WESTERN UNIVERSITY

FACULTY OF ENGINEERING

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

ECE 9013A

Programming for Engineering

Final Report:

Integration of Renewable Energy Sources in Smart Grids by Means of Evolutionary Optimization Algorithms In Python

Developing and Executing Genetic Algorithm in Python

Name: Kumarajeeva Elavarasan

Student Number: 251124570

Contents

Table of Figures

1. Introduction

Growth and progress of each society are always accompanied with energy generation and consumption. According to statistics, energy demand has increased significantly. In 1960, it was 3.3 Gigatons of Oil Equivalent (Gtoe). In 1990, it increased to 8.8 Gtoe which has annual growth of 3.3 percent. In 2020, the energy consumption is projected to be 14 Gtoe/Year. The most critical question remains whether the fossil energy can respond to this growth.

The two main reasons why fossil energy cannot respond to this growth are:

   I.   Limitation of fossil energy resources
   II. Global warming. Fossil energies are one of the most significant parameter of climate changes which is adversely affecting human beings.

Due to aforementioned reasons, governments are investing on renewable energy recourses. In electrical power systems, these energy resources have been widely used. In conventional power system, there was a power plant, transmission line and a consumer in essence. In order to get most of energy resources this configuration has been changed and governments encourage consumers to generate power at low scale. This type of generation is called DG[1]. Even though, DGs have many advantages such as more reliability, peak load response, etc, they engender some issues that power system expert should take them into account. One of the most critical factors that overwhelm power system network is RPM[2][1]. Reactive power management consists of two crucial parameters which are voltage profile and voltage stability. For fulfilling these criteria many reactive power resources have been installed. Flexible AC Transmission Systems[3] is one of the reactive power energy resources that has been proposed to compensate the reactive power shortage. Taking RPM factors into account many solutions have been proposed consisting of DG placement, increasing voltage stability, improving voltage profile, minimizing cost.

---

[1] Distribution Generation
[2] Reactive Power Management
[3] FACTS

2. Related work

By going through literature review in this area, researchers focus on two main issues:

   I.   Power system elements

  II.   Solving the objective functions of proposed solutions by using different algorithms

In [2] for reducing losses in network GA[4] method is used to solve. In [3] by defining sensitive buses, authors by manipulating load flow analysis, reduced the loss in network and consequently they improved the voltage profile. In our project, this algorithm has been emphasized. The single objective function which is solved by OPF[5]. In [5], which the project is based on this article, the authors define RPM as a non-linear, constrained problem:

$$\min_{x \in R^n}$$
$$subject \text{ to} \begin{cases} g(x,u)=0 \\ h(x,u)=0 \end{cases} \tag{1}$$

This single objective functions have been used in many research papers and many solutions have been proposed [3, 6]. In most of these papers, loadability and voltage stability are the main factors to be optimized. Loadability (λ) is defined by the minimum voltage allowable limit.

By taking to account DG optimal placement with respect to maintaining maximum penetration level the optimization problem changes to multi-objective optimization problem.

$$\min F(y) = \alpha f(y) + \beta g(y) \tag{2}$$

Where $f(y)$ maximize voltage stability, $g(y)$ represent the DG penetration level. α and β are weight parameters which in the article and the project are assumed to be 0.5.

---

[4] Genetic Algorithm
[5] Optimal Power Flow

3. Problem Definition

By using objective function from [5], the optimization problem is defined as Eq.3.

$$\min F(y) = \frac{1}{2}(1 - f(y)) + \frac{1}{2}(\frac{1}{g(y)})$$ (3)

Where $f(y) = \lambda$, optimize loading parameter and $g(y) = \dfrac{\sum_{i=1}^{n} P_{DG_i}}{P_{load}}$ is the penetration level.

