

LAMPIRAN 1
LISTING PROGRAM

Data Bus, Daya, Panjang Saluran dan Jumlah Pelanggan

Bus1	0	0	0
Bus 2	0	1,484	0
Bus 3	0	5,139	0
Bus 4	38	0,29	85
Bus 5	84	0,137	84
Bus 6	0	0,489	0
Bus 7	84	0,1222	1
Bus 8	0	0,2555	45
Bus 9	0	0,302	0
Bus 10	168	0,327	653
Bus 11	104	0,513	22
Bus 12	0	0,69	0
Bus 13	47	0,068	0
Bus 14	198	0,068	162
Bus 15	0	0,327	606
Bus 16	0	0,141	0
Bus 17	0	0,053	0
Bus 18	87	0,053	75
Bus 19	0	0,15	0
Bus 20	99	0,15	1
Bus 21	0	0,509	0
Bus 22	170	0,126	567
Bus 23	0	0,301	489
Bus 24	84	0,301	624
Bus 25	156	0,724	0
Bus 26	0	0,205	0
Bus 27	0	0,332	0
Bus 28	203	0,213	454
Bus 29	152	0,276	262
Bus 30	7	0,14	15
Bus 31	32	0,223	37
Bus 32	0	0,748	0
Bus 33	67	0,292	83
Bus 34	0	0,4	0
Bus 35	0	0,45	0
Bus 36	74	0,542	277
Bus 37	17	0,115	2
Bus 38	6	0,358	12
Bus 39	131	0,327	547
Bus 40	0	0,846	0
Bus 41	75	0,297	1
Bus 42	0	0,217	0
Bus 43	0	0,217	0
Bus 44	79	0,434	169
Bus 45	0	0,296	0

Bus 46	117	0,471	365
Bus 47	100	0,341	446
Bus 48	97	0,947	471
Bus 49	0	0,285	0
Bus 50	40	0,285	183
Bus 51	0	0,4	0
Bus 52	59	0,443	324
Bus 53	12	0,806	97
Bus 54	0	0,157	0
Bus 55	26	0,137	180
Bus 56	32	0,455	180
Bus 57	130	0,466	548
Bus 58	47	1,691	264
Bus 59	0	0,541	0
Bus 60	107	1,201	613
Bus 61	0	1,323	0
Bus 62	64	1,281	419
Bus 63	26	0,769	189
Bus 64	50	0,491	294
Bus 65	54	0,909	332
Bus 66	19	0,194	105
Bus 67	73	0,194	6
Bus 68	0	0,154	0
Bus 69	0	0,409	0
Bus 70	53	0,286	264
Bus 71	120	0,915	469
Bus 72	0	0,727	0
Bus 73	29	0,143	125
Bus 74	0	0,143	0
Bus 75	0	0,094	0
Bus 76	65	0,094	383
Bus 77	113	0,949	496
Bus 78	184	0,591	321
Bus 79	0	0,051	0
Bus 80	0	0,051	0
Bus 81	132	0,268	486
Bus 82	67	0,204	280
Bus 83	0	0,14	0
Bus 84	0	0,834	0
Bus 85	0	0,834	0
Bus 86	94	0,390	556
Bus 87	0	0,398	0
Bus 88	7	1,786	46
Bus 89	160	0,115	839
Bus 90	0	0,234	0
Bus 91	86	0,234	509
Bus 92	33	0,889	239

Bus 93	65	0,913	471
Bus 94	100	0,575	436
Bus 95	0	0,057	0
Bus 96	100	0,273	1
Bus 97	0	0,027	0
Bus 98	126	0,066	1
Bus 99	0	0,103	0
Bus 100	165	0,103	652
Bus 101	116	0,652	179
Bus 102	0	0,122	0
Bus 103	0	0,603	0
Bus 104	0	0,192	0
Bus 105	24	0,168	154
Bus 106	60	0	271
Bus 107	74	0,18	297
Bus 108	0	1,192	0
Bus 109	123	0,146	320
Bus 110	67	0,255	481
Bus 111	0	0,5	0
Bus 112	61	0,5	305
Bus 113	0	0,139	0
Bus 114	18	1,999	117
Bus 115	42	0,107	258
Bus 116	109	0,5450	547
Bus 117	51	0,411	227
Bus 118	663	1,268	224

Listing Program Kondisi Eksisting

```
function [saidifi,Saidi,saiditot,Saifi,saifitot,fitness,fitL] =
costSA(posisi)
% konstanta laju kegagalan
l_sutm=0.2; r_sutm=3;
l_sakelar_pisah=0.003; r_sakelar_pisah=10;
l_penutup_balik=0.005; r_penutup_balik=10;
l_bus=0; r_bus=0;
l_trafo_distribusi=0.005; r_trafo_distribusi=10;
l_windturbine = 0.02; r_windturbine = 50;
l_mikrohidro = 0.032; r_mikrohidro = 200;

load busdata;
assignin('base','busdata',busdata);

S = zeros(size(busdata, 1), 1);
U = zeros(size(busdata, 1), 1);
S(1) = 0.643;
U(1) = (S(1)*2);
S(2) = l_sakelar_pisah + (busdata(2,3)*l_sutm) + S(1);
U(2) = (l_sakelar_pisah*r_sakelar_pisah) + (busdata(2,3)*l_sutm*r_sutm) +
U(1);
for i=3:size(busdata,1)
if busdata(i,1)== 6 %bus6
    S(i) = (busdata(i,3)*l_sutm)+ S(3);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(3);
elseif busdata(i,1)== 11 %bus11
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(9);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdata(i,3)*l_sutm*r_sutm) + U(9);
elseif busdata(i,1)== 17 %bus17
    S(i) = (busdata(i,3)*l_sutm) + S(13);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(13);
elseif busdata(i,1)== 22 %bus22
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(19);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdata(i,3)*l_sutm*r_sutm) + U(19);
elseif busdata(i,1)== 26 %bus26
    S(i) = (busdata(i,3)*l_sutm) + S(12);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(12);
elseif busdata(i,1)== 32 %bus32
    S(i) = (busdata(i,3)*l_sutm) + S(26);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(26);
elseif busdata(i,1)== 38 %bus38
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdata(i,3)*l_sutm*r_sutm) + U(35);
elseif busdata(i,1)== 39 %bus39
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdata(i,3)*l_sutm*r_sutm) + U(35);
elseif busdata(i,1)== 54 %bus54
    S(i) = (busdata(i,3)*l_sutm) + S(51);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(51);
elseif busdata(i,1)== 65 %bus65
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(61);
```

```

    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
    (busdata(i,3)*l_sutm*r_sutm) + U(61);
elseif busdata(i,1)== 67 %bus67
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(43);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
    (busdata(i,3)*l_sutm*r_sutm) + U(43);
elseif busdata(i,1)== 75 %bus75
    S(i) = (busdata(i,3)*l_sutm) + S(68);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(68);
elseif busdata(i,1)== 95 %bus95
    S(i) = (busdata(i,3)*l_sutm) + S(83);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(83);
elseif busdata(i,1)== 117 %bus117
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(103);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
    (busdata(i,3)*l_sutm*r_sutm) + U(103);
elseif busdata(i,1)== 21 && busdata(i,1)== 23 && busdata(i,1)== 42 &&
busdata(i,1)== 74 && busdata(i,1)== 87 && busdata(i,1)== 97 &&
busdata(i,1)== 99 %bus21, bus23, bus42, bus74, bus87, bus97, bus99
    S(i) = (busdata(i,3)*l_sutm) + S(i-2);
    U(i) = (busdata(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdata(i,1)== 11 && busdata(i,1)== 29 && busdata(i,1)== 53 &&
busdata(i,1)== 89 && busdata(i,1)== 106 && busdata(i,1)== 115 %bus11,
bus29, bus53, bus89, bus106, bus115
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(i-2);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
    (busdata(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdata(i,1)== 16 && busdata(i,1)== 40 && busdata(i,1)== 69 &&
busdata(i,1)== 72 && busdata(i,1)== 102 %bus16, bus40, bus69, bus72, bus102
    S(i) = l_sakelar_pisah + (busdata(i,3)*l_sutm) + S(i-1);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdata(i,3)*l_sutm*r_sutm) +
U(i-1);
elseif busdata(i,1)== 8 && busdata(i,1)== 45 %bus8, bus45
    S(i) = l_sakelar_pisah + (busdata(i,3)*l_sutm) + S(i-2);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdata(i,3)*l_sutm*r_sutm) +
U(i-2);
elseif busdata(i,1)== 3 && busdata(i,1)== 9 && busdata(i,1)== 12 &&
busdata(i,1)== 13 && busdata(i,1)== 19 && busdata(i,1)== 27 &&
busdata(i,1)== 35 && busdata(i,1)== 43 && busdata(i,1)== 49 &&
busdata(i,1)== 51 && busdata(i,1)== 59 && busdata(i,1)== 61 &&
busdata(i,1)== 68 && busdata(i,1)== 79 && busdata(i,1)== 80 &&
busdata(i,1)== 83 && busdata(i,1)== 84 && busdata(i,1)== 85 &&
busdata(i,1)== 90 && busdata(i,1)== 103 && busdata(i,1)== 104 &&
busdata(i,1)== 108 && busdata(i,1)== 111 && busdata(i,1)== 113
    S(i) = (busdata(i,3)*l_sutm) + S(i-1);
    U(i) = (l_bus*r_bus)+ (busdata(i,3)*l_sutm*r_sutm) + U(i-1);
else
    S(i) = l_trafo_distribusi + (busdata(i,3)*l_sutm) + S(i-1);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
    (busdata(i,3)*l_sutm*r_sutm)+ U(i-1);
end
dx = find(posisi==busdata(i,1),1);
if ~isempty(dx)
    S(i) = (busdata(i,3)*l_sutm) + l_sakelar_pisah;
    U(i) = (busdata(i,3)*l_sutm*r_sutm)+ (l_sakelar_pisah*r_sakelar_pisah);
end
end
end

```

```

Saidi = zeros(size(busdata,1),1);
Saifi = zeros(size(busdata,1),1);
fitL = zeros(size(busdata,1),1);
jumpel = sum(busdata(:,4));

for i=1:size(busdata,1)
    Saifi(i) = (S(i)*busdata(i,4))/jumpel;
    Saidi(i) = (U(i)*busdata(i,4))/jumpel;
    fitL(i) = 1/(Saifi(i)*Saidi(i));
end

assignin('base','Saifi',Saifi);
assignin('base','Saidi',Saidi);
assignin('base','Jumpel',jumpel');

%Jumlah SAIFI total
saifitot = sum(Saifi);
assignin('base','saifitot',saifitot');

%jumlah SAIDI total
saiditot = sum(Saidi);
assignin('base','saiditot',saiditot');

saidifi = saifitot*saiditot;
assignin('base','saidifi',saidifi');

%Nilai Fitness
fitness = 1/(saiditot*saifitot);
assignin('base','fitness',fitness');

```

Listing Program Kondisi Grid – Wind Turbine

```
function [saidifi,Saidi,saiditot,Saifi,saifitot,fitness,fitL] =
costDG(posisi)
% konstanta laju kegagalan
l_sutm=0.2; r_sutm=3;
l_sakelar_pisah=0.003; r_sakelar_pisah=10;
l_bus=0; r_bus=0;
l_trafo_distribusi=0.005; r_trafo_distribusi=10;
l_windturbine = 0.02; r_windturbine = 50;

load busdataDG;
assignin('base','busdataDG',busdataDG);

S = zeros(size(busdataDG, 1), 1);
U = zeros(size(busdataDG, 1), 1);
S(1) = 0.643;
U(1) = (S(1)*2);
S(2) = l_sakelar_pisah + (busdataDG(2,3)*l_sutm) + S(1);
U(2) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataDG(2,3)*l_sutm*r_sutm) +
U(1);
for i=3:size(busdataDG,1)
if busdataDG(i,1)== 6 %bus 6
    S(i) = (busdataDG(i,3)*l_sutm)+ S(3);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(3);
elseif busdataDG(i,1)== 11 %bus 11
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(9);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(9);
elseif busdataDG(i,1)== 17
    S(i) = (busdataDG(i,3)*l_sutm) + S(13);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(13);
elseif busdataDG(i,1)== 22
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(19);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(19);
elseif busdataDG(i,1)== 26
    S(i) = (busdataDG(i,3)*l_sutm) + S(12);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(12);
elseif busdataDG(i,1)== 32
    S(i) = (busdataDG(i,3)*l_sutm) + S(26);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(26);
elseif busdataDG(i,1)== 38
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(35);
elseif busdataDG(i,1)== 39
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(35);
elseif busdataDG(i,1)== 54
    S(i) = (busdataDG(i,3)*l_sutm) + S(51);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(51);
elseif busdataDG(i,1)== 65
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(61);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(61);
```



```

elseif busdataDG(i,1)== 67
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(43);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(43);
elseif busdataDG(i,1)== 75
    S(i) = (busdataDG(i,3)*l_sutm) + S(68);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(68);
elseif busdataDG(i,1)== 95
    S(i) = (busdataDG(i,3)*l_sutm) + S(83);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(83);
elseif busdataDG(i,1)== 118
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(117);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(117);
elseif busdataDG(i,1)== 21 && busdataDG(i,1)== 23 && busdataDG(i,1)== 42
&& busdataDG(i,1)== 74 && busdataDG(i,1)== 87 && busdataDG(i,1)== 97 &&
busdataDG(i,1)== 99
    S(i) = (busdataDG(i,3)*l_sutm) + S(i-2);
    U(i) = (busdataDG(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdataDG(i,1)== 11 && busdataDG(i,1)== 29 && busdataDG(i,1)== 53 &&
busdataDG(i,1)== 89 && busdataDG(i,1)== 106 && busdataDG(i,1)== 115
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(i-2);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdataDG(i,1)== 16 && busdataDG(i,1)== 40 && busdataDG(i,1)== 69 &&
busdataDG(i,1)== 72 && busdataDG(i,1)== 102
    S(i) = l_sakelar_pisah + (busdataDG(i,3)*l_sutm) + S(i-1);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataDG(i,3)*l_sutm*r_sutm)
+ U(i-1);
elseif busdataDG(i,1)== 8 && busdataDG(i,1)== 45
    S(i) = l_sakelar_pisah + (busdataDG(i,3)*l_sutm) + S(i-2);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataDG(i,3)*l_sutm*r_sutm)
+ U(i-2);
elseif busdataDG(i,1)== 3 && busdataDG(i,1)== 9 && busdataDG(i,1)== 12 &&
busdataDG(i,1)== 13 && busdataDG(i,1)== 19 && busdataDG(i,1)== 27 &&
busdataDG(i,1)== 35 && busdataDG(i,1)== 43 && busdataDG(i,1)== 49 &&
busdataDG(i,1)== 51 && busdataDG(i,1)== 59 && busdataDG(i,1)== 61 &&
busdataDG(i,1)== 68 && busdataDG(i,1)== 79 && busdataDG(i,1)== 80 &&
busdataDG(i,1)== 83 && busdataDG(i,1)== 84 && busdataDG(i,1)== 85 &&
busdataDG(i,1)== 90 && busdataDG(i,1)== 103 && busdataDG(i,1)== 104 &&
busdataDG(i,1)== 108 && busdataDG(i,1)== 111 && busdataDG(i,1)== 113
    S(i) = (busdataDG(i,3)*l_sutm) + S(i-1);
    U(i) = (l_bus*r_bus)+ (busdataDG(i,3)*l_sutm*r_sutm) + U(i-1);
elseif busdataDG(i,1)== 117
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(103);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm)+ U(103);
elseif busdataDG(i,1)== 119
    S(i) = l_trafo_distribusi + l_windturbine + (busdataDG(i,3)*l_sutm)+
S(117);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi) +
(l_windturbine*r_windturbine) + (busdataDG(i,3)*l_sutm*r_sutm) + U(117);
else
    S(i) = l_trafo_distribusi + (busdataDG(i,3)*l_sutm) + S(i-1);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataDG(i,3)*l_sutm*r_sutm)+ U(i-1);
end

```

```

dx = find(posisi==busdataDG(i,1),1);
if ~isempty(dx)
    S(i) = S(i-1)+(busdataDG(i,3)*l_sutm) + l_sakelar_pisah;
    U(i) = U(i-1)+(busdataDG(i,3)*l_sutm*r_sutm)+
(l_sakelar_pisah*r_sakelar_pisah);
end
end

Saidi = zeros(size(busdataDG,1),1);
Saifi = zeros(size(busdataDG,1),1);
fitL = zeros(size(busdataDG,1),1);
jumpel = sum(busdataDG(:,4));

for i=1:size(busdataDG,1)
    Saifi(i) = (S(i)*busdataDG(i,4))/jumpel;
    Saidi(i) = (U(i)*busdataDG(i,4))/jumpel;
    fitL(i) = 1/(Saifi(i)*Saidi(i));
end

assignin('base','Saifi',Saifi);
assignin('base','Saidi',Saidi);
assignin('base','Jumpel',jumpel');

%SAIFI total
saifitot = sum(Saifi);
assignin('base','saifitot',saifitot');

%SAIDI total
saiditot = sum(Saidi);
assignin('base','saiditot',saiditot');

saidifi = saifitot*saiditot;
assignin('base','saidifi',saidifi');

%Nilai Fitness
fitness = 1/(saiditot*saifitot);
assignin('base','fitness',fitness');

```

Listing Program Kondisi Grid – Mikrohidro

```
function [saidifi,Saidi,saiditot,Saifi,saifitot,fitness,fitL] =
costDG(posisi)
% konstanta laju kegagalan
l_sutm=0.2; r_sutm=3;
l_sakelar_pisah=0.003; r_sakelar_pisah=10;
l_bus=0; r_bus=0;
l_trafo_distribusi=0.005; r_trafo_distribusi=10;
l_mikrohidro = 0.032; r_mikrohidro = 200;

load busdataM;
assignin('base','busdataM',busdataM);

S = zeros(size(busdataM, 1), 1);
U = zeros(size(busdataM, 1), 1);
S(1) = 0.643;
U(1) = (S(1)*2);
S(2) = l_sakelar_pisah + (busdataM(2,3)*l_sutm) + S(1);
U(2) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataM(2,3)*l_sutm*r_sutm) +
U(1);
for i=3:size(busdataM,1)
if busdataM(i,1)== 6 %bus 6
    S(i) = (busdataM(i,3)*l_sutm)+ S(3);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(3);
elseif busdataM(i,1)== 11 %bus 11
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(9);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(9);
elseif busdataM(i,1)== 17
    S(i) = (busdataM(i,3)*l_sutm) + S(13);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(13);
elseif busdataM(i,1)== 22
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(19);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(19);
elseif busdataM(i,1)== 26
    S(i) = (busdataM(i,3)*l_sutm) + S(12);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(12);
elseif busdataM(i,1)== 32
    S(i) = (busdataM(i,3)*l_sutm) + S(26);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(26);
elseif busdataM(i,1)== 38
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(35);
elseif busdataM(i,1)== 39
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(35);
elseif busdataM(i,1)== 54
    S(i) = (busdataM(i,3)*l_sutm) + S(51);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(51);
elseif busdataM(i,1)== 65
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(61);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(61);
```

```

elseif busdataM(i,1)== 67
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(43);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(43);
elseif busdataM(i,1)== 75
    S(i) = (busdataM(i,3)*l_sutm) + S(68);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(68);
elseif busdataM(i,1)== 95
    S(i) = (busdataM(i,3)*l_sutm) + S(83);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(83);
elseif busdataM(i,1)== 117
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(103);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(103);
elseif busdataM(i,1)== 21 && busdataM(i,1)== 23 && busdataM(i,1)== 42 &&
busdataM(i,1)== 74 && busdataM(i,1)== 87 && busdataM(i,1)== 97 &&
busdataM(i,1)== 99
    S(i) = (busdataM(i,3)*l_sutm) + S(i-2);
    U(i) = (busdataM(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdataM(i,1)== 11 && busdataM(i,1)== 29 && busdataM(i,1)== 53 &&
busdataM(i,1)== 89 && busdataM(i,1)== 106 && busdataM(i,1)== 115
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(i-2);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdataM(i,1)== 16 && busdataM(i,1)== 40 && busdataM(i,1)== 69 &&
busdataM(i,1)== 72 && busdataM(i,1)== 102
    S(i) = l_sakelar_pisah + (busdataM(i,3)*l_sutm) + S(i-1);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataM(i,3)*l_sutm*r_sutm)
+ U(i-1);
elseif busdataM(i,1)== 8 && busdataM(i,1)== 45
    S(i) = l_sakelar_pisah + (busdataM(i,3)*l_sutm) + S(i-2);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataM(i,3)*l_sutm*r_sutm)
+ U(i-2);
elseif busdataM(i,1)== 3 && busdataM(i,1)== 9 && busdataM(i,1)== 12 &&
busdataM(i,1)== 13 && busdataM(i,1)== 19 && busdataM(i,1)== 27 &&
busdataM(i,1)== 35 && busdataM(i,1)== 43 && busdataM(i,1)== 49 &&
busdataM(i,1)== 51 && busdataM(i,1)== 59 && busdataM(i,1)== 61 &&
busdataM(i,1)== 68 && busdataM(i,1)== 79 && busdataM(i,1)== 80 &&
busdataM(i,1)== 83 && busdataM(i,1)== 84 && busdataM(i,1)== 85 &&
busdataM(i,1)== 90 && busdataM(i,1)== 103 && busdataM(i,1)== 104 &&
busdataM(i,1)== 108 && busdataM(i,1)== 111 && busdataM(i,1)== 113
    S(i) = (busdataM(i,3)*l_sutm) + S(i-1);
    U(i) = (l_bus*r_bus)+ (busdataM(i,3)*l_sutm*r_sutm) + U(i-1);
elseif busdataM(i,1)== 57
    S(i) = l_trafo_distribusi + l_mikrohidro + (busdataM(i,3)*l_sutm)+ S(56);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi) +
(l_mikrohidro*r_mikrohidro) + (busdataM(i,3)*l_sutm*r_sutm) + U(56);
else
    S(i) = l_trafo_distribusi + (busdataM(i,3)*l_sutm) + S(i-1);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataM(i,3)*l_sutm*r_sutm)+ U(i-1);
end
dx = find(posisi==busdataM(i,1),1);
if ~isempty(dx)
    S(i) = S(i-1)+(busdataM(i,3)*l_sutm) + l_sakelar_pisah;
    U(i) = U(i-1)+(busdataM(i,3)*l_sutm*r_sutm)+
(l_sakelar_pisah*r_sakelar_pisah);

```

```

end
end

Saiddi = zeros(size(busdataM,1),1);
Saifii = zeros(size(busdataM,1),1);
fitL = zeros(size(busdataM,1),1);
jumpel = sum(busdataM(:,4));

for i=1:size(busdataM,1)
    Saifii(i) = (S(i)*busdataM(i,4))/jumpel;
    Saiddi(i) = (U(i)*busdataM(i,4))/jumpel;
    fitL(i) = 1/(Saifii(i)*Saiddi(i));
end
assignin('base','Saifii',Saifii);
assignin('base','Saiddi',Saiddi);
assignin('base','fitL',fitL);
assignin('base','Jumpel',jumpel);

%SAIFI total
saifitot = sum(Saifii);
assignin('base','saifitot',saifitot);

%SAIDI total
saiditot = sum(Saiddi);
assignin('base','saiditot',saiditot);

saidifi = saifitot*saiditot;
assignin('base','saidifi',saidifi);

%Nilai Fitness
fitness = 1/(saiditot*saifitot);
assignin('base','fitness',fitness);

```

Listing Program Kondisi Grid –Wind Turbine – Mikrohidro

```
function [saidifi,Saidi,saiditot,Saifi,saifitot,fitness,fitL] =
costDG(posisi)
% konstanta laju kegagalan
l_sutm=0.2; r_sutm=3;
l_sakelar_pisah=0.003; r_sakelar_pisah=10;
l_bus=0; r_bus=0;
l_trafo_distribusi=0.005; r_trafo_distribusi=10;
l_windturbine = 0.02; r_windturbine = 50;
l_mikrohidro = 0.032; r_mikrohidro = 200;

load busdataWM;
assignin('base','busdataWM',busdataWM);

S = zeros(size(busdataWM, 1), 1); %laju
U = zeros(size(busdataWM, 1), 1);
S(1) = 0.643;
U(1) = (S(1)*2);
S(2) = l_sakelar_pisah + (busdataWM(2,3)*l_sutm) + S(1);
U(2) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataWM(2,3)*l_sutm*r_sutm) +
U(1);
for i=3:size(busdataWM,1)
if busdataWM(i,1)== 6 %bus 6
    S(i) = (busdataWM(i,3)*l_sutm)+ S(3);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(3);
elseif busdataWM(i,1)== 11 %bus 11
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(9);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm) + U(9);
elseif busdataWM(i,1)== 17
    S(i) = (busdataWM(i,3)*l_sutm) + S(13);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(13);
elseif busdataWM(i,1)== 22
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(19);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm) + U(19);
elseif busdataWM(i,1)== 26
    S(i) = (busdataWM(i,3)*l_sutm) + S(12);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(12);
elseif busdataWM(i,1)== 32
    S(i) = (busdataWM(i,3)*l_sutm) + S(26);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(26);
elseif busdataWM(i,1)== 38
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm) + U(35);
elseif busdataWM(i,1)== 39
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(35);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm) + U(35);
elseif busdataWM(i,1)== 54
    S(i) = (busdataWM(i,3)*l_sutm) + S(51);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(51);
elseif busdataWM(i,1)== 65
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(61);
```

```

    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm) + U(61);
elseif busdataWM(i,1)== 67
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(43);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm) + U(43);
elseif busdataWM(i,1)== 75
    S(i) = (busdataWM(i,3)*l_sutm) + S(68);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(68);
elseif busdataWM(i,1)== 95
    S(i) = (busdataWM(i,3)*l_sutm) + S(83);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(83);
elseif busdataWM(i,1)== 21 && busdataWM(i,1)== 23 && busdataWM(i,1)== 42
&& busdataWM(i,1)== 74 && busdataWM(i,1)== 87 && busdataWM(i,1)== 97 &&
busdataWM(i,1)== 99
    S(i) = (busdataWM(i,3)*l_sutm) + S(i-2);
    U(i) = (busdataWM(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdataWM(i,1)== 11 && busdataWM(i,1)== 29 && busdataWM(i,1)== 53 &&
busdataWM(i,1)== 89 && busdataWM(i,1)== 106 && busdataWM(i,1)== 115
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(i-2);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm) + U(i-2);
elseif busdataWM(i,1)== 16 && busdataWM(i,1)== 40 && busdataWM(i,1)== 69 &&
busdataWM(i,1)== 72 && busdataWM(i,1)== 102
    S(i) = l_sakelar_pisah + (busdataWM(i,3)*l_sutm) + S(i-1);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataWM(i,3)*l_sutm*r_sutm)
+ U(i-1);
elseif busdataWM(i,1)== 8 && busdataWM(i,1)== 45
    S(i) = l_sakelar_pisah + (busdataWM(i,3)*l_sutm) + S(i-2);
    U(i) = (l_sakelar_pisah*r_sakelar_pisah) + (busdataWM(i,3)*l_sutm*r_sutm)
+ U(i-2);
elseif busdataWM(i,1)== 3 && busdataWM(i,1)== 9 && busdataWM(i,1)== 12 &&
busdataWM(i,1)== 13 && busdataWM(i,1)== 19 && busdataWM(i,1)== 27 &&
busdataWM(i,1)== 35 && busdataWM(i,1)== 43 && busdataWM(i,1)== 49 &&
busdataWM(i,1)== 51 && busdataWM(i,1)== 59 && busdataWM(i,1)== 61 &&
busdataWM(i,1)== 68 && busdataWM(i,1)== 79 && busdataWM(i,1)== 80 &&
busdataWM(i,1)== 83 && busdataWM(i,1)== 84 && busdataWM(i,1)== 85 &&
busdataWM(i,1)== 90 && busdataWM(i,1)== 103 && busdataWM(i,1)== 104 &&
busdataWM(i,1)== 108 && busdataWM(i,1)== 111 && busdataWM(i,1)== 113
    S(i) = (busdataWM(i,3)*l_sutm) + S(i-1);
    U(i) = (l_bus*r_bus)+ (busdataWM(i,3)*l_sutm*r_sutm) + U(i-1);
elseif busdataWM(i,1)== 57
    S(i) = l_trafo_distribusi + l_mikrohidro + (busdataWM(i,3)*l_sutm)+ S(56);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi) +
(l_mikrohidro*r_mikrohidro) +(busdataWM(i,3)*l_sutm*r_sutm) + U(56);
elseif busdataWM(i,1)== 117
    S(i) = l_trafo_distribusi + l_windturbine + (busdataWM(i,3)*l_sutm)+
S(103);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi) +
(l_windturbine*r_windturbine) +(busdataWM(i,3)*l_sutm*r_sutm) + U(103);
else
    S(i) = l_trafo_distribusi + (busdataWM(i,3)*l_sutm) + S(i-1);
    U(i) = (l_trafo_distribusi*r_trafo_distribusi)+
(busdataWM(i,3)*l_sutm*r_sutm)+ U(i-1);
end
dx = find(posisi==busdataWM(i,1),1);
if ~isempty(dx)

