PETIR: Jurnal Pengkajian dan Penerapan Teknik Informatika
Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

Optimalisasi Biaya Perawatan Sistem *Hybrid* Turbin Angin dan Panel Surya Menggunakan Metode *Linear Programming* (Linprog) dan *Genetic Algorithm*

Muhammad Afdal[1]; Muchammad Yusuf[1]; Herminarto Nugroho[1)]*

1. Program Studi Teknik Elektro, Fakultas Teknologi Industri, Universitas Pertamina, DKI Jakarta 12220, Indonesia

[*)]Email: herminarto.nugroho@universitaspertamina.ac.id

Received: 31 Maret 2022 | Accepted: 8 September 2022 | Published: 28 November 2022

ABSTRACT

The hybrid system is an effort to answer the efficiency challenges of renewable energy systems. With the hybrid system, power supply disturbances can be controlled and make the system more reliable. In this study, the hybrid system used is a combination of solar panels and wind turbines. In the process of designing a hybrid system, analysis studies for many factors are certainly needed. The aim is to optimize the minimum cost with the right energy capacity. One of them is determining the number of solar panels and wind turbines to meet the daily and annual average power. To achieve this can use the optimization method. The method used in this research is linear programming, minimum function with constraints (fmincon), and genetic algorithms. From the research results it is known that the use of the linear programming method is easier and more effective than the minimum function with constraints (fmincon), and the genetic algorithm. This is because the objective function is a simple linear function, but all three yield the same global optimal and minimum cost value

Keywords: *linear programming (linprog); fmincon; genetic algorithm (ga); global optimum*

ABSTRAK

Sistem hibrid adalah salah satu upaya untuk menjawab tantangan efisiensi dari sistem energi baru terbarukan. Dengan adanya sistem hibrid stabilitas suplai daya dapat terkontrol dan menjadikan sistem lebih efisien. Pada penelitian ini sistem hibrid yang digunakan adalah gabungan antara panel surya dan turbin angin. Dalam proses perancangan sistem hibrid tentu dibutuhkan studi analisis untuk banyak faktor. Tujuannya adalah untuk optimalisasi biaya yang minimun dengan kapasitas energi yang tepat. Salah satunya adalah menentukan jumlah panel surya dan turbin angin untuk memenuhi daya rata-rata harian maupun pertahunnya. Untuk mencapai hal itu dapat menggunakan metode optimasi. Metode yang diapakai pada penelitian ini adalah *linear programming*, *function minimum with constrain (fmincon)*, dan *genetic algorithm (GA)*. Dari hasil penelitian diketahui bahwa penggunaan metode *linear programming* lebih mudah dan efektif daripada *function minimum with constrain (fmincon),* dan *genetic algorithm (GA).* Hal ini dikarenakan fungsi objektifnya adalah fungsi linear yang sederhana, namun ketiganya menhasilkan nilai *global optimum* dan nilai *cost minimum* yang sama.

Kata kunci: *linear programming (linprog); fmincon; genetic algorithm (ga); global optimum*

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

1. PENDAHULUAN

Di era industri 4.0 teknologi mengalami kemajuan yang sangat pesat. Salah satunya adalah teknologi di bidang energi terbarukan. Banyak negara yang berlomba-lomba mengimplementasikan dan mengoptimalkan energi terbarukan yang tersedia di negara masing-masing. Dari sekian banyak jenis energi terbarukan, ada dua jenis energi terbarukan yang paling banyak dikembangkan dan populer. Yakni, energi angin (*wind turbin*) dan energi matahari (*solar energy*). Kedua *renewable energy* tersebut banyak diaplikasikan secara bersamaan dengan menerapkan *hybrid energy system*. *Hybrid energy system* sendiri adalah sistem yang menggunakan dua jenis energi yang berbeda untuk memenuhi kebutuhan beban. Biasanya pada sistem ini dilengkapi sistem peyimpanan energi berupa baterai yang digunakan untuk menyimpan listrik. Baterai nantinya akan menyuplai listrik ketika energi yang dihasilkan oleh *solar panel* dan *wind turbine* tidak mencukupi beban yang dibutuhkan. Sistem ini memiliki banyak keunggulan yakni, efisiensi yang baik, handal, emisi yang kecil, lebih ramah lingkungan dan biayanya relatif lebih murah [1].

Dalam implementasinya, sistem hibrid yang menggabungkan turbin angin dan panel surya memiliki karateristiknya tersendiri. Pada turbin angin energi angin dimanfaatkan sebagai sumber satu-satunya. Daya listrik yang dihasilkan per hari atau jamnya oleh turbin angin sangat dipengaruhi oleh kecepatan angin di suatu lokasi dan kualitas turbin anginnya. Turbin angin dapat bekerja 24 jam dan pemeliharaannya tergolong tidak sulit, serta biaya perawatannya relatif murah . Selain itu, daya yang dihasilkan oleh turbin angin cukup besar jika dibandingkan dengan energi terbarukan dari panel surya [1]. Panel surya memanfaatkan energi matahari yang kemudian dikonversi oleh solar sel menjadi energi listrik. Pada sistem panel surya tingkat radiasi matahari sangat berpengaruh, yakni lamanya penyinaran matahari dalam satu hari. Secara umum panel surya dapat mengkonversi energi matahari dalam satu hari adalah lima jam. Selebihnya panel surya masih dapat menghasilkan listrik tetapi sangat kecil dalam satuan watt [2].

