Stochastic Energy Management Scheduling System For Microgrid

G.Iswaryalakshmi, M.E. Student, Ms. S.Shenbagaramalakshmi M.E.,
P.S.R.RengasamyCollege of Engineering for Women, Sivakasi,
VirudhunagarDist, Tamil Nadu 626140

ABSTRACT

Due to environmental concerns and ever-increasing fuel costs, governments offer incentives for clean and sustainable energy production from Distributed Generations (DGs) such as Wind Turbine (WT) and Photovoltaic (PV) generators. Optimal operation of Microgrids (MGs) and management of demand side are necessary to increase the efficiency and reliability of distribution networks. In this project, the stochastic operation scheduling of a MG consisting of non-dispatchable resources including WT and PV and dispatchable resources including Phosphoric Acid Fuel Cell (PAFC), and electrical storage as Battery Energy Storage System (BESS) is investigated to minimize operation cost and emissions. In this work, to propose a well-known technique called 'Multi objective genetic algorithm (MOGA)' algorithm. With this intelligent control method, it is possible to achieve minimum operating cost and also possible to manage the load variability. The proposed work is implemented on MATLAB R2014a software with the real time data collected for solar and wind power systems. The results are showing the effectiveness of the proposed optimization technique.

I. INTRODUCTION

Recently, the awareness of energy, economic, and environmental challenges like growing demand, inadequacy of fossil fuels in future, and emission pollutants has been increased across the world. Integration of small-scale DGs, mostly based on renewable resources, near consumers has introduced MGs a promising solution for environmental/economic challenges. MGs can result in higher efficiency, reduced losses, and environmental benefits due to using renewable energy sources such as wind and solar. If the variability of DG powers is successfully mitigated using different technologies such as energy storage, MGs also offer acceptable power quality. If an efficient EMS is properly developed for an MG, reliability can also be improved through MG architecture especially at times of events and peak demand hours. In addition, EMS reduces MG operational cost and optimizes energy usage by exchanging power from/to the main grid depending on generations and demands under a suitable market policy. Furthermore, EMS of MG determines optimal scheduling of DGs and supplies demands using BESSs to manage uncertainty of DGs. Literature studies have focused on different aspects of MG energy scheduling. Some researches consider single-objective optimizations mostly minimizing operation costs of MGs by economic UC of generation units, while some others have concentrated on environmental/economic energy management in MGs.

II. LITERATURE SURVEY

Farzanet. al., propose atypical microgrid portfolio includes photovoltaic (PV) and wind resources, gas-fired generation, demand-response capabilities, electrical and thermal storage, combined heat and power (CHP), and connectivity to the grid. Advanced technologies such as fuel cells may also be included. This article describes the problems encountered in analyzing prospective microgrid economics and environmental and reliability performance and presents some results from the software tools developed for these tasks.

Mostafaet. al., implementing the smart grid, electric energy consumption, generation resources, energy storage, plug-in electric vehicles , should be managed and optimized in a way that saves energy, improves efficiency, enhance reliability and maintain security while meeting the increasing demand at minimum operating cost. As a consequence, energy management systems are receiving more attention from the researchers and utilities. Accordingly, the main concern of this work is to investigate different energy management systems whether owned by a customer or by the distribution system utility.

Wang et. al., implement the renewable energy resources such as wind and solar are an important component of a microgrid. However, the inherent intermittency and variability of such resources complicates microgrid operations. Meanwhile, more controllable loads (e.g., plug-in electric vehicles), distributed generators (e.g., micro gas turbines and diesel generators), and distributed energy storage devices (e.g., battery banks) are being integrated into the microgrid operation. To address the operational challenges associated with these technologies and energy resources, this paper formulates a stochastic problem for microgrid energy scheduling.

Fathimaet. al.,trives to bring to light the concept of Hybrid Renewable Energy Systems (HRES) and state of art application of optimization tools and techniques to microgrids, integrating renewable energies. With an extensive literature survey on HRES, a framework of diverse objectives has been outlined for which optimization approaches were applied to empower the microgrid.

III - SYSTEM IMPLEMENTATION

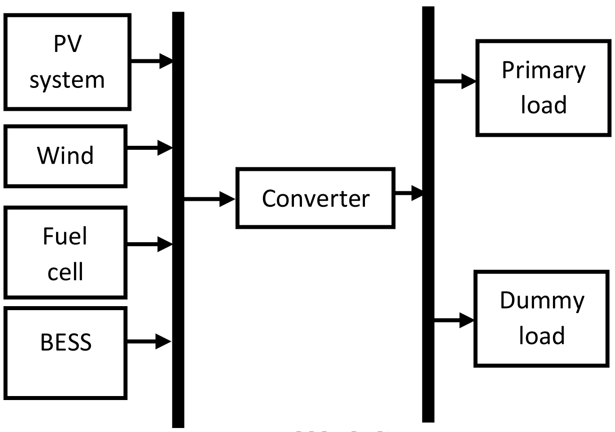
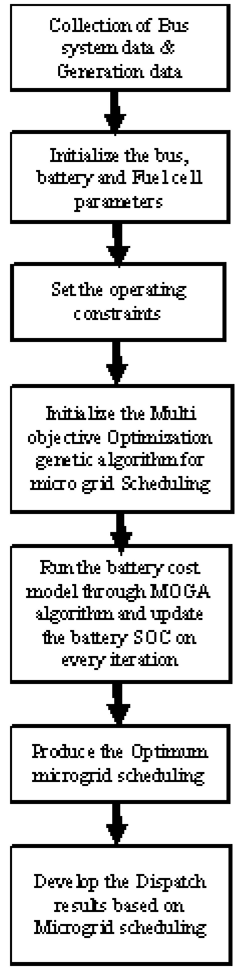