Since the project is defined to be implemented by python the network for test case is a four-bus system which is depicted in Fig.1.
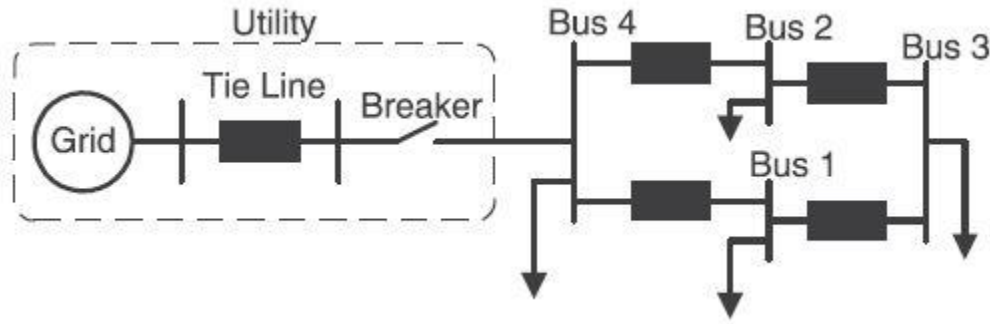

Figure 1- Test case network

The total active and reactive power of the system is 500 MW and 309.86 Mvar. The main algorithm for implementation is GA. However, many efforts have been done to solve the problem with Cuckoo search and PSO[6]. After the GA is done, best DG location and the maximum loadability factor is derived because of the algorithm.

---

[6] Particle Swarm Optimization

4. Methodology and Results

4.1. Genetic Algorithm

4.1.1. Introduction

There are plenty of evolutionary optimization algorithms. One of the algorithms is Genetic Algorithm (GA). Genetic algorithm is a high-level procedure inspired heavily from natural selection. Natural selection is the concept of survival and reproduction of the species. It also acts as a key mechanism of evolution. Genetic Algorithm usually gives high-quality solutions to the optimization problem. It also involves operations like selection, crossover and mutation which are heavily bio-inspired.

4.1.2. Genetic Algorithm

It was developed by John Holland in the year 1960. It was heavily inspired from 'Darwin's theory of evolution'. It was further improved and extended by David E. Goldberg in 1989. Genetic algorithms are useful in almost all fields like optimization, design, and applications. Random changes are applied to the current solution to generate new solutions. Multiobjective optimization problems are difficult to be solved with conventional methods. It has become very famous in recent years.

