



**İSTANBUL ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**YÜKSEK LİSANS TEZİ**

**RÜZGAR ENERJİSİNİN OPTİMİZASYONU**

**Elektrik Mühendisi Gönenç AYDIN  
Elektrik-Elektronik Anabilim Dalı**

**Danışman  
Prof. Dr. Mukden UĞUR**

**Ağustos, 2010**

**İSTANBUL**

Bu çalışma ....../....../ 2010 tarihinde ařađıdaki jüri tarafından .....  
Anabilim Dalı ..... programında Doktora / Yüksek Lisans Tezi olarak  
kabul edilmiştir.

Tez Jürisi

Danışman Adı (Danışman)  
İstanbul Üniversitesi  
Mühendislik Fakültesi

Jüri Adı  
Üniversite  
Fakülte

Jüri Adı  
Üniversite  
Fakülte

Jüri Adı  
Üniversite  
Fakülte

Jüri Adı  
Üniversite  
Fakülte

## **ÖNSÖZ**

Yüksek lisans öğrenimim sırasında ve tez çalışmalarım boyunca gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Prof. Dr. Mukden UĞUR'a en içten dileklerle teşekkür ederim.

Bu çalışma boyunca yardımlarını esirgemeyen aileme ve çalışma arkadaşlarıma teşekkürü borç bilirim.

**Ağustos, 2010**

**Gönenç AYDIN**

## İÇİNDEKİLER

ÖNSÖZ .....	i
İÇİNDEKİLER .....	ii
ŞEKİL LİSTESİ .....	vi
RESİM LİSTESİ .....	viii
TABLO LİSTESİ .....	xiv
SEMBOL LİSTESİ .....	xvii
ÖZET .....	xviii
SUMMARY .....	xx
<b>1. GİRİŞ .....</b>	<b>1</b>
1.1. ÇALIŞMANIN KONUSU .....	1
1.2. ÇALIŞMANIN AMACI.....	1
1.3. ÇALIŞMANIN YÖNTEMİ.....	2
<b>2. ENERJİ .....</b>	<b>4</b>
2.1. ENERJİNİN TANIMI .....	4
2.2. ENERJİ BİRİMLERİ.....	6
<b>3. RÜZGAR ENERJİSİNİN OPTİMİZASYONU .....</b>	<b>8</b>
3.1. OPTİMİZASYON .....	8
3.2. OPTİMİZASYONUN FAYDALARI.....	9
3.3. OPTİMİZASYONUN BAZI UYGULAMA ALANLARI .....	10
3.4. TEMEL OPTİMİZASYON ALANLARI .....	11
3.5. ENERJİ SİSTEMLERİNİN OPTİMİZASYONU.....	14
3.5.1. Rüzgar Enerjisi Optimizasyonu .....	15

<b>4. TEORİK ANALİZ .....</b>	<b>16</b>
<b>4.1. BETZ TEOREMİ.....</b>	<b>17</b>
<b>4.2. KAYIPLAR .....</b>	<b>21</b>
4.2.1. Profil Kayıpları.....	22
4.2.2. Uç Kayıpları.....	22
4.2.3. Girdap Kayıpları .....	23
<b>4.3. HELLMAN YÜKSELTME BAĞINTISI .....</b>	<b>25</b>
<b>4.4. WEIBULL DAĞILIMI VE RAYLEIGH DAĞILIM FONKSİYONU.....</b>	<b>26</b>
<b>5. RÜZGÂR ENERJİSİ OPTİMİZASYONU PROGRAMI .....</b>	<b>29</b>
<b>5.1. PROGRAMIN TANITILMASI.....</b>	<b>29</b>
5.1.1. Giriş Ekranı.....	30
5.1.2. Ana Ekran .....	30
5.1.2.1. Rüzgâr Hızı Analizi Ekranı .....	31
5.1.2.2. Güç Analiz Ekranı .....	38
5.1.2.3. Türbin Analizi Ekranı.....	44
5.1.2.4. Maliyet Analizi Ekranı .....	46
<b>6. MALZEME VE YÖNTEM .....</b>	<b>50</b>
<b>6.1. RÜZGÂR HIZI VERİLERİNİN PROGRAM İÇİNDE</b>	
<b>KULLANILMASI.....</b>	<b>50</b>
<b>6.2. ENERJİ ANALİZ EKSPANINDA VERİLERİN KULLANILMASI .....</b>	<b>56</b>
<b>6.3. TÜRBİN ANALİZ EKSPANINDA VERİLERİN KULLANILMASI .....</b>	<b>59</b>
<b>6.4. MALİYET ANALİZ EKSPANINDA VERİLERİN KULLANILMASI.....</b>	<b>64</b>
<b>7. BULGULAR .....</b>	<b>67</b>
<b>8. TARTIŞMA VE SONUÇ .....</b>	<b>72</b>
<b>KAYNAKLAR .....</b>	<b>76</b>
<b>EKLER .....</b>	<b>79</b>
<b>APPENDIX .....</b>	<b>89</b>
<b>ÖZGEÇMİŞ .....</b>	<b>125</b>

## RESİM LİSTESİ

<b>Resim 2.1</b>	: Fırlatılan topun sahip olduğu enerji.....	4
<b>Resim 2.2</b>	: Yuvarlanmak üzere olan kayanın sahip olduğu enerji .....	5
<b>Resim 2.3</b>	: Gerilmiş bir lastiğin sahip olduğu enerji .....	5
<b>Resim 2.4</b>	: Kimyasal tepkime sonucu yanan yakıtın enerjisi .....	6
<b>Resim 4.1</b>	:Türbine ulaşan rüzgar hızının bileşenleri .....	18
<b>Resim 4.2</b>	: Yavaşlatma faktörü değişimi.....	19
<b>Resim 4.3</b>	: Modern rüzgar türbininde $C_p$ 'nin değişimi .....	21
<b>Resim 4.4</b>	: $C_{P_{schmitz}} - \lambda_A$ .....	23
<b>Resim 4.5</b>	: Yüzey farklılıklarının rüzgar hızına etkisi.....	25
<b>Resim 4.6</b>	: Rüzgarın esme sıklığı .....	27
<b>Resim 5.1</b>	: PROWINDANALYSER ana giriş ekranı.....	30
<b>Resim 5.2</b>	: Rüzgâr analizi ekranı 1.kısım.....	31
<b>Resim 5.3</b>	: Rüzgâr analizi ekranı 2.kısım.....	31
<b>Resim 5.4</b>	: Rüzgâr analizi ekranı 3.kısım.....	32
<b>Resim 5.5</b>	: Veri seçim butonları.....	32
<b>Resim 5.6</b>	: Ortalama, maksimum ve minimum rüzgâr hızlarının bulunduğu grup.....	33
<b>Resim 5.7</b>	: Anlık ve esme sayısına bağlı değerler seçim butonları.....	34
<b>Resim 5.8</b>	: Elektrik tüketim hesap bloğu.....	34
<b>Resim 5.9</b>	: Rüzgar hızı verilerine ait kristal raporlama.....	35

<b>Resim 5.10</b>	: Ocak ayına ait raporlandırma sekmesi.....	36
<b>Resim 5.11</b>	: Nisan ayına ait raporlandırma sekmesi.....	36
<b>Resim 5.12</b>	: Hellman yükseltme bağıntısı kullanım arayüzü.....	37
<b>Resim 5.13</b>	: Hellman kısmı ortalama rüzgâr hızı hesabı arayüzü.....	38
<b>Resim 5.14</b>	: Güç analizi ekranı 1.kısım.....	39
<b>Resim 5.15</b>	: Türbin kayıp hesap grubu 1.kısım.....	40
<b>Resim 5.16</b>	: Türbin kayıp hesap grubu 2.kısım.....	40
<b>Resim 5.17</b>	: Güç analizi ekranı 2.kısım.....	41
<b>Resim 5.18</b>	: Güç analizi ekranı 3.kısım.....	41
<b>Resim 5.19</b>	: Güç faktörü, yavaşlatma faktörü ve türbinden çıkan rüzgâr hızı hesap ekranı.....	42
<b>Resim 5.20</b>	: Türbin ortalama rüzgar gücü, güç faktörü, türbin gücü hesap ekranı.....	42
<b>Resim 5.21</b>	: Ortalama rüzgâr hızlarına göre güç hesap grubu.....	43
<b>Resim 5.22</b>	: Esme saati hesap grubu.....	43
<b>Resim 5.23</b>	: Türbin analizi ekranı 1.kısım.....	45
<b>Resim 5.24</b>	: Türbin analizi ekranı 2.kısım.....	46
<b>Resim 5.25</b>	: Maliyet analizi ekranı 1.kısım.....	47
<b>Resim 5.26</b>	: Maliyet analizi ekranı 2.kısım.....	47
<b>Resim 5.27</b>	: Türbin markasına göre türbin fabrika çıkış fiyatı hesap grubu.....	48
<b>Resim 5.28</b>	: Türbin yan maliyet hesap ekranı.....	48
<b>Resim 5.29</b>	: Birim enerji maliyet hesap ekranı.....	49
<b>Resim 6.1</b>	: Rüzgâr hızı analiz ekranı veri tipi seçimi.....	50
<b>Resim 6.2</b>	: Ortalama yıllık elektrik tüketimi tablosu.....	51
<b>Resim 6.3</b>	: Hellman yükseltme bağıntısı ekranı.....	52

<b>Resim 6.4</b>	: Program içine çekilen ölçüm değerleri.....	53
<b>Resim 6.5</b>	: Maksimum, Minimum ve ortalama rüzgâr hızı değerleri.....	53
<b>Resim 6.6</b>	: Yüzey belirleme ekranı.....	54
<b>Resim 6.7</b>	: Seçilen yüzey tipi ve katsayı değeri.....	54
<b>Resim 6.8</b>	: Yükseklik seçim çubuğu.....	54
<b>Resim 6.9</b>	: Seçim kutuları.....	55
<b>Resim 6.10</b>	: İndirgenmiş rüzgâr hızı değerleri ve kayıt onay ekranı.....	55
<b>Resim 6.11</b>	: Rüzgar ölçüm koşulları.....	56
<b>Resim 6.12</b>	: Enerji analizi ekranı yoğunluk, güç faktörü, yavaşlatma faktörü hesabı.....	57
<b>Resim 6.13</b>	: Güç hesaplamaları.....	57
<b>Resim 6.14</b>	: Kayıp hesap grubu ve hesaplanmış kayıp değerleri.....	58
<b>Resim 6.15</b>	: Esme saati verileri hesap grubu.....	59
<b>Resim 6.16</b>	: Bir günlük ortalama elektrik enerjisi ihtiyacı.....	60
<b>Resim 6.17</b>	: Güç-hız eğrisi.....	60
<b>Resim 6.18</b>	: Firmalar, türbin tipleri, rüzgar hızı verileri.....	61
<b>Resim 6.19</b>	: Üretilmesi gereken enerji.....	62
<b>Resim 6.20</b>	: Karşılama oranı hesabı.....	63
<b>Resim 6.21</b>	: Kapasite faktörü hesabı.....	63
<b>Resim 6.22</b>	: Türbin karşılaştırma ekranı.....	64
<b>Resim 6.23</b>	: Birim enerji maliyet hesabı.....	65
<b>Resim 6.24</b>	: Tedaş birim fiyat ekranı.....	66
<b>Resim 6.25</b>	: Geri ödeme süresinin bulunması.....	66
<b>Resim 7.1</b>	: Türbin karşılaştırma ekranı.....	67



<b>Resim 7.2</b>	: V90-1,8 MW geri ödeme süresi.....	69
<b>Resim 7.3</b>	: V90-3 MW geri ödeme süresi.....	70
<b>Resim 7.4</b>	: V80-1,65 MW geri ödeme süresi.....	70
<b>Resim 7.5</b>	: V80-2 MW geri ödeme süresi.....	71

## TABLO LİSTESİ

<b>Tablo 4.1</b>	: Girdap kaybı için $C_{P_{Schmitz}}-\lambda_A$ değerleri .....	<b>24</b>
<b>Tablo 4.2</b>	: Yüzey farklılıklarının rüzgâr hızına etkisi .....	<b>25</b>
<b>Tablo 4.3</b>	: Yıllık esme saati tablosu .....	<b>28</b>
<b>Tablo 7.1</b>	: Karşılaştırma tablosu.....	<b>68</b>

## SEMBOL LİSTESİ

<b>Btu</b>	: İngiliz ısı birimi
<b>kWh</b>	: Kilowattsaat
<b>cal</b>	: Kalori
<b>j</b>	: Joule
<b>m</b>	: Metre
<b>m<sup>2</sup></b>	: Metrekare
<b>m<sup>3</sup></b>	: Metreküp
<b>W</b>	: Watt
<b>kW</b>	: Kilowatt
<b>MW</b>	: Megawatt
<b>GW</b>	: Gigawatt
<b>m</b>	: Kütle
<b>V</b>	: Hız
<b>{d}</b>	: Türev işareti
<b><math>\rho</math></b>	: Yoğunluk
<b>S</b>	: Yüzey alanı
<b>C<sub>p</sub></b>	: Güç faktörü
<b>n</b>	: Yavaşlatma faktörü
<b><math>\lambda</math></b>	: Tasarım devirlik katsayısı
<b><math>\beta</math></b>	: Kanat açısı

## ÖZET

### RÜZGÂR ENERJİSİNİN OPTİMİZASYONU

Enerji, özellikle de elektrik enerjisi insan hayatındaki önemini gün geçtikçe arttırmaktadır. Günümüz koşullarında enerji olmadan hayatlarımızı sürdürmek neredeyse mümkün değildir. Bunun yanı sıra gelişen teknolojiyle birlikte ortaya çıkan çağdaş gereksinimlerden dolayı, her geçen gün artan elektrik enerjisi kullanımı, bizleri yeni enerji kaynakları aramaya itmektedir.

Günümüzde karşı karşıya bulunduğumuz çevre sorunlarının çözümü ve enerji ihtiyaçlarının sürekli olarak karşılanabilmesi için uzun vadeli ve güvenilir kalkınma planlarının yapılması gerekmektedir. Bu açıdan bakıldığında, temiz ve yenilenebilir enerji kaynakları en etkili ve en verimli çözüm olarak karşımıza çıkmaktadır. Yenilenebilir, temiz, ucuz bir enerji olan rüzgâr enerjisi de bu yeni enerji kaynaklarına bir örnektir. Rüzgâr enerjisi çağdaş ülkelerin gündemine 1980’li yıllarda girmiş ve hızla önemini arttırmıştır. Ülkemizde rüzgâr enerjisine dair adımlar atılması 1990 yılların sonlarına doğru olmuştur.

Türkiye, enerjide dışa bağımlı bir ülke olup, enerji ihtiyacının yarıdan fazlasını dışarıdan ithal etmekte ve bu da ülke ekonomisi üzerinde olumsuz etki yapmaktadır. Dolayısıyla Türkiye’nin geleceği için temiz, yerli ve yenilenebilir enerji kaynaklarının kullanımı oldukça önemlidir. Ağustos 2009 itibariyle Türkiye’nin rüzgâr enerjisi 512,85 MW’a ulaşmıştır. İşletmede olan toplam 39 adet rüzgâr santrali bulunmaktadır. Bu sayının gelecekte daha da artması beklenmektedir.

Bu tez çalışmasında, dünyada ve ülkemizde enerji anlamında yaşanan gelişmelere paralel olarak “Rüzgâr Enerjisi Optimizasyonu” konusu ele alınmış ve konu detaylarıyla irdelenmiştir. Tez genelinde rüzgâr enerjisinin temel prensipleri ve teknik alt yapısını oluşturan öğeler üzerinde durularak, rüzgâr enerjisi santrallerinin tercih edilmesinde, kurulumunda, işletilmesinde, türbin tiplerinin belirlenmesinde ve birim enerji maliyetlerinin ortaya koyulduğu maliyet analizlerinin yapılması maksadıyla optimizasyon amaçlı PROWINDANALYSER programı geliştirilerek, bu konuyla ilgili detay çalışmalar program çıktısı olarak sunulmuştur. Özetle, bu tez kapsamında rüzgâr enerjisinin önemi ve bu enerjiden en iyi şekilde nasıl yararlanılabileceği anlatılmıştır.