Dalam pengoperasian sistem hibrid tentu banyak aspek yang perlu diperhatikan. Salah satunya adalah biaya *maintenance* / perawatannya, mulai dari turbin angin, panel surya, baterai, dan komponen pendukung lainnya. Karenanya perlu dilakukan perhitungan dan analisis yang mendalam ketika proses perencanaan untuk mengoptimalkan biaya perawatan tadi. Untuk menentukan biaya *maintenance* yang optimal dapat menggunakan berbagai metode *optimization*. Pada penelitian kali ini, digunakan tiga metode optimasi yakni *linear programming* (linprog), *function minimum with constraint* (fmincon), dan *genetic algorithm* (GA) yang akan dijelaskan pada bagian 2 jurnal ini.

2. METODE/PERANCANGAN PENELITIAN

2.1. Rumusan Permasalahan Optimasi

Pada jurnal ini di angkat sebuah permasalahan optimasi dalam meminimalkan biaya *maintenance* dari sistem *hybrid* dari *renewable energy* yang menggabungkan panel surya dan *wind turbine*. Salah satu cara untuk menentukan biaya yang paling optimal atau dalam hal ini minimum pada *maintenance* adalah dengan mengubah komposisi/jumlah dari turbin angin atau solar panel itu sendiri. Langkah awal dari sebuah proses optimization adalah menganalisis permasalahan optimasi dengan menentukan *design variable*, *objective function*, dan *constraint*-nya.

Dari permasalahan optimasi yang ada diperoleh formula biaya perawatan atau *cost maintenance* (CM), berikut adalah persamaan untuk 1 tahun atau 365 hari:

$$C_M = 365\left(C_{PV,M}\sum_{t=1}^{24} N_{PV}P_{PV}(t)\Delta t + C_{WT,M}\sum_{t=1}^{24} N_{WT}P_{WT}(t)\Delta t\right) \quad (1)$$

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

Dengan keterangan, nilai, dan satuan parameter dapat dilihat pada Tabel 1 berikut:

Tabel 1. Nilai parameter dari permasalahan optimasi

Simbol	Keterangan Parameter	Nilai	Satuan
C_M	Biaya Perawatan Tahunan	-	$
$C_{PV,M}$	Biaya Perawatan Solar Panel	5	$/WH
$C_{WT,M}$	Biaya Perawatan Turbin Engin	20	$/WH
Δt	*Time Difference*	1	Jam
N_{PV}	Jumlah Solar Panel	-	Buah
N_{WT}	Jumlah turbin angin	-	Buah

Tabel 2 menunjukkan daya rata-rata yang diperlukan dalam waktu 24 jam serta data daya yang dapat dihasilkan oleh solar panel dan juga turbin angin [3],[4].

Tabel 2. Data Daya Rata-Rata yang Dibangkitkan dan Daya Beban untuk Setiap Jam dalam Satu Hari (kWh) Sebelum Dioptimasi

Waktu (t)	Pload (t)	PWT(t)	PPV (t)	ΔP(t)
1	1.39	0.58	0	-0.810
2	1.25	0.49	0	-0.760
3	1.19	0.48	0	-0.710
4	1.22	0.53	0	-0.690
5	1.34	0.47	0	-0.870
6	1.8	0.51	0	-1.290
7	2.66	0.46	0.0016	-2.198
8	2.9	0.46	0.0034	-2.437
9	2.52	0.61	0.0103	-1.899
10	2.21	0.76	0.0246	-1.425
11	2.05	1.1	0.0317	-0.918
12	1.94	1.53	0.0353	-0.375
13	1.82	1.67	0.0366	-0.113
14	1.71	1.89	0.0374	0.217
15	1.62	2.43	0.0368	0.847
16	1.65	2.45	0.0335	0.833
17	1.87	1.91	0.0242	0.064
18	2.29	1.76	0.0134	-0.517
19	2.58	1.57	0.0056	-1.004
20	2.6	1.16	0.0015	-1.438
21	2.54	0.87	0	-1.670
22	2.49	0.76	0	-1.730
23	2.28	0.74	0	-1.540
24	1.79	0.7	0	-1.090
Total	47.71	25.89	0.2959	-21.523

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

Dari permasalahan optimasi pada persamaan 1 dapat diformulasikan ke bentuk matematika optimasi, yang merupakan bentuk umum dari optimisasi. Berikut adalah persamaan umum optimasi terlihat pada persamaan 2.