Fig 1 proposed block diagram

In the present work, the MG energy management is applied while taking into account several types of DERs in order to supply electrical loads at minimum operation cost and emissions. The present study compared to other studies has used various technologies of DERs in the MG energy management. The aforementioned resources are composed of WT, PV, MT, PAFC, and BESS. With respect, the majority of studies have not addressed joint generation reserve scheduling and DR; nonetheless, in this study, these features are simultaneously considered to efficiently manage the MG. Also, a stochastic MOP is formulated to simultaneously minimize the operational cost and emissions of MG. The MOP is converted to SOP using the weighted sum method whose coefficients is determined by a fuzzy operator. The uncertainty of electrical demand, wind speed, and solar radiation is simultaneously taken into account using the PDF of uncertain parameters which generates a set of scenarios. Then, a scenario reduction method based on the DE optimization is used to reduce the number of scenarios. Finally, these stochastic SOP is solved by the aforesaid hybrid algorithm.



In the first block the bus system data are collected. In the second block to initialize the parameters like bus, battery and fuel cell. Then to set operating constraints. In here the operating constraints are Fuel start-up costs, Fuel shut-down costs, Minimum on/off time, Generation capacity constraints, Ramping capacity constraints, Power balance constraints, Transmission capacity. From the operating constraints set the initial parameter for evolutionary algorithm. Here the initial parameters are Max no of iterations, Population size, Gravitational constant, Coriolis constant, Maximum allowed speed and RT constant. Then run the MOGA. Run the battery cost model through MOGA algorithm and update SOC on every iteration. The cost comparison of flexible and predefined sets are obtained in this block. At the final block develop the dispatch results based on MC.

A. Multi-objective Genetic Algorithms

Being a population based approach, GA are well suited to solve multi-objective optimization problems. A generic single-objective GA can be easily modified to find a set of multiple non-dominated solutions in a single run. The ability of GA to simultaneously search different regions of a solution space makes it possible to find a diverse set of solutions for difficult problems with non-convex, discontinuous, and multi-modal solutions spaces. The crossover operator of GA may exploit structures of good solutions with respect to different objectives to create new non-dominated solutions in unexplored parts of the Pareto front. In addition, most multi-objective GA does not require the user to prioritize, scale, or weigh objectives. Therefore, GA has been the most popular heuristic approach to multi-objective design and optimization problems.

a). Fitness Functions

1. Weighted Sum Approaches.

The classical approach to solve a multi-objective optimization problem is to assign a weight $w_i$to each normalized objective function ( ) z_i' **x** so that the problem is converted to a single objective problem with a scalar objective function as follows:

$$\min z = w_1 z_1'(\mathbf{x}) + w_2 z_2'(\mathbf{x}) + \ldots + w_k z_k'(\mathbf{x}) \qquad \ldots..(1)$$

where z_i' (**x**) is the normalized objective function, z_i' (**x**) and 1 $\Sigma w_i = 1$. This approach is *called a priori* approach since the user is expected to provide the weights. Solving a problem with the objective function (1) for a given weight vector $\mathbf{w} = \{w_1\ w_2 \ldots\ w_k\}$ yields a single solution, and if multiple solutions are desired, the problem should be solved multiple times with different weight combinations. The main difficulty with this approach is selecting a weight vector for each run. To automate this process, the weight-based genetic algorithm for multi-objective optimization (WBGA-MO). In the WBGA-MO, each solution $\mathbf{x_i}$in the population uses a different weight vector $\mathbf{w} = \{w_1\ w_2 \ldots\ w_k\}$ in the calculation of objective function (1). The weight vector $\mathbf{w_i}$is embedded within the chromosome of solution $\mathbf{x_i}$.

Therefore, multiple solutions can be simultaneously searched in a single run. In addition, weight vectors can be adjusted to promote diversity of the population.

Other researchers have proposed a multi-objective genetic algorithm based on a weighted sum of multiple objective functions where a normalized weight vector $\mathbf{w_i}$is randomly generated for each solution $\mathbf{x_i}$during the selection phase at each generation. This approach aims to stipulate multiple search directions in a single run without using any additional parameters.

The main advantage of the weighted sum approach is a straightforward implementation. Since a single objective is used in fitness assignment, a single objective GA can be used with minimum modifications. In addition, this approach is computationally very efficient. The main disadvantage of this approach is that not all Pareto-optimal solutions can be investigated when the true Pareto front is non-convex. Therefore, the multi-objective genetic algorithms based on the weighed sum approach have difficulty in finding solutions uniformly distributed over a nonconvex trade-off surface.

2. ***Altering Objective Functions.***

As mentioned earlier, the VEGA is the first GA used to approximate the Pareto optimal set by a set of non-dominated solutions. In the VEGA, population $P_t$is randomly divided into K equal sized sub-populations; P_1, P_2, ...,P_K. Then, each solution in subpopulation $P_i$is assigned a fitness value based on objective function z_i. Solutions are selected from these subpopulations using proportional selection for crossover and mutation. Crossover and mutation are performed on the new population in the same way with the single objective GA. A similar approach is to use only a single objective function which is randomly determined each time in the selection phase.

These approaches are easy to implement and computationally as efficient as a single objective GA. The major drawback of objective switching is that the population tends to converge to solutions which are very superior in one objective, but very poor at others.

3. Pareto-Ranking Approaches.

Pareto-ranking approaches explicitly utilize the concept of Pareto dominance in evaluating fitness or assigning selection probability to solutions. The population is ranked according to a dominance rule, and then each solution is assigned a fitness value based on its rank in the population, not its actual objective function value. Note that herein all objectives are assumed to be minimized. Therefore, a lower rank corresponds to a better solution in the following discussions.