Tez çalışması sekiz bölümden oluşmaktadır.

Birinci bölümde; problemin tanımı yapılmış, çalışmanın amacı ve yöntemi belirlenmiştir.

İkinci bölümde enerjinin tanımlaması yapıлып, öneminden söz edilmiştir. Enerji birimleri hakkında bilgilendirmeler yapılmıştır.

Üçüncü bölümde; optimizasyon konusu işlenmiştir. Bu kısımda optimizasyonun tanımı, faydaları, uygulama alanları anlatılarak enerji optimizasyonu açıklanmıştır. Ayrıca rüzgâr enerjisi optimizasyonu konusu hakkında genel bilgilendirme yapılmıştır.

Dördüncü bölümde, teorik analiz yapılarak, rüzgâr türbinlerinin temel yasa ve kriterlerinden Betz Kriteri, kayıplar, Hellman yükseltme bağıntısı ve Rayleigh dağılım fonksiyonu açıklanmıştır.

Beşinci bölümde, tezin ana konusu olan rüzgâr enerjisi optimizasyon programının tanıtımı yapılmıştır. Programın kısımları, ekran görüntüleri alınarak gösterilmiş, hangi ekrandan ne tür bilgilere ulaşılabileceği anlatılmıştır.

Altıncı bölümde, Çorlu/Ulaş mevkindeki rüzgâr hızı ölçüm cihazına ait veriler excel dosyası formatında programa alınarak; rüzgâr hızı ekranında, güç analiz ekranında, türbin analiz ekranında ve maliyet analizi ekranında işletilmiştir.

Yedinci bölümde, sekizinci bölümde işlenen verilere ait raporlamalar ve karşılaştırmalar yapılarak enerji üretimi için en uygun türbin ve koşullar belirlenmiştir.

Son bölüm olan sekizinci bölümde, tezin bütününde işlenen konu ve varılan sonuçlar belirtilmiştir. Rüzgâr enerjisinin önemi vurgulanmış ve sınırlı olan enerji kaynaklarının kullanımında, optimizasyonla çok verimli sonuçlara varılabileceği açıklanmıştır.

## **SUMMARY**

### **OPTIMIZATION OF WIND ENERGY**

Energy, especially electrical energy has been increasing its importance in human life day by day. Under the circumstances of today's world, it is almost impossible to continue our lives without it. Besides, the technological improvements appearing with the result of the contemporary reasons, the electrical energy consumption increasing day by day forces us to look for new energy resources.

Achieving solutions to environmental problems that we face nowadays and to provide energy demands sustainably, requires long-term potential actions and trustable plans for sustainable development. In this regard, renewable energy resources appear to be the one of the most efficient and effective solutions. Wind energy is also an example for these renewable, clean and cheap energy resources. Wind energy is realized by modern countries in early 1980's and gained significant importance. Wind energy is noticed on late 1990's in our country.

Turkey, which is tied to other countries on energy, imports more half of its energy need from these countries and this situation has negative impact on the country economy. Because of this reason, using clean, own and renewable energy sources is rather important for Turkey's future. Dating from August 2009, Turkey's wind energy usage has reached to 512,85 MW. There exists 39 wind energy power plants under operation. And it is being expected that the number of wind energy power plants are going to rise in the future.

In this thesis, parallel to the energy improvements taking place on world and our country, "Optimization Of Wind Energy" subject is studied scientifically and examined with its details.

In general of this thesis, it is stated on the principles of wind energy and elements that are the part of its technical underground, as to determine the reasons of preferring wind energy power plants, its setting up, operating of the system, defining the types of wind turbines and preparing the cost analyses for showing the unit energy costs, the optimization programme called PROWINDANALYSER is developed and detailed studies about these subjects are presented as programme printout. On the scope of this thesis wind energy importance and how to get the best utility from this energy are explained.

The thesis is consisting from eight chapters.

In the first chapter, the problem is identified, the aim and the system is determined.

In the second chapter, by making the definition of energy, the importance of it, is explained. Energy units are explained.

In the third chapter, optimization subject is mentioned. At this part by pointing out the description of optimization, its utilities, its application fields, energy optimization is explained. Furthermore general information about wind energy optimization is given.

In the fourth chapter, theoretical analysis is studied. Betz Criteria, losses, Hellmann factor and Rayleigh function are explained.

In the fifth chapter, wind energy optimization programme, which is the main topic of thesis, is demonstrated. Taking the print screens of parts of the programme in order to show, which screen lets us know the necessary information.

In the sixth chapter, datums of the wind velocity logger located on Çorlu/Ulaş are taken as Excel file into programme database and executed in the screen of wind, in the screen of power analysis, in the screen of turbin analysis and in the screen of cost analysis.

In the seventh chapter, reportings and comparisons about the datums executed in the eight chapter, are done. The optimum turbin and conditions are defined as to produce energy.

In the eight and also the last chapter, the main theme of the thesis and the reaching conclusions are explained. The importance of the wind power is emphasized and in the usage of the limited energy sources, applying optimization's importancy is showed.

## **1. GİRİŞ**

### **1.1. ÇALIŞMANIN KONUSU**

Ateşin bulunuşundan bu yana, enerji kaçınılmaz olarak yaşamımızın vazgeçilmez bir parçası haline gelmiştir. Özellikle gelişen teknolojiye paralel olarak, artan nüfusumuz, sanayileşmemiz ve refah düzeyimizin artışı enerji tüketimimizi hızlı bir şekilde arttırmaktadır. Günlük yaşam temposunda o kadar hızlı ve bol miktarda enerji kullanılmaktadır ki; günün birinde bitebileceğini, yaşam döngüsünün bozulabileceğini ve hatta yaşamın sonlanabileceği akıldan geçmemektedir. İhtiyaçlarımızı karşılamak için durmadan enerji kaynakları tüketilmekte ve bu kaynaklar sınırsızmış gibi düşünülmektedir.

Enerjinin kaliteli, ekonomik, güvenilir ve temiz olarak ülkelerin sanayilerinde ve insanların günlük yaşantısında yer edebilmesi için enerji kaynak çeşitliğinin sağlanması, konvansiyonel enerji kaynaklarının yanında, yenilenebilir enerji kaynaklarının kullanıma sunulması ve sürdürülebilir enerji politikalarının hepimiz tarafından benimsenip, bilinçli tüketimin yaşamın her alanına yayılması çok önemlidir.

Ayrıca üzerinde yoğunlaşılması gereken diğer bir konu da dünyanın bilinçsizce seçilen, kullanılan enerji kaynakları sebebiyle hızla kirlenmekte ve dünya üzerindeki yaşamın bundan olumsuz etkilenmekte olmasıdır. Bu konuda atılacak önemli adımlardan bir tanesi; yenilenebilir enerji politikalarını benimsemek, doğadan alınan enerjiyi, ona zarar vermeden kullanıp, geri verebilmektir.

### **1.2. ÇALIŞMANIN AMACI**

Bu tez çalışmasının amacı; yenilenebilir enerji kaynaklarının kullanıma dikkat çekmek, bunlardan en çok kullanılan, dünyada ve ülkemizde de hızla yaygınlaşan rüzgâr



enerjisinin üretiminde, optimal koşulların sağlanabilmesi maksadıyla kilit rol oynayan rüzgâr hızı ölçüm değerlerinin, buna bağlı güç değerlerinin, uygun türbin seçiminin ve aynı şekilde ilgili maliyet analizlerinin oluşturulabileceği bir rüzgâr enerjisi optimizasyon programı geliştirmektir.

Rüzgâr enerjisi kullanımının hızla yaygınlaştığı şu dönemde, ülkemizdeki kaynakların bilinçli ve rüzgâr enerjisi açısından faydalı alanların en uygun şekilde kullanılabilmesinde, konuyla ilgili detay analizlerin ve alt yapı çalışmalarının sürdürülebilmesinde, seçim kriterlerinin projeye başlanmadan önce belirlenmesi büyük önem taşımaktadır. Rüzgâr enerjisi ve diğer enerji kaynakları için de böyle bir konunun araştırılmasının, bundan sonraki çalışmalar ve uygulamalar için yol gösterici ve faydalı olacağı kanısındayım.

Özet olarak bu tezde amaçlanan; geçmişten bu yana enerjiyi kullanmada kat ettiğimiz yol ve kullandığımız kaynaklar hakkındaki bilgilerle rüzgâr enerjisi konusunda, verimli şekilde kullanabileceğimiz sistematik bir çalışma oluşturmaktır.

### **1.3. ÇALIŞMANIN YÖNTEMİ**

Bu tez çalışmasında ana konu “rüzgâr enerjisi” olmuştur. Daha detay konu olarak “rüzgâr enerjisi sistemlerinde optimizasyon” konusuna girilmiştir. Teze, temel enerji bilgileri verilerek başlanmış, enerji türleri ve çeşitleri anlatılmıştır. Rüzgâr enerjisi kısmında dünyada ve Türkiye’de rüzgâr enerjisinin durumu belirtilmiş, rüzgâr enerjisiyle ilgili olarak, programın da alt yapısını oluşturan teorik kısım anlatılmış ve Can Enerji Entegre Elektrik Üretim A.Ş.’ne ait Ulaş mevkinde bulunan rüzgâr ölçüm direğine ait rüzgâr hızı verileri programda kullanılarak ilgili çalışma gerçekleştirilip bu veriler ışığında sonuçlar gösterilmiştir. Rüzgar enerjisi konusuyla ilgili bir çok üniversite kütüphanesinde ve internet ortamında kapsamlı bir literatür araştırılması yapılmıştır. Tez kapsamında geliştirilen program dahilinde yer alan türbin üretici firmalara ait katalog bilgileri incelenip, program içindeki ilgili kısımlarda kullanılmıştır. Geliştirilen program, Visual Studio.Net 2008 ortamında C# diliyle hazırlanmış olup,

veri tabanı olarak M.S.Sql Server kullanılmıştır. Program 4 ana bölümünden oluşmaktadır. Bunlar:

- *Rüzgâr Hızı Analizi*
- *Güç Analizi*
- *Türbin Analizi*
- *Maliyet Analizi*

kısımlarıdır. Bu kısımlarda gerekli analizler yapılarak, bulgu ve sonuç kısmında karşılaştırmalar yapıp, bölgenin rüzgâr enerjisi bakımından durumu, tercih edilebilecek türbin tipleri, güç verileri, kapasite faktör verileri, mali veriler eşliğinde sunulmuştur

## 2. ENERJİ

### 2.1. ENERJİNİN TANIMI

Bir çok kelime gibi enerji kelimesi çok çeşitli anlamlara gelir. Bildiğimiz üzere enerjinin kendisini göremeyiz, koklayamayız veya diğer duyu organlarımızla algılayamayız. Sadece etkilerini sezebiliriz. Bu anlamda, enerjinin bizim kurguladığımız bir meta veya keşfettiğimiz bir kavram olduğunu düşünebiliriz. Ama sonuçlarına baktığımızda, faydalarının günümüz dünyasının şekillenmesine yol açtığını görebiliriz.

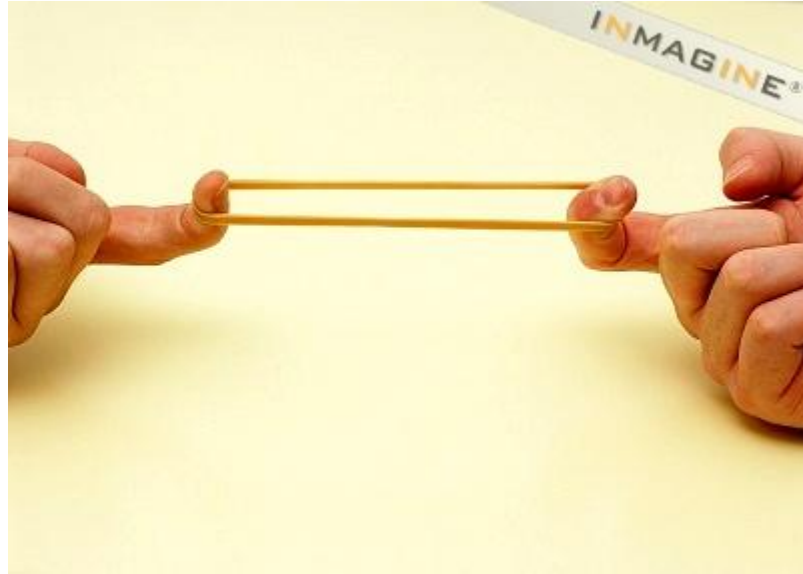
Enerji türlü biçimlerde bulunabilir. Fırlatılmış bir topta enerjinin kinetik hali, uçurum kıyısında yuvarlanmayı bekleyen kayada yer çekimine bağlı potansiyel enerji, gerilmiş bir silgi de elastik enerji, benzinde kimyasal enerji, uranyumda nükleer enerji, sıcak bir patatesten ısı enerjisi ve santrallerden hatlar boyunca evlerimize kadar uzanan elektrik enerjisi, enerjinin belli başlı biçimlerine örnek olabilirler. Enerjinin bu bir yığın çeşidi de birbirine dönüştürülebilir yapıdadır.



Resim: 2.1 Fırlatılan topun sahip olduğu enerji kinetik enerjidir.



Resim 2.2: Yuvarlanmak üzere olan kayanın sahip olduđu enerji, potansiyel enerjidir.



Resim 2.3: Gerilmiş bir lastiğın sahip olduđu enerji, elastik enerjidir.

Bu deęişimler sonrası enerji miktarını ölçerseniz, toplam enerjinin hiçbir suretle deęişmediğini ve korunduğunu gözlemleyeceksiniz. Enerji, enerji türleri arasında geçiş yapabilir de, olduk yere yaratılıp, yok edilemez. Bu prensip belki de fiziğin en temel prensiplerinden birisidir. Bilimsel kavramlardan biri olan enerjinin sakınımı, termodinamiğin ilk kanunudur.

Enerji çevrelerinde daha faydalı bir enerji türü olan kimyasal enerji, daha az faydalı olan dağılmış ısı enerjisiye çevrilir. Bunun nedeni enerjinin kimyasal biçimde tutulmasının sınırlı oluşu ve bu enerji dönüşümlerinin çoğu zaman istenmeyen etkilerinin oluşundan kaynaklanmaktadır. Isıl enerji en az yarar sağlayan enerji türüdür. Yine de diğer enerji türlerine dönüşebilir. Fakat bu dönüşümde muhakkak enerji kayıpları vardır. Ayrıca ısı enerjisi zamanla geniş bir alana dağılma eğilimindedir. Bu enerjinin etki ettiği yüzey alanı genişledikçe, etkinliği aynı oranda azalmaktadır.



Resim 2.4: Kimyasal reaksiyon sonucu yanan yakıtın enerjisi mekiğe itiş gücü sağlıyor.

## 2.2. ENERJİ BİRİMLERİ

Enerji çok değişik birimlerde ölçülebilir; joule, kalori, kilowattsaat, elektron-volt, İngiliz ısı birimi (btu), quad yani kuadrilyon BTU vb...

Görüldüğü üzere bir yığın enerji birimi mevcuttur. Bu birimler arasında farklı çevirme bağıntıları da bulunur. Uluslararası bilimsel çalışma için enerji birimi, *joule* olarak kabul edilmiştir. Joule diğerlerine nazaran daha küçük bir enerji niceliğini temsil eder.

