itu *differential evolution* akan terlihat kelebihannya [3], [5], [6].

Metode ini diinspirasi dari teori evolusi Darwin yang bekerja dengan beberapa individu dalam satu populasi, yang setiap individu merupakan kandidat solusi sebuah masalah. Istilah yang ada dalam proses evolusi seperti gen, kromosom, mutasi, reproduksi, dan lain-lain dinyatakan dengan bilangan digunakan untuk memperoleh generasi unik dan terbaik sebagai solusi yang optimum. Dengan istilah yang digunakan pada *differential evolution* dipaparkan pada tabel 2.1 [2].

**Tabel 2.1** Terminologi yang digunakan dalam *differential evolution*

<b>Terminologi</b>	<b>Keterangan</b>
Gen	Nilai yang merupakan satuan dasar yang membentuk satu arti khusus. Dalam <i>differential evolution</i> , gen ini berbentuk nilai biner, <i>float</i> , integer, maupun karakter.
Kromosom	Gabungan beberapa gen yang membentuk nilai khusus.
Individu	Merupakan satu nilai atau keadaan yang menyatakan salah satu solusi yang paling mungkin dari permasalahan yang akan diselesaikan.
Populasi	Sejumlah individu yang diproses bersamaan dalam satu satuan daur proses evolusi.
Generasi	Menyatakan satuan daur proses evolusi
Nilai <i>fitness</i>	Seberapa baik nilai dari suatu individu atau solusi yang didapatkan untuk masalah optimasi tertentu.

Pada *differential evolution* terdapat perbedaan dengan teknik pencarian dan optimasi secara umum, yaitu sebagai berikut:

1. *Differential evolution* berjalan dalam pengkodean pada kumpulan set calon solusi, tidak secara langsung pada solusi masalah.
2. *Differential evolution* mencari pada sejumlah populasi dari solusi, tidak pada satu solusi.
3. *Differential evolution* menerapkan fungsi *fitness* (kecocokan) untuk menilai setiap calon solusi dalam populasi.
4. *Differential evolution* menggunakan aturan probabilistik bukan menggunakan aturan deterministik.

Ada 3 kelebihan dalam penerapan *differential evolution* untuk masalah yang membutuhkan optimasi, sebagai berikut:

1. *Differential evolution* tidak memiliki banyak kebutuhan perhitungan secara matematis dari masalah optimasi. Karena memiliki sifat evolusioner, *differential evolution* akan mencari calon solusi tanpa memperhitungkan pengerjaan suatu masalah di dalamnya secara khusus. *Differential evolution* dapat menyelesaikan berbagai fungsi objektif dengan bermacam-macam batasan, baik linier maupun tidak linier, yang terdapat pada ruang pencarian diskrit, kontinu, ataupun campuran.
2. Sifat acak operator-operator evolusi membuat *differential evolution* lebih efektif pada pencarian global (dalam probabilitas). Pendekatan awal memberikan pencarian lokal dengan prosedur bertahap yang semakin mengerucut, saat perbandingan nilai-nilai dari titik-titik di dekatnya dan berpindah pada titik optimum relatif. Global optimal dapat ditemukan hanya jika satu masalah memiliki sifat khusus yang perlu untuk menjaga satu diantara beberapa lokal optimum merupakan suatu global optimum.
3. *Differential evolution* memberikan fleksibilitas yang besar untuk mengkombinasikan beberapa sifat yang bergantung domain untuk membuat suatu solusi yang cocok untuk optimasi tertentu.

### **2.6.1 Struktur Differential Evolution**

*Differential evolution* berjalan pada sekelompok calon solusi yang dinamakan populasi. Tiap calon solusi dalam populasi dinamakan kromosom. Tiap kromosom terdiri atas sekelompok gen atau bit yang menggambarkan sifat dan karakteristik.

Populasi awal dibangkitkan secara acak, lalu populasi berikutnya adalah hasil evolusi kromosom-kromosom melalui perulangan yang dinamakan generasi. Pada tiap generasi, kromosom akan melalui proses evaluasi dengan menggunakan fungsi objektif dinamakan dengan fungsi *fitness*. Nilai *fitness* dari suatu kromosom akan merepresentasikan kualitas kromosom atau individu dalam populasi tersebut.

Generasi selanjutnya disebut dengan istilah anakan (*offspring*) dibentuk melalui gabungan dua kromosom generasi saat itu yang bersifat sebagai induk (*parent*) dengan melalui operator pindah silang (*crossover*). Selain operator pindah silang, satu kromosom anak dapat juga terbentuk dengan melalui operator mutasi.

### 2.6.2 Representasi Kromosom Solusi

Pada *differential evolution* representasi atau gambaran kromosom diperlukan sebagai perwakilan setiap individu yang berada dalam suatu populasi. Skema perwakilan menentukan bagaimana masalah dibentuk dalam *differential evolution* dan operator genetik yang dipergunakan. Setiap individu atau kromosom disusun dari rangkaian gen dengan urutan khusus yang dapat terdiri atas digit biner (0 dan 1), *floating point*, *integer*, matriks, simbol, dan lain-lain.

Perbandingan *differential evolution* dengan susunan berupa angka riil dan biner didapatkan bahwa dengan angka riil *differential evolution* dapat menjadi lebih efektif, karena dapat memberikan hasil yang lebih sesuai. Akan tetapi pada sisi lain perwakilan biner memiliki kelebihan dalam hal simplisitas. Dengan menggunakan perwakilan biner, model solusi dan penggunaan operator *differential evolution*, seperti *crossover* dan mutasi, menjadi lebih sederhana.

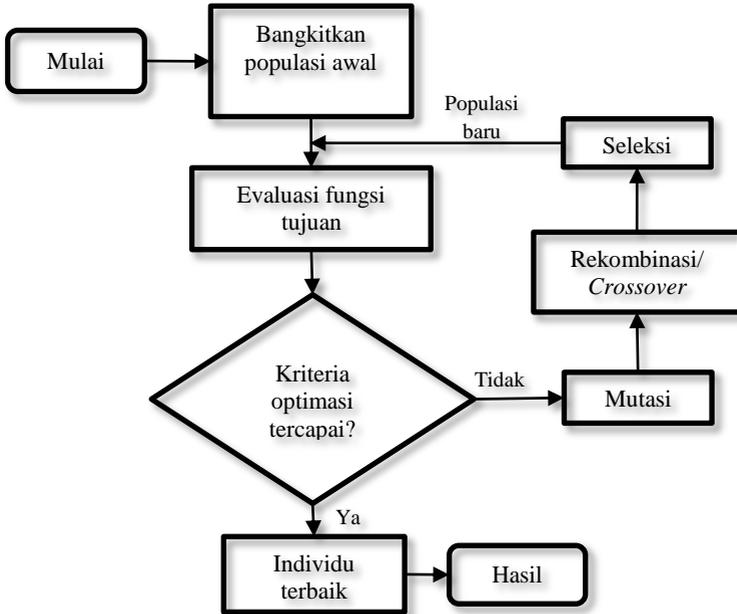
### 2.6.3 Mekanisme Kerja Differential Evolution

Terdapat lima langkah proses kerja dan penggunaan *differential evolution* yaitu membangkitkan populasi awal, evaluasi, mutasi, *crossover*, dan seleksi. Proses ini akan terulang sampai didapat hasil yang diharapkan atau sudah tercapainya kriteria terminasi. Diagram alir *differential evolution* digambarkan pada gambar 2.6.

Pada proses pertama gambar 2.6 [2], *differential evolution* akan membentuk sekelompok individu pada populasi yang acak sebagai kandidat solusi. Pada blok kedua, semua kandidat solusi akan dievaluasi nilainya dengan memasukkan nilai tiap individu ke fungsi objektifnya sehingga diperoleh nilai ketahanan hidupnya (*fitness value*). Pada blok ketiga, nilai individu terbaik dibandingkan dengan kriteria optimasi yang diharapkan, apakah sudah memenuhi dengan syarat yang ditentukan, jika sudah memenuhi maka hasil yang diperoleh akan ditampilkan beserta dengan nilai *fitness*, akan tetapi jika belum memenuhi maka akan dilakukan metode pada blok keempat.

Pada proses keempat dilakukan sejumlah individu dipilih untuk dilakukan mutasi, mutasi ini dapat berpengaruh baik namun dapat juga berpengaruh buruk. Pada proses kelima individu baru diciptakan dari populasi sebelumnya untuk mendapatkan sifat campuran yang memiliki sifat unggul yang merupakan gabungan kedua orang tuanya. Pada proses keenam melakukan seleksi jumlah individu yang lolos menjadi generasi selanjutnya dengan melihat nilai ketahanan hidup generasi sebelumnya.

Jika individu memiliki ketahanan hidup rendah, maka individu tersebut akan tersingkir. Proses ini terus terulang sampai populasi menjadi mengerucut, atau syarat optimisasi telah dipenuhi ataupun jumlah generasi yang ditentukan sudah tercapai.



**Gambar 2.6** Diagram alir *differential evolution*

#### 2.6.4 Pembangkitan Populasi Awal

Populasi awal pada *differential evolution* adalah sekelompok individu yang dipilih dengan dua cara yaitu dengan memilih individu secara acak, atau mengarahkan populasi awal ke suatu keadaan yang lebih mengarah ke pemecahan masalah. Perihal kedua yang perlu dipertimbangkan dalam membangkitkan populasi awal adalah jumlah populasi karena jika jumlah populasi terlalu sedikit maka populasi akan terlalu cepat mencapai konvergensi, konvergensi prematur, sehingga populasi bertumpu pada satu *local optima*, sedangkan jika jumlah populasi terlalu banyak maka perhitungan akan menjadi terlalu rumit dan akan membutuhkan waktu perhitungan yang lama dan menyebabkan hasil perhitungan tidak cepat diperoleh.

### 2.6.5 Evaluasi Fungsi Kebugaran (*Fitness Function*)

Untuk memeriksa hasil optimasi, diperlukan fungsi kebugaran yang memberikan representasi hasil (*solution*) yang telah dikodekan. Selama proses, induk harus digunakan untuk reproduksi, pindah silang dan mutasi untuk mendapatkan keturunan (*offspring*). Jika *differential evolution* dirancang dengan bagus, populasi akan mengalami konvergensi dan mengarah ke sebuah solusi yang optimal. Dalam *differential evolution*, sebuah fungsi kebugaran (*fitness*)  $f(x)$  harus dibuat untuk masing-masing masalah yang akan dipecahkan. Dengan menggunakan kromosom tertentu, fungsi objektif atau fungsi evaluasi akan mengevaluasi dan menilai kondisi masing-masing kromosom. Setiap gen  $x_i$  ( $i=1,2,\dots,N$ ) digunakan untuk menghitung  $f_k(x)$  ( $k=1,2,\dots$ , ukuran populasi).

Pada optimasi saat generasi awal, nilai *fitness* masing-masing individu masih mempunyai jarak yang besar. Selama dengan bertambahnya generasi, beberapa kromosom mengungguli populasi dan menyebabkan rentang nilai *fitness* semakin kecil. Hal ini dapat menyebabkan konvergensi dini.

Masalah dalam *differential evolution* adalah beberapa kromosom dengan nilai kebugaran yang tinggi, tetapi bukan merupakan nilai optimal, mengungguli populasi dan mengakibatkan *differential evolution* mengerucut pada *local minimum*. Ketika konvergen tercapai, kinerja *differential evolution* untuk mencapai solusi yang lebih baik menjadi hilang. Pindah silang antara kromosom induk yang hampir identik akan menghasilkan keturunan yang identik. Dalam hal ini hanya operator mutasi yang dapat menghasilkan kromosom yang relatif baru dan salah satu cara untuk menghindari kromosom yang bukan merupakan solusi optimal mengungguli populasi.

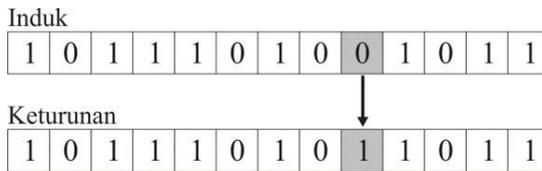
### 2.6.6 Operator Genetik

Ada 2 jenis operator yang melakukan fungsi penting dalam proses *differential evolution* yaitu mutasi dan *crossover* [4], [5]. Jenis dan pengaplikasian dari operator tergantung pada jenis pengkodean yang dipergunakan dan juga masalah optimasi yang diangkat.

Istilah vektor digunakan untuk menyatakan individu. Jumlah populasi sebagai  $M$ , jumlah gen (bit) dalam satu individu sebagai  $N$ , generasi sebagai  $G$ .

### 2.6.6.1 Mutasi

Operator mutasi dipergunakan untuk mengubah satu atau lebih nilai gen dalam satu individu. Cara kerjanya dengan membangkitkan sebuah nilai random  $r_k$  dengan  $k=1,2,\dots$ panjang kromosom. Faktor mutasi ( $F$ ) ditetapkan dan dipergunakan untuk mengatur frekuensi operator mutasi. Apabila nilai random  $r_k < F$  maka gen ke- $k$  kromosom tersebut terpilih untuk mengalami mutasi. Mutasi dengan mengubah gen 0 dengan 1 atau sebaliknya gen 1 menjadi gen 0. Ilustrasi kinerja operator mutasi dapat ditunjukkan pada gambar 2.16 dibawah ini. Faktor mutasi yang bagus berada pada kisaran 0.3 sampai 0.5. Faktor mutasi yang kecil mengakibatkan solusi terkurung pada lokal minimum dan probabilitas yang terlalu besar menyebabkan konvergensi sukar terjadi. Mutasi memperbesar daerah pencarian nilai optimum.



**Gambar 2.7** Ilustrasi operator mutasi

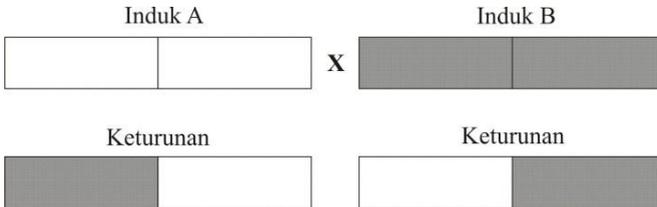
Dari sejumlah individu yang ada dalam populasi dipilih secara random tiga individu yang berbeda satu sama lain. Individu yang terpilih random dinyatakan dengan subskrip  $r_1, r_2, r_3$ . Individu *donor*  $v_{i,G+1}$  diperoleh dengan:

$$v_{i,G+1} = x_{r_1,G} + F(x_{r_2,G} - x_{r_3,G}) \quad (2.25)$$

### 2.6.6.2 Pindah Silang (Crossover)

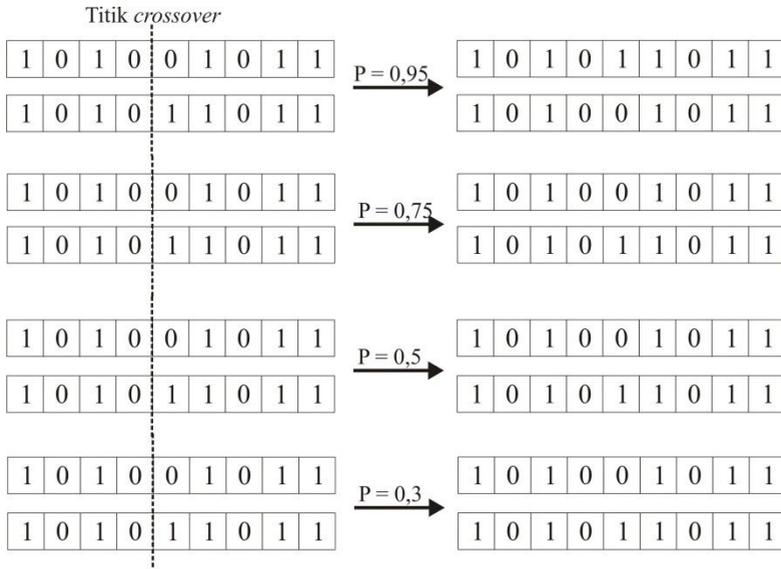
Fungsi dari pindah silang yaitu menghasilkan kromosom anak dari kombinasi materi-materi gen dua kromosom induk. Cara kerjanya dengan membangkitkan sebuah nilai acak  $r_k$  dimana  $k=1,2,\dots$ , ukuran populasi. Hanya beberapa kromosom mengalami proses pindah silang, jumlah kromosom yang mengalami proses pindah silang ditetapkan dari nilai probabilitas nilai pindah silang yang ditetapkan. Kemungkinan pindah silang ( $CR$ ) ditetapkan dan dipergunakan untuk mengatur frekuensi operator pindah silang. Apabila nilai  $r_k < CR$  maka kromosom ke- $k$  akan terpilih untuk mengalami pindah silang. Pindah silang yang

paling sederhana yaitu *one point crossover*. Posisi titik persilangan ditetapkan secara acak pada jarak 1 sampai panjang kromosom. Ilustrasi kinerja operator pindah silang dapat ditunjukkan pada gambar 2.8 dibawah ini.



**Gambar 2.8** Ilustrasi operator crossover

Gambar 2.15 diilustrasikan *one point crossover*. Jika ditetapkan probabilitas *crossover* (CR) 0,9 maka kromosom akan mengalami pindah silang jika nilai acaknya kurang dari probabilitas *crossover* yang telah ditetapkan (kurang dari 0,9).



**Gambar 2.9** Proses dan cara kinerja crossover

Gen individu *trial*,  $u_{i,G+1}$ , didapatkan dari gen individu *target*,  $x_{i,G}$ , dan gen individu *donor*,  $v_{i,G+1}$ . Rekombinasi menggabungkan solusi sukses dari generasi sebelumnya. Gen (bit) individu *donor*,  $v_{i,G+1}$ , menggantikan individu *trial*,  $u_{i,G+1}$ , dengan dipengaruhi oleh probabilitas CR.

$$u_{j,i,G+1} = \begin{cases} v_{j,i,G+1} & \text{jika } rand_{j,i} \leq CR \text{ atau } j = I_{rand} \\ x_{j,i,G} & \text{jika } rand_{j,i} > CR \text{ dan } j \neq I_{rand} \end{cases} \quad (2.26)$$

$$i = 1, 2, \dots, M; \quad j = 1, 2, \dots, N$$

Notasi  $rand_{j,i}$  berupa acak integer dan  $I_{rand}$  yaitu acak integer dari  $[1, 2, \dots, N]$  dan menjamin nilai  $v_{i,G+1} \neq x_{i,G}$ .

### 2.6.7 Proses Seleksi

Proses *differential evolution* yaitu proses seleksi adalah proses memilih kromosom yang akan menjadi generasi berikutnya. Probabilitas seleksi terbentuk berdasarkan nilai *fitness* suatu kromosom, sehingga kromosom yang lebih baik nilai *fitness*-nya memiliki probabilitas terpilih yang lebih besar.

Individu *target*,  $x_{i,G+1}$ , dibandingkan dengan individu *trial*,  $u_{i,G+1}$ , dan salah satu yang memiliki nilai *fitness* terkecil diterima menjadi generasi selanjutnya.

$$x_{i,G+1} = \begin{cases} u_{i,G+1} & \text{jika } f(u_{i,G+1}) \leq f(x_{i,G}) \\ x_{i,G} & \text{yang lain} \end{cases} \quad (2.27)$$

$$i = 1, 2, \dots, M$$

Mutasi, pindah silang (*crossover*), dan seleksi terus dilakukan hingga kriteria terminasi tercapai.

### 2.6.8 Kriteria Terminasi

*Differential evolution* bekerja dari satu generasi ke generasi berikutnya dengan cara menyeleksi dan mereproduksi induk hingga memenuhi kriteria terminasi. Kriteria terminasi yang sering dipakai yaitu dengan memberikan batas maksimum iterasi. Cara terminasi lainnya dengan memasukkan kriteria konvergensi populasi. *Differential evolution* akan memaksa seluruh populasi mengerucut pada suatu solusi.