The first Pareto ranking technique was proposed as follows:

Step 1. Set i=1 and TP=P

Step 2. Identify non-dominated solutions in TP and assigned them set to F_i.

Step 3. Set $TP = TP \setminus F_i$. If TP=∅ go to Step 4, else set i=i+1 and go to Step 2.

Step 4. For every solution $\mathbf{x} \in P$ at generation t, assign rank $r1(\mathbf{x},t) = i$if $\mathbf{x} \in F_i$.

In the procedure above, F_1, F_2, ... are called non-dominated fronts, and $F1$ is the Pareto front of population P. Fonseca and Fleming used a slightly different rank assignment approach follows:

$$r_2(\mathbf{x},t) = 1 + nq(\mathbf{x},t)$$

where$nq(\mathbf{x},t)$ is the number of solutions dominating solution **x** at generation t. This ranking method penalizes solutions located in the regions of the objective function space which are dominated (covered) by densely populated sections of the Pareto front. For example, in Figure 1b solution **i**is dominated by solutions **c, d** and **e.** Therefore, it is assigned a rank of 4 although it is in the same front with solutions **f, g** and **h** which are dominated by only a single solution.

The SPEA uses a ranking procedure to assign better fitness values to non-dominated solutions at underrepresented regions of the objective space. In the SPEA, an external list *E* of a fixed size stores non-dominated solutions that have been investigated thus far during the search. For each solution $\mathbf{y}\in E$, a strength value is defined as,

$$s(\mathbf{y},t)=\frac{np(\mathbf{y},t)}{N_P+1}$$

where$np(\mathbf{y},t)$ is the number solutions that **y** dominates in *P*. The rank $r(\mathbf{y},t)$ of a solution $\mathbf{y}\in E$is assigned as $3r(\mathbf{y},t)=s(\mathbf{y},t)$ and the rank of a solution $\mathbf{x}\in P$is calculated as,

$$r_3(\mathbf{x},t)=1+\sum_{\mathbf{y}\in E,\mathbf{y}\succ\mathbf{x}} s(\mathbf{y},t)$$

Figure 1c illustrates an example of the SPEA ranking method. In the former two methods, all non-dominated solutions are assigned a rank of 1. This method, however, favors solution **a** (in the figure) over the other non-dominated solutions since it covers the least number of solutions in the objective function space. Therefore, a wide, uniformly distributed set of non-dominated solutions is encouraged.

Accumulated ranking density strategy also aims to penalize redundancy in the population due to overrepresentation. This ranking method is given as,

$$r_4(\mathbf{x},t)=1+\sum_{\mathbf{y}\in P,\mathbf{y}\succ\mathbf{x}} r(\mathbf{y},t)$$

To calculate the rank of a solution **x**, the rank of the solutions dominating this solution must be calculated first. Figure 1d shows an example of this ranking method (based on r_2). Using ranking method r_4, solutions **i**, **l** and **n** are ranked higher than their counterparts at the same non dominated front since the portion of the trade-off surface covering them is crowded by three nearby solutions **c, d** and **e**.

b). Diversity: Fitness Assignment, Fitness Sharing, and Niching.

Maintaining a diverse population is an important consideration in multi-objective GA to obtain solutions uniformly distributed over the true Pareto font. Without taking any preventive measures, the population tends to form relatively few clusters in multi-objective GA. This phenomenon is called genetic drift, and several approaches are used to prevent genetic drift, as follows.

1. *Fitness Sharing*

Fitness sharing aims to encourage the search in unexplored sections of a Pareto front by artificially reducing fitness of solutions in densely populated areas. To achieve this goal, densely populated areas are identified and a fair penalty method is used to penalize the solutions located in such areas.

Step 1. Calculate the Euclidean distance between every solution pair **x** and **y** in the normalized objective space between 0 and 1 as

$$dz(\mathbf{x},\mathbf{y})=\sqrt{\sum_{k=1}^{K}\left(\frac{z_k(\mathbf{x})-z_k(\mathbf{y})}{z_k^{\max}-z_k^{\min}}\right)^2}$$

Where $z_k^{\max}$ and $z_k^{\min}$ are the maximum and minimum value of the objective function $z_k(\cdot)$ observed so far during the search, respectively.

Step 2. Based on these distances, calculate a niche count for each solution $\mathbf{x}\in P$as

$$nc(\mathbf{x},t)=\sum_{\substack{\mathbf{y}\in P\\ r(\mathbf{y},t)=r(\mathbf{x},t)}}\max\left\{\frac{\sigma_{\text{share}}-d(\mathbf{x},\mathbf{y})}{\sigma_{\text{share}}},0\right\}$$

where$\sigma_{\text{share}}$is the niche size.

Step 3. After calculating niche counts, the fitness of each solution is adjusted as follows:

$$f'(\mathbf{x},t)=\frac{f(\mathbf{x},t)}{nc(\mathbf{x},t)}$$

In the procedure above, σ_{share} defines a neighborhood of solutions in the objective space (Figure 1a). The solutions in the same neighborhood contribute to each other's niche count.

Therefore, a solution in a crowded neighborhood will have a higher niche count reducing the probability of selecting that solution as a parent. As a result, niching limits the proliferation of solutions in one particular neighborhood of the objective function space.

Another alternative is to use the Hamming distance (the distance in the decision variable space) between two solutions **x** and **y** which is defined as

$$dx(\mathbf{x},\mathbf{y})=\sqrt{\frac{1}{M}\sum_{i=1}^{M}(x_i-y_i)^2}$$

in the calculation of niche count. Equation (3) is a measure of structural differences between two solutions. Two solutions might be very close in the objective function space while they have very different structural features. Therefore, fitness sharing based on the objective function space may reduce diversity in the decision variable space.

