

MICROGRID ONLINE MANAGEMENT USING GENETIC ALGORITHM

¹SURMADHUR PANT, ²ARVIND SINGH BISHT

Abstract- In India renewable energy resources having a huge potential and the installed capacity is very less as compared to the potential. India is a fast growing economy in the world and the economy is directly depending upon the power sector. Power sector plays an important role in the Indian economy. If we are looking forward to improve the Indian economy then it is necessary that power sector performs very well. By using the renewable energy resources we can also reduce the pollution content. United Nation launched a various program to reduce the carbon emission and provide the different type of subsidy to enhance the use of renewable energy resources. MNRE also provides a different type of subsidy for different type user. In this Paper we discuss about the micro grid. firstly we model different micro grid combinations using different renewable energy resources. After discuss the micro grid modeling then we discuss about the online management of micro grid using Genetic Algorithm. By using these mathematics methods we can easily optimized the micro-grid.

Keywords- Microgrid, Genetic Algorithm, Multiobjective Function, Operating Cost, Maintenance Cost, Crossover Procedure, Mutation

1.Introduction- This paper develop a original intelligent technique to manage the operation of micro grid units for housing or industrialized utilization. Genetic algorithms are used to find optimal settings of the micro grid units depending on complete economic and ecological models. The objective of the genetic algorithm is to develop an intelligent management tool , which can be used for environmentally constrained economic problems of the micro grid.

- Mr.surmadhur pant has completed masters degree program in electric power engineering in University, INDIA, PH-8126532624. E-mail:sur20pant@gmail.com
- Mr.Arvind singh bisht has completed masters degree program in electric power engineering in UTU University, INDIA , PH-8449052119. E-mail: bishtarvind10@gmail.com

The problem can be classify as a multi objective optimization problem and nonlinear programming problem. The purpose of the tool is to find the optimal amount of the generated power by minimizing the operating cost and the emission level all together while gratifying the load demand and ready constraints. The trade-off curve is obtained by forming an objective function which combines the operating costs and emission level objectives. The next step is to use genetic algorithm to determine the global optimum solutions.

2.Multiobjective Function Optimization-

Multiobjective (MO) optimization is useful to the problem of the environmental of the MG. The

stated problem is solved as a nonlinear constrained MO optimization problem. Previous to the optimization part of system model from real industrial data are made. The model takes into concern the maintenance and operation cost as well as the reduction of emission of NO_x, CO₂, and SO₂. The optimization is required at reducing the function of cost of the system while constraining it to get together the demand of the customer and system safety. The results make sure the efficiency of the proposed approach to assure the load and the cost reduction and the emissions in single run. Solving the problem of environmental in the generation of power has received attention. The problems of environmental have been efficiently solved by classical technique, goal programming, and fuzzy approval maximizing approach. To execute the model the following item have to be defined by:

- The power demand by the loads.
- Information regarding locally available energy: This mainly consists solar irradiation data W/m², temperature °C, wind speed m/s, as well as cost of fuels \$/liter for the Diesel Generator and natural gas price for supplying the FC and MT \$/kWh.
- Daily purchase and sold power tariffs in \$/kWh.
- Start up costs in \$/h.
- Technical and financial performance of supply option: These characteristics consists for example rated power for PV, power curve for wind turbine(WT), fuel consumption characteristics DG and FC.

- Operating, maintenance cost and the total emissions: Operating and maintenance costs must be given (\$/h) for DG, FC, and MT.
- Emission levels must be given in kg/h for FC, DG and MT.

2.1 Proposed Objective Function- the system requirements are determined subject to:

1. Minimizing the operation costs (\$/h).
2. Minimizing the emission (kg/h).
3. Ensuring that the load is served according to given constraints.

2.2 Operating Cost-The only difference in the operating cost function is that there is no emission cost. As the problem is treated as multiobjective, the emissions are computed separately as emission level. The three different scenarios are modified as follows:

Scenario 1

The purpose of the cost function is to optimize the operation cost $CF(P)$ of the MG in \$/h as follows:

$$NCF(P) = \sum_{i=1} (C_i \times F_i(P_i) + OM_i(P_i)) \quad \dots\dots\dots 1$$

$\{P_i = PFC_i, P_j = PMT_j, P_k = PDG_k : i = 1, \dots, N1; j = N + 1, \dots, N2; k = N2 + 1, \dots, N\}$. P is N vector of the generators active power and is defined as $P = (P_1, P_2, \dots, P_N)$ The operating and maintenance costs of the generating unit OM_i .

