

WIND FARM OPTIMIZATION

A THESIS SUBMITTED TO  
THE GRADUATE SCHOOL OF MIDDLE EAST TECHNICAL UNIVERSITY  
OF  
MIDDLE EAST TECHNICAL UNIVERSITY

BY  
SOGAND YOUSEFBEIGI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR  
THE DEGREE OF MASTER OF SCIENCE  
IN  
AEROSPACE ENGINEERING

FEBRUARY 2013



Approval of the thesis:

**WIND FARM OPTIMIZATION**

Submitted by **SOGAND YOUSEFBEIGI** in partial fulfillment of the requirements for the degree of **Master of Science in Aerospace Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen  
Dean, Graduate School of **Natural and Applied Sciences**

\_\_\_\_\_

Prof. Dr. Ozan Tekinalp  
Head of Department, **Aerospace Engineering**

\_\_\_\_\_

Prof. Dr.I. Sinan Akmandor  
Supervisor, **Aerospace Engineering Dept., METU**

\_\_\_\_\_

**Examining Committee Members:**

Prof. Dr. M. Cevdet Çelenligil  
Aerospace Engineering Dept., METU

\_\_\_\_\_

Prof. Dr.I. Sinan Akmandor  
Aerospace Engineering Dept., METU

\_\_\_\_\_

Asst. Prof. Dr. Demirkan Çöker  
Aerospace Engineering Dept., METU

\_\_\_\_\_

Assoc. Prof. Dr. D. Funda Kurtuluş  
Aerospace Engineering Dept., METU

\_\_\_\_\_

Inst. Dr. Tahsin A. Çetinkaya  
Mechanical Engineering Dept., METU

\_\_\_\_\_

**Date:** 01.02.2013

**I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.**

Name,Last Name: SOGAND YOUSEFBEIGI

Signature:

## **ABSTRACT**

### **WIND FARM OPTIMIZATION**

Yousefbeigi, Sogand  
M.S, Department of Aerospace Engineering  
Supervisor : Prof. Dr.I. Sinan Akmandor  
February 2013, 70 pages

In this thesis, a mixed integer linear program is used to formulate the optimization process of a wind farm. As a start point, a grid was superimposed into the wind farm, in which grid points represent possible wind turbine locations. During the optimization process, proximity and wind interference between wind turbines were considered in order to found the power loss of the wind farm. Power loss was analyzed by using wind interference coefficient, which is a function of wind intensity interference factor (WIIF), weibull distribution and power of the wind turbines. Two different programs; Genetic Algorithm and Lingo, were used to solve the MILP optimization formula and results were compared for different cases in the conclusion part.

*Keywords: Micro-siting, Wake effect, Genetic Algorithm, LINGO, Maximum power*

## ÖZ

### RÜZGAR ÇİFTLİĞİ OPTİMİZASYONU

Yousefbeigi,Sogand  
Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü  
Tez Yöneticisi : Prof. Dr.I. Sinan Akmandor  
Şubat 2013, 70 sayfa

Bu tezde “mixed integer linear program” (MILP) kullanılarak rüzgar çiftliği optimizasyonu formüle edilmiştir. Öncelikle, her bir grid noktasının olası rüzgar türbini yerini temsil ettiği bir grid, rüzgar çiftliğiyle birleştirildi. Bu süreçte, rüzgar çiftliğindeki güç kaybını bulmak amacıyla rüzgar türbinlerinin birbirlerine yakınlıkları ve etkileşimleri dikkate alındı. Rüzgar yoğunluğu etkileşim faktörünün (WIF), weibull dağılımının ve rüzgar türbini gücünün bir fonksiyonu olan güç kaybı; rüzgar etkileşim katsayısı kullanılarak analiz edildi. Genetik Algoritma ve Lingo adlı iki ayrı program MILP optimizasyon formülünü çözmek için kullanıldı ve farklı durumlar için elde edilenler sonuç bölümünde karşılaştırıldı.

*Anahtar Kelimeler. Küçük- Konumlandırma, Wake etkisi, Genetic Algoritma, Lingo, Azami power*

## **ACKNOWLEDGMENTS**

I would like to express my appreciation to my advisor, Professor Dr. İ. Sinan AKMANDOR for his great guidance and support during my master thesis. He always gave really useful ideas and comments when I encountered any problem during this work. Working with him was an invaluable experiment for me and helped improve myself in different fields.

My special thank to Stuart Donovan and Gary Nates for answering my questions about their work.

I offer sincere thanks to my beloved family for their support throughout my studies.

Finally, I would like to show my acknowledgment to my dear friend Denizhan Yavaş for his help about Matlab.

## TABLE OF CONTENTS

ABSTRACT .....	v
ÖZ .....	vi
ACKNOWLEDGMENTS .....	vii
TABLE OF CONTENTS .....	viii
LIST OF TABLES .....	x
LIST OF FIGURES .....	xi
CHAPTERS	
1. INTRODUCTION .....	1
1.1. Background of Renewable Energy .....	1
1.2. Wind Energy .....	1
1.3. Wind Farm .....	2
1.4. Micro-siting .....	2
1.5. Thesis Objective .....	2
1.6. Research Process .....	3
2. WIND ASSESSMENT RESOURCES .....	5
2.1. Wind Data Analysis .....	5
2.2. Wind Direction .....	5
2.3. Wind Speed .....	6
2.3.1. Wind Speed Frequency Distribution .....	7
2.3.2. Weibull Distribution .....	8
2.4. Power .....	9
2.4.1. Power Curve .....	9
2.4.2. Power Output .....	10
3. OPTIMIZATION PROCESS .....	12
3.1. Wind Turbine Wake .....	12
3.1.1. Velocity Deficit .....	14
3.2. Proximity .....	18
3.3. Mixed Integer Program .....	19
3.3.1. General Form .....	19
3.4. Formulation Process .....	20
3.4.1. Proximity Constraint .....	22
3.4.2. Interference Constraint .....	23
3.4.3. Wake Intensity Interference Factor .....	24
3.4.4. Interference Coefficient .....	29
3.5. Optimization Model .....	30
4. THE GENETIC ALGORITHM .....	33
4.1. Overview .....	33
4.2. Methodology .....	33
4.2.1. Search Space and Fitness Landscapes .....	33
4.3. Initialization .....	33
4.3.1. Algorithm Parameters .....	33
4.4. Evaluation .....	36
4.5. Selection .....	36
4.5.1. Roulette Wheel Selection .....	36
4.5.2. Rank Selection .....	36
4.5.3. Elitism Selection .....	40
4.6. Genetic Algorithm Operators .....	40
4.6.1. Cross Over .....	40
4.6.2. Mutation .....	42
4.7. Regeneration .....	43



4.8. Termination .....	44
4.9. Encoding .....	44
4.10. Summary of the Genetic Algorithm .....	46
5. LINGO .....	47
5.1. Overview .....	47
5.2. LINGO Formulation .....	47
5.2.1. A Set .....	47
5.2.2. A DATA Section .....	48
5.2.3. A Lingo Model .....	48
5.6. Set Looping Function .....	49
5.7. LINGO Operators .....	49
5.8. Main Features of LINGO .....	49
5.9. Solver Status Window .....	50
6. RESULTS .....	53
6.1. Case One (Different Wind Directions, Same Wind Speed Frequency Distributions) .....	54
6.1.1. 9*9 Grid Form for First Case .....	55
6.1.2. 10*10 Grid Form for First Case .....	57
6.2. Case Two (Different Wind Directions, Different Wind Speed Frequency Distributions) .....	60
6.2.1. 9*9 Grid Form for Second Case .....	60
6.2.2. 10*10 Grid Form for Second Case .....	63
6.3. Conclusion and Recommendations .....	65
REFERENCES .....	67
APPENDICES .....	69
A. CALCULATIONS OF $\omega(r)$ AND $\varphi(r)$ .....	69

## LIST OF TABLES

### TABLES

Table 2.1. Sector -Wise Weibull A- and k-Parameter.....	9
Table 4.1. Random Numbers for Columns.....	41
Table 6.1. Maximum Turbine Numbers for Various Models.....	53
Table 6.2. Optimization Solutions of Different Grid Spacings for Case One.....	54
Table 6.3. Power Output of Each Wind Turbine for First Case(9*9 Grid Form).....	57
Table 6.4. Power Output of Each Wind Turbine for First Case(10*10 Grid Form).....	59
Table 6.5. Optimization Solutions of Different Grid Spacings for Second Case.....	60
Table 6.6. Power Output of Each Wind Turbine for Second Case(9*9 Grid Form).....	62
Table 6.7. Power Output of Each Wind Turbine for Second Case(10*10 Grid Form).....	65

## LIST OF FIGURES

### FIGURES

Figure 2. 1. Sample Wind rose .....	6
Figure 2. 2. Wind Direction Frequency Distribution .....	6
Figure 2. 3. Wind Speed Distribution .....	7
Figure 2. 4. Wind Speed Frequency Distribution .....	7
Figure 2. 5. Actual Wind Speed Frequency Distribution and Weibull Fit.....	8
Figure 2. 6. Sample Power Curve .....	10
Figure 2. 7. Vestas V80 Power Curve.....	11
Figure 3. 1. Modified PARK Model.....	14
Figure 3. 2. Winnd Speed Deficit By Using Modified PARK Model .....	15
Figure 3. 3. Illustration of Multiple Wakes in a Wind Farm .....	16
Figure 3. 4. Ashadow Depiction of Downwind Turbine .....	16
Figure 3. 5. Shadow of Upwind Turbine Over Downwind Turbine.....	16
Figure 3. 6. Vertical Distance Between Wind Turbines.....	18
Figure 3. 7. Horizontal Distance Between Wind Turbines .....	18
Figure 3. 8. Proximity Zone of Wind Turbines .....	20
Figure 3. 9. Wind Turbines' Interference Zone .....	21
Figure 3. 10. Vertices and Edge Schematic of Two Wind Turbines .....	21
Figure 3. 11. Proximity Constraint .....	22
Figure 3. 12. Clique Structure .....	23
Figure 3. 13. .Proximity and Interference Constraints .....	23
Figure 3. 14. Turbine Wake Depiction.....	24
Figure 3. 15. Second Region Schematic.....	25
Figure 3. 16. Complementary Angles .....	26
Figure 3. 17. Third Region .....	26
Figure 3. 18. WIIF Variations for Fixed Distance .....	28
Figure 3. 19. WIIF Variations at a Fixed Angle .....	29
Figure 4. 1. Flow Chart of a Simple GA.....	34
Figure 4. 2. Schematic of Roulette Wheel Selection.....	35
Figure 4. 3. Chromosomes' Fitness Range Before Ranking.....	39
Figure 4. 4. Chromosomes' Fitness Range After Ranking.....	39
Figure 4. 5. Row Cross Over Schematic .....	41
Figure 4. 6. Column Cross Over Schematic .....	42
Figure 4. 7. Mutation Process.....	43
Figure 4. 8. Genetic Algorithm Process .....	45
Figure 5. 1. Lingo Solver Status Window.....	50
Figure 6. 1. Wind Turbines Placement in the Wind Farm For One Direction.....	54
Figure 6. 2. Wind Turbines Placement in the Wind Farm for First Case (9*9 Grid Form) .....	55
Figure 6. 3. Fitness Graph of 9*9 Grid Shape for First Case (9*9 Grid Form).....	56
Figure 6. 4. Coordinates of Wind Turbines for First Case(9*9 Grid Form) .....	56
Figure 6. 5. Wind Turbines Placement in the Wind Farm for First Case (10*10 Grid Form) .....	55
Figure 6. 6. Fitness Graph of 9*9 Grid Shape for First Case (10*10 Grid Form).....	55
Figure 6. 7. Coordinates of Wind Turbines for First Case(9*9 Grid Form) .....	59
Figure 6. 8. Wind Turbines Placement in the Wind Farm for Second Case (9*9 Grid Form) .....	61
Figure 6. 9. Fitness graph of the Wind Farm for Second Case (9*9 Grid Form).....	61
Figure 6. 10. Coordinates of Wind Turbines for Second Case (9*9 Grid Form) .....	62

Figure 6. 11. Wind Turbines Placement in the Wind Farm for Second Case (10*10 Grid Form) .....	63
Figure 6. 12. Fitness graph of the Wind Farm for Second Case (10*10 Grid Form) .....	64
Figure 6. 13. Coordinates of Wind Turbines for Second Case (10*10 Grid Form) .....	64
A . 1. $\omega(r)$ and $\varphi(r)$ Depictions.....	69

# CHAPTER 1

## INTRODUCTION

### 1.1. Background of Renewable Energy

Renewable energy is a term that refers to natural resources based energies and these sources are sustainable for a long time without any damages and negative effects to the environment. Renewable energies have different types such as wind, water and sunlight sources. All along the life of the human being, the use of the renewable energy has been always important and is back to early years in history when human first found a fire.

Nowadays, one of the most challenging topics among the other kinds of renewable energy is the wind energy. It seems that humans realized the benefit of the wind in 200 B.C., when Chinese first invented a windmill.

### 1.2. Wind Energy

Wind is used in a process to generate useful kind of energies like mechanical and electrical energies. Wind energy or wind power refers to this conversion process in which wind turbines are used. Wind power is an alternative energy types to fossil fuel and has no negative effects on nature like fossils. Since wind is a sustainable energy source, wind energy has become widespread during last 20 years. Although wind energy has several disadvantages, it seems that the positive sides of this renewable energy are more attractive for worldwide industry. Main advantages of wind energy are;

- Wind is clean and wind energy is not affected the environment negatively
- Wind turbines occupies a smaller place in compare to other power stations
- Economically, it is more efficient since wind is free and always sustainable

### **1.3. Wind Farm**

As wind turbines have become common during years, people started to think about using multiple wind turbines in a same location in order to gain more power. This idea brought about concept of wind farm which consists of many individual wind turbines, up to over a hundred in some cases, in the same place to produce electric power. Most of the largest wind farms are located in United States and China. For instance, Alta Energy Centre with 1020 MW capacity in USA and Dabancheng wind farm with 500 MW in china are two examples of these wind farms. The number and the type of wind turbines may vary from project to project. But the main goal in all the wind farms is to maximize the captured energy while minimizing the overall cost. Therefore, all the necessary conditions must be analyzed during the construction of the wind farm to make it more efficient. Therefore, during the design process there are some steps that should be considered in order to maximize the power output and minimize the cost [1]. Some of these steps are listed below:

- Study of the wind climate assessment data such as wind rose and weibull distribution
- Find the best type of turbine for current project and analyzed its characteristics considering the price and its stability under various climate conditions
- The design of a reliable electrical collector system with respect to road construction costs, maintenance cost and minimizing cable energy losses
- Micro-siting of wind turbines and put them somewhere close to an interconnection point to minimize the collector system cost and energy cable loss
- Obstacles and limitations related to environment or regulations such as noise level, road condition, high terrain
- Overall cost
- Constructability
- Electricity rates
- Economical side of the wind farm considering payment schedule, currency exchange fluctuation, discounting factors

### **1.4. Micro-siting**

As it was mentioned before, the main objective of a wind farm construction is to capture more energy by placing wind turbines in a same location. It is important to put the wind turbines in a place that they can produce more power in comparison to other positions. Micro-siting indicates the exact position of each wind turbine in a wind farm since the place of the wind turbine matters in a wind farm. The exact position of wind turbines can be found by using different methods such as Finite Difference methods and Genetic Algorithm. No matter what the method is, there are some factors that should be analyzed during the optimization process. Proximity and interferences between wind turbines are the two important criteria in the micro-siting.

### **1.5. Thesis Objective**

The scope of the current thesis is to find the best locations for wind turbines in a hypothetical wind farm in order to produce maximum power. A grid is imposed onto the wind farm and each grid point shows the possible location of a wind turbine.

A mixed integer linear program was used in the optimization process to model the optimization formula. Proximity and interference between wind turbines were analyzed to minimize the wake effects in the wind farm.

In this work, the best distance between each grid point was found based on the maximum produced power. However, this value may be about to three turbine diameters which the optimum distance for many manufactures. Moreover, velocity deficit in the wind farm was taken into account to compute each turbine's velocity. And power calculations for each turbine were done considering the wake effect.

Two different programs were used in the optimization process; Matlab and Lingo. Matlab was used to write a Genetic Algorithm code considering the model and optimization formula. The results of GA and Lingo will be compared later to see if there is any difference between solutions and also find the advantages and disadvantages of each program for different conditions.

## 1.6. Research Process

During the literature survey, a comprehensive research was done to find the best way for micro-siting a wind farm. The research process had three main steps. In the first step, I tried to found out the concept of the micro-siting and understand clearly what the main factors in wind farm optimization are. Some of these works are listed here;

1. ***“Wind turbines type and number choice using combinatorial optimization”***, by Ivan Mustakerov and Daniela Borissova

*Main Idea:* Helped to understand the concept of micro-siting. And Introduce a combinatorial optimization model for wind turbines type and number choice and placement considering the given wind conditions.

2. ***“Optimal Micro-Siting of Wind Turbines by Genetic Algorithms Based on Improved Wind and Turbine Models”***, by Chunqiu Wan, Jun Wang, Geng Yang, Xiaolan Li and Xing Zhang

*Main Idea:* The improved wind and turbine models are formulated into an optimal control framework in terms of minimizing the cost per unit energy of the wind farm, which is solved by a binary-encoded genetic algorithm.

3. ***“Design of wind farm layout using ant colony algorithm”***, by Yunus Eroğlu, Serap Ulusam Seçkiner

*Main Idea:* presents an ant colony algorithm for maximizing the expected energy output. The algorithm considers wake loss, which can be calculated based on wind turbine locations, and wind direction.

4. ***“Design of wind farm layout for maximum wind energy capture”***, by Andrew Kusiak, Zhe Song

*Main Idea:* A generic model for optimizing in-land wind farm layout is presented. The optimization model considered wind farm radius and turbine distance constraints.

5. ***“The shadow effect of large wind farms: measurements, data analysis and modelling”***, by Sten Frandsen, Rebecca Barthelmie, Ole Rathmann, Hans E. Jørgensen, Jake Badger, Kurt Hansen, Søren Ott, Pierre-Elouan Rethore, Søren E. Larsen, Leo E. Jensen

**Main Idea:** Helped to understand the shadow problems and wake effects. And, it gives a prediction of the power losses from a wind.

All above source was like a first step to understand different sides of the micro-siting. In the second step, I had to find a model for wind farm optimization and after understanding and finding out various ways of wind farm optimization, my supervisor Prof.Dr.I.Sinan AKMANDOR suggested choosing the following works to be the key papers in this work..

6. “*Wind Turbine Interference in a Wind Farm Layout Optimization by Using Mixed Integer Linear Programming Model*”, by Rosalind Archer, Gary Nates, Stuart Donovan and , Hamish Waterer

**Main Idea:** Develops a wind intensity interference coefficient which captures the interference caused by an upwind turbine on a downwind turbine in the same wind flow.

7. “*Whither the wind blows: wind flow modelling and wind farm layout optimisation*”(A thesis submitted in fulfilment of the requirements for the degree of Master of Engineering in Engineering Science, University of Auckland, 2008.)”, by Stuart Burnet Donovan

**Main Idea:** A mixed integer programming (MIP) model is developed which aims to maximize the net revenue from the wind farm, accounting for energy yield, turbine installations costs, transmission costs, turbine wake, turbine noise, and turbulence.

Therefore, I used the optimization model of Rosalind Archer, Gary Nates, Stuart Donovan and Hamish Waterer in this work for in optimization process of the wind farm.



## CHAPTER 2

### WIND ASSESSMENT RESOURCES

Wind resource assessment includes some important information about wind properties in a specific environment. These characteristics must be analyzed carefully since output power of a wind farm is directly related to these characteristics.

#### 2.1. Wind Data Analysis

Required data about wind is measured using anemometers and wind vanes placed at different heights. After that, desired information such as wind speed and wind direction will be collected by help of anemometers and wind vanes every ten minute [2].