One of the disadvantages of the fitness sharing based on niche count is that the user has to select a new parameter σ_{share}. To address this problem, Deb and Goldberg and Fonseca and Fleming developed systematic approaches to estimate and dynamically update σ_{share}.

Another disadvantage of niching is computational effort to calculate niche counts. However, benefits of fitness sharing surpass the burden of extra computational effort in many applications.

2. *Crowding Distance*

Crowding distance approaches aim to obtain a uniform spread of solutions along the best

known Pareto front without using a fitness sharing parameter. For example, the NSGA-II [9] use a crowding distance method as follows (Figure 2b):

Step 1. Rank the population and identify non-dominated fronts F_1, F_2, ...,F_R. For each front j=1, ...,R repeat Steps 2 and 3.

Step 2. For each objective function k, sort the solutions in Fjin the ascending order. Let $l=|F_j|$ and $\mathbf{x}_{[i,k]}$ represent the ith solution in the sorted list with respect to the objective function k. Assign [1,] $cd_k[\mathbf{x}_{[i,k]}]=\infty$ and $cd_k[\mathbf{x}_{[i,k]}]=\infty$, and for i=2, ..., l assign

$$cd_k(\mathbf{x}_{[i,k]}) = \frac{z_k(\mathbf{x}_{[i+1,k]}) - z_k(\mathbf{x}^k_{[i-1,k]})}{z_k^{\max} - z_k^{\min}}$$

Step 3. To find the total crowding distance $cd(\mathbf{x})$ of a solution **x,** sum the solution crowding distances with respect to each objective, i.e.,

$$cd(\mathbf{x}) = \sum_k cd_k(\mathbf{x})$$

The main advantage of the crowding approach described above is that a measure of population density around a solution is computed without requiring a user-defined parameter. In the NSGA-II, this crowding distance measure is used as a tie-breaker as in the selection phase that follows. Randomly select two solutions **x** and **y**; if the solutions are in the same non dominated front, the solution with a higher crowding distance wins. Otherwise, the solution with the lowest rank is selected.

3. Cell-Based Density

The objective space is divided into K-dimensional cells (see Figure 2c). The number of solutions in each cell is defined as the density of the cell, and the density of a solution is equal to the density of the cell in which the solution is located. This density information is used to achieve diversity similarly to the fitness sharing approach.

The main advantage of the cell based density approach is that a global density map of the objective function space is obtained as a result of the density calculation. The search can be encouraged toward sparsely inhabited regions of the objective function space based on this map.

c). Elitisim

Elitism in the context of single-objective GA means that the best solution found so far during the search has immunity against selection and always survives in the next generation. In this respect, all non-dominated solutions discovered by a multi-objective GA are considered as elite solutions. However, implementation of elitism in multi-objective optimization is not as straightforward as in single objective optimization mainly due to the large number of possible elitist solutions. Earlier multi-objective GA did not use elitism. However, most recent multiobjective GA and their variations use elitism. Multi-objective GA using elitist strategies tend to outperform their non-elitist counterparts. Multi-objective GA uses two strategies to implement elitism: (i) maintaining elitist solutions in the population, and (ii) storing elitist solutions in an external secondary list and reintroducing them to the population.

1. Strategies to Maintain Elitist Solutions in the Population

Random selection does not ensure that a non-dominated solution will survive in the next generation. A straightforward implementation of elitism in a multi-objective GA is to copy all non-dominated solution in population $P_t$to population P_{t+1}, then fill the rest of P_{t+1} by selecting from the remaining dominated solutions in Pt. This approach will not work when the total number of non-dominated parent and offspring solutions is larger than NP. To address this problem, several approaches have been proposed.

In this multi-objective GA, the population includes only non dominated solutions. If the size of the population reaches an upper bound N_{max}, N_{max}-N_{min} solutions are removed from the population giving consideration to maintaining the diversity of the current non-dominated front. To achieve this, the Pareto domination tournament selection is used as follows. Two solutions are randomly chosen and the solution with the higher niche count is removed since all solutions are non-dominated. A similar pure elitist multi-objective GA with a dynamic population size.

The NSGA-II uses a fixed population size of N. In generation t, an offspring population $Qt$of size N is created from parent population $Pt$and non-dominated fronts $F1$, $F2$, ...,FR are identified in the combined population $Pt \cup Q$t. The next population $Pt+1$ is filled starting from solutions in $F1$, then $F2$, and so on as follows. Let k be the index of a non-dominated front F_k that $|F_1 \cup F_2 \cup ... \cup F_k| \leq N$ and $|F_1 \cup F_2 \cup ... \cup F_k \cup F_{k+1}| > N$. First, all solutions in fronts $F_1, F_2, ..., F_k$ are copied to P_{t+1}, and then the least crowded (N-$|P_{t+1}|$) solutions in F_{k+1} are added to P_{t+1}. This approach makes sure that all non-dominated solutions (F_1) are included in the next population if $|F_1| \leq N$, and otherwise the selection based on a crowding distance will promote diversity.

2. Elitism with External Populations

When an external list is used to store elitist solutions, several issues must be addressed. The first issue is which solutions are going to be stored in elitist list E. Most multi-objective GA store non-dominated solutions investigated so far during the search, and E is updated each time a new solution is created by removing elitist solutions dominated by the new solution or adding the new solution if it is not dominated by any existing elitist solution. This is a computationally expensive operation. Several data structures were proposed to efficiently store, update, and search in list E. Another issue is the size of list E. Since there might possibly exist a very large number of Pareto optimal solutions for a problem, the elitist list can grow extremely large. Therefore, pruning techniques were proposed to control the size of E. For example, the SPEA uses the average linkage clustering method, to reduce the size of E to an upper limit N when the number of the non-dominated solutions exceeds N as follows.