Scenario 2

The cost function assumes the form:

$$NCF(P) = \sum_{i=1} (C_i \times F_i(P_i) + OM_i(P_i) + STC_i) \dots 5.2$$

2.3 Implementation Of The Algorithm-When designing MGs, several goals should be set include reduction in emission and generation cost. To attain this it is significant to highlight all factors influence the main goal.

- Power output of wind turbine(WT) is considered according to the relation between the output power and wind speed.
- Power output of photovoltaic(PV) is calculated according to the effects of the temperature and solar radiations that are different from the standard test conditions.
- Here the WT and PV distribute power which is free of cost that is in term of running as well the emission free, the output power is considers as a negative load ,hence the load which is difference between the real and micro source output can be calculated if the output from PV and WT is lesser than the load demand.
- The power from batteries are needed when the PV,WT are inadequate to serve up the load , for the moment the charge and discharge of the batteries are monitoring.
- Calculate the unmet load that cannot served by MT,WT and batteries storages.
- Choose serving the load by another source (FC,MT or DG) according to the objective function.
- If the output power is not sufficient then buy power from the main grid and if the output power

go beyond the load demand sell the exceed power to the main grid.

3.Multiobjective Optimization Problem-

Multiobjective optimization is method of finding out the best solution between different objective, usually conflicting objective. . A general multiobjective mathematical formulation is expressed as:

$$\min \{f_1(x), f_2(x), \dots, f_k(x)\} \dots \dots \dots 5.3$$

subject to $x \in S$

where k is the number of the objective functions f_i ($i \geq 2$), $f_i: R^n \rightarrow R$ $i = 1, \dots, k$. The vector of objective functions is denoted by $F(x) = (f_1(x), f_2(x), \dots, f_k(x))^T$. The decision variable vector $x = (x_1, x_2, \dots, x_n)^T$ belongs to a (nonempty) feasible region (set) S , which is a subset of the decision variable space R^n .

3.1 Genetic Algorithm(GA)-Genetic algorithms

(GA) and the directly linked evolutionary algorithms are a class of global non gradient method which have full-grown in popularity. GA are modeled after mechanism of natural selection. Every optimization parameter x_n is encoded by a gene using a suitable representation such as a real number or a strings of bits. The related genes for all parameter x_1, \dots, x_n form a chromosome able of describing individual design solution. A set of chromosome presenting various individual designs solution comprises a population where the fittest are chosen to reproduce. Mating are performed using crossover to combine gene from

different parent to produce children. The children are inserted into the population and procedure starts over again.

Initialization

In the initialization, one generates, often randomly, a population from which new generations are formed. At this point one also needs to define the terminating condition so that the algorithm stops running once an acceptable solution is found.