Jika jumlah perbedaan antar individu semakin kecil dibandingkan dengan nilai batas minimal, *differential evolution* akan berhenti. *Differential evolution* dapat dihentikan karena tidak membaiknya solusi setelah sejumlah iterasi.

## BAB 3

### PEMODELAN SISTEM SIMULASI

Pada bab sebelumnya telah dibahas mengenai teori yang mendasari penelitian tugas akhir ini. Pada bab ini dijelaskan perancangan sistem yang akan dibangun pada penelitian ini yang meliputi parameter simulasi, pemodelan sistem, langkah-langkah proses *differential evolution* pada antena *array* untuk mengurangi *sidelobe level*.

Pada tugas akhir ini akan dioptimasi jumlah elemen aktif dari elemen antena dengan *differential evolution* kemudian dievaluasi nilai *sidelobe level* dari pola radiasi antena *array* yang terbentuk. Elemen antena yang dioptimasi adalah isotropis dengan jumlah elemen berbeda-beda bergantung pada konfigurasi *array* yang akan diuji.

#### 3.1 Parameter Sistem pada Simulasi

Pada simulasi akan digunakan antena *array* linier dengan jumlah total elemen menentukan elemen tertentu yang aktif dan tidak aktif. Di sini digunakan tiga jumlah total *array* linier yaitu:

1. *Array* linier dengan total 15 elemen *array*
2. *Array* linier dengan total 30 elemen *array*
3. *Array* linier dengan total 60 elemen *array*.

Jarak antar elemen ditentukan setengah dari panjang gelombang ( $d = \lambda/2$ ) dengan  $\lambda = c/f$ ,  $c = 3 \times 10^8$  m/s. Untuk proses simulasi *differential evolution* digunakan parameter sebagai berikut,

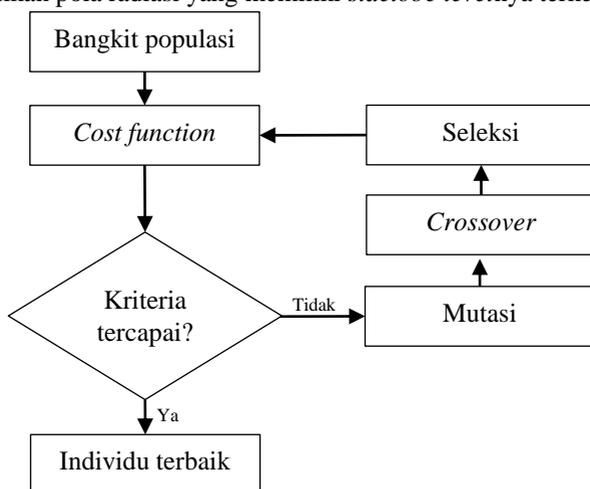
1. Banyaknya jumlah bit per elemen sebanyak 1 bit sehingga panjang kromosom individu adalah jumlah elemen dikalikan dengan jumlah bit per elemen. Misalkan untuk *array* linier dengan total 30 elemen *array*, panjang kromosom yang terbentuk adalah  $30 \times 1$  bit.
2. Banyaknya jumlah kromosom/individu dalam satu populasi adalah 20. Idealnya banyak individu dalam satu populasi sama dengan 10 kali jumlah bit dalam satu individu tetapi pada kerja *differential evolution* ini akan menggunakan 20 kromosom untuk mengurangi waktu konvergensi.
3. Jumlah maksimal generasi adalah 200 kali. Artinya adalah proses algoritma ini akan berulang prosesnya untuk mencari individu paling baik sebanyak maksimum 200 kali. Batas iterasi maksimal ini juga bertindak sebagai kriteria terminasi.

4. Bit bernilai 1 pada elemen ke-k merepresentasikan amplitudo arus eksitasi  $In$  pada elemen ke-k tersebut sebesar 1. Sedangkan Bit bernilai 0 pada elemen ke-k merepresentasikan amplitudo arus eksitasi  $In$  pada elemen ke-k tersebut sebesar 0.

### 3.2 Pemodelan Sistem

Pemodelan sistem yang digunakan dalam antenna *array* linier untuk mengurangi *sidelobe* pada simulasi ini ditunjukkan pada gambar 3.1. Algoritma dijalankan dengan menggunakan software MATLAB R2014b. Proses optimasi pada simulasi ini pada intinya adalah mengoptimasi arus eksitasi yang masuk pada elemen ke-k sehingga kombinasi arus eksitasi pada semua elemen *array* linier yang terbentuk mempunyai pola radiasi yang puncak *sidelobe level*nya minimal.

Terminologi proses differential evolution, satu elemen antenna *array* dinamakan kromosom yang terdiri dari atas bit (gen), satu antenna *array* yang terdiri dari sejumlah elemen dinamakan dengan individu. Program *differential evolution* terdiri dari  $M$  individu yang setiap individu akan mengalami proses evaluasi, mutasi, *crossover*, dan seleksi untuk mendapatkan individu yang mengandung solusi, dimana pola radiasi yang terbentuk dari individu yang mengandung solusi akan menghasilkan pola radiasi yang memiliki *sidelobe level*nya terkecil.



**Gambar 3.1** Diagram Alir Simulasi

### 3.3 Prosedur Simulasi Antena Array Linier menggunakan *Differential Evolution*

Pada subbab ini akan dipaparkan proses dari pengaplikasian *differential evolution*. *Differential evolution* digunakan untuk mencari individu yang memiliki nilai *fitness value* yang optimum. Proses *differential evolution* meliputi pembangkitan individu, evaluasi, mutasi, *crossover*, seleksi dan kriteria terminasi.

#### 3.3.1 Pembangkitan Individu

Langkah pertama yang dilakukan dalam penelitian ini adalah membangkitkan individu-individu yang merepresentasikan kombinasi arus eksitasi yang masuk pada elemen ke- $k$ . Individu-individu ini dibangkitkan secara acak dalam bentuk matriks dengan ukuran  $M \times N$  dengan  $M$  adalah jumlah individu dan  $N$  adalah jumlah bit dalam satu individu,  $N$  dapat dicari dari jumlah bit per elemen dikalikan jumlah elemen dalam satu individu. Dalam tugas akhir ini digunakan antena *array* linier dengan jumlah elemen berbeda-beda dan jumlah bit per elemen / panjang kromosom sebanyak 1, sehingga nilai  $N$  adalah jumlah elemen  $\times$  1 bit per individu. Sedangkan  $M$  adalah jumlah individu dalam satu populasi. Idealnya banyaknya individu dalam satu populasi adalah sama dengan 10 kali dari jumlah bit dari jumlah bit dalam satu individu, tapi pada penyusunan tugas akhir ini nilai  $M$  yang digunakan adalah 20 individu.

#### 3.3.2 Pembentukan Kromosom Individu

Kromosom terdiri dari sekumpulan set gen yang dikelompokkan. Dalam tugas akhir ini, jumlah kromosom yang dievaluasi tiap generasi ditentukan sejumlah 20 kromosom. Satu gen merepresentasikan satu elemen antena, sehingga setiap kromosom terdiri dari 15 gen untuk antena *array* 15 elemen, 30 gen untuk antena *array* 30 elemen dan 60 gen untuk antena *array* 60 elemen.

$$\text{Populasi} = \begin{bmatrix} \text{Kromosom}_1 \\ \text{Kromosom}_2 \\ \text{Kromosom}_3 \\ \vdots \\ \text{Kromosom}_N \end{bmatrix} = \begin{bmatrix} \text{gen}_{11} & \text{gen}_{12} & \dots & \text{gen}_{1M} \\ \text{gen}_{21} & \text{gen}_{22} & \dots & \text{gen}_{2M} \\ \text{gen}_{31} & \text{gen}_{23} & \dots & \text{gen}_{3M} \\ \vdots & \vdots & \ddots & \vdots \\ \text{gen}_{N1} & \text{gen}_{N2} & \dots & \text{gen}_{NM} \end{bmatrix} \quad (3.1)$$

Perintah MATLAB untuk membangkitkan populasi  $M \times N$  dalam matriks adalah:

```

N=30;      %Jumlah kolom/jumlah elemen dalam 1 individu
M=20;      %Jumlah baris/jumlah individu
A=[ones(M,1), randi([0 1], M,N-2), ones(M,1)];% bit
ujung bernilai 1, tengah random 1 dan 0

```

Pada tabel 3.1 dapat dilihat sebagian individu hasil dari proses pembangkitan individu.

**Tabel 3.1** Sebagian individu hasil dari proses pembangkitan individu.

Individu											
1	1	1	1	0	1	1	0	0	0	1	...
2	1	0	0	0	1	1	1	0	0	0	...
3	1	1	0	0	1	0	0	1	1	1	...
...											

### 3.3.3 Konversi Kromosom

Populasi yang dibangkitkan berupa biner (0 dan 1) dengan panjang bit dalam satu individu dinyatakan dengan  $N$ . Sebelum individu-individu ini dinilai, maka tiap individu dikonversi terlebih dahulu dari nilai biner menjadi nilai *far field*.

Konversi dilakukan setiap individu, sehingga konversi dilakukan sebanyak jumlah individu dalam satu populasi. Proses encoding *differential evolution* ini diawali dengan konversi nilai biner yang dibangkitkan menjadi nilai medan jauh (*far field*) dalam dB untuk setiap individu atau kombinasi eksitasi elemen antena *array*. Hal ini didasari oleh persyaratan bahwa jarak setiap elemen antena tidak boleh lebih besar dari  $0.5\lambda$  sehingga dipilih jarak setiap elemen antena sebesar  $0.5\lambda$ .

Dengan menggunakan metode encoding kromosom ini maka dimungkinkan nilai medan jauh (*far field*) dalam dB yang berbeda satu sama lain. Sebagian dari perintah MATLAB untuk proses encoding kromosom adalah.

```

A=[ones(M,1), randi([0 1], M,N-2), ones(M,1)];% bit
ujung bernilai 1, tengah random 1 dan 0

```

```

fix_var= (cos(2*pi*(pos.*cos(phi))+delta_s));
FF_un = abs(A*fix_var);

```

```
Best(Iter,:) = A(1,:)
```

```
for m=1:M
    FFnorm(m,:) = FF_un(m,:) / max(FF_un(m,:));
    FFdB(m,:) = 20 * log10(FFnorm(m,:));
end
```

Program ini akan terus diulang sehingga sampai semua individu yang dibangkitkan mengalami proses konversi nilai kromosom ke nilai *far field*.

### 3.3.4 Evaluasi Individu

Proses evaluasi dipergunakan untuk mengetahui individu yang dibangkitkan memiliki solusi atau tidak memiliki solusi. Solusi yang dituju adalah pola radiasi yang optimal dari antenna *array* yang dibentuk dengan cara menghitung nilai medan jauh (*far field*). Setiap nilai medan jauh (*far field*) individu ini menjadi nilai *cost value* individu tersebut, kemudian dihitung *fitness function* individu tersebut.

$$f \left\{ \begin{matrix} \text{Kromosom}_1 \\ \text{Kromosom}_2 \\ \text{Kromosom}_3 \\ \vdots \\ \text{Kromosom}_N \end{matrix} \right\} = \begin{bmatrix} \text{Cost Function}_1 \\ \text{Cost Function}_2 \\ \text{Cost Function}_3 \\ \vdots \\ \text{Cost Function}_N \end{bmatrix} \quad (3.2)$$

Dalam tugas akhir ini, *cost value* atau nilai PSLN setiap individu diperoleh dari fungsi persamaan medan jauh yang dipergunakan sebagai fungsi objektif yang mengevaluasi setiap individu. *Cost value* yang kecil mengindikasikan bahwa individu tersebut memiliki nilai *fitness* yang tinggi sehingga menjadikan individu tersebut sebagai calon himpunan solusi.

$$FF(u) = \max \left| 2 \sum_{n=1}^N a_n \frac{\cos(2\pi n d u + \delta_s)}{FF_{max}} \text{elpat}(u) \right| \quad (3.3)$$

dimana  $N$  = jumlah elemen *array*,

$a_n$  = amplitudo eksitasi pada elemen,  $n = \begin{cases} 0 & \text{off} \\ 1 & \text{on} \end{cases}$

$d$	= jarak antar elemen
$u$	= $\cos(\phi)$
$\phi$	= sudut garis yang diukur melewati elemen antena
$\delta_s$	= <i>steering phase</i> = $-2\pi ndu_s$
$elpat(u)$	= pola elemen (isotropis=1)
$FF_{max}$	= puncak <i>main beam</i>

Dengan persamaan 3.3, nilai *fitness function* diperoleh dengan mencari total nilai medan jauh (*far field*) yang mempunyai puncak *sidelobe* level paling kecil untuk keseluruhan daerah yang dihitung.

Pada penelitian ini, diinginkan pola radiasi individu memiliki puncak *sidelobe* level sekecil mungkin dengan elemen aktif paling sedikit. Titik pengamatan merupakan titik dimana nilai puncak *sidelobe level* (*fitness value*) individu itu diuji.

Proses yang identik juga diterapkan pada antena dengan 15, 30 dan 60 elemen isotropis. Dari proses evaluasi akan didapatkan satu individu yang mengandung solusi, individu tersebut tersusun elemen-elemen yang memiliki kombinasi nilai eksitasi tiap elemen yang menghasilkan puncak *sidelobe* level terendah.

Representasi elemen *array* ketika bernilai 0 adalah OFF dan 1 adalah ON. Dengan pengurutan *array* dari ujung kiri ke kanan dan elemen pertama dan terakhir ditentukan pada nilai 1 (ON) untuk mencegah ukuran dimensi *array* berubah.

Perintah MATLAB untuk proses evaluasi individu-individu populasi yang dibangkitkan adalah

```
fix_var= (cos(2*pi*(pos.*cos(phi))+delta_s));
FF_un = abs(A*fix_var);
Best(Iter,:) = A(1,:);

for m=1:M
    FFnorm(m,:)=FF_un(m,:)/max(FF_un(m,:));
    FFdB(m,:)=20*log10(FFnorm(m,:));
end
```

### 3.3.5 Proses Mutasi

Operator berikutnya pada *differential evolution* adalah mutasi gen. Operator ini berguna untuk menghadirkan gen yang tidak ada pada

inisialisasi populasi atau pada saat pembangkitan individu. Kromosom individu dimutasi dengan menambahkan nilai random yang sangat kecil dengan kemungkinan yang rendah. Kemungkinan atau probabilitas mutasi ( $F$ ) diartikan sebagai presentasi dari total jumlah gen pada populasi yang mengalami mutasi. Kemungkinan mutasi mengendalikan banyaknya gen baru yang akan dimunculkan untuk dievaluasi. Jika kemungkinan mutasi terlalu kecil, banyak gen yang mungkin berpotensi menjadi solusi tapi tidak pernah dimunculkan. Akan tetapi jika kemungkinan mutasi terlalu besar, maka individu *donor* terlalu banyak gangguan random, sehingga individu *donor* akan kehilangan kemiripan dengan individu *targetnya*.

Mutasi dilakukan untuk memperkaya kombinasi variasi calon solusi yang ada. Cara kerjanya dengan membangkitkan sebuah nilai random  $r_k$  dimana  $k = 1, 2, \dots$ , panjang kromosom. Probabilitas mutasi ( $F$ ) ditentukan di awal dan dipergunakan untuk mengendalikan frekuensi operator mutasi. Apabila nilai random  $r_k < F$  maka gen ke- $k$  kromosom tersebut akan terpilih untuk mengalami mutasi. Mutasi dengan mengganti gen 0 dengan 1 atau sebaliknya gen 1 dengan gen 0. Probabilitas mutasi yang dipergunakan pada tugas akhir ini adalah 0.5.

Perintah MATLAB untuk proses mutasi individu-individu yang dibangkitkan adalah

```
R=randperm(M);
j=R(1);
k=R(2);
p=R(3);
u=R(4);
v=R(5);
if j==m
    j=R(6);
elseif k==m
    k=R(6);
elseif p==m
    p=R(6);
elseif u==m
    u=R(6);
elseif v==m
    v=R(6);
end
V=A(j, :)+F*(A(k, :)-A(p, :));
Vbulat=abs(floor(V));
```

### 3.3.6 Proses Crossover

*Crossover* merupakan operator dari *differential evolution* yang menggunakan indukan untuk membentuk kromosom baru. Operasi ini tidak selalu dikenakan pada semua individu yang ada. Individu terpilih secara random untuk melakukan *crossover* dengan kemungkinan atau probabilitas *crossover* (*CR*) yang telah ditentukan sebelumnya. Prinsip dari *crossover* adalah gen (bit) individu *donor*,  $v_{i,G+1}$ , menggantikan individu *trial*,  $u_{i,G+1}$ , dengan dipengaruhi oleh probabilitas *CR* untuk menghasilkan individu baru yang lebih bervariasi.

Individu yang dihasilkan dari proses ini akan menggantikan individu-individu yang tidak memenuhi kriteria sebagai solusi. Sehingga jumlah individu dalam satu generasi akan berjumlah sama dengan nilai threshold populasi yang telah ditentukan sebelumnya. Jika ditentukan probabilitas *crossover* (*CR*) 0.9, kromosom akan mengalami pindah silang jika nilai randomnya kurang dari probabilitas *crossover* yang telah ditentukan (kurang dari 0.9).

Pada tugas akhir ini proses *crossover* memiliki fungsi menghasilkan mengganti kromosom individu *trial* dengan kromosom individu *donor*. Proses *crossover* terjadi pada setiap elemen dari kromosom tersebut. *Crossover* terjadi dalam setiap individu dalam populasi solusi yang dibangkitkan.

Perintah MATLAB untuk melakukan proses *crossover* pada individu-individu yang dibangkitkan adalah

```
% Crossover, hasilnya masuk ke matriks U
jrand=floor(rand()*N+1);
for n=1:N
    R1=rand();
    if (R1<C || n==jrand)
        U(1,n)=Vbulat(1,n);
    else
        U(m,n)=A(m,n);
    end
end
```

### 3.3.7 Proses Seleksi

Seleksi dipergunakan untuk memilih dan memilah individu-individu mana yang akan terpilih setelah melalui proses *crossover* dan mutasi. Seleksi dipergunakan untuk mendapatkan calon individu induk

yang berpotensi menjadi solusi. Probabilitas seleksi dibentuk berdasarkan nilai fitness *value* suatu individu. Semakin kecil nilai fitness *value*nya, maka semakin besar kemungkinan terpilih karena individu tersebut memiliki nilai puncak *sidelobe* level lebih kecil. Metode seleksi yang dipergunakan pada tugas akhir ini adalah membandingkan secara langsung nilai fitness function dari individu *target*,  $x_{i,G+1}$ , dan individu *trial*,  $u_{i,G+1}$ . Kemudian salah satu yang memiliki nilai *fitness* terkecil diterima menjadi generasi selanjutnya.

Perintah MATLAB untuk melakukan proses seleksi individu-individu yang dibangkitkan adalah

```
if abs(U(1,:) * fix_var) < abs(A(m,:) * fix_var)
    Tr=U(1,:);
else
    Tr=A(m,:);
end
```

### 3.3.8 Penentuan Kriteria Terminasi

*Differential evolution* bekerja dari satu generasi ke generasi selanjutnya dengan mereproduksi individu hingga memenuhi kriteria terminasi. Kriteria terminasi yang dipakai pada tugas akhir ini adalah dengan memberikan batasan maksimum generasi, yaitu dengan nilai maksimum generasi adalah 200 generasi. Jika proses *differential evolution* mencapai salah kriteria tersebut program akan berhenti atau dengan kata lain tercapainya konvergensi.

*[Halaman ini sengaja dikosongkan]*

## BAB 4 ANALISA HASIL SIMULASI

Pada bab ini akan dibahas mengenai analisa hasil simulasi dari proses *differential evolution* pada antena *array* untuk mengurangi *sidelobe level*. Hasil simulasi ini berdasarkan model sistem dan parameter-parameter simulasi yang sudah dibahas pada bab sebelumnya. Proses mengurangi *sidelobe* dari antena *array* linier ini adalah dengan menemukan kombinasi unik dari amplitudo eksitasi masing-masing elemen yang menyusun antena *array* menggunakan metode *differential evolution* dan membandingkan hasilnya dengan antena *array* tanpa optimasi dari kombinasi unik metode *differential evolution*.