```

```

        S(i) = S(i-1)+(busdataWM(i,3)*l_sutm) + l_sakelar_pisah;
        U(i) = U(i-1)+(busdataWM(i,3)*l_sutm*r_sutm)+
(l_sakelar_pisah*r_sakelar_pisah);
    end
end

Saidi = zeros(size(busdataWM,1),1);
Saifi = zeros(size(busdataWM,1),1);
fitL = zeros(size(busdataWM,1),1);
jumpel = sum(busdataWM(:,4));

for i=1:size(busdataWM,1)
    Saifi(i) = (S(i)*busdataWM(i,4))/jumpel;
    Saidi(i) = (U(i)*busdataWM(i,4))/jumpel;
    fitL(i) = 1/(Saifi(i)*Saidi(i));
end

assignin('base','Saifi',Saifi);
assignin('base','Saidi',Saidi);
assignin('base','Jumpel',jumpel);

%SAIFI total
saifitot = sum(Saifi);
assignin('base','saifitot',saifitot);

%SAIDI total
saiditot = sum(Saidi);
assignin('base','saiditot',saiditot);

saidifi = saifitot*saiditot;
assignin('base','saidifi',saidifi);

%Nilai Fitness
fitness = 1/(saiditot*saifitot);
assignin('base','fitness',fitness);

```


Listing Program Metode SA

```
_function [saidifi,solusi,bestsaidifi,saidi,saifi,fitness]=sa

fobj=@(x) costDG(x); % fungsi objek
nVar=16; %Variabel jumlah sectionalizer yg dicari (8,9,12,16)

%% Parameter SA
T_init= 100; % Initial temperature
T_min = 1e-20; % Final stopping temperature
F_min = 1e-6; % Min value of the function
max_rej=100; % Maximum number of rejections / iterasi
max_run=300; % Maximum number of runs
max_accept=10; % Maximum number of accept
k = 1; % Boltzmann constant
alpha=0.95; % Cooling factor
Enorm=1e-2; % Enorm (Variasi dengan 1e-2 dan 1e-3)

Lb=2*ones(1,nVar); %batas bawah
Ub=118*ones(1,nVar); %batas atas
guess=Lb+(Ub-Lb).*rand(size(Lb)); % Initial guess dikalikan nilai random
guess=simplebounds(guess,Lb,Ub);

% Inisialisasi
i= 0; j = 0; accept = 0; totaleval = 0;
% Inisialisasi suhu dan energi
T =(T_init);
E_init = fobj(round(guess));
E_old = E_init;
E_new = E_old;

[Saidifi,Saidi,saiditot,Saifi,saifitot,fitness,fitL] =
costDG(round(guess)); %memanggil fungsi objek
sd_new=saiditot; %nilai saiditotal = nilai sd baru
sd_old=sd_new; %nilai sd baru = nilai sd lama
sf_new=saifitot; %nilai saifitotal = nilai sf baru
sf_old=sf_new; %nilai sf baru = nilai sf lama
ft_new=fitness; %nilai fitness = nilai ft baru
ft_old=ft_new ; %ft baru = ft lama

best=guess; % inisialisasi guess
sd_it=[];
sf_it=[];
ft_it=[]; %inisialisasi variable

% Mulai Simulated Annealing
E_it=[];
while ((T > T_min) && (j <= max_rej) && (E_new>F_min)) %batasan optimasi
    i = i+1;
    % Check jika nilai max_run dan accept memenuhi
    if (i >= max_run) || (accept >= max_accept) %kondisi if jika tidak
    terpenuhi maka akan skip
        % cooling factor
        T = alpha*T;
        totaleval = totaleval + i;
        % reset counters
```

```

        i = 1; accept = 1; %penanda iterasi
    end
    % Function evaluations at new locations
    s=0.1*(Ub-Lb);
    ns=best+s.*randn(1,nVar);
    ns=simplebounds(ns,Lb,Ub);
    E_new = fobj(round(ns));
    [Saidifi,Saidi,saiditot,Saifi,saifitot,fitness,fitL] =
costDG(round(ns)); %untuk menghasilkan nilai fitness dan saifi tiap iterasi
    sd_new=saiditot;
    sf_new=saifitot;
    ft_new=fitness;
    % Decide to accept the new solution
    DeltaE=E_new-E_old;
    % Accept if improved
    if (-DeltaE > Enorm)
        best = ns; E_old = E_new;
        sd_old=sd_new;
        sf_old=sf_new;
        ft_old=ft_new;
        accept=accept+1; j = 0;
    end
    % Accept with a small probability if not improved
    if (DeltaE<=Enorm && exp(-DeltaE/(k*T))>rand )
        best = ns; E_old = E_new;
        sd_old=sd_new;
        sf_old=sf_new;
        ft_old=ft_new;
        accept=accept+1;
    else
        j=j+1;
    end
    % Update the estimated optimal solution
    f_opt=E_old;
    sd_opt=sd_old;
    sf_opt=sf_old;
    ft_opt=ft_old;

    sd_it = [sd_it, sd_old];
    sf_it = [sf_it, sf_old]; %untuk menampilkan data grafik
    ft_it = [ft_it, ft_old];
    E_it = [E_it, E_old];
end

% Menampilkan hasil
disp(strcat('Final Temp. :', num2str(T)));
disp(strcat('Total Reject :', num2str(j)));
disp(strcat('Evaluations :', num2str(totaleval)));
disp(strcat('Best solution:', num2str(best))); %Menampilkan solusi
lokasi
disp(strcat('Best saidi:', num2str(sd_opt))); %Menampilkan nilai
best saidi
disp(strcat('Best saifi:', num2str(sf_opt))); %Menampilkan nilai
best saifi
disp(strcat('Best fitness:', num2str(ft_opt))); %Menampilkan nilai
best fitness

```

```

%Menampilkan nilai
assignin('base','Solusi',best);
assignin('base','Best_Saifi',sf_opt);
assignin('base','Best_Saidi',sd_opt);

% plot(E_it);
saidifi=E_it;
fitness=ft_it; %output dari fungsi
saidi=sd_it;
saifi=sf_it;
solusi=round(best);
bestsaidifi=f_opt;

assignin('base','E_it',E_it);
assignin('base','sd_it',sd_it); %tabel iterasi
assignin('base','sf_it',sf_it);
assignin('base','ft_it',ft_it);

%plot grafik Best SAIDI, Best SAIFI, Best Fitness
figure;
plot(sd_it,'-r');
xlabel('iteration');
ylabel('SAIDI (hour/year)');
title('Result of Best SAIDI');

figure;
plot(saifi,'-g');
xlabel('iteration');
ylabel('SAIFI (failure/year)');
title('Result of Best SAIFI');

figure;
plot(fitness,'-b');
xlabel('iteration');
ylabel('Fitness');
title('Result of Best Fitness');

% Application of simple constraints
function s=simplebounds(s,Lb,Ub)
% Apply the lower bound
ns_tmp=s;
I=ns_tmp<Lb; %pemberian batasan bus 2-118
ns_tmp(I)=Lb(I);

% Apply the upper bounds
J=ns_tmp>Ub;
ns_tmp(J)=Ub(J);
% Update this new move
s=ns_tmp;

```

Listing Program Metode ACO

```
function [saidifi,solusi,bestsaidifi]=aco(param)

%% Problem Definition

fobj=@(x) costDG(x); % fungsi biaya
nVar=8; % jml sectionalizer yang ditentukan (8,9,12,16)

%% ACO Parameters

MaxIt=100; % max iterasi
nPop=40; % jml Semut
alpha=1; % alpha
rho=0.3; % Laju penguapan rho (Variasi dengan 0,1 dan 0,3)

Lb=(2).*ones(1,nVar); % lower bound
Ub=(118).*ones(1,nVar); % upper bound
guess=Lb+(Ub-Lb).*rand(size(Lb));
guess=simplebounds(guess,Lb,Ub);

n_node=118; % number of nodes for each param

% template = [min(min(Lb)):max(max(Ub))];
% n_node = size(template, 1);
cost_best_prev=inf;
% tambahan
cost_saidi_prev=inf;
cost_saifi_prev=inf;
cost_fitness_prev=inf;

% simpan sbg best cost saat ini
BestCost=zeros(MaxIt,1);
% tambahan
BestSaidi=zeros(MaxIt,1);
BestSaifi=zeros(MaxIt,1);
BestFitness=zeros(MaxIt,1);

% untuk menyimpan best solution per iterasi
BestSol=zeros(MaxIt, nVar);

%% Generating Nodes
T=ones(n_node,nVar); %.*eps; % Pheromone Matrix
dT=zeros(n_node,nVar); % Change of Phormone

Nodes = zeros(n_node, nVar); % untuk nyimpan node
prob = zeros(n_node, nVar); % untuk nyimpan transition probabilities
for i=1:nVar
    Nodes(:,i) =linspace(Lb(i),Ub(i),n_node); % Node generation at equal
    spaced points
end
% for i=1:nVar
% Nodes(:, i) = template;
% end
```

```

%% Iteration loop
ant=zeros(nPop, nVar);
cost=zeros(nPop, 1);

semuaSemut = zeros(MaxIt, nPop, nVar);

%timeStart = tic;
for iter=1:MaxIt
    % build matrix transition probabilities
    for tour_i=1:nVar
        prob(:,tour_i)= (T(:,tour_i).^alpha) ./ (sum(T(:,tour_i)));
    end

    for A=1:nPop
        % pilih node pakai transition probabilities yg tadi
        for tour_i=1:nVar
            node_sel=rand
            node_ind=1;
            prob_sum=0;
            for j=1:n_node
                prob_sum=prob_sum+prob(j,tour_i)
                if prob_sum>=node_sel
                    node_ind=j;
                    break
                end
            end
            % isi semut
            ant(A,tour_i)=node_ind;
        end
        guess = zeros(1, nVar);
        for varNum=1:nVar
            guess(varNum) = Nodes(ant(A, varNum), varNum);
        end
        guess=simplebounds(guess,Lb,Ub);
        semuaSemut(iter, A, :) =(guess);

        % hitung cost semut ini
        cost(A)=fobj(round(guess));
        % tambahan

[saidifi,Saidi,saiditot,Saifi,saifitot,fitness,fitL]=costDG(round(guess));
        fit=fitness;
        sd_new=saiditot;
        sf_new=saifitot;
    end
    % cari semut terbaik
    [cost_best,cost_best_ind]=min(cost);
    % tambahan
    cost_saidi=min(sd_new);
    cost_saifi=min(sf_new);
    cost_fitness=max(fit);

    % Elitism
    if ((cost_best>cost_best_prev) && (iter>1))
        [cost_worst,cost_worst_ind]=max(cost);
    end
end

```

```

    ant(cost_worst_ind,:)=best_prev_ant;
    cost_best=cost_best_prev;
    cost_best_ind=cost_worst_ind;
    % tambahan
    cost_saidi=cost_saidi_prev;
    cost_saifi=cost_saifi_prev;
    cost_fitness=cost_fitness_prev;

else
    cost_best_prev=cost_best;
    best_prev_ant=ant(cost_best_ind,:);
    %tambahan
    cost_saidi_prev=cost_saidi;
    cost_saifi_prev=cost_saifi;
    cost_fitness_prev=cost_fitness;

end

% simpan cost terbaik untuk iterasi saat ini -- untuk grafik
BestCost(iter) = cost_best;
% tambahan
BestSaidi(iter) = cost_saidi;
BestSaifi(iter) = cost_saifi;
BestFitness(iter) = cost_fitness;

guess = zeros(1, nVar);
for varNum=1:nVar
    guess(varNum) = Nodes(ant(cost_best_ind, varNum), varNum);
end
s=0.1*(Ub-Lb);
%ns=best+s.*randn(1,nVar);
guess=simplebounds(guess,Lb,Ub);
% simpan solusi terbaik untuk iterasi saat ini
BestSol(iter,:) =(guess);

% tampilkan di command window
disp(strcat('Iteration : ',num2str(iter)));
disp(strcat('Best Param : ',num2str(BestSol(iter,:))));
disp(strcat('Best Saidi : ', num2str(sd_new)));
disp(strcat('Best Saifi : ', num2str(sf_new)));
disp(['Best Fitness : ' num2str(BestFitness(iter))]);
%disp(['Best Cost : ' num2str(BestCost(iter))]);

assignin('base','Best_Saidi',sd_new);
assignin('base','Best_Saifi',sf_new);
assignin('base','Best_Fitness',fit);
assignin('base','Best_Solution',BestSol(iter,:));

dT=zeros(n_node,nVar); % hitung perubahan Pheromone
for tour_i=1:nVar
    dT(ant(cost_best_ind,tour_i),tour_i)= 2*sum(cost(:) ==
min(cost))*min(cost)/max(cost)
end

% update feromone
T= (1-rho).*T
T= T + dT

```

```

end

%% Grafik-grafik hasil
figure;
plot(BestSaidi, '-r');
xlabel('Iteration');
ylabel('SAIDI (hour/year)');
title('Result of Best SAIDI');

figure;
plot(BestSaifi, '-g');
xlabel('Iteration');
ylabel('SAIFI (failure/year)');
title('Result of Best SAIFI');

figure;
plot(BestFitness, '-b');
xlabel('Iteration');
ylabel('Fitness');
title('Result of Best Fitness');

saidifi=BestCost;
saidi=BestSaidi;
saifi=BestSaifi;
fitness=BestFitness;

solusi=round(BestSol(iter, :));
bestsaidifi=BestCost(iter);

% Application of simple constraints
function s=simplebounds(s,Lb,Ub)
% Apply the lower bound
ns_tmp=s;
I=ns_tmp<Lb;
ns_tmp(I)=Lb(I);

% Apply the upper bounds
J=ns_tmp>Ub;
ns_tmp(J)=Ub(J);
% Update this new move
s=ns_tmp;

```

LAMPIRAN 2
WAKTU PROGRAM DAN
SPEKIFIKASI LAPTOP

Waktu *Running* Metode SA

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
sa	1	12.518 s	5.303 s	
newplot	3	4.195 s	3.490 s	
costDG	366	1.206 s	1.206 s	
xlabel	3	1.036 s	1.011 s	
sa>@(x)costDG(x)	183	0.697 s	0.022 s	
newplot>ObserveAxesNextPlot	3	0.667 s	0.088 s	
cla	3	0.579 s	0.003 s	
graphics\private\clo	3	0.483 s	0.483 s	
num2str	7	0.348 s	0.226 s	
ylabel	3	0.130 s	0.130 s	
num2str>handleNumericPrecision	4	0.119 s	0.045 s	
strcat	7	0.096 s	0.094 s	
graphics\private\claNotify	3	0.093 s	0.093 s	
ishold	3	0.075 s	0.075 s	
num2str>convertUsingRecycledSprintf	4	0.074 s	0.074 s	
title	3	0.063 s	0.062 s	
sa>simplebounds	183	0.042 s	0.042 s	
newplot>ObserveFigureNextPlot	3	0.038 s	0.038 s	
graph2d\private\labelcheck	9	0.026 s	0.026 s	

Waktu Running Metode ACO

<u>Function Name</u>	<u>Calls</u>	<u>Total Time</u>	<u>Self Time*</u>	Total Time Plot (dark band = self time)
aco	1	172.568 s	151.800 s	
costDG	8000	19.903 s	19.903 s	
aco>@(x)costDG(x)	4000	10.556 s	0.191 s	
aco>simplebounds	4101	0.258 s	0.258 s	
num2str	500	0.152 s	0.066 s	
strcat	400	0.115 s	0.109 s	
num2str>handleNumericPrecision	400	0.079 s	0.011 s	
num2str>convertUsingRecycledSprintf	400	0.068 s	0.068 s	
xlabel	3	0.049 s	0.045 s	
newplot	3	0.042 s	0.030 s	
linspace	8	0.024 s	0.024 s	
title	3	0.016 s	0.016 s	
ylabel	3	0.014 s	0.014 s	
newplot>ObserveAxesNextPlot	3	0.012 s	0.002 s	
cla	3	0.010 s	0.002 s	
graphics\private\clo	3	0.008 s	0.008 s	
int2str	100	0.007 s	0.007 s	



View basic information about your computer

Windows edition

Windows 8.1 Single Language

© 2013 Microsoft Corporation. All rights reserved.

[Get more features with a new edition of Windows](#)



System

Manufacturer: Asus

Processor: Intel(R) Core(TM) i3-4030U CPU @ 1.90GHz 1.90 GHz

Installed memory (RAM): 4,00 GB (3,89 GB usable)

System type: 64-bit Operating System, x64-based processor

Pen and Touch: Full Windows Touch Support with 10 Touch Points



Asus support

Phone number: 1-888-678-3688

Support hours: Mon-Fri: 9:00am-6:00pm

Website: [Online support](#)

Computer name, domain and workgroup settings

Computer name: user

Full computer name: user

Computer description:

Workgroup: WORKGROUP



Windows activation

Windows is activated [Read the Microsoft Software Licence Terms](#)

Product ID: 00261-20000-00000-AA168

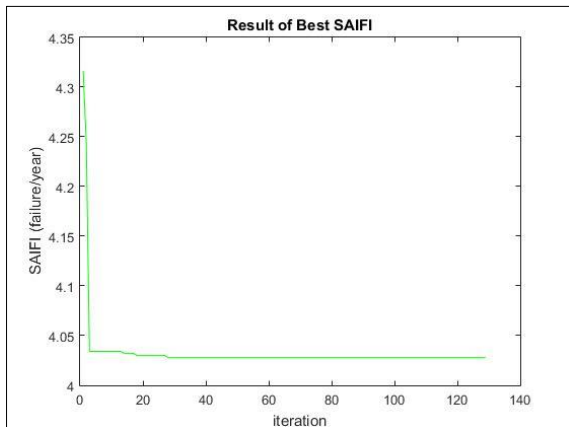
[Change product key](#)

LAMPIRAN 3
LOKASI DAN GRAFIK
PERCOBAAN

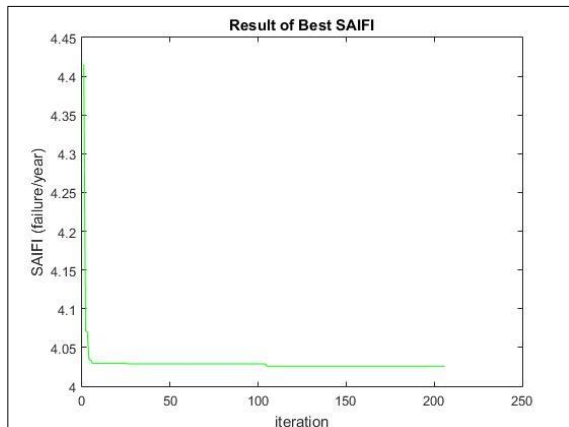
Skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan SA

percobaan	Best SAIFI (kali/tahun)	Best SAIDI (jam/tahun)	CAIDI (jam / tahun)	Lokasi Relokasi
Skenario Grid - Wind Turbine				
1	4,0279	12,2137	3,032275	9, 13, 20, 28, 33, 34, 35, 47, 86, 91, 94, 98, 102, 110, 118, 118
2	4,026	12,1949	3,029036	7, 7, 8, 10, 21, 34, 42, 43, 56, 74, 79, 86, 89, 91, 100, 118
3	4,0276	12,2115	3,031955	5, 9, 15, 24, 27, 33, 34, 35, 47, 57, 62, 96, 98, 100, 108, 118
4	4,0261	12,1957	3,02916	2, 8, 12, 40, 41, 45, 47, 48, 92, 98, 106, 114, 115, 118
5	4,0286	12,2208	3,03351	2, 3, 9, 20, 34, 78, 82, 84, 91, 98, 107, 118, 118, 118, 118, 118
6	4,0251	12,186	3,027502	2, 8, 9, 13, 35, 41, 42, 46, 50, 64, 80, 81, 81, 98, 116, 118
7	4,0274	12,2087	3,03141	12, 13, 36, 42, 43, 48, 50, 68, 78, 80, 102, 107, 110, 113, 115, 118
8	4,0247	12,1821	3,026834	2, 3, 5, 7, 12, 13, 33, 43, 46, 46, 50, 57, 58, 80, 84, 99
9	4,0253	12,2877	3,052617	2, 3, 20, 33, 34, 37, 40, 42, 45, 63, 79, 80, 81, 87, 90, 114
10	4,0284	12,2195	3,033338	2, 9, 43, 44, 46, 59, 61, 68, 74, 77, 80, 81, 81, 108, 111, 118

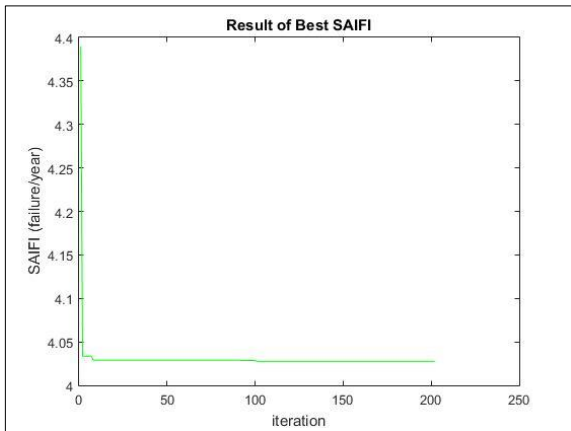
Grafik SAIFI skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan SA



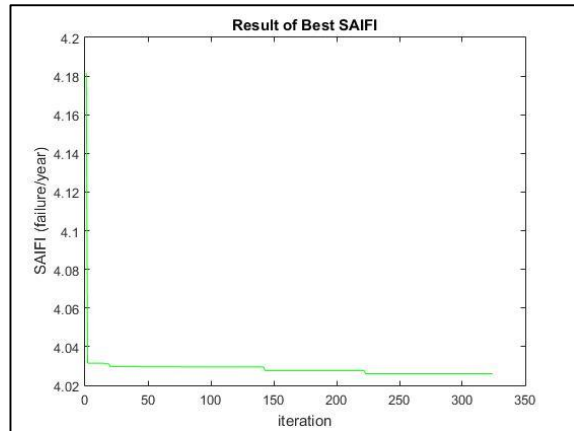
(Percobaan ke 1)



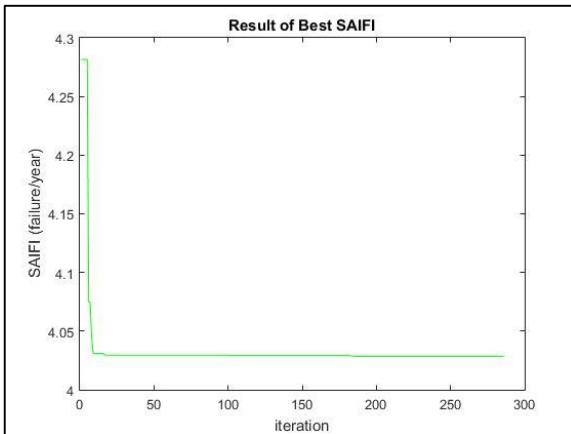
(Percobaan ke 2)



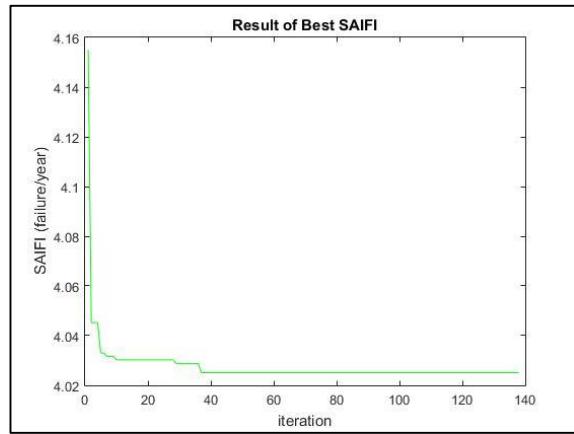
(Percobaan ke 3)