$$\min_{x} f(x) \quad \text{s.t. } h(x) = 0 \; ; \; g(x) \leq 1 \tag{2}$$

Permaslahan minimalisasi cost maintenance dibuah ke bentuk umum menjadi berikut ini pada persamaan 3 :

$$\min_{N_1,N_2} \left[365 \left(\left(C_{PV,M} N_1 \sum_{t=1}^{24} P_{PV}(t) \Delta t \right) + \left(C_{WT,M} N_2 \sum_{t=1}^{24} P_{WT}(t) \Delta t \right) \right) \right] \quad \text{s.t. } \Delta P \geq 0 \; ; \; N_1, N_2 \geq 1 \tag{3}$$

Keterangan :

- *Design variable* = $x, N_1, N_2,$
- *Objective function* = $f(x), f(N)$
- *Inequality Constraint* = $g(x), \; \Delta P \geq 0 \; ; \; N_1, N_2 \geq 1$
- *Equality constraint* = $h(x)$

Dari *constraint* yang diperoleh dapat di sederhanakan secara matematis dengan persamaan selisih daya (ΔP). ΔP adalah selisih daya antara daya beban (P_{Load}) dengan total daya yang dibangitkan oleh panel surya (P_{PV}) dan turbin angin (P_{WT}). Gunakan persamaan 4 berikut untuk menghiitung selisih daya dengan asumsi waktu 1 hari (24 jam) berdasarkan data pada Tabel 2.

$$\Delta P = (N_1 P_{PV} + N_2 P_{WT}) - P_{Load} \tag{4}$$

Dengan syarat constrain yang harus dipenuhi yakni, selisih daya tidak boleh negatif. Karenanya syarat *constrain* tadi dapat di sederhakan menjadi berikut:

$$P_{Load} - \left(N_1 \sum_{t=1}^{24} P_{PV}(t) + N_2 \sum_{t=1}^{24} P_{WT}(t) \right) \geq 0$$

$$\left(N_1 \sum_{t=1}^{24} P_{PV}(t) + N_2 \sum_{t=1}^{24} P_{WT}(t) \right) \leq P_{Load} \tag{5}$$

Persamaan 5 dapat menggantikan *constrain* ΔP ≥ 0 pada persamaan 3. Dengan penyederhanaan formula matematis dari permasalahan optimasi tadi dapat ditulis ulang formulasinya menjadi persamaan 6 berikut:

$$\min_{N_1,N_2} \left[365 \left(\left(C_{PV,M} N_1 \sum_{t=1}^{24} P_{PV}(t) \Delta t \right) + \left(C_{WT,M} N_2 \sum_{t=1}^{24} P_{WT}(t) \Delta t \right) \right) \right] \tag{6}$$

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

s.t.

$$\left(N_1 \sum_{t=1}^{24} P_{PV}(t) + N_2 \sum_{t=1}^{24} P_{WT}(t)\right) \leq P_{Load}$$

$$N_1, N_2 \geq 1$$

Setelah memperoleh formula optimasinya, selanjutnya adalah menentukan metode yang tepat untuk memperoleh nilai optimum. Nantinya akan diperoleh kombinasi dari jumlah panel surya dan wind turbine. Karena system yang digunakan adalah *hybrid*, maka minimal turbin angin dan panel surya yang dipakai adalah 1. Namun, jika hanya masing-masing satu buah turbin angin dan panel surya yang dipakai. Maka beban daya yang diinginkan tidak akan tercapai. Oleh karena itu, selisih dari daya beban daya yang dibangkitkan dari system *hybrid* di harapkan tidak minus. Artinya selisih daya harus positif atau nol.

2.2. Metode Optimasi

Metode optimasi adalah metode yang digunakan untuk mencari solusi yang terbaik atau yang mendekati dari suatu permasalahan optimasi di dunia ini [5]. Metode yang dapat digunakan untuk menyelsaikan permasalahan optimasi sangatlah banyak. Dalam menentukan suatu metode yang tepat haruslah menganalisa permasalahan optimasi dengan merepresentasikannya kedalam bentuk formula umum, dengan menentukan *design variable*, fungsi objektif, dan *constrain* nya. Dalam penelitian ini penulis menggunakan tiga buah metode/ *toolbox* yakni, *linprog*, *fmincon*, dan *genetic algorithm*.

2.2.1. Linear Programming

Linear programming adalah metode optimasi yang paling umum digunakan dan salah satu metode yang mudah di aplikasikan pada permasalahan optimasi. *Linear programming* bertujuan untuk menemukan solusi yang optimal dari sautu permasalahan optimasi dengan syarat fungsi dan *constraint*-nya haruslah linear, baik itu *constrain* yang *linear equality* atau *inequaity* [6][7].