Step 1. Initially, assign each solution $\mathbf{x} \in E$ to a cluster c_i, $C=\{c_1,c_2,\ldots,c_M\}$

Step 2. Calculate the distance between all pairs of clusters c_i and c_j as follows

$$d_{c_i,c_j} = \frac{1}{|c_i| \cdot |c_j|} \sum_{\mathbf{x} \in c_i, \mathbf{y} \in c_j} d(\mathbf{x}, \mathbf{y})$$

Here, the distance $d(\mathbf{x},\mathbf{y})$ can be calculated in the objective function space using equation (2) or in the decision variable space using equation (3).

Step 3. Merge the cluster pair c_i and c_j with the minimum distance among all clusters into a new cluster.

Step 4. If $|C| \leq N$, go to Step 5, else go to Step 2.

Step 5. For each cluster, determine a solution with the minimum average distance to all other solutions in the same cluster (called centroid solution). Keep the centroid solutions for every cluster and remove other solutions from E.

The final issue is the selection of elitist solutions from E to be reintroduced to the population. In, solutions for P_{t+1} are selected from the combined population of P_t and E_t. To implement this strategy, population P_t and E_t are combined together, a fitness value is assigned to each solution in the combined population $P_t \cup E_t$, and then, N solutions are selected for the next generation P_{t+1} based on the assigned fitness values. Another strategy is to reserve a room for n elitist solutions in the next population. In this strategy, N - n solutions are selected from parents and newly created offspring and n solutions are selected from E_t.

d). Constraint Handling

Most real-world optimization problems include constraints that must be satisfied. Single objective GA use four different constraint handling strategy: (i) discarding infeasible solutions, (ii) reducing the fitness of infeasible solutions by using a penalty function, (iii) if possible, customizing genetic operators to always produce feasible solutions, and (iv) repairing infeasiblesolutions. Handling of constraints has not been adequately researched for multi-objective GA. For instance, all major multi-objective GA assumed problems without any constraints. While constraint handling strategies (i), (iii), and (iv) are directly applicable in the multiobjective case, implementation of penalty function strategies, which is by far the most frequently used constraint handling strategy in single-objective GA, is not straightforward in multiobjective GA, mainly due to fact that fitness assignment is usually based on the non-dominance rank of a solution, not on its objective function values.

To address infeasibility in multiobjective problems as follows:

Step 1. Randomly chose two solutions $\mathbf{x}$ and $\mathbf{y}$ from the population.

Step 2. If one of the solutions is feasible and the other one is infeasible, the winner is the feasible solution, and stop. Otherwise, if both solutions are infeasible go to Step 3, else go to step 4.

Step 3. In this case, solutions $\mathbf{x}$ and $\mathbf{y}$ are both infeasible. Then, select a random reference set C among infeasible solutions in the population. Compare solutions $\mathbf{x}$ and $\mathbf{y}$ to the solutions in reference set C with respect to their degree of infeasibility. In order to achieve this, calculate a measure of infeasibility (e.g., the number of constraints violated or total constraint violation) for solutions $\mathbf{x}$, $\mathbf{y}$, and in set C. If one of solutions $\mathbf{x}$ and $\mathbf{y}$ is better and the other one is worse than the best solution in C, with respect to the calculated infeasibility measure, then the winner is the least infeasible solution. However, if there is a tie, that is both solutions $\mathbf{x}$ and $\mathbf{y}$ are either better or worse than the best solution in C, then their niche counts in the decision variable space (equation (3)) is used for selection. In this case, the solution with the lower niche count is the winner.

Step 4. In this case, solutions $\mathbf{x}$ and $\mathbf{y}$ are both feasible. Then, select a random reference set C among feasible solutions in the population. Compare solutions **x** and **y** to the solutions in set *C*. If one of them is non-dominated in set *C*, and the other is dominated by at least one solution, the winner is the former. Otherwise, there is a tie between solutions **x** and **y**, and the niche count of the solutions are calculated in the decision variable space. The solution with the smaller niche count is the winner of the tournament selection.

The procedure above is a comprehensive approach to deal with infeasibility while maintaining diversity and dominance of the population. Main disadvantages of this procedure are its computational complexity and additional parameters such as the size of reference set *C* and niche size. Modifications are also possible. In Step 4, for example, the niche count of the solutions can be calculated in the objective function space instead of the decision variable space. In Step 3, the solution with the least infeasibility can be declared as the winner without comparing solutions **x** and **y** to a reference set *C* with respect to infeasibility. Such modifications can reduce the computational complexity of the procedure.

The constrain-domination concept and a binary tournament selection method based on it, called a constrained tournament method. A solution **x** is said to constrain dominate a solution **y** if either of the following cases are satisfied:

Case 1: Solution **x** is feasible and solution **y** is infeasible.

Case 2: Solutions **x** and **y** are both infeasible; however, solution **x** has a smaller constraint violation than **y.**

Case 3: Solutions **x** and **y** are both feasible, and solution **x** dominates solution **y**.

In the constraint tournament method, first non-constrain-dominance fronts *F*1, *F*2, *F*3,....,*FR* are identified in a similar way defined, but by using the constrain-domination criterion instead of the regular domination concept. Note that set *F*1 corresponds to the set of feasible non dominated solutions in the population and front *Fi* is more preferred than *Fj*for *i*<*j*. In the constraint tournament selection, two solutions **x** and **y** are randomly chosen from the population. Between **x** and **y**, the winner is the one in a more preferred non-constrain-dominance front. If solutions **x** and **y** are both in the same front, then the winner is decided based on niche counts or crowding distances of the solution. The main advantages of the constrained tournament method are that it requires fewer parameters and it can be easily integrated to multi-objective GA.