4.1.2.1. Encoding

The genetic information is added to the chromosome of the individual. The three genes correspond to Bus location, Reactive power injected, Loadability factor.
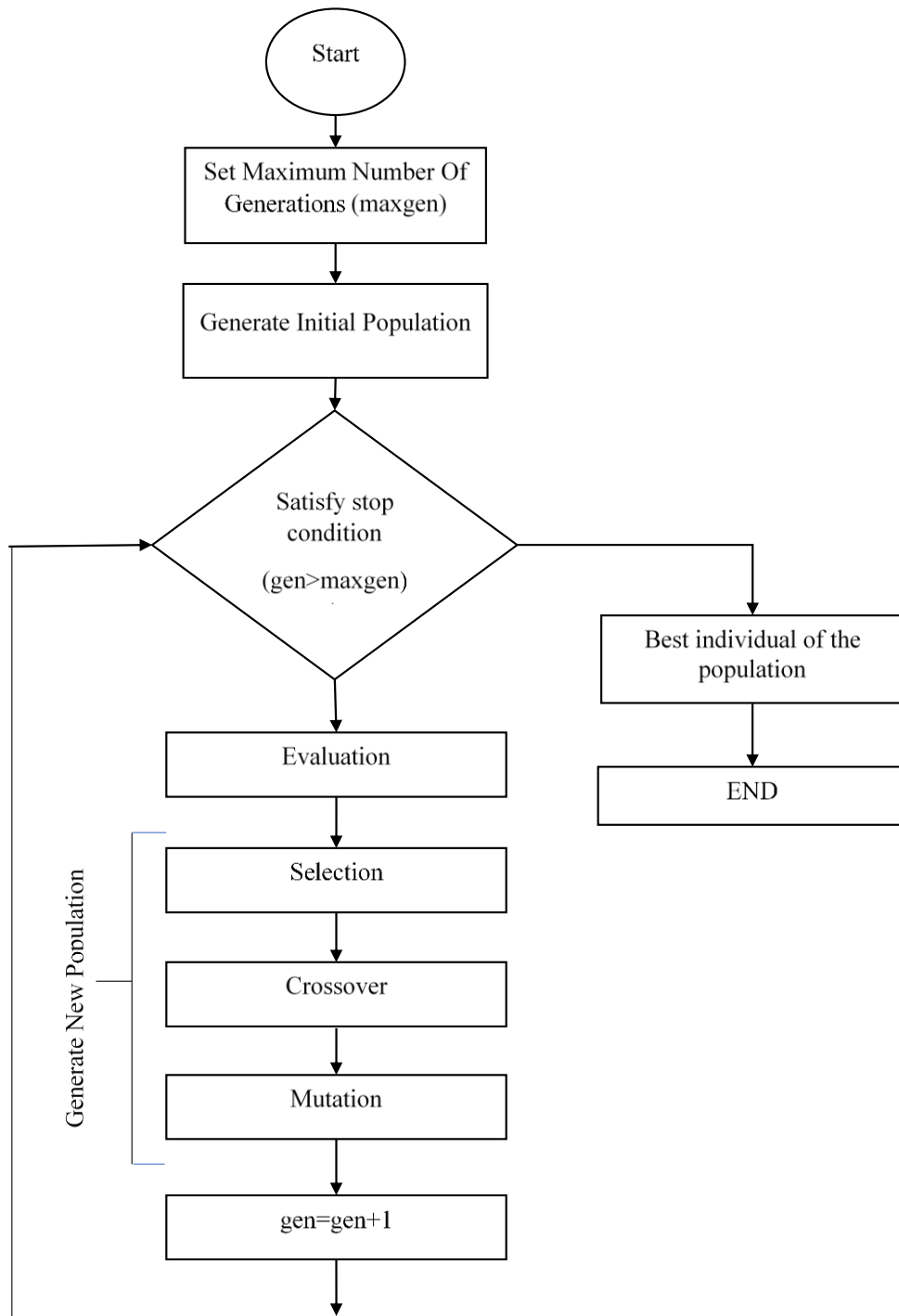
Figure 2- GA Flowchart

4.1.2.2. Initial Population

The size of the population is determined by the problem's nature. There are lot of ways to obtain the population. In our case, we have limits of different variables of the system. So, the most suitable and optimal choice is randomization. We must decide the number of genes for each chromosome. It may have hundreds of potential solutions. The initial population is randomly generated. An individual is described by a couple of parameters called as genes. A string called Chromosome is formed by linking of genes. In our case, we have five chromosomes with three genes. The three genes are loadability, Reactive power injected and location of the bus. A random function which is in-built is used to generate random numbers.

For example, λ = numpy.random.uniform(low=0, high=1, size=(5,1))

Where λ is the loadability factor with random values generated from 0 to 1

Initial population of the 4 bus system.

	λ (p.u.)	Bus_{DG}	Q_{DG} (Mvar)
Chromosome 1	0.13205	2	149.537
Chromosome 2	0.69965	2	187.455
Chromosome 3	0.4859	2	233.818
Chromosome 4	0.18272	3	209.539
Chromosome 5	0.10121	3	221.811

Figure 3- Initial Population

4.1.2.3. Evaluation

The function is evaluated for all the individuals in the initial population. The individual with bigger function value is considered as the best individual in the population. It shows the estimated survival probability in the next generation. And they are sorted and ranked from higher to lower values.

$$\min F(y) = \alpha\big(1 - f(y)\big) + \beta \left(\frac{1}{g(y)}\right)$$

where:

- f(y) =λ, optimize loading parameter
- g(y)= $\frac{P_{DG}}{P_{LOAD}}$, represents penetration level
- α and β are weight parameters with values $\frac{1}{2}$

```
Initial Population
[[1.10052686e-01 2.00000000e+00 2.47747591e+02]
 [2.49871335e-01 2.00000000e+00 2.19241439e+02]
 [1.07921940e-01 2.00000000e+00 1.43457888e+02]
 [6.07987171e-01 3.00000000e+00 2.49772423e+02]
 [6.89213963e-01 3.00000000e+00 1.66901665e+02]]
FITNESS
[[1.4540652 ]
 [1.51535971]
 [2.18871072]
 [1.19691755]
 [1.65328101]]
Generation :  0
EVALUATION
[[1.07921940e-01 2.00000000e+00 1.43457888e+02]
 [6.89213963e-01 3.00000000e+00 1.66901665e+02]
 [2.49871335e-01 2.00000000e+00 2.19241439e+02]
 [1.10052686e-01 2.00000000e+00 2.47747591e+02]
 [6.07987171e-01 3.00000000e+00 2.49772423e+02]]
```