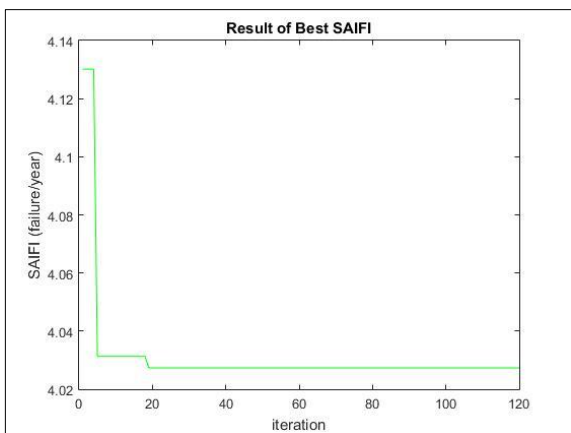
(Percobaan ke 4)



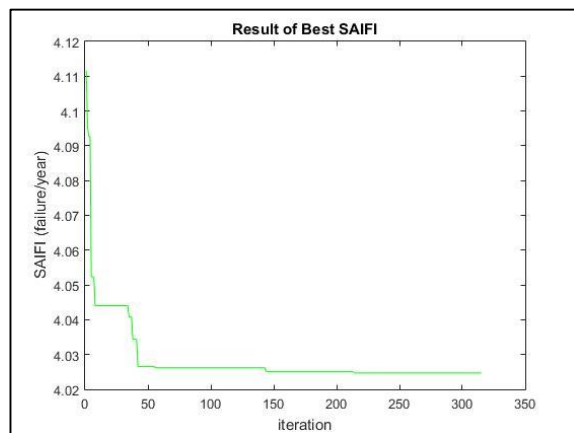
(Percobaan ke 5)



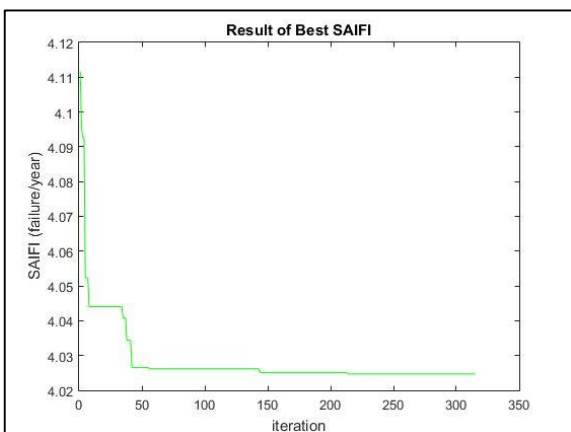
(Percobaan ke 6)



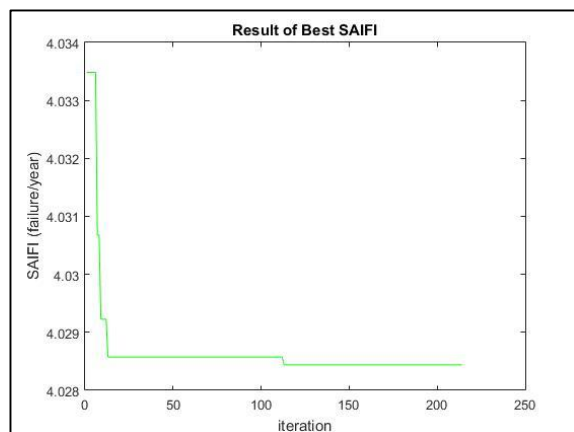
(Percobaan ke 7)



(Percobaan ke 8)

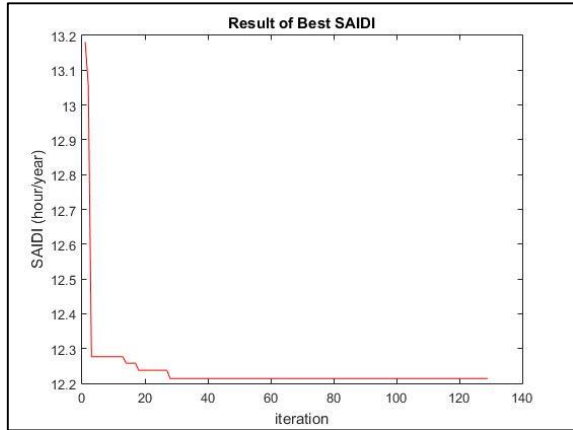


(Percobaan ke 9)

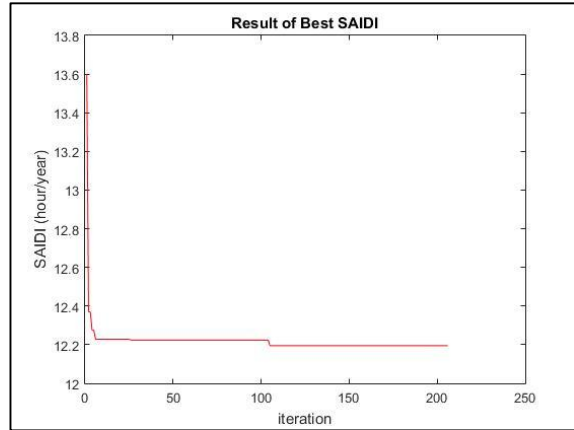


(Percobaan ke 10)

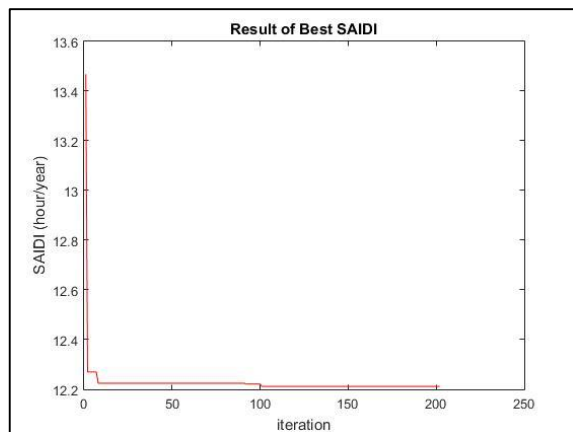
Grafik SAIDI skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan SA



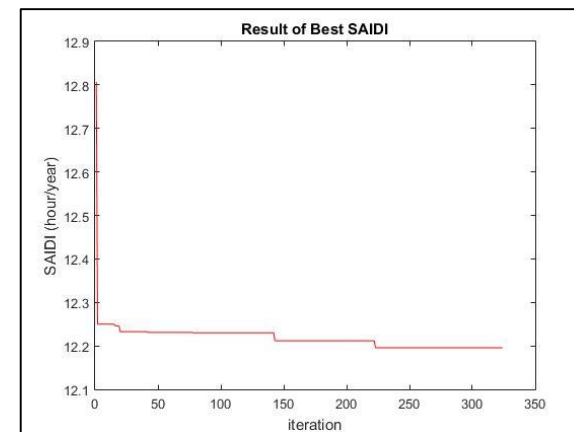
(Percobaan ke 1)



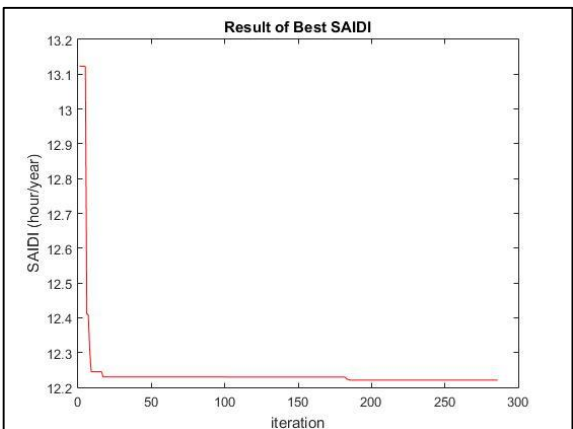
(Percobaan ke 2)



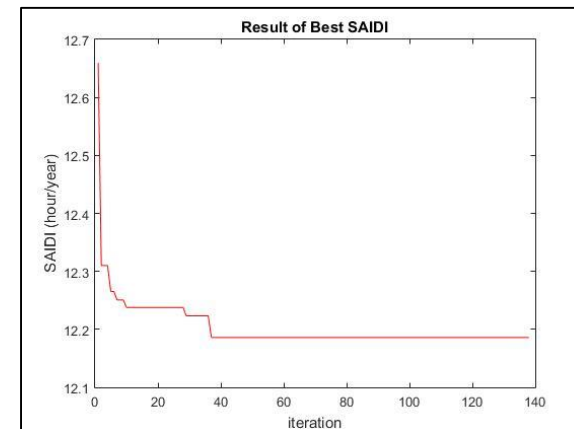
(Percobaan ke 3)



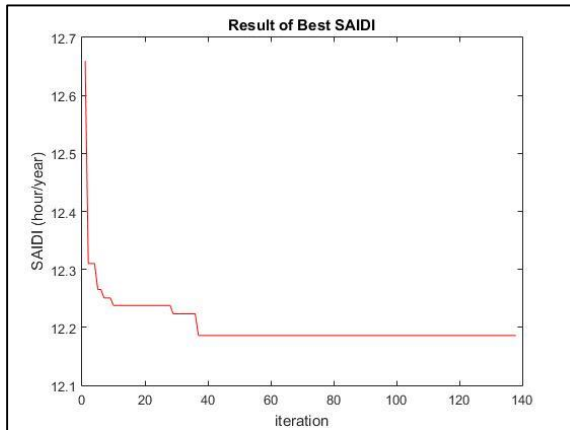
(Percobaan ke 4)



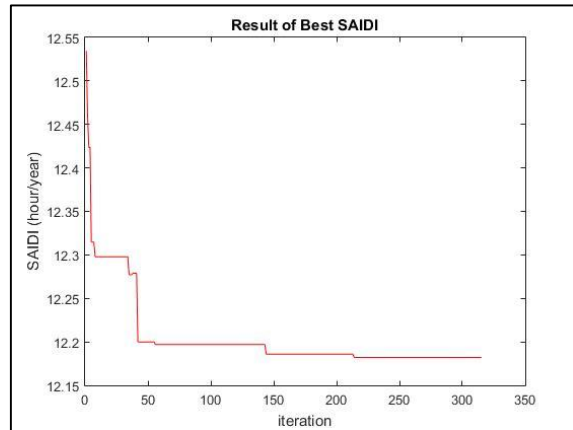
(Percobaan ke 5)



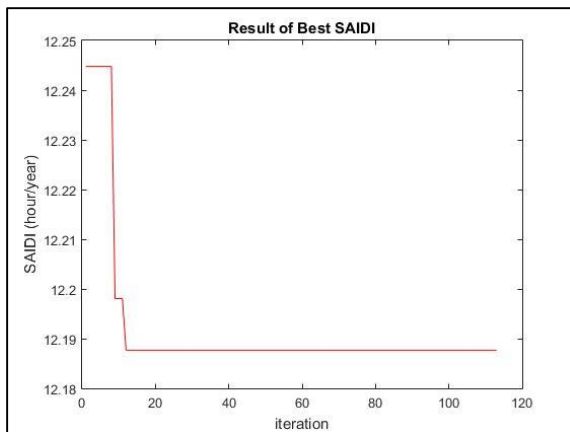
(Percobaan ke 6)



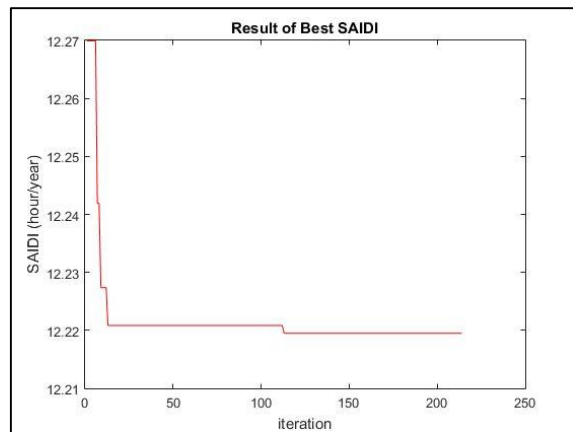
(Percobaan ke 7)



(Percobaan ke 8)

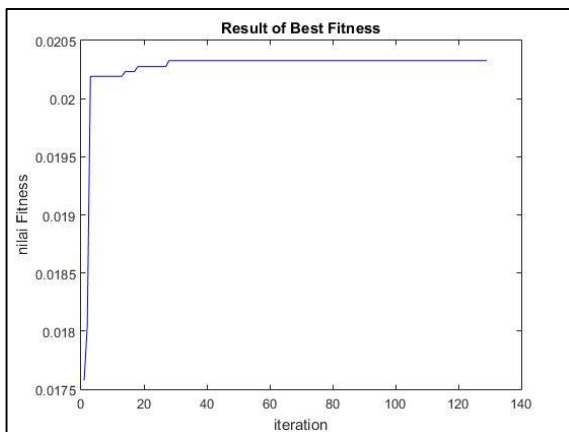


(Percobaan ke 9)

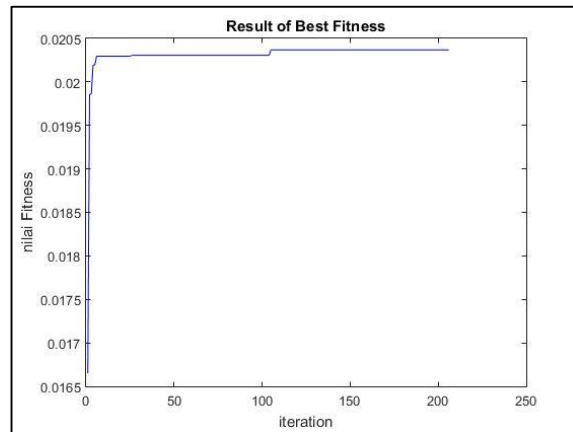


(Percobaan ke 10)

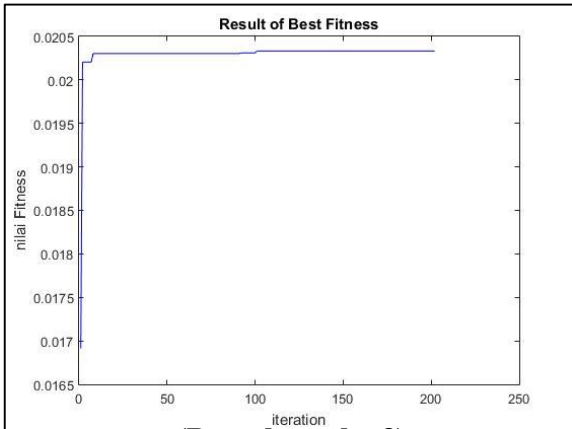
Grafik *fitness* skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan SA



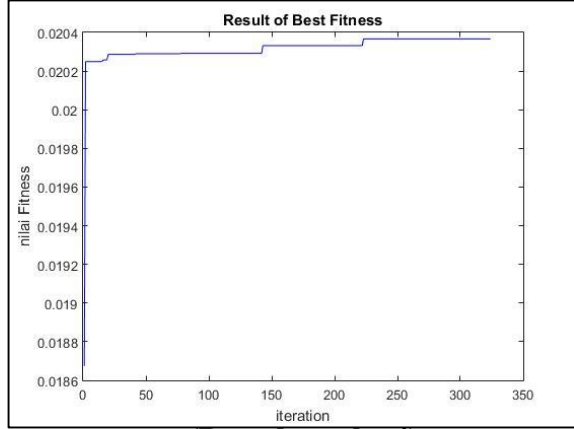
(Percobaan ke 1)



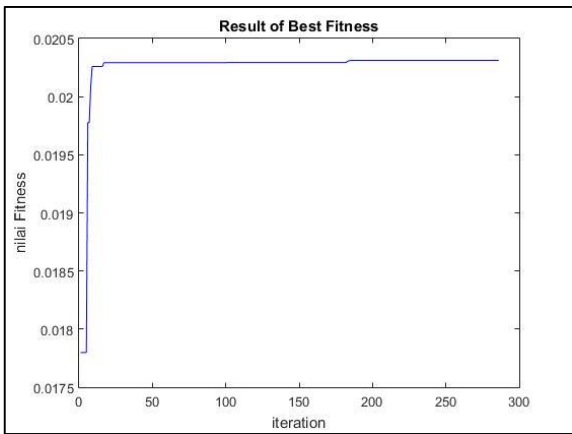
(Percobaan ke 2)



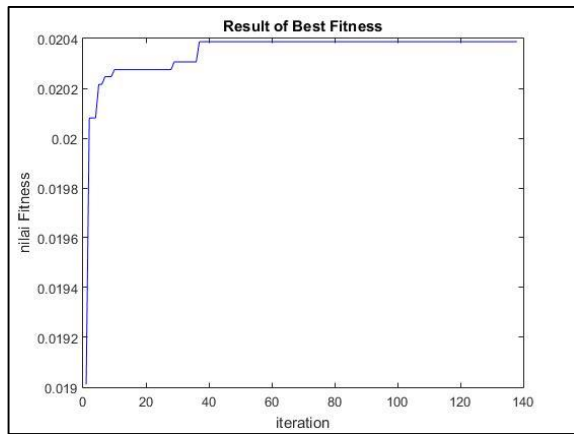
(Percobaan ke 3)



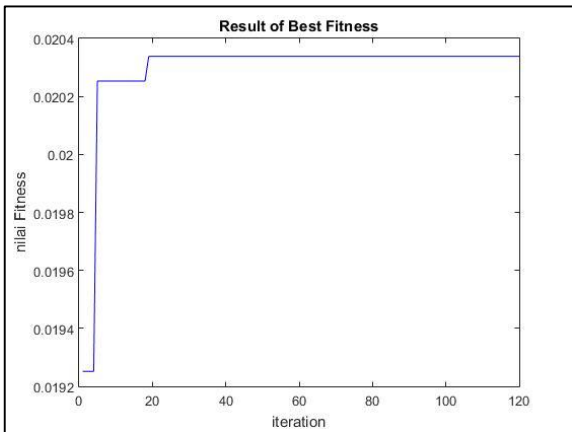
(Percobaan ke 4)



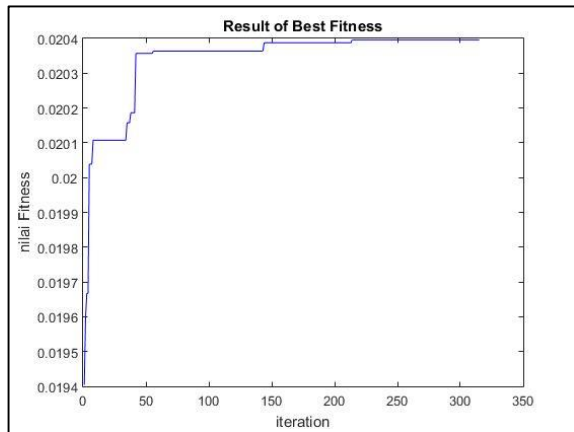
(Percobaan ke 5)



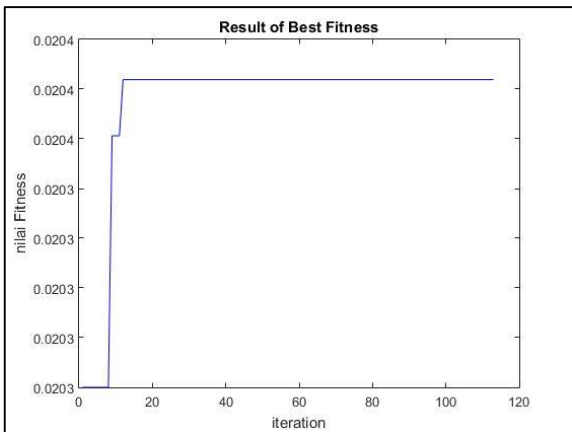
(Percobaan ke 6)



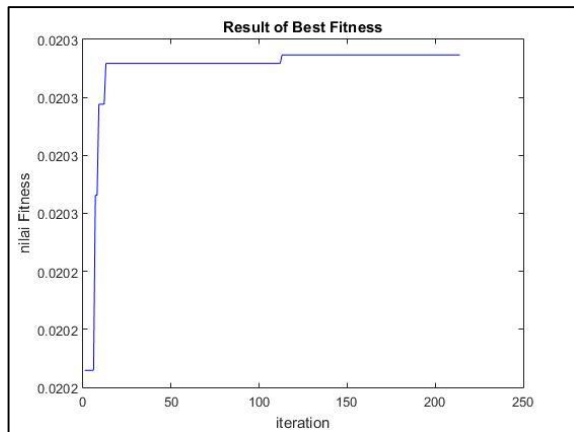
(Percobaan ke 7)



(Percobaan ke 8)



(Percobaan ke 9)

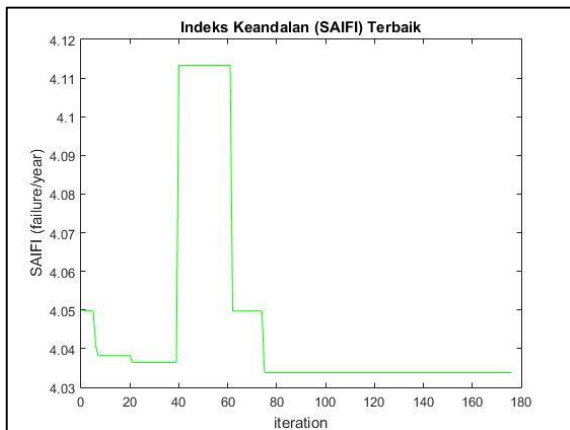


(Percobaan ke 10)

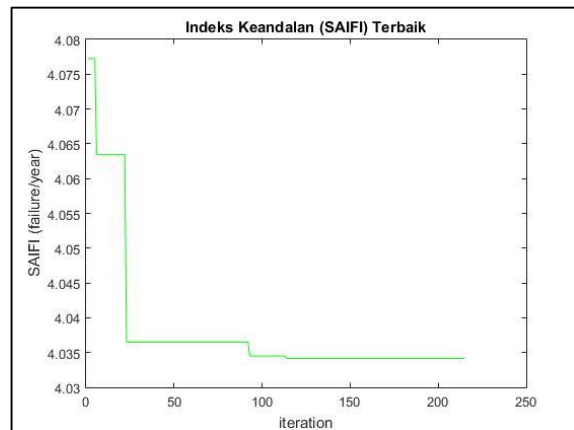
Skenario 2 lokasi sectionalizer dengan 10x percobaan menggunakan SA

percobaan	Best SAIFI (kali/tahun)	Best SAIDI (jam/tahun)	CAIDI (jam / tahun)	Lokasi Relokasi
Skenario Grid - Mikrohidro				
1	4,0338	12,2444	3,03545	2, 9, 12, 13, 13, 13, 43, 52, 53, 57, 71, 80, 96, 107, 110
2	4,0342	12,248	3,036042	9, 18, 25, 43, 44, 46, 47, 57, 64, 77, 99, 103, 103, 107, 118, 118
3	4,032	12,2264	3,032341	3, 23, 34, 42, 43, 57, 66, 71, 82, 83, 84, 88, 105, 114, 118, 118
4	4,0351	12,2579	3,037818	2, 10, 19, 35, 43, 56, 57, 70, 82, 84, 85, 86, 86, 98, 107, 118
5	4,0383	12,2894	3,043211	2, 2, 2, 2, 8, 13, 23, 23, 36, 57, 57, 64, 71, 89, 112, 112
6	4,0349	12,2557	3,037423	2, 4, 4, 21, 28, 31, 35, 36, 41, 42, 49, 57, 60, 63, 64, 77
7	4,0316	12,2227	3,031724	7, 9, 40, 41, 50, 57, 62, 63, 71, 76, 84, 87, 87, 90, 91, 108
8	4,0349	12,2554	3,037349	30, 33, 41, 44, 48, 57, 62, 71, 73, 78, 94, 97, 101, 103, 111, 118
9	4,0342	12,248	3,036042	2, 12, 21, 38, 41, 43, 44, 56, 57, 62, 85, 89, 94, 100, 109
10	4,0317	12,2236	3,031872	2, 3, 8, 12, 40, 57, 60, 69, 72, 77, 86, 86, 89, 108, 114, 118

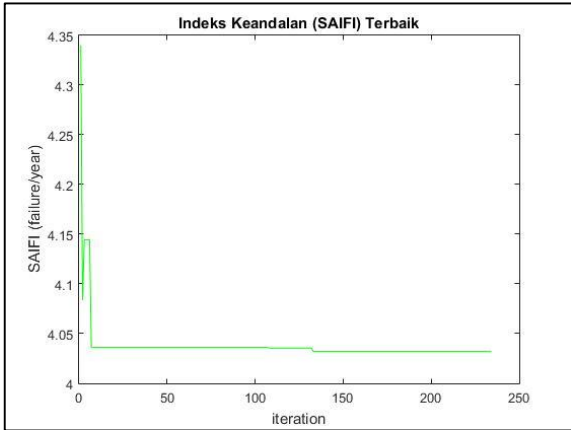
Grafik SAIFI skenario 2 lokasi sectionalizer dengan 10x percobaan menggunakan SA



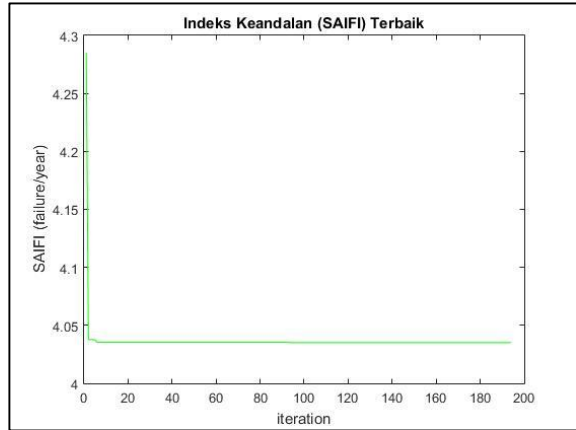
(Percobaan ke 1)



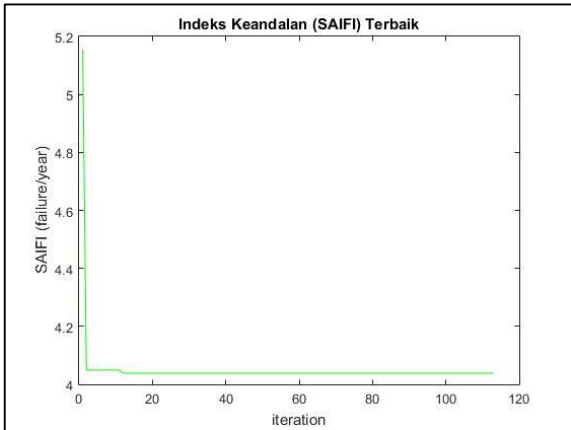
(Percobaan ke 2)