In order to get a reliable wind data base, it is necessary to detect implausible values and to filter these. Therefore, a MCP method (Measure-Correlate-Predict) is used to maximize the data availability. Missing data is predicted by data of the corresponding measuring period of reference anemometers of the same mast or suitable masts in the vicinity. During this process, the correlation between two time series is used to fill in the periods of missing and filtered data in a target time series with predicted data of the same period of a reference time series. This procedure can basically also be used to extend the time period of one time series to the time period of a reference time series. Condition for applying the MCP process is an adequate correlation coefficient between the concurrent measurement periods.

#### 2.2. Wind Direction

It is obvious that wind direction varies over time and has not a basic direction form. Wind rose is a tool used to present all wind directions in a single graph. It depicts the wind speed distribution and wind direction at a particular location in polar coordinates. The frequency of winds for each direction is then plotted by different numbers of sectors with the same degree. The graphic in the next page (A.Al- Mohamad, H. Karmeh) is a sample of a wind rose. This wind rose has twelve sections which mean the degree between them is thirty. Each sector shows the wind speed distribution for the corresponding direction. The radial length of each sector indicates the number of data taken for that sector. The sections are numbered from 0 to 330 in clockwise direction starting from North direction to North-North-West. Therefore the most data was taken for West-North-West direction about 3000.

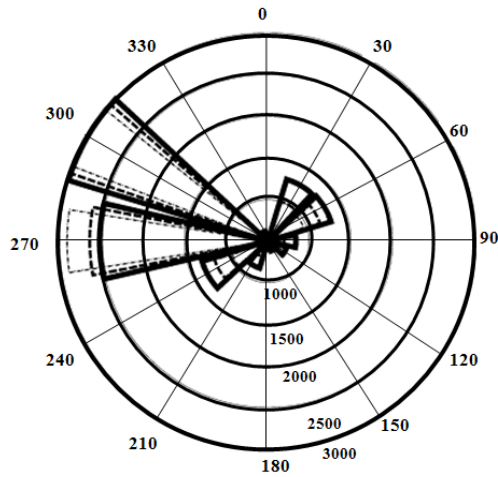


Figure 2. 1. Sample Wind rose

For the current thesis wind speed data was measured in a specific area in Turkey for 17 months with 80m mast.

### 2.3. Wind Speed

Wind speed is a scalar quantity which is measured by an anemometer and is a primary factor in power calculation. Therefore, it is important to have enough knowledge about the wind speed in a specific terrain. In the first place, we have to choose a place which has a good speed distribution during the year.

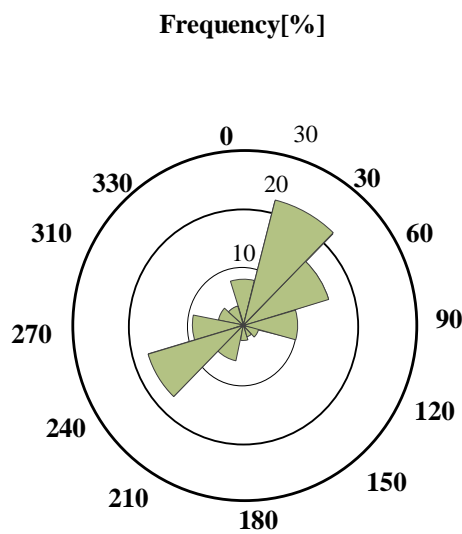


Figure 2. 2. Wind Direction Frequency Distribution

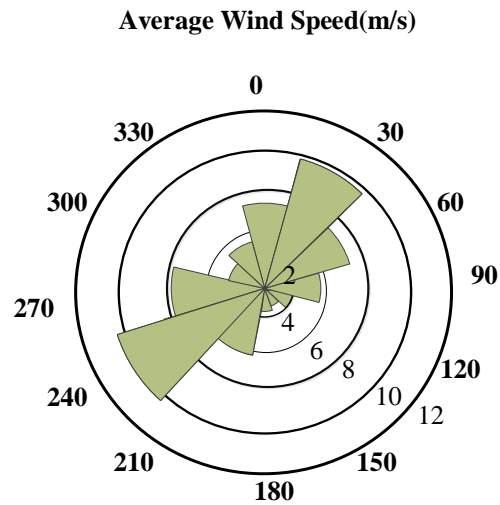


Figure 2. 3. Wind Speed Distribution

### 2.3.1. Wind Speed Frequency Distribution

Since wind speeds are stochastic (variable) for different hours in a day, the best way to find out the wind speed for wind farm optimization is to use them in terms of distribution rather than fixed numbers [2]. To generate the wind speed frequency distribution, the recorded data are used to indicate the frequency of each speed value in a recorded period. For the current thesis the speed frequency distribution is shown in following graph.

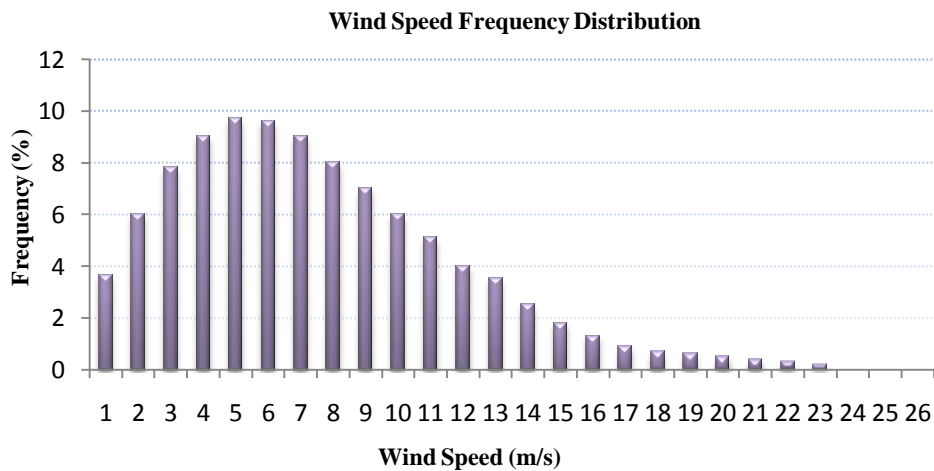


Figure 2. 4. Wind Speed Frequency Distribution

### 2.3.2. Weibull Distribution

Another efficient way to model wind speed distribution is Weibull distribution which is used to describe the size distribution of different values such as wind speed [15]. Weibull probability distribution function is written in the following form:

$$f(x; A, k) = \begin{cases} \frac{k}{A} \left(\frac{x}{A}\right)^{k-1} e^{-\left(\frac{x}{A}\right)^k} & x \geq 0 \\ 0 & x \leq 0 \end{cases} \quad (2.1)$$

Where,  $k$  is shape parameter and  $A$  is scale parameter and  $x$  is a variable. The shape of the Weibull distribution depends on shape parameter.

Weibull distribution is a statistical way to shows how often winds of different speeds will be seen at a location with a certain average (mean) wind speed. It has several benefits to use Weibull distribution such as;

- Choose the best wind turbine for a wind farm, based on turbine's cut-in(the wind speed at which the turbine starts to generate usable power)and cut-out(the speed at which the turbine hits the limit of its alternator and can no longer put out increased power output with further increases in wind speed)speeds.
- Numerical calculations of wind turbine energy output is simplified

While using Weibull distribution to for a wind turbine, a cumulative distribution function (CDF) form is used.

$$F(v; k; A) = 1 - e^{-\left(\frac{v}{A}\right)^k} \quad (2.2)$$

In the next graph, both the wind speed distribution and Weibull cumulative distribution are shown. It can be seen that the recorded data are fitted in to Weibull distribution and create a more smooth line than the actual wind distribution.

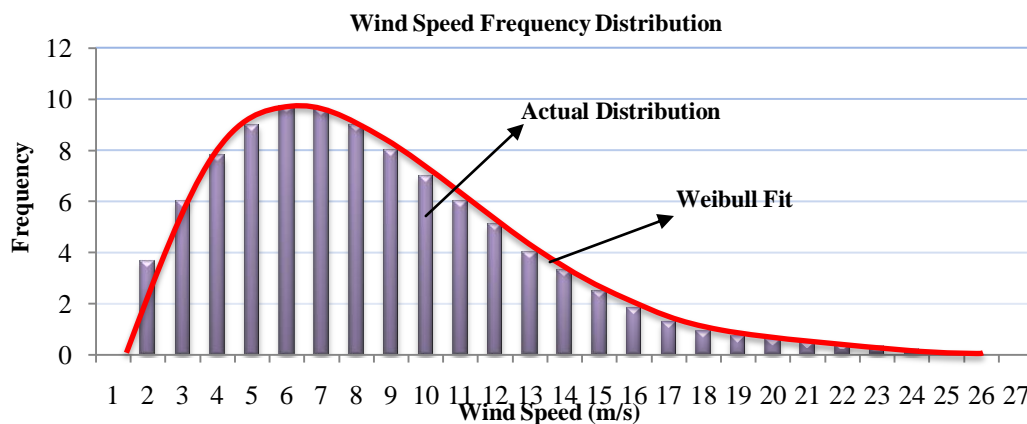


Figure 2. 5. Actual Wind Speed Frequency Distribution and Weibull Fit

In the graph below, scale and shape parameters are shown for all sections. As it is obvious, A and K varies for different directions. The A parameter is related to mean speed of wind and so is important with regarded captured energy [2].

Table 2. 1. Sector-Wise Weibull A- and k-Parameter

Sector	Direction(°)	f(%)	A(m/s)	k
N	345-15	8,1	7,85	1,80
NNE	15-45	21,6	9,91	2,73
ENE	45-75	15,3	7,33	2,46
E	75-105	9,4	6,23	2,40
ESE	105-135	2,9	4,47	1,61
SSE	135-165	1,5	3,14	1,46
S	165-195	2,1	3,78	1,40
SSW	195-225	5,7	8,28	1,54
WSW	225-255	16,9	11,74	2,24
W	255-285	9,0	8,59	1,76
WNW	285-315	4,3	4,72	1,62
NNW	315-345	3,3	5,32	1,50
<b>All Data</b>		100	8,34	1,81

## 2.4. Power

Wind power of a wind turbine is the conversion of wind energy into a useful form of energy, to make electrical power. Power of a typical wind turbine is calculated by the following formula:

$$P = C_p * (0,5) * V^3 * \rho * A \quad (2.3)$$

Where;

$C_p$ =Power Coefficient

V=wind velocity (m/s)

$\rho$ =air density (kg/m<sup>3</sup>)

A=wind turbine area (m<sup>2</sup>)

### 2.4.1. Power Curve

One of the main characteristics of a wind turbine is its power curve, which is a graph that shows the power value for various wind speeds. Power curve is different for every wind turbine and related to the physical features of the wind turbines. Therefore, it is useful to use it when an especial turbine is sought for a project.

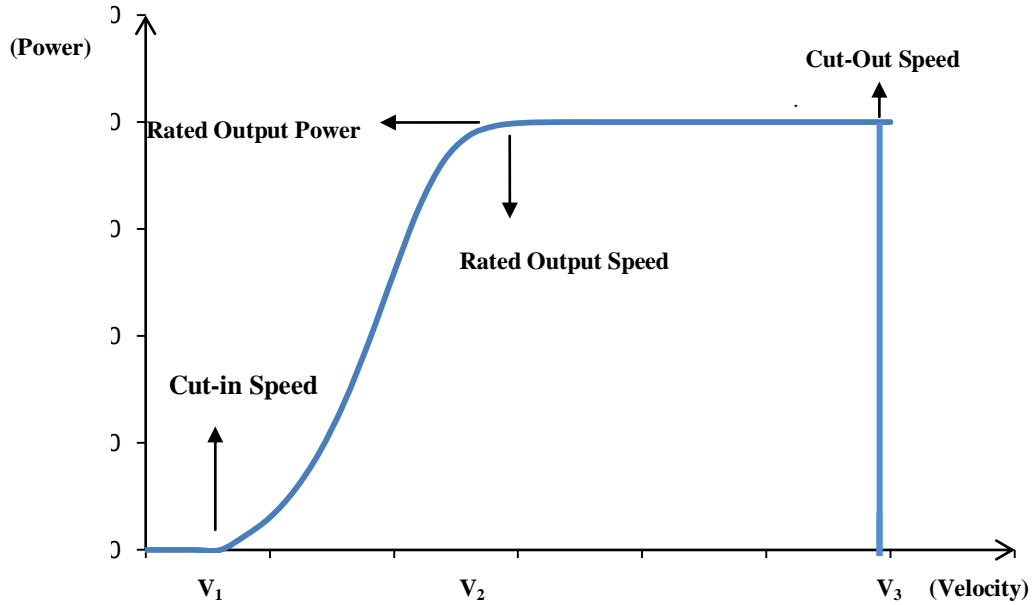


Figure 2. 6. Sample Power Curve

#### Cut-in speed:

The speed at which turbine first start to rotate and create power is called cut-in speed. This happens because at very low speed, the exerted torque by wind is not high enough to make blades rotate. After times go pass, speed increases and torque becomes sufficient enough to make the wind turbine rotate and produce power.

#### Rated output power and rate output wind speed:

It is a limit both for power output and velocity and occurs after cut-in speed area. As wind speed increases, the power output also rises and reaches a level in which generated power does not goes up any more. This value is based on wind turbine design and varies from one turbine to another but typically with large turbines, it is done by adjusting the blade angles so as to keep the power at the constant level.

#### Cut-out speed:

Cut-out speed indicates a speed value above the rated output speed. As speed continue to rises, the forces on the turbine structure also increases. Therefore, a risk of damage to rotor may occur at these speeds. To prevent any damages to wind turbine, rotor is brought to standstill position at cut-out speed.

#### 2.4.2. Power Output

To calculate the output power by using the power curve, the below formula is used:

$$P_{output} = \int_{v=0}^{v=\infty} P(v) * F(v) = \int_{v=0}^{v=\infty} P(v) * \left(1 - e^{-\left(\frac{v}{A}\right)^K}\right) dv \quad (2.4)$$

Where  $P(v)$  is the power function in the power curve and  $F(v)$  is the Weibull cumulative distribution function. Therefore, the output power is found by integrating the product of  $P(v)$  and  $F(v)$ .

### Vestas V80-2.0 MW

For the current work, Vestas V80 was used during the optimization process. In the figure 2.7 the power curve of this wind turbine is depicted.

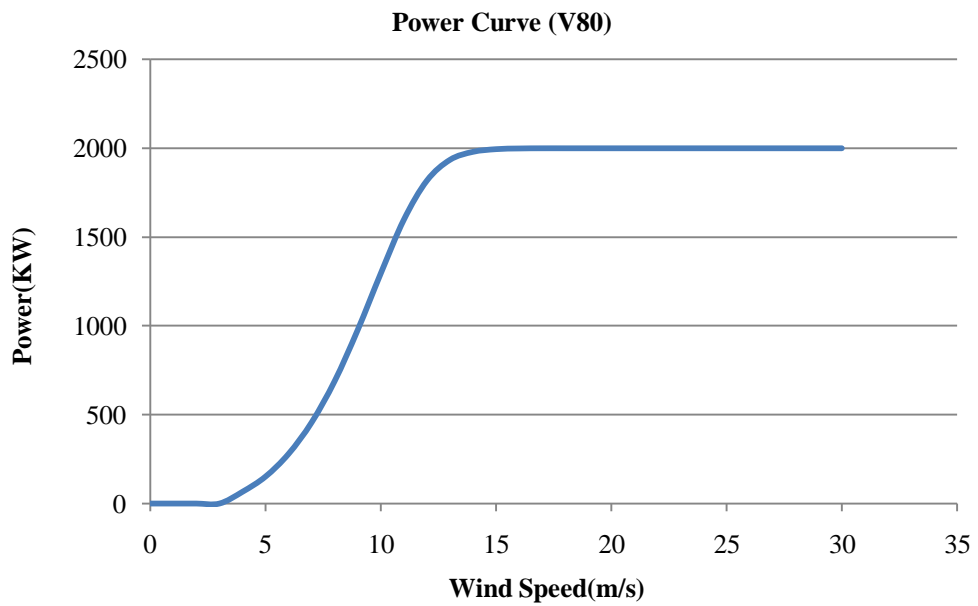


Figure 2. 7. Vestas V80 Power Curve

For vestas V80 the cut-in and cut-out values are 4 and 25 respectively. And between these values, the power varies as a function of cubic of the wind speed. The power values then change from zero to 2000KW in Vestas V80.





## CHAPTER 3

### OPTIMIZATION PROCESS

Before start to optimize a wind farm, we have to describe characteristics of that farm. Average wind speeds, wind direction and terrain properties are the most important factors in wind farm optimization. The first two characteristics were mentioned in chapter 2 and the terrain in this work is a hypothetical flat model consists of definite numbers of grid points. Further information will be given in the next chapters about the model.

In addition to the wind farm properties, interactions between wind turbines are also important and cannot be ignored since they directly affect the captured power. Therefore, proximity among wind turbines and turbine wake effects were analyzed during the optimization process. To model the optimization problem, a mixed integer program was used in this thesis which is presented in this chapter.

#### 3.1. Wind Turbine Wake

When the wind passes through the rotor of the wind turbine, a reduction in speed occurs behind the rotor. It means the downwind air of the wind turbine has a lower wind speed and higher turbulence, therefore, for downstream wind turbine energy production is not the same with upstream wind turbine since there is a deficit in wind speed. Here the upstream wind turbine is the one that is place in the free stream air direction. In free stream, the wind speed has not yet been slowed down, deflected, or otherwise impacted by a wind turbine or other obstructions [3].

Wind turbines convert kinetic energy from the wind into mechanical energy; therefore, a wind turbine in downstream produces less power since the wind contains less kinetic energy while passing through a turbine. Wake effects refer to this wind speed reduction and diminish in energy production in a wind farm based on interactions between wind turbines [3].

A wake can be divided into two regions which are near and far wake regions. The near area is the area just behind the wind turbine rotor and is approximately about one rotor diameter. Near wake region is related to rotor's properties such as number of blades, blade aerodynamics, including stalled flow, 3-D effects and the tip vortices [4]. The far wake region, on the other hand, is the area after the near wake region and is more important in wind farm analysis. In this region the main focus is on wake models, wake interference, turbulence models, environmental and atmospheric conditions and topographical effects [4]. Therefore, while the near wake region is based on characteristics of wind turbine, the far wake region is related to the interaction among wind turbines in a wind farm.

### 3.1.1. Velocity Deficit

As it was mentioned before, when wind flows through a wind turbine, there is a transfer of kinetic energy from the wind to the rotors of the turbine, which causes a reduction in the wind speed. Therefore, velocity of the downstream turbine and upstream turbine is not the same and has to be calculated. It is matter that how many wind turbines are placed in upstream and have effect over a downwind turbine. There are different methods to modelling the wake effect among wind turbines such as Ainslie's model [5], Frandsen's model [6], the Mosaic Tile model [7] and Jensen's model [8]. In this work, the N.O. Jensen wake model was used. It is a simple single wake model based on the assumption that a wake expanding linearly and the wind speed has a rectangular profile [9]. This model also called modified PARK model (GH, 2006 and Jensen, 1983) which describes the turbine wake and defines a velocity model for downwind turbine when there is no other turbines between downwind and upwind turbines.

#### Single Wake (PARK model)

It should be taken into account that the described model is for single wake model that defines the downstream wind speed of a single turbine. Therefore, the deficit velocity can be described only when the wind flows between two wind turbines, that is there is only one upstream and one downstream turbines. As it was mentioned before, in this model it is assumed that wake expands linearly with an angle  $\alpha$  (wake angle).