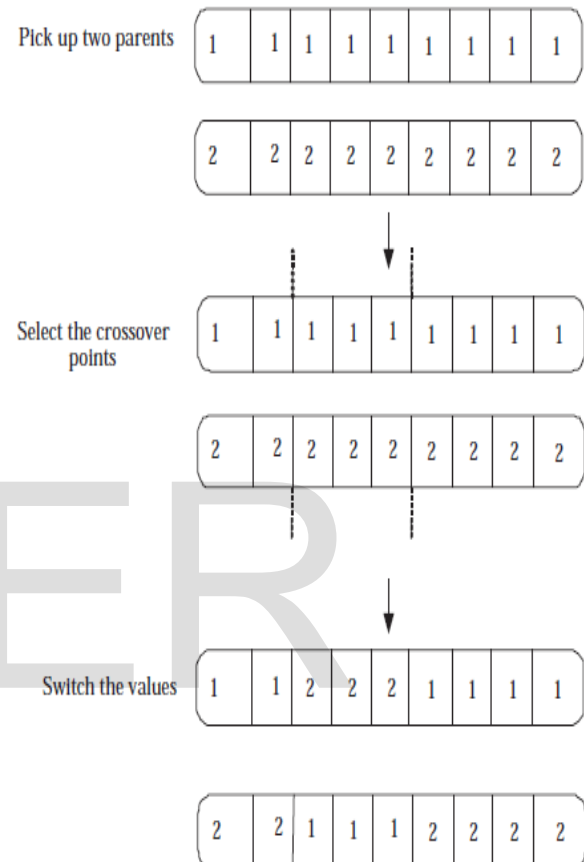
Crossover

Crossover is one of the genetic operators used in producing new candidates using the features of the existing ones. The crossover procedure consists of three parts. First one selects two parents from the population. Then the crossover points are selected. The selection of crossover points is done at random, usually so that the distribution from which the points are drawn from is uniform. In fig two crossover points are marked with dotted lines. Once the points are defined, two off springs are generated by interchanging the values between the two parents. In the genetic algorithm, crossover is the operator that spreads the favorable characteristics of the members around the population. A number of different crossover techniques have been presented in the literature.

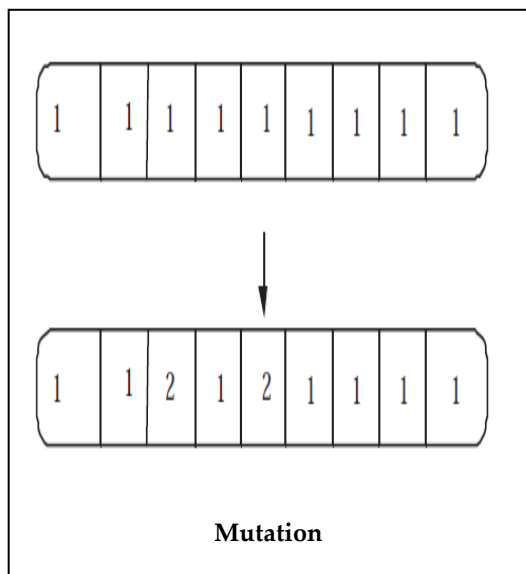
Mutation

Mutation is the operator that causes totally new characteristics to appear in the members of the population. In many cases the mutations, result in off springs that are worse than the other members, but sometimes the result has characteristics that

make it better. First, one selects a member from the population to be mutated and a point of mutation. Then the value at the point of mutation is replaced by another value that is picked randomly from the set of all possible values.



crossover procedure



Evaluation

After the population is manipulated using the genetic operators, the fitness of each new offspring is evaluated. To do this one needs to have a numerical function and a fitness function.

Selection

In the selection, the weakest individuals in the population are eliminated. Most fit offspring survive to the next generation. Population size specifies how several individuals there are in every generation. With huge population size, the genetic algorithm searches the solution space more systematically thereby minimizing the chance that the algorithm will return a local minimum that is not global minimum. However, huge population size also causes the algorithm to execute more slowly. Population size of 100 is used in this study. Mutation option identifies how the GA makes little random variation in individuals in the population to generate mutation children. Mutation gives genetic diversity and enables the genetic algorithm (GA) to search broader space. The mutation function adds an arbitrary number taken

from a Gaussian allotment with mean 0 to every entry of the parent vector. Gaussian mutation function is applied in this study. Crossover option states how the genetic algorithm (GA) combines two individuals or parents to create a crossover child for the next generation. Crossover function specifies the part of the next generation, other than elite children that produced by crossover. Set Crossover portion to be a fraction in between 0 and 1 either by entering fraction in text box or move the sliders. The value "0.8" is used. Selection options state how the genetic algorithm (GA) chooses parents for the next generation. The Tournament selection is employed in the thesis. This chooses each parent by choosing Tournament players at random and then picks the best individual out of it set to be a parent. Tournament size must be at least 2. The value of Tournament size 4 is used.

3.2. Multiobjective Genetic Algorithms

The simplest and the most clear approach to multiobjective optimizations is to unite the objective into one aggregate function and to treat the problem like a single objective optimization problem. Therefore it is most commonly used for its simplicity and computational efficiency.