Dengan menentukan karakteristik dan sifat antena *array* seperti jarak antar elemen antena *array*, jumlah elemen total antena *array* dan parameter *differential evolution* yang sudah ditentukan, akan dihasilkan simulasi kinerja antena *array* dalam pembentukan pola radiasi dari kombinasi unik amplitudo eksitasi masing-masing elemen penyusun.

Hasil simulasi yang akan ditampilkan dari program *differential evolution* untuk mengurangi *sidelobe level* adalah:

1. Pola radiasi dan *peak sidelobe level* hasil metode *differential evolution* dengan jumlah elemen bervariasi.
2. Perbandingan pola radiasi sebelum dan sesudah metode *differential evolution* diterapkan.
3. Kombinasi unik dari amplitudo eksitasi masing-masing elemen penyusun dengan jumlah elemen bervariasi.

Persebaran data hasil eksperimen meliputi rata-rata (suatu nilai yang mewakili sekumpulan data), standar deviasi (perbedaan data individu terhadap rata-rata sampel), variansi (ukuran seberapa jauh kumpulan bilangan tersebar, variansi nol mengindikasikan bahwa semua nilai sama serta variansi yang rendah mengindikasikan bahwa suatu data berdekatan dengan nilai rata-rata), serta nilai maksimal dan minimal.

**Tabel 4.1** Parameter *differential evolution*

No	Parameter	Jumlah	Keterangan
1	Jumlah populasi	20	Jumlah individu dalam populasi
2	Jumlah gen	30	Jumlah gen dalam individu
3	Probabilitas mutasi	0,5	Probabilitas terjadinya mutasi
4	Probabilitas <i>crossover</i>	0,9	Probabilitas terjadinya <i>crossover</i>
5	Maksimum iterasi	200	Jumlah iterasi maksimal

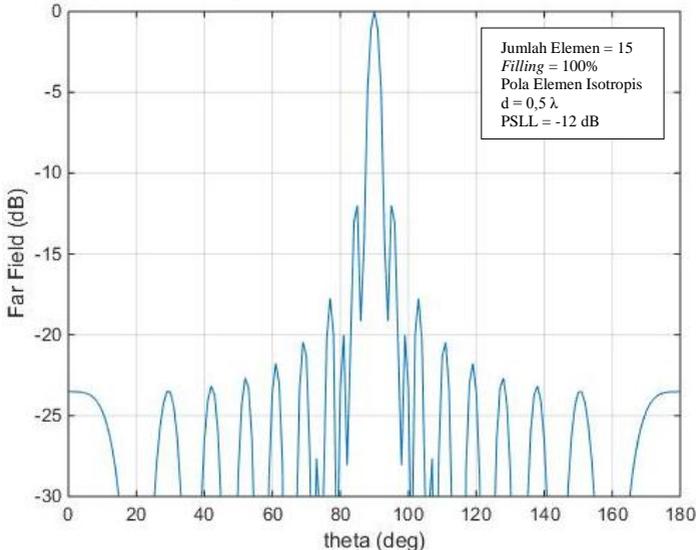
6	Jumlah eksperimen	30	Jumlah eksperimen
---	-------------------	----	-------------------

#### 4.1 Hasil Simulasi Antena Array Linier 15 Elemen Isotropis

Dalam simulasi ini, elemen *array* yang menyusun *array* linier merupakan elemen *array* isotropis. Pertama akan disimulasikan 15 elemen isotropis tanpa metode *differential evolution* (DE). Simulasi berikutnya adalah penggunaan metode DE untuk mendapatkan kombinasi unik terbaik yang memiliki pucak level *sidelobe* paling kecil. Kemudian membandingkan hasil simulasi tanpa metode DE dan dengan metode DE.

##### 4.1.1 Hasil Simulasi 15 Elemen Isotropis Tanpa Metode DE

Hasil simulasi 15 elemen isotropis tanpa optimasi, dengan jumlah elemen aktif yang tereksitasi sebanyak 15 elemen atau 100 % dari jumlah keseluruhan dan jarak antar elemen adalah  $0,5 \lambda$ . Nilai pucak level *sidelobe* atau *Peak Sidelobe Level* (PSLL) adalah -12 dB pada arah *broadside*. *Dense array* adalah *array* yang masih aktif (tereksitasi) semua elemen penyusunnya, dapat dikatakan bahwa antena *array* sebelum dioptimasi merupakan *dense array* (*filling* 100%).



**Gambar 4.1** *Dense array* dari 15 elemen isotropis tanpa metode DE

#### 4.1.2 Hasil Simulasi 15 Elemen Isotropis dengan Metode DE

Simulasi 15 elemen isotropis dengan metode DE dilakukan sebanyak 30 kali eksperimen. Dari 30 kali eksperimen tersebut dipilih satu kombinasi gen individu yang memiliki nilai PSLL paling kecil, satu kombinasi gen individu yang memiliki nilai aktif total elemennya (*filling*) paling kecil dan satu kombinasi gen individu yang memiliki sifat diantara keduanya (PSLL dan persentase *filling* yang kecil).

Rata-rata, standar deviasi, variansi, nilai maksimal dan minimal ditunjukkan oleh tabel 4.2 dengan hasil 30 kali eksperimen DE 15 elemen isotropis pada tabel 4.3. Gambar 4.2 menunjukkan plot elemen aktif (%) terhadap PSLL (dB) dari 30 kali hasil eksperimen DE 15 elemen.

**Tabel 4.2** Persebaran data hasil eksperimen DE 15 elemen isotropis

Parameter	PSLL (dB)	Iterasi ke-	Elemen aktif
Rata-rata	-16,8941	65,33	80,78 %
Standar deviasi	0,9102	46,71	6,41 %
Variansi	0,8285	2181,61	41,14 %
Maksimal	-13,9170	168	86,67 %
Minimal	-18,2010	7	56,67 %

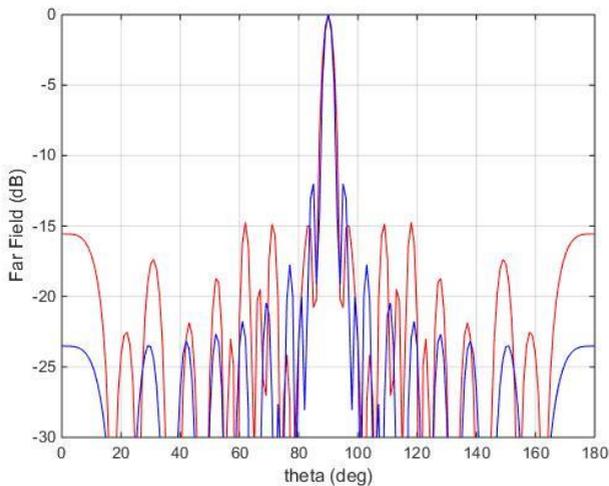
**Tabel 4.3** Hasil 30 eksperimen DE 15 elemen isotropis

Run ke-	PSLL (dB)	Iterasi ke-	Elemen aktif
1	-13,5893	12	86,67 %
2	-14,2157	10	80,00 %
3	-14,2157	5	80,00 %
4	-14,5365	2	86,67 %
5	-14,7442	12	80,00 %
6	-14,7466	4	86,67 %
7	-14,5365	23	86,67 %
8	-14,7442	7	80,00 %
9	-14,7466	53	86,67 %
10	-14,7442	27	80,00 %
11	-14,2157	11	80,00 %
12	-14,2157	17	80,00 %
13	-12,6361	2	80,00 %
14	-12,6631	2	73,33 %
15	-14,7442	21	80,00 %
16	-14,7466	5	86,67 %

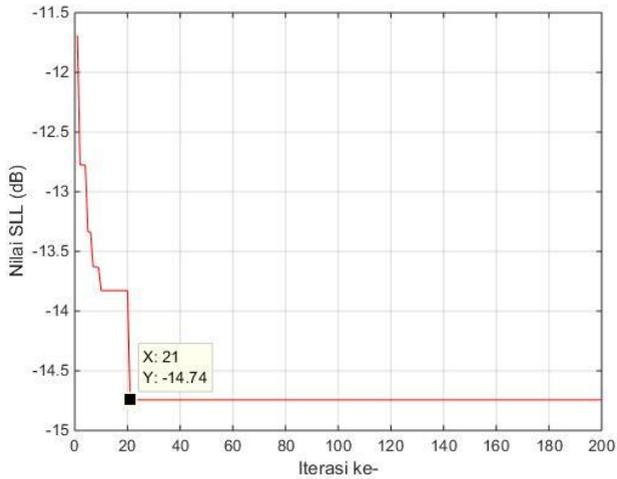


gen individu pertama yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.3. Konvergensi gen individu pertama yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.4.

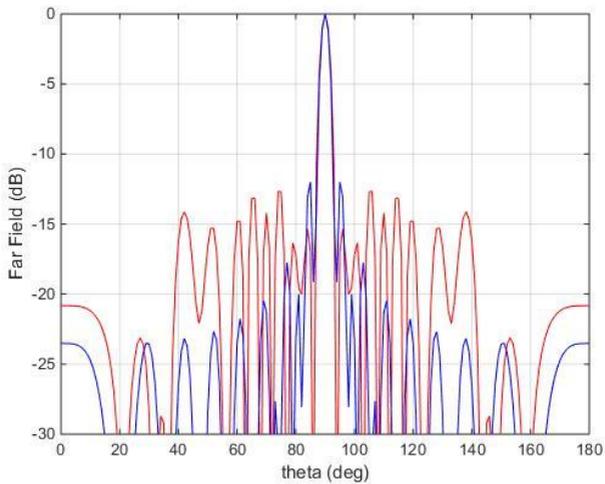
Kombinasi gen individu kedua memiliki nilai PSLL dan *filling* kecil adalah 11011111100011, dengan nilai PSLL -12,6631 dB (gambar 4.5). Terjadi konvergensi saat iterasi ke-2 (gambar 4.6), memiliki *filling* sebesar 73,33 % dari total 15 elemen yang ada. Pola radiasi gen individu kedua yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.5. Konvergensi gen individu kedua yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.6.



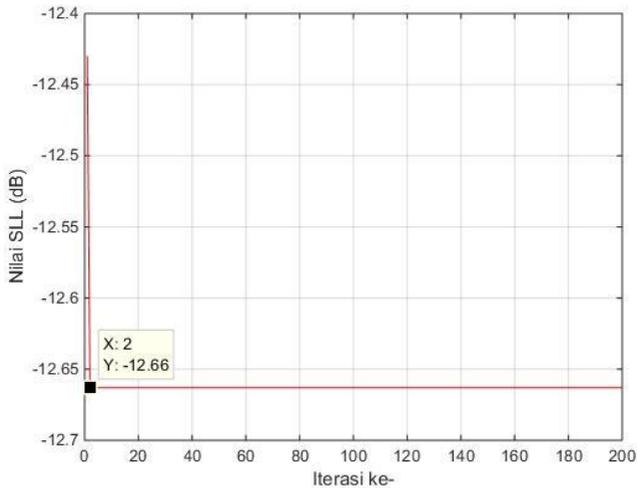
**Gambar 4.3** Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen isotropis



**Gambar 4.4** Konvergensi gen individu pertama yang terpilih pada DE 15 elemen isotropis

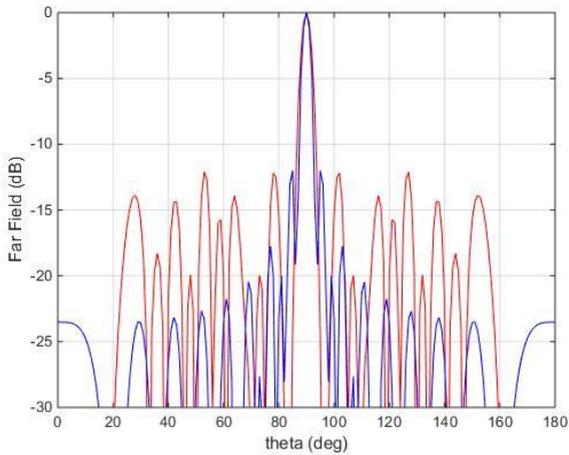


**Gambar 4.5** Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen isotropis

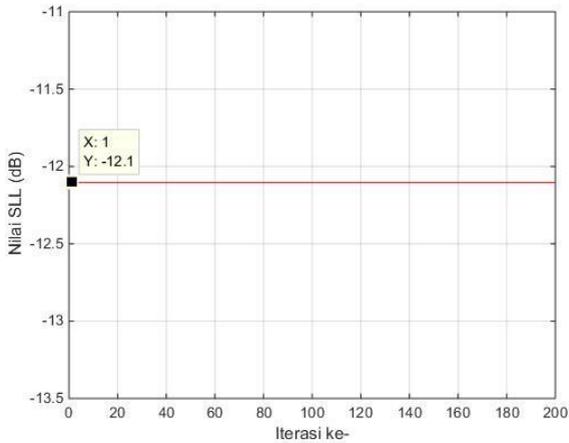


**Gambar 4.6** Konvergensi gen individu kedua yang terpilih pada DE 15 elemen isotropis

Kombinasi gen individu ketiga memiliki nilai *filling* paling kecil adalah 111111110001001, dengan nilai PSSL -12,1037 dB (gambar 4.7). Terjadi konvergensi saat iterasi pertama (gambar 4.8), memiliki *filling* sebesar 66,67 % dari total 15 elemen yang ada. Pola radiasi gen individu ketiga yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.7. Konvergensi gen individu ketiga yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.8.



**Gambar 4.7** Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen isotropis



**Gambar 4.8** Konvergensi gen individu ketiga yang terpilih pada DE 15 elemen isotropis

Perbandingan hasil simulasi 15 elemen tanpa DE dan dengan DE. Kombinasi gen individu pertama, 11111111110001, dapat menurunkan

PSLL sebesar -2,7442 dB dengan tiga elemen dinonaktifkan. Kombinasi gen individu kedua, 11011111100011, dapat menurunkan PSLL sebesar -0,6631 dB dengan empat elemen dinonaktifkan. Kombinasi gen individu ketiga, 111111110001001, dapat menurunkan PSLL sebesar -0,1037 dB dengan lima elemen dinonaktifkan.

Dari ketiga kombinasi gen individu diatas sudah dapat menghasilkan nilai PSLL yang lebih kecil dibandingkan nilai PSLL sebelum diterapkannya metode DE, dengan lebar *beam* tidak berubah signifikan. Pemilihan jumlah elemen yang aktif dapat ditentukan sesuai dengan kebutuhan dari penggunaan antenna *array*. Tabel 4.4 menunjukkan kombinasi gen yang terpilih dari 15 elemen antenna *array*.

**Tabel 4.4** Kombinasi gen yang terpilih pada DE 15 elemen isotropis

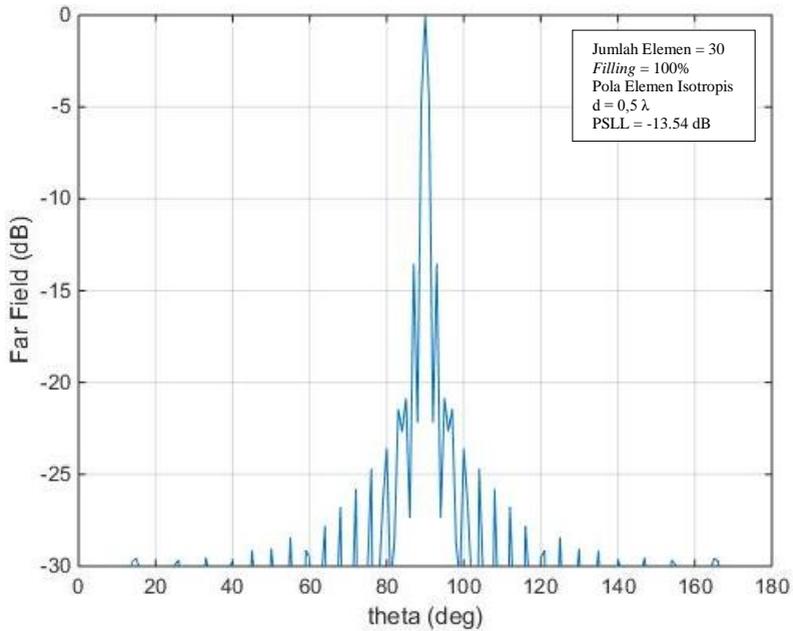
No	Kombinasi Gen	PSLL	Iterasi ke-	Elemen Aktif
1	111111111110001	-14,7442 dB	21	80,00 %
2	110111111100011	-12,6631 dB	2	73,33 %
3	111111110001001	-12,1037 dB	1	66,67 %

## 4.2 Hasil Simulasi Antena Array Linier 30 Elemen Isotropis

Dalam simulasi ini, elemen *array* yang menyusun *array* linier merupakan elemen *array* isotropis. Pertama akan disimulasikan 30 elemen isotropis tanpa metode *differential evolution* (DE). Simulasi berikutnya adalah penggunaan metode DE untuk mendapatkan kombinasi unik terbaik yang memiliki pucak level *sidelobe* paling kecil. Kemudian membandingkan hasil simulasi tanpa metode DE dan dengan metode DE.

### 4.2.1 Hasil Simulasi 30 Elemen Isotropis Tanpa Metode DE

Hasil simulasi 30 elemen isotropis tanpa optimasi, dengan jumlah elemen aktif yang tereksitasi sebanyak 30 elemen atau 100 % dari jumlah keseluruhan dan jarak antar elemen adalah  $0,5 \lambda$ . Nilai pucak level *sidelobe* atau *Peak Sidelobe Level* (PSLL) adalah -13.54 dB pada arah *broadside*.



**Gambar 4.9** Dense array dari 30 elemen isotropis tanpa metode DE

#### 4.2.2 Hasil Simulasi 30 Elemen Isotropis dengan Metode DE

Simulasi 30 elemen isotropis dengan metode DE dilakukan sebanyak 30 kali eksperimen. Dari 30 kali eksperimen tersebut dipilih satu kombinasi gen individu yang memiliki nilai PSLL paling kecil, satu kombinasi gen individu yang memiliki nilai aktif total elemennya (*filling*) paling kecil dan satu kombinasi gen individu yang memiliki sifat diantara keduanya (PSLL dan persentase *filling* yang kecil).

Rata-rata, standar deviasi, variansi, nilai maksimal dan minimal ditunjukkan oleh tabel 4.5 dengan hasil 30 kali eksperimen DE 30 elemen isotropis pada tabel 4.6. Gambar 4.10 menunjukkan plot elemen aktif (%) terhadap PSLL (dB) dari 30 kali hasil eksperimen DE 30 elemen.