Jika melihat formula optimasi pada persamaan (6), sangat jelas bahwa fungsi $f(N)$ merupakan fungsi yang linier. Oleh karena itu, metode *linear programming* merupakan solusi yang cocok digunakan pada permasalahan optimasi pada penelitian ini. Pada *toolbox* linprog, permasalahan optimasi di penelitian ini menggunakan linear programming dengan *all type constrain*. Constraint yang dipakai adalah *inequality constarin*, dan batas bawah/ minimal (*lower bounds*) dengan *syntax* berikut [8].

```
x = linprog(f,A,b,Aeq,beq,lb,ub,options)[x,fval,exitflag,output] =
linprog(___)
```

Dengan acuan nilai parameter pada tabel 1 dan 2 , kode pemrograman dituliskan di MATLAB seperti berikut ini:

```
tic
% <<<<<<<<<Percobaan_1 Solve Linear Program With
Constrain>>>>>>>>>
options = optimoptions('linprog','Algorithm','dual-simplex');
% Menentukan parameter
Cpv = 5;
```

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

```
Cwt = 20;
Ppv = 295.9;
Pwt = 25890;
% Menentukan constrain
A = [-295.9, -25890];
b = [-47710];
lb = [1; 1];
ub = [];
Aeq = [];
beq = [];
% Membuat Fungsi Objektif
f_cost = [540017.5 188997000];
% Solve linear program
[x,f_opt,exitflag,output] =
linprog(f_cost,A,b,Aeq,beq,lb,ub,options)
toc
```

2.2.2. Fmincon

Toolbox fmincon merupakan toolbox di MATLAB yang digunakan untuk minimum dari fungsi multivariable dengan constrain. Tolls ini dapat digunakan untuk menyelesaikan permasalahan optimasi yang linear maupun non linear. Pada penelitian ini digunakan *syntax fmincon linear with all type constrain*, berikut adalah syntaxnya :

```
x = fmincon(fun,x0,A,b,Aeq,beq,lb,ub,nonlcon,options)
```

Toolbox fmincon memiliki 5 opsi algorithma yang bisa digunakan yakni, *interior-point'*, *'trustregion- reflective'*, *'sqp'*, *'sqp-legacy'*, dan *'active-set'*. Namun, pada kasus di penelitian ini dengan fungsi yang linear digunakan algoritma *interior point*, dan *active set*. Secara *default* pada *fmincon* algoritma yang dipakai adalah *interior point* [9]. Sedangkan algoritma *active set* efektif pada bebarapa masalah yang memiliki *constrain* yang kurang *smooth* [10] [7].

Dengan acuan nilai parameter pada tabel 1 dan 2 , kode pemrograman dituliskan di MATLAB seperti berikut ini :

```
tic
% Menentukan Parameter
Cpv = 5;
Cwt = 20;
Ppv = 295.9;
Pwt = 25890;
deltat = 1
%Fungsi Objektif
cost_fun =@(N)(365*((Cpv*N(1)*Ppv*deltat)+(Cwt*N(2)*Pwt*deltat)));
%set initial value dan constrain
N_0 =[1; 1];
lb = [1; 1];
ub = [];
```

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

```
A = [-295.9, -25890];
b = -47710;
Aeq = [];
beq = [];
nonlcon = [];
%Set optimization Process and run
[N_opt, fval,exitflag,output] =
fmincon(cost_fun,N_0,A,b,Aeq,beq,lb,ub,..
nonlcon,options)
toc
```

2.2.3. Genetic Algorithm

Genetic algorithm (GA) pada dasarnya meniru konsep evolusi dari seleksi alam yang di kemukakan Charles Darwin pada tahun 1859. Pada konsep ini *fitness* dari suatu individu menentukan keberlangsungan makhluk hidup, jika *fitness* nya tinggi maka individu tersebut akan bertahan dan jika *fitness* nya rendah maka individu tersebut akan tersingkir atau mati. Selain itu Darwin jua mengemukakan bahwa mutasi genetik, reproduksi, dan persilangan (*cross-over*) mempengaruhi seleksi alam [11].

Dari konsep dasar pemikiran tadi diaplikasikan dalam metode *optimization* dengan komputasi algoritma untuk memperoleh solusi yang paling optimal dari suatu permasalahan optimasi dengan banyaknya solusi yang memungkinkan. Dalam metode algoritma genetik hal yang perlu dilakukan pertama kali adalah menentukan populasi awal atau *initial population* [12], [13], [14]. Kemudian melalui iterasi akan diperoleh generasi baru degan tiga tahapan yaitu, *select parents*, *create off-spring* (*cross over*), dan mutasi. Proses algoritma genetik tersebut akan selesai jika semua *fitness* sudah merata atau tidak ada lagi individu yang bisa di eliminasi [11][15].