The weighted sum approach combines k objectives f_i using weights, $w_i, i = 1, \dots, k$:

$$\text{fitness} = w_1 f_1(P) + w_2 f_2(P) + \dots + w_k f_k(P)$$

$w_i > 0$ and $\sum w_i = 1$.

MO optimization problem where both operation costs and emissions are optimized simultaneously. In order to see the diversity of the Pareto optimal set over the trade-off surface, the trade-off relation can be obtained by minimizing. The relationship trade-off curve between the two objectives, where the Pareto optimal set has a number of non-dominated solutions.

3.2.1 Genetic Algorithm For Microgrid

% Introduction GENETIC ALGORITHM FOR MICROGRID

% Rosenbrock's Function

% Dejong's second function:

% $f(x) = 100 \cdot (x(2) - x(1)^2)^2 + (1 - x(1))^2$

% This function is notorious in optimization because of the slow

% convergence most methods exhibit when trying to minimize this function.

% This function has a unique minimum at the point $x^* = (1,1)$ where it has a

% function value $f(x^*) = 0$;

% We can view the code for this fitness function. type `dejong2fcn.m`

% We use the function `PLOTObjective` in the toolbox to plot the

% function `DEJONG2FCN` over the range = `[-2 2;-2 2]`.

`plotobjective(@dejong2fcn,[-2 2;-2 2]);`

% Genetic Algorithm Solution

% To start, we will use the Genetic Algorithm, GA, alone to find the

% minimum of Rosenbrock's function. We need to supply GA with a function

% handle to the fitness function `dejong2fcn.m`.

Also, GA needs to know the

% how many variables are in the problem, which is two for this function.

`FitnessFcn = @dejong2fcn;`

`numberOfVariables = 2;`

% Some plot functions can be selected to monitor the performance of the

% solver.