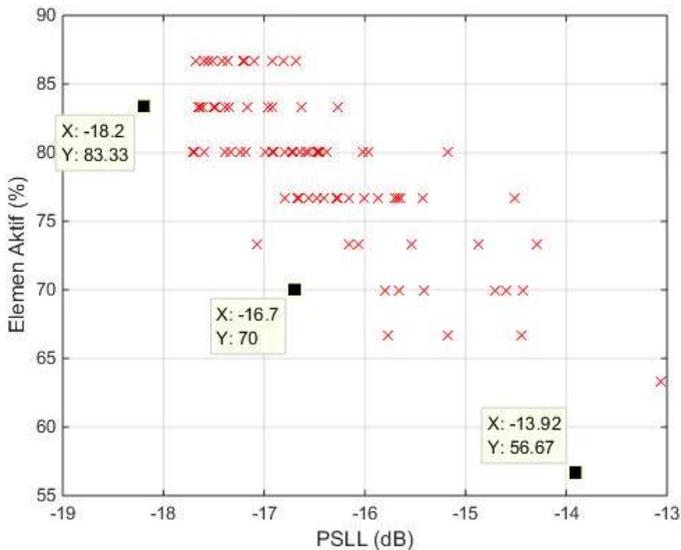
**Tabel 4.5** Persebaran data hasil eksperimen DE 30 elemen isotropis

Parameter	PSLL (dB)	Iterasi ke-	Elemen aktif
-----------	-----------	-------------	--------------

Rata-rata	-16,8941	65,33	80,78 %
Standar deviasi	0,9102	46,71	6,41 %
Variansi	0,8285	2181,61	41,14 %
Maksimal	-13,9170	168	86,67 %
Minimal	-18,2010	7	56,67 %

**Tabel 4.6** Hasil 30 eksperimen DE 30 elemen isotropis

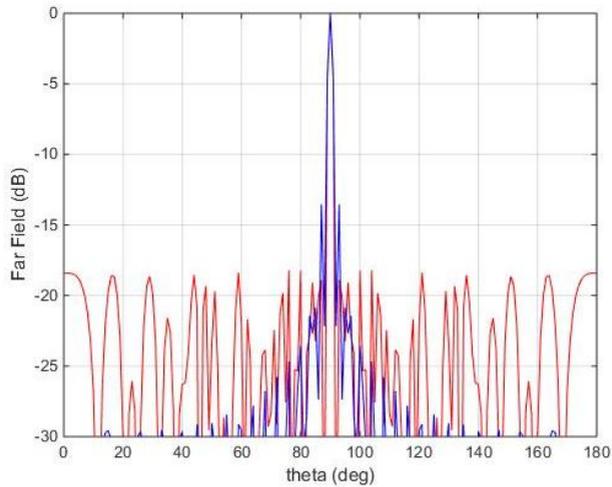
Run ke-	PSLL (dB)	Iterasi ke-	Elemen aktif
1	-18,2010	11	83,33 %
2	-17,6543	43	83,33 %
3	-17,4177	147	86,67 %
4	-17,5104	17	86,67 %
5	-16,3806	17	80,00 %
6	-17,5939	43	86,67 %
7	-16,4477	40	80,00 %
8	-17,7082	34	80,00 %
9	-16,7903	145	76,67 %
10	-17,6129	86	83,33 %
11	-16,6253	46	83,33 %
12	-17,1647	104	83,33 %
13	-17,5570	79	86,67 %
14	-16,6627	16	76,67 %
15	-15,9723	7	80,00 %
16	-17,6964	168	80,00 %
17	-16,1594	65	76,67 %
18	-17,3925	52	83,33 %
19	-17,6147	23	83,33 %
20	-17,2402	113	80,00 %
21	-16,9940	72	80,00 %
22	-16,6869	52	86,67 %
23	-14,8726	44	73,33 %
24	-17,0983	138	86,67 %
25	-15,5369	19	73,33 %
26	-16,9151	95	86,67 %
27	-17,5059	65	83,33 %
28	-17,1903	138	86,67 %
29	-16,7025	67	70,00 %
30	-13,917	14	56,67%



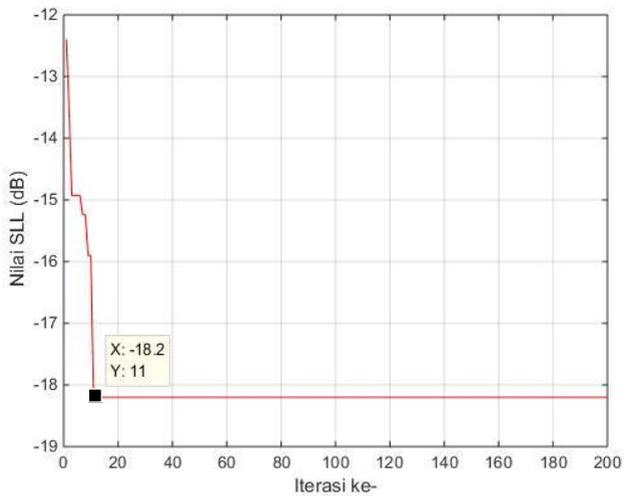
**Gambar 4.10** Plot hasil 30 kali eksperimen DE 30 elemen isotropis

Kombinasi gen individu pertama memiliki nilai PSLL paling kecil adalah 11111111101111 011011101011111, dengan nilai PSLL -18,2010 dB (gambar 4.11). Terjadi konvergensi saat iterasi ke-11 (gambar 4.12), memiliki *filling* sebesar 83,33 % dari total 30 elemen yang ada. Pola radiasi gen individu pertama yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.11. Konvergensi gen individu pertama yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.12.

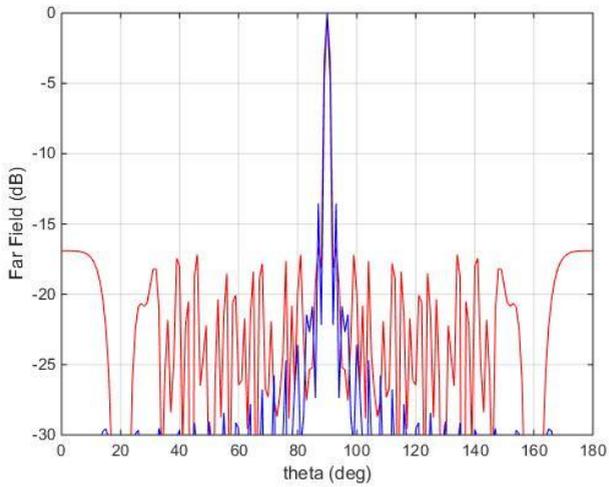
Kombinasi gen individu kedua memiliki nilai PSLL dan *filling* kecil adalah 10111111111101 110111100001001, dengan nilai PSLL -16,7025 dB (gambar 4.13). Terjadi konvergensi saat iterasi ke-67 (gambar 4.14), memiliki *filling* sebesar 70,00 % dari total 30 elemen yang ada. Pola radiasi gen individu kedua yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.13. Konvergensi gen individu kedua yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.14.



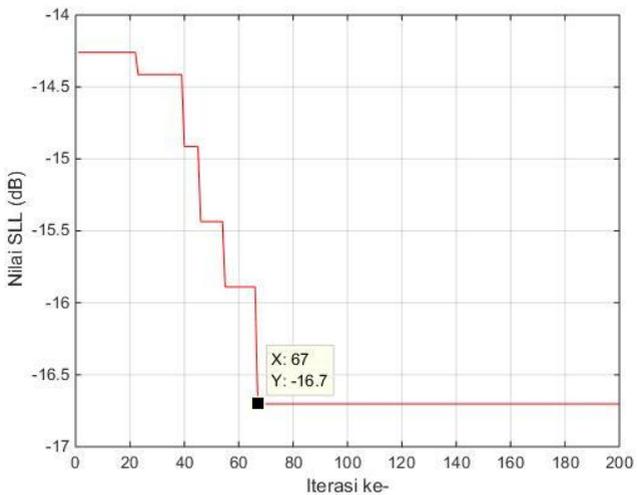
**Gambar 4.11** Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen isotropis



**Gambar 4.12** Konvergensi gen individu pertama yang terpilih pada DE 30 elemen isotropis

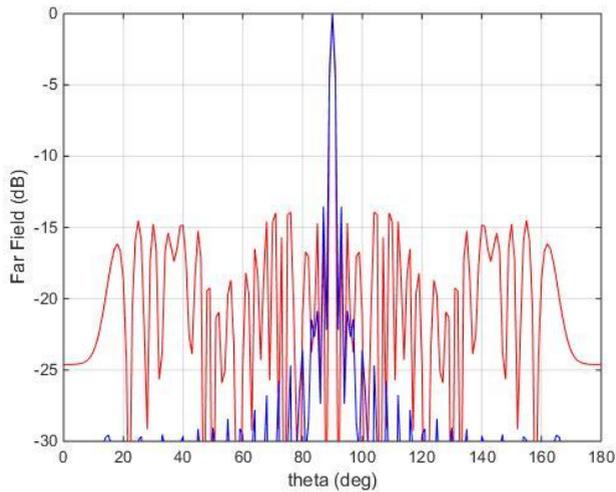


**Gambar 4.13** Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen isotropis

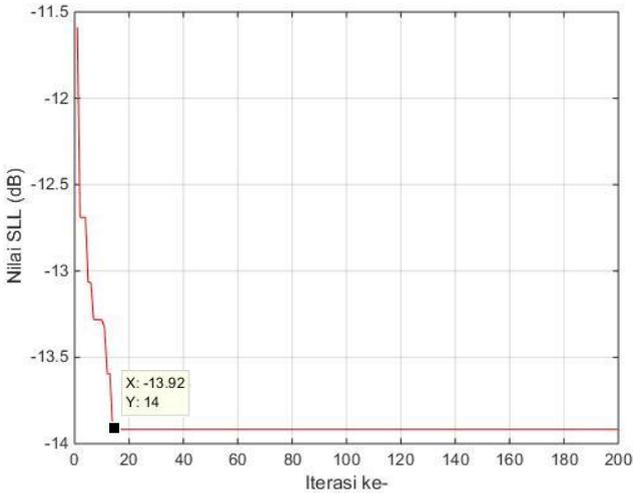


**Gambar 4.14** Konvergensi gen individu kedua yang terpilih pada DE 30 elemen isotropis

Kombinasi gen individu ketiga memiliki nilai *filling* paling kecil adalah 110101111001000 111101011100001, dengan nilai PSLL - 13,917 dB (gambar 4.15). Terjadi konvergensi saat iterasi ke-14 (gambar 4.16), memiliki *filling* sebesar 56,67 % dari total 30 elemen yang ada. Pola radiasi gen individu ketiga yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.15. Konvergensi gen individu ketiga yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.16.



**Gambar 4.15** Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen isotropis



**Gambar 4.16** Konvergensi gen individu ketiga yang terpilih pada DE 30 elemen isotropis

Perbandingan hasil simulasi 30 elemen tanpa DE dan dengan DE. Kombinasi gen individu pertama, dapat menurunkan PSLL sebesar -4,6610 dB dengan 5 elemen dinonaktifkan. Kombinasi gen individu kedua, dapat menurunkan PSLL sebesar -3,1625 dB dengan 9 elemen dinonaktifkan. Kombinasi gen individu ketiga, dapat menurunkan PSLL sebesar -0,3770 dB dengan 13 elemen dinonaktifkan.

Dari ketiga kombinasi gen individu diatas sudah dapat menghasilkan nilai PSLL yang lebih kecil dibandingkan nilai PSLL sebelum diterapkannya metode DE, dengan lebar *beam* tidak berubah signifikan. Pemilihan jumlah elemen yang aktif dapat ditentukan sesuai dengan kebutuhan dari penggunaan antenna *array*. Tabel 4.3 menunjukkan kombinasi gen yang terpilih dari 30 elemen antenna *array*.

**Tabel 4.7** Kombinasi gen yang terpilih pada DE 30 elemen isotropis

No	Kombinasi Gen	PSLL	Iterasi ke-	Elemen Aktif
1	111111111101111 011011101011111	-18,2010 dB	11	83,33 %
2	101111111111101 110111100001001	-16,7025 dB	67	70,00 %

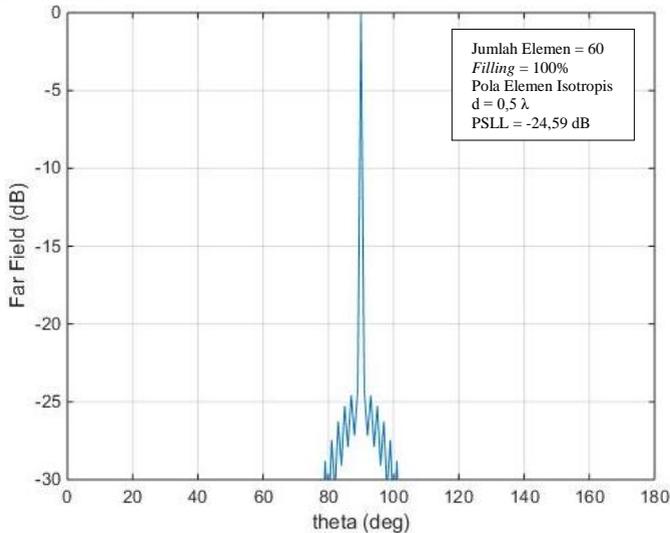
3	110101111001000 111101011100001	-13,917 dB	14	56,67 %
---	------------------------------------	------------	----	---------

### 4.3 Hasil Simulasi Antena Array Linier 60 Elemen Isotropis

Dalam simulasi ini, elemen *array* yang menyusun *array* linier merupakan elemen *array* isotropis. Pertama akan disimulasikan 60 elemen isotropis tanpa metode *differential evolution* (DE). Simulasi berikutnya adalah penggunaan metode DE untuk mendapatkan kombinasi unik terbaik yang memiliki pucak level *sidelobe* paling kecil. Kemudian membandingkan hasil simulasi tanpa metode DE dan dengan metode DE.

#### 4.3.1 Hasil Simulasi 60 Elemen Isotropis Tanpa Metode DE

Hasil simulasi 60 elemen isotropis tanpa optimasi, dengan jumlah elemen aktif yang tereksitasi sebanyak 60 elemen atau 100 % dari jumlah keseluruhan dan jarak antar elemen adalah  $0,5 \lambda$ . Nilai pucak level *sidelobe* atau *Peak Sidelobe Level* (PSLL) adalah -24,59 dB pada arah *broadside*.



**Gambar 4.17** Dense array dari 60 elemen isotropis tanpa metode DE

### 4.3.2 Hasil Simulasi 60 Elemen Isotropis dengan Metode DE

Simulasi 60 elemen isotropis dengan metode DE dilakukan sebanyak 30 kali eksperimen. Dari 30 kali eksperimen tersebut dipilih satu kombinasi gen individu yang memiliki nilai PSLL paling kecil, satu kombinasi gen individu yang memiliki nilai aktif total elemennya (*filling*) paling kecil dan satu kombinasi gen individu yang memiliki sifat diantara keduanya (PSLL dan persentase *filling* yang kecil).

Rata-rata, standar deviasi, variansi, nilai maksimal dan minimal ditunjukkan oleh tabel 4.8 dengan hasil 30 kali eksperimen DE 30 elemen isotropis pada tabel 4.9. Gambar 4.18 menunjukkan plot elemen aktif (%) terhadap PSLL (dB) dari 30 kali hasil eksperimen DE 60 elemen.

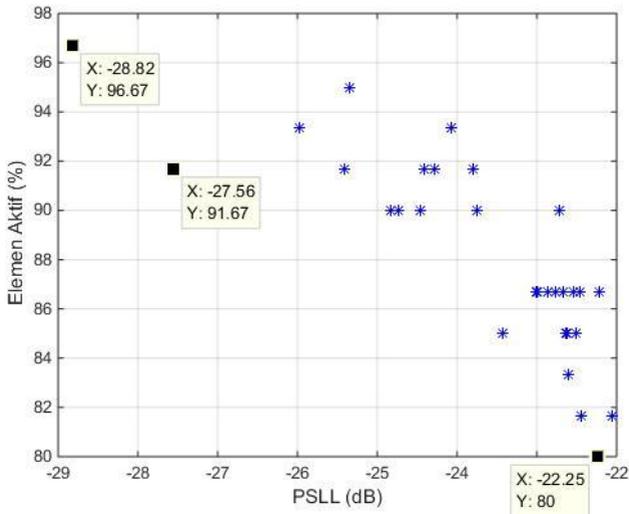
**Tabel 4.8** Persebaran data hasil eksperimen DE 60 elemen isotropis

Parameter	PSLL (dB)	Iterasi ke-	Elemen aktif
Rata-rata	-23,7420	147,5	88,22 %
Standar deviasi	1,6293	47,1	4,08 %
Variansi	2,6548	2217,2	16,65 %
Maksimal	-22,0630	200,0	96,67 %
Minimal	-28,8248	42,0	80,00 %

**Tabel 4.9** Hasil 30 eksperimen DE 60 elemen isotropis

Run ke-	PSLL (dB)	Iterasi ke-	Elemen aktif
1	-22,6732	147	86,67
2	-23,4317	127	85,00
3	-22,4658	47	86,67
4	-22,7648	94	86,67
5	-22,4361	135	81,67
6	-27,5613	193	91,67
7	-22,5363	143	86,67
8	-25,4073	138	91,67
9	-22,0630	42	81,67
10	-22,5039	189	85,00
11	-22,5980	123	83,33
12	-22,7102	192	90,00
13	-25,3520	178	95,00
14	-22,8709	179	86,67
15	-22,6312	156	85,00
16	-23,7569	180	90,00

17	-22,6145	171	85,00
18	-24,7411	181	90,00
19	-22,2543	108	80,00
20	-22,2114	67	86,67
21	-24,0682	196	93,33
22	-22,9909	200	86,67
23	-23,0125	194	86,67
24	-23,7990	110	91,67
25	-24,8257	144	90,00
26	-24,4175	181	91,67
27	-24,2896	198	91,67
28	-25,9815	66	93,33
29	-28,8248	177	96,67
30	-24,4658	170	90,00

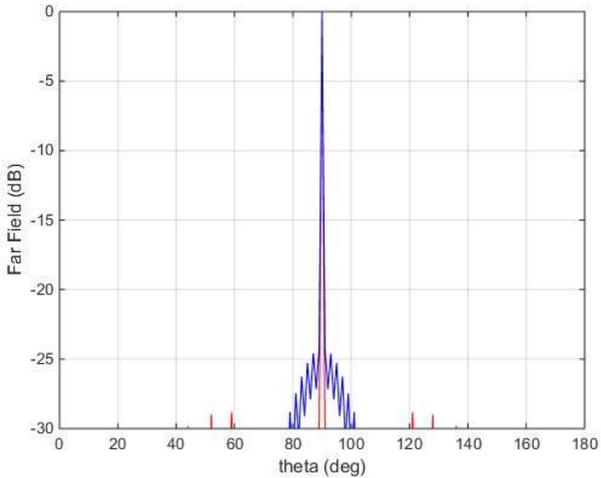


**Gambar 4.18** Plot hasil 30 kali eksperimen DE 60 elemen isotropis

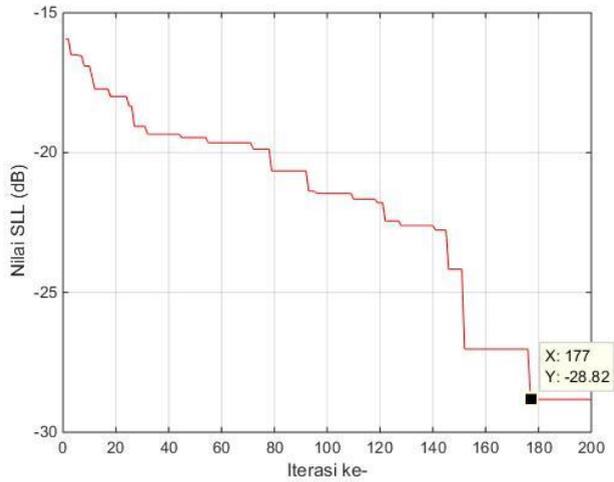
Kombinasi gen individu pertama memiliki nilai PSLL paling kecil adalah 1111111111111111 1111111111111111 1111111111111111 111111111111001, dengan nilai PSLL -28,8248 dB (gambar 4.19). Terjadi konvergensi saat iterasi ke-177 (gambar 4.20), memiliki *filling*

sebesar 96,67 % dari total 60 elemen yang ada. Pola radiasi gen individu pertama yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.19. Konvergensi gen individu pertama yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.20.