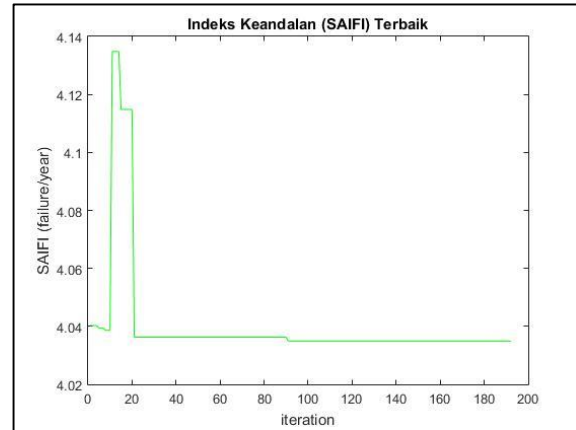
(Percobaan ke 3)



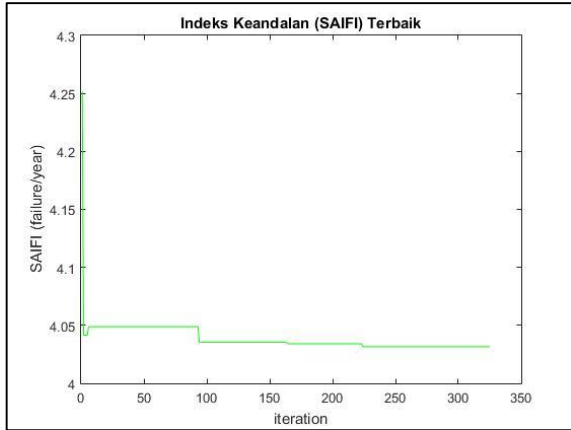
(Percobaan ke 4)



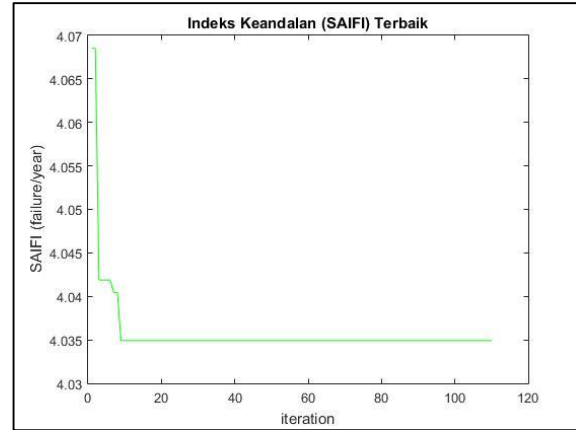
(Percobaan ke 5)



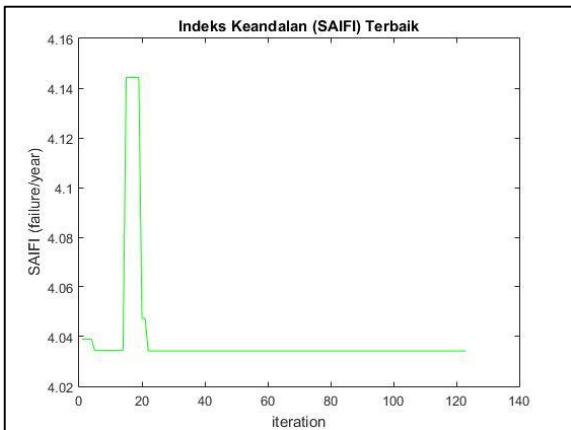
(Percobaan ke 6)



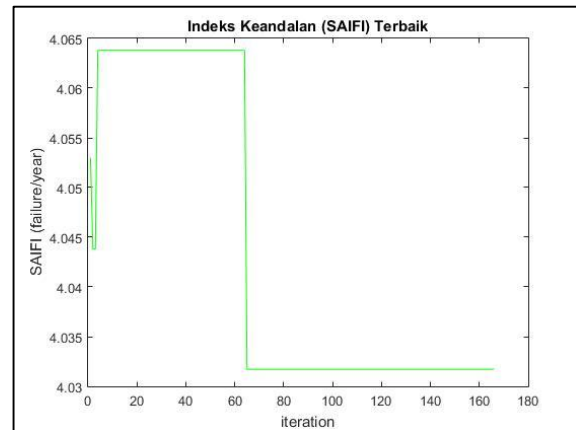
(Percobaan ke 7)



(Percobaan ke 8)

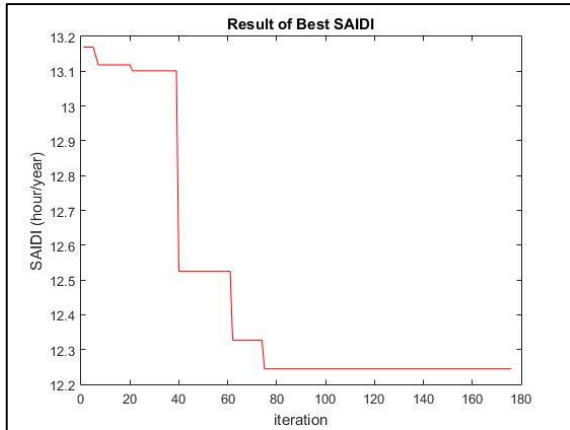


(Percobaan ke 9)

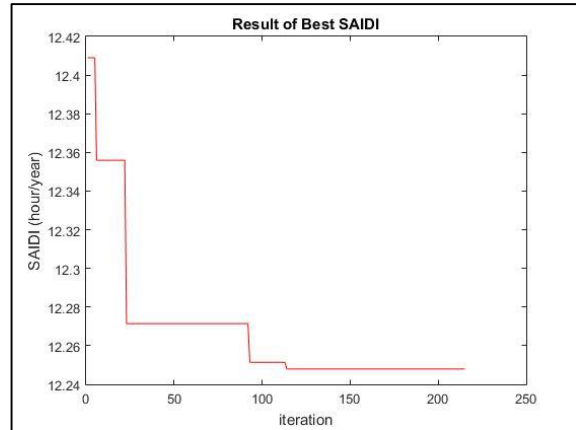


(Percobaan ke 10)

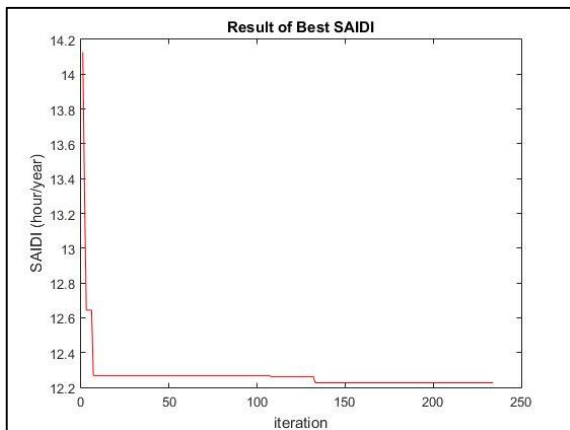
Grafik SAIDI skenario 2 lokasi sectionalizer dengan 10x percobaan menggunakan SA



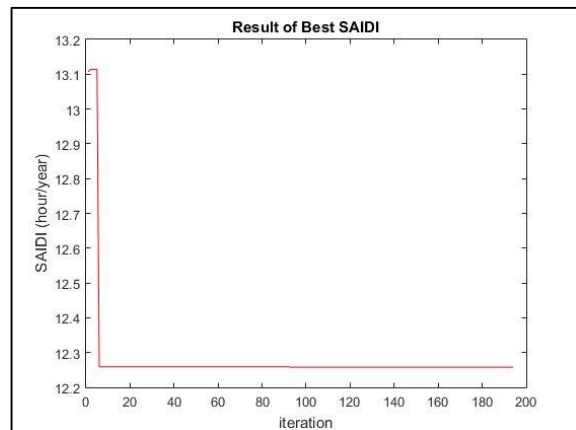
(Percobaan ke 1)



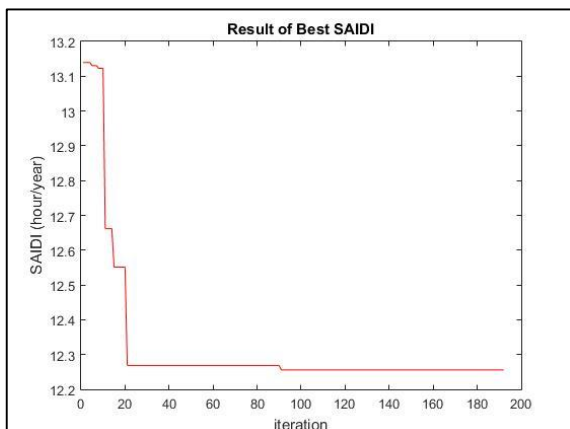
(Percobaan ke 2)



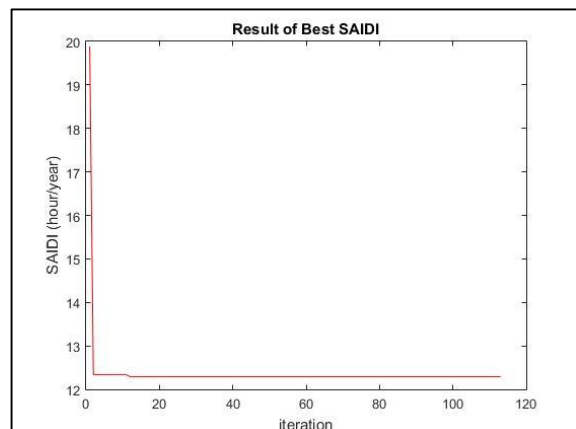
(Percobaan ke 3)



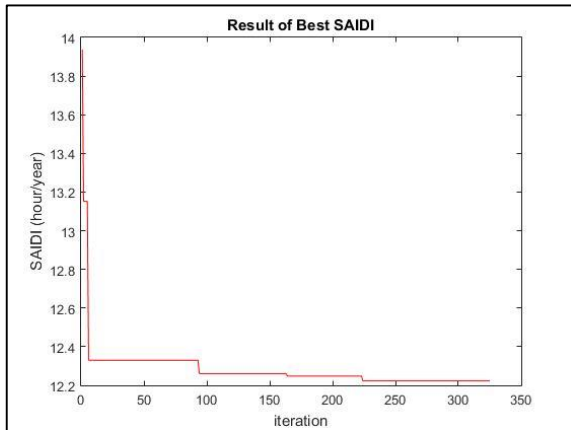
(Percobaan ke 4)



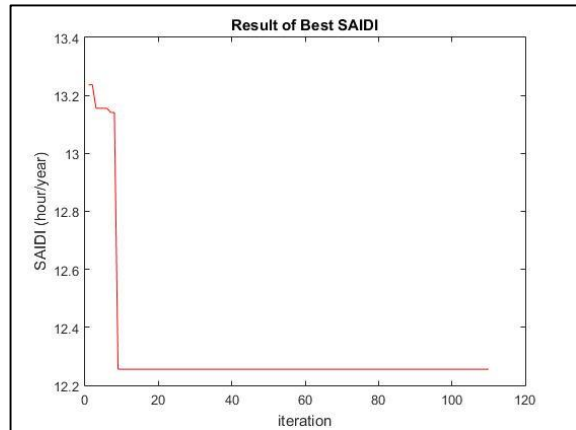
(Percobaan ke 5)



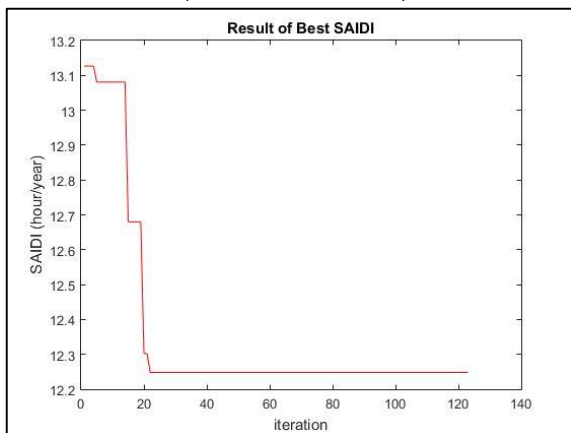
(Percobaan ke 6)



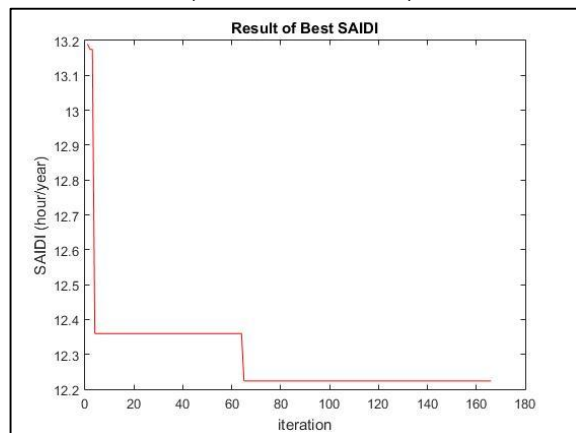
(Percobaan ke 7)



(Percobaan ke 8)

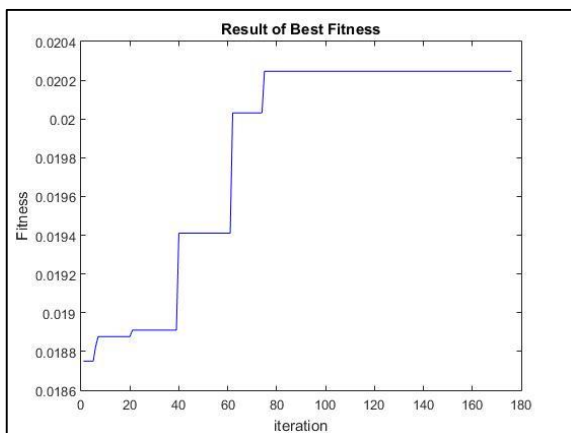


(Percobaan ke 9)

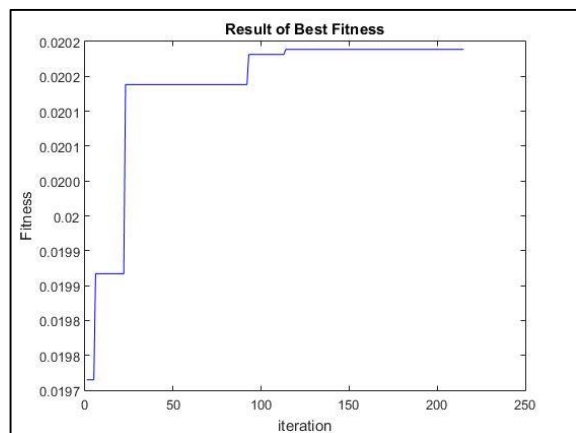


(Percobaan ke 10)

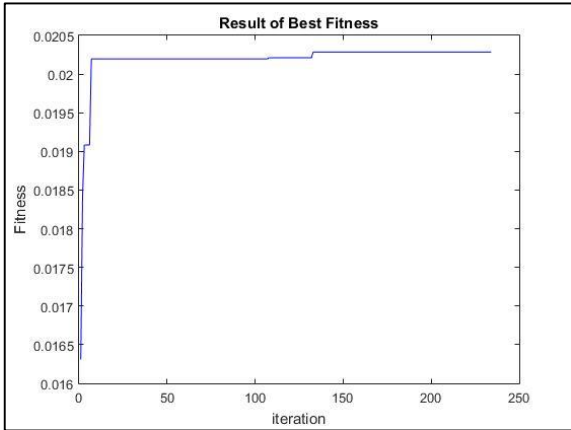
Grafik fitness skenario 2 lokasi sectionalizer dengan 10x percobaan menggunakan SA



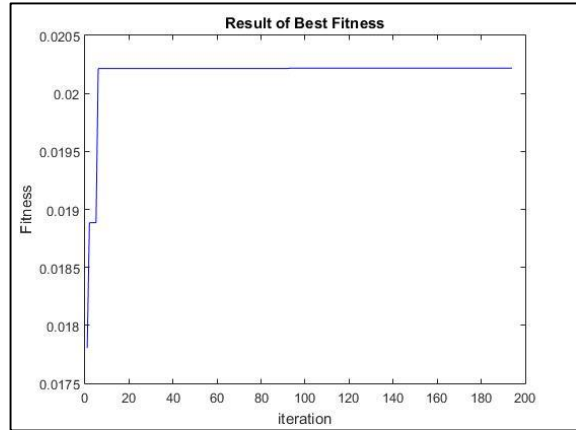
(Percobaan ke 1)



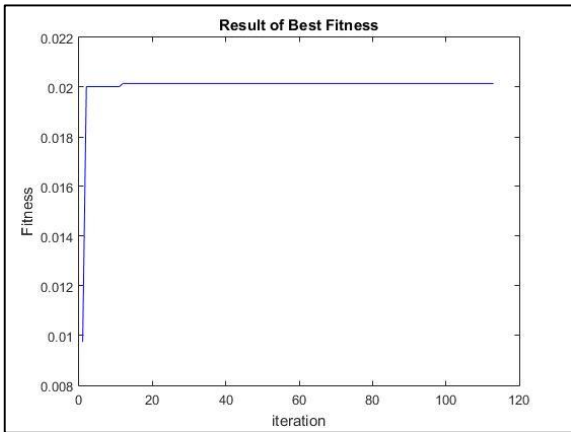
(Percobaan ke 2)



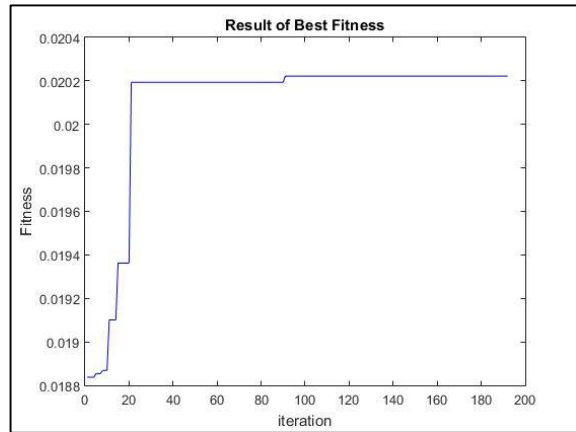
(Percobaan ke 3)



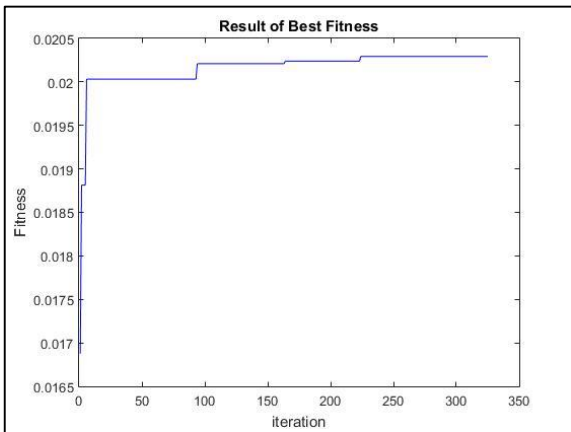
(Percobaan ke 4)



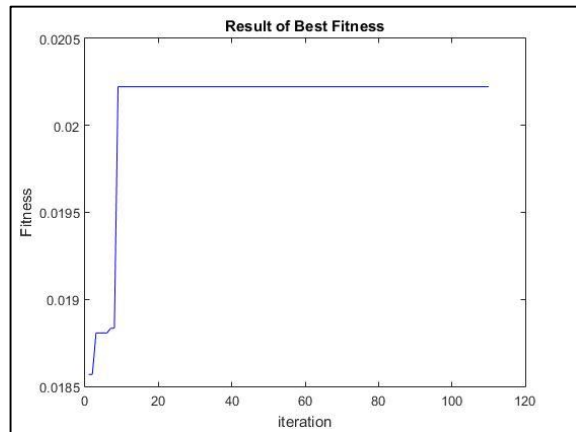
(Percobaan ke 5)



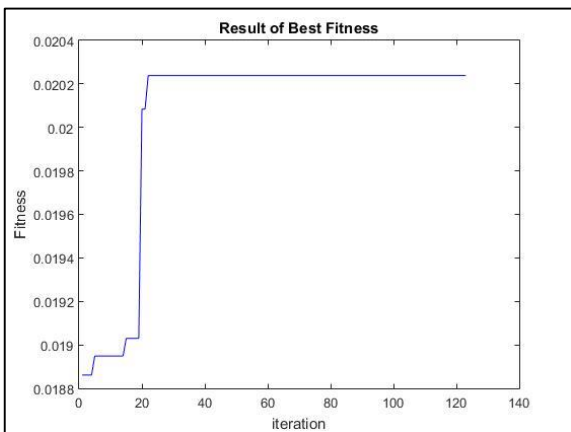
(Percobaan ke 6)



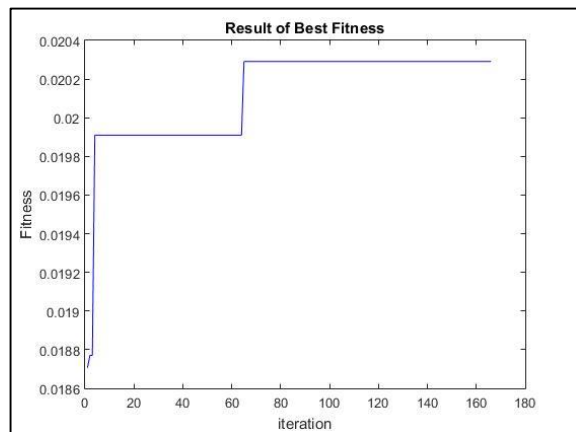
(Percobaan ke 7)



(Percobaan ke 8)



(Percobaan ke 9)

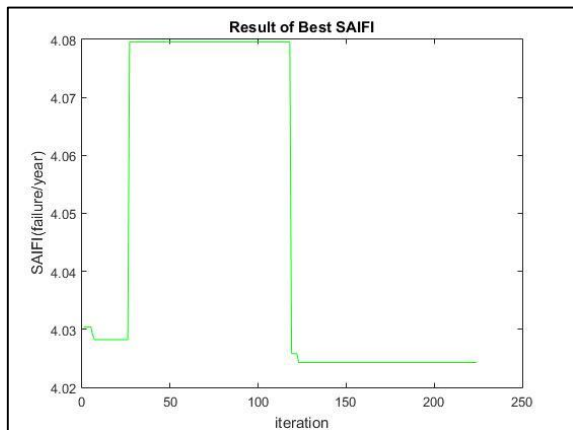


(Percobaan ke 10)

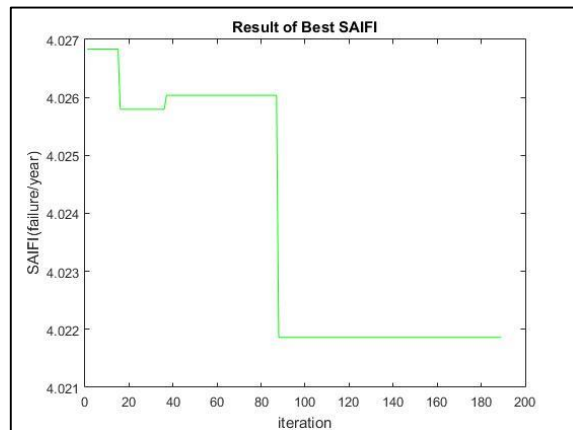
Skenario 3 lokasi sectionalizer dengan 10x percobaan menggunakan SA

percobaan	Best SAIFI (kali/tahun)	Best SAIDI (jam/tahun)	CAIDI (jam / tahun)	Lokasi Relokasi
Skenario Grid – Wind Turbine – Mikrohidro				
1	4,0243	12,2383	3,0411	21, 2, 33, 37, 45, 56, 57, 68, 70, 91, 100, 101, 102, 106, 118, 118
2	4,0219	12,214	3,036873	2, 27, 40, 42, 46, 57, 62, 70, 78, 78, 82, 83, 85, 86, 97, 99
3	4,022	12,2151	3,037071	2, 5, 10, 12, 33, 40, 46, 57, 66, 69, 82, 84, 91, 93, 105, 116
4	4,0208	12,2034	3,035068	2, 27, 33, 33, 34, 41, 42, 49, 57, 60, 70, 77, 89, 93, 98, 118
5	4,0261	12,256	3,044137	4, 13, 18, 20, 27, 57, 66, 74, 76, 76, 83, 85, 89, 99, 118, 118
6	4,0217	12,2123	3,036601	2, 20, 40, 41, 41, 46, 55, 57, 57, 68, 72, 76, 77, 80, 80, 106
7	4,0227	12,2215	3,038134	2, 3, 14, 15, 38, 40, 56, 57, 68, 85, 99, 100, 101, 106, 116, 118
8	4,0237	12,2321	3,040013	2, 9, 14, 29, 53, 57, 66, 68, 77, 80, 85, 87, 91, 118, 118, 118
9	4,0232	12,2272	3,039173	2, 12, 12, 15, 23, 25, 40, 42, 49, 57, 70, 72, 90, 93, 105, 118
10	4,0206	12,2016	3,034771	8, 34, 35, 44, 45, 51, 51, 56, 57, 61, 62, 83, 84, 87, 97, 114

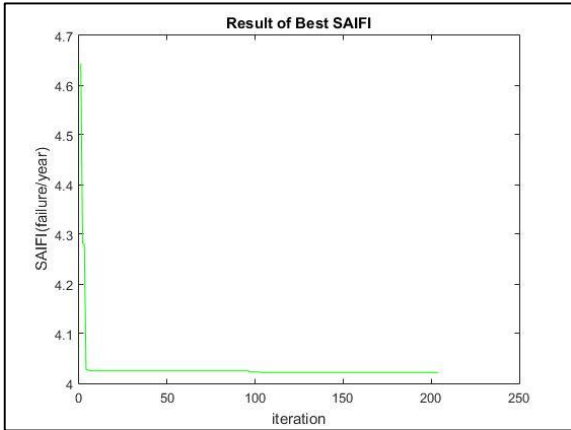
Grafik SAIFI skenario 3 lokasi sectionalizer dengan 10x percobaan menggunakan SA