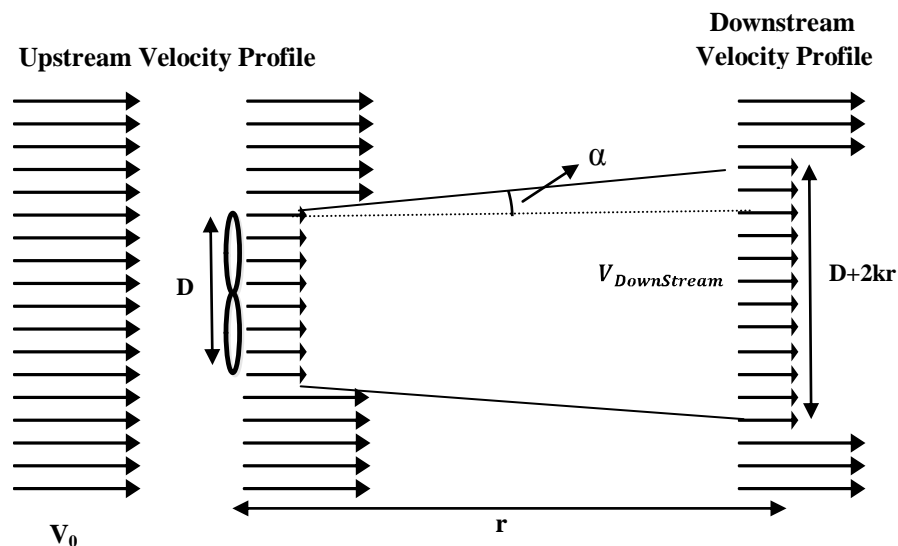


Figure 3. 1. Modified PARK Model

Where  $V_0$  is the wind speed at distance at upstream (free stream velocity) and  $D$  is the upwind wind turbine diameter.  $C_t$  is the thrust coefficient and  $K$  is the entrainment constant or opening angle which represents the effects of atmospheric stability.

Jensen experimentally found the value of  $k$  to be 0.075 for onshore applications and 0.04 for offshore applications [3].

$$V_{DownStream} = V_0 \left[ 1 - \left( \frac{D}{D+2Kr} \right)^2 * (1 - \sqrt{1 - C_t}) \right] \quad (3.1)$$

Or,

$$V_{DownStream} = V_0 \left[ 1 - \left( \frac{D}{D+2rtan\alpha} \right)^2 * (1 - \sqrt{1 - C_t}) \right] \quad (3.2)$$

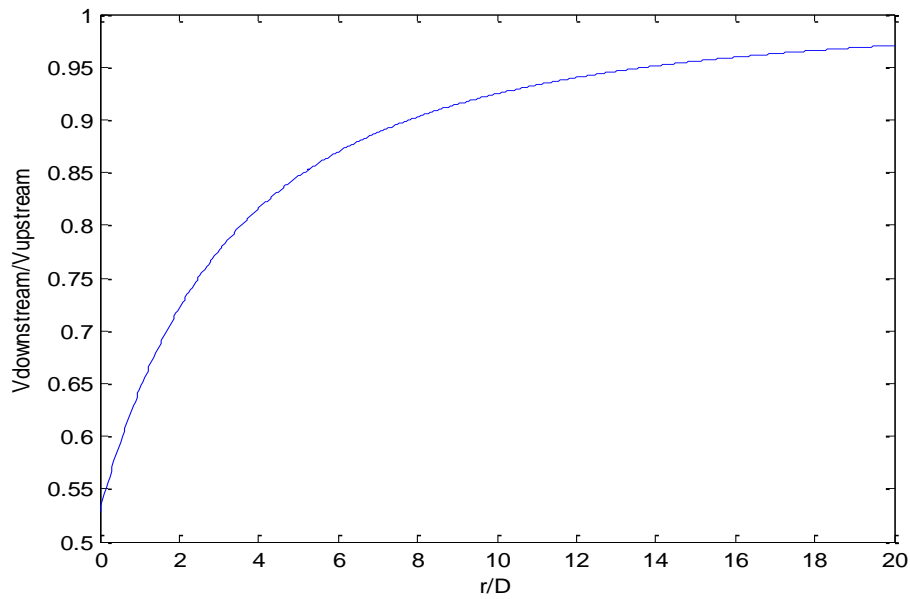


Figure 3. 2. Winnd Speed Deficit By Using Modified PARK Model

In above graph, the wind speed deficit is shown using PARK model. It is obvious that, as the  $r/D$  term increases, the fraction of the downwind turbine velocity to upwind turbine velocity approaches to one. It means  $V_{Downstream}$  gets bigger since distance between upwind and downwind turbine increases. This is due to the diminishing of the wake effect.

### Merged Wake

When there are more than two wind turbines in a wind farm, it is quite possible that one turbine is affected by several wakes. Therefore, all the effects should be taken into consideration. By using merged wake model, multiple wakes effect over a single downwind turbine is calculated. The main assumption here is that the kinetic energy deficit of interacting wakes is equal to the sum of the energy deficits of the individual wakes [3].

The following shape indicate that wind turbine “j” is not only affected by turbine “i”, it is also in the wake direction of other upwind turbines.

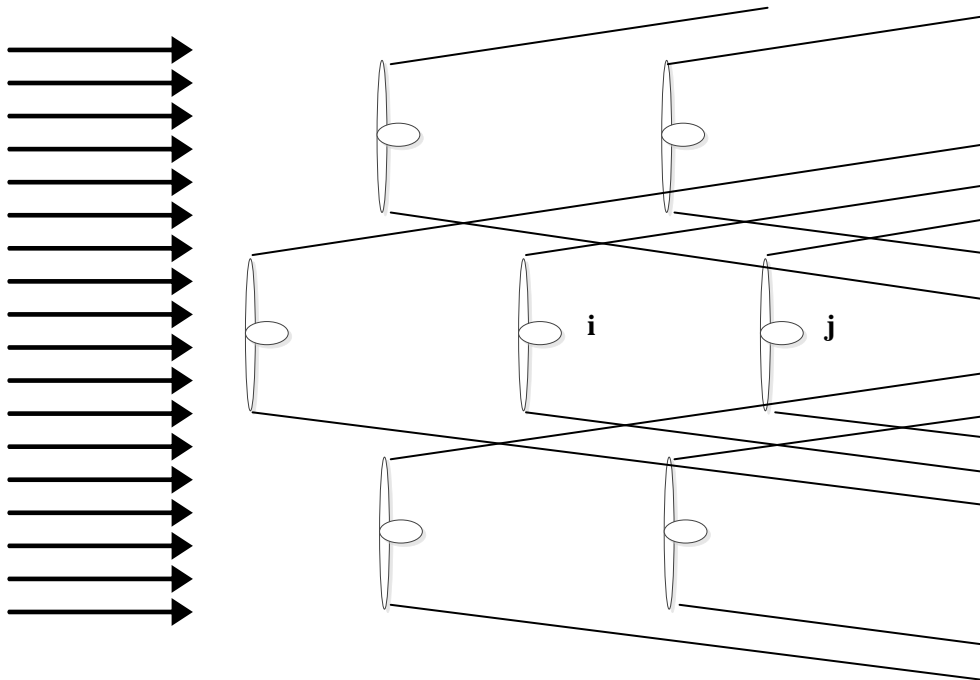
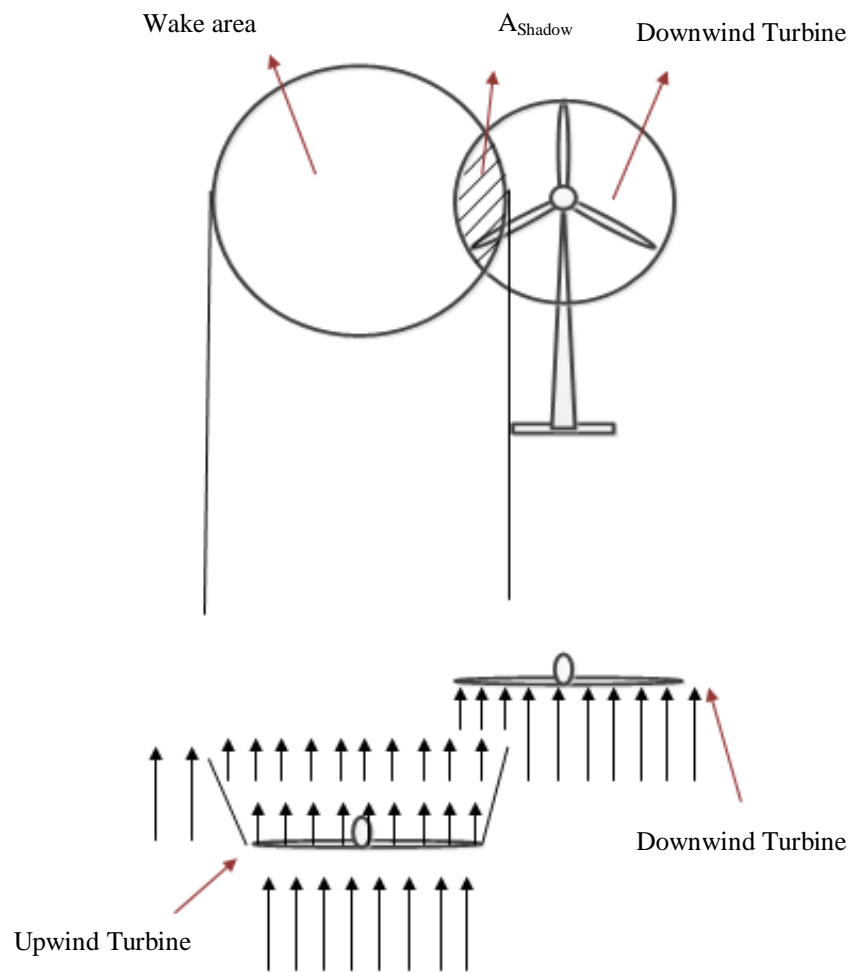
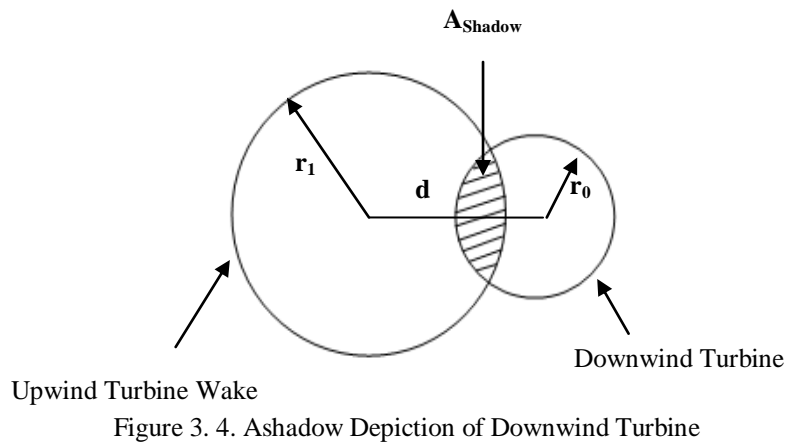


Figure 3. 3. Illustration of Multiple Wakes in a Wind Farm

Therefore, in these situations the following formula is used:

$$V_{downwind} = V_{upwind} \left[ 1 - \sum_{i=1}^n (1 - \sqrt{1 - C_t}) \left( \frac{r_0}{r_1} \right)^2 \left( \frac{A_{shadow}}{A_0} \right) \right] \quad (3.3)$$

The  $A_{shadow}$  is the area of upwind's wake over the downwind turbine that makes a shadow over that turbine. In the next graph, it can be seen that upwind wind turbine has a shadow because of its wake over downwind turbine.



Then  $A_{Shadow}$  is calculated from the following formula;

$$A_{Shadow} = \arccos\left(\frac{r_1^2 + d^2 - r_0^2}{2 * r_1 * d}\right) * r_1^2 + \arccos\left(\frac{r_0^2 + d^2 - r_1^2}{2 * r_0 * d}\right) * r_0^2 - \sin\left[\arccos\left(\frac{r_1^2 + d^2 - r_0^2}{2 * r_1 * d}\right)\right] * r_1 * d \quad (3.4)$$

### 3.2. Proximity

Proximity refers to distance between wind turbines in a wind farm. It is important to define the distance between wind turbines before start to model the optimization formula. The energy production is affected by the distance between upwind turbines and downwind turbines and it is decreasing as the space among wind turbines reduces. This happens because one of the main factors in increasing the interactions and wake effects in a wind farm is the proximity between turbines. Therefore, by decreasing the distance between wind turbines, the wake effects increase. Moreover, if turbines placed too close to each other, blades may crash with one another. And a physical damage will occur. So, the minimum distance among wind turbines is supposed to be about bigger than one diameter of a wind turbine in vertical direction if the same turbines are used in a wind farm. For horizontal direction, since blades are not crash with each other, the minimum distance should be defined based on other aspects such as wake interferences or even manufacturer's requirement. However, distances smaller than one diameter result in high power losses.

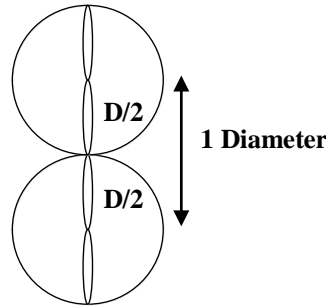


Figure 3. 6. Vertical Distance Between Wind Turbines

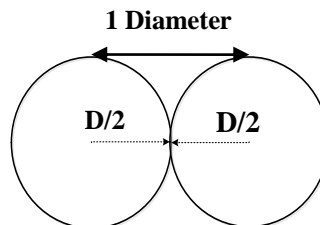


Figure 3. 7. Horizontal Distance Between Wind Turbines

Since these factors affect the gained power, the proximity should be considered carefully in order to find the best optimization model. Imperfect spacing between wind turbines, together with wake effects have negative effects both.

### 3.3. Mixed Integer Program

Integer programming is mathematical programming tool for optimization or feasibility problems in which some or all of the variables are restricted to be integers [15]. A mixed-integer program (MIP) is a linear program with a set of restriction that some, but not necessarily all, of the variables must be integer-valued. Mixed-integer programming introduces a way for optimizing process for various problems in different areas and consists of several variables that represent actions that can be taken in the system being modelled [4]. These variables are related to a function that indicates the quality of the solution by mapping each possible set of decisions into a single score, no matter if it is a minimization or maximization problem. The solution is limited by a set of constraints, which are mentioned by restricting functions of the decision variables to be equal to, not more than, or not less than, a certain numerical value [4]. Moreover, problem can be restricted by having some limitations over the set of values to which a variable might be assigned. To solve a problem with discrete problem, mixed integer programming examine a subset of possible solutions implicitly and use an optimization theory to prove that there is not any other best solution to be found. The general form of a linear programming problem is mentioned in the next part. Since MILP is also a linear program, this general form represents MILP programming model.

#### 3.3.1. General Form

Mixed integer programs generally consist of a set of decision variables, which are the unknown quantities or decisions that are to be optimized. In addition to them, an “objective function” is used in the mixed integer program which is a linear function of the decision variables and shows the quality of the problem by giving the best solution. The objective function can be maximized or minimized based on the objective of the problem. Moreover, the decisions that must be made are subject to certain requirements and restrictions of a system [4]. All these limitations are enforced in to the problem by using a set of constraints in the model. The general form of the linear programming is shown below:

$$\begin{array}{ll}
 \text{Minimize or Maximize} & c_1x_1 + c_2x_2 + \dots + c_nx_n \\
 \text{Subject to} & a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n (\leq, = \text{ or } \geq)b_1 \\
 & a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n (\leq, = \text{ or } \geq)b_2 \\
 & \dots \\
 & a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n (\leq, = \text{ or } \geq)b_m \\
 & x_j \geq 0 \quad \forall j = 1, \dots, n
 \end{array}
 \tag{3.5}$$

- $c_j$  ,  $\forall j = 1, \dots, n$  are objective coefficients and are associated with corresponding decisions in optimization problems.

- The  $b_1, \dots, b_m$  are used to indicate the constraints' values and available source or requirements.
- $a_{ij}$  represents the amount of resources or requirements that will be used by decision  $j$ .

A solution that satisfies all constraints is called a feasible solution and a feasible solution that found the best objective function value is the optimal solution. When no solution is found, the MIP is called infeasible, however, sometimes a feasible solution may be found but it is not the optimal solution. These problems are called unbounded and occur when it is possible to get infinitely good objective function values with feasible solutions.

### 3.4. Formulation Process

In the present work, the mixed integer program(MIP) formulation is used to formulate toptimization model of a wind farm layout. The objective is to find the best positiond for wind turbines in the wind farm layout in order to produce the maximum power. The model is subject to constraints on turbine proximity and turbine interference. In contrast to heuristic models, the MIP model guarantee on the quality of the solution [5]. The model is developed by superimposing an orthogonal grid onto the physical topography with regular lattice spacing as a function of wind turbine diameter. This is done in order to find the best turbines location with a minimum separation distance. Each grid point represents a possible turbine position that may possess a turbine at the end of the optimization process. According to the constraints in model, the area around each grid point is divided into three distinct regions which are [5].

The first region is includes those places that another turbine cannot be placed based on the proximity restrictions. Turbines have to be placed outside of the proximity zone where turbines are too close to each other. However, those turbines outside of the proximity zone may still have interference effects on each other which have negative effects over the total energy yield [2]. The following graph shows proximity zones by grey colour which are overlapped since turbines are too close to each other. The black colour represents the area occupied by turbine blade. Therefore, the solution is infeasible based on proximity constraint.

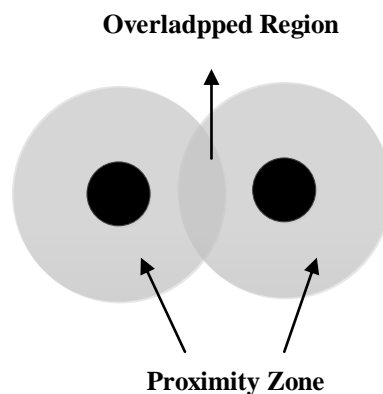


Figure 3. 8. Proximity Zone of Wind Turbines



The second area is related to those points that another turbine may be placed but there will be an interference effect between the two turbines. It means that, although proximity zones are not overlapping each other at those locations, an interference effect occurs when turbines are placed at that position. From figure 3.9, it can be seen that, although turbines are out of proximity zone, their interference boundaries are still overlapping each other.

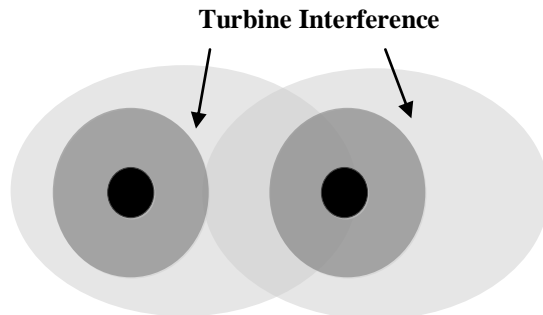


Figure 3. 9. Wind Turbines' Interference Zone

And the last region is those places that locating another turbine has no effect.

For the current optimization problem, the mixed integer program is based on a vertex packing problem which is in accordance with model's constraints. The formula is consists of vertices  $V$  and edges  $E \subseteq V \times V$ , which represents a graph  $G = (V, E)$ . The vertices  $V$  refer to possible turbine locations or intersection points of grid and edges  $E$  show the relationship between these vertices which is based on proximity and interference between turbines. Therefore, to optimize the wind farm, the vertex packing problem chooses the best combination of vertices that generates the most power [6].

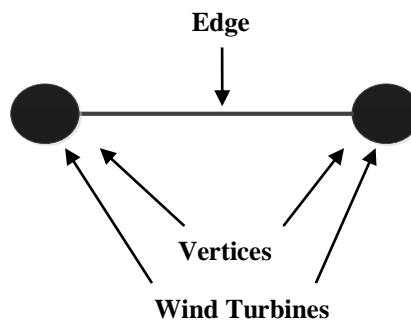


Figure 3. 10. Vertices and Edge Schematic of Two Wind Turbines

The relationship between wind turbines can be either proximity or interference. When the proximity exists between two vertices, there cannot be an interference edge between them. In other words if  $E_I$  represents the interference edges and  $E_P$  represents the proximity edges, then we have;

$$E = E_P \cup E_I \text{ and } E_P \cap E_I = \emptyset.$$

### 3.4.1. Proximity Constraint

The proximity introduces a limit for minimum distance for vertices in a grid to guarantee that there will not be a physical clash between turbine blades. Therefore, as it was mentioned before no other turbines can be located in another turbine proximity zone.