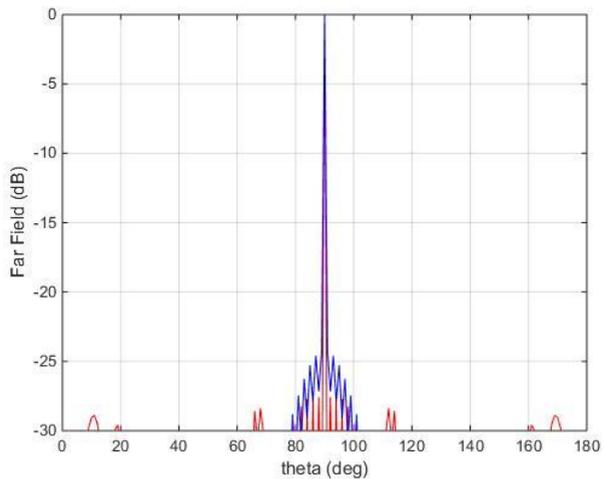
Kombinasi gen individu kedua memiliki nilai PSLL dan *filling* kecil adalah 1011111111111111 1111111111111111 1111111111111111 11111111000011, dengan nilai PSLL -27,5613 dB (gambar 4.21). Terjadi konvergensi saat iterasi ke-193 (gambar 4.22), memiliki *filling* sebesar 91,67 % dari total 60 elemen yang ada. Pola radiasi gen individu kedua yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.21. Konvergensi gen individu kedua yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.22.



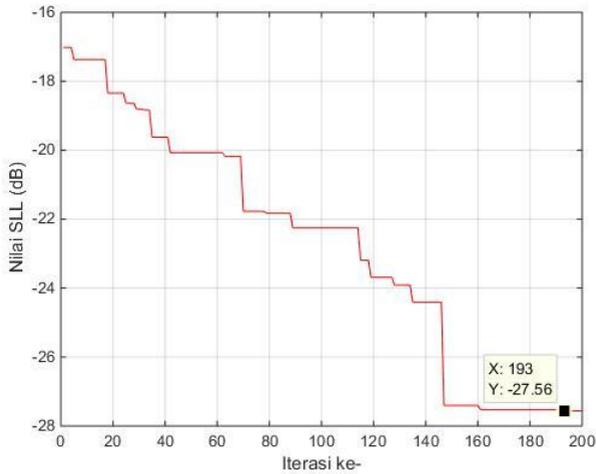
**Gambar 4.19** Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen isotropis



**Gambar 4.20** Konvergensi gen individu pertama yang terpilih pada DE 60 elemen isotropis

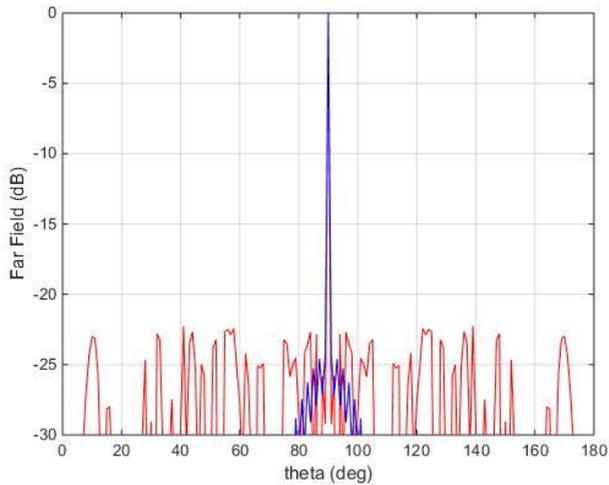


**Gambar 4.21** Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen isotropis

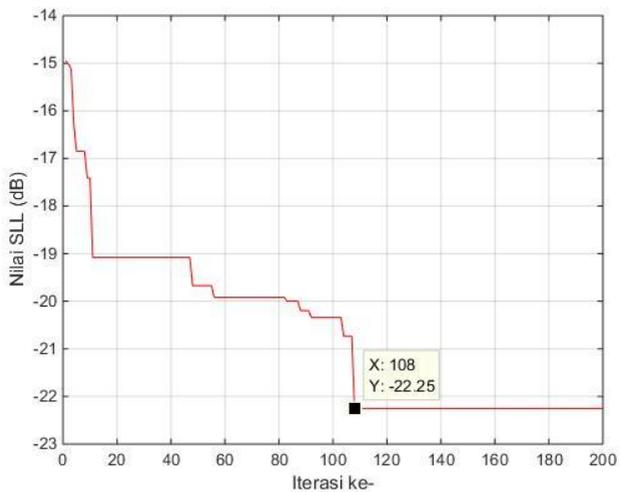


**Gambar 4.22** Konvergensi gen individu kedua yang terpilih pada DE 60 elemen isotropis

Kombinasi gen individu ketiga memiliki nilai *filling* paling kecil adalah 111111111111101 111111111101100 011011111111110 11111001001101, dengan nilai PSL -22,2543 dB (gambar 4.23). Terjadi konvergensi saat iterasi ke-108 (gambar 4.24), memiliki *filling* sebesar 80,00 % dari total 60 elemen yang ada. Pola radiasi gen individu ketiga yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.23. Konvergensi gen individu ketiga yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.24.



**Gambar 4.23** Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen isotropis



**Gambar 4.24** Konvergensi gen individu ketiga yang terpilih pada DE 60 elemen isotropis

Perbandingan hasil simulasi 60 elemen tanpa DE dan dengan DE. Kombinasi gen individu pertama, dapat menurunkan PSLL sebesar -4,2348 dB dengan dua elemen dinonaktifkan. Kombinasi gen individu kedua, dapat menurunkan PSLL sebesar -2,9713 dB dengan lima elemen dinonaktifkan. Kombinasi gen individu ketiga belum dapat menurunkan PSLL, perbedaan PSLL sebesar -2,3357 dB dengan 12 elemen dinonaktifkan.

Dari ketiga kombinasi gen individu diatas didapatkan dua nilai PSLL yang lebih kecil dibandingkan nilai PSLL sebelum diterapkannya metode DE dan satu nilai PSLL yang lebih besar dibandingkan nilai PSLL sebelum diterapkannya metode DE, dengan lebar *beam* tidak berubah signifikan. Pemilihan jumlah elemen yang aktif dapat ditentukan sesuai dengan kebutuhan dari penggunaan antenna *array*. Tabel 4.4 menunjukkan kombinasi gen yang terpilih dari 60 elemen antenna *array*.

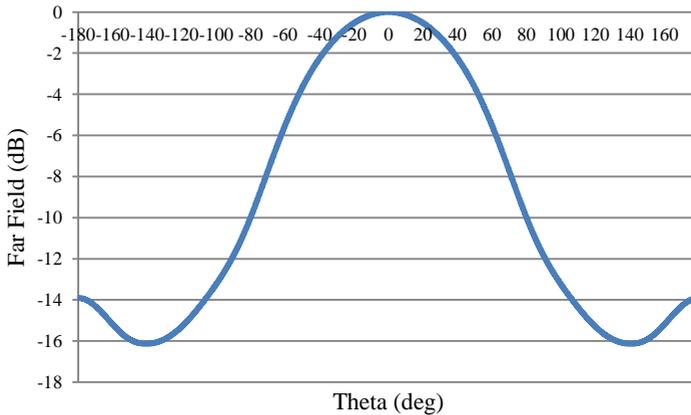
**Tabel 4.10** Kombinasi gen yang terpilih pada DE 60 elemen isotropis

No	Kombinasi Gen	PSLL	Iterasi ke-	Elemen Aktif
1	1111111111111111 1111111111111111 1111111111111111 111111111111001	-28,8248 dB	177	96,67 %
2	1011111111111111 1111111111111111 1111111111111111 111111111000011	-27,5613 dB	193	91,67 %
3	111111111111101 111111111101100 011011111111110 111111001001101	-22,2543 dB	108	80,00 %

#### 4.4 Hasil Simulasi Antena *Array* Linier 15 Elemen Mikrostrip Slot

Pengaplikasian pola elemen mikrostrip slot mengacu pada desain dan bentuk konfigurasi mikrostrip slot pada Lampiran C. Gambar 4.25 merupakan pola elemen mikrostrip slot. Hasil perkalian faktor *array* dan pola elemen mikrostrip slot akan menghasilkan pola far field. Untuk menghasilkan pola Far Field, pola elemen mikrostrip slot dibatasi dari

sudut theta  $-180^0$  hingga  $180^0$  yang bertujuan untuk menyesuaikan dengan pola faktor *array*.

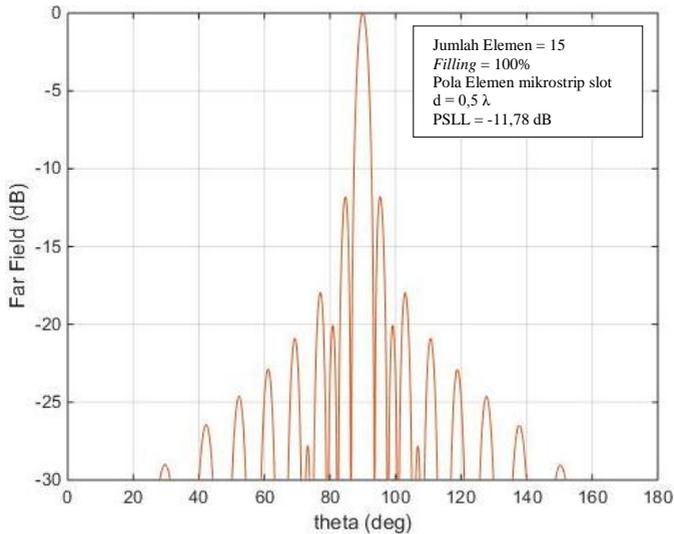


**Gambar 4.25** Pola elemen mikrostrip slot

#### 4.4.1 Simulasi 15 Elemen Mikrostrip Slot Tanpa Metode DE

Dalam simulasi ini, elemen *array* yang menyusun *array* linier merupakan elemen *array* mikrostrip slot. Pertama akan disimulasikan 15 elemen mikrostrip slot tanpa metode *differential evolution* (DE). Simulasi berikutnya adalah penggunaan metode DE untuk mendapatkan kombinasi unik terbaik yang memiliki puncak level *sidelobe* paling kecil. Kemudian membandingkan hasil simulasi tanpa metode DE dan dengan metode DE.

Hasil simulasi 15 elemen isotropis tanpa optimasi, dengan jumlah elemen aktif yang tereksitasi sebanyak 15 elemen atau 100 % dari jumlah keseluruhan dan jarak antar elemen adalah  $0,5 \lambda$ . Nilai puncak level *sidelobe* atau *Peak Sidelobe Level* (PSLL) adalah -11,78 dB pada arah *broadside*.



**Gambar 4.26** Dense array dari 15 elemen mikrostrip slot tanpa metode DE

#### 4.4.2 Simulasi 15 Elemen Mikrostrip Slot dengan Metode DE

Simulasi 15 elemen mikrostrip slot dengan metode DE dilakukan sebanyak 30 kali eksperimen. Dari 30 kali eksperimen tersebut dipilih satu kombinasi gen individu yang memiliki nilai PSLL paling kecil, satu kombinasi gen individu yang memiliki nilai aktif total elemennya (*filling*) paling kecil dan satu kombinasi gen individu yang memiliki sifat diantara keduanya (PSLL dan persentase *filling* yang kecil).

Rata-rata, standar deviasi, variansi, nilai maksimal dan minimal ditunjukkan oleh tabel 4.11 dengan hasil 30 kali eksperimen DE 15 elemen mikrostrip slot pada tabel 4.12. Gambar 4.27 menunjukkan plot elemen aktif (%) terhadap PSLL (dB) dari 30 kali hasil eksperimen DE 15 elemen.

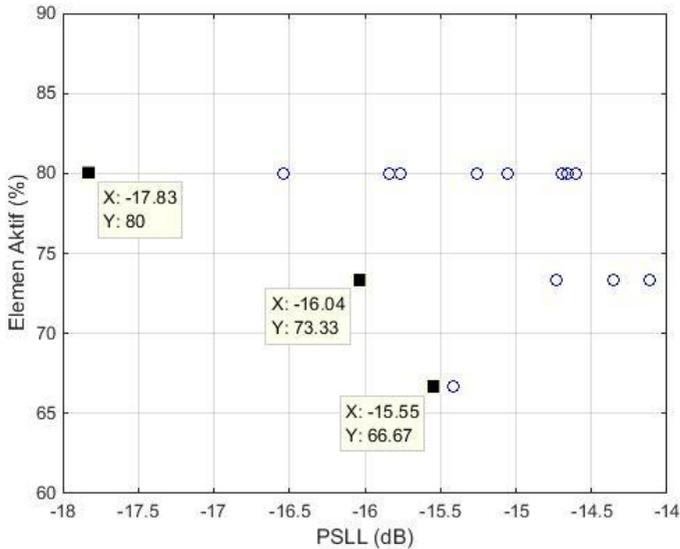
**Tabel 4.11** Persebaran data hasil eksperimen DE 15 elemen mikrostrip slot

Parameter	PSLL (dB)	Iterasi ke-	Elemen aktif
Rata-rata	-15,7471	18,1000	76,4444 %
Standar deviasi	1,0353	23,1298	4,5430 %
Variansi	1,0718	534,9897	20,6386 %

Maksimal	-14,1141	120,0000	80,0000 %
Minimal	-17,8315	1,0000	66,6667 %

**Tabel 4.12** Hasil 30 eksperimen DE 15 elemen mikrostrip slot

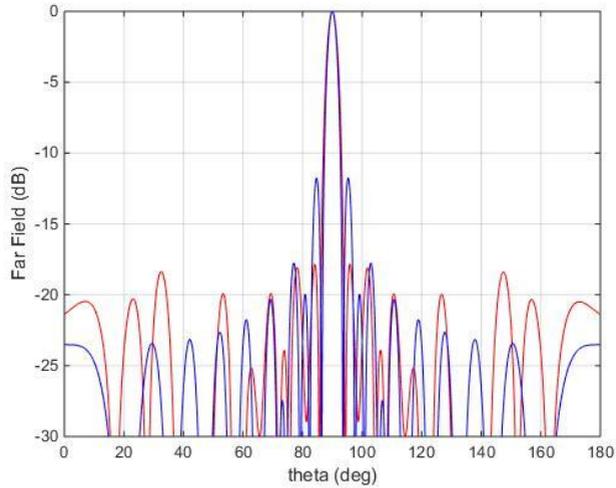
Run ke-	PSLL (dB)	Iterasi ke-	Elemen aktif
1	-16,0400	34	73,33 %
2	-17,8315	4	80,00 %
3	-14,6956	13	80,00 %
4	-17,8315	19	80,00 %
5	-15,7631	9	80,00 %
6	-14,6632	12	80,00 %
7	-16,0357	21	73,33 %
8	-16,0400	26	73,33 %
9	-17,8315	18	80,00 %
10	-15,8389	4	80,00 %
11	-15,5492	6	66,67 %
12	-14,6632	6	80,00 %
13	-16,0357	3	73,33 %
14	-14,7355	32	73,33 %
15	-14,3575	53	73,33 %
16	-15,5492	1	66,67 %
17	-15,7631	7	80,00 %
18	-16,0357	20	73,33 %
19	-15,0521	6	80,00 %
20	-14,6077	3	80,00 %
21	-14,6632	6	80,00 %
22	-14,1141	2	73,33 %
23	-16,5429	21	80,00 %
24	-15,7631	5	80,00 %
25	-16,0357	23	73,33 %
26	-15,8389	5	80,00 %
27	-17,8315	20	80,00 %
28	-16,0357	41	73,33 %
29	-15,2553	120	80,00 %
30	-15,4127	3	66,67 %



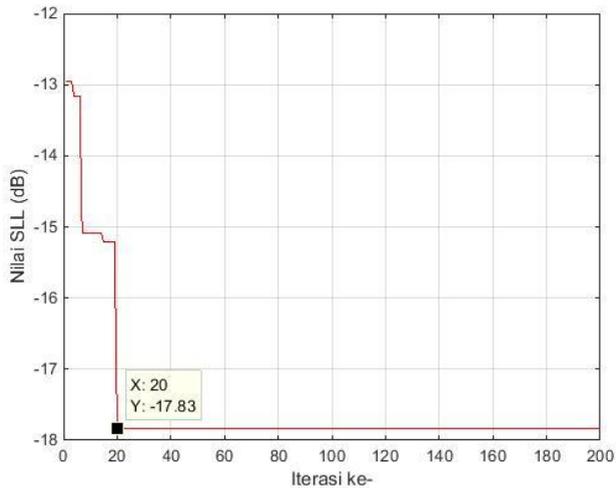
**Gambar 4.27** Plot hasil 30 kali eksperimen DE 15 elemen mikrostrip slot

Kombinasi gen individu pertama memiliki nilai PSLL paling kecil adalah 11111111010101, dengan nilai PSLL -17,8315 dB (gambar 4.28). Terjadi konvergensi saat iterasi ke-20 (gambar 4.29), memiliki *filling* sebesar 80 % dari total 15 elemen yang ada. Pola radiasi gen individu pertama yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.28. Konvergensi gen individu pertama yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.29.

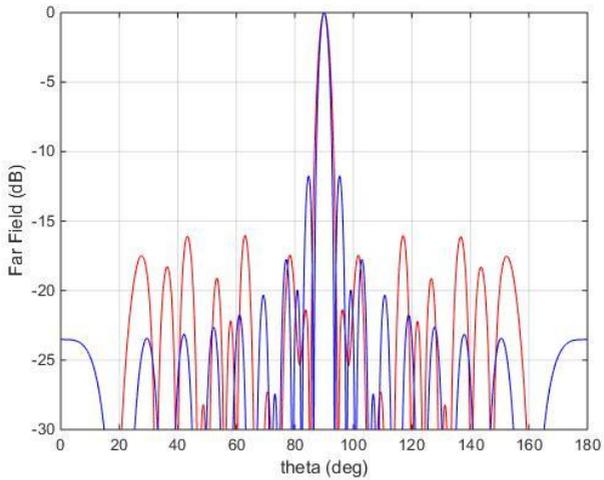
Kombinasi gen individu kedua memiliki nilai PSLL dan *filling* kecil adalah 111111110101001, dengan nilai PSLL -16,0400 dB (gambar 4.30). Terjadi konvergensi saat iterasi ke-34 (gambar 4.31), memiliki *filling* sebesar 73,33 % dari total 15 elemen yang ada. Pola radiasi gen individu kedua yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.30. Konvergensi gen individu kedua yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.31.



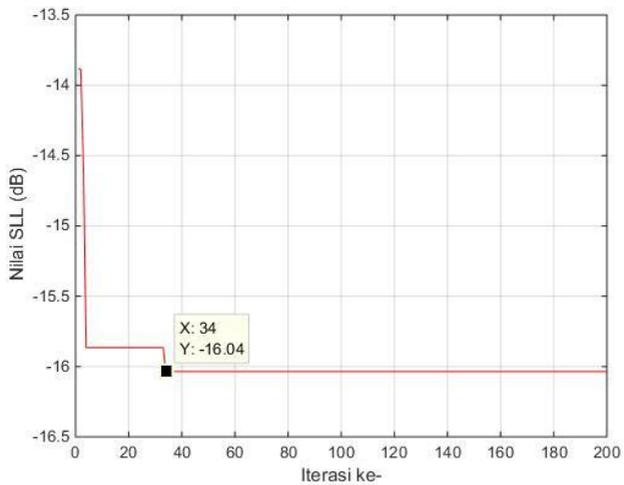
**Gambar 4.28** Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen mikrostrip slot



**Gambar 4.29** Konvergensi gen individu pertama yang terpilih pada DE 15 elemen mikrostrip slot



**Gambar 4.30** Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen mikrostrip slot

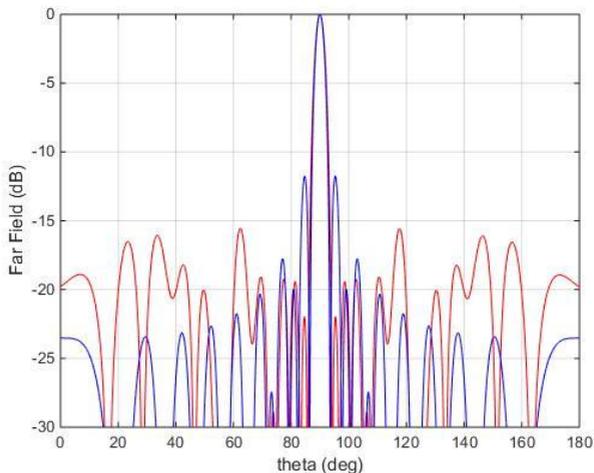


**Gambar 4.31** Konvergensi gen individu kedua yang terpilih pada DE 15 elemen mikrostrip slot

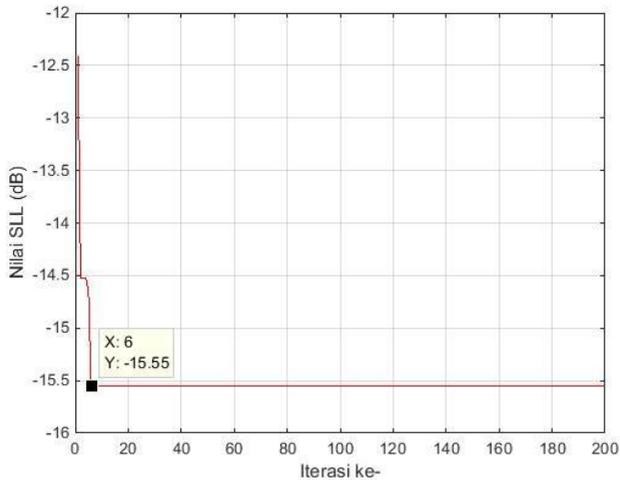
Kombinasi gen individu ketiga memiliki nilai PSLL dan *filling* kecil adalah 111101101010101, dengan nilai PSLL -15,5492 dB (gambar 4.32). Terjadi konvergensi saat iterasi ke-6 (gambar 4.33), memiliki *filling* sebesar 66,67 % dari total 15 elemen yang ada. Pola radiasi gen individu ketiga yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.32. Konvergensi gen individu ketiga yang terpilih pada DE 15 elemen ditunjukkan oleh gambar 4.33.