Yavaşça havaya atılmış bir tenis topundaki kinetik enerjiye veya masanızın kenarından düşmeyi bekleyen kaleminizdeki potansiyel enerjiye aşağı yukarı denktir. Belki bu kadar çok birim yerine, tek bir birimle enerjinin ifadesi olsa birçok anlamda enerji algısı kolaylaşacak ve insanların enerjiye bakış açıları da değişecektir. Yine bilimsel çevrelerce çok kullanılan başka bir enerji birimi de, *kaloridir*. Kalori orjinal tanım olarak; 1 gram suyun sıcaklığını 1 derece değiştirebilen ısı birimidir. Bu değişim 4.2 Joule'e karşılık gelmektedir.

$$1 \text{ Kalori} = 4,186 \text{ Joule'dür.} \quad (2.1)$$

Vücudumuzun yiyeceklerden aldığı kalori, genel olarak kilokaloridir. İngiliz birimler sistemin de, kilo kalorinin benzeri, İngiliz termal birimidir. **BTU** Bir pound (0,453kg'lık) luk suyun sıcaklığını 1 derece Fahrenheit değiştiren ısı birimidir. Yani pound kilogramdan küçük ve bir derece Fahreneit ta bir derece Santigrad'tan küçüktür. 1 BTU'da 1 kilokaloriden yaklaşık 1/4 oranında küçüktür, yine yaklaşık olarak 1000 Joule'e denktir.

Elektrik enerjisi yaygın olarak kilowattsaat (kWh) olarak ölçülür. Bir kilowatt-saat tam olarak 3,6 milyon joule bu da yaklaşık olarak 860 kilokalori veya 3400 BTU'ya denktir.

Özetlersek;

$$1 \text{ kcal} = 4186 \text{ j} = 3,97 \text{ BTU} = 0,0011 \text{ kWh} \quad (2.2)$$

$$1 \text{ BTU} = 1054 \text{ j} = 0,252 \text{ kcal} = 0,000293 \text{ kWh} \quad (2.3)$$

$$1 \text{ kWh} = 3,600,000 \text{ j} = 860 \text{ kcal} = 3413 \text{ BTU} \quad (2.4)$$

Bunlar enerji birimleri arasında çok yaygın olarak kullanılan dönüşüm formülleridir.

Akılda kalması açısından şu şekilde bir dönüşüm oranına yuvarlanabilir:

$$1 \text{ kcal} = 4000 \text{ J} = 4 \text{ BTU} = 0,001 \text{ kWh} \quad (2.5)$$

### 3. RÜZGAR ENERJİSİNİN OPTİMİZASYONU

#### 3.1. OPTİMİZASYON

İnsanlar yaşamları boyunca karşılaştıkları sorunları çözüm arayışları zamanla bu çözümleri modeller üzerinde arama yaklaşımını doğurmuştur. Matematik ve bilgisayardaki gelişmeleri dış dünyanın problemlerini matematiksel olarak problemlenip, bu çözümleri modelleyip çözerek bu çözümleri gerçek hayata yansıtma olanağı vermiştir.

Matematiksel modelleme tekniği öncelikle doğrusal ve az sayıda değişkenin kullanılmasıyla başlamıştır. Bir süre sonra doğrusallık varsayımını her problem için geçerli olmadığı anlaşılmıştır. Bu durumda doğrusal olmayan modellemeye gidilmiştir. Ancak doğrusal olmayan modellerin kendine özgü çözümleri uygulamada birçok sorunu beraberinde getirmiştir. Zamanla geliştirilen bazı yöntemlerle doğrusal olmayan modellerin hızla çözümlenmesi sağlanmış ve bu optimizasyon teorisini geliştirmiştir.

Kısacası optimizasyon; bir problemde belirli koşullar altında mümkün olan alternatifler içinden en iyisini seçmektir. Bazı durumlarda herhangi bir nedenle en iyiye ulaşmak mümkün olmayabilir.

Optimizasyon mevcut koşullar altında en iyiyi arama sürecidir. Optimizasyon çalışması, belli kısıtlar altında bir fonksiyonun maksimumunu ya da minimumunu bulmayla ilgilenen bir matematiksel çalışmadır. Optimizasyon modelleri ise bir sistem çıktısını en iyilemek için, sistemin ilişkilerinin matematiksel ifadelerle tanımlanmış biçimidir.

Optimizasyon ilk ortaya çıkışını, 1940'larda George Dantzig'in askeri uygulamalar için (eğitim programları ve zaman çizelgesi) matematiksel teknikleri kullanmasıyla göstermiştir. O tarihten bu yana, doğrusal programlama teknikleri ve uzantıları üretim

araçlarını programlamadan hava yolları yönetimine geniş bir alanda uygulanmıştır. Bugün optimizasyon yönelem araştırması, yapay zeka ve bilgisayar bilimleri gibi bir çok alanda karar vermenin nesnel aracıdır.

Karar değişkenlerinin sadece belli sayıdaki kesikli değerler alabildiği durumlar için ortaya çıkmıştır. Örneğin, bir hastanede personel planlaması probleminde personel sayısı sonlu küme oluşturduğundan optimum personel çalışma planı problem kesikli optimizasyon türünden bir problemdir.

Kesikli optimizasyon, değişkenlerin kesirli değer alabildiği sürekli değişkenlerle optimizasyona nazaran daha zordur. Kesikli optimizasyon problemleri için hızlı ve güvenilir çözüm veren genel bir çözüm yöntemi yoktur. En iyi çözümü elde etmek için birbiriyle rekabet eden çok sayıda yöntem bulunmaktadır. Farklı uygulama alanları/problemleri farklı çözüm yöntemi gerektirir.

Son zamanlardaki eğilim ise bir problem için, tek bir yöntemin etkin olmadığı durumlarda optimizasyon tekniklerinin bileşimini kullanmaktır. Bu teknikler bugün büyük ölçekli problemleri bile yüksek kalite ile çözebilmektedir.

### **3.2. OPTİMİZASYONUN FAYDALARI**

*İşlemsel etkinlik:* Optimizasyonla kaynakların daha iyi kullanımını sağlayacak kararlar verebiliriz. Örneğin, mevcut kaynaklarla üretim miktarını maksimize edebiliriz. Enerji tüketimini azaltabilir, bir işlemi gerçekleştirmek için gerekli işgücünü azaltabiliriz.

*Kar maksimizasyonu ve maliyet minimizasyonu:* Optimizasyon bir işletmede kazancı maksimize etmek ya da maliyeti minimize etmek için kullanılabilir. Örneğin, mevcut kaynaklarla daha fazla üretim gerçekleştirerek, birim başına maliyeti düşürebilir, kısıtları ihlal etmeksizin belirli özellikteki ürünü daha ucuz hammadde ile üretebiliriz.

*Anlama ve performans değerlendirmesi:* Optimizasyon bir sistemin cari performansını mümkün en iyi performansla karşılaştırma imkanı verir. Kaynak kısıtlarının değiştirilmesinin ilave getireceği ya da azaltacağı maliyet hakkında bilgi edinmeye imkan sağlar. Eğer o zaman (what-if) analizi yaparak çeşitli alternatif faaliyet senaryoları üretmeye katkıda bulunur.



### 3.3. OPTİMİZASYONUN BAZI UYGULAMA ALANLARI

Optimizasyona konu olan bazı problemleri bilim alanlarına göre şöyle örneklendirebiliriz:

*Matematik:* Matematikte optimizasyon bir reel fonksiyonu uygun değerler kümesinden belli değerleri sistematik biçimde seçerek maksimize ya da minimize etmeyi ifade eder.

*Bilgisayar bilimleri:* Bilgisayar bilimlerinde optimizasyon, sistemin daha az kaynak gerektirerek daha etkin çalışmasını sağlayacak biçimde modifiye edilmesi prosesidir. İşlem zamanının azaltılması, bant genişliğinin azaltılması, hafıza gereksiniminin en aza indirilmesi, compile edilen program kodunun etkinliğinin artırılması ya da sistemin diğer özelliklerinin en aza indirilmesi / en çoğa çıkarılmasıdır. Örneğin, bir bilgisayar programı daha hızlı biçimde çalıştıracak şekilde ya da daha az bellek gerektirecek biçimde optimize edilebilir.

*İnternet arama motorları optimizasyonu (Search engine optimization):* Arama motorlarından bir web sitesine yönlenen trafiğin hacim ve kalitesinin algoritmalar yoluyla en iyileştirilmesi prosesidir. Tipik olarak, bir sitenin sayfasının arama listesindeki sırasının en başa getirilmesi arzulanır. İnternet pazarlama stratejisinde, arama motoru optimizasyonu (SEO) arama motorunun nasıl çalıştığını ve insanların ne için arama yaptığını dikkate alır. Bir web sitesini optimize etme, temel olarak web sitesinin içeriğini yazma ve HTML kodunun her ikisinin anahtar kelimelerle ilgisini artırma ve arama motorunun indeksleme faaliyetinin önündeki bariyerleri kaldırma hedefine yöneliktir.

*Görüntü arama optimizasyonu (Image search optimization):* İmaj arama motorları üzerinde belli bir anahtar kelimenin ayırıcı özelliğini artırmak için, web sayfasının içeriğini düzenleme prosesidir. Görüntü arama optimizasyonu, arama motoru optimizasyonunun bir alt kümesidir.

*Proses optimizasyonu (Process optimization):* Bazı kısıtları ihlal etmeksizin belirli parametreler kümesini optimize edecek şekilde bir prosesi ayarlama sürecidir. Bu bağlamda en temel hedefler maliyeti en aza indirme, çıktı maksimizasyonu, kar maksimizasyonu vb. Proses optimizasyonunda optimal performansı etkilemek için ayarlanabilecek üç adet parametre vardır:

*İşlem prosedürü:* İşlem yöntemleri kişiden kişiye vardiyadan vardiyaya değişkenlik gösterebilir. Tezgahların otomasyonu değişkenliği azaltmada önemli biçimde yardımcı olabilir. Ancak, kontrol işlemleri operatörler tarafından yapılıyorsa ve tezgahlar el yordamıyla çalıştırılıyorsa otomasyonun yardım gücü azalır.

*Kontrol optimizasyonu:* Kimyasal ya da petrol rafinerisi gibi üretim tezgahlarında yüzlerce kontrol döngüsü olabilir. Her bir kontrol döngüsü belli bir sıcaklığı sürdürme gibi prosesin bir parçasının kontrolü için gerçekleştirilir. Eğer kontrol döngüsü doğru bir biçimde dizayn edilmez ve ayarlanmazsa proses optimum düzeyinin altında çalışır.

*Ürün optimizasyonu (Product optimization):* Belli bir ürünü daha talep edilir hale getirmek için kalitesini iyileştirme bağlamında yapılacak değişikliklerin yöntemleridir. Bir ürün bir çok özelliğe sahiptir. Örneğin bir soda şişesi çeşitli paketleme opsiyonlarına, meyve aromalarına, besin değerlerine sahiptir. Küçük değişiklikler yaparak bir ürünü optimize etmek mümkündür. Amaç, ürünü daha istenir hale getirmek ve “satın alma isteği”, “inandırıcılık”, “satılma sıklığı” gibi bazı pazarlama ölçümlerini artırmaktır.

### **3.4. TEMEL OPTİMİZASYON ALANLARI**

- *Doğrusal programlama (Linear programming):* Amaç fonksiyonu  $f$  ve uygun çözüm kümesi  $A$ 'yı tanımlayan kısıt fonksiyonlarının tamamının doğrusal eşitlik ya da eşitsizlikler olduğu problemleri çalışır.
- *Tamsayılı programlama (Integer programming):* Tüm değişkenlerin tamsayı değerli olması biçiminde kısıtlandığı doğrusal programlama problemlerini çalışır. Değişkenler

doğrusal programlamanın aksine kesiklidir. Bunlar tamsayı değerler veya 0-1 biçiminde olurlar.

- *Karesel programlama (Quadratic programming)*: A kümesinin doğrusal eşitlik ya da eşitsizlik fonksiyonları ile tanımlandığı, amaç fonksiyonunun karesel formda olduğu problemlerle çalışır.
- *Doğrusal olmayan programlama (Nonlinear programming)*: Amaç fonksiyonu ya da kısıtların ya da her ikisinin doğrusal olmayan ifadeler içerdiği problemleri çalışır.
- *Konveks programlama (Convecs programming)*: Amaç fonksiyonunun konveks ve, eğer varsa, kısıt kümesi A'nın konveks küme olduğu durumları çalışır. Bu doğrusal olmayan programlamanın ya da doğrusal veya karesel programlamanın özel bir durumu olarak görülebilir.
- *Yarı tanımlı programlama (Semidefinite programming)*: Değişkenlerin yarı-tanımlı matris oluşturduğu konveks optimizasyonun bir alt alanıdır. Hem doğrusal hem de konveks karesel programlamanın genelleştirilmiş bir biçimidir.
- *Stokastik programlama (Stochastic programming)*: Bazı kısıtların ya da parametrelerin tesadüfi değişkenlere bağımlı olduğu durumlarda çalışır.
- *Dayanıklı programlama (Robust programming)*: Stokastik programlamadaki gibi yerdeki belirsizliği muhafaza eden bir optimizasyon alanıdır. Fakat problem, belirsizliği tesadüfi değişkenler yoluyla ele alarak değil, girdi verisindeki doğru olmamayı hesaba katarak amaç fonksiyonuna ceza fonksiyonları ekleyip çözer.
- *Kombinatorial optimizasyon (Combinatorial optimization)*: Uygun çözüm kümesinin kesikli olduğu veya kesikli hale indirildiği problemlerle ilgilenir.
- *Sonsuz-boyutlu optimizasyon (Infinite-dimensional optimization)*: Uygun çözümler kümesinin sonsuz-boyutlu uzayın bir alt kümesi olduğu (fonksiyonlar uzayı gibi) durumlardır.

- *Heuristik algoritmalar*: Bir çok uygulamada, bir problemin kabul edilebilir çözümünü üreten, fakat bu çözümün doğruluğunun formel (matematiksel) ispatının yapılamadığı algoritmalar. Çözüm belki kesin optimumu verebilir ancak bunun matematiksel olarak ispatı yapılamaz. Bu nedenle, heuristik algoritmalar yakın optimum çözüm verecek şekilde tasarlanır. Heuristik, problemin optimal çözümünün bilinen bir yöntemi olmadığına ya da bilinen bir yöntem olmasına rağmen çözüm zamanının ekonomik olmadığı ve yakın optimal çözümlerin tatmin edici olduğu durumlarda kullanılır.
- *Metaheuristik algoritmalar*: Genel heuristik algoritmalar, arama yöntemlerini rastgele stratejilere dayalı olarak yapan algoritmalar. Genetik algoritmalar, karınca kolonisi algoritmaları, tabu arama algoritmaları vb. bu türden algoritmalar.
- *Kısıt tatmini optimizasyonu (Constraint satisfaction optimization)*: Amaç fonksiyonu  $f^*$  in sabit olduğu, kısıtları sağlayan çözümlerin elde edildiği problemleri çalışır.
- *Ayırıcı programlama (Disjunctive programming)*: Tüm kısıtların değil, en az bir kısıtın sağlanması gerektiği problemleri çalışır. Kısıt tatmini programlamanın bir alt dalıdır.
- *Yörünge optimizasyonu (Trajectory optimization)*: Hava ve uzay araçlarının yörüngesini optimize etmek için kullanılan özel bir optimizasyon alanıdır. Belirlenen performans ölçüsünü (amaç fonksiyonu) belli kısıtlar altında maksimize ya da minimize edecek şekilde, bir yörüngeyi tasarlanmasını çalışır.
- *Değişkenlik hesabı (calculus of variations)* : Fonksiyonlarla ilgilenen sıradan hesabın aksine, fonksiyonlarla ilgilenen matematiğin bir dalıdır. Böyle fonksiyonlar örneğin, bilinmeyen fonksiyonları ve onun türevlerini içeren integral biçiminde formüle edilebilir. Örneğin, iki nokta arasındaki bir eğri için en kısa uzunluğun bulunması probleminde kısıt yoksa tabii ki en kısa yolu bir doğru tanımlar. Ancak kısıtlar mevcutsa, bir çok çözüm olabilir ve en kısa yolu tanımlayanı bulmak zor olabilir. Seçilen güzergahtaki küçük değişimlere karşın, bir amaç fonksiyonunun nasıl değiştiğini

dikkate alarak, zaman boyutundaki bir çok nokta boyunca tanımlanmış amaç fonksiyonunu optimize etmeye ilgilendir.