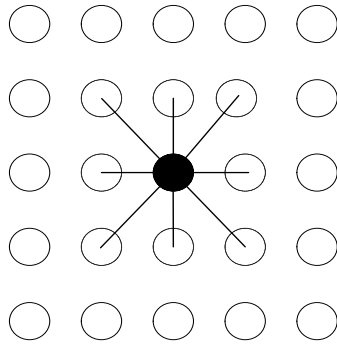


Figure 3. 11. Proximity Constraint

The above graphic depicts a solid vertex with black colour that has an edge with each vertex around it. If a turbine placed on the solid vertex, no other turbine can be positioned in around vertices because the around vertices are too close to the main turbine and may be overlapping each other proximity zone. Therefore, these positions are not suitable to locate another turbine. To model this proximity constraint we can assume two vertices  $i$  and  $j$  that have an edge with each other. If a turbine positioned in  $i$  vertex, the around space cannot occupied by vertex  $j$ . It can be written in following form:

$$x_i + x_j \leq 1, \forall (i, j) \in E \tag{3.6}$$

This constraint occurs for every vertex and edge pair in a grid. However, this formulation can be improved by expanding it for a larger subset of vertices accordance with their proximity zone. This larger subset of vertices is defined as a maximal clique  $Q$  which is consists of all vertices that are connected to each other by an edge [6].

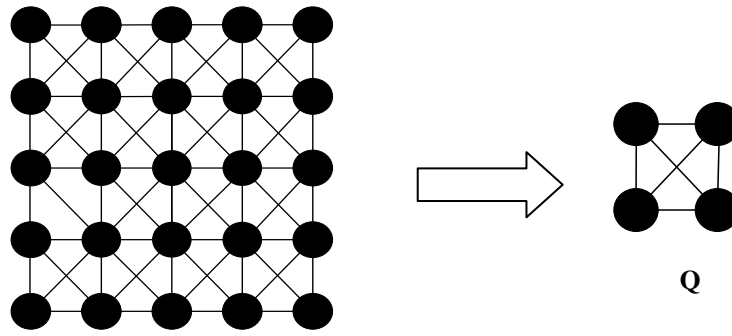


Figure 3. 12. Clique Structure

The subset  $Q$ , includes four vertices that are connected by an edge to each other that the sum of the turbines placed in that subset has to be less than or equal to one. Moreover, the cardinality of a clique is related to the turbine diameter and distance between the grid points.

### 3.4.2. Interference Constraint

In this work, the interference model is not based on an arbitrary distance between vertices in a grid. However, it is related to total captured power. That means a vertex may not only have edges with closest vertices, but it may also have edges with other vertices that are not in its proximity zone. The following graph shows a solid vertex that has edges with other vertices. These edges not only show the proximity constraints, but also interference effects.

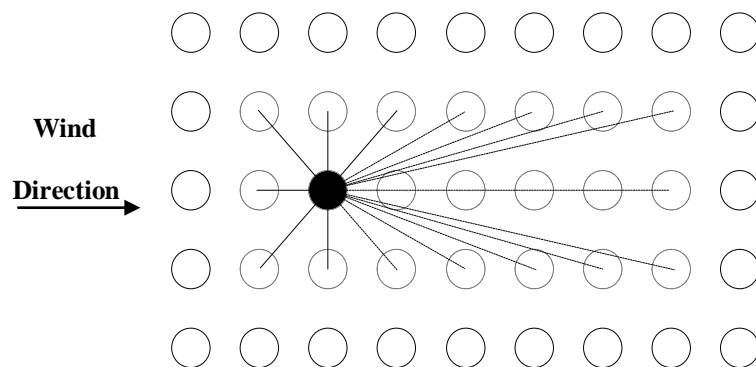


Figure 3. 13. Proximity and Interference Constraints

If  $x_u$  represents a turbine positioned at vertex u and  $x_v$  represents a turbine positioned at v, then  $I_{uv}$  refers to power loss because of the interference between u and v vertices. In other words,  $I_{uv}$  is wake interference coefficient and it is based on wake intensity interference factor and is discussed in following parts. The mathematical formulation for this constraint is then;

$$I_{ij}(x_i + x_j - 1) \leq Z_i \quad (3.7)$$

Here  $Z_i$  is zero if there is not any turbine at  $x_i$  and  $x_j$ . And if  $Z_i=1$ , then a turbine is positioned at both  $x_i$  and  $x_j$  and the gain power reduces by amount of  $I_{ij}$ .

### 3.4.3. Wake Intensity Interference Factor

The next step in formulation process is to calculate the  $I_{ij}$ . The main factor in  $I_{ij}$  calculation is to define a wake interference factor for various positions. Wake intensity interference factor represents the intensity of the interference between wind turbines. It is related to geometry of the wind turbines and wind direction distribution. The any particular wind direction is assumed to be equi-probable in every wind sector. It means, an expected wind direction exists for each sector and it is called the bisector of the wind sector. Moreover each sector has an angle of  $\mu = \frac{2\pi}{n}$  radians where n is the number of sectors. Since wind direction varies through all possible angles within the wind sectors, the extreme angles of the turbine wake represents the interference boundary and interior limit of that wind sector. Any point in the interference boundary is affected by the turbine wake for that wind direction. However, it should be noted that any particular wind direction can be considered here, therefore, there are a range of wind direction angles where the turbine wake affected that point. Note that, turbine wake is expanding by amount of  $\alpha$  angle and therefore interference boundary, interior limit and wind sector are not parallel to each other.

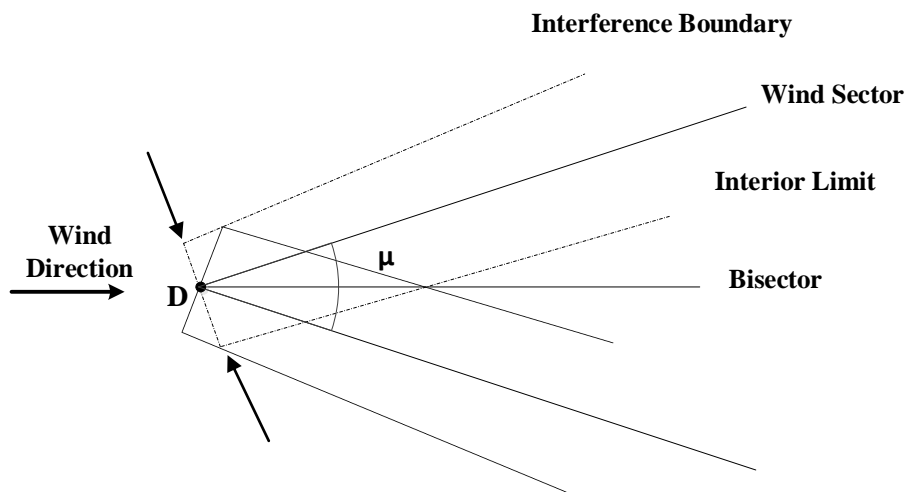


Figure 3. 14. Turbine Wake Depiction

The wind interference intensity factor (WIIF) is defined as a ratio of this angle to  $\mu$  and denotes by  $\Psi(r, \theta)$  and varies from 0 to 1. To find the exact value of the WIIF, it is divided in to four distinct regions which are to be calculated separately. It is zero when the point is outside of the interference boundary ( $\Psi(r, \theta)_1 = 0$ ); however, for remain three regions it should be calculated separately. The second region is somewhere between the interference boundary and the interior limit.

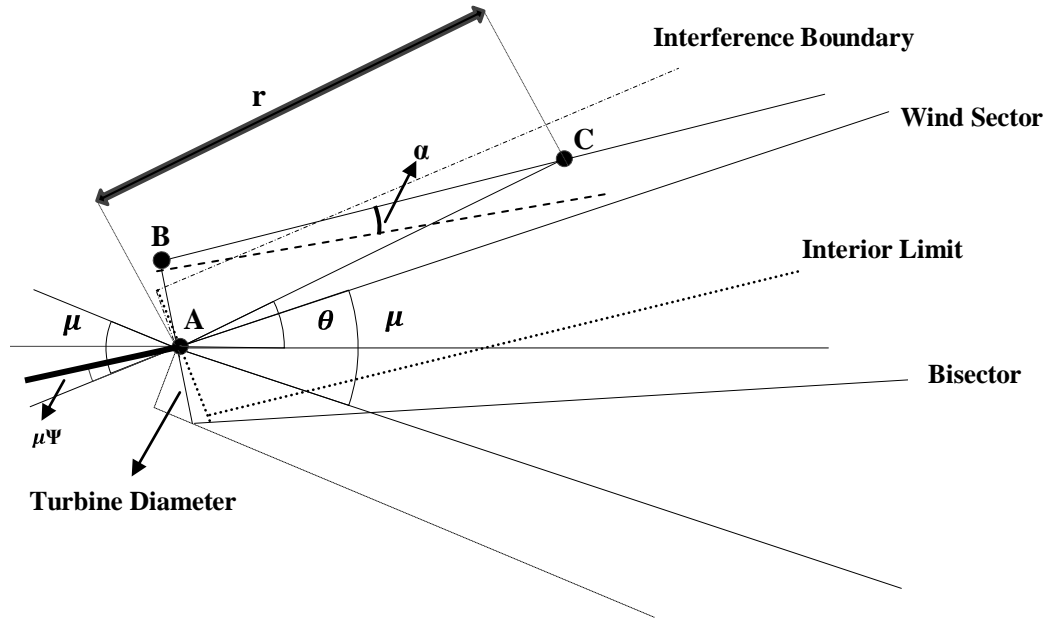


Figure 3. 15. Second Region Schematic

In the above graph a downwind turbine C is placed in a location between interference boundary and interior limit and distance between upwind turbine A and downwind turbine C is denotes by r. The angle  $\theta$  is the angle between downwind turbine and bisector. It is assumed that, the highlighted line in the left of the upwind turbine A is perpendicular to wind turbine diameter. Since ' $\Psi$ ' is a fraction of  $\mu$ , the interference in a wind section is shown by  $\mu\Psi$  and it will occurs if a downwind turbine located at point C. Therefore, to find a formula for ' $\Psi$ ', a relationship should be made between  $\mu$  and ' $\Psi$ ' and other known variables. To begin with, we should use the angle between the bisector line and  $|AB|$ , which is equal to  $|\theta| + \angle BAC$ . Now we have to expand these two terms in order to find a definition for  $\Psi$  in this region.

$$|\theta| + \angle BAC = |\theta| + \pi - (\angle ABC + \angle ACB) = |\theta| + \pi - \left[ \left( \frac{\pi}{2} \right) + \alpha \right] + \angle ACB \quad (3.8)$$

To obtain  $\angle ACB$ , law of sine is used :  $\frac{D/2}{\sin \angle ACB} = \frac{r}{\sin(\pi/2 + \alpha)}$

$$\text{Then we can write: } \sin \angle ACB = \frac{D}{2r} * \cos \alpha \implies \angle ACB = \arcsin \left( \frac{D * \cos \alpha}{2r} \right) \quad (3.9)$$

Therefore; equation (3.8) becomes:  $|\theta| + \pi \left[ \left( \frac{\pi}{2} + \alpha \right) + \arcsin \left( \frac{D \cdot \cos \alpha}{2r} \right) \right]$  (3.10)

If we assume that,  $\alpha + \arcsin \left( \frac{D \cdot \cos \alpha}{2r} \right) = \xi(r)$ , then equation (3.10) will have the following form:

$$|\theta| + \frac{\pi}{2} - \xi(r) \quad (3.11)$$

By using complementary angles which are shown below, we can find the WIIF.

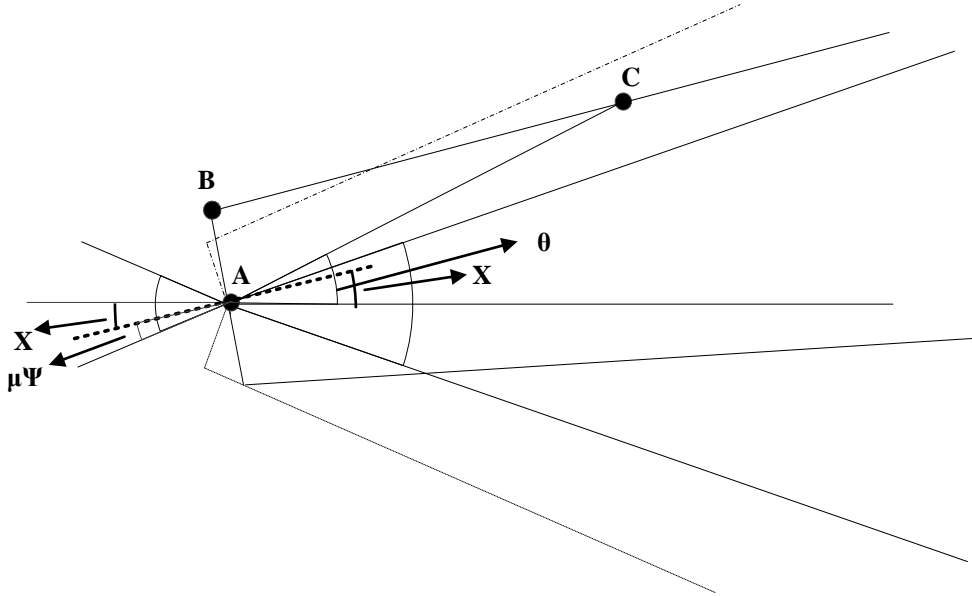


Figure 3.16. Complementary Angles

$$X + \frac{\pi}{2} = |\theta| + \angle BAC, \quad (3.12)$$

$$\text{Where; } X = \frac{\mu}{2} - \mu\Psi \text{ and } |\theta| + \angle BAC = |\theta| + \frac{\pi}{2} - \xi(r) \quad (3.13)$$

$$\text{Therefore; } \Psi(r, \theta)_2 = \frac{1}{\mu} \left( \frac{\mu}{2} + \xi(r) - |\theta| \right) \quad (3.14)$$

The third region is where the downwind turbine is placed in the interior limit at a radius greater than  $r_B$ .

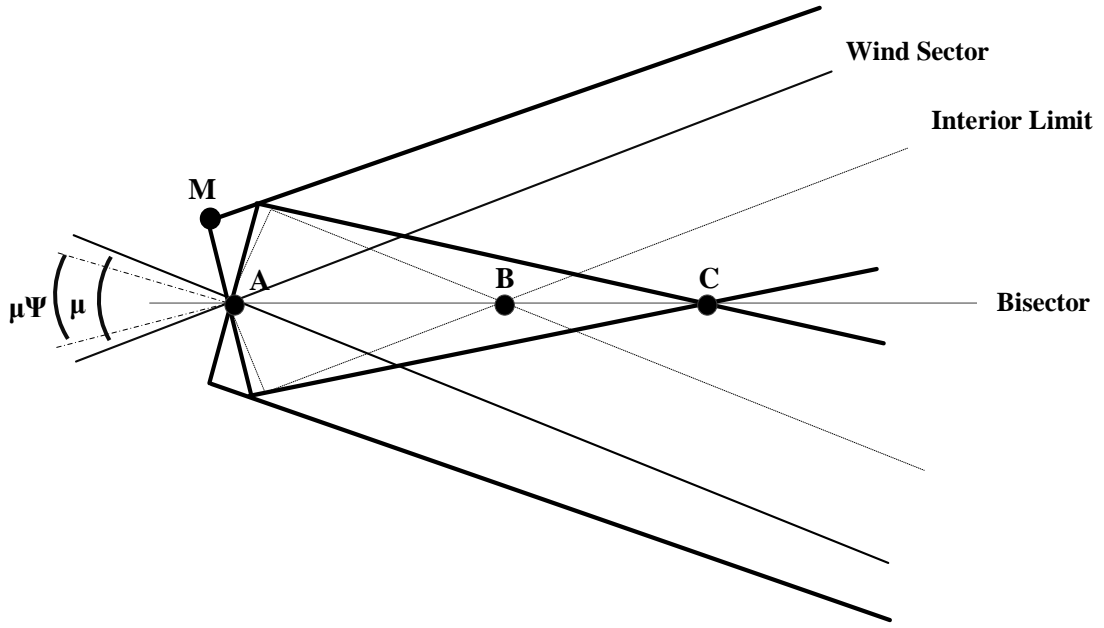


Figure 3.17. Third Region

$$\angle CMA = \pi - (\angle AMC + \angle ACM) = \pi - \left[ \left( \frac{\pi}{2} + \alpha \right) + \left( \arcsin \left( \frac{D \cdot \cos \alpha}{2r} \right) \right) \right] = \frac{\pi}{2} - \xi(r) \quad (3.15)$$

$$\text{We also have; } \frac{\mu\Psi}{2} + \angle CMA = \frac{\pi}{2} \quad (3.16)$$

$$\text{Therefore; } \Psi(r, \theta)_3 = \frac{2 * \xi(r)}{\mu} \quad (3.17)$$

The last region represents a case when downwind turbine is located within the interior boundary at a radius less than  $r_B$ . It means that downwind turbine is too close to upwind turbine that the WIIF takes its maximum value, 1.

To summarize four distinct regions WIIF is defined below.

(3.18)

$$\Psi(r, \theta) = \begin{cases} \frac{1}{\mu} \left( \frac{\mu}{2} + \xi(r) - |\theta| \right) & |\varphi(r)| < |\theta(r)| < |\omega(r)| \quad r \geq \frac{D}{r} \\ \frac{2 * \xi(r)}{\mu} & |\theta(r)| < |\varphi(r)| \quad r \geq r_B \\ 1 & |\theta(r)| < |\varphi(r)| \quad \frac{D}{2} \leq r \leq r_B \\ 0 & \text{otherwise} \end{cases}$$

Where;

$$\xi(r) = \alpha + \arcsin\left(\frac{D \cdot \cos \alpha}{2r}\right) \quad (3.19)$$

$$\varphi(r) = \pm\left(\frac{\mu}{2} - \xi(r)\right) \quad (3.20)$$

$$\omega(r) = \pm\left(\frac{\mu}{2} + \xi(r)\right) \quad (3.21)$$

$$r_B = \frac{D \cdot \cos \alpha}{2 \sin\left(\frac{\mu}{2} - \alpha\right)} \quad (3.22)$$

Calculations of  $\varphi(r)$ ,  $\omega(r)$  and  $r_B$  are describe in appendix part. Variations of WIIF is depicted below for fixed r distance (5m) and fixed  $\theta$ (zero) angle when  $\alpha = 0,075$ ,  $D=1$  and  $\mu = \frac{2\pi}{n} = \frac{2\pi}{12} = \frac{\pi}{6}$ .

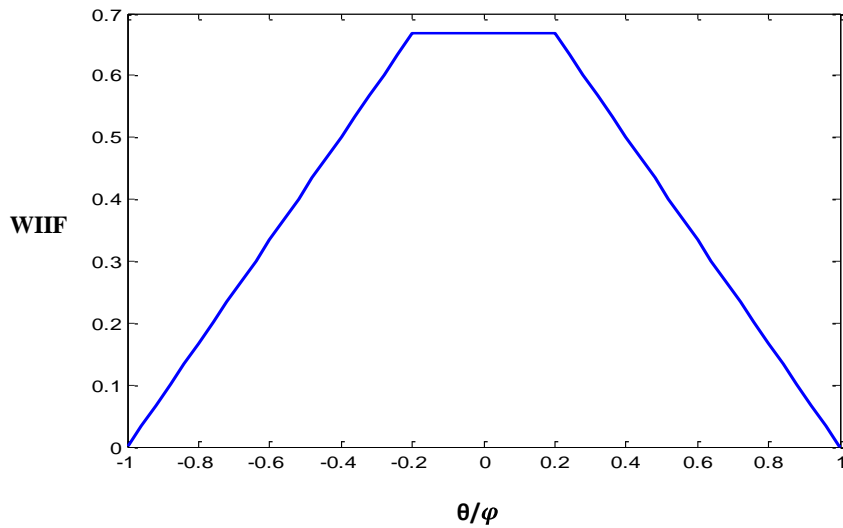


Figure 3. 18. WIIF Variations for Fixed Distance

In this graph, wind interference intensity has its minimum value zero outside of interference boundary and increases linearly to gain its maximum value at the interior limit.