Perbandingan hasil simulasi 15 elemen tanpa DE dan dengan DE. Kombinasi gen individu pertama, dapat menurunkan PSLL sebesar -6,0515 dB dengan tiga elemen dinonaktifkan. Kombinasi gen individu kedua, dapat menurunkan PSLL sebesar -4,2600 dB dengan empat elemen dinonaktifkan. Kombinasi gen individu ketiga dapat menurunkan PSLL, dengan perbedaan PSLL sebesar -3,7692 dB dengan lima elemen dinonaktifkan.

Dari ketiga kombinasi gen individu diatas sudah dapat menghasilkan nilai PSLL yang lebih kecil dibandingkan nilai PSLL sebelum diterapkannya metode DE, dengan lebar *beam* tidak berubah signifikan. Pemilihan jumlah elemen yang aktif dapat ditentukan sesuai dengan kebutuhan dari penggunaan antenna *array*. Tabel 4.3 menunjukkan kombinasi gen yang terpilih dari 30 elemen antenna *array*.



**Gambar 4.32** Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 15 elemen mikrostrip slot



**Gambar 4.33** Konvergensi gen individu ketiga yang terpilih pada DE 15 elemen mikrostrip slot

**Tabel 4.13** Kombinasi gen yang terpilih pada DE 15 elemen mikrostrip slot

No	Kombinasi Gen	PSLL (dB)	Iterasi ke-	Elemen Aktif
1	11111111010101	-17,8315	20	80,00 %
2	111111110101001	-16,0400	34	73,33 %
3	111101101010101	-15,5492	6	66,67 %

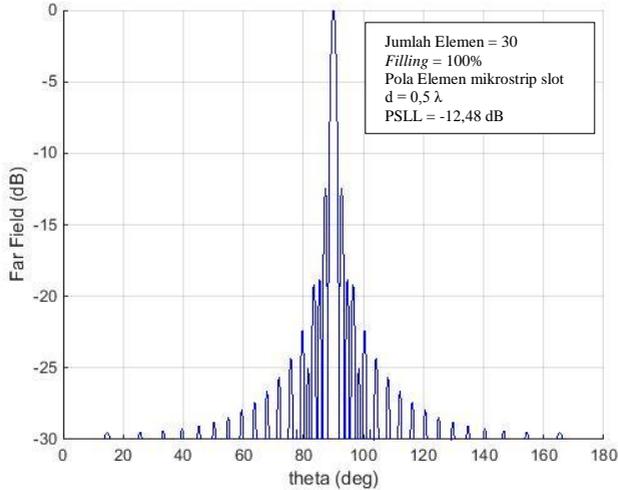
#### 4.5 Hasil Simulasi Antena Array Linier 30 Elemen Mikrostrip Slot

Dalam simulasi ini, elemen *array* yang menyusun *array* linier merupakan elemen *array* mikrostrip slot. Pertama akan disimulasikan 30 elemen mikrostrip slot tanpa metode *differential evolution* (DE). Simulasi berikutnya adalah penggunaan metode DE untuk mendapatkan kombinasi unik terbaik yang memiliki puncak level *sidelobe* paling kecil. Kemudian membandingkan hasil simulasi tanpa metode DE dan dengan metode DE.

##### 4.5.1 Simulasi 30 Elemen Mikrostrip Slot Tanpa Metode DE

Hasil simulasi 30 elemen mikrostrip slot tanpa optimasi, dengan jumlah elemen aktif yang tereksitasi sebanyak 30 elemen atau 100 %

dari jumlah keseluruhan dan jarak antar elemen adalah  $0,5 \lambda$ . Nilai puncak level *sidelobe* atau *Peak Sidelobe Level (PSLL)* adalah  $-12,48 \text{ dB}$  pada arah *broadside*.



**Gambar 4.34** *Dense array* dari 30 elemen mikrostrip slot tanpa metode DE

#### 4.5.2 Simulasi 30 Elemen Mikrostrip Slot dengan Metode DE

Simulasi 30 elemen mikrostrip slot dengan metode DE dilakukan sebanyak 30 kali eksperimen. Dari 30 kali eksperimen tersebut dipilih satu kombinasi gen individu yang memiliki nilai PSLL paling kecil, satu kombinasi gen individu yang memiliki nilai aktif total elemennya (*filling*) paling kecil dan satu kombinasi gen individu yang memiliki sifat diantara keduanya (PSLL dan persentase *filling* yang kecil).

Rata-rata, standar deviasi, variansi, nilai maksimal dan minimal ditunjukkan oleh tabel 4.14 dengan hasil 30 kali eksperimen DE 30 elemen mikrostrip slot pada tabel 4.15. Gambar 4.35 menunjukkan plot elemen aktif (%) terhadap PSLL (dB) dari 30 kali hasil eksperimen DE 60 elemen.

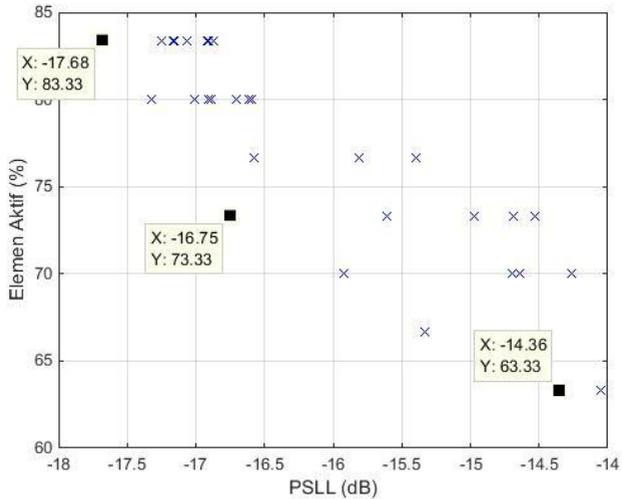
**Tabel 4.14** Persebaran data hasil eksperimen DE 30 elemen mikrostrip slot

Parameter	PSLL (dB)	Iterasi ke-	Elemen aktif
Rata-rata	-16,0882	57,8667	76,5556 %

Standar deviasi	1,1125	43,8168	6,2197 %
Variansi	1,2377	1919,9126	38,6845 %
Maksimal	-14,0533	166,0000	83,3333 %
Minimal	-17,6802	4,0000	63,3333 %

**Tabel 4.15** Hasil 30 eksperimen DE 30 elemen mikrostrip slot

<b>Run ke-</b>	<b>PSLL (dB)</b>	<b>Iterasi ke-</b>	<b>Elemen aktif</b>
1	-16,0400	34	73,33 %
2	-17,8315	4	80,00 %
3	-14,6956	13	80,00 %
4	-17,8315	19	80,00 %
5	-15,7631	9	80,00 %
6	-14,6632	12	80,00 %
7	-16,0357	21	73,33 %
8	-16,0400	26	73,33 %
9	-17,8315	18	80,00 %
10	-15,8389	4	80,00 %
11	-15,5492	6	66,67 %
12	-14,6632	6	80,00 %
13	-16,0357	3	73,33 %
14	-14,7355	32	73,33 %
15	-14,3575	53	73,33 %
16	-15,5492	1	66,67 %
17	-15,7631	7	80,00 %
18	-16,0357	20	73,33 %
19	-15,0521	6	80,00 %
20	-14,6077	3	80,00 %
21	-14,6632	6	80,00 %
22	-14,1141	2	73,33 %
23	-16,5429	21	80,00 %
24	-15,7631	5	80,00 %
25	-16,0357	23	73,33 %
26	-15,8389	5	80,00 %
27	-17,8315	20	80,00 %
28	-16,0357	41	73,33 %
29	-15,2553	120	80,00 %
30	-15,4127	3	66,67 %



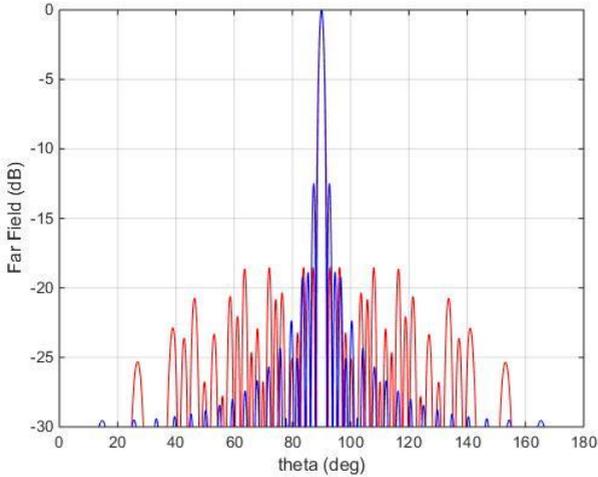
**Gambar 4.35** Plot hasil 30 kali eksperimen DE 30 elemen mikrostrip slot

Kombinasi gen individu pertama memiliki nilai PSLL paling kecil adalah 111111111111111 110101101100111, dengan nilai PSLL -17,6802 dB (gambar 4.36). Terjadi konvergensi saat iterasi ke-60 (gambar 4.37), memiliki *filling* sebesar 83,33 % dari total 30 elemen yang ada. Pola radiasi gen individu pertama yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.36. Konvergensi gen individu pertama yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.37.

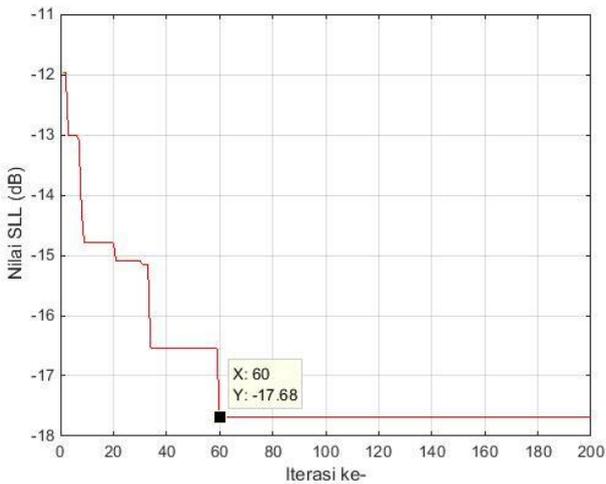
Kombinasi gen individu kedua memiliki nilai PSLL dan *filling* kecil adalah 111111111111101 010111001101001, dengan nilai PSLL -16,7506 dB (gambar 4.38). Terjadi konvergensi saat iterasi ke-42 (gambar 4.39), memiliki *filling* sebesar 73,33 % dari total 30 elemen yang ada. Pola radiasi gen individu kedua yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.38. Konvergensi gen individu kedua yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.39.

Kombinasi gen individu ketiga memiliki nilai PSLL dan *filling* kecil adalah 111100111101010 110011111010001, dengan nilai PSLL -14,3582 dB (gambar 4.40). Terjadi konvergensi saat iterasi ke-13 (gambar 4.41), memiliki *filling* sebesar 63,33 % dari total 30 elemen yang ada. Pola radiasi gen individu kedua yang terpilih pada DE 30

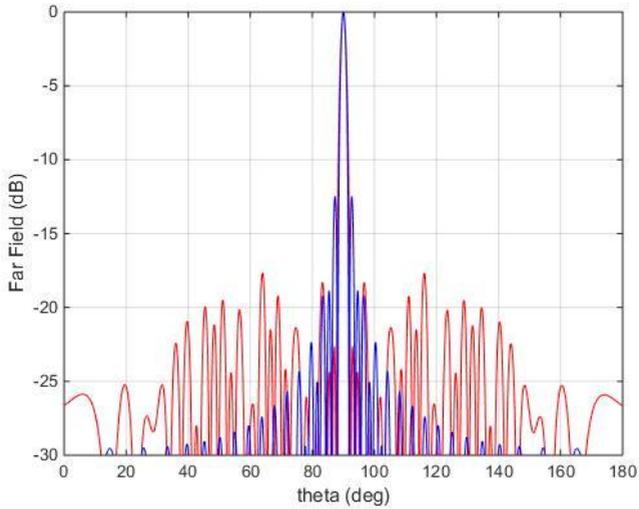
elemen ditunjukkan oleh gambar 4.40. Konvergensi gen individu ketiga yang terpilih pada DE 30 elemen ditunjukkan oleh gambar 4.41.



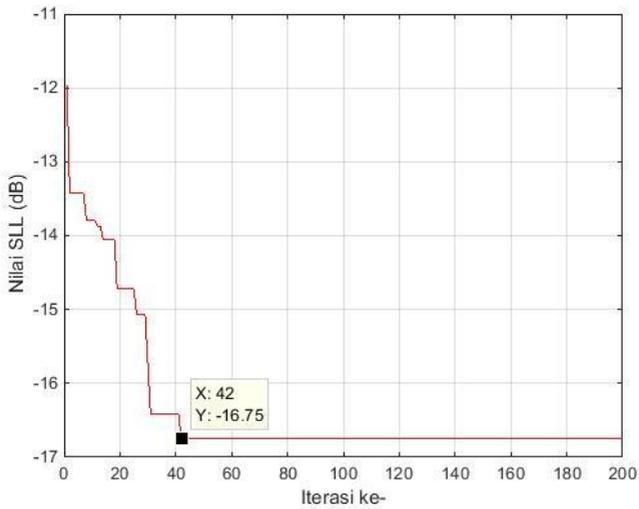
**Gambar 4.36** Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen mikrostrip slot



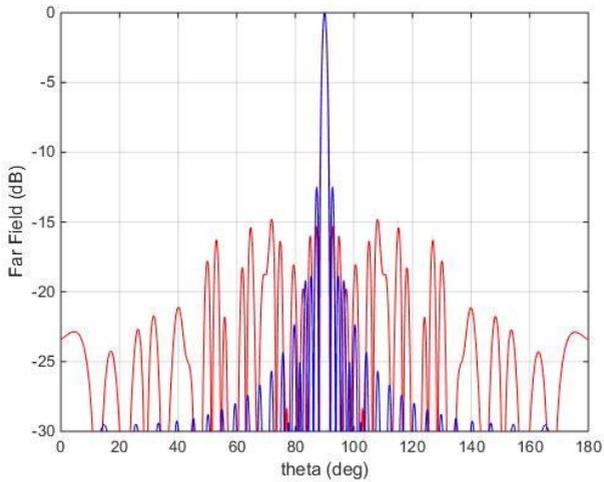
**Gambar 4.37** Konvergensi gen individu pertama yang terpilih pada DE 30 elemen mikrostrip slot



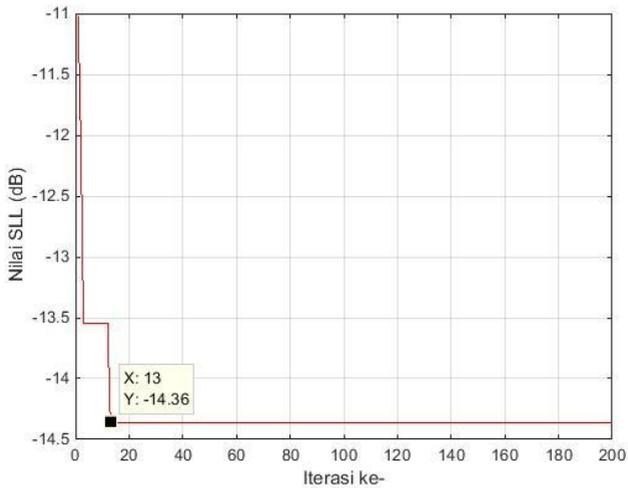
**Gambar 4.38** Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen mikrostrip slot



**Gambar 4.39** Konvergensi gen individu kedua yang terpilih pada DE 30 elemen mikrostrip slot



**Gambar 4.40** Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 30 elemen mikrostrip slot



**Gambar 4.41** Konvergensi gen individu ketiga yang terpilih pada DE 30 elemen mikrostrip slot

Perbandingan hasil simulasi 30 elemen tanpa DE dan dengan DE. Kombinasi gen individu pertama, dapat menurunkan PSLL sebesar -5,2002 dB dengan lima elemen dinonaktifkan. Kombinasi gen individu kedua, dapat menurunkan PSLL sebesar -4,2706 dB dengan delapan elemen dinonaktifkan. Kombinasi gen individu ketiga dapat menurunkan PSLL, dengan perbedaan PSLL sebesar -1,8782 dB dengan 11 elemen dinonaktifkan.

Dari ketiga kombinasi gen individu diatas sudah dapat menghasilkan nilai PSLL yang lebih kecil dibandingkan nilai PSLL sebelum diterapkannya metode DE, dengan lebar *beam* tidak berubah signifikan. Pemilihan jumlah elemen yang aktif dapat ditentukan sesuai dengan kebutuhan dari penggunaan antena *array*. Tabel 4.16 menunjukkan kombinasi gen yang terpilih dari 30 elemen antena *array*.

**Tabel 4.16** Kombinasi gen yang terpilih pada DE 30 elemen mikrostrip slot

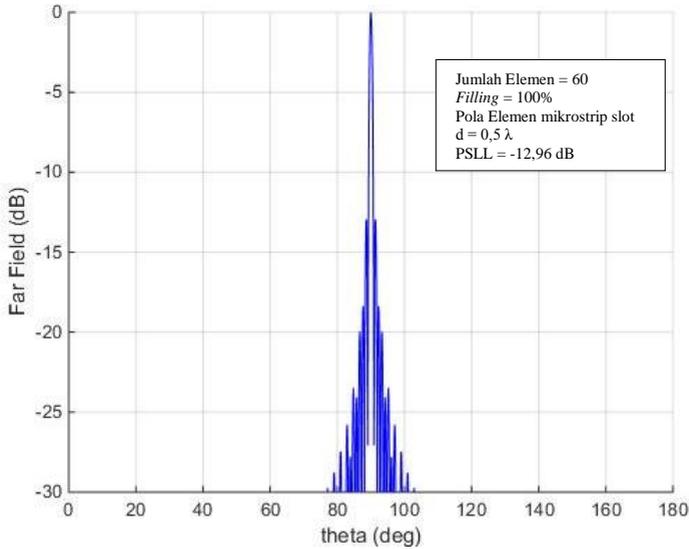
No	Kombinasi Gen	PSLL (dB)	Iterasi ke-	Elemen Aktif
1	1111111111111111 110101101100111	-17,6802	60	83,33%
2	111111111111101 010111001101001	-16,7506	42	73,33%
3	111100111101010 110011111010001	-14,3582	13	63,33%

#### 4.6 Hasil Simulasi Antena Array Linier 60 Elemen Mikrostrip Slot

Dalam simulasi ini, elemen *array* yang menyusun *array* linier merupakan elemen *array* mikrostrip slot. Pertama akan disimulasikan 60 elemen mikrostrip slot tanpa metode *differential evolution* (DE). Simulasi berikutnya adalah penggunaan metode DE untuk mendapatkan kombinasi unik terbaik yang memiliki pucak level *sidelobe* paling kecil. Kemudian membandingkan hasil simulasi tanpa metode DE dan dengan metode DE.