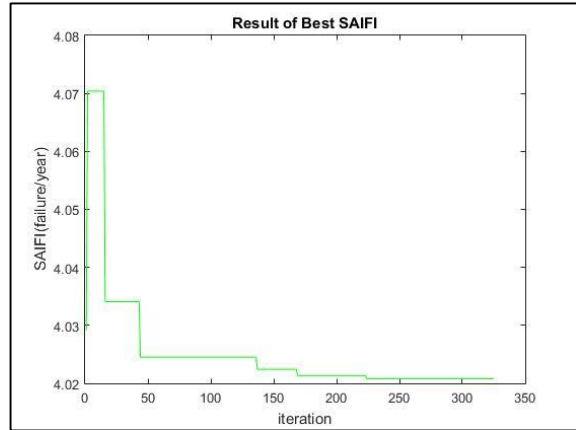
(Percobaan ke 1)



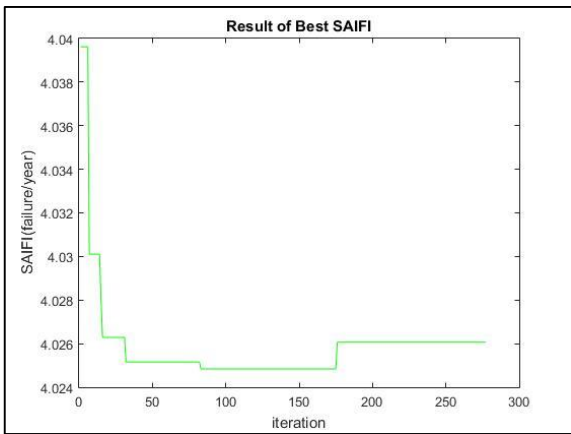
(Percobaan ke 2)



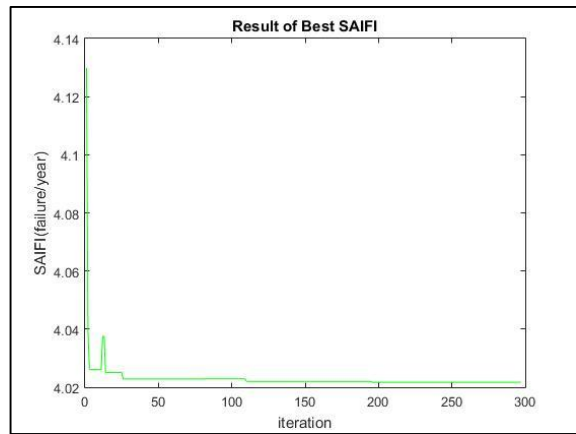
(Percobaan ke 3)



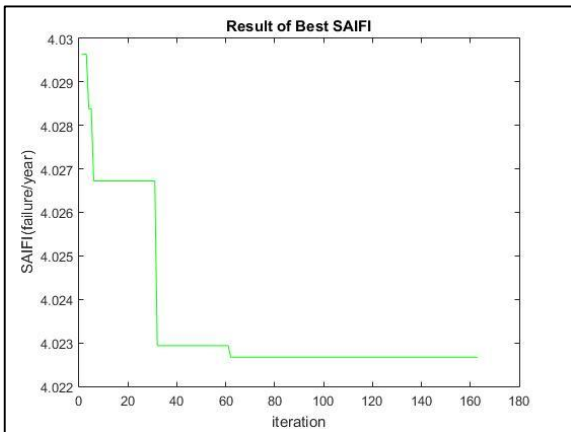
(Percobaan ke 4)



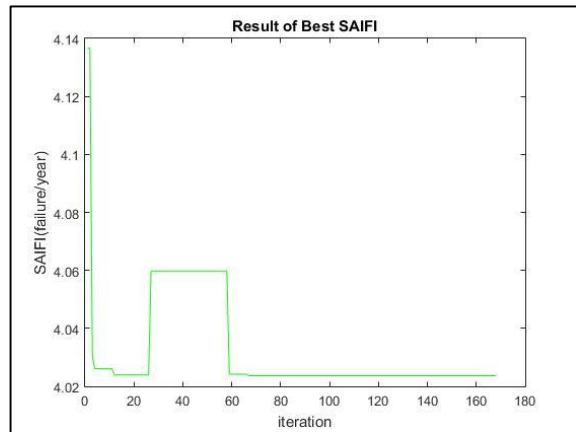
(Percobaan ke 5)



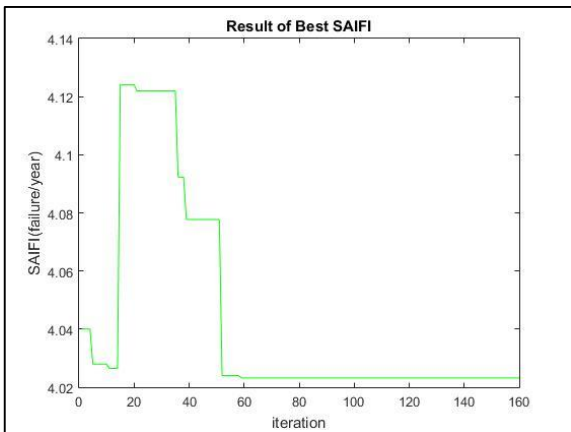
(Percobaan ke 6)



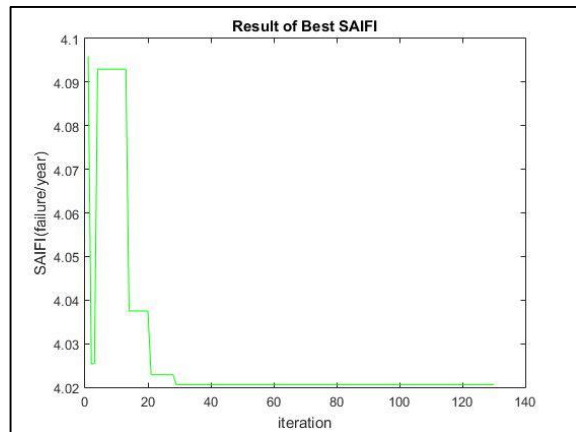
(Percobaan ke 7)



(Percobaan ke 8)

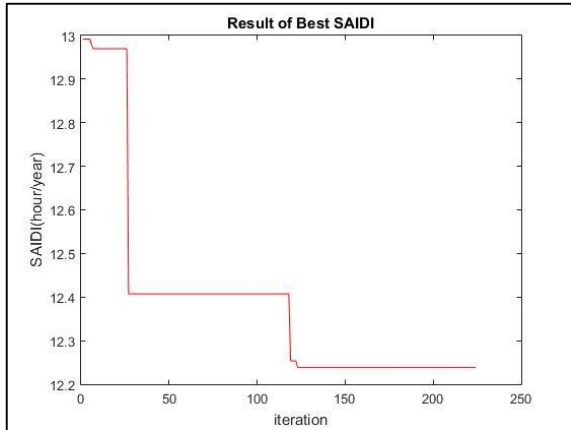


(Percobaan ke 9)

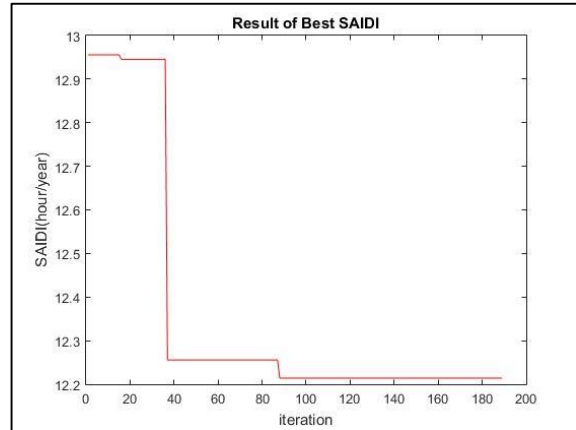


(Percobaan ke 10)

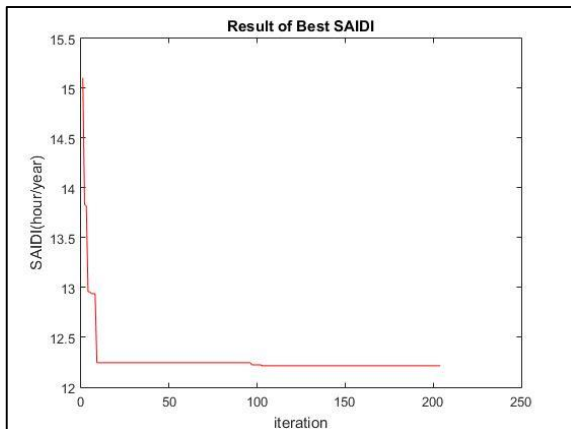
Grafik SAIDI skenario 3 lokasi sectionalizer dengan 10x percobaan menggunakan SA



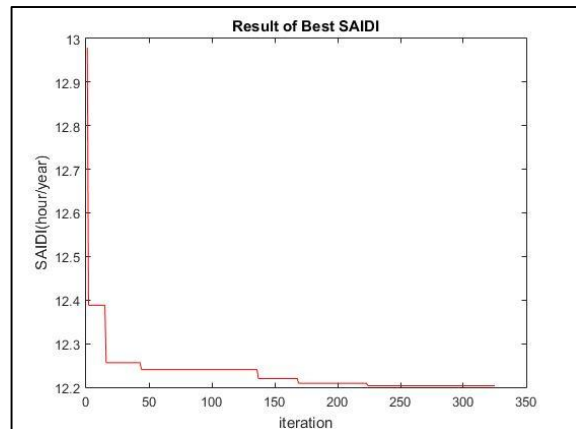
(Percobaan ke 1)



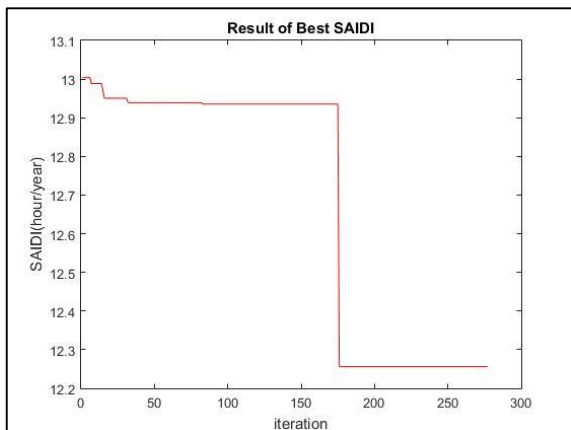
(Percobaan ke 2)



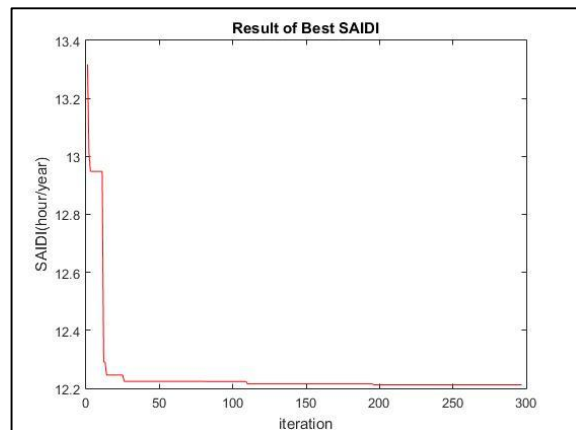
(Percobaan ke 3)



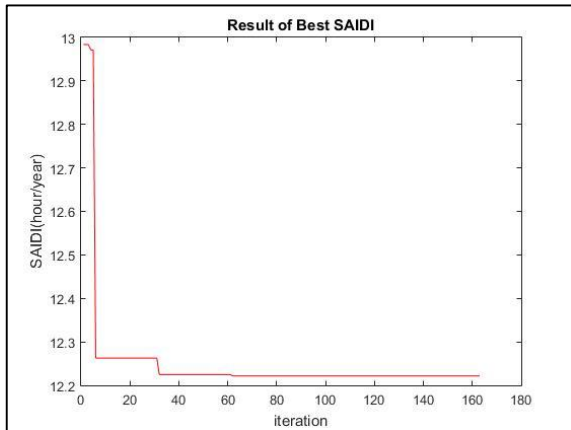
(Percobaan ke 4)



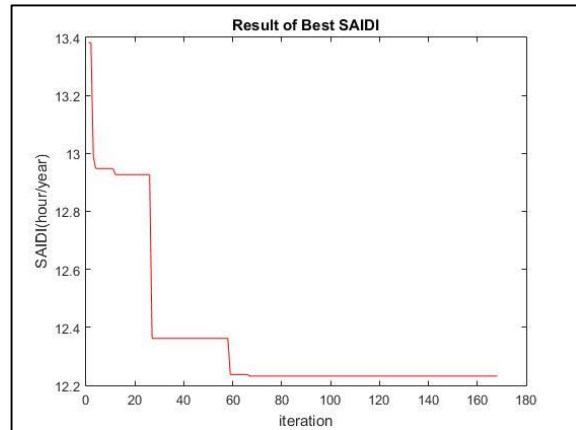
(Percobaan ke 5)



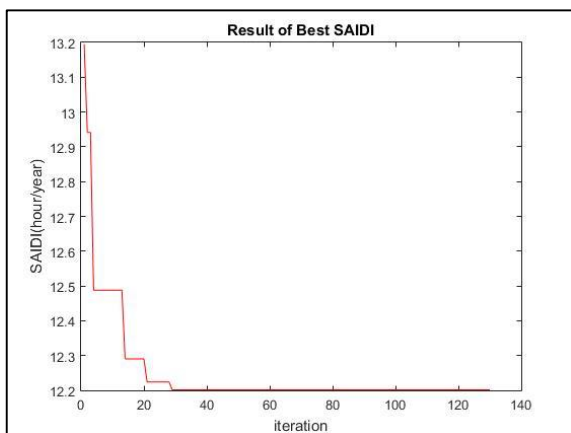
(Percobaan ke 6)



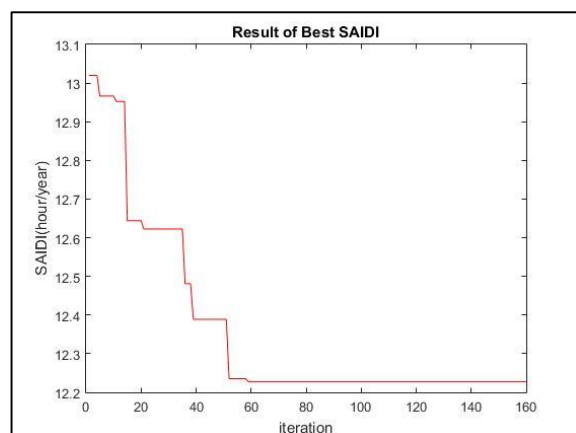
(Percobaan ke 7)



(Percobaan ke 8)

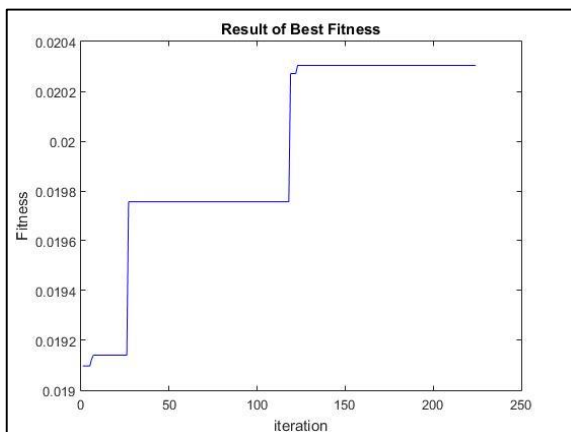


(Percobaan ke 9)

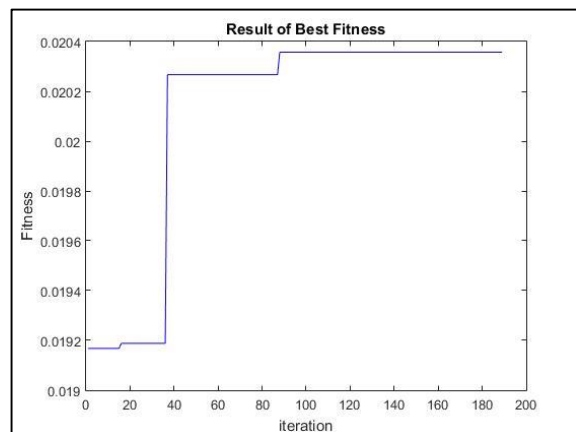


(Percobaan ke 10)

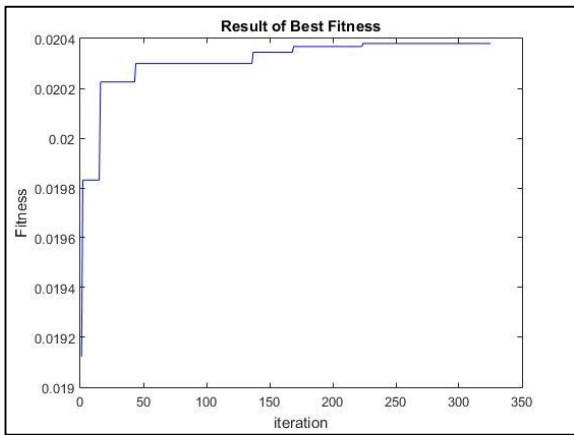
Grafik *fitness* skenario 3 lokasi sectionalizer dengan 10x percobaan menggunakan SA



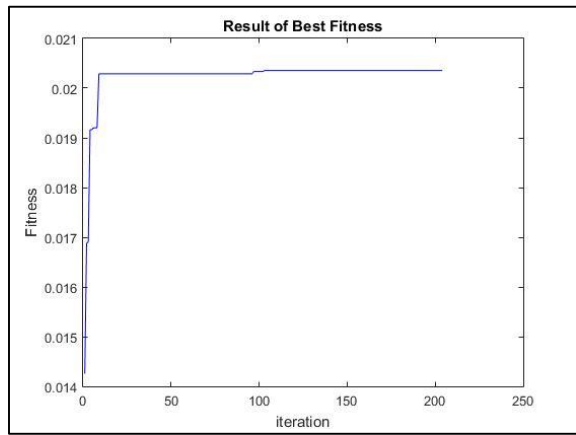
(Percobaan ke 1)



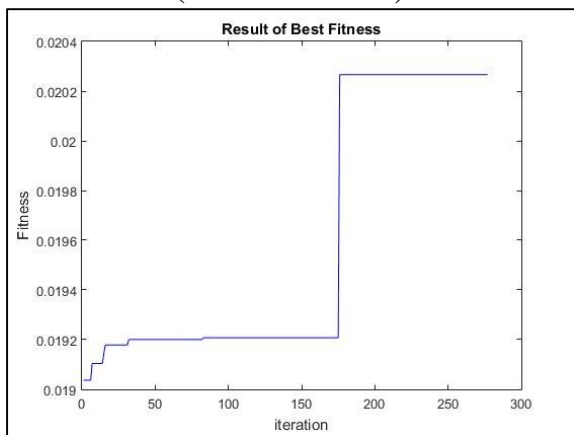
(Percobaan ke 2)



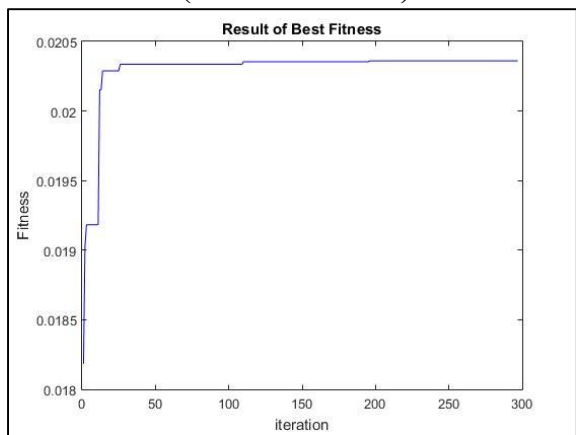
(Percobaan ke 3)



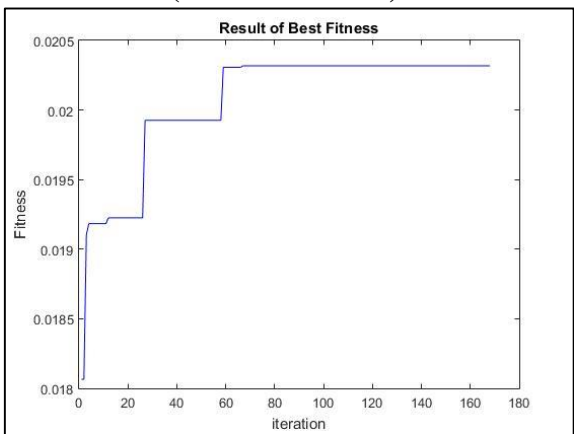
(Percobaan ke 4)



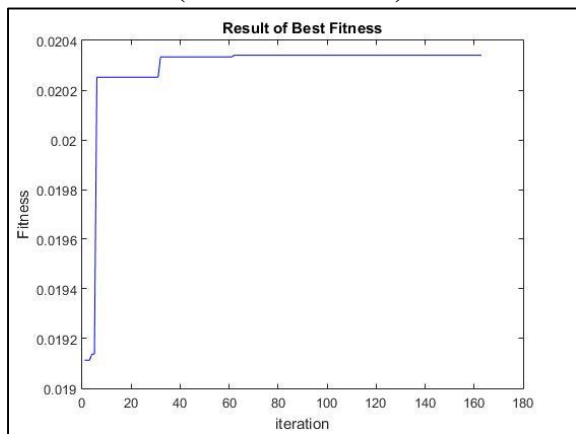
(Percobaan ke 5)



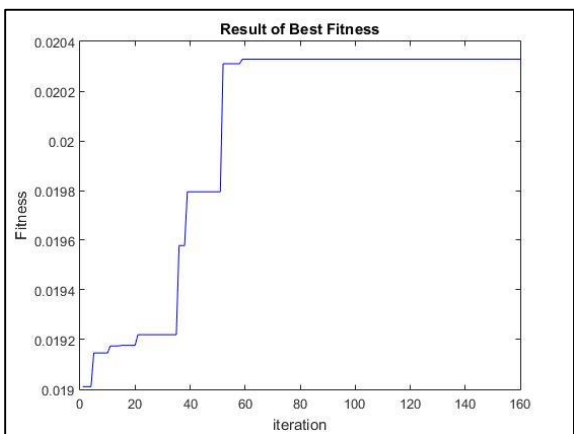
(Percobaan ke 6)



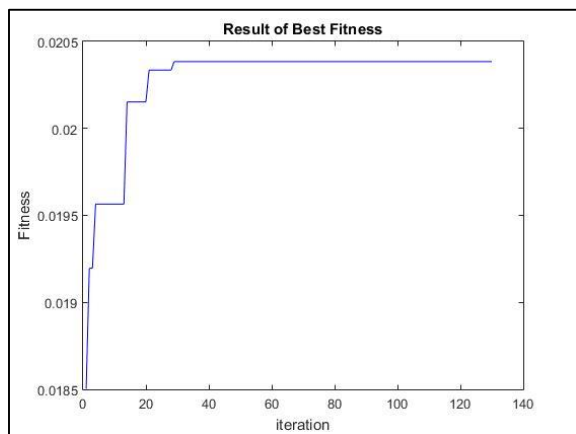
(Percobaan ke 7)



(Percobaan ke 8)



(Percobaan ke 9)

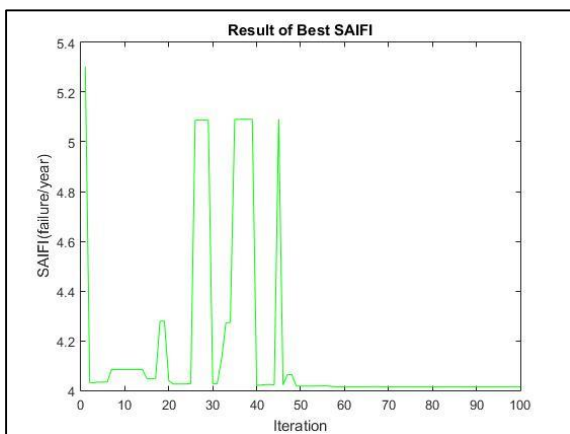


(Percobaan ke 10)

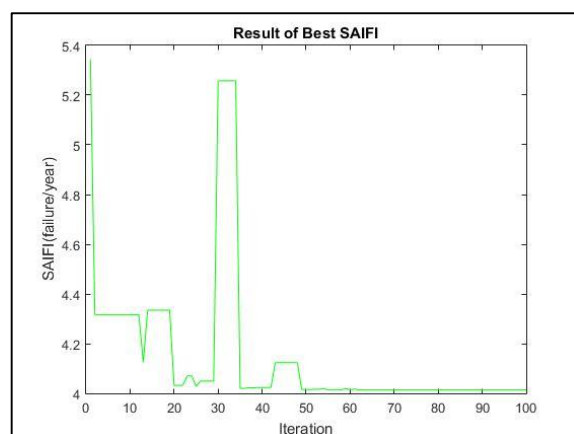
Skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan ACO

percobaan	Best SAIFI (kali/tahun)	Best SAIDI (jam/tahun)	CAIDI (jam / tahun)	Lokasi Relokasi
Skenario Grid - Wind Turbine				
1	4,0147	12,0826	3,00959	3, 7, 8, 9, 12, 33, 35, 40, 42, 43, 77, 79, 80, 97, 105
2	4,0138	12,073	3,007873	3, 7, 8, 9, 12, 33, 34, 35, 40, 41, 42, 43, 76, 81, 96, 100
3	4,015	12,0853	3,010037	3, 7, 8, 9, 12, 33, 34, 35, 41, 42, 43, 68, 82, 85, 98, 102
4	4,0147	12,0818	3,00939	3, 7, 8, 9, 12, 33, 34, 40, 41, 42, 43, 76, 79, 81, 98, 101
5	4,0159	12,0938	3,011479	3, 7, 8, 9, 12, 35, 40, 41, 42, 46, 47, 76, 77, 78, 79, 82
6	4,0156	12,0908	3,010957	3, 7, 8, 9, 12, 33, 35, 42, 43, 68, 77, 79, 80, 81, 83, 96
7	4,0165	12,0997	3,012498	3, 7, 8, 9, 12, 33, 34, 40, 42, 50, 68, 76, 78, 83, 97, 109
8	4,0169	12,1043	3,013344	7, 8, 9, 12, 13, 33, 34, 35, 40, 42, 48, 76, 78, 79, 81, 97
9	4,0145	12,08	3,009092	3, 7, 8, 9, 12, 33, 35, 40, 41, 42, 43, 47, 76, 77, 78, 102
10	4,0144	12,0795	3,009042	3, 7, 8, 9, 12, 33, 34, 35, 40, 41, 43, 76, 77, 79, 96, 97