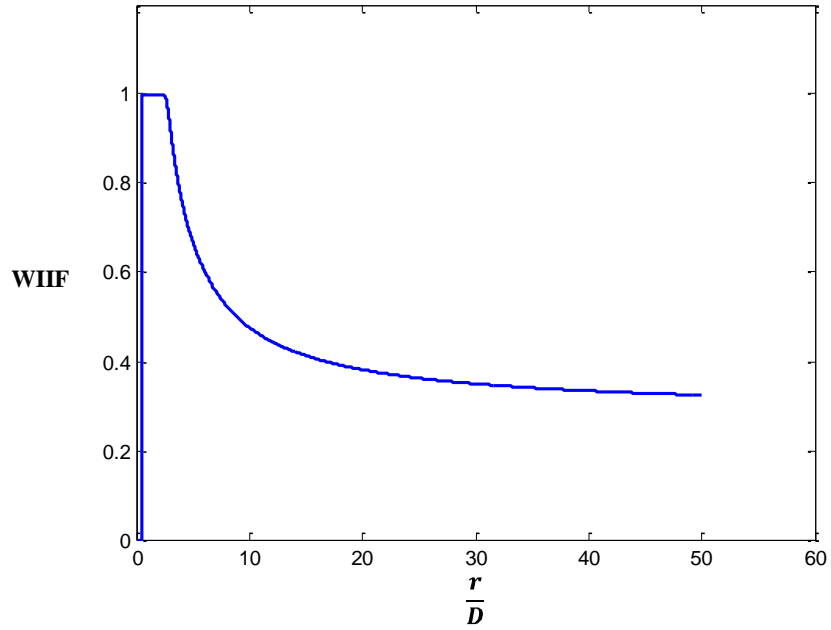


Figure 3. 19. WIIF Variations at a Fixed Angle

In second figure, radial variation of WIIF is drawn for fixed angle. It can be seen that, WIIF reaches its maximum value one when  $r/D$  has its minimum value and WIIF decreases as  $r/D$  gets larger. That means, distance between downwind turbine and upwind turbine increases and therefore intensity of interference reduces.

#### 3.4.4. Interference Coefficient

Interference coefficient indicates the power loss occurred by interference effects between upwind and downwind turbine. Weibull distribution, power curve and wind intensity interference coefficients are used to calculate interference coefficient [7].

$$I_{ij}^d = \Psi(r, \theta) \left( \int_0^\infty P(u_i) F_i^d(u_i) du_i - \int_0^\infty P(u_j) F_j^d(u_j) du_j \right) \quad (3.23)$$

$I_{ij}^d$  indicates the power loss magnitude between upwind 'i' and downwind turbine 'j' for specific wind direction d.  $F_i^d(u_i)$ ,  $F_j^d(u_j)$  are weibull distribution and  $P(u_i)$ ,  $P(u_j)$  are power values for both downwind and upwind wind turbines for wind direction d.

According to previous calculations, the interference coefficient is used to optimize the output power of a wind farm with respect to proximity and interference constraints. To do that, as it was mentioned before, a mixed integer linear programming model is used to find the optimal numbers of wind turbines and also the best locations of them to maximize the captured power.

All possible wind directions are taking into account to find WIIF for each wind sector between wind turbines. Besides that,  $I_{ij}^d$  is calculated for each downwind, upwind turbine pairs to find the total power loss of the wind farm.

### 3.5. Optimization Model

The following presented optimization model shows a maximization formula for wind farm output power which is based on power losses in a wind farm due to interferences between wind turbines. This formula helps to find an overall captured power by using a MILP in order to find the most optimum solution.

**Maximise**

(3.24)

$$\sum_{i \in V, d \in S} w_i^d (P_i^d x_i - Z_i^d)$$

**Subject To**

$$\sum_{i \in q} x_i \leq 1 \quad q \in Q$$

$$I_{ij}^d (x_i + x_j - 1) \leq Z_i^d \quad i \in V, d \in S, j \in L_i^d$$

$$\sum_{i \in V} x_i \leq b$$

$$x_i \in \{0,1\}, Z_i^d \geq 0 \quad i \in V, d \in S$$

$w_i^d$  denotes the time fraction for wind blowing in d direction.  $Z_i^d$  shows the magnitude of power loss for d direction in i downwind turbine because of the interference generated by upwind turbine. Moreover,  $x_i$  is used to show whether there is a turbine at i location or not. If a turbine exists in i location then  $x_i$  is one, otherwise, it is zero. Here, V indicates the set of possible turbine locations and  $L_i^d$  denotes the set of possible turbine locations that would cause interference at i if turbines were placed in both locations for each possible turbine location  $i \in V$  [7]. Q is related to proximity constraint and indicates the collection of maximal sets of turbine locations for which at most one location can be chosen due to the minimum separation distance imposed by the turbine manufacturer [7]. And b denotes the maximum number of turbines that can be placed in the wind farm.

Below is the summary of the optimization process;

- Superimpose a grid onto the wind farm. The intersections of the grid define our set of possible turbine locations (n nodes).
- For every possible turbine location (grid point) calculate the interference intensity factor that a turbine in that location would cause on all other locations.
- MIP will choose best locations for the number of turbines that is specified.
- The rotor blade faces the primary wind direction at all times, so we don't need to worry about the movement of the turbine. We can just assume that whenever the wind is blowing, the turbine will be facing it.
- Calculating power values for each node.
- Wind interference coefficients for each edge connecting each node will be found.
- Repeat the calculations (power, interference etc) 12 times.



## CHAPTER 4

### THE GENETIC ALGORITHM

#### 4.1. Overview

Genetic Algorithms (GAs) were invented by John Holland in the 1960s and were developed by Holland and his students and colleagues at the University of Michigan in the 1960s and the 1970s [8]. Genetic algorithms are a kind of *heuristic* method that is used for solving "search for solutions or "optimization" problems. They are a class of computational models and are based on natural selection and mimic the process of natural evolution to find the optimal solution to a specific problem. Genetic Algorithms are often viewed as function optimizers although the range of problems to which Genetic Algorithms have been applied is quite broad [9]. During the optimization process the best solution for problems is achieved by using the techniques based on inheritance, mutation, selection and crossover.

#### 4.2. Methodology

As it was mentioned in previous part, Genetic Algorithms are adaptive search algorithms that mimic the natural selection and genetic during the implementation to find the optimal answer for a problem. The principles of the GA are inspired by Charles Darwin's theory about evolution and are used in different fields such as Engineering, Computational Science and Physics.

It is started with a population of chromosomes which is randomly generated and is known as a set of solutions. After that, the fitness for every solution in each generation is evaluated and among these solutions, those that have a bigger chance to produce a better solution are used to form a new population. The "goodness" of a solution is typically defined with respect to the current population [9]. After this step, selected solutions are modified to create a new population and this process is repeated until the termination criteria have been reached. Moreover, search space and fitness landscapes are required to define for a typical Genetic Algorithm.

##### 4.2.1. Search Space and Fitness Landscapes

The concept of search space refers to a collection of candidate solutions (feasible solutions) which are best among others for a specific problem. Therefore, every feasible solution is represented by each point in the research space and can be shown by its value or fitness for the problem.

Another important concept is the fitness landscape which is defined as a representation of the space of all possible genotypes (particular set of genes in all chromosomes) along with their fitnesses. Here the fitness measures the quality of the candidate solution and always depends on the problem.

## A Typical Genetic Algorithm

A simple genetic algorithm has five main parts:

1. Creating the initial population
2. Selecting the best chromosomes to be parents for next generation
  - 2.a. Make different combinations from offsprings
  - 2.b. Modify the new offsprings
  - 2.c. Replacing the old generation by the new one
3. Check the solution whether it met the optimization criteria or not?
4. If yes, then terminate the algorithm and get the solution
5. If not, then repeat the process until it reaches the termination point

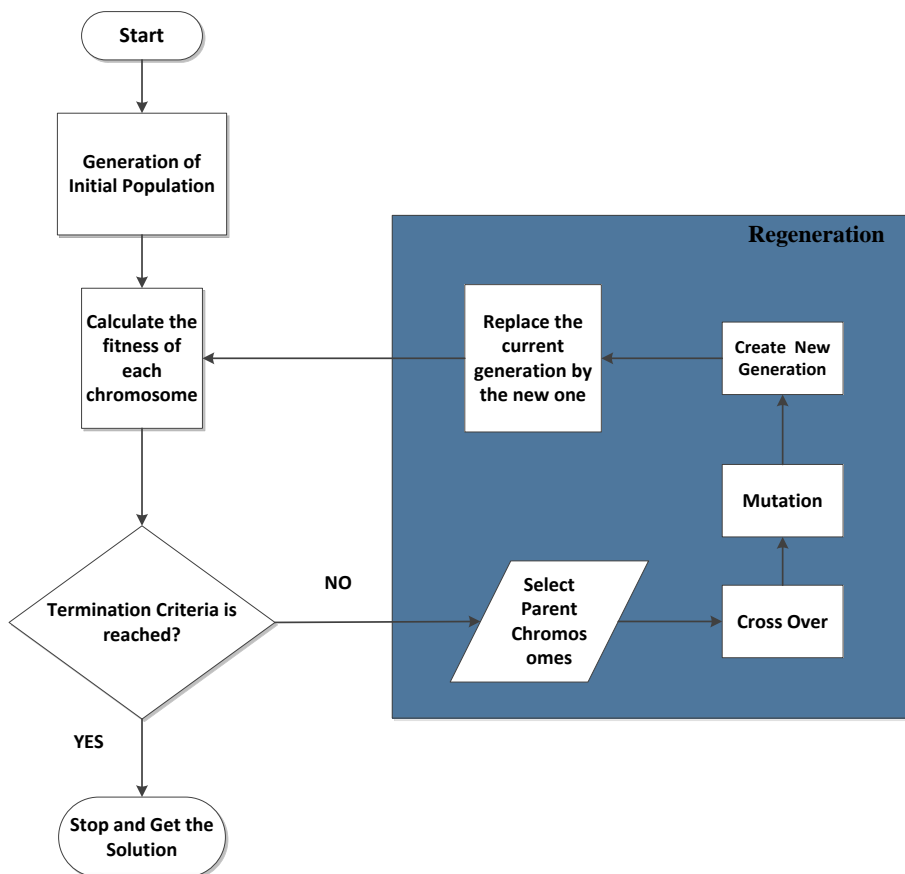


Figure 4. 1. Flow Chart of a Simple GA

### 4.3. Initialization

Initialization process involves defining the necessary variables (Algorithm Parameters) for the optimization such as; population size, crossover and mutation probability. Moreover, an initial population is generated randomly whose dimension is based on the nature of the problem.

#### 4.3.1. Algorithm Parameters

- **The population size (PopSize):** Is the total number of solutions in each generation. The initial population is an important factor because it affects the solution quality [10]. Low number of population size is not suggested for most of the engineering problems, because it decreases the optimization reliability. In other words, as the population size increases, the possibility of finding the optimal solution also increases. It should be considered that initial population is generated randomly and its size is based on the objectives of the problem. For this work, we first generate a population with twice of the size of the initial population and choose the best population among them and create first population. It means, if we want to have one hundred initial populations, we produce 200 populations and then choose the best 100 populations among them. These 100 populations consist of those chromosomes that produce a possible solution according to the constraints of the problem. This method improves the quality of the solution, since number of objective function evaluation increases.
- **Number of generations (NumofGeneration):** In every repetition of the algorithm (generation), some percent of the population is selected from the best population of the previous generation and are transferred directly to the present generation. And the rest of the populations are created with respect to the values of the cross over and mutation rates, elitism rate and others algorithm parameters.  
If the algorithm does not show any improvements in the answer after specific numbers of repetition, it will stop and show the optimum solution. Otherwise, the algorithm is repeated based on the defined repetitions number (Number of generations).
- **Crossover rate (CrossRate):** Is the probability that a pair of chromosomes will be crossed (exchange) to create offspring in order to find a better solution. The cross rate value is typically high about 80%-95%.
- **Mutation rate (MutateRate):** Is the probability that a gene on a chromosome will be modified randomly by adding genetic diversity to prevent from “local optimal traps [11]. In contrast to cross over rate, the mutation rate is generally low about 0.5%-1%.
- **Elitism Rate:** It is a percentage of the best individuals that are directly transformed to the next generation. In other words, elitism rate increases the performance of the algorithm by keep some of the best chromosomes at each generation that may not transmitted or can be damaged during the cross over and mutation processes [8].
- **Break Condition:** The algorithm will stop if there is not any improvement in the answer after specific numbers of repetitions.

#### **4.4. Evaluation**

After defining the genetic parameters, the next step is to evaluate the each chromosome. In every generation, each chromosome must be defined according to the values of the each gene in that chromosome and a selection process is done to find the elite chromosomes.

#### **4.5. Selection**

Another important point in the genetic algorithm process is the selection part. Actually, the selection is related to the fitness of the chromosomes. This selection helps to choose the individuals (parents) that have a good fitness to generate children (offspring) with higher fitness and also determines the number of offspring that will produce. Therefore, chromosomes are chosen regarding their fitness value to be used in crossover and mutation procedures to create a better generation. Different methods are used to implement the selection process such as; Roulette wheel selection, Rank selection, Boltzman selection, tournament selection, rank selection, steady state selection and some others.

In the presented work, parents were selected by using roulette wheel selection scheme, rank selection and elitism were used to complete the selection procedure.

##### **4.5.1. Roulette Wheel Selection**

It is the simplest selection method which represents a wheel with different slices. Each slice denotes the fitness of a chromosome. The bigger is the fitness of the parent, the large the segment of the wheel is. Then the wheel is rotated with respect to the number of individuals in the population. During each spin, the individual is selected to be a parent based on the segment on which roulette wheel stops [8]. As it was mentioned before, the chance of the selection increases if the fitness of the chromosome is high enough. However, during this process some of the chromosomes with high fitness may be lost since some of the individuals may not find their ways to transform in to the next generation [12]. In following graphs, fitness values are shown for five chromosomes and when the wheel is rotating at the selection point, one chromosome will be selected. It can be seen that, chromosome five represents the bigger slice; therefore it has more chance to be selected during the selection process.



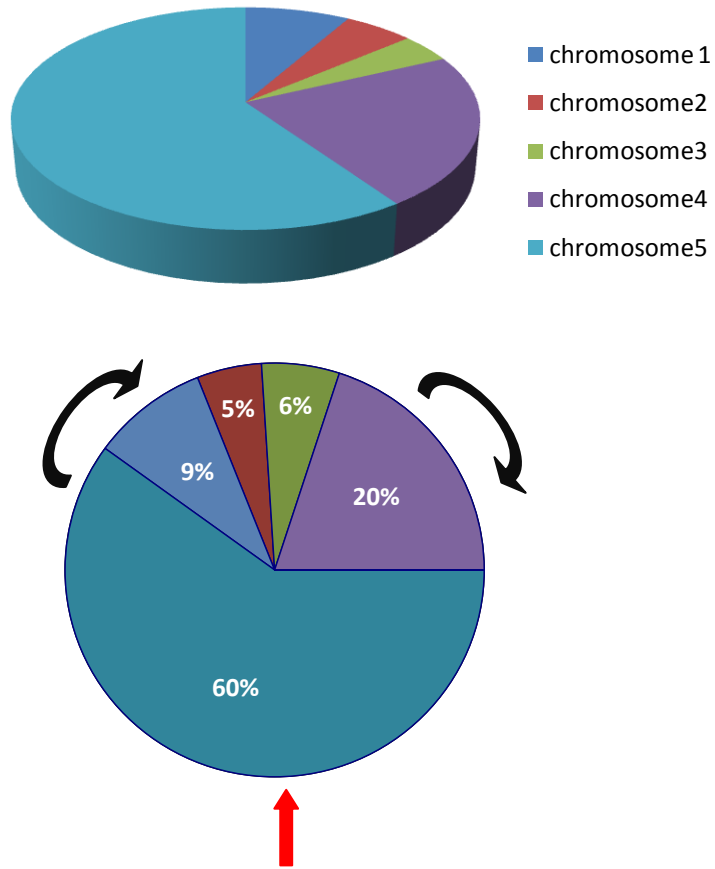


Figure 4. 2. Schematic of Roulette Wheel Selection

The red arrow shows the selection point and wheel is rotating to select the best chromosomes based on their fitness values.

The formulation below shows the average population fitness for the  $i_{th}$  generation. Where  $N$  represent the size of the population [12];

$$\overline{FRW}_{i,j} = \frac{\sum_{j=1}^N FRW_j}{N} \quad (4.1)$$

And,  $PRW_j$  is the probability of the selection of the  $j_{th}$  segment. Here,  $FRW_j$  denotes the fitness of the chromosome  $j$ ;

$$PRW_j = \frac{FRW_j}{\sum_{j=1}^N FRW_j} \quad (4.2)$$

The process of the roulette wheel can be summarizing as following:

1. Find the sum of the all chromosome fitness in the population
  2. Create a random integer between 0 and the sum
  3. Generate a loop based on the sum and the integer value.
- Loop through the chromosomes and summing the expected values. Stop the process when the sum is greater than or equal to the integer. The selected individual is the one which expected value puts the sum above this limit.

#### 4.5.2. Rank Selection

During the roulette wheel selection scheme, some of the chromosomes are not selected when the difference among fitnesses is large. One chromosome may occupy the largest segment of the wheel by 95% while other individuals have only 5% of the wheel. Therefore, they have a little chance to be selected in compare to the other chromosome.

Actually, the rank selection is similar to the roulette wheel but the main point here is that selection is based on the ranking of the population. In this process, the population is ranked based on its fitness value and each individual gets a fitness value after the ranking. The one is assigned to worst value and N is assigned to the best value. Chromosomes are then selected according to their probability selection which is based on relative fitness not the absolute fitness.

Ranking selection prevents the too-quick convergence (premature convergence) by controlling the selective pressure since a uniform method of scaling is used during the selection process [12]. It decreases the selection pressure when the fitness differences are high in early generations. Moreover, the selection pressure is amplified while the fitness differences are low. Therefore, the selection process is done regardless of the absolute fitness differences and all the chromosomes have a chance to be selected. The graph in the next page is depicted the Ranking Selection Method.

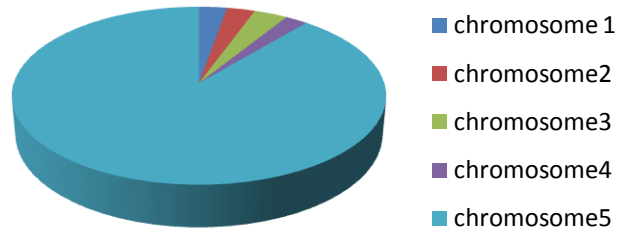


Figure 4. 3. Chromosomes' Fitness Range Before Ranking

Here, the chromosome 5 has a bigger chance for selection. Therefore, other chromosome may be ignored during the selection process. After ranking process, it expected that a better range of fitness occurs between chromosomes as following;

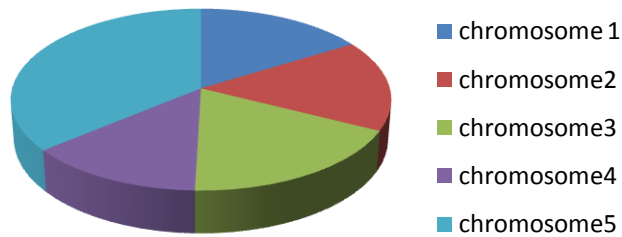


Figure 4. 4. Chromosomes' Fitness Range After Ranking

The formulation below shows the selection probability of each chromosome in the ranking selection [4];

$$PRANK_i = \frac{r_{ij}}{rsum_i} = \frac{r_{ij}}{\sum_{j=1}^N r_{ij}} \quad (4.3)$$

Where,  $i$  varies from 1 to  $n_{gen}$  and  $j$  varies from 1 to  $N$ .

### **4.5.3. Elitism Selection**

As it was mentioned before, some of the best chromosome may be destroyed during the cross over and mutation processes. Therefore, Elitism selection is used to avoid losing of some chromosomes during the selection process. In other words, Elitism lets individuals to not subject to the cross over and mutation in the next generation and remain unchanged. This elitist selection increases the performance of the genetic algorithm while generating the next population.