Pada penelitian ini *syntax* yang digunakan adalah optimasi dengan *linear constrain* dan *bounds* untu mencari solusi dan *function value*. Berikut adalah *syntax* yang digunakan [11].

```
[x,fval,exitflag,output,population]
=ga(cost_fun,nvars,A,b,Aeq,beq,..
    lb,ub,nonlcon,IntCon,options)
```

Dengan acuan nilai parameter pada tabel 1 dan 2 , kode pemrograman dituliskan di MATLAB seperti berikut ini :

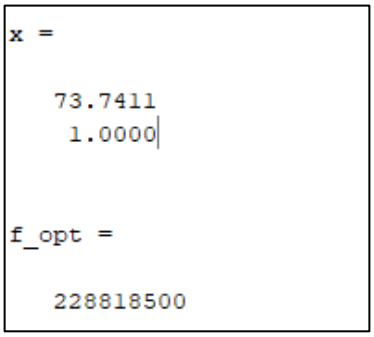
```
tic
options =
optimoptions('ga','DIsplay','iter','PlotFcn',@gaplotbestf,'TolFun'
,1e-
     09);
rng default % For reproducibility
%Menentukan Parameter
Cpv = 5;
Cwt = 20;
Ppv = 295.9;
Pwt = 25890;
```

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

```
nvars = 2;
% Membuat Fungsi Objektif
cost_fun =@(N)(365*((Cpv*N(1)*Ppv)+(Cwt*N(2)*Pwt)));
%set initial value
IntCon = 2;
A = [-295.9, -25890];
b = -47710;
lb = [1; 1];
ub = [];
Aeq = [];
beq = [];
nonlcon = [];
%Set optimization Process
[N,f_opt,exitflag,output,population] =
ga(cost_fun,nvars,A,b,Aeq,beq,…
     lb,ub,nonlcon,IntCon,options)
toc
```

3. HASIL DAN PEMBAHASAN

Metode *linear programming* adalah metode yang sangat simple dan mudah. Karena output yang dihasilkan langsung *global optimization* tanpa memakai *iteration*. Berikut adalah tampilan di *command window* hasil simulasi dengan *toolbox linear programming*.



Gambar 1. Tampilan *command window* hasil *Linear Programming*

Dari gambar di atas, nilai optimal dari jumlah turbin angin dan panel surya langsung diketahui yakni $N_1 = 73{,}7411 \approx 74$ dan $N_2 = 1$. Selain itu nilai dari fungsi objektif minimum dalam hal ini biaya *maintenance* pertahun untuk system hybrid dipeoleh sebesar US $ 228,818,500.

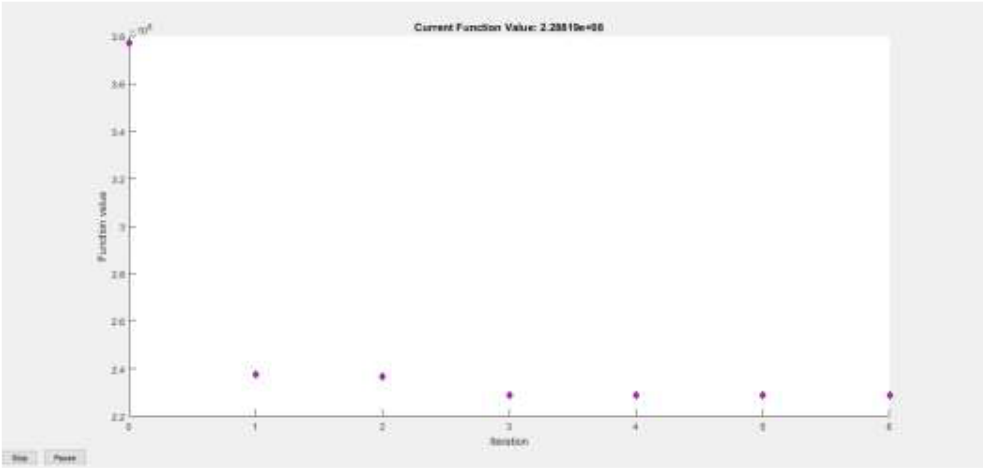
Sedangkan simulasi dengan metode *fmincon* dilakukan dengan menggunakan dua algorithma, yakni *interior point* (*default*) dan *active set* untuk membandingan hasil mana yang lebih optimal. Berikut adalah data perbandingan dengan variasi tolirasi fungsinya (TolFun) pada algorima *interior point* (*default*) dan *active set* pada Tabel 3 .