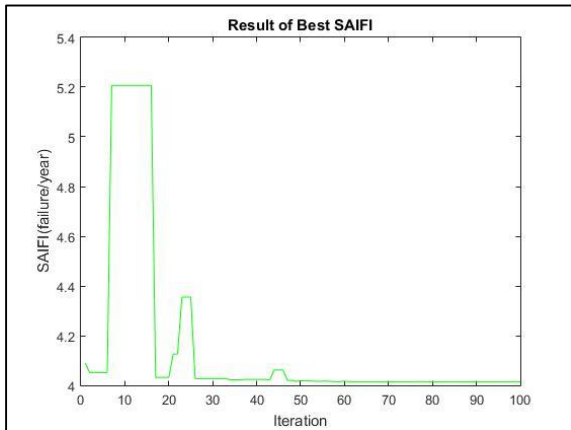
Grafik SAIFI skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan ACO



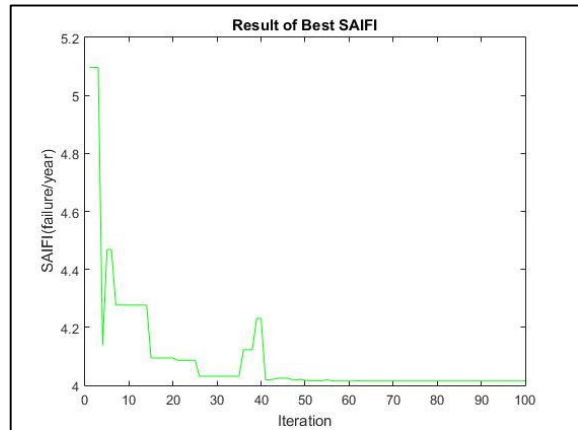
(Percobaan ke 1)



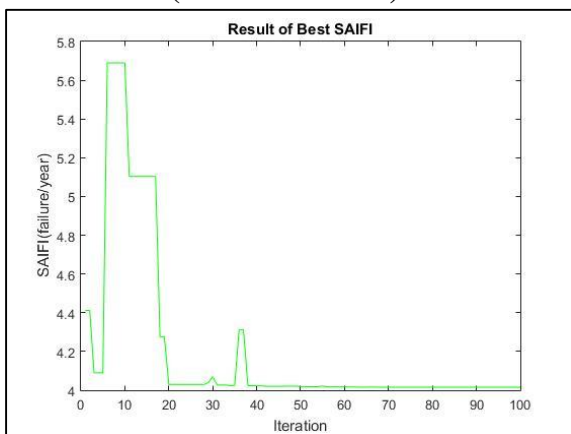
(Percobaan ke 2)



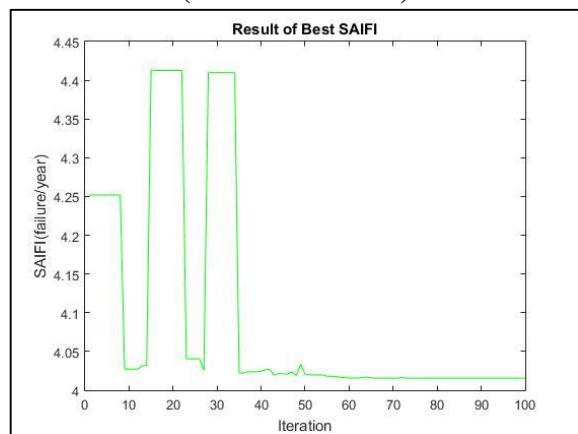
(Percobaan ke 3)



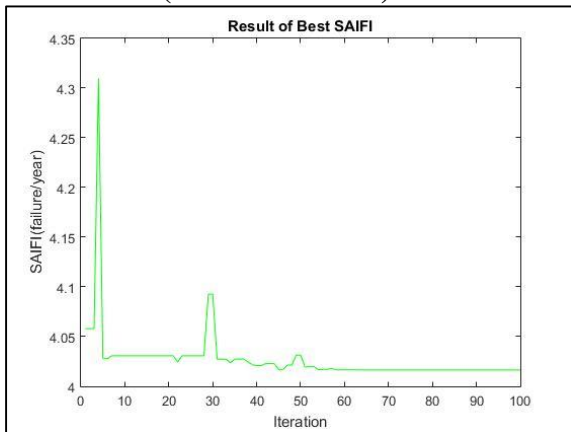
(Percobaan ke 4)



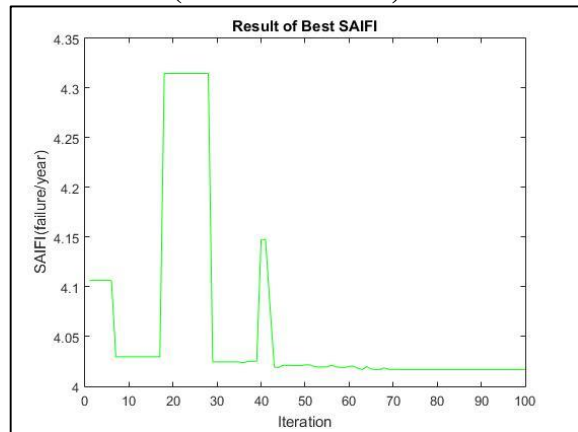
(Percobaan ke 5)



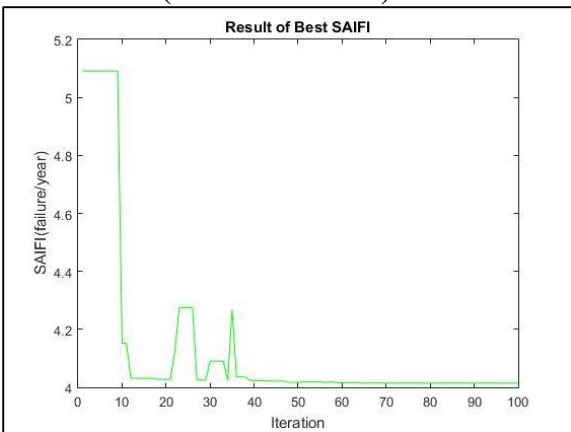
(Percobaan ke 6)



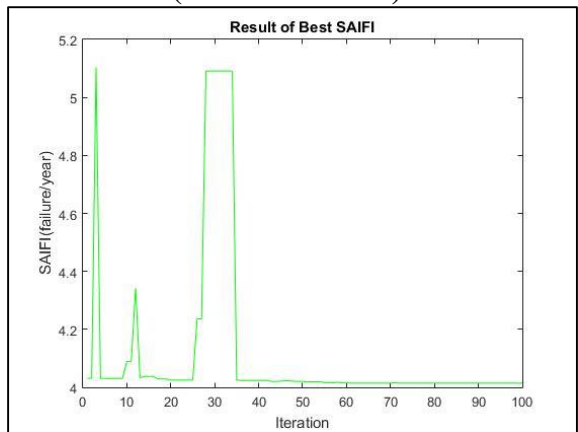
(Percobaan ke 7)



(Percobaan ke 8)

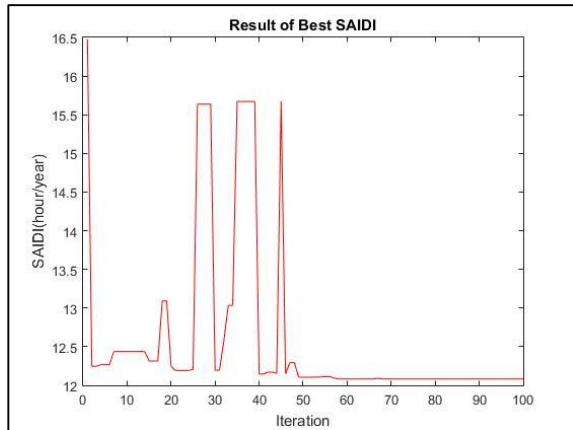


(Percobaan ke 9)

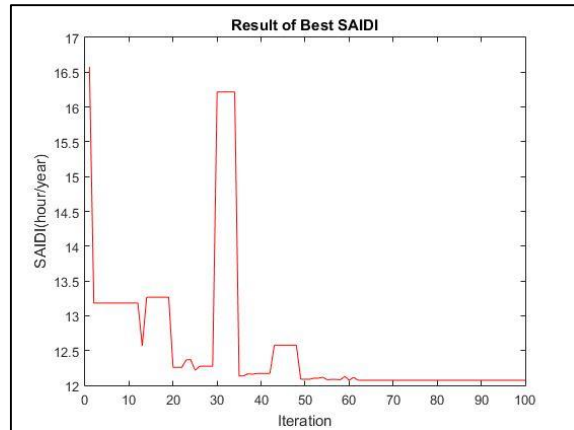


(Percobaan ke 10)

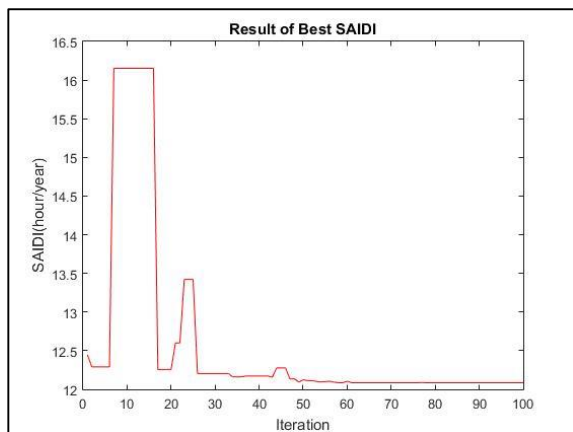
Grafik SAIFI skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan ACO



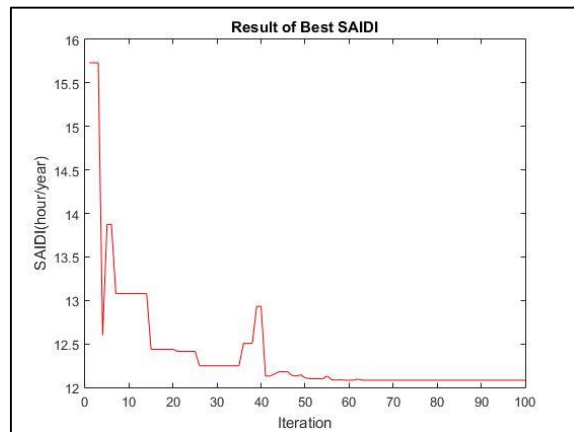
(Percobaan ke 1)



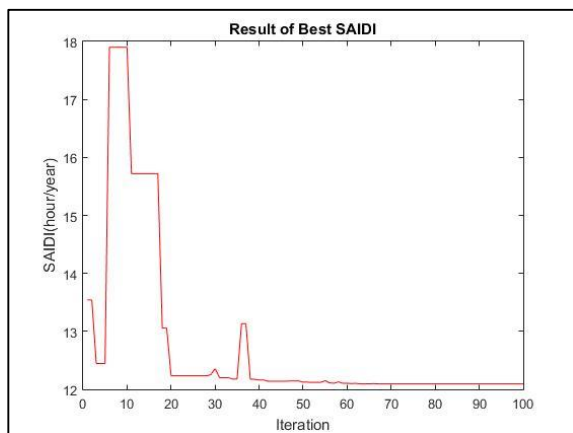
(Percobaan ke 2)



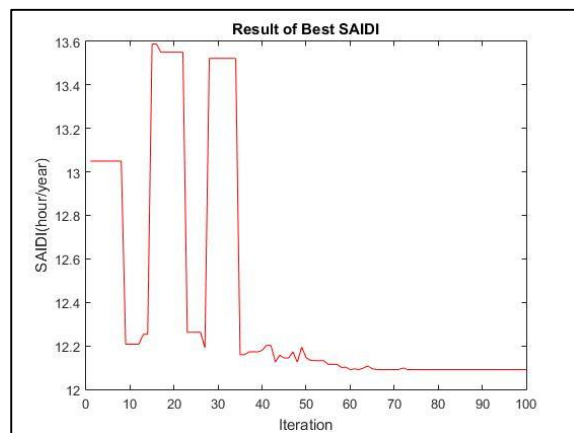
(Percobaan ke 3)



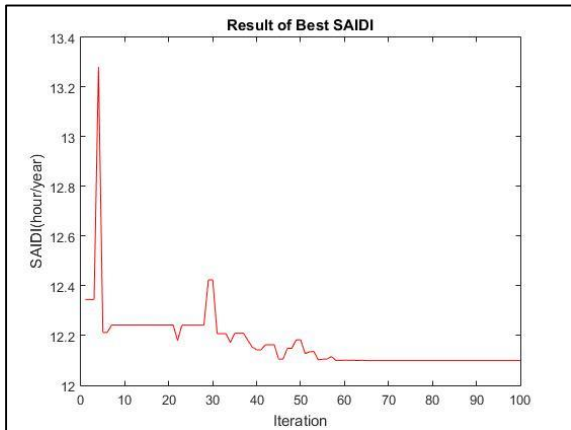
(Percobaan ke 4)



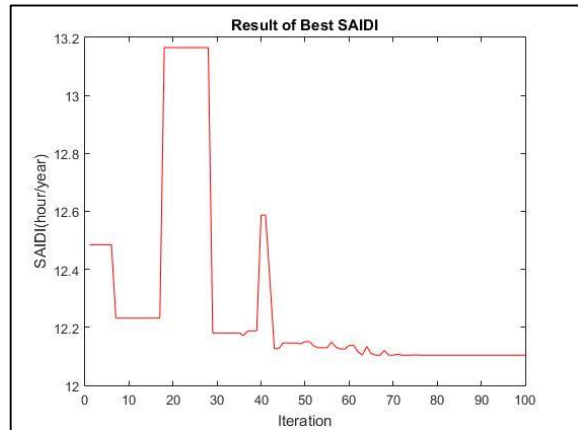
(Percobaan ke 5)



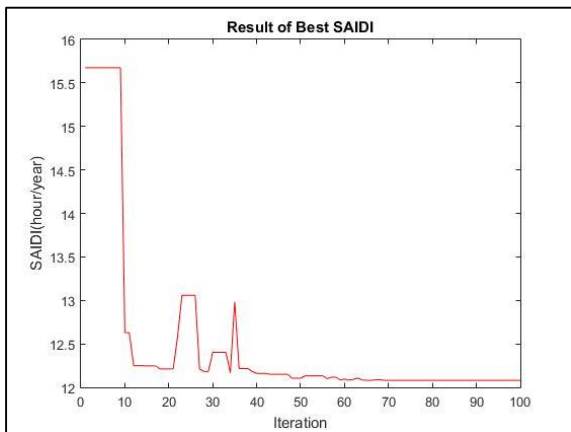
(Percobaan ke 6)



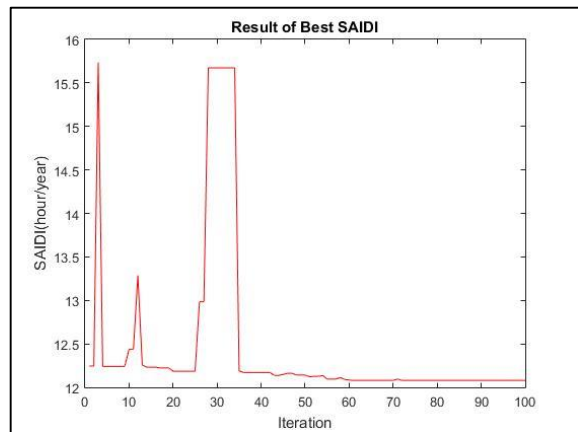
(Percobaan ke 7)



(Percobaan ke 8)

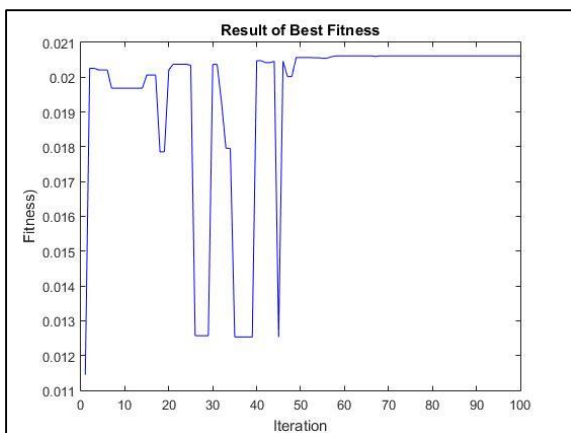


(Percobaan ke 9)

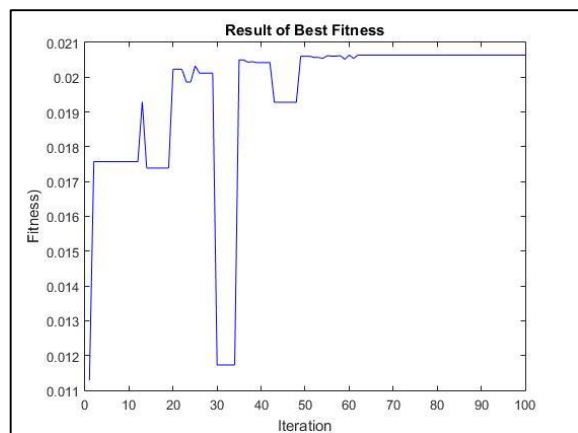


(Percobaan ke 10)

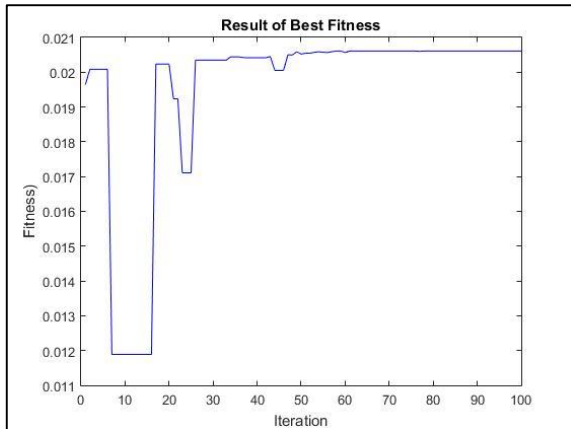
Grafik *fitness* skenario 1 lokasi *sectionalizer* dengan 10x percobaan menggunakan ACO



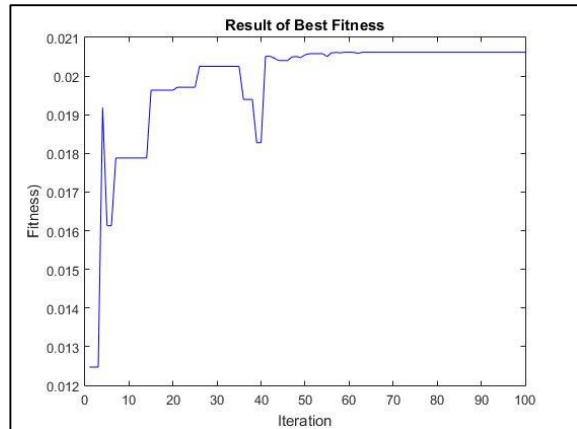
(Percobaan ke 1)



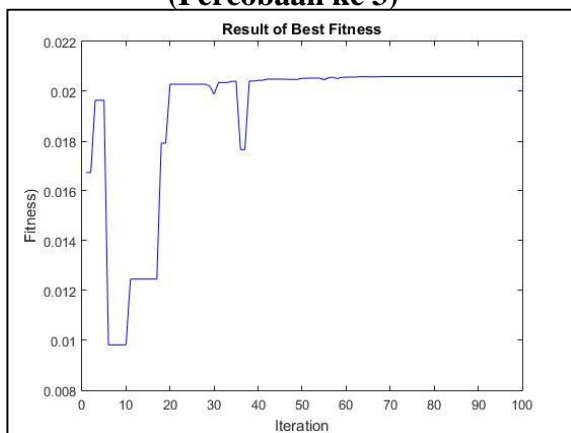
(Percobaan ke 2)



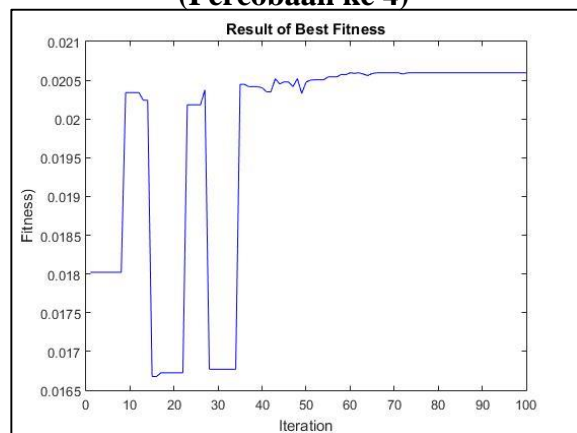
(Percobaan ke 3)



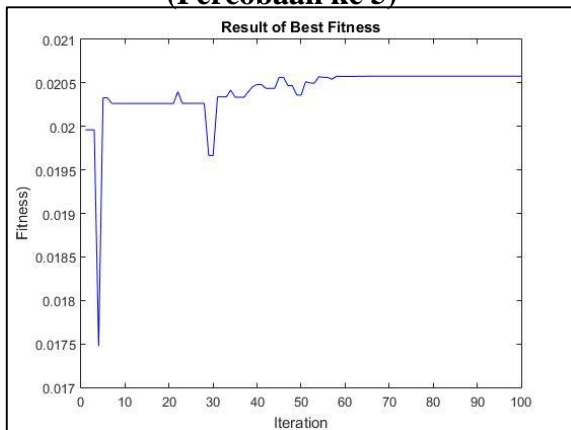
(Percobaan ke 4)



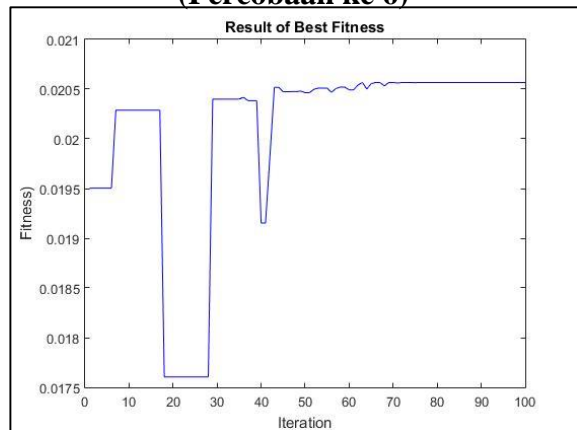
(Percobaan ke 5)



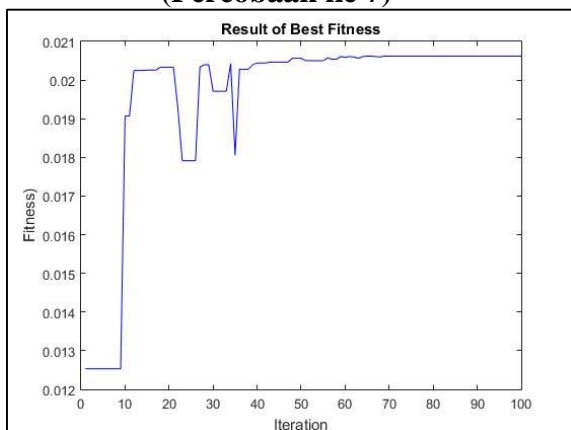
(Percobaan ke 6)



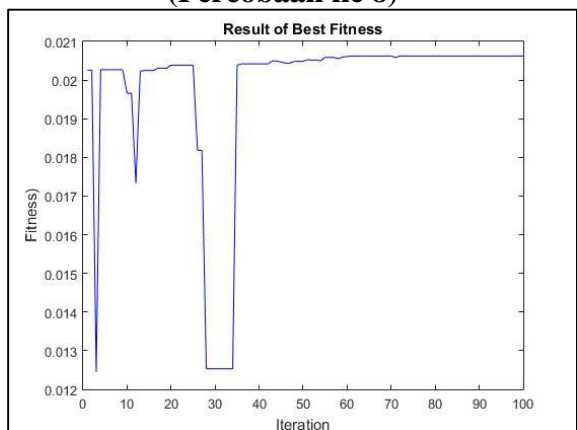
(Percobaan ke 7)



(Percobaan ke 8)



(Percobaan ke 9)

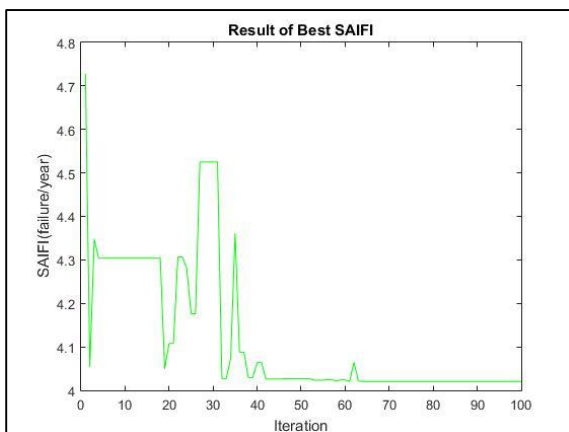


(Percobaan ke 10)

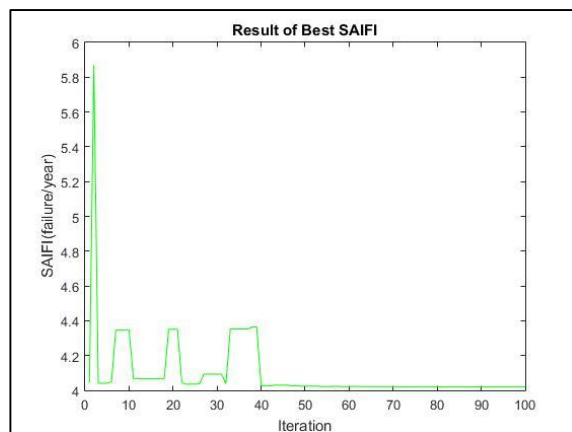
Skenario 2 lokasi sectionalizer dengan 10x percobaan menggunakan ACO

percobaan	Best SAIFI (kali/tahun)	Best SAIDI (jam/tahun)	CAIDI (jam / tahun)	Lokasi Relokasi
Skenario Grid - Mikrohidro				
1	4,0205	12,1112	3,012362	3, 7, 8, 9, 12, 33, 35, 41, 42, 43, 48, 51, 57, 77, 79, 82
2	4,0214	12,1201	3,013901	3, 7, 8, 9, 12, 33, 34, 35, 43, 46, 57, 59, 68, 77, 78, 102
3	4,0216	12,1226	3,014372	3, 7, 8, 9, 12, 33, 34, 35, 40, 49, 57, 77, 79, 81, 85, 102
4	4,0217	12,1236	3,014546	7, 8, 9, 12, 33, 34, 35, 40, 41, 43, 57, 64, 68, 81, 104, 106
5	4,0214	12,1199	3,013851	3, 8, 9, 12, 33, 34, 35, 40, 41, 42, 48, 57, 68, 83, 98, 110
6	4,21	12,1166	2,878052	3, 7, 8, 9, 12, 13, 33, 40, 41, 42, 43, 46, 57, 76, 79, 106
7	4,0206	12,1119	3,012461	3, 7, 8, 9, 12, 25, 33, 34, 35, 40, 42, 43, 57, 80, 85, 96
8	4,0202	12,1085	3,011915	3, 7, 8, 9, 12, 33, 34, 35, 42, 43, 47, 48, 57, 77, 78, 81
9	4,0194	12,1002	3,010449	3, 7, 8, 9, 12, 33, 34, 35, 40, 41, 42, 45, 57, 68, 79, 97
10	4,0209	12,1158	3,013206	3, 7, 8, 9, 12, 34, 35, 40, 42, 43, 44, 50, 57, 78, 82, 106