IV – SIMULATION RESULTS

A 6 bus multi- machine system is taken here for the analysis purpose as shown in the figure. It consists of 6 buses, 3 feeders, 1 diesel generator, 2 wind generators, 1 PV generator and 3 battery storage systems 1 transformer and 15 loads are connected on a 13.8kV main grid. The length of each cable is 50 km and positive, zero sequence component of impedance is (0.015240+j 0.027432) ohms per conductor per phase. The rating of generators and battery are given below in the following tables. The renewable energy source and batteries are having the rating as like in the table 4.1

Table 1 Generation and load specifications

S.No.	Details	Power
1	Diesel Generator	50kW
2	Solar plant	50kW
3	Wind plant 1	20kW
4	Wind plant 2	20kW
5	VRB battery	10kW
6	AGM battery	12kW
7	Maximum load	50kW

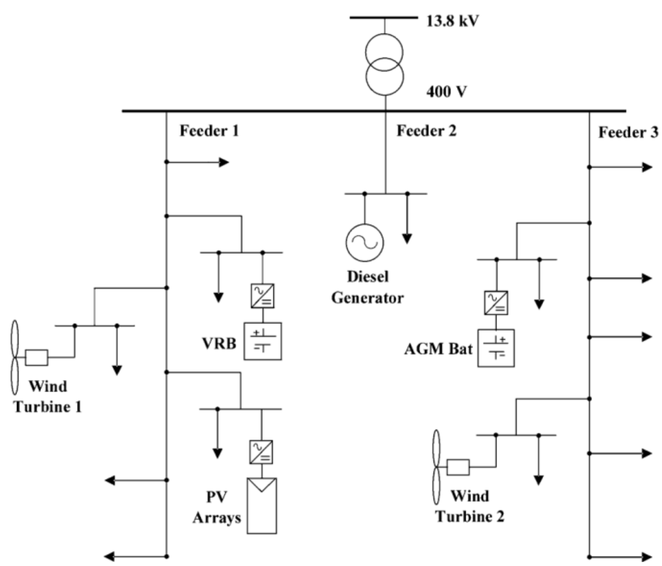


Figure 2 Single line diagram of 6 Bus system

Table 2 Diesel Generator data

a_{dg}	b_{dg}	c_{dg}	Start-up	c_{gen}
$3.10^{-4}\frac{gal/h}{kW^2}$	$0.052\frac{gal/h}{kW}$	$0.8gal/h$	$1	$4/gal
$T_{dg}^{up,min}$	$T_{dg}^{dw,min}$	P_{dg}^{min}	P_{dg}^{max}	Ini. state
$2hr$	$2hr$	$5kW$	$50kW$	$1hr$

Table 3 Battery data

	L_r	DOD_r	C_r	V_r
VRB	10000 cycles	30%	$40kWh$	$60V$
AGM	1000 cycles	50%	$30kWh$	$60V$

	SOC_{min}	SOC_{max}	Ini. SOC	kWh_f price
VRB	0.3	0.8	0.5	$0.1\$/kWh_f$
AGM	0.5	1	0.5	$0.59\$/kWh_f$

Table 4 Wind and Solar power generation data

Hour	Wind Power1 (kW)	Wind Power2 (kW)	Solar Power (kW)
1.00	2.31	16.95	0.00
2.00	3.36	15.70	0.00
3.00	3.80	15.10	0.00
4.00	5.00	14.00	0.00
5.00	5.57	9.00	1.81
6.00	7.33	7.35	8.81
7.00	8.91	6.05	20.07
8.00	10.07	5.70	31.09
9.00	10.86	4.50	40.05
10.00	10.93	4.00	46.08
11.00	13.36	3.65	49.03
12.00	16.90	3.45	48.77
13.00	17.47	3.70	46.05
14.00	19.83	3.25	41.06
15.00	21.43	2.70	33.33
16.00	21.76	3.40	23.82
17.00	18.76	3.45	0.00
18.00	18.67	2.30	0.00
19.00	17.21	2.70	0.00
20.00	17.24	2.85	0.00
21.00	18.21	3.05	0.00
22.00	19.40	3.45	0.00
23.00	16.91	4.10	0.00
24.00	17.73	4.25	0.00

Table 5 Load demand data

Hour	Demand (kW)	Hour	Demand (kW)	Hour	Demand (kW)	Hour	Demand (kW)
1:00	32.93	7:00	33.06	13:00	49.20	19:00	47.76
2:00	30.40	8:00	39.71	14:00	48.27	20:00	44.71
3:00	29.13	9:00	44.94	15:00	47.93	21:00	42.69
4:00	28.43	10:00	47.37	16:00	47.69	22:00	42.19
5:00	28.60	11:00	47.91	17:00	48.99	23:00	41.07
6:00	28.74	12:00	48.39	18:00	49.94	24:00	36.53

The power generated form the renewable generation like wind and solar power outputs (in kW) is plotted in the bellowed figure.

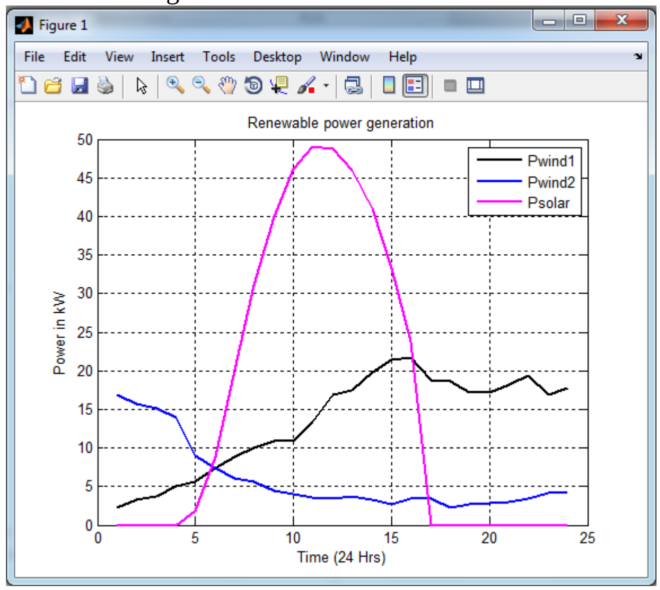


Figure 3 Renewable power generation

The load demand and the P_net values are tabulated in the following table and the curve response is shown in figure.