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

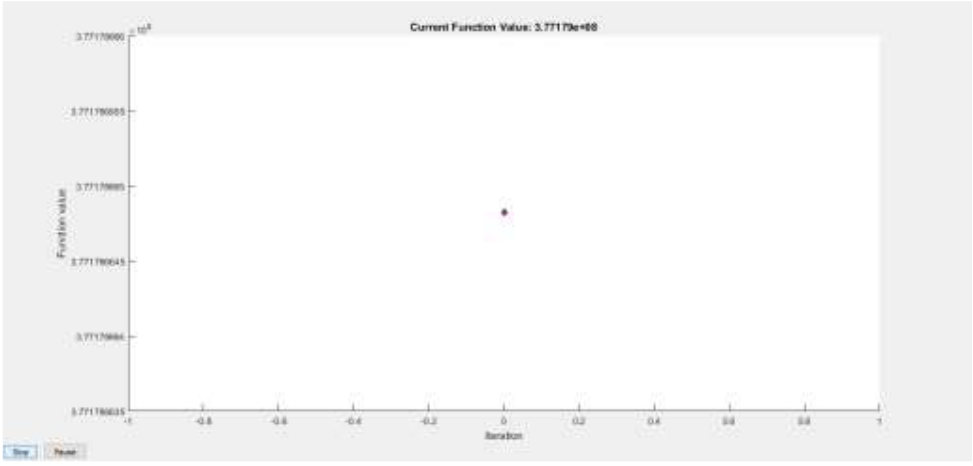
Tabel 3. Perbandingan penggunaan algorithma *interior point (default)* dan *active set*

TolFun	Algorithm							
	Interior Point				Active Set			
	Iteration	N1	N2	Maintenance Cost ($)	Iteration	N1	N2	Maintenance Cost ($)
1.00E+00	0	1,99	1,99	3.77E+08	2	73,74	1	2.29E+08
1.00E-01	0	1,99	1,99	3.77E+08	2	73,74	1	2.29E+08
1.00E-02	2	88,55	1	2.37E+08	2	73,74	1	2.29E+08
1.00E-03	2	88,55	1	2.37E+08	2	73,74	1	2.29E+08
1.00E-04	2	88,55	1	2.37E+08	2	73,74	1	2.29E+08
1.00E-05	2	88,55	1	2.37E+08	2	73,74	1	2.29E+08
1.00E-06	3	73,82	1	2.29E+08	2	73,74	1	2.29E+08
1.00E-07	4	73,74	1	2.29E+08	2	73,74	1	2.29E+08
1.00E-08	5	73.74	1	2.29E+08	2	73,74	1	2.29E+08
1.00E-09	5	73.74	1	2.29E+08	2	73,74	1	2.29E+08
1.00E-10	6	73.74	1	2.29E+08	2	73,74	1	2.29E+08

Berikut adalah hasil simulasi optimasi untuk masing-masing algoritma *interior point* dan *active set*:

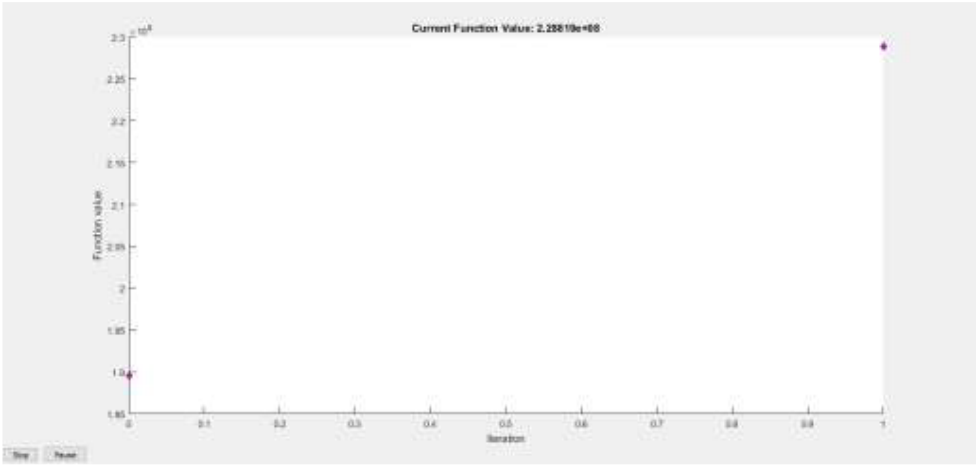


Gambar 2. Saat Fungsi Toleransinya (*Tolfun*) 1e-1 Pada Algorithma *Interior Point*



Gambar 3. Hasil optimasi Saat Nilai Outputnya Stabil Setelah Iterasi Ke 4 dan Seterusnya Pada *Interior Point*

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634



Gambar 4. Output Pada Algorithma *Active set* Selalu Sama Berapapun *TolFun* Yang Diberikan

Dari hasil simulasi diketahui dengan menggunakan algoritma *active set* langsung diperoleh nilai global optimum dan nilai objektifnya dengan jumlah iterasi yang sedikit yakni 2. Sedangkan pada set default internal point yang diperoleh adalah *local optima* dahulu pada toleransi fungsi awal dan setelah iterasi ke 4 menunjukkan nilai *global optimum* yang stabil. Selain itu, pada *toolbox fmincon* juga terdapat nilai inisiasi awal yang di set terlebih dahulu secara *random*. Perlu dilihat pengaruh perubahan inisiasi awal N_0 pada output optimasinya seperti yang terlihat pada tabel 3 berikut.