Figure 4- Initial Population, Fitness, Evaluation Results

4.1.2.4. Selection

It is a process of genetic algorithm where the individuals are selected for breeding afterwards. The group of parents are obtained after selection process. The first four parents in the sorted list are selected as parents who are going to participate in the crossover process. The last one is reserved for mutation process.

4.1.2.5. Crossover

A couple of parents are selected for performing the crossover operation. A random point is also selected to implement the crossover operation. Let us consider an example with four parents with each of them having five genes. The process starts with the first couple. A crossover point is selected between genes 2 and 3. So, new offspring would be created by copying genes 1 and 2 from the first parent and the remaining three genes from the second parent.
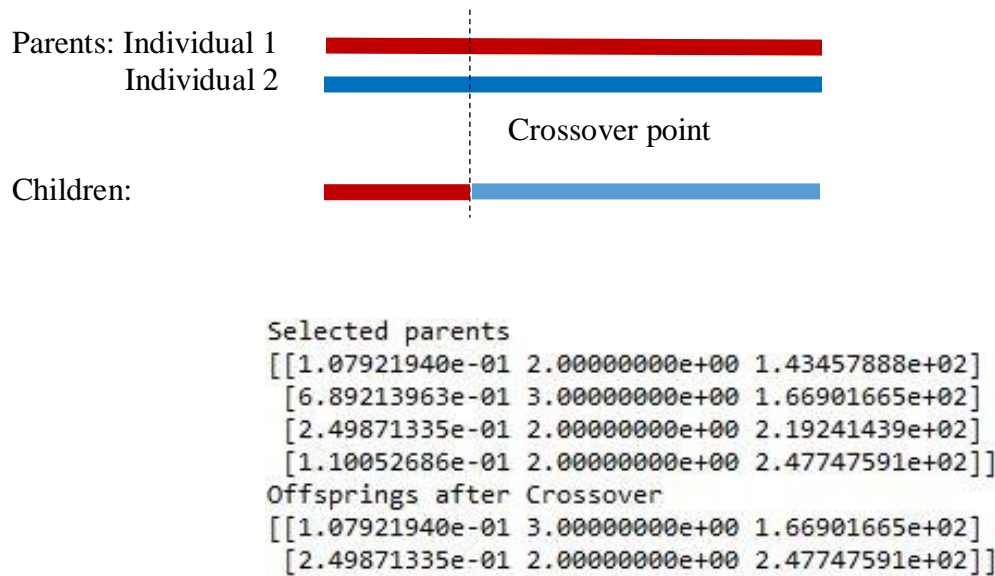
Parents: Individual 1

Individual 2

Crossover point

Children:

```
Selected parents
[[1.07921940e-01 2.00000000e+00 1.43457888e+02]
 [6.89213963e-01 3.00000000e+00 1.66901665e+02]
 [2.49871335e-01 2.00000000e+00 2.19241439e+02]
 [1.10052686e-01 2.00000000e+00 2.47747591e+02]]
Offsprings after Crossover
[[1.07921940e-01 3.00000000e+00 1.66901665e+02]
 [2.49871335e-01 2.00000000e+00 2.47747591e+02]]
```

Figure 5- Selected Parents & offspring after Crossover

The crossover point is between genes 1 and 2. The first crossover child will have its gene 1 from first parent and genes 2, 3 from the second parent. The second crossover child will have its gene 1 from third parent and genes 2, 3 from the fourth parent. The offspring are stacked together with 'numpy.vstack' command