Table 6Pload and Pnet value

Hour	Load demand (kW)	Pnet (kW)	Hour	Load demand (kW)	Pnet (kW)
1.00	32.93	13.66	13.00	49.20	-18.02
2.00	30.40	11.34	14.00	48.27	-15.86
3.00	29.13	10.23	15.00	47.93	-9.53
4.00	28.43	9.43	16.00	47.69	-1.30
5.00	28.60	12.22	17.00	48.99	26.78
6.00	28.74	5.25	18.00	49.94	28.97
7.00	33.06	-1.98	19.00	47.76	27.84
8.00	39.71	-7.15	20.00	44.71	24.62
9.00	44.94	-10.46	21.00	42.69	21.42
10.00	47.37	-13.64	22.00	42.19	19.34
11.00	47.91	-18.13	23.00	41.07	20.06
12.00	48.39	-20.74	24.00	36.53	14.55

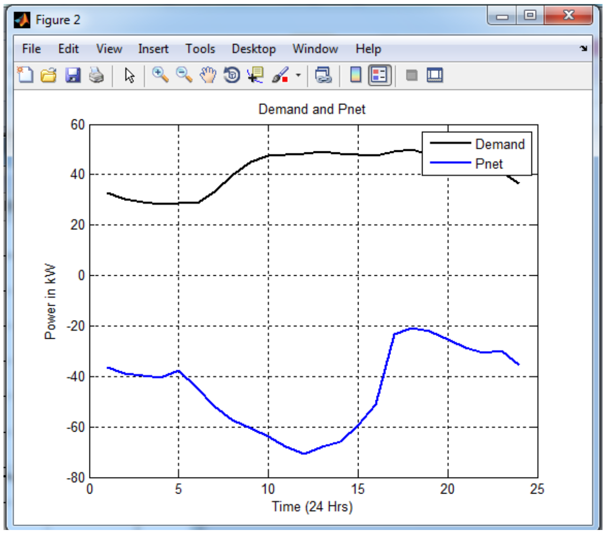


Figure 4 Demand and Pnet profile

The proposed work addressing a stochastic optimization of micro grid based on battery cost model. This work proposes an idea of optimizing the micro grid power delivery based on battery operating condition.

Because batteries are clean to environment and can introduce minimum fuel cost when compared to the existing Diesel generators. The cost model is I need of the battery specification especially VRB batteries are highly recommended for the microgrid operation because of its lower DoD compared to other battery. In order to include the battery cost model, the following table 4.8 has been referred in this work. The detailed co-efficient of the battery during charging and discharging is shown in the following table.

Table 7 VRB battery model coefficients

(i,k)	a_k^i	b_k^i	c_k^i	d_k^i
(o,v)	$0.2414V_r$	$0.9925V_r$	–	–
(d,i)	$\frac{1.0719}{V_r \times 10^{-3}}$	$0.0183I_r$	$0.0210I_r$	–
(c,i)	$\frac{-0.3093}{V_r \times 10^{-3}}$	$\frac{1.0397}{V_r \times 10^{-3}}$	$0.0604I_r$	$-0.122I_r$

The multi objective genetic algorithm is implemented in this work for the optimization of unit commitment as well as economic dispatch.

The Economic dispatch from the proposed stochastic model is shown in the following figures. The power delivery is based on the load requirement as well as battery SOCs, renewable energy generations.