Tabel 4. Pengaruh Inisiasi Awal Pada Output Optimasi Pada Algorithm *Interior Point* dan *Active Set*

Inisiasi awal	Interior Point			Waktu Komputasi	Active Set			Waktu Komputasi
	Iterasi	N1	N2		Iterasi	N1	N2	
[0,0]	6	73,74	1	0,383898	2	73,74	1	0,281346
[1,1]	6	73,74	1	0,355789	2	73,74	1	0,26922
[10,15]	7	73,74	1	0,530265	2	73,74	1	0,318385
[20,1]	6	73,74	1	0,411219	2	73,74	1	0,298793
[40,70]	7	73,74	1	0,449073	2	73,74	1	0,366680
[30,30]	6	73,74	1	0,409259	2	73,74	1	0,363358
[74,1]	6	73,74	1	0,443038	2	73,74	1	0,328205
Rata-rata				0,431772				0,317998

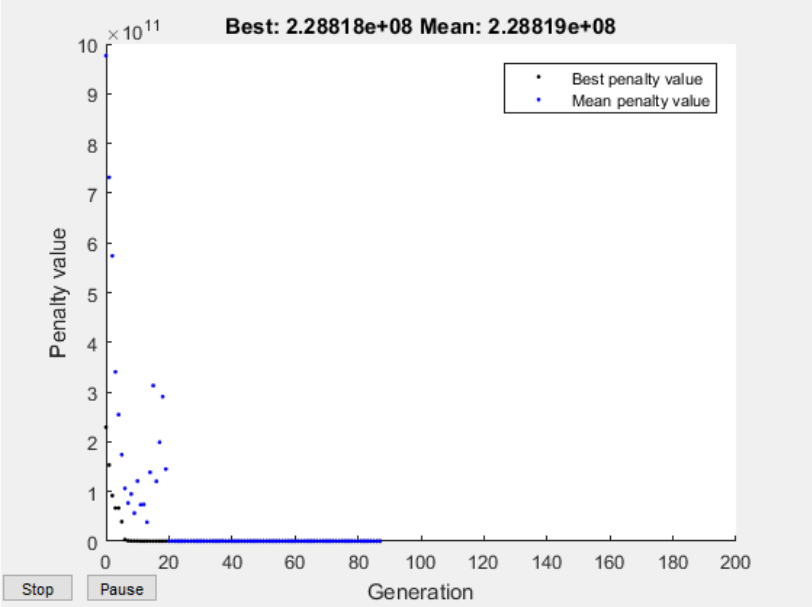
Dari hasil perbandingan Tabel 4 diketahui inisiasi awal berpengaruh pada algorithma *interior point*, berbeda dengan algoritma *active set* yang iterasinya sama. Untuk rata-rata waktu komputasi yang ada menggunakan *interior point* membutuhkan waktu sedikit lebih lama jika dibandingkan dengan *active set*. Hal tersebut sesuai dengan karakteristik *active set* yang menambah kecepatan komputasi dan efektif untuk problem dengan *constrain* yang *nonsmooth*. Maka dapat kaitkan bahwa pada permasalahan optimasi di penelitian ini menggunakan algoritma *active set* lebih efektif.

Pada simulasi dengan metode *genetic algorithm* dilakukan dengan memvariasikan fungsi toleransinya dan melihat apa pengaruhnya terhadap generasi dan populasinya.

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

Tabel 5. Pengaruh perubahan *TolFun* Pada Metode *Genetic Algotrithm*

TolFun	Jumlah Generasi	Jumlah Wind Turbine dan Solar Panel		Maintenance Cost ($)	Waktu Komputasi
		N1	N2		
1.00E+00	56	73,74	1	2.29E+08	2.034671
1.00E-01	57	73,74	1	2.29E+08	2.105161
1.00E-02	61	73,74	1	2.29E+08	2.152402
1.00E-03	68	73,74	1	2.29E+08	2.274878
1.00E-04	70	73,74	1	2.29E+08	2.591114
1.00E-05	70	73,74	1	2.29E+08	2.274209
1.00E-06	78	73,74	1	2.29E+08	2.552632
1.00E-07	79	73,74	1	2.29E+08	2.599431
1.00E-08	86	73,74	1	2.29E+08	2.659957
1.00E-09	87	73,74	1	2.29E+08	2.778511
1.00E-10	94	73,74	1	2.29E+08	2.955407



Gambar 5. Grafik Point Dari Jumlah Generasi Dengan *Penalty Value* atau *Cost Maintanance* (CM)

Jika melihat Gambar 4 dan 5 terlihat bahwa di generasi awal *penalti value* atau nilai dari *cost function*nya sangat besar mencapai 1011. Namun, dengan bergantinya generasi selanjutnya penalti *value* tadi turun sampai pada titik nilai minimum di 2.288e+08, pada generasi ke 24 nilanya mulai stabil di angka tersebut. Hal ini terjadi karena *fitness* sudah merata atau tidak ada lagi individu yang bisa di eliminasi, dengan kata lain generasi setelah 24 sudah merata.