## **4.6. Genetic Algorithm Operators**

In the next step, it is necessary to create the next generation by using the selected parents. And, to have a better solution two other processes are used here which are cross over and mutation. Cross over create different combinations of the individuals and mutation causes the genetic diversity. The quality of the genetic algorithm is increased by using these operators since the next generation is going to have a better fitness than the previous generation.

### **4.6.1. Cross Over**

Cross over is used to create different combinations of offsprings for the next generation. It chooses different genes from each chromosome in order to create a new child (offspring). The process is done based on the cross over probability we have defined before and is completed by randomly selecting genes of the two individuals from each column and row.

There are four different types of cross over which are; single point cross over, two point cross over, uniform and arithmetic cross over. The uniform cross over was used in this thesis work, in which bits are randomly copied from the first or from the second parent. This process is performed for both rows and columns.

#### **Row Cross Over**

In this step, a random number between 0 and 1 is assumed for each row. If this number is less than the 0.5, then the corresponding row is selected for the offspring. On the other hand, the desired row will be chosen from the second parent.

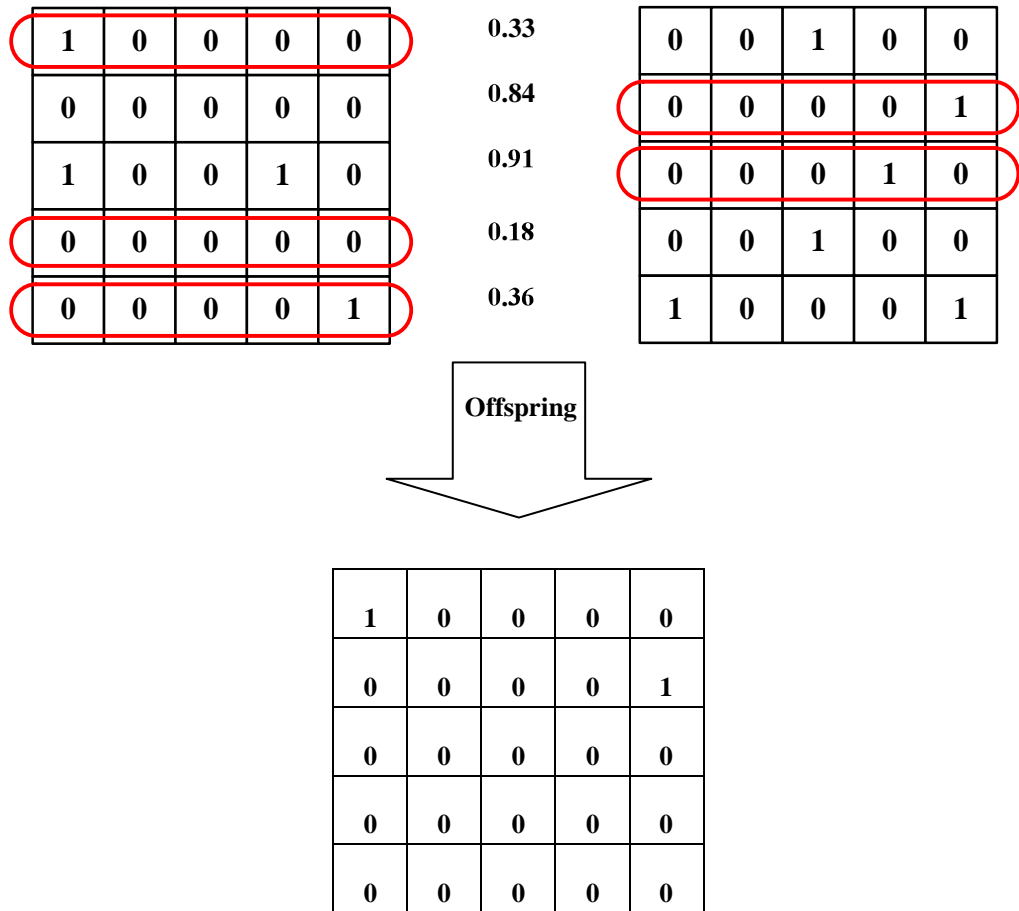


Figure 4. 5. Row Cross Over Schematic

### Column Cross Over

This is like row cross over process except here columns are going to be selected instead of the rows. Random numbers for column are shown in the following table and cross over process is done with respect to these values.

Table 4. 1. Random Numbers for Columns

Column1	Column2	Column3	Column4	Column5
0.75	0.45	0.64	0.53	0.25

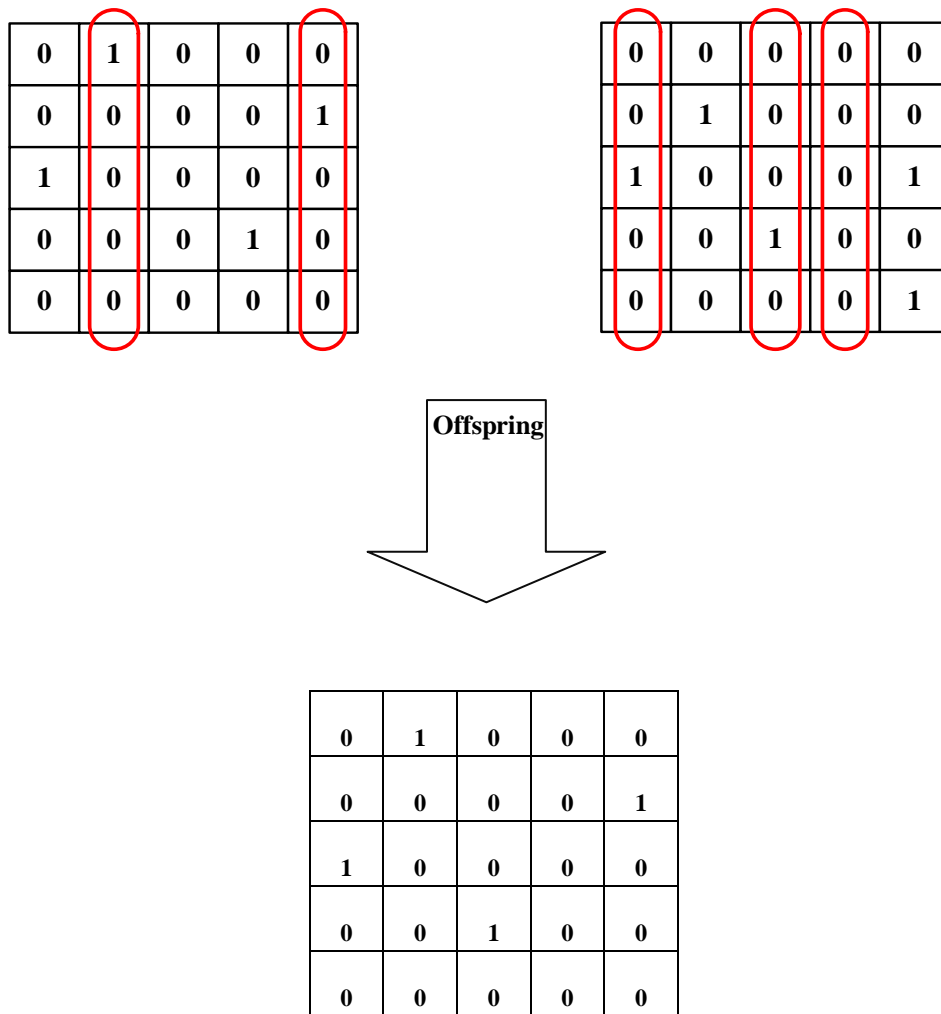


Figure 4. 6. Column Cross Over Schematic

#### 4.6.2. Mutation

The mutation process is used after cross over in order to prevent the loss of any gene during the transformation and helps the solution not to fall in the local optimum. Therefore, it prevents the premature convergence of the algorithm by creating genetic diversity. During the mutation, the gene is mutated base on the mutation probability of the algorithm. For the present thesis the following mutation was used in order to produce new offsprings.

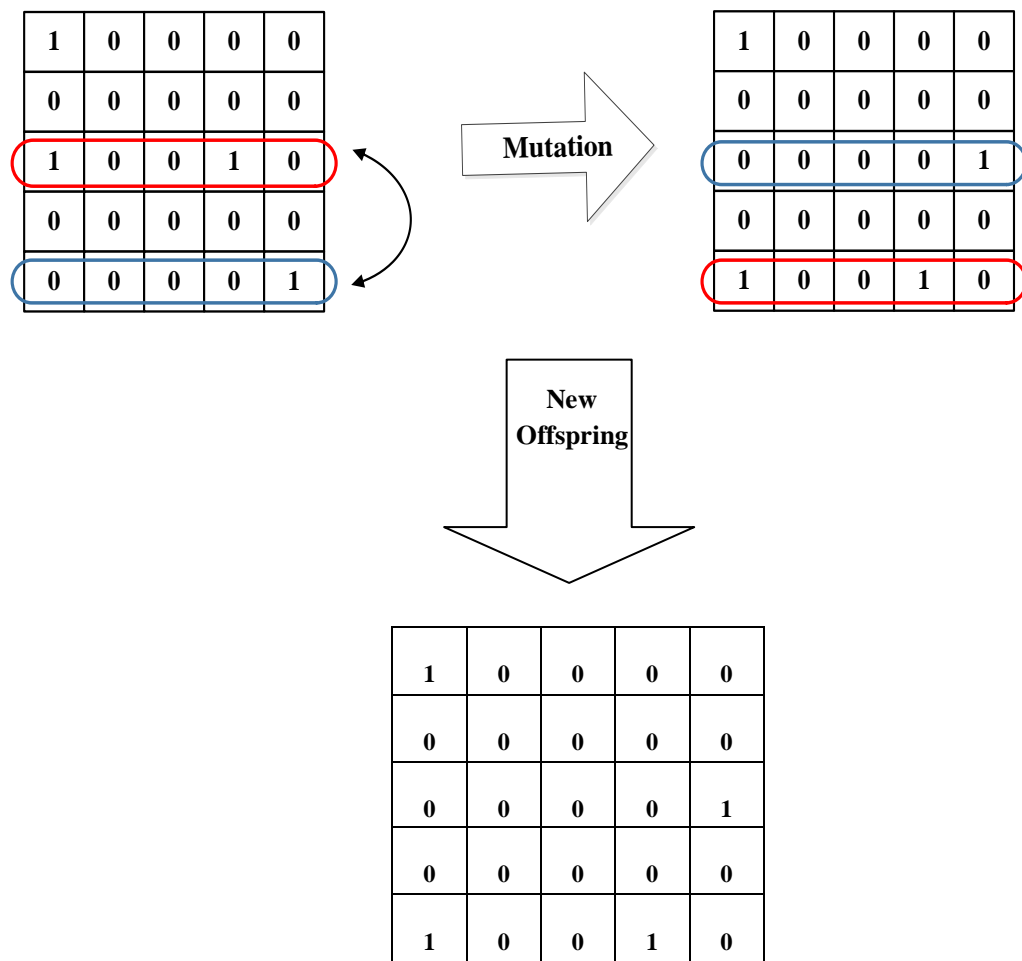


Figure 4. 7. Mutation Process

During the mutation process, two rows or columns of corresponding chromosome are chosen and replaced randomly. If solution is not possible, all genetic algorithm operations will be done again to obtain a possible solution.

#### 4.7. Regeneration

It was mentioned that a new generation is created by using cross over and mutation from the current generation. This new generation will have a better fitness than the previous one because the best chromosomes were selected to produce new offsprings. After each repetition, the old generation is replaced by the new generation which was subjected to selection, cross over and mutation processes. This leads to transform the best solutions from the current generation to a new one; therefore, the objective function is going to have a better value after each regenerating. If any improvements do not occur after definite number of repetitions, the algorithm will stop and shows the optimum answer.

#### **4.8. Termination**

The algorithm will stop after several repetitions when it reaches the termination criteria. User can define a termination criteria based on the problem. For example, if a termination criterion is 60 generation count, it means that GA will stop after 60 generation count if the algorithm gives the same answer for last 60 regenerations. It means there is not any improvement in the answer during 60 regenerations and the best answer is taken after 60 repetitions. For the current thesis, all the required steps were developed in the Matlab. The following flow chart is summarized the written algorithm.

#### **4.9. Encoding**

For the current work results are shown with 1 and 0 to represents the locations where turbine exists or do not exists. Due to the nature of the problem, when a turbine is located in a place it is shown by 1 and when there is not any turbine, 0 is used to.



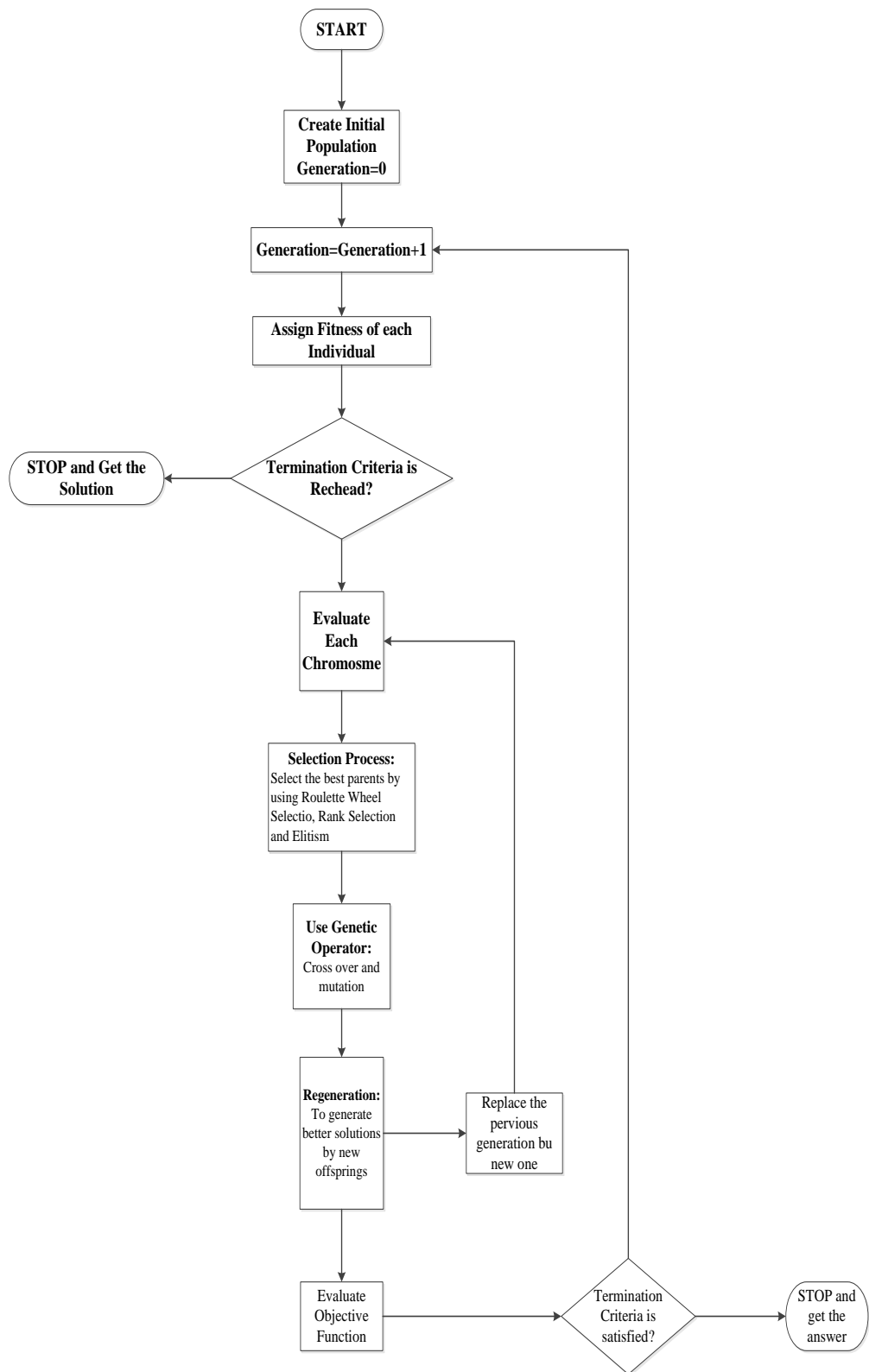


Figure 4. 8. Genetic Algorithm Process

#### 4.10. Summary of the Genetic Algorithm

- Genetic Algorithms are search based methods that mimics the evolution theory
- The main goal is to look forward the best solution by generating different combination and modifications
- Probabilistic rules are used in Genetic Algorithms rather than deterministic ones
- GAs search through a population, not only a single point
- The main things that affect the solution are objective function and fitness values. Derivative and auxiliary information do not require during the process.

## CHAPTER 5

### LINGO

#### 5.1. Overview

LINGO is mathematical language software that was first developed in 1988 by LINDO Systems Inc. It is used to model and solve different types of optimization problems such as linear programming, non linear programming (convex & no-convex/Global), Quadratic, Quadratically Constrained, Second Order Cone, Stochastic and integer programming easier, faster and more efficient [16]. A comprehensive integrated package is implemented in the LINGO to build, express and edit problems for optimization models. Moreover, it contains a set of fast built-in solvers that are used to overcome different problems occurred during the optimizing process. The compatibility problems among the modelling language component and the solver components are minimized in the LINGO since all of the LINGO solvers are directly related to the modelling environment [16].

#### 5.2. LINGO Formulation

Every Lingo code has three main parts; a set section, a data section and the model itself.

##### 5.2.1. A Set

Set is an important part in modelling the optimization problem that helps to define the structure of the data and consists of the groups of related objects. By using the sets, a series of similar constraints can be written in a single statement. Moreover, complex and long formulas are expressed easily and quickly. Each member has several characteristics that are called attributes and can whether be known or unknowns that LINGO solves for [13]. Two kinds of sets are exist; primitive and derived sets.

**Primitive Sets:** They are the main sets in the Lingo model and consist of fundamental objects that cannot be reduced later. These sets can be either implicit or explicit. Implicit form consists of each member that is written individually. However, in explicit term user only enter the first and the last member.

**Derived Sets:** They are dependent sets that are related to pre-existing sets and generated by other components which are called parents (either primitive or derived). Each derived set can be either sparse or dense. In sparse sets, there is only a subset of the parent sets and can be defined either in explicit form or membership filter. The explicit listing includes all the members individually. In the membership filter, a logical condition is used to specify the members. And all the members in this method should satisfy this condition.

Dense sets, on the other hand, consist of all combinations of the members in parent sets.

### **Sets Section:**

Sets section is optional (there may be no sets section) and can be written anywhere in a model. The important point is that the sets section must be defined before the description of the model's constraints occurs. The user has to define all the sets in this section;

- Defining Primitive Sets: A primitive set is specified by;
  - Name of the set
  - Associated objects and their attributes(optional)
- Defining Derived set: It is specified by;
  - Name of the set
  - Description of the Pre-existing Sets
  - Associated objects and their attributes(optional)

### **5.2.2. A DATA Section**

Another section that a Lingo formulation contains is a data section which separates the data part from Lingo model. It whether defines the certain values of the specific sets attributes or shows the place of the data set that Lingo must link to it. By the help of this section, larger models can be scaled and maintained easily.

In some cases problems are self-contained, which means data are directly build in Lingo formulation. However, other types of problems may store necessary data in somewhere else like EXCEL. In this cases a function @OLE ( ) is used to entered those data in to Lingo and also to place them into spreadsheets.

### **5.2.3. A Lingo Model**

It is the main part of the Lingo formulation and defines the mathematical model that is implemented in to Lingo in an optimization process [14]. It consists of three main parts;

**Objective function:** Every optimization problem has a main part that describes the objective of the problem. It is a single formula that defines the model that should be optimized.

**Variables:** There are some quantities in optimization process that affect the optimal value of the objective function. These quantities can be changed in order to find the best solution.

**Constraints:** Constraints limit the variables in the optimization process so that they are not allowed to have some specific values.

## 5.6. Set Looping Function

The main important benefit of Lingo is that it lets the users to define a problem in a single statement rather than writing each constraint individually. Set looping function is implemented to do this process in Lingo in order to make the modelling part easier and more efficient. Moreover, by help of set looping functions, all the members of a set can be iterated during the optimization process. Following functions are use set looping functions in Lingo [14].