4.1.2.6. Mutation

The reserved and remaining individuals takes place in the mutation process. Mutation rate has been set to 0.5. A random number 'r' from 0 to 1 is generated and compared with the mutation rate. If the random number is lesser than mutation rate, all genes of the participant would get mutated. As a result, a new offspring will be born.

```
After Mutation
[  0.3622309    3.           175.42363915]
```

Figure 6- Genes after Mutation

4.1.2.7. New Population

After the process of crossover and mutation, a new population is formed. The new population comprises of off springs from crossover and mutation process. It also comprises of individuals selected from elitism operator. Elitism is the process of retaining the best individuals of current generation unaltered in the subsequent generation. For example, if there are five individuals, two offspring are obtained from crossover between two couples (four individuals). One offspring is obtained as a result of mutation. And two individuals with the top function values are obtained from elitist operator to carry it with subsequent generations.

```
New Population
[[1.07921940e-01 2.00000000e+00 1.43457888e+02]
 [6.89213963e-01 3.00000000e+00 1.66901665e+02]
 [1.07921940e-01 3.00000000e+00 1.66901665e+02]
 [2.49871335e-01 2.00000000e+00 2.47747591e+02]
 [3.62230899e-01 3.00000000e+00 1.75423639e+02]]
```

4.1.2.8. Final Solution

These entire operations are carried on for number of generations. The evolution process is implemented continuously until a stop parameter is satisfied. The most prevalent stop condition is to reach the maximum permissible generations. By this way, an optimal or best individual is obtained by genetic algorithm after the maximum permissible generation.

```
Best solution :  [[[1.07921940e-01 2.00000000e+00 1.43457888e+02]]]
```

Figure 7-Best Solution

5. Conclusion

This project has tried to show the fact that increasing tendency to use renewable energies has led to new issues, and reactive power management is one of the most crucial ones. Reactive power management involves different factors. Reactive power management is modelled as an optimization problem and three algorithms were used to solve the problem. Genetic algorithm is the main algorithm which defines the best location for the added DG and defines the penetration level with maximum loadability.

6. References

[1]   T. J. E. Miller, *Reactive power control in electric systems*. Wiley, 1982.

[2]   I. Niazazari, B. Vahidi, and H. A. Abyaneh, "Loss reduction of wind turbine with optimization of blade length using genetic algorithm," *Science International,* vol. 25, no. 4, 2013.

[3]   H. Hedayati, S. Nabaviniaki, Akbarimajd, and A. Akbarimajd, "A method for placement of DG units in distribution networks," *IEEE transactions on power delivery,* vol. 23, no. 3, pp. 1620-1628, 2008.

[4]   H. Raoufi and M. Kalantar, "Reactive power rescheduling with generator ranking for voltage stability improvement," *Energy Conversion and Management,* vol. 50, no. 4, pp. 1129-1135, 2009.

[5]   M. Alonso, H. Amaris, and C. Alvarez-Ortega, "Integration of renewable energy sources in smart grids by means of evolutionary optimization algorithms," *Expert Systems with Applications,* vol. 39, no. 5, pp. 5513-5522, 2012.

[6]   K. Y. Lee, X. Bai, and Y.-M. Park, "Optimization method for reactive power planning by using a modified simple genetic algorithm," *IEEE Transactions on Power Systems,* vol. 10, no. 4, pp. 1843-1850, 1995.

[7]   R. B. Payne and M. D. Sorensen, *The cuckoos*. Oxford University Press, 2005.

[8]   C. T. Brown, L. S. Liebovitch, and R. Glendon, "Lévy flights in Dobe Ju/'hoansi foraging patterns," *Human Ecology,* vol. 35, no. 1, pp. 129-138, 2007.

[9]   X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, 2009: IEEE, pp. 210-214.

[10]   X.-S. Yang and S. Deb, "Engineering optimisation by cuckoo search," *arXiv preprint arXiv:1005.2908,* 2010.

[11]   Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, 2001, vol. 1: IEEE, pp. 81-86.

[12]   C. M. Martinez and D. Cao, *iHorizon-Enabled Energy management for electrified vehicles*. Butterworth-Heinemann, 2018.