- *Optimal kontrol teorisi*: Değişkenlik hesabının geliştirilmiş halidir.
- *Dinamik programlama (Dynamic programming)*: Optimizasyon stratejisinin bir problemi daha küçük alt problemlere bölmeye dayalı olduğu durumları çalışır. Bu alt problemleri ilişkilendiren denkleme Bellman Denklemi denir.
- *Denge kısıtlı matematiksel programlama (mathematical programming with equilibrium constraints)*: Kısıtların Değişken eşitsizlikler ve tamamlayıcılar içerdiği durumlarla ilgilendir. Mühendislik tasarımlarında, çok düzeyli oyunlarda ve ekonomik denge problemlerinde kullanılır.

### 3.5. ENERJİ SİSTEMLERİNİN OPTİMİZASYONU

Günümüz dünyasında yaşanan petrol ve doğal krizleri, enerji anlamında girilen dar boğazlar, artan talep, sürekli değişen fiyat oranları ve çevresel tahribatlar nedeniyle, yerli ve yenilenebilir enerji kaynaklarıyla birlikte bunların verimli kullanımının önemi giderek artmaktadır. Özellikle yeni kurulan santraller ve elektrik dağıtım, iletim sistemlerinin bunlara göre revizyonu durumlarında tüm bu sistematığı etkileyen girdilerin ve içinde bulunan koşulların son derece iyi optimize edilmesi gerekmektedir.

Özellikle enerji kaynakları bakımından dışa bağımlı bir hale gelmemek için, üreteceğimiz enerji miktarından, tüketicilere kadar geçilen her kısımda verimliliği artırıcı çalışmalarda bulunmamızı kaçınılmaz kılmaktadır. Yeni kurulan santrallerin maliyetleri düşünüldüğünde, bunların geri dönüş sürelerinin belirlenmesinde optimizasyonun nedenli büyük bir rol üstlenebileceğinin bir göstergesidir.

Tezimiz kapsamında, kullanacağımız optimizasyon yöntemi; proses optimizasyonu içinde yer almaktadır. Bir girdiyle başlayan ve bu girdiye katma değer katılarak belirli bir çıktı üreten birbiriyle bağlantılı etkinlikler dizisi olan bu optimizasyon yöntemiyle, enerji üretim sürecine etki eden girdiler (rüzgar hızı, hava yoğunluğu, ortam sıcaklığı, türbin yüksekliği, zemin türü) ve ilerleyen aşamalarda, bu veriler ışığında bulunan ortalama ve toplam güç verileri, kayıp güç verileri, türbin tipi verilerinin irdelenmesi sonucu ulaşılan maliyet analizleri ve verimlilik verileriyle kurulacak ya da kurulması planlanan sistem hakkında kritik yorumlamalar yapılabilir.

### **3.5.1. Rüzgar Enerjisi Optimizasyonu**

Rüzgar enerjisi sistemlerinin kurulmasından devreye alınmasına kadar geçen süre zarfı büyük önem arz etmektedir. Daha öncesinde rüzgar santralının kurulması için planlanan bölgede yapılacak rüzgar hızı belirleme çalışmaları (saatlik, günlük, aylık, dönemlik, yıllık), bölge zemin etüdlerinin yapılması tüm bu çalışmaların temelini oluşturmaktadır. Optimizasyon çalışmasına da bu aşamada rüzgar hızı verilerinin incelenip, bölgenin rüzgar haritası çıkartılarak, maksimum, minimum, ortalama rüzgar hızı değerleri tespit edilerek, rüzgar enerjisi kullanılarak üretilebilecek maksimum, minimum ve ortalama rüzgar güçleriyle, saatlik, günlük, aylık, dönemlik ve yıllık değerler hesap edilerek başlanabilir. Burdan elde edilecek sonuçlarla, bölgenin rüzgar hızı ve dolayısıyla enerjisi bakımından verimli veya kısır bir bölge olduğunun tespiti yapılabilir.

Bundan sonraki aşamada, eğer istenen hesaplama değerlerine ulaşılmışsa, bölgenin fiziksel alt yapısına uygun ve hesaplama değerleriyle örtüşen türbin tipi belirlenip, kataloglar veya program dahilinde varsa türbin üreticileri veri tabanından bu standartları sağlayan türbin üreticileri ve ilgili türbinleri seçilip, kıyaslamak üzere çalışmalar yapılır.

Son aşama olarak da, maliyet analizleri gelmektedir. Çünkü enerji sistemlerinin değerlendirilmesinde, birim enerji maliyeti en önemli parametrelerden bir tanesidir. Bunların yanında işletme maliyeti olarak kabul edebileceğimiz, bakım-onarım maliyetleri, sigorta maliyetleri de optimizasyon çalışma çerçevesinde incelenmelidir.

Tüm bu analizlerden çıkan sonuçlar eşliğinde, ilk adım da bölgeye rüzgar enerjisi sisteminin kurulması durumu irdelenerek, uygun verileri sağlayan noktalar belirlenip, sistemi taşıyacak türbinler belirlenerek ilgili maliyet analizleriyle beraber rüzgar enerjisi sisteminin optimizasyonu sağlanmış olacaktır.

## 4. TEORİK ANALİZ

Yatay eksenli rüzgar türbinlerinin performansları Betz teorisiyle ifade edilir. Bu temel formül ilk defa Alman mühendis Albert Betz tarafından 1919 yılında ortaya kondu. Bu formül yatay ve düşey eksenli rüzgar türbinlerinde uygulandı.

### 4.1. BETZ TEOREMİ

Bu teoremden Betz rüzgar rotorunun ideal olduğu varsayılır. Başka bir deyişle rotor havaya karşı sürüklenme direnci göstermeyen sonsuz sayıda kanattan oluşmaktadır.  $V$  sabit hızında hareket eden havanın kinetik enerjisindeki değişimi güç olarak formülize edersek:

$$P = dE/dt \quad \dots(4.1)$$

olur.

Kinetik enerji olarak ifade edersek:

$$E = \frac{1}{2} \cdot m \cdot V_r^2 \quad \dots(4.2)$$

olur.

Eşitlik (4.2) 'yi eşitlik (4.1) 'de yerine koyarsak:

$$\begin{aligned}
P &= dE/dt \\
&= d/dt ( \frac{1}{2} \cdot m \cdot V^2 ) \\
&= \frac{1}{2} \cdot m \cdot V_r^2 \qquad \dots (4.3)
\end{aligned}$$

elde edilir.

Buradaki kütleli debi:

$$m = \rho \cdot A \cdot V_r \qquad \dots(4.4)$$

bağıntısıyla verilir. Buradan hareketle rüzgarın gücü,

$$P = \frac{1}{2} \rho \cdot A \cdot V_r^3 \qquad \dots(4.5)$$

olarak hesaplanır.

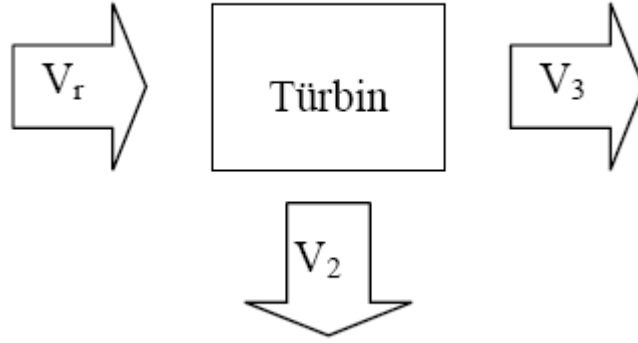
Birim analizi yaparsak,

$$\text{kg/m}^3 \cdot \text{m}^2 \cdot \text{m}^3/\text{s}^3 = \text{kg} \cdot \text{m/s}^2 \cdot \text{m/s} = \text{N} \cdot \text{m/s} = \text{Watt}$$

olur.

Rüzgar, kinetik enerjisi nedeniyle doğal bir potansiyele sahiptir. Bunun bilinen fiziksel konular ve teknolojik imkanlar sayesinde yararlı enerjiye çevrilen miktarına **rüzgar enerjisi teknik potansiyeli** denir.





Resim 4.1 Türbine ulaşan rüzgar hızının bileşenleri.

Buradan türbin gücünü hesaplamak istersek:

$$P_T = \frac{1}{2} \cdot \rho \cdot A \cdot V_2^2 \cdot (V_r^2 - V_3^2) \quad \dots(4.6)$$

Froude- Rankie teoreminden ortalama türbin içindeki rüzgar hızı,

$$V_2 = (V_r + V_3) / 2 \quad \dots(4.7)$$

olarak hesaplanır.  $V_2$ , (4.6) numaralı eşitlikte yerine yazılırsa türbin gücü,

$$P_T = \frac{1}{2} \cdot \rho \cdot A \cdot [(V_r + V_3) / 2]^2 \cdot (V_r^2 - V_3^2) \quad \dots(4.8)$$

Ya da

$$P_T = \frac{1}{2} \cdot \rho \cdot A \cdot V_r^3 \cdot \frac{1}{2} \cdot (1 + V_3/V_r) \cdot (1 - (V_3/V_r)^2) \quad \dots(4.9)$$

Burada (4.5) numaralı eşitlik (4.9) numaralı eşitlikte yerine yazılırsa:

$$P_T = P_r \cdot \frac{1}{2} \cdot [(1 + V_3/V_r) \cdot (1 - (V_3/V_r)^2)] \quad \dots(4.10)$$

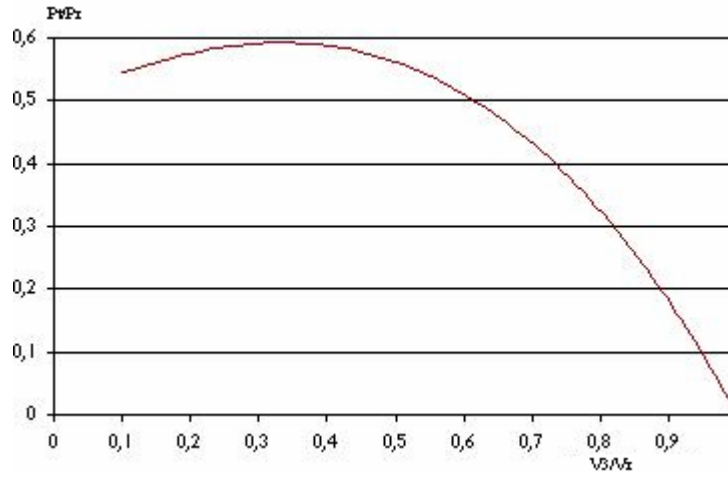
bulunur. Burada güç faktörü  $C_p = P_T / P_r$  ve yavaşlatma faktörü de  $n = V_3 / V_r$  olarak tanımlanırsa (4.10) numaralı eşitlikte ;

$$C_p = \frac{1}{2} \cdot (1 + n) \cdot (1 - n^2) \quad \dots(4.11)$$

şekline dönüşür. Bu denklemde maksimum güç faktörü  $C_p$ 'yi bulmak için yavaşlatma faktörü  $n$ 'in türevi alınıp sıfıra eşitlenirse,

$$\begin{aligned} d C_p / dn &= \frac{1}{2} \cdot d/dn (1 - n^2 + n - n^3) \\ n_1 &= 1/3, \quad n_2 = -1 \end{aligned}$$

bulunur. Yavaşlatma faktörüne ait grafik aşağıdaki şekilde verilmiştir.



Resim 4.2: Yavaşlatma faktörü değişimi ([www.windpower.dk](http://www.windpower.dk)).

Yavaşlatma faktörü  $n$ , hiçbir zaman negatif değer olamayacağından  $1/3$  alınır ve (4.11) eşitliğinde yerine konursa;

$$\begin{aligned} C_{pmax} &= \frac{1}{2} \cdot (1 + 1/3) \cdot (1 - 1/9) \\ C_{pmax} &= 0,5926 \end{aligned}$$

Serbest pervaneden alınacak maksimum verim olarak bulunur. Yani bir rüzgar türbiniyle, rüzgarın tüm enerjisinden faydalanabilmek mümkün değildir. Rüzgar, rüzgar türbininden, rüzgardan aldığımız enerji ölçüsünde yavaşlamış olarak çıkar. Eğer rüzgardaki tüm enerjiyi alabilseydik, rüzgar türbininden durgun halde çıkması gerekirdi.

Fakat bu durumda da türbine rüzgarın diğer taraftan girmesi engellenir ve hiç enerji elde edilemezdi (Ç.ŞEN, 2003).

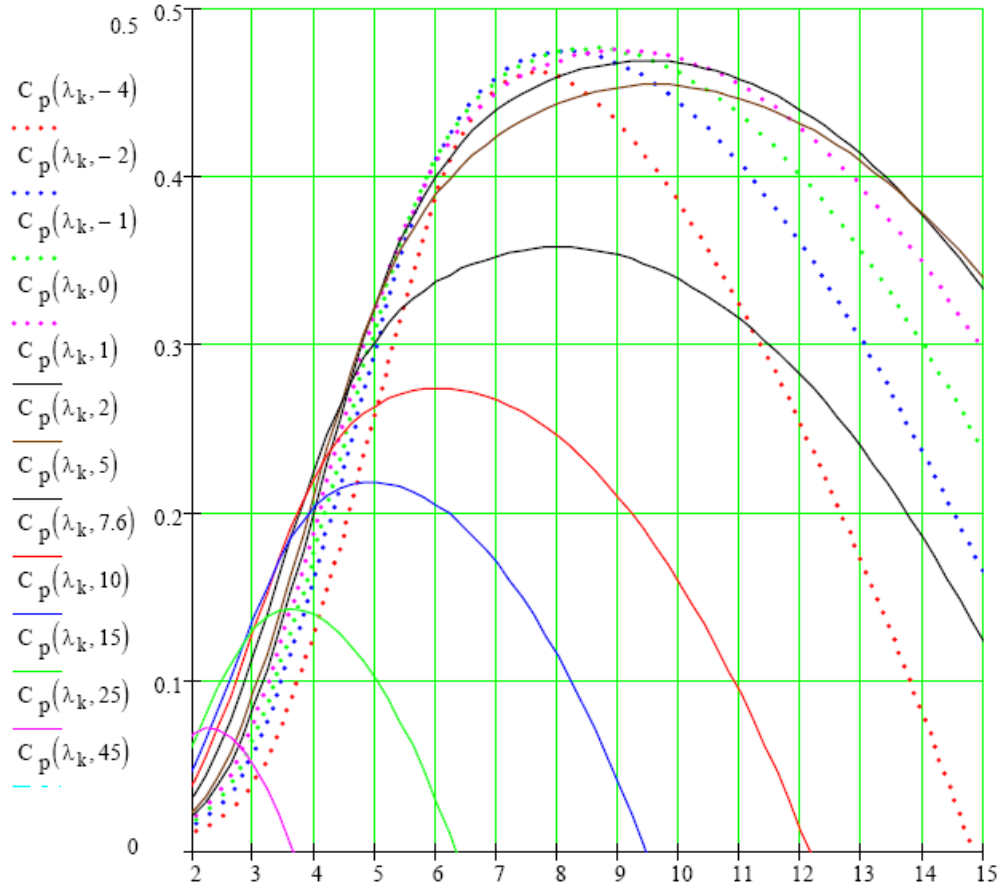
Çağdaş rüzgar türbinleri için  $C_p$  değeri yaklaşık olarak 0,40'tır. Bunun nedeni hava direnci, rotorun oluşturduğu türbülans ve aktarma organları ile elektrik sistemi gibi noktalardaki kayıplardır.

Belirttiğimiz  $C_p$  kendi başına bir değişken olmayıp,  $\lambda$  dediğimiz tasarım devirlik katsayısı olarak ifade ettiğimiz, hareket halindeki havanın kanat profiline göre döndürme hızını ifade eder.

$$P = \frac{1}{2} \cdot \rho \cdot \pi \cdot R^2 \cdot V^3 \cdot C_p$$

olarak ifade edebiliriz. Rüzgar hızı ve hava yoğunluğu kontrol edemediğimiz faktörlerdir. Fakat kanat yarıçapı değişmez bir niceliktir. Tork kontrolü rüzgar türbinleri için kaçınılmazdır. Güç faktörü bu kontrol için yegane kontrol büyüklüğüdür. Bazı rüzgar türbinlerin de, kanatlar aşırı rüzgar hızlarına karşı  $C_p$  büyüklüğünü düşürecek şekilde tasarlanırlar. Bu aerodinamik tork kontrol yöntemi, *yük regülasyonu* olarak bilinir. Bu regülasyon sayesinde rüzgar türbini, aşırı rüzgar hızlarına yine aynı şekilde shaft da belirli değer üstü veya altı tork değerlerine ulaşmamış olur.