1. @FOR; Used to create constraints for a set's members
2. @SUM; Make a summation process for all the members of a set
3. @MIN; Find the minimum of an expression for the set's members
4. @MAX; Find the maximum of an expression for the set's members
5. @PROD; Calculate an expression's product over all members of a set

## 5.7. LINGO Operators

It uses operators that are similar to operators in Excel;

- Addition: +
- Subtraction: -
- Multiplication: \*
- Division: /
- For exponents:  $X^n$
- Equals: =
- Greater than or less than: > or <

## 5.8. Main Features of LINGO

- Problems can be defined in a natural form that is similar to standard mathematical notion
- Constraints of the members of a set can be expressed in a single statement rather than defining them explicitly
- Set-based models in Lingo, let users to create and solve problems involving large numbers of similar variables and constraints
- Since Lingo is case-insensitive, variables can be shown as either lowercase or uppercase
- Lingo can be in contact with outside data sources like spreadsheet, standard text and databases
- Integer programming can be done in Lingo for both linear and non linear constraints
- Lingo allows users to find the solution of several related models in a single run.
- Can model and solve different types of optimization problems such as linear programming, non linear programming (convex & nonconvex/Global), Quadratic, Quadratically Constrained, Second Order Cone, Stochastic and integer programming easier, faster and more efficient

## 5.9. Solver Status Window

During the optimization process, a window is open to show the necessary information about progress of the solver and the dimensions of model.

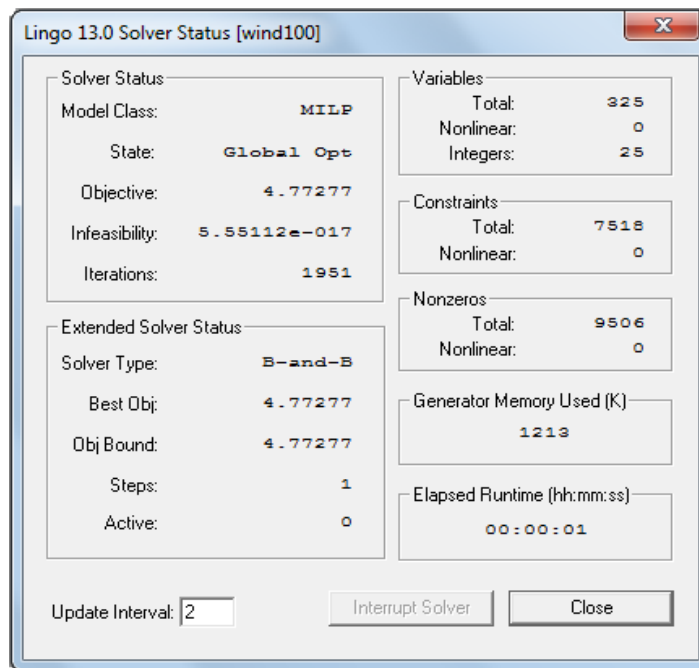


Figure 5. 1. Lingo Solver Status Window

- **Model Class:** Shows the type of the model such as; LP,QP,MILP,MIQP
- **State:** Displays the status of the solver. Can be Global Optimum, Local Optimum, Feasible, Infeasible, Unbounded, Interrupted, Undetermined
- **Objective:** Gives the current value of the objective function
- **Infeasibility:** Shows the amount of constraints that are violated
- **Interrupt Solver Button:** The solver is stopped on the next iteration by using this button and can restore and report the best solution for many cases except for linear programming models.
- **Update Interval Field:** It is related to the update of the solver by Lingo every n seconds and can be changed by user.
- **Variables Box:** Contains nonlinear, integer and total numbers of the variables in the model
- **Constraints Box:** Shows nonlinear and total numbers of constraints in a model
- **Nonzeros Box:** Total number of nonzero coefficients in a model is shown in this box. Actually, it is a counter of the total number of times variable appear in all of the constraints.
- **Generator Memory:** Shows the amount of memory that is used during the solving process and can be changed by user.

- ***Solver Type:*** Indicates the type of solver that is used in the optimization process. It can be Branch-and-Bound Solver, Global Solver and Multistart Solver.
- ***Best Obj:*** Shows the best objective value of the objective function
- ***Obj Bound:*** Gives the theoretical bound value of the objective function
- ***Steps:*** Displays the number of steps that solver is in it
- ***Active:*** It is the numbers of active sub-problems that did not analyzed yet





## CHAPTER 6

### RESULTS

Results are found for different cases by using both Genetic Algorithm and Lingo. But before that, to see the wind direction effect over the optimization, a single wind direction was assumed during the optimization process. After that, two different cases were analysed to optimize the wind farm. In the first case, wind was assumed to have twelve different directions with the same wind speed frequency distribution. In second case however, results were found according to a wind rose includes twelve different directions with different wind speed frequency distribution. A wind farm with a 2.16km\*2.16km dimension was used in the optimization process. According to these dimensions six different grid spacing was analyzed to find the best distance between grid points. Before that, based on proximity constraint maximum numbers of turbines that can be placed in a wind farm were found for each model. It is important to find the maximum number of turbines, because Matlab shows a busy status and does not work if a proper value is not given for maximum turbine numbers. All the optimization processes were done on Matlab and LINGO 13.0 on a 2.26 GHz Packard bell with 4.00 GB RAM.

Table 6. 1. Maximum Turbine Numbers for Various Models

Grid Number	Maximum Wind Turbine
5*5	9
6*6	9
7*7	16
8*8	16
9*9	25
10*10	25

#### **One Direction Wind;**

In this part wind was assumed to blow from WNW direction. Actually, it is not possible that wind blow from only one direction in a long period of time; however, these results are shown only to see the wind direction effect over the power output of the wind farm. Since the wind is blowing from WNW direction, it is expected that wind turbines placed in locations where the wind interference is not high. In the next page, the wind farm schematic is shown for this special case. Distance between possible turbine locations is 3D or 240m since the dimension of the wind farm is 2.16 km\*2.16km

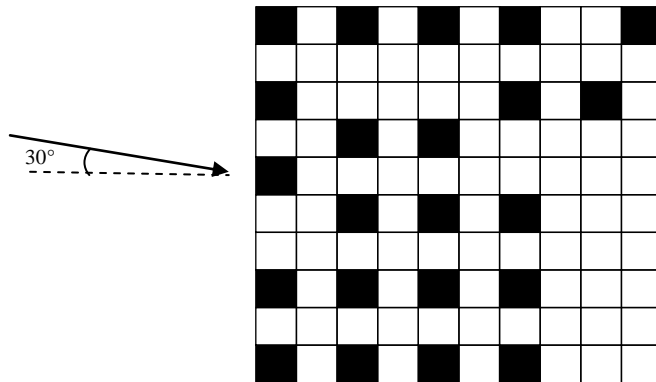


Figure 6. 1. Wind Turbines Placement in the Wind Farm For One Direction

Due to the above graph, wind turbines are more placed in the right side. It is based on the fact that, interference intensity is higher in left side. Therefore, to maximize the power wind turbines are placed in the positions where power loss is not very high. Since in this case wind direction is blowing only from WNW, no other direction affects the power output of the wind turbines. Therefore, it is expected that power loss will not be very high in each possible turbine location. It means a turbine in this case may have only one upwind turbine that affects its overall output power. However, the same wind turbine in other cases may have more than one upwind turbine and it results in a higher power loss and lower power output for that wind turbine. The maximum power production in this case is 29.8773 MWh for 22 located wind turbines. We can see that wind direction has a big effect over the optimization of a wind farm. However, as it was mentioned before, wind direction is not the same over a long time and varies from one direction to several directions.

### 6.1. Case One (Different Wind Directions, Same Wind Speed Frequency Distributions)

In this case optimization was done based on twelve different wind directions with the same frequency distribution. Results were obtained for six different grid spacing by using GA and Lingo. The below table shows the maximum power and run time for GA and Lingo;

Table 6. 2. Optimization Solutions of Different Grid Spacings for Case One

Grid Numbers	Distance btw Grid(meter)	Number of located turbines	Best objective value by GA(MWh)	Best objective value by Lingo(MWh)	Run Time for GA (seconds)	Run Time for Lingo (seconds)
5*5	540	9	14.2189	14.21894	32.831	1
6*6	432	9	9.6070	9.607030	19.8415	1
7*7	360	16	18.3286	18.32863	69.3357	68
8*8	308.57	16	15.2721	15.27206	72.7955	115
9*9	270	25	22.2215	22.22151	181.7229	273
10*10	240	25	19.3130	19.313001	155.4307	9977

It can be seen that GA and Lingo both gives the same answer. However, it is possible that GA gives several different results for each run. Therefore, to find the most optimum answer, user has to run the GA several times for some cases especially for big models. It is due to the nature of the GA, since it is a search base program and generates chromosomes randomly. So, each time different forms of populations may be created and the user has to find the best solution by these various population forms.

On the other hand, Lingo always gives the same answer which is the most optimum result no matter how many times you run it. But, as the model gets bigger the run time for Lingo increases up to 2 or 3 hours while in Matlab it is about a few minutes. From results, it can be concluded that Lingo and GA give the same answer. However, it takes much more time for Lingo to solve the problem especially for large models. Based on the results, it can be concluded that when time is matter, GA is suggested but if the objective is to find the most optimum answer Lingo seems a better program. Moreover, placement of the wind turbines is exactly the same for both programs.

Based on results, the maximum power is obtained when the grid spacing is 270meter which is about 3.375 wind turbine diameter. Although both 9\*9 and 10\*10 grid forms can contain 25 wind turbines, the best objective value for 9\*9 is higher than 10\*10 due to the larger distance between grid points. This conclusion is valid for all grid types. For example, in 5\*5 grid form, spacing is 540m with maximum 9 wind turbines which results in 14.2189 MWh and in 6\*6 grid form, spacing is 432m with 9 maximum wind turbines with 9.6070 power production. Since the grid spacing is larger in 5\*5 grid type, it is expected that the power loss will be in a lower value in comparison to 6\*6 grid form. Therefore, maximum power production is higher in the 5\*5 grid spacing. In the next parts, the placements for 9\*9 and 10\*10 grid shapes will be shown.

### 6.1.1. 9\*9 Grid Form for First Case

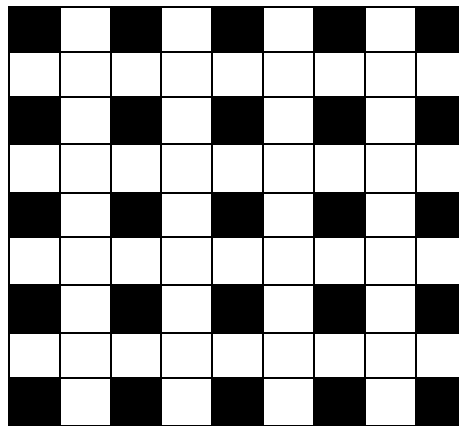


Figure 6. 2. Wind Turbines Placement in the Wind Farm for First Case (9\*9 Grid Form)

The above graph consists of black and white squares which are representation of wind turbine placements. The black squares represents the places where a turbine is located and white squares shows that there is not any turbine in that points. Convergence history (Fitness) shows that after 50 generation count, maximum power is found to be 22.2215 MWh which is the same answer by Lingo.

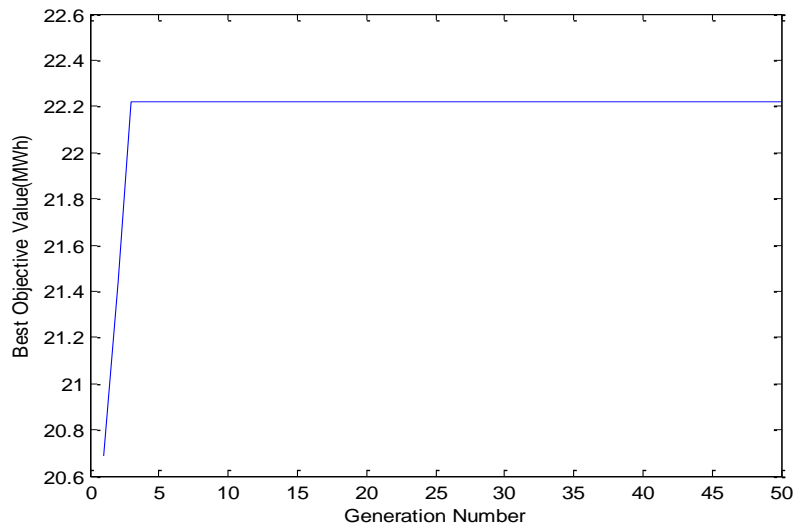


Figure 6. 3. Fitness Graph of 9\*9 Grid Shape for First Case (9\*9 Grid Form)

In the next page, wind farm is shown in x and y coordinates for each wind turbine. Black dots indicate the located turbines and the white dots are the positions where there is not a turbine.

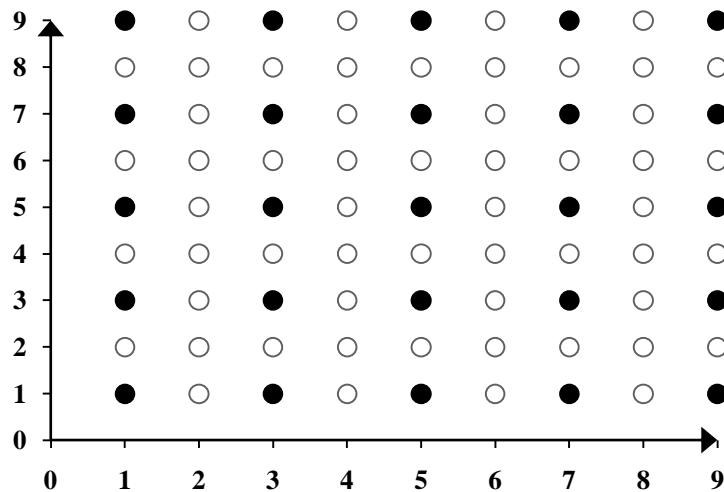


Figure 6. 4. Coordinates of Wind Turbines for First Case(9\*9 Grid Form)

Maximum power production value was calculated for each located wind turbine and is written below;

Table 6. 3. Power Output of Each Wind Turbine for First Case (9\*9Grid Form)

(x,y)	Power(MWh)
(1,9)	1.0972
(3,9)	0.9621
(5,9)	0.9084
(7,9)	0.9673
(9,9)	1.0972
(1,7)	0.9673
(3,7)	0.7717
(5,7)	0.6860
(7,7)	0.7717
(9,7)	0.9621
(1,5)	0.9084
(3,5)	0.6860
(5,5)	0.6502
(7,5)	0.6860
(9,5)	0.9084
(1,3)	0.9621
(3,3)	0.7717
(5,3)	0.6860
(7,3)	0.7717
(9,3)	0.9673
(1,1)	1.0972
(3,1)	0.9673
(5,1)	0.9084
(7,1)	0.9621
(9,1)	1.0972
<b>Total</b>	<b>22.2215(MWh)</b>

Maximum power is obtained in corners of the wind farm which indicates that the interference is lower in comparison with points inside of the wind farm. Since the frequency distribution is equal for all directions, wind turbines are located in a regular shape and power outputs for most points are the same. For example, points (1,9) and (9,1) has the same values.

#### 6.1.2. 10\*10 Grid Form for First Case

This form gives the second highest power output. The spacing difference between this form and the previous form is 30 meters and the power production difference is about 2,9085 MWh. Wind turbine placements is different than the 9\*9 grid form since the number of grid points in increasing in this form. In the next page, the shape of the wind farm is shown.

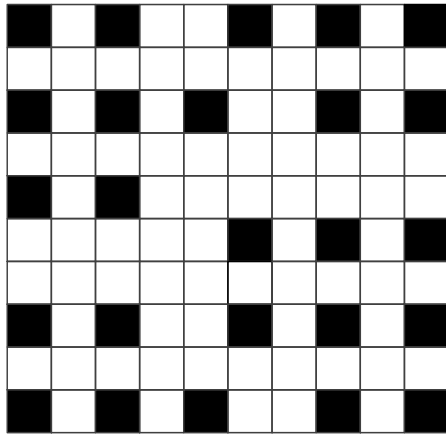


Figure 6. 5. Wind Turbines Placement in the Wind Farm for First Case (10\*10 Grid Form)

The regular form of the wind farm is due to similar wind speed frequency of the wind rose. Wind turbines are placed in a same form especially in corners of the wind farm. In the below graph, the fitness history of the solution is depicted;

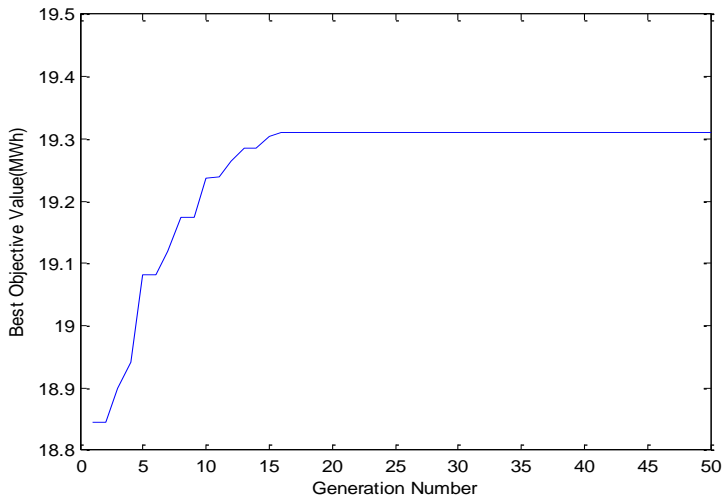


Figure 6. 6. Fitness Graph of 9\*9 Grid Shape for First Case (10\*10 Grid Form)

It can be seen that, the maximum power production of 19.3130 MWh is reached after 50 generation count. Related power outputs for each locate turbine is written in the next part.

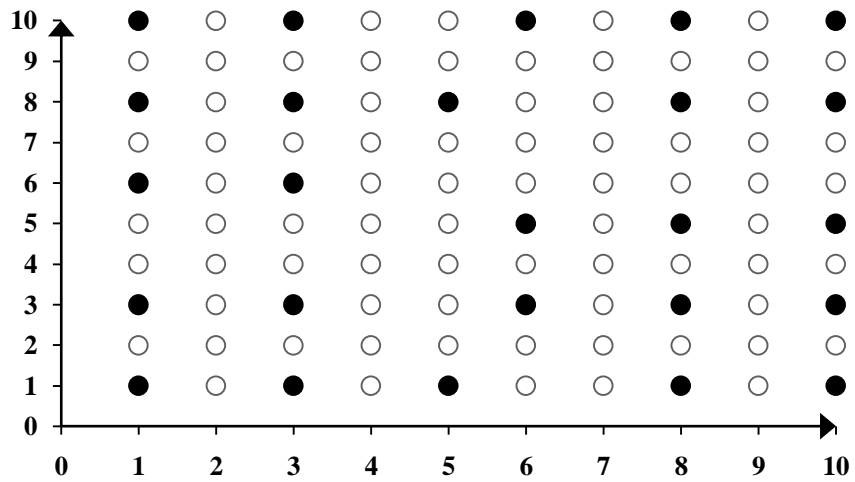


Figure 6. 7. Coordinates of Wind Turbines for First Case(9\*9 Grid Form)

As in the previous part, black dots represent the located turbines. Power productions for these points are shown in the following table;

Table 6. 4. Power Output of Each Wind Turbine for First Case (10\*10Grid Form)

(x,y)	Power(MWh)
(1,10)	1.0487
(3,10)	0.8567
(6,10)	0.7454
(8,10)	0.8736
(10,10)	1.0496
(1,8)	0.8759
(3,8)	0.6576
(5,8)	0.5373
(8,8)	0.6651
(10,8)	0.8528
(1,6)	0.5165
(3,6)	0.7647
(6,5)	0.7591
(8,5)	0.4395
(10,5)	0.5108
(1,3)	0.8528
(3,3)	0.6651
(6,3)	0.5315
(8,3)	0.6576
(10,3)	0.8759
(1,1)	1.0496
(3,1)	0.8736
(5,1)	0.7454
(8,1)	0.8567
(10,1)	1.0487
<b>Total</b>	<b>19.3103</b>

Like in 9\*9 grid form, powers are the same in most of the points especially in the corners. For example, points (3,10) and (8,10) have the same power production. These results occur based on the same wind speed frequency. Since all the directions have the same wind speed frequency, powers repeat themselves for nearly all of the points in the wind farm.