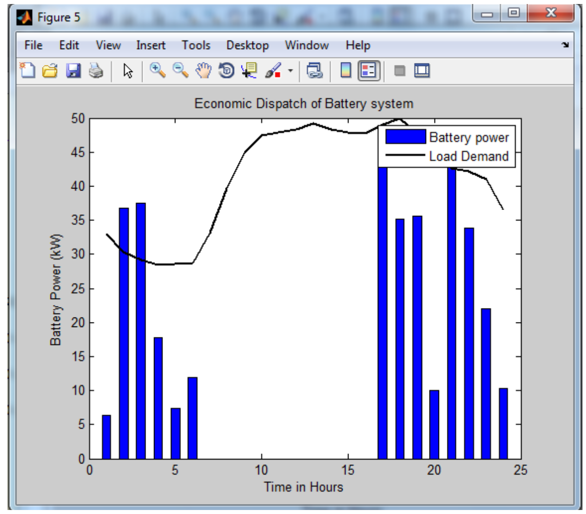


Figure 5 Economic dispatch of battery system

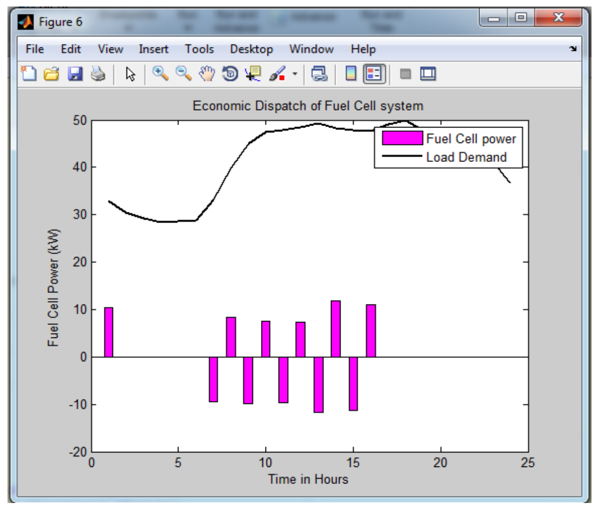


Figure 6 Economic dispatch of Fuel cell system

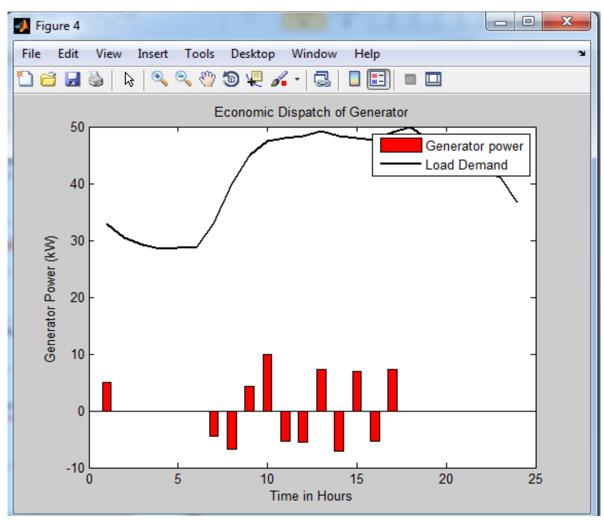


Figure 7 Economic dispatch of Generator

The above figures consists of optimized power ratings of Diesel generators, VRB battery and fuel cell system. The positive power represents the power delivery to the grid, and negative power represents (especially batteries) charging instances. The battery as well as Diesel generators may kept idle whenever the load demands very low. The stochastic optimization of micro gird is shown as a curve in the following figure

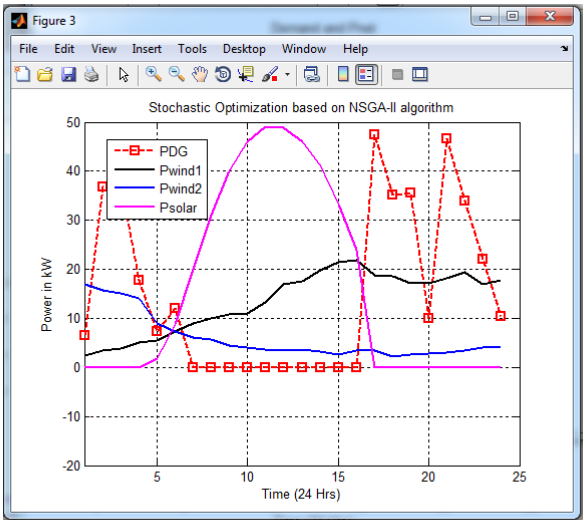


Figure 8 power responses after stochastic optimization of microgrid

V. CONCLUSION

The main aim of the work is to promote the battery operation in power grids and to improve the electricity distribution as well as to reduce the fuel cost. This proposed work is most suitable for the stand alone system. Still now, the diesel generators are the only a bulky source to face the peak loads in standalone systems because the renewable energy resources are stochastic in nature. This can be solved by operating the system by adding vehicle battery storage system and it is possible to schedule the diesel generators and also can reduce the fuel cost. This project to design optimal control for a novel regional PEV charging station system, which serves its demand by wind/solar generation and electricity from the utility grid. This project genesis a simulated annealing optimization based economic schedule of microgrid based on battery storage system. This optimal control system is focused on minimizing its operational cost. The MATLAB simulation results are showing the effectiveness of the proposed work on 7 bus stand-alone systems.

REFERENCES

[1]. Mahmoud MS, Hussain SA, Abido M. Modeling and control of microgrid: an overview.J Franklin Inst 2014;351:2822–59.

[2]. Basak P, Chowdhury S, neeDey SH, Chowdhury S. A literature review on integrationof distributed energy resources in the perspective of control, protection and stabilityof microgrid. Renew Sustain Energy Rev 2012;16:5545–56.

[3]. Farzan F, Lahiri S, Kleinberg M, Gharieh K, Farzan F, Jafari M. Microgrids for funand profit: The economics of installation investments and operations. IEEE PowerEnergy Mag 2013;11:52–8.

[4]. Fathima AH, Palanisamy K. Optimization in microgrids with hybrid energy systems–a review. Renew Sustain Energy Rev 2015;45:431–46.

[5]. Mostafa HA, El Shatshat R, Salama M. A review on energy management systems, In:T&D conference and exposition, 2014 IEEE PES; 2014. p. 1–5.

[6]. Sadeghi M, Kalantar M. The analysis of the effects of clean technologies from economicpoint of view. J Cleaner Prod 2015;102:394–407.

[7]. Zhao B, Shi Y, Dong X, Luan W, Bornemann J. Short-term operation scheduling inrenewable-powered microgrids: a duality-based approach. IEEE Trans Sustain

[8]. Energy 2014;5:209–17.

[9]. Aghajani G, Shayanfar H, Shayeghi H. Presenting a multi-objective generationscheduling model for pricing demand response rate in micro-grid energy management.Energy Convers Manage 2015;106:308–21.

[10]. Prodan I, Zio E. A model predictive control framework for reliable microgrid energymanagement. Int J Electr Power Energy Syst 2014;61:399–409.

[11]. Mohammadi S, Soleymani S, Mozafari B. Scenario-based stochastic operationmanagement of microgrid including wind, photovoltaic, micro-turbine, fuel cell andenergy storage devices. Int J Electr Power Energy Syst 2014;54:525–35.