`options = gaoptimset('PlotFcns',`

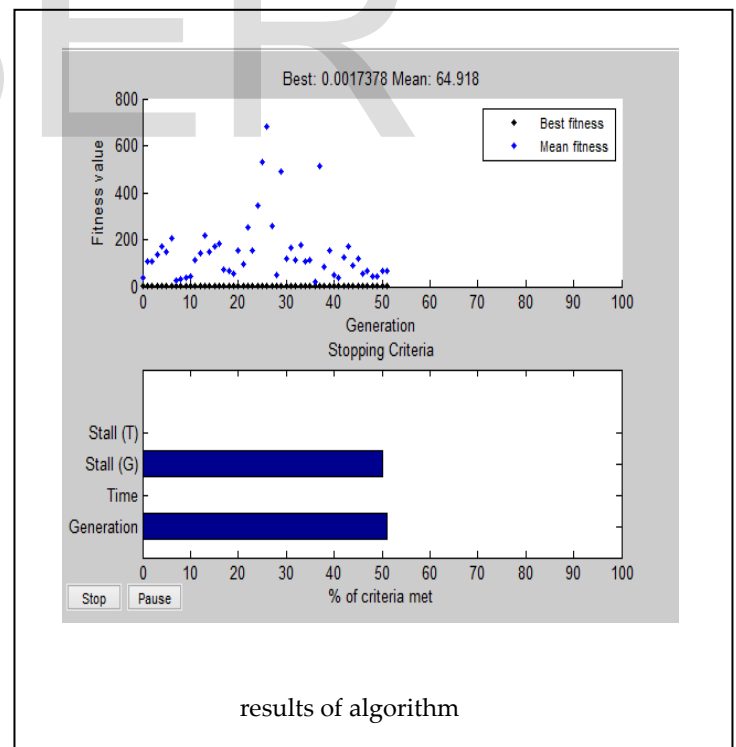
`{@gaplotbestf,@gaplotstopping});`

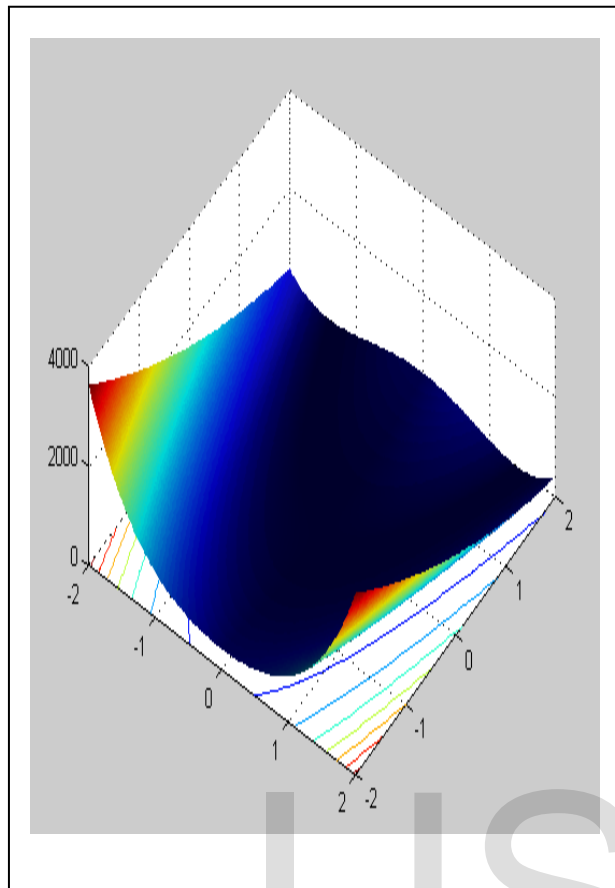
% We run GA with the above inputs.

`[x,fval] =`

`ga(FitnessFcn,numberOfVariables,[],[],[],[],[],[],o`
`ptions)`

% The global optimum is at $x^* = (1,1)$.





3.2.2 Program For Constrained Problem For Micro Grid

```
% CONSTRAINED PROBLEM
% We want to minimize a simple fitness function
of two variables x1
% and x2
% min f(x) = 100 * (x1^2 - x2) ^2 + (1 - x1)^2;
% x
% such that the following two nonlinear
constraints and bounds are
% satisfied
% x1*x2 + x1 - x2 + 1.5 <=0, (nonlinear constraint)
% 10 - x1*x2 <=0, (nonlinear constraint)
% 0 <= x1 <= 1, and (bound)
% 0 <= x2 <= 13 (bound)
% Coding the Fitness Function
```

```
% We create an M-file named simple_fitness.m
with the following
% code in it:
% function y = simple_fitness(x)
% y = 100 * (x(1)^2 - x(2)) ^2 + (1 - x(1))^2;
% The Genetic Algorithm (GA) function assumes
the fitness function will
% take one input x where x has as many elements
as number of variables in
% the problem. The fitness function computes the
value of the function and
% returns that scalar value in its one return
argument y.
% Coding the Constraint Function
% We create an M-file named simple_constraint.m
with the following
% code in it:
% function [c, ceq] = simple_constraint(x)
% c = [1.5 + x(1)*x(2) + x(1) - x(2);
% -x(1)*x(2) + 10];
% ceq = [];
% The GA function assumes the constraint function
will take one
% input x where x has as many elements as
number of variables in the
% problem. The constraint function computes the
values of all the
% inequality and equality constraints and returns
two vectors c and ceq
% respectively.
ObjectiveFunction = @simple_fitness;
nvars = 2; % Number of variables
LB = [0 0]; % Lower bound
UB = [1 13]; % Upper bound
```



```

ConstraintFunction = @simple_constraint;

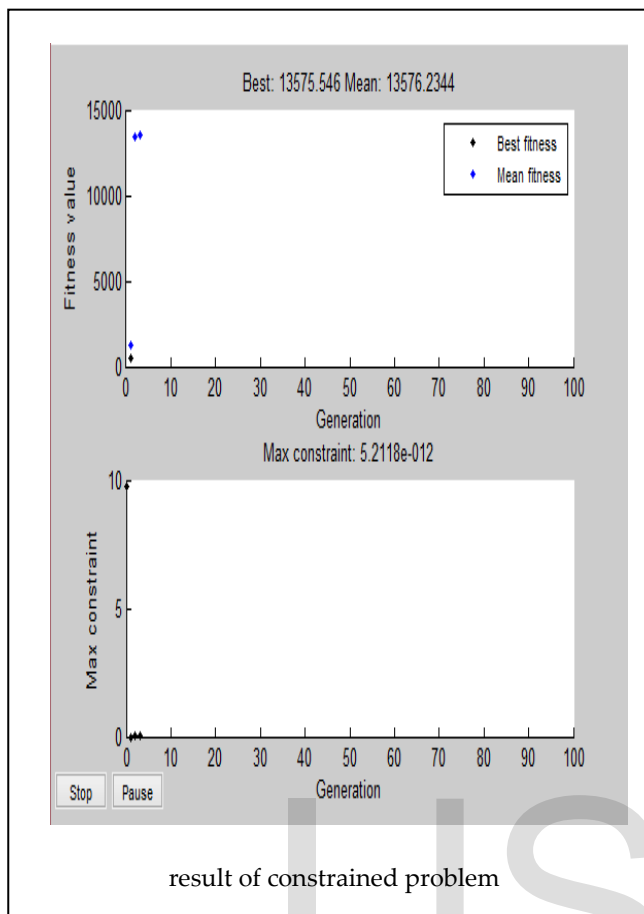
[x,fval] =
ga(ObjectiveFunction,nvars,[],[],[],LB,UB, ...
    ConstraintFunction)
% We specify MUTATIONADAPTFEASIBLE as
the mutation function for our
% minimization problem by using GAOPTIMSET
function.
options =
gaoptimset('MutationFcn',@mutationadaptfeasible)
;
% Next we run the GA solver.
[x,fval] =
ga(ObjectiveFunction,nvars,[],[],[],LB,UB, ...
    ConstraintFunction,options)
% Adding Visualization
% Next we use GAOPTIMSET to create an options
structure to select two plot
% functions. The first plot function is
GAPLOTBESTF, which plots the
% best and mean score of the population at every
generation. The second
% plot function is GAPLOTMAXCONSTR, which
plots the maximum constraint
% violation of nonlinear constraints at every
generation. We can also
% visualize the progress of the algorithm by
displaying information to
% the command window using the 'Display'
option.
options =
gaoptimset(options,'PlotFcns',{'gaplotbestf','gaplo
tmaxconstr'}, ...
    'Display','iter');