Aşağıdaki şekil modern türbinlerde  $C_p$ 'yi  $\lambda$ 'nın bir fonksiyonu olarak göstermektedir. Bazı rüzgar türbinleri kanat eğim ayarı uygulayarak, geniş rüzgar hızı aralıklarında mümkün olduğu kadar çok enerji üretebilecek şekilde tasarlanmışlardır.



Resim 4.3: Modern rüzgar türbininde  $C_p$ 'nin  $\lambda$ 'nın ve kanat açısı ( $\beta$ )'nin fonksiyonu olarak değişimi

## 4.2. KAYIPLAR

Maksimum güç faktörü değeri olan *0,5926 Betz sınırı* olarak adlandırılır. Fakat hiç bir zaman bir rüzgar türbinin gerçek üretimi, teoride hesap edilen değer kadar olmaz. Daha düşüktür. Bunun da nedeni kayıplardır. Rüzgar türbinin pervanesinde oluşan bu kayıpları şöyle sıralayabiliriz:

#### 4.2.1. Profil Kayıpları

Profil kayıpları, ihmal edilen direnç kuvvetlerinden kaynaklanır. Hesaplamalarda;

$$\lambda_A = V_C / V_r = R \cdot n \cdot \pi / (30 \cdot V_r) \quad \dots(4.12)$$

$$\varepsilon = C_A / C_W \quad \dots(4.13)$$

$$\xi_{\text{Profil}}(\lambda_A, \varepsilon) = 1 - \lambda_A / \varepsilon \quad \dots(4.14)$$

ile dikkate alınır.

$\lambda_A$  = Tasarım devirlilik sayısı,

$V_C$  = Çevresel hız,

$\varepsilon$  = Kayma sayısı,

$C_A$  = Kaldırma kuvveti kat sayısı,

$C_W$  = Direnç kuvveti kat sayısı

#### 4.2.2. Uç Kayıpları

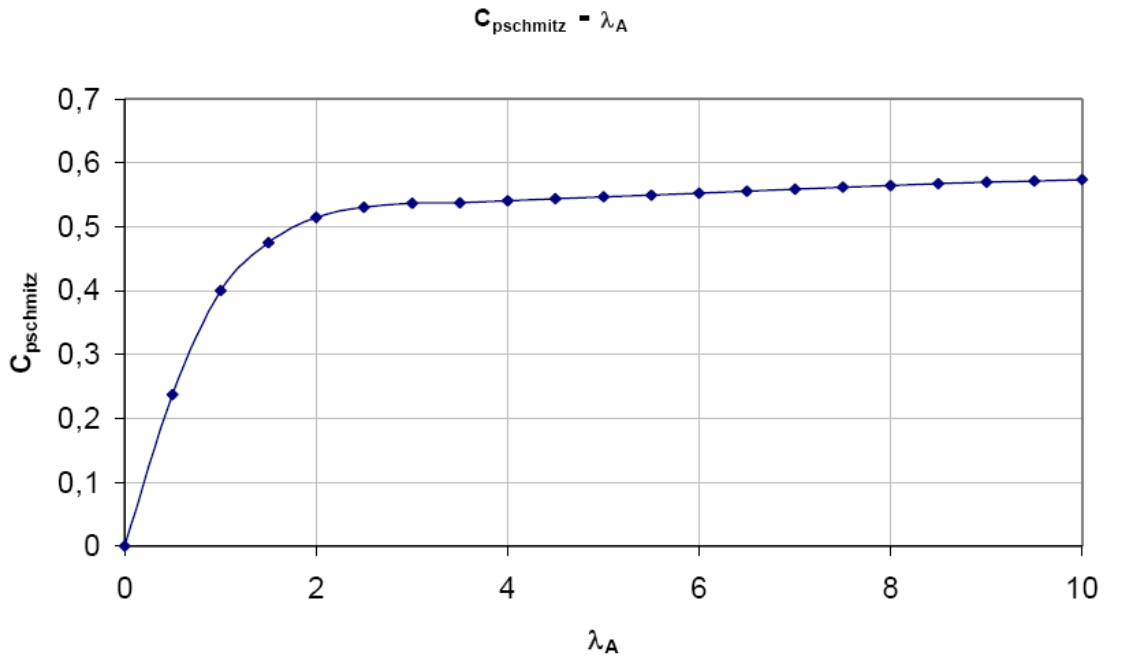
Kanat ucunda, profil alt kısmından profil üst kısmına doğru hava akımı oluşur. Kanat uçlarındaki bu akım ile kanada gelen hava akımı üst üste binerek, gittikçe genişleyen girdap oluştururlar. Hesaplamalarda bu kayıp;

$$\xi_{\text{uç}}(Z, \lambda_A) = 1 - [1,84 / (Z * \lambda_A)] \quad \dots(4.15)$$

ile dikkate alınır. Burada kanat sayısı ( $Z$ ) ile gösterilir.

### 4.2.3. Girdap Kayıpları

Betz Kriteri'ne göre rüzgar hızı, pervane düzlemi öncesi ve sonrasında doğrultusunu değiştirmez. Halbuki, kanada çarpan hava kütlesi, kanat sonrasında doğrultusunu değiştirir ve girdap oluşturur. Girdap kayıpları Schmitz tarafından verilen  $C_{P_{Schmitz}} - \lambda_A$  eğrisi yardımıyla hesaba katılır.



Resim 4.4:  $C_{P_{Schmitz}} - \lambda_A$  (Özdamar, A., 2002).

Tüm bu kayıpları göz önünde bulundurursak, gerçek güç faktörünü:

$$C_{P_{gercek}} = C_{P_{Schmitz}}(\lambda_A) * \xi_{Profil}(\lambda_A, \varepsilon) * \xi_{uç}(Z, \lambda_A) \quad \dots(4.16)$$

olarak buluruz.

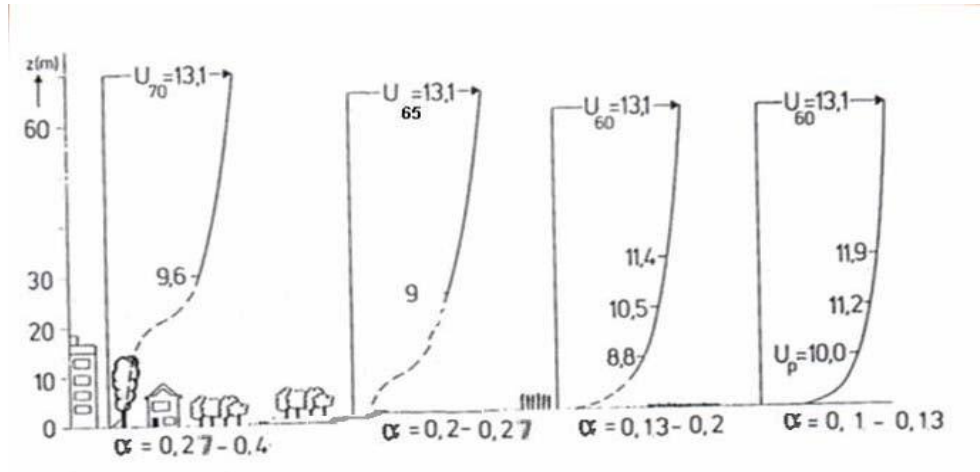
Tablo 4.1: Girdap kaybı için  $C_{Pschmitz}-\lambda_A$  deęerleri (Özdamar, A., 2002).

$\lambda_A$	$C_{Pschmitz}$
0	0
0,5	0,238
1	0,4
1,5	0,475
2	0,515
2,5	0,531
3	0,537
3,5	0,538
4	0,541
4,5	0,544
5	0,547
5,5	0,550
6	0,553
6,5	0,556
7	0,559
7,5	0,562
8	0,565
8,5	0,568
9	0,570
9,5	0,572
10	0,574

### 4.3. HELLMAN YÜKSELTME BAĞINTISI

Enerji Bakanlığı'nın kabul ettiği rüzgar ölçüm yüksekliği 10 ile 30 metre ve en az 1 senelik, saatlik ortalamalar şeklindedir. Bunun nedeni, ölçümleri standartlaştırmak ve ölçüm maliyetlerini düşürmektir.

Rüzgar hızının yükseklikle değişimi Hellman bağıntısı ile verilir. Türbinler kuleler üzerinde günümüzde 80-90 metrelere varan yüksekliklere yerleştirilmeye başlanmışlardır. Rüzgar hızı, yerden yüksekliğe bağlı olarak belirgin bir şekilde değişiklik gösterir.



Resim 4.5: Yüzey farklılıklarının rüzgar hızına etkisi ( Yalçın, C., 1998, s.23 )

Tablo 4.2: Yüzey farklılıklarının rüzgar hızına etkisi (Özdamar, A., 2000 (4), s.4)

DURUM	$\alpha$
Deniz Kıyı Şeridi	0,1-0,13
Yeşil ve Ekili Alan	0,13-0,2
Ağaçlı Alan	0,2-0,27
Yüksek Bina ve Kentsel Alan	0,27-0,4



10 ila 30 metrede ölçülen bu rüzgar hızlarını, türbin tahmini yerleştirme yüksekliğine çıkartılması için Hellman Yükseltme Bağıntısı kullanılır. Buna göre ;

$$V_{Rist} / V_{Rölç} = ( H_{ist} / H_{ölç} )^{\alpha} \quad \dots( 4.17 )$$

olarak bulunur. Burada;

$V_{Rölç}$  : Ölçülen rüzgar hızı (m/s)

$V_{Rist}$  : İstenen rüzgar hızı (m/s)

$H_{ist}$  : İstenen yükseklik (m)

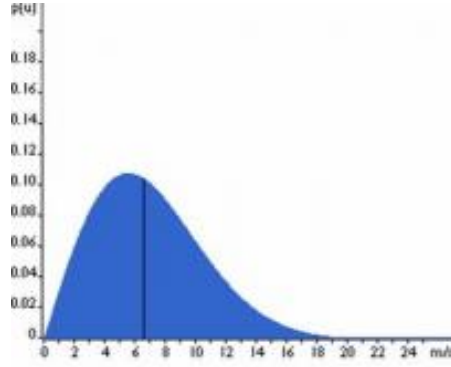
$H_{ölç}$  : Ölçüm yapılan yükseklik (m)

$\alpha$  : Yüzey durumu

#### 4.4. WEIBULL DAĞILIMI VE RAYLEIGH DAĞILIM FONKSİYONU

Meteorolojik ölçümler sonucu elde edilen rüzgar verileri uzun vadeli rüzgar kayıtları elde etmek ayrıca, farklı site ve farklı yüksekliklerdeki rüzgar özelliklerini belirlemek için değerlendirmeye alınır. Değerlendirmede, çeşitli rüzgar hızı olasılık dağılımları ve bunları matematiksel olarak modellemekte kullanılan fonksiyonlara başvurulur. Bu fonksiyonlardan en çok kullanılanlardan biri de Weibull Dağılımı ve Rayleigh fonksiyonudur.

Weibull Dağılımı, rüzgarın belirli bir periyottaki değişimi ve dağılımının bulunmasında kullanılan iki parametrelili bir ifadedir. Eğer bir yıl boyunca rüzgar ölçülürse, rüzgarın hangi şiddet değerinde ve hangi sıklıkta estiğini gösteren grafik aşağıdaki şekildeki (Şekil 4.6) gibi olacaktır. Bu dağılımın altında kalan alanın toplam olabirliği “1” dir.



Resim 4.6: Rüzgarın hangi sıklıkta estiğini gösteren şekil

Rayleigh fonksiyonu Weibull'un basitleştirilmiş bir versiyonu olarak düşünülebilir. Rüzgarın belli bir periyotta değişimi ve dağılımı, enerji üretimi değerlendirmeleri için çok önemlidir. Türbin tasarımcıları; türbin iyileştirmesinde ve maliyetleri en aza indirmede rüzgar dağılımı ve değişimi ile ilgili bilgilere gerek duyarlar.

Bir yerde sadece ortalama rüzgar hızı ( $V_{\text{ort}}$ ) biliniyorsa; Rayleigh Dağılım Fonksiyonu yardımıyla herhangi bir rüzgar hızının ( $V_{\text{ri}}$ ), esme saati ( $h_r$ ) yüzdesi bulunabilir. Bunun sonucunda ortaya çıkan rüzgar hızları bir olasılık yoğunluğu dağılımıdır. Dağılım şematik olarak çizilirse bu dağılımın altında kalan alan 1'e eşittir. Çünkü, rüzgarın sıfır dahil herhangi bir hızda esme olasılığı %100'dür.

Rayleigh dağılım fonksiyonuna göre esme hızı saati,

$$h_r = \pi / 2 * V_{\text{ri}} / V_{\text{ort}}^2 * e^{[-(\pi/4) * (V_{\text{ri}} / V_{\text{ort}})^2]} \quad \dots (4.18)$$

olarak bulunur (Özdamar, A., 2001(2)).

### Örnek:

Söke'de 10 metre yükseklikte ölçülen ortalama rüzgar hızı 6,14 m/s'dir. 1 yıllık her bir rüzgar hızı için esme saatini bulunuz.

( 4.18 ) numaralı eşitlikten faydalanırsak

$$h_r = \pi / 2 \cdot V_{ri} / V_{rot}^2 \cdot e^{[(\pi/4) \cdot (V_{ri} / V_{rot})^2]} \\ = 0,0416645 \cdot V_{ri} \cdot e^{-0,0208225 V_{ri}^2}$$

olarak bulunur. Buna göre yıllık esme saati için Tablo 4.3'deki değerler bulunur.

Tablo 4.3: Yıllık esme saati tablosu

$V_{ri}$ ( m / s )	$h_{ri}$	%	Yıllık Esme Saati
1	0,04088	4	358,15
2	0,0773	7	677,79
3	0,105849	10	927,24
4	0,124	12	1086,76
5	0,1313	13	1150,9
6	0,128	12	1127,79
7	0,118	11	1035,57
8	0,102	10	897,79
9	0,084	8	738,47
10	0,066	6	578,23
11	0,049	5	432,01
12	0,035	3	308,53
13	0,024	2	210,89
14	0,015	1	138,12
15 ve Üzeri	0,009	0,9	86,74

## **5. RÜZGÂR ENERJİSİ OPTİMİZASYONU PROGRAMI ( PROWINDANALYSER 1.0 )**

### **5.1. PROGRAMIN TANITILMASI**

PROWINDANALYSER 1.0 programının geliştirilme amacı, yukarıda optimizasyon kısmında karşılaşılabilecek önemli noktalara ait verilerin analiz edilebilmesidir. Program, Microsoft Visual Studio.NET 2008 ortamında C# programlama dili ve SQL Server 2008 Express Edition veri tabanı kullanılarak tarafımdan gerçekleştirilmiştir.

Program iki ana kısımdan oluşmaktadır:

*a.Giriş Ekranı*

*b.Ana Ekran*

*b.1.Rüzgâr Analizi*

*b.2.Enerji Analizi*

*b.3.Türbin Analizi*

*b.4.Maliyet Analizi*

*b.5.Program Açıklaması*

### 5.1.1. Giriş Ekranı



Resim 5.1: PROWINDANALYSER ana giriş ekranı

Programın kullanıcı açılış arayüzüdür. Kullanıcı giriş butonuna tıklayarak ana ekrana geçiş yapmaktadır.