##### 4.6.1 Simulasi 60 Elemen Mikrostrip Slot Tanpa Metode DE

Hasil simulasi 60 elemen mikrostrip slot tanpa optimasi, dengan jumlah elemen aktif yang tereksitasi sebanyak 60 elemen atau 100 % dari jumlah keseluruhan dan jarak antar elemen adalah  $0,5 \lambda$ . Nilai pucak level *sidelobe* atau *Peak Sidelobe Level* (PSLL) adalah -12,48 dB pada arah *broadside*.



**Gambar 4.42** Dense array dari 60 elemen mikrostrip slot tanpa metode DE

#### 4.6.2 Simulasi 60 Elemen Mikrostrip Slot dengan Metode DE

Simulasi 60 elemen mikrostrip slot dengan metode DE dilakukan sebanyak 30 kali eksperimen. Dari 30 kali eksperimen tersebut dipilih satu kombinasi gen individu yang memiliki nilai PSLL paling kecil, satu kombinasi gen individu yang memiliki nilai aktif total elemennya (*filling*) paling kecil dan satu kombinasi gen individu yang memiliki sifat diantara keduanya (PSLL dan persentase *filling* yang kecil).

Rata-rata, standar deviasi, variansi, nilai maksimal dan minimal ditunjukkan oleh tabel 4.17 dengan hasil 30 kali eksperimen DE 60 elemen mikrostrip slot pada tabel 4.18. Gambar 4.18 menunjukkan plot elemen aktif (%) terhadap PSLL (dB) dari 30 kali hasil eksperimen DE 60 elemen.

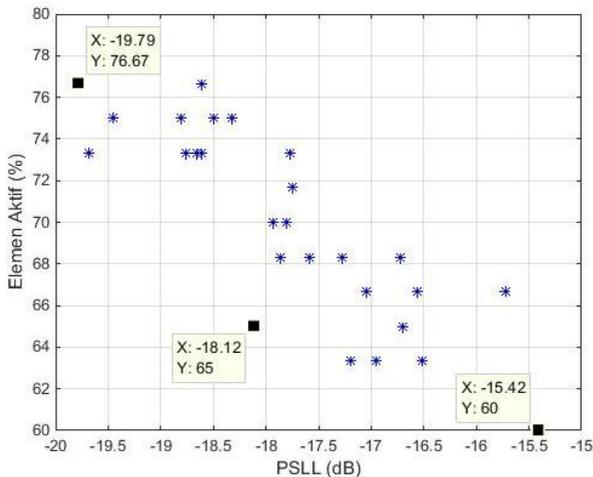
**Tabel 4.17** Persebaran data hasil eksperimen DE 60 elemen mikrostrip slot

Parameter	PSLL (dB)	Iterasi ke-	Elemen aktif
Rata-rata	-17,8773	84,1333	70,4444 %
Standar deviasi	1,1233	48,2992	4,9506 %
Variansi	1,2618	2332,8092	24,5083 %

Maksimal	-15,4207	195,0000	78,3333 %
Minimal	-19,7864	12,0000	60,0000 %

**Tabel 4.18** Hasil 30 eksperimen DE 60 elemen mikrostrip slot

Run ke-	PSLL (dB)	Iterasi ke-	Elemen aktif
1	-18,6582	58	73,33 %
2	-17,8595	74	68,33 %
3	-17,2761	13	68,33 %
4	-17,7760	21	73,33 %
5	-15,7176	42	66,67 %
6	-16,5608	42	66,67 %
7	-17,0501	144	66,67 %
8	-16,6941	111	65,00 %
9	-18,6112	129	76,67 %
10	-18,4960	195	75,00 %
11	-18,3202	100	75,00 %
12	-18,8046	154	75,00 %
13	-16,7211	81	68,33 %
14	-19,1435	121	78,33 %
15	-17,5831	58	68,33 %
16	-19,4504	181	75,00 %
17	-19,3207	27	78,33 %
18	-18,7584	81	73,33 %
19	-17,1903	60	63,33 %
20	-15,4207	23	60,00 %
21	-16,9498	37	63,33 %
22	-18,6127	122	73,33 %
23	-19,7864	68	76,67 %
24	-19,6866	95	73,33 %
25	-17,9364	74	70,00 %
26	-17,7422	85	71,67 %
27	-16,5127	12	63,33 %
28	-17,7527	111	71,67 %
29	-17,8062	83	70,00 %
30	-18,1197	122	65,00 %



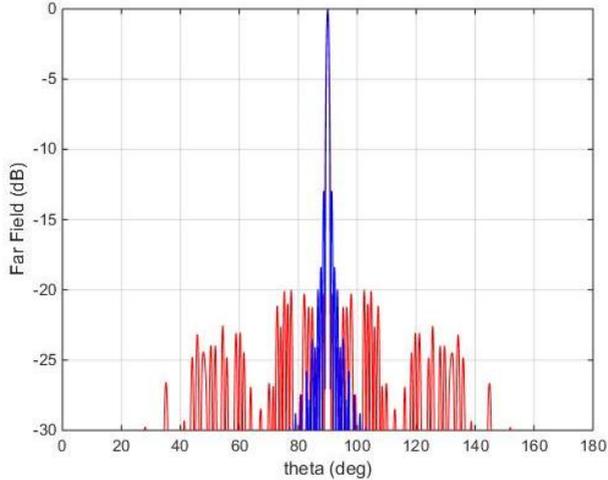
**Gambar 4.43** Plot hasil 30 kali eksperimen DE 60 elemen mikrostrip slot

Kombinasi gen individu pertama memiliki nilai PSLL paling kecil adalah 11111111111111 111111111110110 111111110100110 000111110001001, dengan nilai PSLL -19,7864 dB (gambar 4.44). Terjadi konvergensi saat iterasi ke-68 (gambar 4.45), memiliki *filling* sebesar 76,67 % dari total 60 elemen yang ada. Pola radiasi gen individu pertama yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.44. Konvergensi gen individu pertama yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.45.

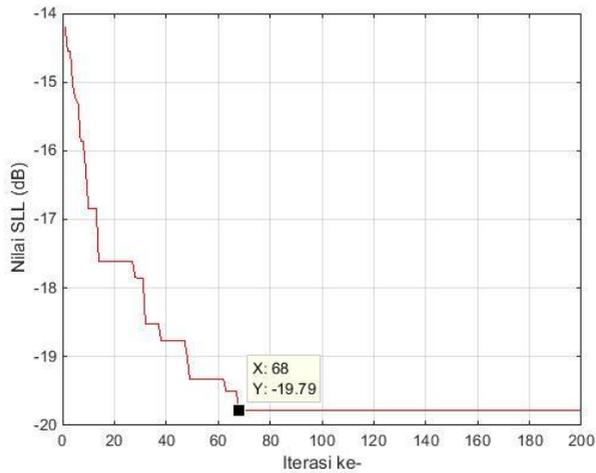
Kombinasi gen individu kedua memiliki nilai PSLL dan *filling* kecil adalah 111111101111110 110111111001011 101111100011000 100101000101101, dengan nilai PSLL -18,1197 dB (gambar 4.38). Terjadi konvergensi saat iterasi ke-122 (gambar 4.39), memiliki *filling* sebesar 65 % dari total 60 elemen yang ada. Pola radiasi gen individu kedua yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.38. Konvergensi gen individu kedua yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.39.

Kombinasi gen individu ketiga memiliki nilai PSLL dan *filling* kecil adalah 101101011101010 101111111110111 101001000111100 110100001010011, dengan nilai PSLL -15,4207 dB (gambar 4.40). Terjadi konvergensi saat iterasi ke-23 (gambar 4.41), memiliki *filling* sebesar 60 % dari total 60 elemen yang ada. Pola radiasi gen individu

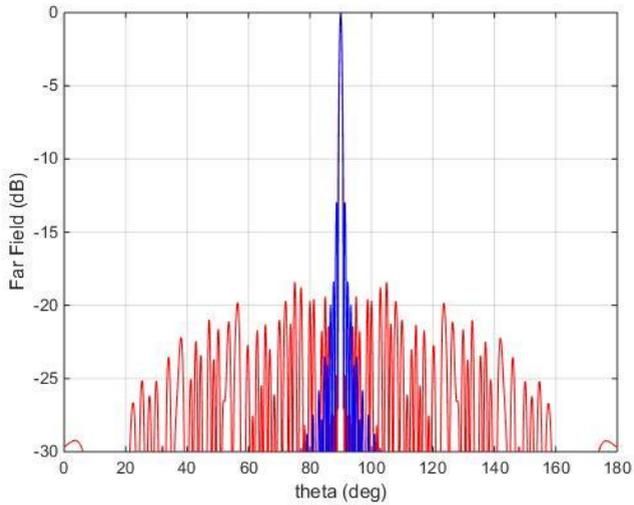
kedua yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.40. Konvergensi gen individu ketiga yang terpilih pada DE 60 elemen ditunjukkan oleh gambar 4.41.



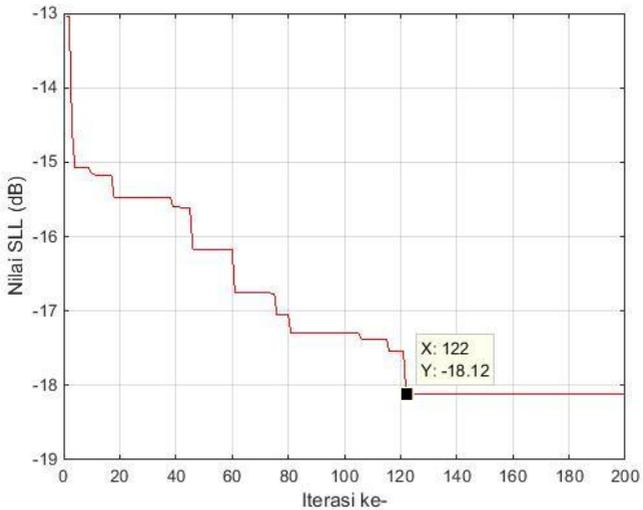
**Gambar 4.44** Pola radiasi gen individu pertama yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen mikrostrip slot



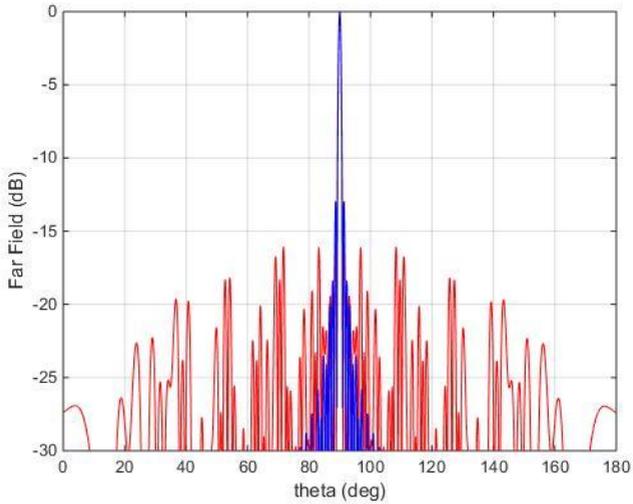
**Gambar 4.45** Konvergensi gen individu pertama yang terpilih pada DE 60 elemen mikrostrip slot



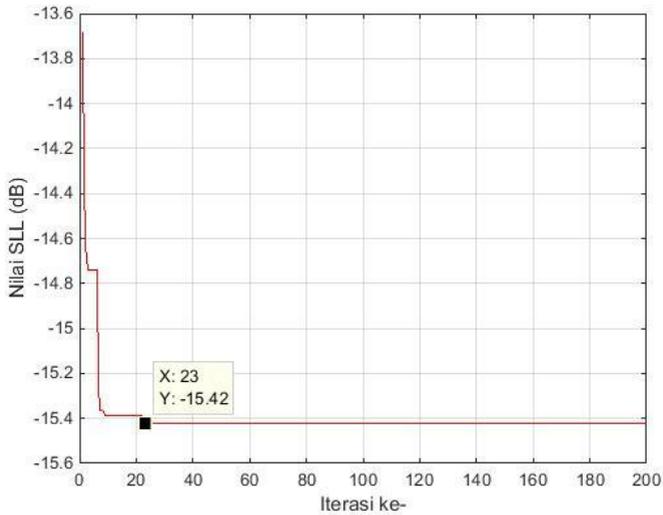
**Gambar 4.46** Pola radiasi gen individu kedua yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen mikrostrip slot



**Gambar 4.47** Konvergensi gen individu kedua yang terpilih pada DE 60 elemen mikrostrip slot



**Gambar 4.48** Pola radiasi gen individu ketiga yang terpilih pada DE (merah) dan tanpa DE (biru) untuk 60 elemen mikrostrip slot



**Gambar 4.49** Konvergensi gen individu ketiga yang terpilih pada DE 60 elemen mikrostrip slot

Perbandingan hasil simulasi 30 elemen tanpa DE dan dengan DE. Kombinasi gen individu pertama, dapat menurunkan PSLL sebesar -5,2002 dB dengan lima elemen dinonaktifkan. Kombinasi gen individu kedua, dapat menurunkan PSLL sebesar -4,2706 dB dengan delapan elemen dinonaktifkan. Kombinasi gen individu ketiga dapat menurunkan PSLL, dengan perbedaan PSLL sebesar -1,8782 dB dengan 11 elemen dinonaktifkan.

Dari ketiga kombinasi gen individu diatas sudah dapat menghasilkan nilai PSLL yang lebih kecil dibandingkan nilai PSLL sebelum diterapkannya metode DE, dengan lebar *beam* tidak berubah signifikan. Pemilihan jumlah elemen yang aktif dapat ditentukan sesuai dengan kebutuhan dari penggunaan antena *array*. Tabel 4.19 menunjukkan kombinasi gen yang terpilih dari 60 elemen antena *array*.

**Tabel 4.19** Kombinasi gen yang terpilih pada DE 60 elemen mikrostrip slot

No	Kombinasi Gen	PSLL (dB)	Iterasi ke-	Elemen Aktif
1	1111111111111111 111111111110110 111111110100110 000111110001001	-19,7864	68	76,67 %
2	1111111011111110 110111111001011 101111100011000 100101000101101	-18,1197	122	65 %
3	101101011101010 101111111110111 101001000111100 110100001010011	-15,4207	23	60 %

#### 4.7 Sintesis

Untuk meningkatkan gain dan direktivitas (keterarahan) sistem komunikasi maka diperlukan antena yang memiliki *array* linier dengan jumlah banyak. Pada sistem komunikasi jarak jauh, seperti sistem satelit dan sistem RADAR, antena yang memiliki *beamwidth* kecil dan gain yang besar sangat diperlukan untuk mendapatkan informasi semaksimal mungkin tanpa terganggu oleh *sidelobe* level yang juga merupakan sinyal informasi yang berpotensi menginterferensi sinyal informasi utama. Untuk mendesain antena yang memiliki pola radiasi yang diinginkan, dapat dilakukan beberapa teknik optimasi pada parameter

antena diantaranya optimalisasi konfigurasi arus eksitasi masing-masing elemen antenna. Salah satu teknik optimasi antenna yang digunakan adalah *differential evolution* (DE).

Berdasarkan hasil simulasi yang diperoleh dapat diamati bahwa pola radiasi yang terbentuk pada suatu antenna *array* linier menggunakan *differential evolution* pada antenna linier 15 elemen, antenna linier 30 elemen, maupun antenna linier 60 elemen bentuk dari pola radiasinya sudah sesuai dengan yang diharapkan bahwa arah *sidelobe* menjauhi sudut  $\theta = \pi/2$  arah *broadside* dan nilai *sidelobe* levelnya menjadi kecil. Hasil yang diperoleh berbeda jika dibandingkan dengan pola radiasi *array* linier tanpa optimasi metode *differential evolution*. Hal ini menunjukkan bahwa metode *differential evolution* yang digunakan telah berhasil mengoptimasi arus eksitasi masing-masing elemen *array* antenna linier.

Dari hasil simulasi didapatkan bahwa pada simulasi metode DE 15 elemen dan 30 elemen dapat dengan cepat memperoleh kombinasi arus eksitasi terbaik hasil pola radiasinya memiliki nilai puncak *sidelobe* level (PSLL) lebih kecil dari *dense array* masing-masing elemen. Pada simulasi metode DE 60 elemen memperoleh kombinasi arus eksitasi terbaik lebih lambat, namun hasil pola radiasinya memiliki nilai puncak *sidelobe* level (PSLL) tetap lebih kecil dari *dense array* awal.

Dari gambar plot nilai *fitness* PSLL terlihat bahwa untuk antenna dengan *array* 15 elemen dan *array* 30 elemen nilai PSLL akan stabil dan tidak berubah sampai batas iterasi tercapai kurang dari iterasi ke-100. Hal ini menunjukkan bahwa setelah iterasi ke-100 solusi yang ditawarkan tidak lebih baik dari solusi yang sudah ditemukan pada iterasi sebelumnya, dan tidak memberikan perbaikan pada pola radiasi yang telah ditemukan. Sehingga diperlukan penurunan batasan kriteria terminasi lebih kecil pada antenna dengan *array* 15 elemen dan *array* 30 elemen.

Dari gambar plot nilai *fitness* PSLL terlihat bahwa untuk antenna dengan *array* 60 elemen nilai PSLL masih belum konvergen dan dapat berubah sampai batas iterasi hampir tercapai. Hal ini menunjukkan bahwa antenna dengan *array* 60 elemen memiliki waktu konvergensi yang lebih lama dibandingkan *array* 15 elemen dan 30 elemen. Diperlukan peningkatan batasan kriteria terminasi pada antenna dengan *array* 60 elemen, sehingga kemungkinan mencapai nilai konvergensi PSLL yang lebih kecil menjadi lebih besar.

Hasil dari 30 kali eksperimen yang diplot pada nilai PSSL terhadap persentase elemen aktif, seperti yang ditunjukkan oleh gambar 4.2, 4.10, 4.18, 4.27, 4.35, dan 4.43, disebut dengan grafik pareto. Nilai-nilai yang tersebar pada grafik pareto akan membentuk sebuah garis diagonal imajiner yang membatasi daerah optimal untuk masalah optimasi tertentu. Dengan menggunakan dua sumbu sebagai objek atau tujuan optimasi, maka akan didapatkan satu nilai optimal pada tiap sumbu dan satu nilai optimal yang berlaku pada kedua sumbu. Jumlah titik/nilai optimal akan selalu bertambah jika objek tujuan optimasi semakin banyak.

## **BAB 5**

### **PENUTUP**

Pada bab ini akan diuraikan beberapa kesimpulan dari hasil pembahasan pada bab sebelumnya dan saran mengenai masalah yang bisa dibahas sebagai kelanjutan dari tugas akhir ini.