```

```

% Next we run the GA solver.
[x,fval] =
ga(ObjectiveFunction,nvars,[],[],[],LB,UB, ...
    ConstraintFunction,options)
% Providing a Start Point
% A start point for the minimization can be
provided to GA function by
% specifying the InitialPopulation option. The GA
function will use the
% first individual from InitialPopulation as a start
point for a
% constrained minimization. Refer to the
documentation for a description of
% specifying an initial population to GA.
X0 = [0.5 0.5]; % Start point (row vector)
options =
gaoptimset(options,'InitialPopulation',X0);
% Next we run the GA solver.
[x,fval] =
ga(ObjectiveFunction,nvars,[],[],[],LB,UB, ...
    ConstraintFunction,options)

```

Photovoltaic And Battery Based Power Generation".

[4] Swami Prakash Srivastava Surat Prakash Srivastava," Solar Energy And Its Future Role In Indian Economy",International Journal of Environmental Science: Development and Monitoring (IJESDM) ISSN No. 2231-1289, Volume 4 No. 3 (2013).

[5] Frank S. Young," Flexible Ac Transmission Systems: Technology For The Future", Ch2991-8/91/0000-0216 \$1.00 0 1991 Ieee.

[6] J.I. San Martín, I. Zamora 2, J.J. San Martín, V. Aperribay, P. Eguía," Hybrid Technologies: Fuel Cells and Renewable Energies".

Reference-[1] M. Abdulkadir, A. S. Samosir and A. H. M. Yatim, "Modeling And Simulation Based Approach Of Photovoltaic System In Simulink Model",Arpn Journal of Engineering and Applied Sciences.

[2] Harini M., Ramaprabha R. and Mathur B. L.," Modeling Of Grid Connected Hybrid Wind/Pv Generation System Using MATLAB",ARPJ Journal of Engineering and Applied Sciences.

[3] Alias Khamis, Mohamed, H. Shareef, "Modeling And Simulation Of A Microgrid Testbed Using

IJSER