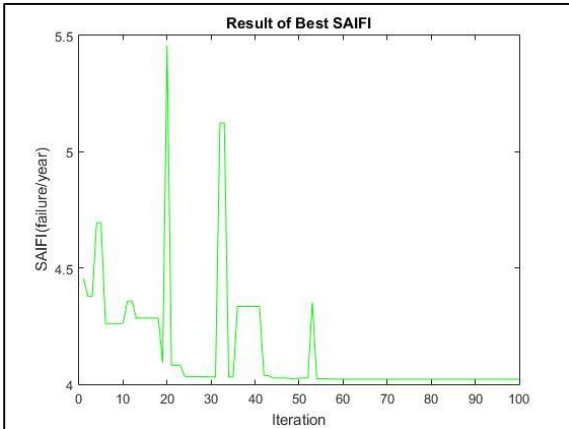
Grafik fitness skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan SA



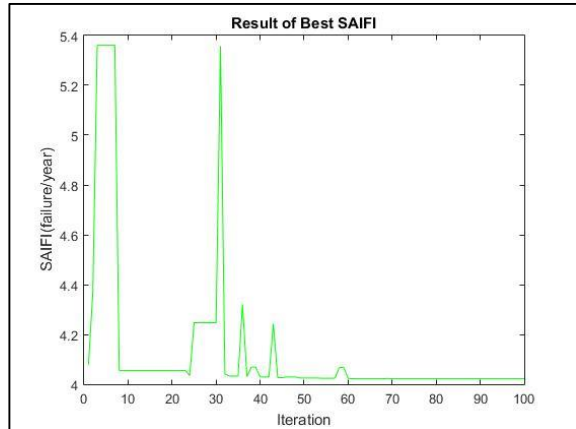
(Percobaan ke 1)



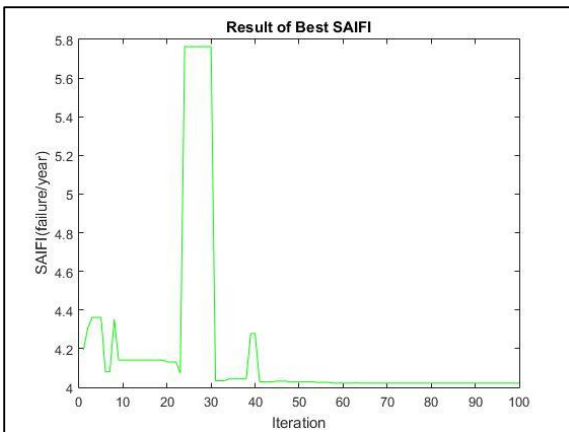
(Percobaan ke 2)



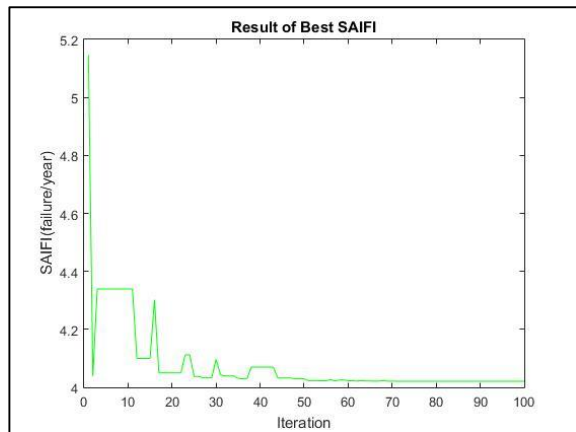
(Percobaan ke 3)



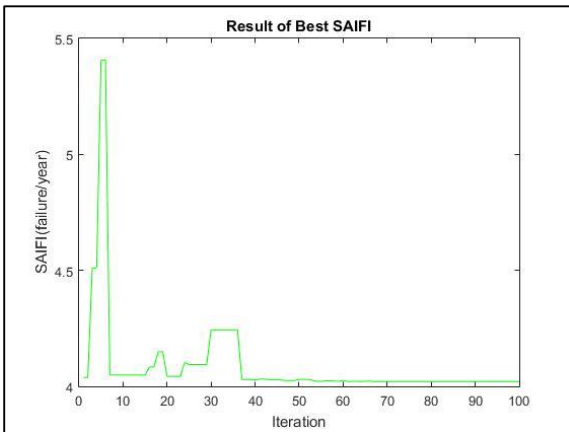
(Percobaan ke 4)



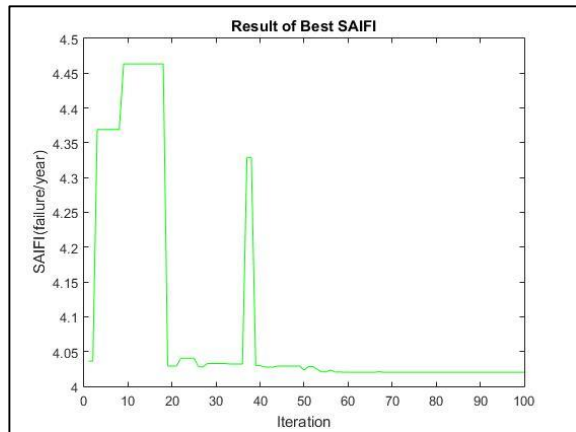
(Percobaan ke 5)



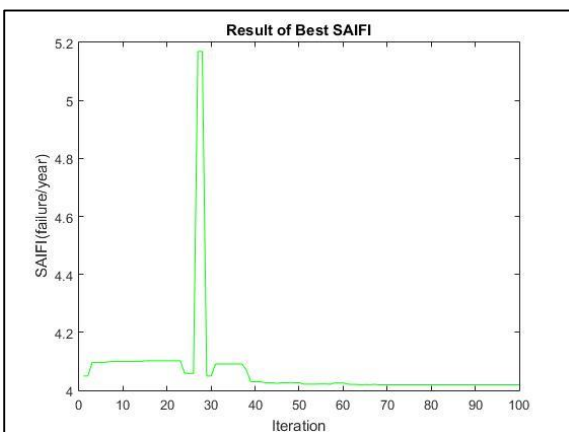
(Percobaan ke 6)



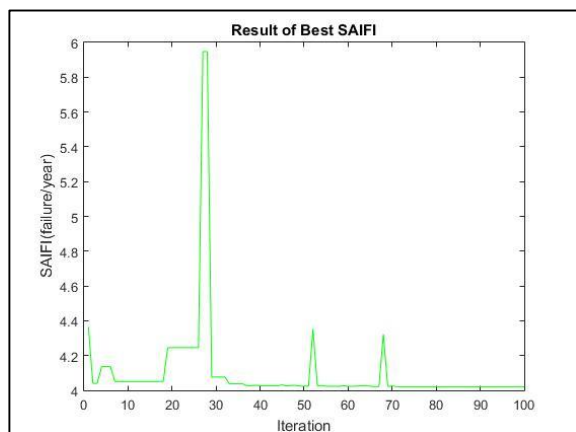
(Percobaan ke 7)



(Percobaan ke 8)

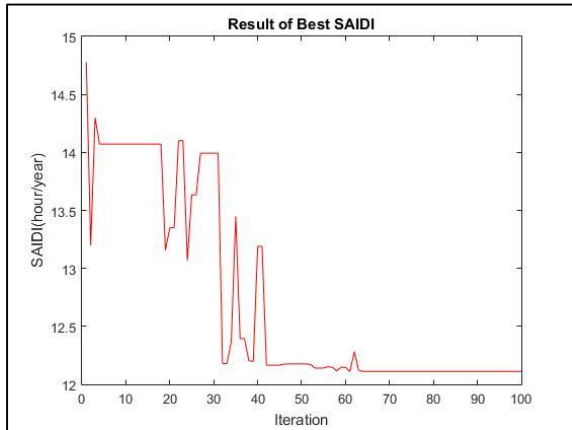


(Percobaan ke 9)

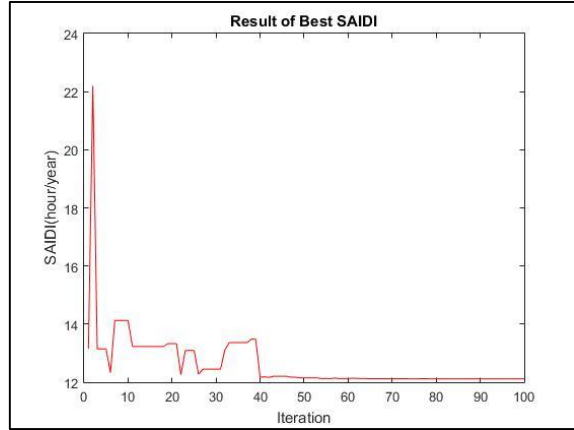


(Percobaan ke 10)

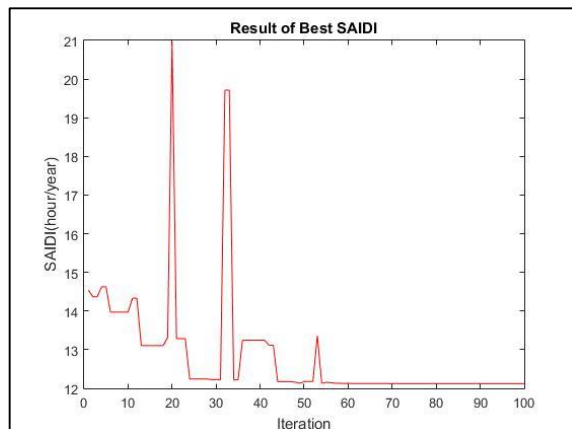
Grafik SAIDI skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan ACO



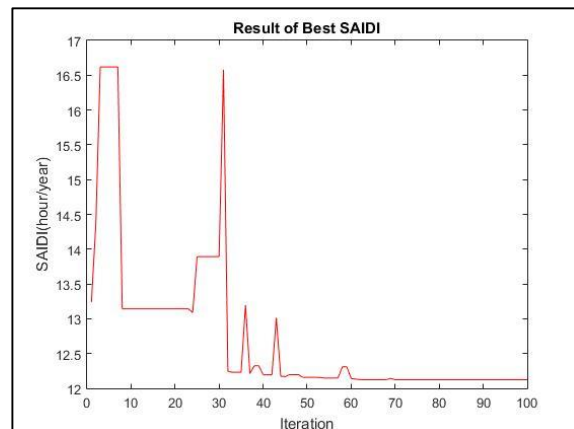
(Percobaan ke 1)



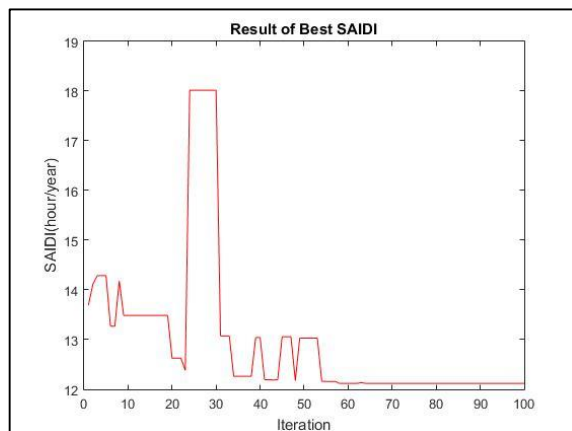
(Percobaan ke 2)



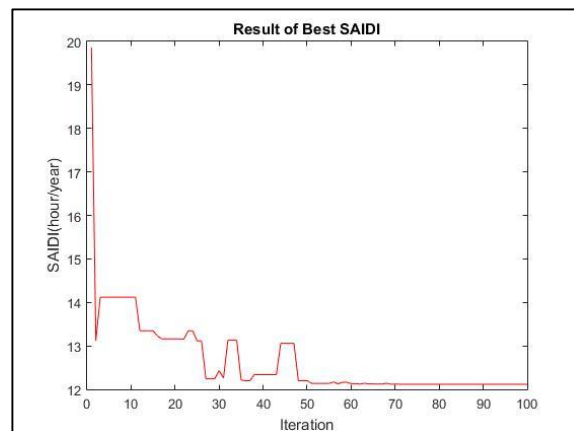
(Percobaan ke 3)



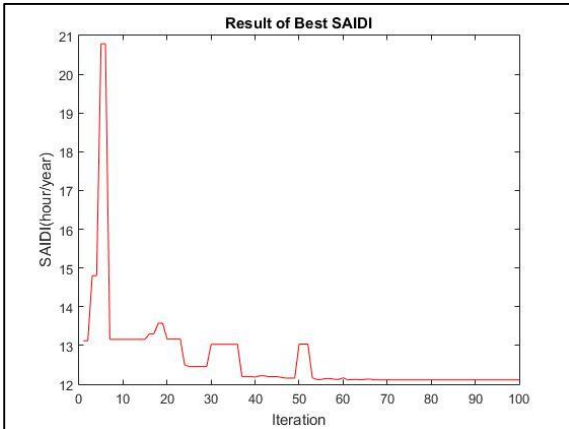
(Percobaan ke 4)



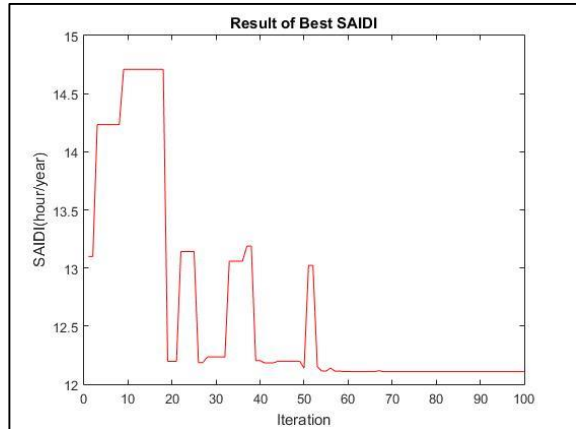
(Percobaan ke 5)



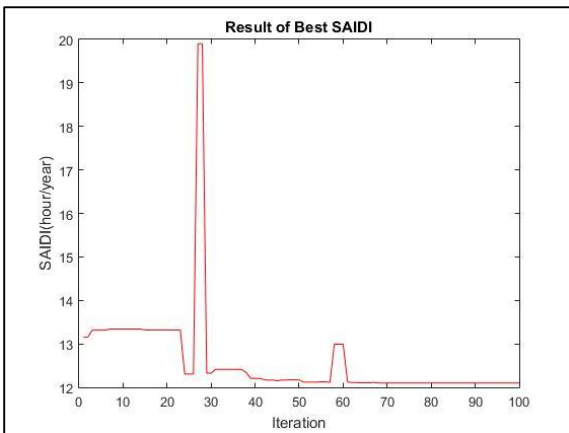
(Percobaan ke 6)



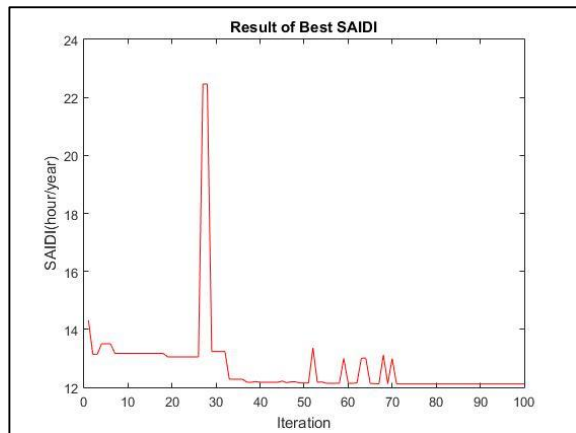
(Percobaan ke 7)



(Percobaan ke 8)

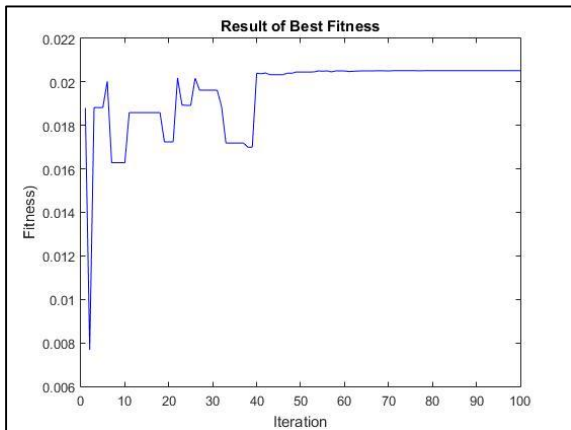


(Percobaan ke 9)

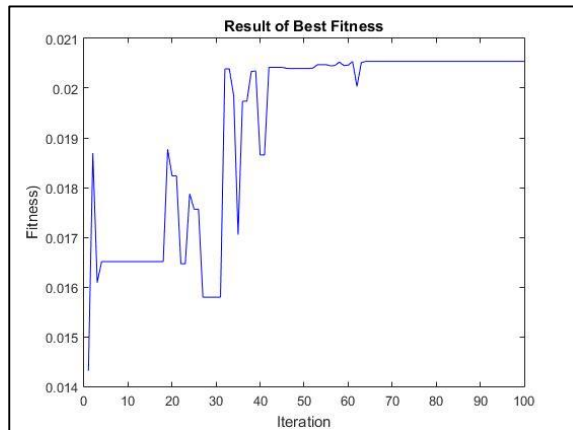


(Percobaan ke 10)

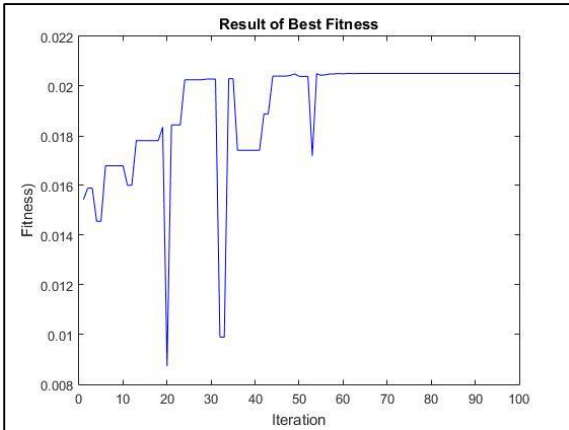
Grafik *fitness* skenario 1 lokasi sectionalizer dengan 10x percobaan menggunakan ACO



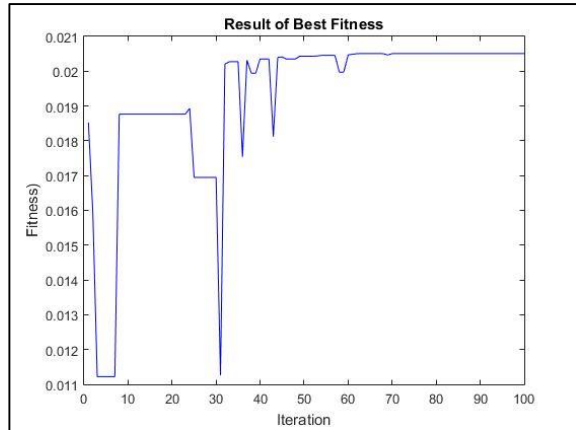
(Percobaan ke 1)



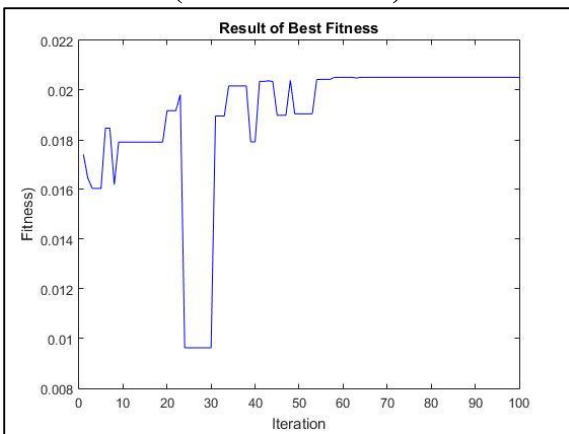
(Percobaan ke 1)



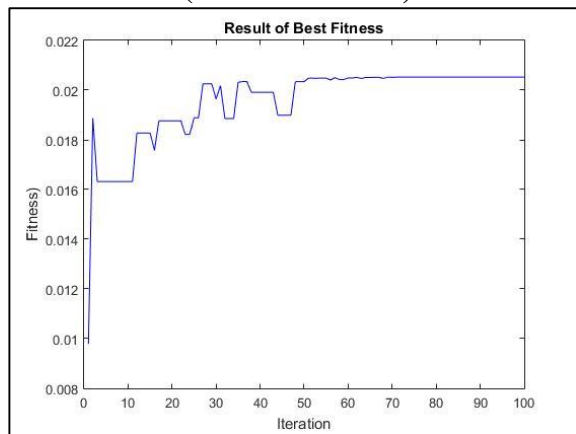
(Percobaan ke 3)



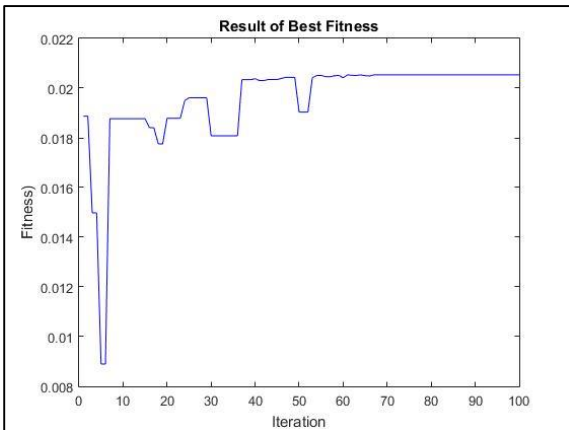
(Percobaan ke 4)



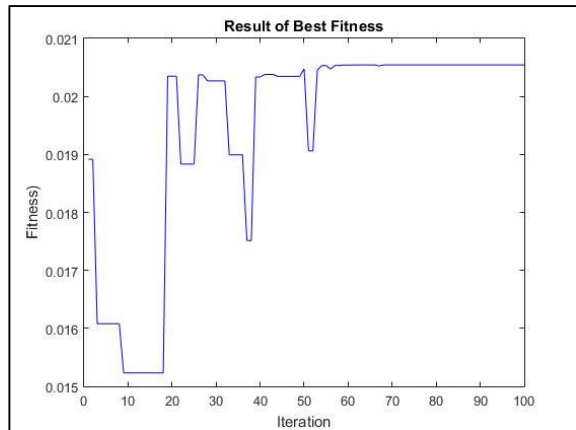
(Percobaan ke 5)



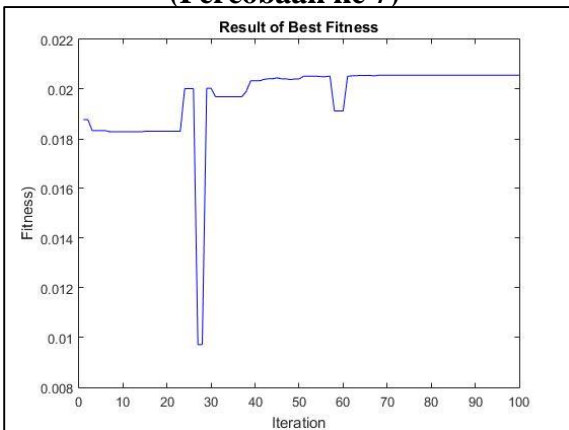
(Percobaan ke 6)



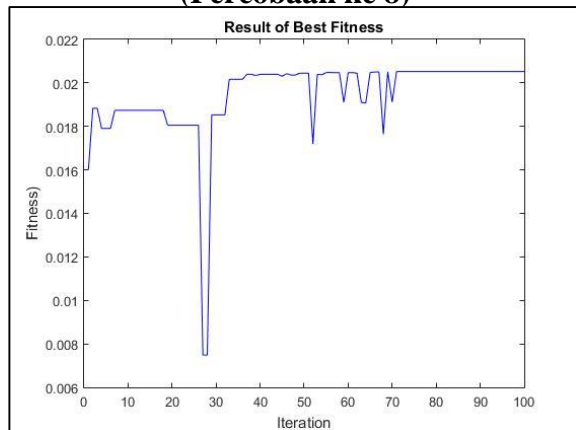
(Percobaan ke 7)



(Percobaan ke 8)



(Percobaan ke 9)

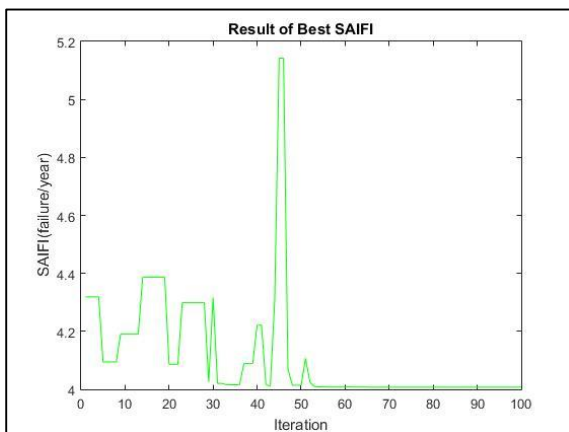


(Percobaan ke 10)