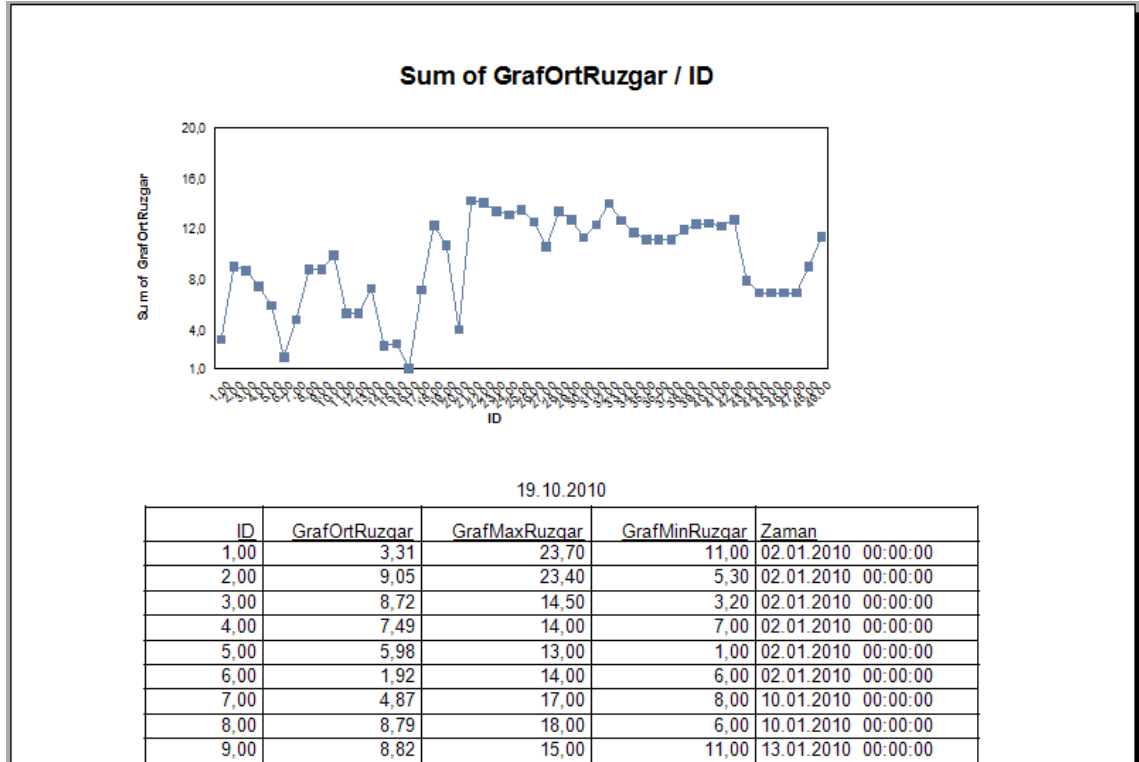
### 5.1.2. Ana Ekran

Programda kullanıcının karşılaştığı ikinci ekrandır. Ana penceredir. Rüzgâr analizi ekranı, enerji analizi ekranı, türbin analizi ekranı, maliyet analizi ekranı, program tanıtım ekranı olmak üzere 5 ayrı sekme bloğundan meydana gelmiştir. Sekme blokları arası işlem bağımsızlığı vardır. Kullanıcı rüzgâr analizi ekranıyla işlem zincirine başlayıp, sırasıyla güç analizi, türbin analizi ve maliyet analizi ekranlarına geçiş yapabilmektedir.

### 5.1.2.1. Rüzgâr Hızı Analizi Ekranı

Resim 5.2: Rüzgar hızı analizi ekranı 1.kısım

Resim 5.3: Rüzgar hızı analizi ekranı 2.kısım



Resim 5.4: Rüzgar hızı analizi ekranı 3.kısım

Kullanıcının varsayılan olarak karşılaştığı ilk ekrandır. Bu kısımda kullanıcının rüzgar hızı verilerini alabilmesi için bir çok opsiyon sunulmuştur.

- LOGGER CİHAZI
- EXCEL
- RS 232
- RS 485
- DENEYSEL

Resim 5.5: Veri alma kaynağı seçim butonları.

Bu butonlardan eğer “*Logger Cihazı*” tercih edilecek olursa, rüzgar hızı verileri direk logger cihazı üstünden alınarak programdaki ilgili alanlarda işlenmek üzere dahil edilir. Eğer “*EXCEL*” butonu (radiobutton) tercih edilecek olursa, herhangi bir şekilde excel dökümü halinde olan rüzgar hızı kayıtlarını program içine işlenmek üzere alabilecektir. Eğer “*RS-232*” veya “*RS-485*” bağlantıları tercih edilecek olursa, cihaz veya bir bilgisayar üzerinden veriler programa dahil edilebilecektir. “*Deneysel*” opsiyonda da kullanıcı test ya da kontrol amaçlı rüzgar hızı datalarıyla program işletebilecektir. Ayrıca kullanıcıya “*Deneysel*” opsiyonda uygun zaman periyotlarında çalışıp verileri ona göre yorumlayabilmesi için *senelik, 6 aylık, 3 aylık, aylık, günlük ve saatlik* zaman dilimleri de bulunmaktadır.

Eğer kullanıcı ay bazında verilerin yorumunu almak isterse mevsimlere ya da belli aylara göre rüzgar hızı verilerini görmek isterse; ana ekranda bulunan, tüm ayların bulunduğu kutulara giderek istediği ayları işaretleyip araştırmasını bu yönde sürdürebilmektedir.

Rüzgar hızı verilerinin analizinde alınan ölçüm sonuçları içinden (senelik, dönemlik, aylık, günlük, saatlik) maksimum, minimum ve ortalama rüzgar hızı değerleri belirlenir. İlgili kısımlarda veri olarak programa işlenir. Gerek logger cihazı verileri olsun, gerek excel ya da data bağlantı verileri, gelen veriler işlenerek bu kısımlarda kullanıcıya gösterilir.

ORTALAMA RÜZGAR HIZI	<input type="text"/>	m/s
MAXIMUM RÜZGAR HIZI	<input type="text"/>	m/s
MINIMUM RÜZGAR HIZI	<input type="text"/>	m/s

Resim 5.6: Ortalama, max., min. rüzgar hızlarının bulunduğu grup.

Kullanıcı rüzgar hızı verilerini, anlık değerler halinde ; on dakikada bir, otuz dakikada bir, saatte bir, günlük, aylık ya da yıllık periyotta işleyebilmesi için böyle bir opsiyon



program dahilinde sunulmuştur. Diğer bir opsiyonda ise kullanıcı rüzgar hızı verilerini rüzgar esme sayısına bağlı olarak alıp, program içinde işleme koyabilmektedir.

ANALİZ TİPİ

ANLIK DEĞER  ESME SAYISINA BAĞLI DEĞER

Resim 5.7: Anlık ve esme sayısına bağlı işlem seçim bölümü.

Programın sistematüğinde rüzgar hızı analizi yapılan bölge ya da enerji anlamında beslenecek kapasitenin, projeye başlanmadan belirlenmesi büyük önem taşımaktadır. Bununla ilgili olarak kişi başı elektrik tüketim miktarının ortalama bir değer olarak belirlenip, özellikle dönemsel olarak ortaya konması (yaz dönemi, kış dönemi) gerekmektedir.

ELEKTRİK TÜKETİMİ

YAZ DÖNEMİ KİŞİ BAŞI TÜKETİM

kw

KIŞ DÖNEMİ KİŞİ BAŞI TÜKETİM

kw

DÖNEMSEL TÜKETİM

KIŞ SEZONU

EYLÜL

EKİM

KASIM

ARALIK

OCAK

ŞUBAT

YAZ SEZONU

NİSAN

MAYIS

HAZİRAN

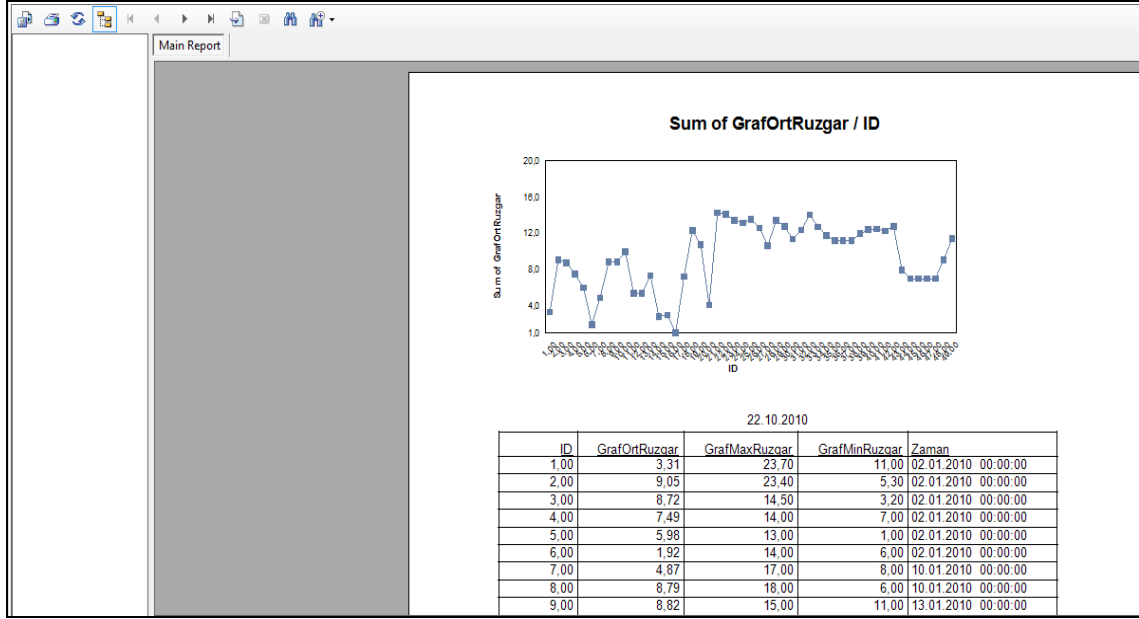
TEMMUZ

AĞUSTOS

BUL

Resim 5.8: Elektrik tüketim hesap grubu.

Dışarıdan alınarak program içinde işlenen veriler, M.S. SQL Server veri tabanına kaydedilerek server üzerinde saklanabilir veya programın kurulu bulunduğu ayrı bir bilgisayar üzerinden de uzaktan ulaşılabilir. Ayrıca ölçüm analizleriyle ilgili kristal raporlamalar yapılarak gerekirse grafiksel olarak da rüzgar hızı verilerinin analizleri yapılabilir.

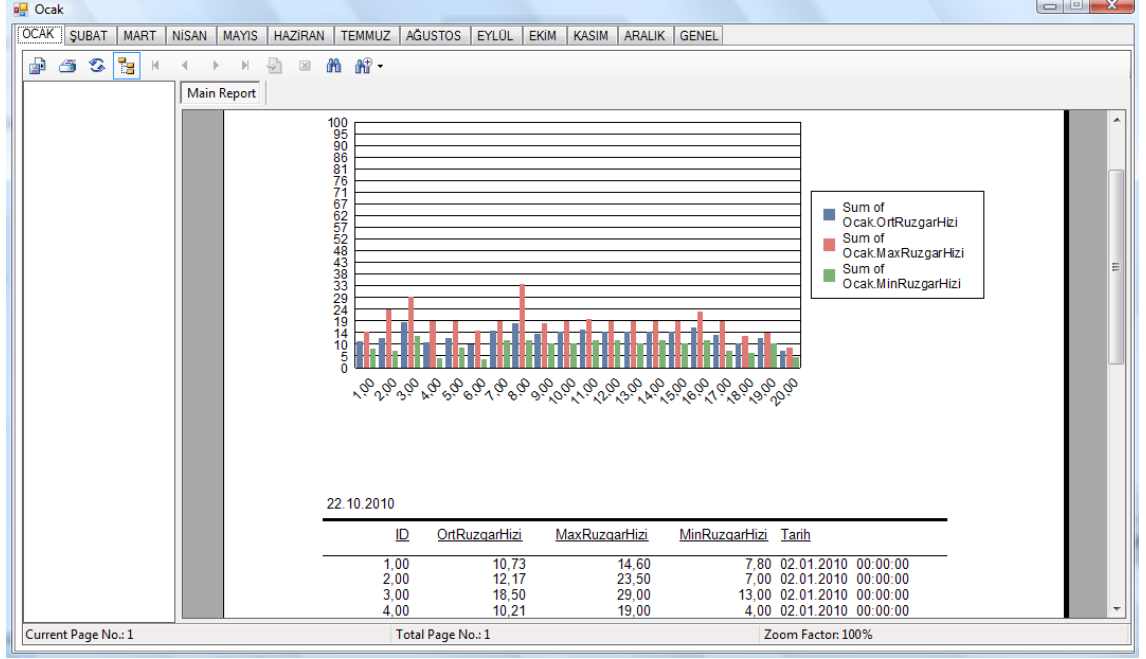


Resim 5.9: Rüzgar hızı verilerine ait kristal raporlama.

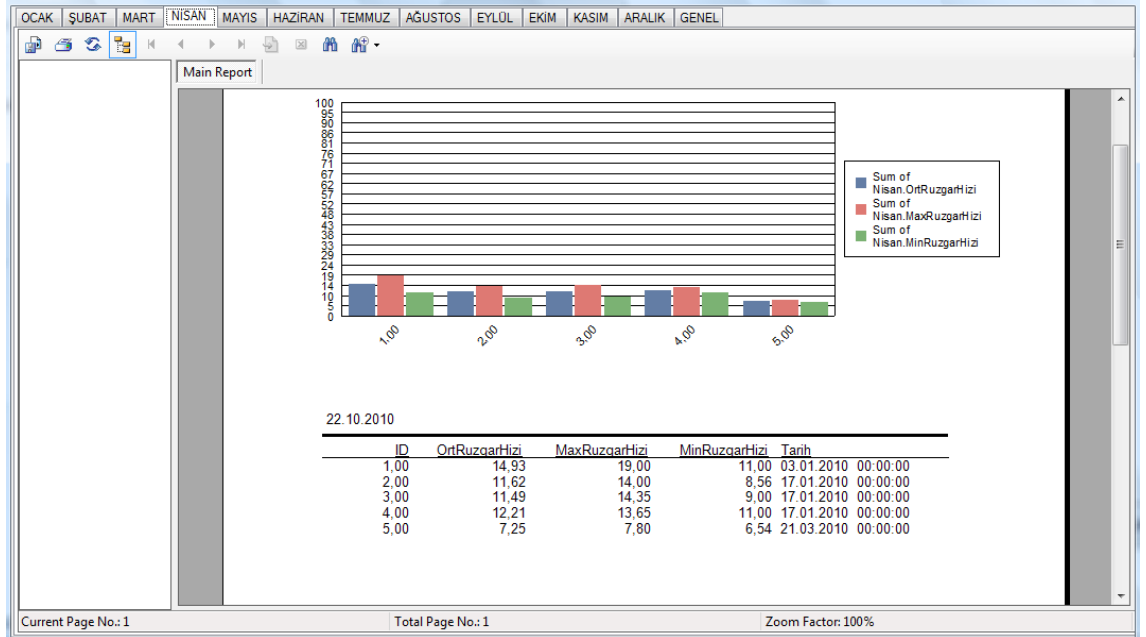
Aylara göre bir raporlama ya da ay bazında araştırma sonuçları incelenmek isterse, “*AYLARA GÖRE RÜZGAR HIZI DAĞILIMI*” butonuna basılarak, aylık bazdaki veriler ve grafiksel çözümlenmeleri izlenebilir.

Yıllık bir raporlama ya da grafiksel dağılım görüntülenmek isteniyorsa, “*YILLIK RÜZGAR DAĞILIMI*” butonuna basılabilir.

Aylık verilere ait olan raporlar, kristal raporlama şeklinde yapıp, kullanıcının hem veri takibini daha rahat yapabilmesi, hem de veri tabanı içinde verilerin daha düzenli bir şekilde saklanıp, ilişkilendirilebilmesi için program dahilinde önemli bir kısmı oluşturmaktadır. Ayrıca bir çok farklı çalışma için arşiv görevi de görmektedir.



Resim 5.10: Ocak ayına ait raporlandırma sekmesi.