Dari hasil pengujian *genetic algorithm* diketahui bahwa fungsi toleransi mempengaruhi jumlah generasinya, semakin kecil toleransinya maka generasi akan semakin banyak, begitupun sebaliknya. Fungsi toleransi juga mempengaruhi waktu komputasinya, semakin kecil toleransinya maka waktu komputasi akan semakin lama. Namun, *global optimum* dan nilai fungsi obkejtifnya tetap sama dan langsung diperoleh.

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

4. KESIMPULAN DAN SARAN

Dari hasil penelitian ini diketahui bahwa, untuk permasalahan optimasi menentukan biaya minimum dalam perawatan tahunan sistem energi hibrid panel surya dan turbin angin dapat menggunakan tiga metode/*toolbox* yang tersedia pada MATLAB. Yakni, *linear programming (linprog), fmincon,* dan *genetic algorithm (ga).* Ketiga metode tersebut menghasilkan nilai optimal yang sama untuk jumlah panel surya $N_1 = 73{,}74$, karena jumlah panel surya bilangan bulat dalam satuan *pieces* sehingga digenapkan keatas menjadi $N_1 = 74$ dan jumlah turbin angin $N_2 = 1$. Dengan asumsi daya harian tidak ada deficit atau kekurangan suplai daya ($\Delta P \geq 0$) maka *cost minimum* pertahunnya untuk perawatan *system hybrid* ini adalah US $ US $ 228,818,000.

DAFTAR PUSTAKA

[1] H. Harmini and T. Nurhayati, "Pemodelan sistem pembangkit hybrid energi solar dan angin," *Elektrika*, vol. 10, no. 2, pp. 28–32, 2018.

[2] D. Almanda and B. P. Piliang, "Perbandingan Sistem Pendingin pada Konsentrasi Water Coolant, Air Mineral, dan Air Laut Menggunakan Panel Surya Fleksibel Monocrystaline 20 Wp," *Resist. (elektRonika kEndali Telekomun. tenaga List. kOmputeR)*, vol. 2, no. 2, pp. 73–82, 2019.

[3] A. Suryadi, P. T. Asmoro, and A. Solihin, "Hybrid electric power plant using wind turbine savonius helix and solar cell as an alternative power source in the lightning tower at flashing lights," *ADI J. Recent Innov.*, vol. 1, no. 1, pp. 1–6, 2019.

[4] D. K. Geleta and M. S. Manshahia, "Artificial bee colony-based optimization of hybrid wind and solar renewable energy system," in *Handbook of research on energy-saving technologies for environmentally-friendly agricultural development*, IGI Global, 2020, pp. 429–453.

[5] S. S. Rao, *Engineering optimization: theory and practice*. John Wiley \& Sons, 2019.

[6] S. Aji, K. Soemadi, and F. H. Mustofa, "Optimisasi Keuntungan Menggunakan Linear Programming di PT Pertamina Refinery Unit (RU) VI Balongan," *REKA Integr.*, vol. 1, no. 3, 2013.

[7] R. H. Byrd, N. I. M. Gould, J. Nocedal, and R. A. Waltz, "An algorithm for nonlinear optimization using linear programming and equality constrained subproblems," *Math. Program.*, vol. 100, no. 1, pp. 27–48, 2004.

[8] N. Ploskas, N. Samaras, and others, *Linear Programming Using MATLAB®*, vol. 127. Springer, 2017.

[9] Y. Ye, *Interior point algorithms: theory and analysis*, vol. 44. John Wiley \& Sons, 2011.

[10] The Mathworks, "{MATLAB} {R}2015a {D}ocumentation." 2015.

[11] M. Saraswat and A. K. Sharma, "Genetic Algorithm for optimization using MATLAB," *Int. J. Adv. Res. Comput. Sci.*, vol. 4, no. 3, pp. 155–159, 2013.

[12] N. F. Istighfarin, R. A. Rahmastati, and H. Nugroho, "Penerapan Metode Particle Swarm Optimization (PSO) Dan Genetic Algorithm (GA) Pada Sistem Optimasi Visible Light Communication (VLC) Untuk Menentukan Posisi Robot," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 11, no. 1, pp. 279–286, 2020.

[13] H. Nugroho, C. Aditya, and S. Nungsizu, "Penerapan Metode Genetic Alghorithm untuk Meminimalkan Biaya Perawatan Sistem Pembangkit Energi Hibrid Solar Panel dan Turbin Angin," *ENERGI \& KELISTRIKAN*, vol. 13, no. 2, pp. 172–177, 2021.

[14] M. R. Fariez, F. G. Maulana, and H. Nugroho, "Penggunaan Metode Optimasi Genetic Algorithm dalam Penentuan Letak Turbin Angin," *J. Teknol.*, vol. 2, no. 2, 2020.

[15] I. Permadi and others, "Penerapan Algoritma Genetika untuk Optimasi Penjadwalan Tebangan

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1634

Hutan," *JUITA J. Inform.*, vol. 1, no. 1, 2010.