## 6.2. Case Two (Different Wind Directions, Different Wind Speed Frequency Distributions)

In this case optimization was done again based on twelve different wind directions. But the frequency distribution was not the same in this part. Results were obtained for six different grid spacings by using GA and Lingo. The below table shows the maximum power and run time for GA and Lingo;

Table 6. 5. Optimization Solutions of Different Grid Spacings for Second Case

Grid Numbers	Distance btw Grids (meter)	Number of located turbines	Best objective value by GA(MWh)	Best objective value by Lingo(MWh)	Run Time for GA (seconds)	Run Time for Lingo (seconds)
<b>5*5</b>	540	9	15.17	15.16999	31.4355	1
<b>6*6</b>	432	9	11.1526	11.15260	69.4110	1
<b>7*7</b>	360	16	21.4991	21.49905	79.6094	14
<b>8*8</b>	308.57	16	17.1499	17.14994	86.7006	94
<b>9*9</b>	270	25	25.0375	25.03748	204.0375	315
<b>10*10</b>	240	25	21.8320	21.83201	148.4525	11713

### 6.2.1. 9\*9 Grid Form for Second Case

Like the in the first case, the highest objective values are obtained in 9\*9 and 10\*10 grid forms since the maximum located turbines are 25 for both cases. However, the distance between grid points or the possible turbine locations is higher in 9\*9 grid spacing than 10\*10 form. Therefore, maximum power production is higher in 9\*9 form than the others. Therefore, to obtain the maximum power output the distance between grids must be 270m or 3.375 turbine diameter. The wind turbines placement is shown below for 9\*9 grid spacing form in the next page. It is obvious that the wind farm shape in this case exactly the same as in the previous case. However, since the frequency distribution is not the same, the power output for each located wind turbine will not be the same.



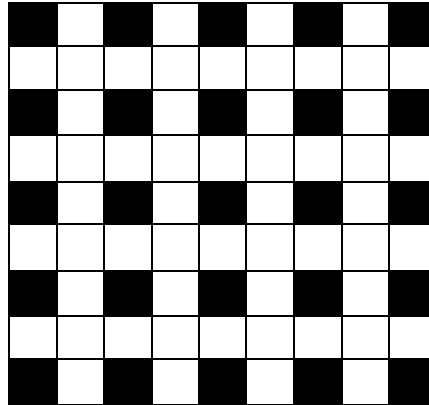


Figure 6. 8. Wind Turbines Placement in the Wind Farm for Second Case (9\*9 Grid Form)

The fitness graph of the third case is drawn for 9\*9 grid spacing form and shown below. The maximum power production of 25.0375 MWh is obtained for 50 generation number.

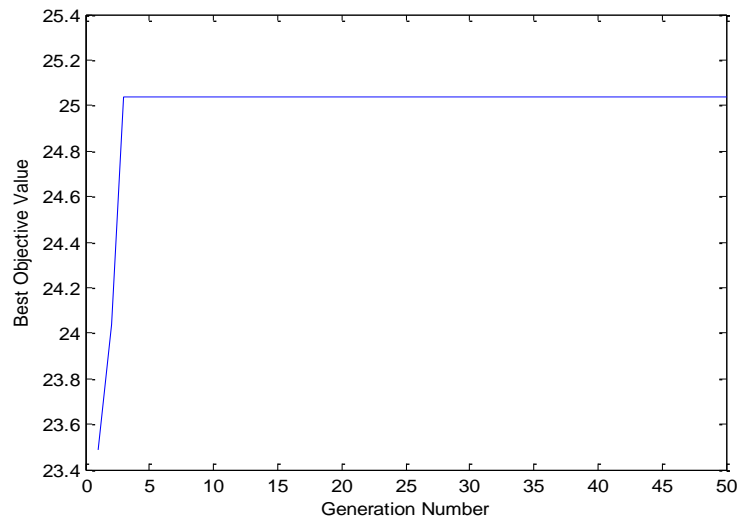


Figure 6. 9. Fitness graph of the Wind Farm for Second Case (9\*9 Grid Form)

Wind farm is in the next page in x and y coordinates for each wind turbine. Black dots again show the positioned wind turbines, while the white ones are for empty places.

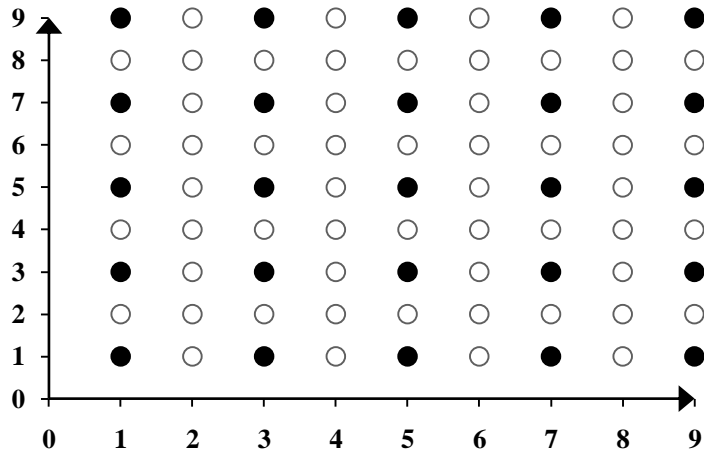


Figure 6. 10. Coordinates of Wind Turbines for Second Case (9\*9 Grid Form)

Maximum power production values were calculated for each positioned wind turbine and are written in table 6.5.

Table 6. 6. Power Output of Each Wind Turbine for Second Case (9\*9 Grid Form)

(x,y)	Power(MWh)
(1,9)	1.2930
(3,9)	1.1174
(5,9)	1.1134
(7,9)	1.1973
(9,9)	1.3406
(1,7)	1.1645
(3,7)	0.8538
(5,7)	0.7753
(7,7)	0.9544
(9,7)	1.1898
(1,5)	1.1695
(3,5)	0.7862
(5,5)	0.7390
(7,5)	0.8687
(9,5)	1.1258
(1,3)	1.2104
(3,3)	0.9264
(5,3)	0.6673
(7,3)	0.7233
(9,3)	0.9784
(1,1)	1.2867
(3,1)	1.0361
(5,1)	0.7794
(7,1)	0.7956
(9,1)	0.9448
<b>Total</b>	<b>25.0375(MWh)</b>

In contrast to first case, in this part, power values varies for each point since the wind speed frequency is not the same for all directions. It can be seen that, the maximum power production occurs in point (9,9) which is due to the high speed frequency in 30 and 60 degree directions. Moreover, it is obvious that for most of the points inside of the wind farm, power values are lower than power values in out border of the wind farm. This is due to the high power loss or high interference effect inside of the wind farm.

### 6.2.2. 10\*10 Grid Form for Second Case

Wind turbines placements for second case are also shown for 10\*10 grid form. The following graph is depicted the wind farm shape for 10\*10 grid spacing. Distance between the grid points is 140 meters and the maximum power production was found to be 21.8320 MWh.

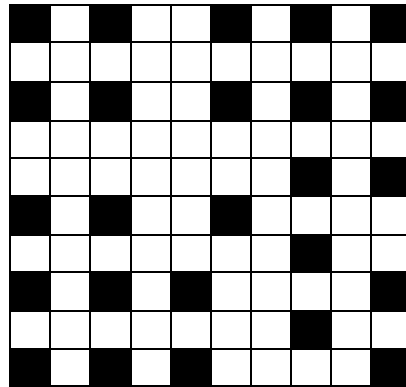


Figure 6. 11. Wind Turbines Placement in the Wind Farm for Second Case (10\*10 Grid Form)

Wind farm here is not as regular as in the previous form since the wind speed frequency is not the same for all directions. Moreover, it is expected that the produced power for each located wind turbine will not be the same for different points. The fitness graph of the solution in the next page shows that the maximum power output is reached after 50 generation count.

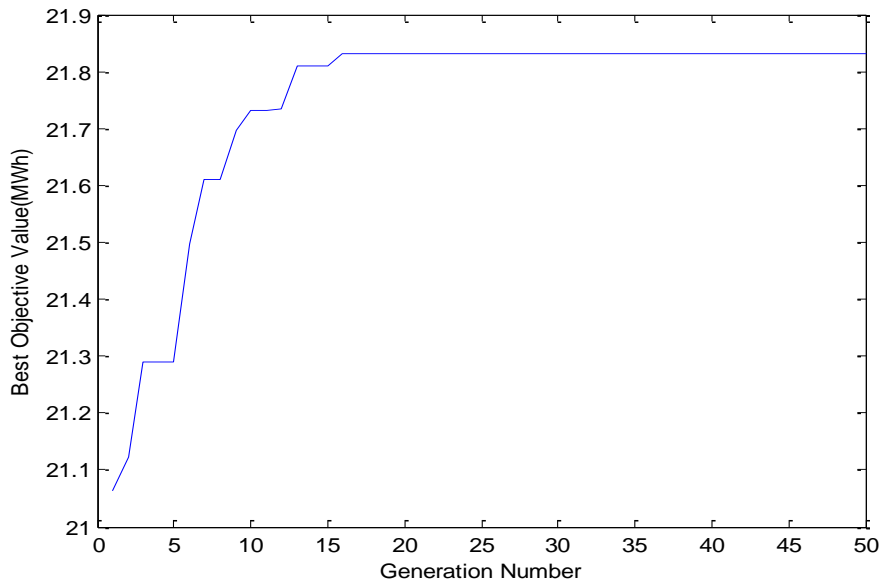


Figure 6. 12. Fitness graph of the Wind Farm for Second Case (10\*10 Grid Form)

In the following part, produced power for each point is analysed. Coordinates of each positioned turbine is shown in below graph.

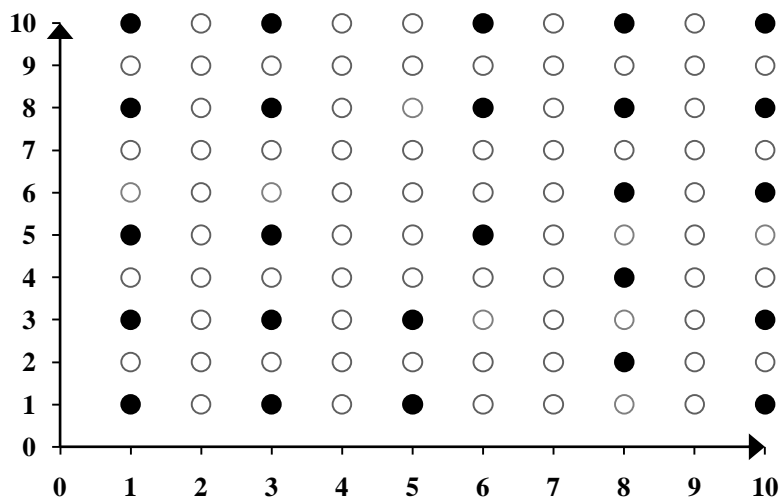


Figure 6. 13. Coordinates of Wind Turbines for Second Case (10\*10 Grid Form)

Table 6. 7. Power Output of Each Wind Turbine for Second Case (10\*10 Grid Form)

(x,y)	Power(MWh)
(1,10)	1.2453
(3,10)	0.9863
(6,10)	0.9409
(8,10)	1.1082
(10,10)	1.2914
(1,8)	1.0640
(3,8)	0.7132
(6,8)	0.6257
(8,8)	0.8218
(10,8)	1.0486
(8,6)	0.5007
(10,6)	0.6518
(1,5)	0.8967
(3,5)	1.0367
(6,5)	0.6243
(8,4)	1.1161
(1,3)	0.7949
(3,3)	0.5230
(5,3)	0.6262
(10,3)	0.8475
(8,2)	1.2491
(1,1)	0.9361
(3,1)	0.6393
(5,1)	0.6560
(10,1)	0.8883
<b>Total</b>	<b>21.8320</b>

It is obvious that powers do not repeat and their values are different for each point since the wind speed frequency distribution is different for each wind direction. The highest value occurs in (10,10) point which normal due to the high speed frequency in 30 (NNE) and 60 (ENE) degrees wind directions.

### 6.3. Conclusion and Recommendations

As a conclusion, numerical solution for micro-siting of the wind farm was found for various models. Lingo and GA gives the same answers; however, solving times for Lingo is much larger than GA. On the other hand, while LINGO guarantees the most optimum solution, to find the best objective value in GA, one has to run it several time. Therefore, if time is matter GA seems a better program than LINGO.

Moreover, distance between wind turbines was changed and different grid spacings were superimposed into the wind farm with 2.16km\*2.16km dimension. Based on the results, as the distance between grids get larger the best object value also increases. However, as the wind farm has a fixed value, there is a limitation on numbers of possible turbine locations.

It means that 5\*5 grid form has the biggest spacing distance among the other forms; however, due to the proximity constraint only 9 turbines can be located in the wind farm. Therefore, the maximum

power output is not as high as in 9\*9 grid form. Therefore, 270m distance between grids seems to be the best form for the optimization of this wind farm.

The optimization was done at first for one direction to better see the effect of wind direction over the power output of the wind turbines. However, since wind speed and wind direction varies over time two other cases were analysed. In the first case frequency distribution was assumed to be the same for all directions and in the second case frequency was not the same. In the first case, power values were the same for most point especially in the corners of the wind farm. However, in the second case power outputs were depend on the frequency distribution and the larger value was in the right top corner.

For future studies, a three dimensional terrain can be analyzed by the presented optimization model. Moreover, different turbine types may be used in a single wind farm to compare the results and find the best wind turbines for a single wind farm.

## REFERENCES

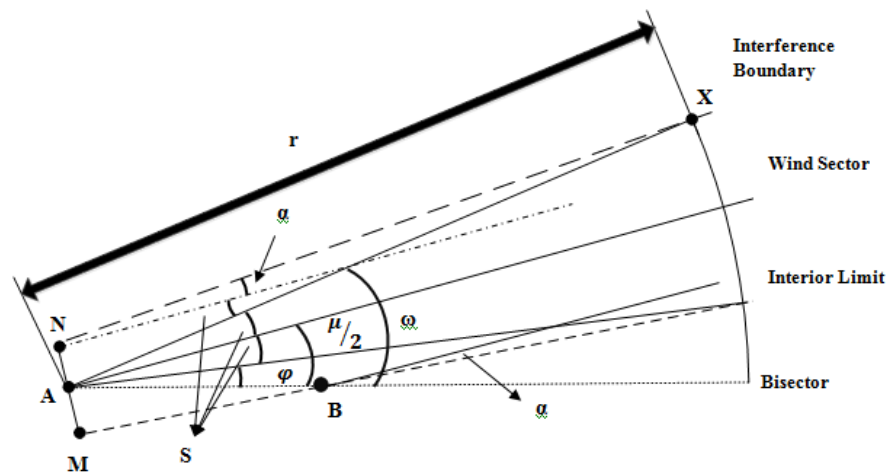
- [1] Michel Carreau, Michael Morgenroth, Oleg Belashov, Asma Mdimagh, Alain Hertz, Odile Marcotte. *Wind Farm Design Optimization*.
- [2] Stuart Burnet Donovan. *Whither the Wind Blows: Wind Flow Modelling and Wind Farm Layout Optimisation*, University of Auckland; 2008.
- [3] Kimberlye. Diamond, Ellen, Crivella. *Wind Turbine Wakes, Wake Effect Impacts, and Wind Leases: Using Solar Access Laws as the Model for Capitalizing on Wind Rights During the Evolution of Wind Policy Standards*; 2011.
- [4] L.J. Vermeer, J.N. Sørensen, A. Crespo. *Wind turbine wake aerodynamics*; 2003.
- [5] Ainslie JF. *Calculating the flow field in the wake of wind turbines*. *Journal of Wind Engineering and Industrial Aerodynamics* 1988;27:213e24.
- [6] Frandsen S, Barthelmie R, Pryor S, Rathmann O, Larsen S, Højstrup J, et al. *Analytical modeling of wind speed deficit in large offshore wind farms*. *Wind Energy* 2006;9:39e53.
- [7] Rathmann O., Frandsen S.T., Barthelmie R.J. *Wake modeling for intermediate and large wind farms*. European Wind Energy Conference and Exhibition, Milan, May 2007, p8.
- [8] Jensen NO. *A note on wind generator interaction*. Risø National Laboratory; 1983.
- [9] F. González-Longatt, P. Wall, V. Terzija. *Wake effect in wind farm performance: Steady-state and dynamic behavior*; 2012.
- [3] Mikel de Prada Gila, Oriol Gomis-Bellmunt, Andreas Sumper, Joan Bergas-Jané. *Power generation efficiency analysis of offshore wind farms connected to a SLPC (single large power converter) operated with variable frequencies considering wake effects*; 2012.
- [4] J. Cole Smith, Z. Caner Taşkın. *A Tutorial Guide to Mixed-Integer Programming Models and Solution Techniques*, University of Florida; March 26, 2007.
- [5] Stuart Donovan. *An improved mixed integer programming model for wind farm layout optimisation*; 2006.
- [6] Stuart Donovan. *Wind Farm Optimization*; University of Auckland.
- [7] Rosalind Archer, Gary Nates, Stuart Donovan, Hamish Waterer. *Wind Turbine Interference in a Wind Farm Layout Optimization Mixed Integer Linear Programming Model*; 2011.
- [8] Mitchell Melanie. *An Introduction to Genetic Algorithms*; Fifth Edition, 1999.
- [9] Darrell Whitley. *A Genetic Algorithm Tutorial*; Colorado State University,
- [10] Pedro A. Diaz-Gomez, Dean F. Hougen. *Initial Population for Genetic Algorithms: A Metric Approach*; University of Oklahoma, Norman, Oklahoma, USA.
- [11] Wen En-Yang Lin, Wen Yung-Lee, Tzung-Pei Hong. *Adapting Crossover and Mutation Rates in Genetic Algorithms*; 2003.
- [12] Rakesh Kumar, Jyotishree. *Blending Roulette Wheel Selection & Rank Selection in Genetic Algorithms*; 2012
- [13] LINDO Systems Inc . *Lingo User Guide*; 2010.
- [14] Kris Thornburg, Anne Hummel LINGO 8.0 Tutorial.
- [15] [http://en.wikipedia.org/wiki/Main\\_Page](http://en.wikipedia.org/wiki/Main_Page)(last visited on 21.01.2013)
- [16] <http://www.lindo.com/>(last visited on 15.01.2013)
- [17] Ivan Mustakerov and Daniela Borissova. *Wind turbines type and number choice using combinatorial optimization*.
- [18] Chunqiu Wan, Jun Wang, Geng Yang, Xiaolan Li and Xing Zhang . *Optimal Micro-Siting of Wind Turbines by Genetic Algorithms Based on Improved Wind and Turbine Models*.





## APPENDIX A

### CALCULATIONS OF $\omega(r)$ AND $\varphi(r)$



A . 1.  $\omega(r)$  and  $\varphi(r)$  Depictions

$$\mu/2 = \alpha + \angle ABM$$

$$\frac{r_B}{\sin BMA} = \frac{AM}{\sin ABM}$$

$$r_B = \frac{AM * \sin(\pi/2 + \alpha)}{\sin(\mu/2 - \alpha)}$$

$$\mu/2 = \alpha + \angle ABM \text{ and } \sin BMA = \sin(\pi/2 + \alpha) = \cos(\alpha)$$

$$r_B = \frac{D * \cos(\alpha)}{2 * \sin(\mu/2 - \alpha)}$$

$$\omega(r) = s + \mu/2$$

$$s = \angle AXN + \alpha = \xi(r)$$

Therefore;

$$\omega(r) = \xi(r) + \mu/2$$

And,  $\varphi(r) = \mu/2 - s$

$$\varphi(r) = \mu/2 - \xi(r)$$