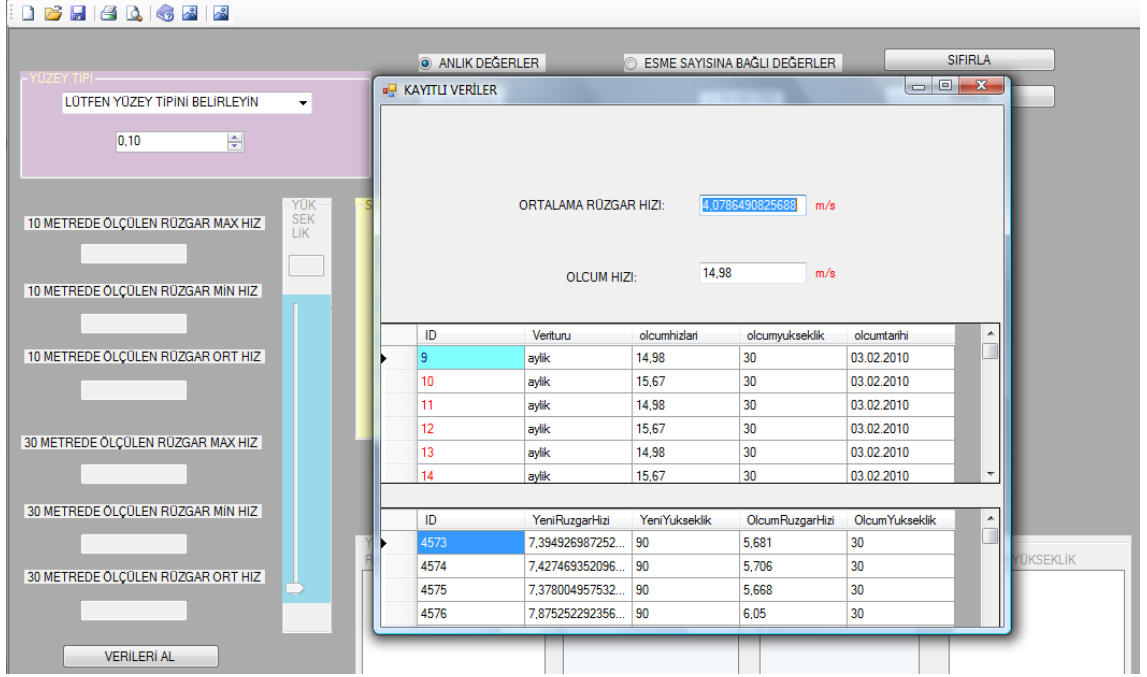


Resim 5.11: Nisan ayına ait raporlandırma sekmesi.

Bilindiği üzere rüzgar hızı ölçümleri 10 ile 30 metreki rüzgar hızı değerlerine göre yapılır, ama modern rüzgar türbinlerinin ortalama yükseklikleri 80-90 metreler civarındadır. Yani rüzgar hızı analiz sayfasına gelen verilerin kullanılacak türbin yüksekliklerine göre yeniden hesaplanması gerekmektedir. Programın içinde de böyle bir bölüm bulunup, esme sayısına bağlı rüzgar değerleri veya anlık rüzgar değerlerine göre analiz yapıлып, yeniden hesaplanır. Ayrıca rüzgar tarlasının kurulacağı bölgenin yüzey tipiyle ilgili opsiyonlar da bu kısımda yer almaktadır. Veriler burada işlenerek enerji analizi kısmı için veri tabanında oluşturulan tabloya kayıt olur ve veri tabanı tabloları arası kurulmuş olan ilişkilendirmeler aracılığıyla tüm program dahilinde kullanılabilir.

Resim 5.12: Hellman yükseltme bağıntısı kullanım arayüzü.

Bu kısımda yükseltme bağıntısıyla hesaplanan verilere göre ortalama rüzgar hızı hesabının yeniden yapıldığı ve veriler arasında bağlantıları gösteren ikinci bir arayüz daha bulunmaktadır.



Resim 5.13: Hellman kısmı ortalama rüzgar hızı hesabı arayüzü.

#### 5.1.2.2. Güç Analiz Ekranı

Bu kısımda rüzgar hızı analizi ekranı tarafından listelenen, ortalaması, maksimum ve minimum değerleri bulunan rüzgar hızı verilerinin, taşıdıkları maksimum, minimum ve ortalama güç değerleri hesaplanır. Aynı şekilde rüzgar türbini seçiminde önemli bir nokta olan güç faktörü ve yavaşlatma faktörüne göre de sahip oldukları güçler hesaplanır. Ekrandaki farklı bir grup içinde seçilecek türbin tipine göre veya kullanıcı tarafından belirlenecek değerlere göre türbin kayıpları da enerji analiz ekranındaki hesaplamalar içinde yer alır. Bu kayıp verilerinin ışığında gerçek güç faktörü dediğimiz türbinin üretebileceği optimum güç oranı da hesap edilir. Hesaplama denklemleri Betz teoremi doğrultusunda oluşturulmuştur.

Resim 5.14: Güç analizi ekranı 1.kısım.

Betz teoremi gereğince güç hesabında yararlanılmak üzere rüzgar hızı analizi ekranında hava yoğunluğunu hesap edebilmek için, türbinin kurulacağı yüksekliğe ait sıcaklık ve basınç değerlerinin bulunduğu bir ekran bulunmaktadır. Burada kullanıcının girdiği veriler ışığında hava yoğunluğu değeri belirlenmekte ve aynı şekilde yine Betz teoremine göre güç hesabında kullanılmak üzere, türbin tipine ve dolayısıyla boyutlarına göre değişen, “*SÜPÜRÜLEN RÜZGAR ALANI*” diye belirttiğimiz güç hesabında belirleyici rol üstlenen bir kısımda mevcuttur. Bu değer alanına veri girişi ister kullanıcı tarafından yapılabilir, istenirse de program içinde kayıtlı bulunan rüzgar türbini katalog bilgilerinin bulunduğu çoklu seçim kutusundan, kullanıcı istediği türbini seçerek bu değer otomatik olarak ilgili veri kutusuna gelmesini sağlayabilir.

KAYIPLAR

PROFİL KAYBI HESABI:

UÇ KAYBI HESABI:

GİRDAP KAYBI HESABI:

$\lambda_A$ (Dizayn Devirlik Sayısı)

$\epsilon$ (Kayma Sayısı)

$V_{\text{ç}}$ (Çevresel Hız)  m/dak

$R$ (Yarıçap)  m

$CA$ (Kaldırma Kuvveti Katsayısı)

$CW$ (Direnç Kuvveti Katsayısı)

$Z$ (Kanat Sayısı)

$n_d$ (Devir Sayısı)  rpm

$V_r$ (Ortalama Rüzgar Hızı)  m/s

KAYIPLARI HESAPLA

Resim 5.15: Türbin kayıp hesap grubu 1.kısım.

GERÇEK GÜÇ FAKTÖRÜ:

MAKSİMUM GÜÇ:  W

GERÇEK ORT. GÜÇ:  W

GERÇEK DEĞERLERİ HESAPLA

Resim 5.16: Türbin kayıp hesap grubu 2.kısım.

PROWIND

ANALİZ | MALİYET ANALİZİ | HAKKINDA

**ENERJİ ANALİZİ**

HELLMAN

TÜRBİNDEN ÇIKAN RÜZGAR HIZI  Ortalama Rüzgar Hızına Göre

Rüzgar Çıkış Hızı

YAVAŞLATMA FAKTÖRÜ HESABI

TÜRBİN İÇİNDEKİ RÜZGAR HIZI VKİVR=

TÜRBİN KATALOG DEĞERLERİNİ KULLAN

DİĞER FAKTÖRLER

HAVANIN YOĞUNLUĞU

SÜPÜRÜLEN RÜZGAR ALANI

KAYIPLAR

PROFİL KAYBI HESABI: label18

UÇ KAYBI HESABI: label19

GİRİŞ KAYBI HESABI: label20

JA

e

Vç

R

CA

CW

Z

nd

Vr

KAYIPLARI HESAPLA

TÜRBİN RÜZGAR HIZINA GÖRE GÜÇ HESABI

TÜRBİN ORTALAMA RÜZGAR HIZINA GÖRE

TÜRBİN ORTALAMA RÜZGAR GÜCÜ

GÜÇ FAKTÖRÜYLE TÜRBİN RÜZGAR HIZLARI

GÜÇ FAKTÖRÜYLE TÜRBİN GÜCÜ

GÖRE GÜÇ HESABI:

GÖRE ORT GÜÇ HESABI:

GERÇEK GÜÇ FAKTÖRÜ:

MAXIMUM GÜÇ:

GERÇEK ORT GÜÇ:

GERÇEK DEĞERLERİ HESAPLA

Resim 5.17: Güç analizi ekranı 2.kısım.

PROWIND

YILLIK ORTALAMA RÜZGAR HIZI:

SEZONLUK ORTALAMA RÜZGAR HIZI:

AYLIK ORTALAMA RÜZGAR HIZI:

GÜNLÜK ORTALAMA RÜZGAR HIZI:

GÜÇ HESAPLARI

ORTALAMA RÜZGAR HIZINA GÖRE GÜÇ HESABI:

TÜRBİN RÜZGAR HIZLARINA GÖRE ORT GÜÇ HESABI:

TÜRBİN KATALOG DEĞERLERİNİ KULLAN

TÜRBİN RÜZGAR HIZINA GÖRE GÜÇ HESABI

TÜRBİN ORTALAMA RÜZGAR HIZINA GÖRE

TÜRBİN ORTALAMA RÜZGAR GÜCÜ

GÜÇ FAKTÖRÜYLE TÜRBİN RÜZGAR HIZLARI

GÜÇ FAKTÖRÜYLE TÜRBİN GÜCÜ

CW

Z

nd

Vr

GERÇEK GÜÇ FAKTÖRÜ:

MAXIMUM GÜÇ:

GERÇEK ORT GÜÇ:

GERÇEK DEĞERLERİ HESAPLA

ESME SAATI

RÜZGAR HIZLARI

ESME SAATI

RÜZGARIN ESME YÜZDESİ:

RÜZGARIN YILLIK ESME SAATI:

Resim 5.18: Güç analizi ekranı 3.kısım.



Rüzgar hızı analiz ekranında belirlenen işlem döngüsüne göre (deneysel, aylık , excel, Hellmann), veriler ilk olarak “Rüzgar Giriş Hızı” alanına gelir. Kullanıcı tarafından elle girilen ya da türbin katalog grubundan seçilen türbin tipine göre; güç faktörü değeri girilip , veri girişinden sonra ekrana düşen “HESAPLA” butonuna basılarak rüzgarın türbinden çıkış hızına ait değerler bulunur. Program döngüsünde bir sonraki adım, işlenen verileri “Rüzgar Çıkış Hızı” alanına getirmektir. Bundan sonra ortalama rüzgar hızına göre verilerin işlenmesine devam edilir, “TÜRBİN RÜZGAR HIZINA GÖRE GÜÇ HESABI” grubunda, veriler program içinde yazılmış fonksiyonlar ve prosedürler dahilinde hesaplanıp, kullanıcıya ilgili kısımlarda gösterilir.

The screenshot shows a software interface with the following elements:

- At the top, there are four radio buttons for selecting the calculation method: DENEYSEL, AYLIK, EXCEL, and HELLMAN.
- Below these are two main input sections:
  - TÜRBİNE GELEN RÜZGAR HIZI**: A dropdown menu labeled "Rüzgar Giriş Hızı".
  - TÜRBİNDEN ÇIKAN RÜZGAR HIZI**: A dropdown menu labeled "Rüzgar Çıkış Hızı".
- To the right of these sections is a checkbox labeled "Ortalama Rüzgar Hızına Göre".
- On the far right, there is a box titled "DİĞER FAKTÖRLER" containing:
  - A field for "HAVANIN YOĞUNLUĞU".
  - A field for "SÜPÜRÜLEN RÜZGAR ALANI".
- Below the main input sections are two calculation boxes:
  - GÜÇ FAKTÖRÜ HESABI**: A field labeled "PT\PR=" followed by an input box.
  - YAVAŞLATMA FAKTÖRÜ HESABI**: A field labeled "VK\VR=" followed by an input box.
- In the center, between the two calculation boxes, is the text "TÜRBİN İÇİNDEKİ RÜZGAR HIZI".

Resim 5.19: Güç faktörü, yavaşlatma faktörü ve türbinden çıkan rüzgar hızı hesap ekranı.

The screenshot shows a software interface with four empty boxes for displaying calculation results:


- TÜRBİN RÜZGAR HIZINA GÖRE GÜÇ HESABI**: A large empty box.
- TÜRBİN ORTALAMA RÜZGAR HIZINA GÖRE**: A large empty box.
- TÜRBİN ORTALAMA RÜZGAR GÜCÜ**: A large empty box.
- GÜÇ FAKTÖRÜYLE TÜRBİN RÜZGAR HIZLARI**: A large empty box.
- GÜÇ FAKTÖRÜYLE TÜRBİN GÜCÜ**: A large empty box.

Resim 5.20: Türbin ortalama rüzgar gücü ve güç faktörüyle türbin gücü hesap ekranı.


Şekil 5.20’de program içinde işlenen rüzgar hızı verilerinin, sahip oldukları güç değerleri hesaplanmaktadır. Bunun dışında, ortalama rüzgar hızı değerine göre ortalama gücün bulunduğu, güç faktörüne göre ayrı bir ortalama gücün hesap edildiği bir kullanıcı ekranı da bulunmaktadır.

**GÜÇ HESAPLARI**

**ORTALAMA RÜZGAR HIZINA GÖRE GÜÇ HESABI**



**TÜRBİN RÜZGAR HIZLARINA GÖRE ORTALAMA GÜÇ HESABI**



Resim 5.21: Ortalama rüzgar hızlarına göre güç hesap grubu.

Rayleigh dağılım fonksiyonu yardımıyla rüzgar hızları analiz ekranından gelen rüzgar hızı verileriyle, herbir rüzgar hızının esme saatini, esme yüzdesini ve yıllık esme saatleri hesaplanabilir.

**ESME SAATI**

RÜZGAR HIZLARI	ESME SAATI	RÜZGARIN ESME YÜZDESİ	RÜZGARIN YILLIK ESME SAATI
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Resim 5.22: Esme saati hesap grubu.

Özetle güç analizi ekranında türbine gelen rüzgar hızlarına göre güç ve ortalama güç değerleri hesaplanarak, olasılık fonksiyonları yardımıyla rüzgarın yıllık esme saati, yüzdesi, türbin kayıpları da hesaba dahil edilip, yeni güç faktörü değeri elde edilerek optimum güç dağılımı oluşturulur.

### 5.1.2.3. Türbin Analizi Ekranı

Türbin analizi ekranında, türbin üretici firmalar bu firmaların üretimini yaptıkları rüzgar türbini tipleri, rüzgar hızı analizi ekranından gelen rüzgar hızı değerlerine göre, veri tabanına kayıtlı türbin tipi katalog değerlerinden, o rüzgar hızında sağlanabilecek güç miktarları, rüzgar hızı giriş ekranında verilen kişi başı elektrik tüketim değerleri kullanılarak üretilmesi gereken enerjinin hesap edildiği (yıllık, dönemlik, aylık, günlük, saatlik) gruplar bulunmaktadır. Bu gruplar yardımıyla ve türbin tiplerine göre rüzgar hızlarından sağlayacakları güç oranları hesaplanarak, rüzgar enerji santrali kurulması planlanan yer için kaç adet türbinin kullanacağı hesaplanmaktadır. Bu da kullanıcıya bu tip bir enerji santralinde bulunabilecek optimum türbin sayısını vermektedir.