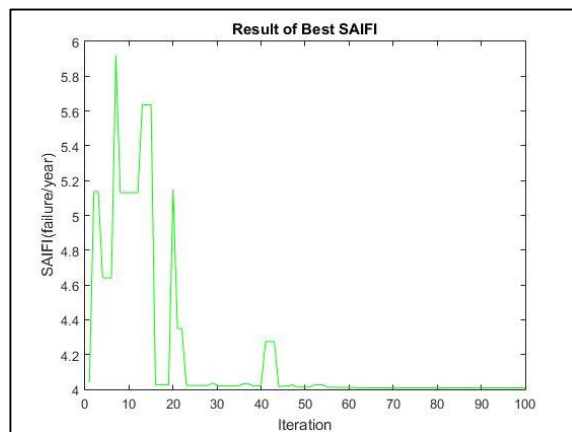
Skenario 3 lokasi *sectionalizer* dengan 10x percobaan menggunakan ACO

percobaan	Best SAIFI (kali/tahun)	Best SAIDI (jam/tahun)	CAIDI (jam / tahun)	Lokasi Relokasi
Skenario Grid – Wind Turbine – Mikrohidro				
1	4,0076	12,071	3,012027	3, 7, 8, 9, 12, 33, 35, 40, 41, 42, 43, 57, 77, 79, 80, 84
2	4,0088	12,0838	3,014318	3, 8, 9, 12, 33, 35, 40, 41, 42, 43, 46, 57, 68, 77, 78, 85
3	4,0082	12,0071	2,995634	3, 7, 8, 9, 12, 33, 34, 35, 41, 42, 45, 57, 68, 76, 81, 96
4	4,0097	12,0927	3,015862	3, 7, 8, 9, 12, 33, 41, 42, 43, 47, 56, 57, 68, 79, 81, 96
5	4,0082	12,0778	3,013273	3, 7, 8, 9, 12, 33, 34, 35, 40, 42, 43, 47, 57, 77, 85, 100
6	4,0085	12,0802	3,013646	3, 7, 8, 9, 12, 15, 33, 34, 35, 40, 43, 57, 68, 76, 77, 98
7	4,0093	12,0879	3,014965	3, 7, 8, 9, 12, 13, 33, 35, 41, 42, 43, 45, 57, 78, 83, 102
8	4,0096	12,0913	3,015588	7, 8, 9, 12, 28, 33, 35, 40, 41, 42, 43, 44, 57, 68, 80, 97
9	4,0093	12,088	3,01499	3, 7, 8, 9, 12, 33, 34, 41, 43, 44, 57, 77, 79, 80, 83, 98
10	4,0103	12,0987	3,016906	3, 7, 8, 9, 33, 34, 35, 40, 42, 45, 51, 57, 68, 78, 86, 100

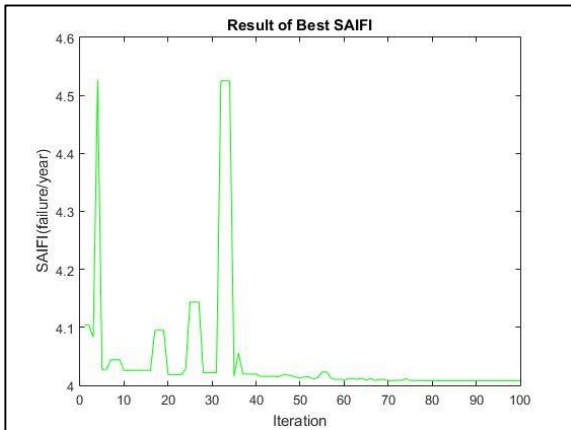
Grafik SAIFI skenario 3 lokasi *sectionalizer* dengan 10x percobaan menggunakan ACO



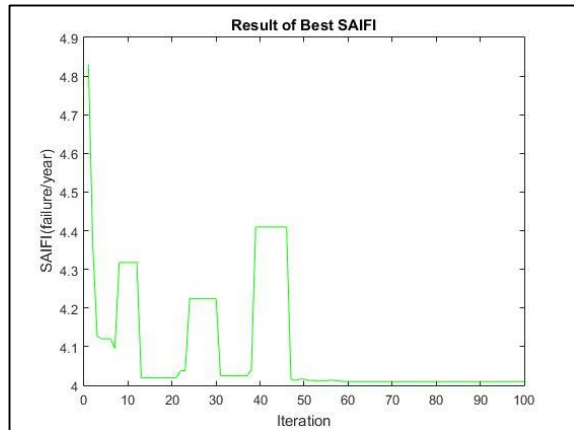
(Percobaan ke 1)



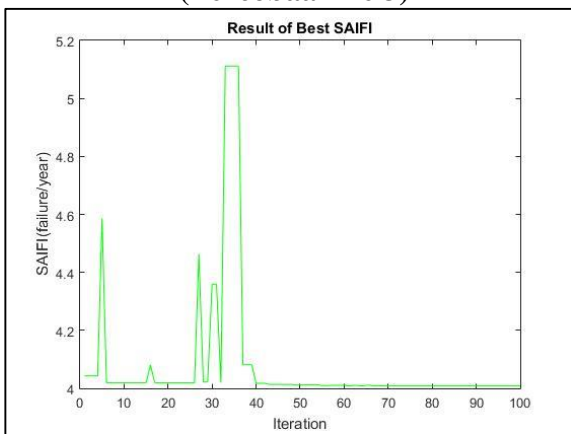
(Percobaan ke 2)



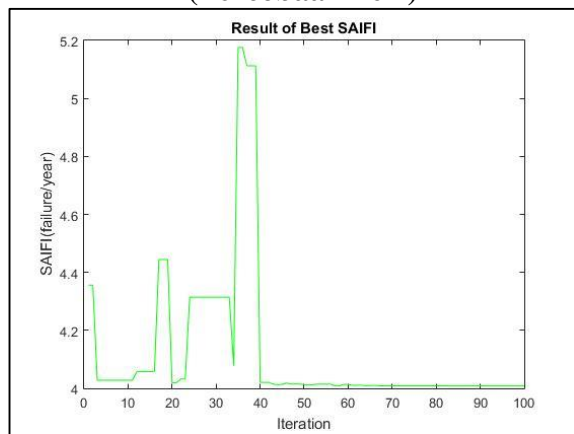
(Percobaan ke 3)



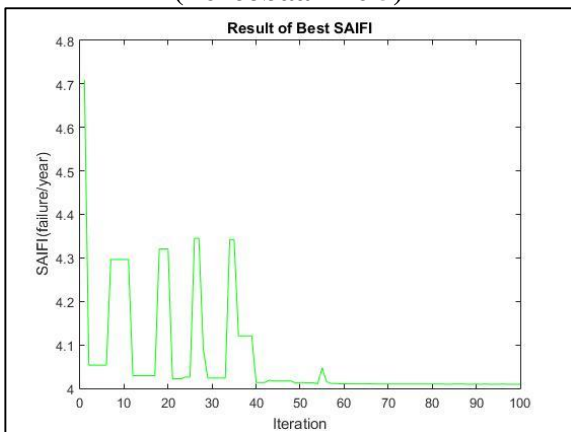
(Percobaan ke 4)



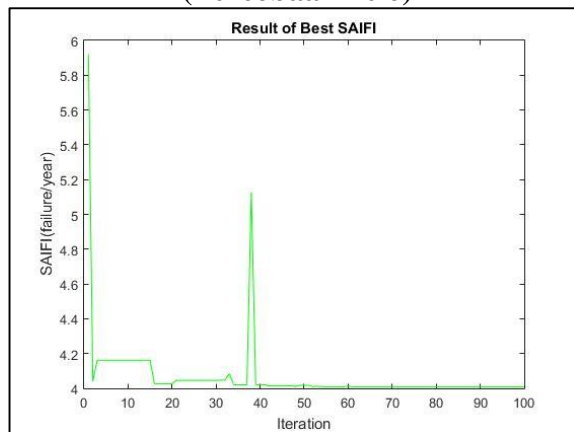
(Percobaan ke 5)



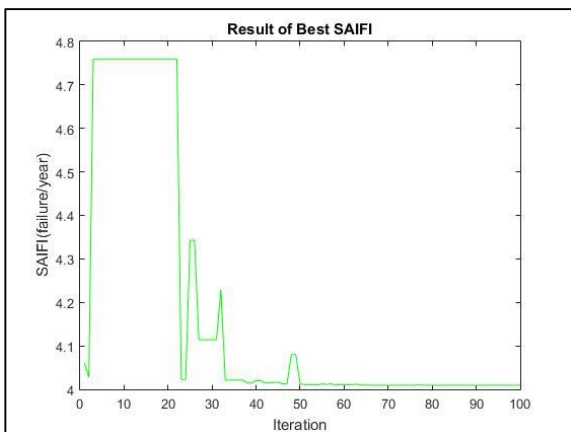
(Percobaan ke 6)



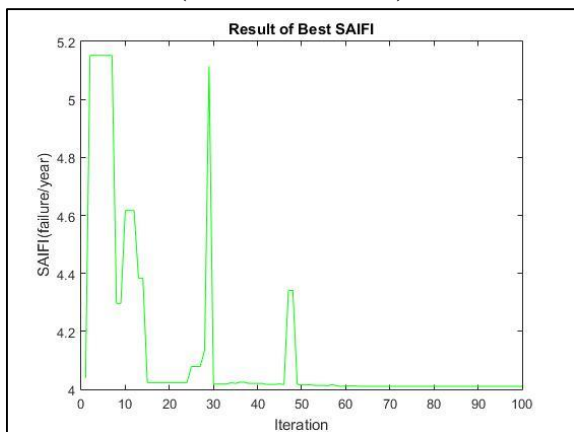
(Percobaan ke 7)



(Percobaan ke 8)

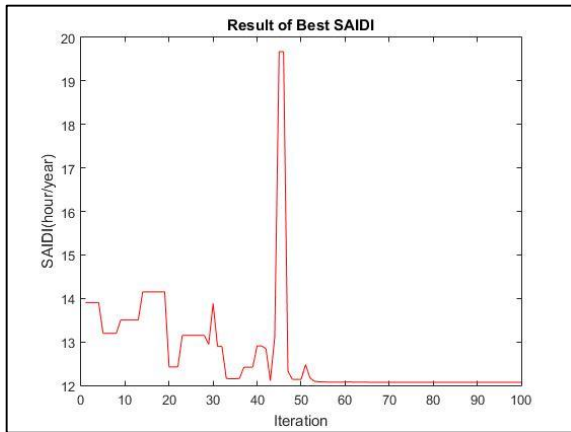


(Percobaan ke 9)

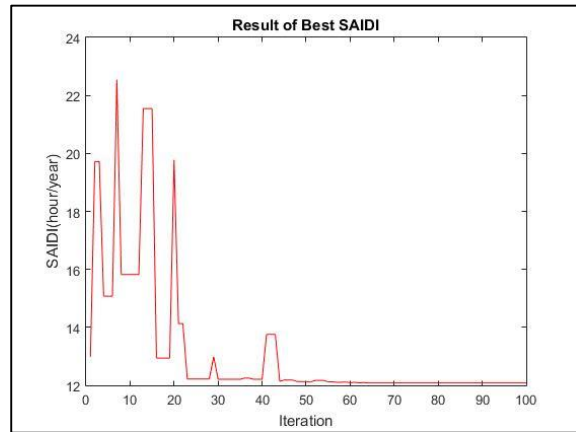


(Percobaan ke 10)

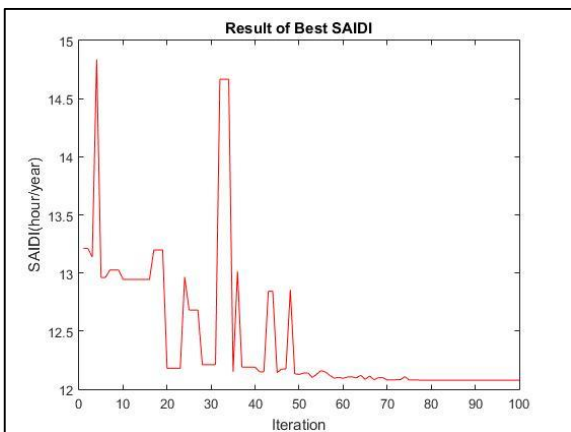
Grafik SAIDI skenario 3 lokasi sectionalizer dengan 10x percobaan menggunakan ACO



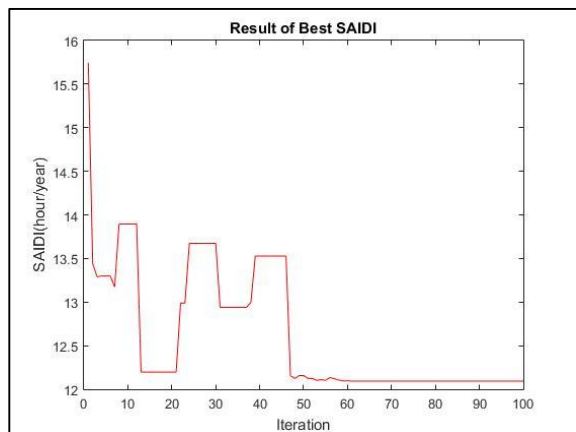
(Percobaan ke 1)



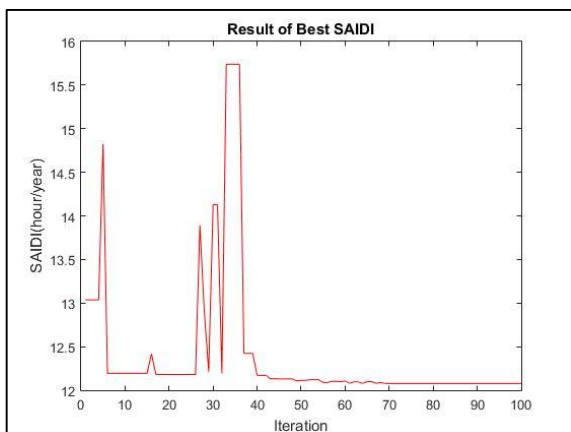
(Percobaan ke 2)



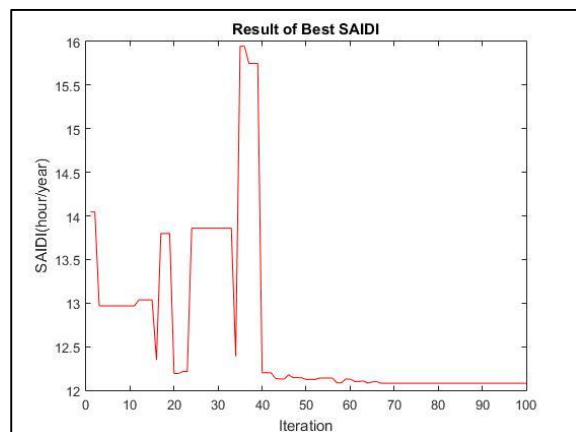
(Percobaan ke 3)



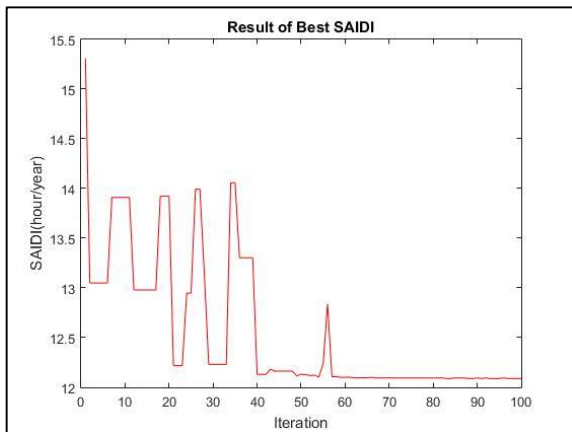
(Percobaan ke 4)



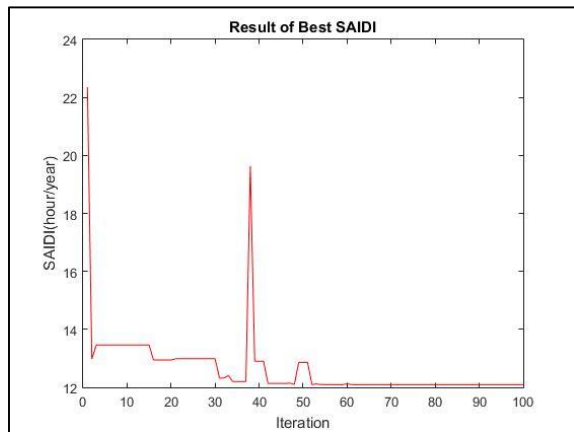
(Percobaan ke 5)



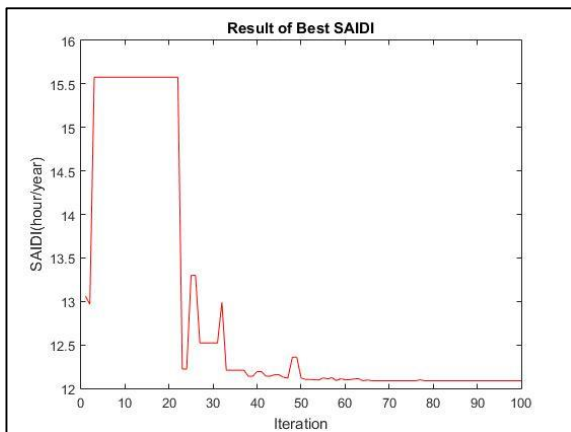
(Percobaan ke 6)



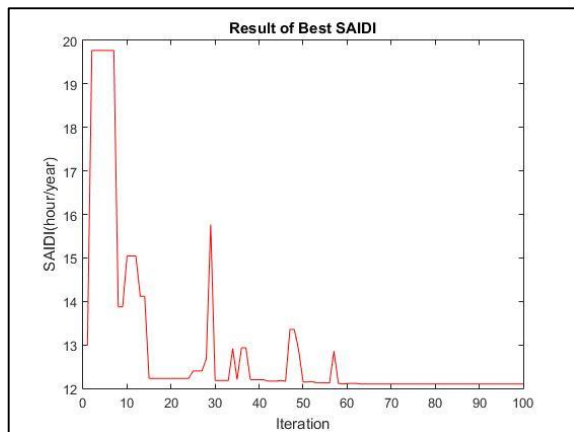
(Percobaan ke 7)



(Percobaan ke 8)

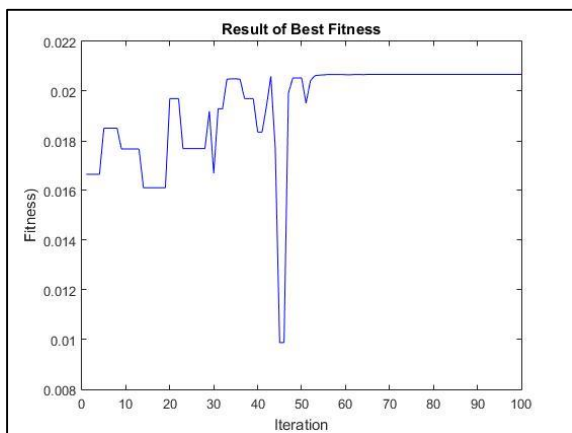


(Percobaan ke 9)

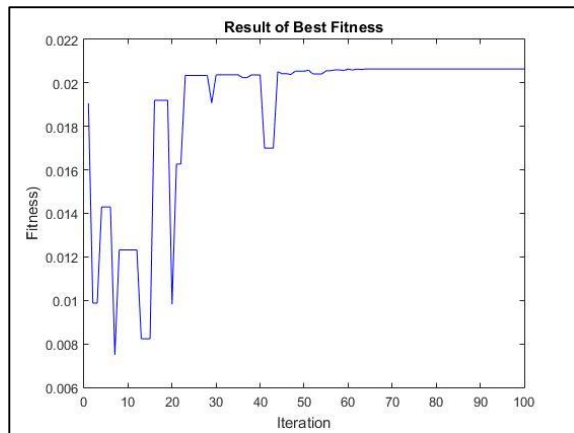


(Percobaan ke 10)

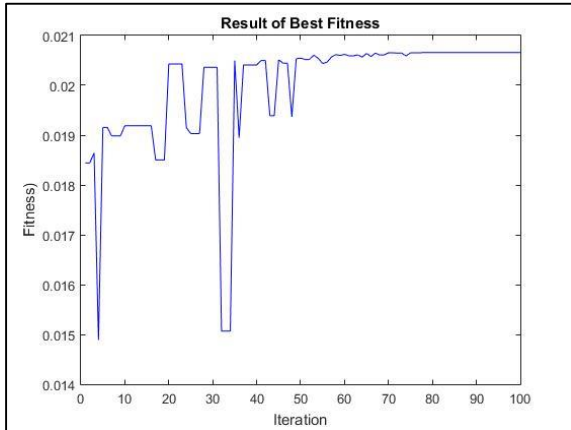
Grafik *fitness* skenario 3 lokasi *sectionalizer* dengan 10x percobaan menggunakan ACO



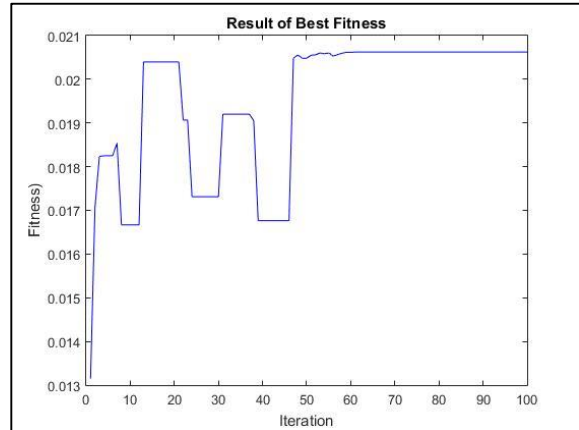
(Percobaan ke 1)



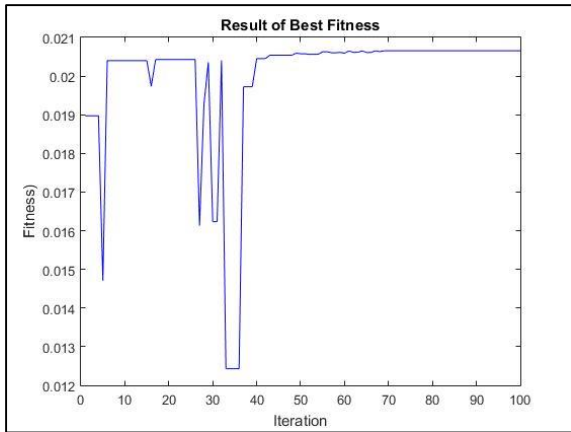
(Percobaan ke 2)



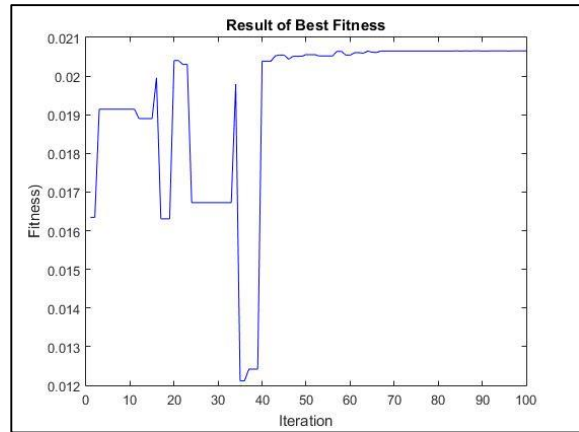
(Percobaan ke 3)



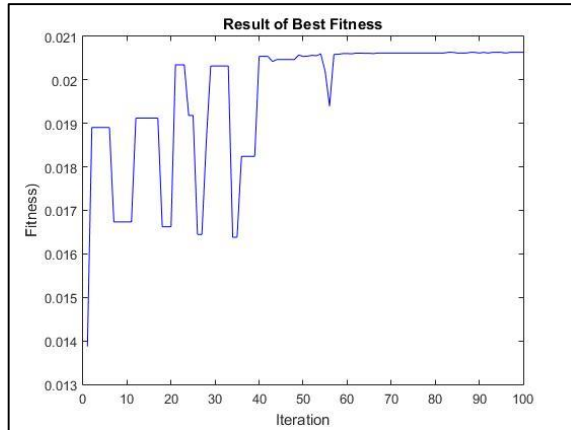
(Percobaan ke 4)



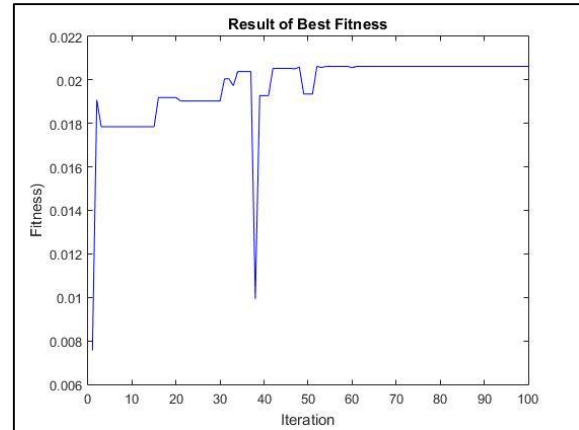
(Percobaan ke 5)



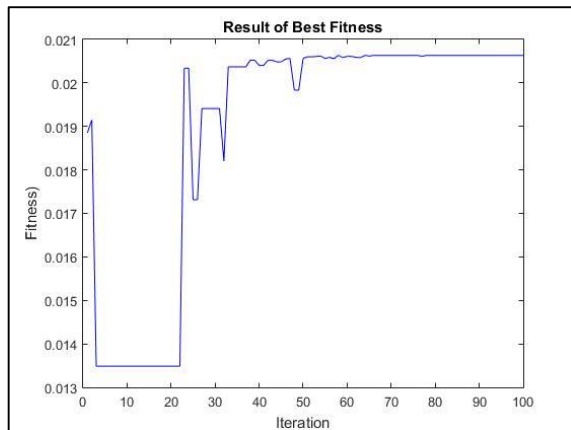
(Percobaan ke 6)



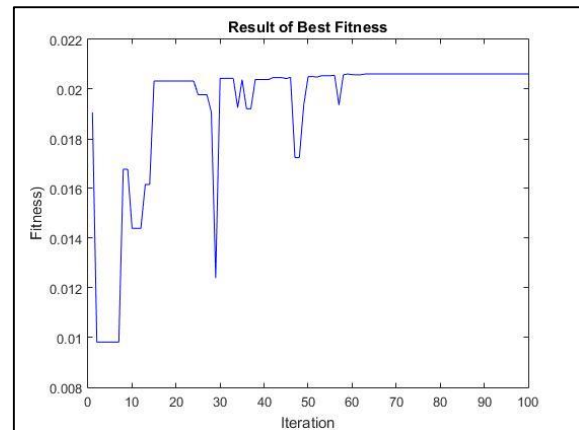
(Percobaan ke 7)



(Percobaan ke 8)



(Percobaan ke 9)



(Percobaan ke 10)