#### **5.1 Kesimpulan**

Berdasarkan hasil simulasi dan setelah dilakukan proses analisa, maka diperoleh kesimpulan sebagai berikut:

1. Metode *differential evolution* terbukti dapat menghasilkan kombinasi solusi untuk mengurangi *sidelobe* level pada antena *array* linier.
2. Dengan mengoptimasi arus eksitasi pada masing-masing elemen antena *array* linier, maka akan diperoleh kombinasi arus eksitasi terbaik yang akan memiliki nilai *sidelobe* level paling kecil.
3. Dari hasil simulasi terlihat bahwa semakin banyak jumlah elemen *array* yang akan dioptimasi, maka akan semakin sedikit elemen yang tidak mengalami eksitasi.
4. Dari hasil simulasi terlihat bahwa semakin banyak jumlah elemen *array* yang akan dioptimasi, maka waktu konvergensi iterasi akan semakin lama.
5. Dari hasil simulasi terlihat bahwa semakin banyak jumlah elemen *array* yang akan dioptimasi, maka perbedaan *sidelobe* level yang diminimalisir menjadi semakin kecil.
6. Hubungan antara puncak *sidelobe* level (PSLL) dengan persentase elemen yang aktif adalah berbanding terbalik, atau memiliki sifat *trade off* satu sama lain. Sehingga pemilihan prioritas dari kebutuhan khusus antena *array* linier perlu dipertimbangkan.
7. Konvergensi *differential evolution* pada penelitian ini akan semakin lambat jika populasi, dan jumlah generasi atau jumlah iterasi diperbanyak.
8. Metode *thinning* dapat bekerja dengan baik pada penerapan metode *differential evolution*, dan metode *differential evolution* mampu menghasilkan *thinned array* dengan performa PSLL yang lebih baik.
9. Rekomendasi parameter optimasi DE untuk mendapatkan hasil optimasi *thinned array* terbaik untuk *dense array* dengan 15-60 elemen adalah jumlah populasi sebanyak 20, jumlah gen yang mewakili satu elemen adalah 1, probabilitas mutasi sebesar 0,5 dan

probabilitas crossover sebesar 0,9 serta maksimum iterasi sejumlah 200.

10. Grafik pareto dapat digunakan untuk optimasi dengan dua objek atau tujuan optimasi dengan membentuk daerah optimal pada masalah optimasi tertentu.

## 5.2 Saran

Ada beberapa hal yang dapat disarankan untuk kelanjutan penelitian ini, antara lain:

1. Penelitian ini dapat dikembangkan dengan menerapkan metode dari algoritma atau program selain metode *differential evolution* untuk mengetahui perbandingan hasil dari performa, pola radiasi serta waktu konvergensi iterasi.
2. Pada tugas akhir ini simulasi antena *array* linier untuk mengurangi *sidelobe* level menggunakan *differential evolution* dengan melakukan optimasi pada kombinasi arus eksitasi masing-masing elemen antena, maka pada penelitian mendatang dapat dioptimasi menggunakan pendekatan parameter lainnya.

## DAFTAR PUSTAKA

- [1] C. A. Balanis, *Antenna Theory: Analysis and Design*, 3rd ed., John Wiley & Sons, 2014.
- [2] R. Storn, K. V. Price, and J. Lampinen, *Differential Evolution—A Practical Approach to Global Optimization*. Berlin, Germany: Springer-Verlag, 2005.
- [3] S. Hota, B. B. Mangaraj, "Performance of a Dipole Linear *Array* Antenna Optimized using Differential Evolution Algorithm", IEEE Conference Publications, 2015.
- [4] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, "Differential Evolution Using A Neighborhood-Based Mutation Operator," IEEE Transactions on Evolutionary Computation, vol. 13, no. 3, pp. 526–553, Jun. 2009.
- [5] Y. Chen, S. Yang, and Z. Nie, "Synthesis of uniform amplitude thinned linear phased *arrays* using the differential evolution algorithm," Electromagnetics, vol. 27, pp. 287–297, 2007.
- [6] C. Rocha-Alicanoa, D. Covarrubias-Rosalesa, C. Brizuela-Rodrigueza, and M. Panduro-Mendoza, "Differential evolution algorithm applied to sidelobe level reduction on a planar *array*," Int. J. Electron. Commun., vol. 61, pp. 286–290, 2007.

*[Halaman ini sengaja dikosongkan]*

# LAMPIRAN A

Departemen Teknik Elektro  
Fakultas Teknologi Elektro – ITS

## TE141599 TUGAS AKHIR – 4 SKS

Nama Mahasiswa : Radifan Aiman Nabil Hibatulhaqqi  
Nomer Pokok : 2213 100 103  
Bidang Studi : Teknik Telekomunikasi Multimedia  
Tugas Diberikan : Semester Genap Th. 2016/2017  
Dosen Pembimbing : 1. Eko Setijadi , ST., MT., Ph.D

13 FEB 2017

Judul Tugas Akhir : **Implementasi metode *Differential Evolution* pada antena *array* untuk mengurangi *side lobe* (*Differential Evolution implementation in antenna array for sidelobe suppressing*)**

### Uraian Tugas Akhir :

Antena *array* memiliki nilai *gain* yang tinggi dan memiliki pola radiasi terarah pada penerimaan sinyal datang di sudut tertentu, hal ini yang mendasari penggunaan antena *array* pada sistem RADAR. Akan tetapi setiap penambahan *array* pada antena maka semakin banyak pola radiasi *sidelobe* yang terjadi. *Main lobe* membawa informasi utama tentang jarak, kecepatan, dan sudut, akan tetapi *sidelobe* adalah radiasi *lobe* yang dituar arah sudut yang diinginkan pada pola radiasi terarah. *Sidelobe* juga membawa informasi utama yang sama dengan *main lobe*. Sehingga pola radiasi *main lobe* dapat terganggu oleh interferensi dari pola radiasi *sidelobe*. Maka dari itu, dibutuhkan metode untuk meminimalisir *sidelobe* yang terbentuk oleh *array* antena sehingga informasi pada *main lobe* dapat diperoleh secara maksimal. Antena *array* memiliki karakteristik gain besar, arah beam yang terarah, kemudahan produksi sehingga cocok untuk penggunaan pada sistem komunikasi satelit, mobile, dan radar yang membutuhkan antena gain tinggi yang terarah. *Array* antena dapat menjadi lebih efisien dengan mengurangi *sidelobe* yang terjadi, dalam penelitian ini digunakan metode *Differential Evolution* (DE) untuk mencari jarak optimum antar *array* agar *sidelobe* pada *array* antena berkurang.

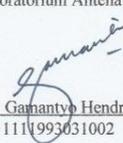
Dosen Pembimbing,



Eko Setijadi, ST., MT., Ph.D.  
Nip : 197210012003121002



Menyetujui,  
Kepala Laboratorium Antena dan Propagasi

  
Prof. Dr. Ir. Gamantyo Hendrantoro, Ph.D.  
Nip : 197011111993031002

*[Halaman ini sengaja dikosongkan]*

## LAMPIRAN B

Lampiran ini berisi script (naskah) simulasi pada perangkat lunak MATLAB yang digunakan.

### A. Script Simulasi 15 Elemen Array Antena dengan metode DE

```
clear all;
clc;
close all;

% 1. Penentuan parameter
N=15;      %Jumlah kolom/jumlah elemen dalam 1
individu
M=20;      %Jumlah baris/jumlah individu
d=0.5;     % jarak antar elemen
F=0.5;     % faktor mutasi
C=0.9;     % Crossover rate
theta_zero=0;
delta_s=-2*pi*d*cos(theta_zero);

% Matriks An (20 x 15)
An=[ones(M,1) randi([0 1], M,N-2)
ones(M,1)]; % ujung dan ujung bernilai 1,
tengah random

% Matriks posisi (181 x 15)
x=1:N;     % jumlah posisi untuk elemen
dalam 1 individu
n=(ones(181,1) * x)'; % jumlah posisi sebanyak
181x15
pos=n*d;   % Posisi yaitu jumlah posisi x jarak

% Matriks Phi (15 x 181)
ydeg=linspace(0,180,181); % dalam degree
yrad=deg2rad(ydeg);      % dalam rad
phi=ones(N,1)*yrad;

% 2. nilai objective function
```

```

% Nilai Far Field
MaxG=200; %Maksimum Generasi
SLLrank_gab=zeros(1,MaxG);
% G=0:1:100;
for Generasi=1:MaxG

    fix_var= (cos(2*pi*(pos.*cos(phi))+delta_s));
    FF_un = abs(An*fix_var)
    Best(Generasi,:) = An(1,:);

    for m=1:M

        FFnorm(m,:)=FF_un(m,:)/max(FF_un(m,:));

        FFdB(m,:)=20*log10(FFnorm(m,:));

        [q,indq]=sort((findpeaks(FFdB(m,:))), 'descend');
        MainB(m)=q(1); % Mainbeam 20 baris
        indexMainB(m)=indq(1); % index Mainbeam 20
baris

        SLL(m)=q(2); % SLL 20 baris
        indexSLL(m)=indq(2); % index SLL 20
baris
        m= m+1;
    end
% plot far field
figure(1)
plot(ydeg,FFdB)
grid on
xlabel('u (deg)')
ylabel('Far Field (dB)')
axis([-0 180 -30 0]);
iter=0;

        iter=iter+1;

        for m=1:M % Untuk tiap individu dlm
populasi
% Mutasi

```

```

R=randperm(M);
j=R(1);
k=R(2);
p=R(3);
u=R(4);
v=R(5);
if j==m
    j=R(6);
elseif k==m
    k=R(6);
elseif p==m
    p=R(6);
elseif u==m
    u=R(6);
elseif v==m
    v=R(6);
end
V=An(j, :)+F*(An(k, :)-An(p, :));

% Crossover, hasilnya masuk ke matriks U
jrand=floor(rand()*N+1);
for n=1:N
    R1=rand();
    if (R1<C || n==jrand)
        U(1,n)=V(1,n);
    else
        U(1,n)=An(m,n);
    end
end

% Seleksi
if abs(U(1, :)*fix_var)<abs(A(m, :)*fix_var)
    Tr=U(1, :);
else
    Tr=An(m, :);
end

% Mengganti hasil seleksi untuk mengganti baris
m pada matrik X
An(m, :)=Tr;

```

```

% Evaluasi tiap individu fitness value nya, dan
masukkan ke matrik Y
    Y(m,1)=abs(A(m,:)*fix_var);

    end

% Pilih nilai fitness value terkecil
    [y,ind1]=sort(Y,1);
    Y_min=y(1,1);
    [Ymin,ind] = min(Y);

% Plot figure dari iterasi
figure(2);
plot(iter,Ymin,'r');
xlabel('Iterasi ke-');
ylabel('Nilai SLL (dB)');
grid on;
hold on;

```

## B. Script Simulasi 30 Elemen Array Antena dengan metode DE

```

clear all;
clc;
close all;

% 1. Penentuan parameter
N=30;    %Jumlah kolom/jumlah elemen dalam 1
individu
M=20;    %Jumlah baris/jumlah individu
d=0.5;   % jarak antar elemen
F=0.5;   % faktor mutasi
C=0.9;   % Crossover rate
theta_zero=0;
delta_s=-2*pi*d*cos(theta_zero);

% Matriks An (20 x 30)
An=[ones(M,1) randi([0 1], M,N-2)
ones(M,1)];    % ujung dan ujung bernilai 1,
tengah random

```

```

% Matriks posisi (181 x 30)
x=1:N; % jumlah posisi untuk elemen
dalam 1 individu
n=(ones(181,1) * x)'; % jumlah posisi sebanyak
181x30
pos=n*d; % Posisi yaitu jumlah posisi x jarak

% Matriks Phi (30 x 181)
ydeg=linspace(0,180,181); % dalam degree
yrad=deg2rad(ydeg); % dalam rad
phi=ones(N,1)*yrad;

% 2. nilai objective function

% Nilai Far Field
MaxG=200; %Maksimum Generasi
SLLrank_gab=zeros(1,MaxG);
% G=0:1:100;
for Generasi=1:MaxG

    fix_var= (cos(2*pi*(pos.*cos(phi))+delta_s));
    FF_un = abs(An*fix_var)
    Best(Generasi,:) = An(1,:);

    for m=1:M

        FFnorm(m,:)=FF_un(m,:)/max(FF_un(m,:));

        FFdB(m,:)=20*log10(FFnorm(m,:));

        [q,indq]=sort((findpeaks(FFdB(m,:))), 'descend');
        MainB(m)=q(1); % Mainbeam 20 baris
        indexMainB(m)=indq(1); % index Mainbeam 20
baris

        SLL(m)=q(2); % SLL 20 baris
        indexSLL(m)=indq(2); % index SLL 20
baris

```

```

        m= m+1;
        end
% Plot Far Field
figure(1)
plot (ydeg,FFdB)
grid on
xlabel('u (deg)')
ylabel('Far Field (dB)')
axis([-0 180 -30 0]);
iter=0;
        iter=iter+1;

        for m=1:M % Untuk tiap individu dlm
populasi
% Mutasi
                R=randperm(M);
                j=R(1);
                k=R(2);
                p=R(3);
                u=R(4);
                v=R(5);
                if j==m
                        j=R(6);
                elseif k==m
                        k=R(6);
                elseif p==m
                        p=R(6);
                elseif u==m
                        u=R(6);
                elseif v==m
                        v=R(6);
                end
                V=An(j,:)+F*(An(k,:)-An(p,:));

% Crossover, hasilnya masuk ke matriks U
                jrand=floor(rand()*N+1);
                for n=1:N
                        R1=rand();
                        if (R1<C || n==jrand)

```

```

                                U(1,n)=V(1,n);
                                else
                                U(1,n)=An(m,n);
                                end
                                end

% Seleksi
    if abs(U(1,:)*fix_var)<abs(A(m,:)*fix_var)
        Tr=U(1,:);
    else
        Tr=An(m,:);
    end

% Mengganti hasil seleksi untuk mengganti baris
m pada matrik X
    An(m,:)=Tr;

% Evaluasi tiap individu fitness value nya, dan
masukkan ke matrik Y
    Y(m,1)=abs(A(m,:)*fix_var);

    end

% Pilih nilai fitness value terkecil
    [y,ind1]=sort(Y,1);
    Y_min=y(1,1);
    [Ymin,ind] = min(Y);

% Plot figure dari iterasi
figure(2);
plot(iter,Ymin,'r');
xlabel('Iterasi ke-');
ylabel('Nilai SLL (dB)');
grid on;
hold on;

```

### C. Script Simulasi 60 Elemen Array Antena dengan metode DE

```

clear all;
clc;
close all;

```

```

% 1. Penentuan parameter
N=60;      %Jumlah kolom/jumlah elemen dalam 1
individu
M=20;      %Jumlah baris/jumlah individu
d=0.5;     % jarak antar elemen
F=0.5;     % faktor mutasi
C=0.9;     % Crossover rate
theta_zero=0;
delta_s=-2*pi*d*cos(theta_zero);

% Matriks An (20 x 60)
An=[ones(M,1) randi([0 1], M,N-2)
ones(M,1)]; % ujung dan ujung bernilai 1,
tengah random

% Matriks posisi (181 x 60)
x=1:N;     % jumlah posisi untuk elemen
dalam 1 individu
n=(ones(181,1) * x)'; % jumlah posisi sebanyak
181x60
pos=n*d;   % Posisi yaitu jumlah posisi x jarak

% Matriks Phi (60 x 181)
ydeg=linspace(0,180,181); % dalam degree
yrad=deg2rad(ydeg);      % dalam rad
phi=ones(N,1)*yrad;

% 2. nilai objective function

% Nilai Far Field
MaxG=200; %Maksimum Generasi
SLlrank_gab=zeros(1,MaxG);
% G=0:1:100;
for Generasi=1:MaxG

    fix_var= (cos(2*pi*(pos.*cos(phi))+delta_s));
    FF_un = abs(An*fix_var)
    Best(Generasi,:) = An(1,:);

```

```

for m=1:M

    FFnorm(m, :)=FF_un(m, :)/max(FF_un(m, :));

    FFdB(m, :)=20*log10(FFnorm(m, :));

[q, indq]=sort((findpeaks(FFdB(m, :))), 'descend');
    MainB(m)=q(1);           % Mainbeam 20 baris
    indexMainB(m)=indq(1); % index Mainbeam 20
baris

    SLL(m)=q(2);           % SLL 20 baris
    indexSLL(m)=indq(2);  % index SLL 20
baris
    m= m+1;
end
% Plot Far Field
figure(1)
plot(ydeg, FFdB)
grid on
xlabel('u (deg)')
ylabel('Far Field (dB)')
axis([-0 180 -30 0]);
iter=0;

        iter=iter+1;

for m=1:M % Untuk tiap individu dlm
populasi
% Mutasi

        R=randperm(M);
        j=R(1);
        k=R(2);
        p=R(3);
        u=R(4);
        v=R(5);
        if j==m
            j=R(6);
        elseif k==m

```

```

        k=R(6);
    elseif p==m
        p=R(6);
    elseif u==m
        u=R(6);
    elseif v==m
        v=R(6);
    end
    V=An(j,:)+F*(An(k,:)-An(p,:));

% Crossover, hasilnya masuk ke matriks U
jrand=floor(rand()*N+1);
for n=1:N
    R1=rand();
    if (R1<C || n==jrand)
        U(1,n)=V(1,n);
    else
        U(1,n)=An(m,n);
    end
end

% Seleksi
if abs(U(1,:)*fix_var)<abs(A(m,:)*fix_var)
    Tr=U(1,:);
else
    Tr=An(m,:);
end

% Mengganti hasil seleksi untuk mengganti baris
m pada matrik X
    An(m,:)=Tr;

% Evaluasi tiap individu fitness value nya, dan
masukkan ke matrik Y
    Y(m,1)=abs(A(m,:)*fix_var);

end

% Pilih nilai fitness value terkecil
[y,ind1]=sort(Y,1);
Y_min=y(1,1);

```

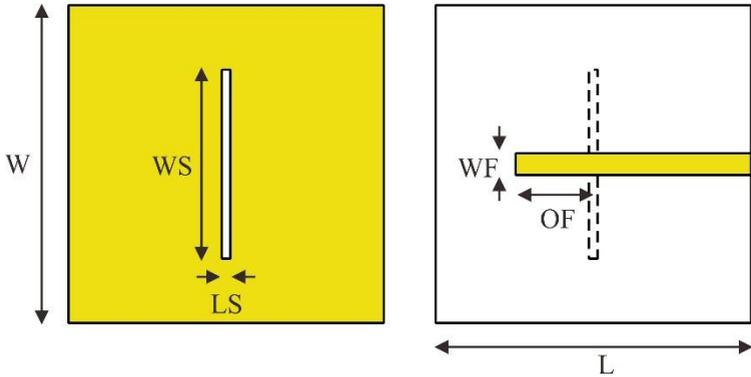
```
[Ymin,ind] = min(Y);

% Plot figure dari iterasi
figure(2);
plot(iter,Ymin,'r');
xlabel('Iterasi ke-');
ylabel('Nilai SLL (dB)');
grid on;
hold on;
```

*[Halaman ini sengaja dikosongkan]*

## LAMPIRAN C

Geometri antena mikrostrip slot



Ukuran	Nilai (mm)	Ukuran	Nilai (mm)
<b>W</b>	73.15	<b>WT</b>	2.93
<b>L</b>	73.15	<b>WF</b>	5
<b>WS</b>	43.5	<b>OF</b>	17
<b>LS</b>	2		

*[Halaman ini sengaja dikosongkan]*

## **BIOGRAFI PENULIS**



Radifan Aiman Nabil Hibatulhaqqi, lahir di Semarang, 24 April 1995. Penulis adalah putra sulung dari tiga bersaudara pasangan Herman Prasetyo dan N. Sri Herlina. Pada tahun 2007, penulis menamatkan jenjang pendidikan dasar di SDI Al-Fajar Bekasi. Penulis melanjutkan ke jenjang pendidikan SMP dan SMA di SMP-IT AL BINAA dan SMA-IT AL BINAA Bekasi, lulus pada tahun 2010 dan 2013. Melanjutkan ke jenjang pendidikan tinggi di Institut Teknologi Sepuluh Nopember (ITS) Surabaya pada Jurusan

Teknik Elektro Bidang Studi Telekomunikasi Multimedia. Selama aktif berkuliah di ITS, penulis mengikuti beberapa kepanitiaan mahasiswa dan sebagai asisten praktikum yang diadakan oleh Laboratorium Bidang Studi Telekomunikasi Multimedia.

*[Halaman ini sengaja dikosongkan]*