Optimum türbin sayısı bulunduktan sonra “*ÜRETİLEN ENERJİ*” grubunda da, “*BİR TÜRBİNİN ÜRETTİĞİ ENERJİ*”, “*TÜRBİN GRUBUNUN ÜRETTİĞİ ENERJİ*”, “*ÜRETİLMESİ GEREKEN TOPLAM ENERJİ*”, “*İHTİYAÇ FAZLASI ENERJİ*”, “*DIŞARDAN TALEP EDİLEN ENERJİ*”, “*TÜRBİN GRUBUNUN YILLIK ENERJİ ÜRETİMİ*”, “*ÜRETİLMESİ GEREKEN YILLIK ENERJİ*”, “*YILLIK İHTİYAÇ FAZLASI ENERJİ*”, “*YILLIK DIŞARDAN TALEP EDİLEN ENERJİ*” bulunarak, türbinin verimini belli eden “*KAPASİTE FAKTÖRÜ*” hesabı yapılır. Kapasite faktörü analizinde dünya ortalaması %20 civarındadır. Bu oranın üzerindeki kapasite faktörüne sahip yerler, enerji üretimi açısından verimlidir. Bu sebepten bu değer optimizasyon anlamında önem taşımaktadır.

Bu ekranda özellikle yurt dışı rüzgar enerjisi piyasalarında, türbin imalatçısı olarak adını duyurmuş şirketlerin katalog bilgileri bulunmaktadır.

## FİRMALAR

VESTAS

NORDEX

SIEMENS

DEWIND

**TÜRBİN SAĞLAYICILARI**

**FİRMALAR**

VESTAS

NORDEX

SIEMENS

DEWIND

**TÜRBİN TİPLERİ**

V52-850KW  
V80-2MW  
V82-1,65MW  
V90-1,8MW  
V90-3MW  
V112-3MW(OFFSHORE)

**RÜZGAR HIZLARI**


**TÜRBİN KATALOG GÜÇ DEĞERLERİ**

**ORTALAMA RÜZGAR HIZI**

**ORTALAMA HIZ KATALOG GÜÇ DEĞERİ**

**TÜRBİN TEKNİK BİLGİ**

**V52-850KW**



3 kanatlı

**STANDART HIZ**

5m/s 100kw

10m/s 600kw

16m/s 850kw

**ROTOR:**

52

**SÜPÜRÜLEN ALAN:**

2,124

**NORMAL DEVİR:**

26

**KALKIŞ HIZI:**

4m/s

Nominal Hız:

Resim 5.23: Türbin analizi ekranı 1.kısım

**ÜRETİLMESİ GEREKEN ENERJİ**

**YILLIK**

**DÖNEMLİK**

**AYLIK**

**GÜNLÜK**

**SAATLİK**

**kwh**

**OPTİMUM TÜRBİN SAYISI:**

**ÜRETİLEN ENERJİ**

**BİR TÜRBİNİN ÜRETTİĞİ ENERJİ:**  **kwh**

**TÜRBİN GRUBUNUN ÜRETTİĞİ ENERJİ:**  **kwh**

**ÜRETİLMESİ GEREKEN TOPLAM ENERJİ:**  **kwh**

**İHTİYAÇ FAZLASI ENERJİ:**  **kwh**

**DIŞARDAN TALEP EDİLEN ENERJİ:**  **kwh**

**TÜRBİN GRUBUNUN YILLIK ENERJİ ÜRETİMİ** **kwh**

**ÜRETİLMESİ GEREKEN YILLIK ENERJİ** **kwh**

Resim 5.24: Türbin analizi ekranı 2.kısım

#### 5.1.2.4.Maliyet Analizi Ekranı

Maliyet analizi ekranı programın yorum ekranı olma özelliğini taşımaktadır. Rüzgar hızı verileri, rüzgar hızı analiz kısmında alınıp, işlenmiş; oradan güç analiz ekranında güç değeri hesapları yapılmış, türbin analiz ekranında santralin yapısına uygun türbin türü ve sayısı belli olup, optimizasyonun son noktası olan maliyet analizi kısmında türbin tipleri karşılaştırılmak üzere artı ve eksileri ile kullanıcıya sunulmuştur.

Resim 5.25: Maliyet analizi ekranı 1.kısım.

DATALAR

TÜRBİN	
ÜRETİLEN ENERJİ(kwh)	
BİRİM SATIŞ FİYATI(\$/kwh)	
BİRİM MALİYET(\$/kwh)	
TÜRBİN FİYATI(\$)	
GERİ ÖDEME SÜRESİ(YIL)	

Resim 5.26: Maliyet analizi ekranı 2.kısım.

Bilindiği üzere enerji sistemlerinin değerlendirilmesinde birim enerji maliyeti en önemli parametredir. Programın bu kısmında kurulması planlanan rüzgar enerjisi santralının diğer sistemlerle kıyaslanabilmesi için, birim enerji maliyeti hesabı yapılmaktadır.

TÜRBİN FABRİKA ÇIKIŞ FİYATLARI

TÜRBİNİN FABRİKA ÇIKIŞ FİYATI:

TÜRBİNİN MARKASI:

TÜRBİN KATALOG DEĞERİ

ORTALAMA EKONOMİK ÖMÜR:

KW BAŞINA DÜŞEN SATIŞ DEĞERİ

Resim 5.27: Türbin markasına göre türbin fabrika çıkış fiyatı hesap grubu.

TÜRBİNİN YAN MALİYETİ

AYRI AYRI MALİYETLERİ HESAPLA  YÜZDE OLARAK BELİRT

BAKIM MALİYETİ:

ONARIM MALİYETİ:

SİGORTA MALİYETİ:

SERMAYE MALİYET AMORTİSMANI:

İŞÇİLİK BEDELİ:

DİĞER MALİYETLER:

TOPLAM YAN MALİYET:

Resim 5.28: Türbin yan maliyet hesap ekranı.

Program birim enerji maliyeti hesabında, sistemin ilk yatırım maliyetine ek olarak; yıllık bakım-onarım, sigorta giderleri, amortisman bedelleri ve işçilik bedellerini de hesaplamaktadır. En sonunda toplam maliyet kısmında tüm bu kalemler biraraya getirilerek, toplam genel maliyet hesabı ortaya çıkmaktadır.

**BİRİM ENERJİ MALİYETİ**

TOPLAM MALİYET:	<input type="text"/>	\$
TOPLAM ÜRETİLEN ENERJİ:	<input type="text"/>	kwh
BİRİM ENERJİ MALİYETİ:	<input type="text"/>	\$/kwh

Resim 5.29: Birim enerji maliyet hesap ekranı.



## 6. MALZEME VE YÖNTEM

Tez çalışmasının bu bölümünde Çorlu Ulaş mevkinde konumlu, Can Enerji Üretim A.Ş.'ne ait rüzgâr ölçüm direğinden alınan rüzgâr hızı verileri ve oluşturulan optimizasyon programı kullanılarak ilgili analizler (rüzgâr hızı analizi, güç analizi, türbin analizi ve maliyet analizi) yapılmış ve veriler eşliğinde karşılaştırılarak, bölge için uygun türbin seçimi yapılmaya çalışılmıştır.

### 6.1. RÜZGÂR HIZI VERİLERİNİN PROGRAM DAHİLİNDE İŞLETİLMESİ

Resim 6.1: Rüzgâr hızı analiz ekranı veri tipi seçimi

Hız verileri ilk önce logger cihazından alınarak, excel kayıtları oluşturulmuştur. Ve bu kayıtlar masaüstünde tutularak, programın veri tabanı olan MS-SQL Server'la ilişkilendirilerek verilerin excel dosyasından, direk veri tabanı dolayısıyla da programa geçişleri sağlanmıştır.

Kayıt altına alınan ölçüm değerleri günlük değerler olup, onar dakikalık paketler şeklinde programa dahil edilmişlerdir. (Ekler kısmında rüzgar hızı kayıtlarına ulaşılabilir.)

Değerleri almadan önce, rüzgâr analizi ekranında bulunan “ELEKTRİK TÜKETİMİ” grubunu kullanarak, santralin kurulması düşünülen bölgenin ya da karşılaması beklenen kişi başı ortalama yıllık enerji ihtiyacı göz önüne alınacaktır. Biz verileri tezimizde kullanırken karşılaması beklenen kişi başı ortalama yıllık enerji ihtiyacını göz önüne alacağız.

Bu değerleri alırken yaz ve kış sezonu olarak paylaşım yapılmıştır. Çünkü özellikle mahallerin kış ve yaz döneminde nüfus , soğutma ve ısıtma durumlarına bağlı olarak ortalama tüketim değerleri değişmektedir. Bu yüzden; Eylül, Ekim, Kasım, Aralık, Ocak, Şubat, Mart ayları Kış sezonunu, Nisan, Mayıs, Haziran, Temmuz, Ağustos ayları yaz sezonunu oluşturmaktadır. Kullanıcı program içinden dönemlere bağlı istediği ayları seçebilmekte ve dönemlik tüketim değerlerini hesaplabilmektedir. Bizim santralimizin hizmet edeceği bölgenin yaz döneminde kişi başı ortalama tüketimi 3500 kw, kış dönemindeki kişi başı ortalama tüketimi 3000 kw’tır. Programın fonksiyonu appendix 1’de belirtilmiştir.

**ELEKTRİK TÜKETİMİ**

**YAZ DÖNEMİ KİŞİ BAŞI TÜKETİM**

3500 kw

**KIŞ DÖNEMİ KİŞİ BAŞI TÜKETİM**

3000 kw

**DÖNEMSEL TÜKETİM**

KİŞ SEZONU

EKİM

KASIM

ARALIK

OCAK

ŞUBAT

MART

YAZ SEZONU

NISAN

MAYIS

HAZİRAN

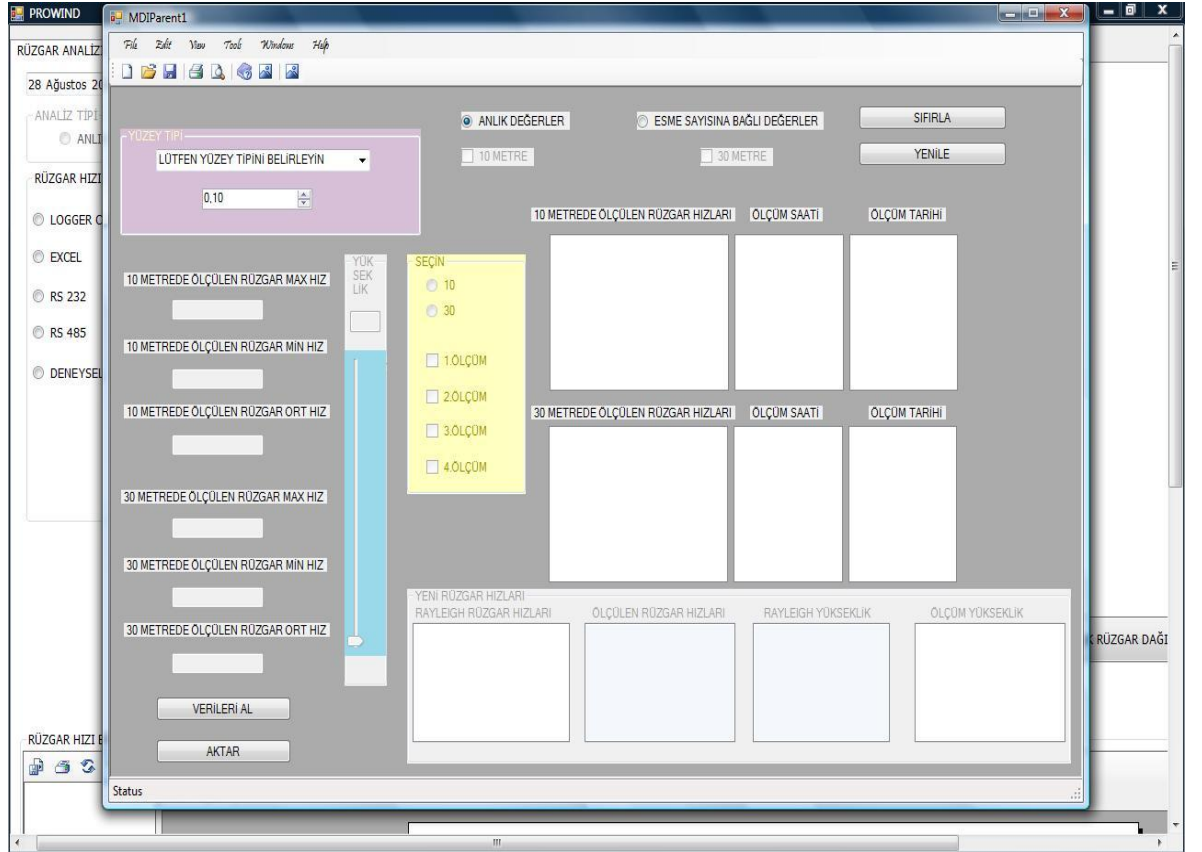
TEMMUZ

AĞUSTOS

77000 kw

Resim 6.2: Ortalama yıllık elektrik tüketimi tablosu

Bir sonraki aşamada, 30 metrede ölçülen rüzgâr hızı değerlerinin Hellman yükseltme bağıntısı kullanılarak kullanıcının belirleyeceği bir yüksekliklerdeki değerleri bulunabilecektir. Bunun için programdaki “HELLMAN YÜKSELTME BAĞINTISINI KULLAN” butonuna tıklayarak, ilgili ekranın açılması sağlanacaktır.



Resim 6.3: Hellman yükseltme bağıntısı ekranı.

Bu kısımdaki “VERİLERİ AL” butonuna tıklanarak, Sql Server’la ilişkilendirilen Excel verilerinin, ölçüm yüksekliğine göre (10 metre veya 30 metre) ilgili alanlara gelmeleri sağlanacaktır.

Resim 6.4: Program içine alınan ölçüm değerleri.

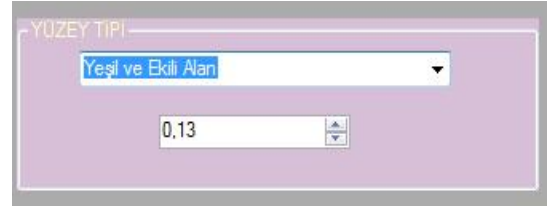
Program verileri ilgili kısımlarda kullanıcıya gösterirken, veri tabanı içinde oluşturulmuş bir sıralama fonksiyonu yardımıyla da, rüzgar hızı verileri arasından ölçülen maksimum, minimum ölçüm değerlerini ve bir ortalama değer hesaplama fonksiyonu aracılığıyla da 30 metrede ölçülen ortalama rüzgar hızı değerini bulmaktadır.

Resim 6.5: Maksimum, minimum ve ortalama rüzgar hızı değerleri.

Hellman arayüzünde bulunan yüzey tipi belirleme grubundan “Yeşil ve Ekili Alan” seçilir. Çünkü, santralin kurulması planlanan bölgenin yapısı bu tipe uymaktadır.

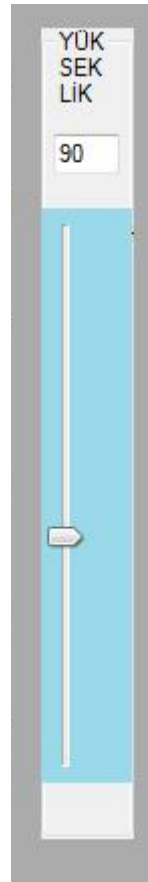


Resim 6.6: Yüzey belirleme ekranı .



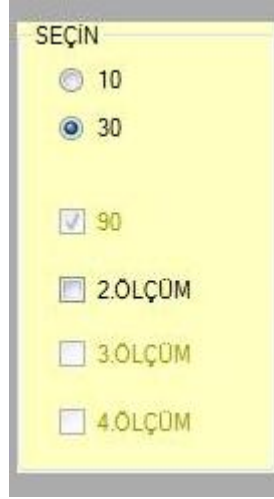
Resim 6.7: Seçilen yüzey tipi ve katsayı değeri.

Yüzey belirleme işlemini tamamladıktan sonra yükseklik belirleme çubuğu (kaydırma çubuğu) kullanılarak türbinlerin çalışması düşünülen yükseklik değeri belirlenip, bu yüksekliğe göre yükseltgenmiş yeni rüzgâr hızı değerleri bulunacaktır. Biz yüksekliğimizi türbin piyasasında şuan itibariyle kullanılan türbinlerin boyutlarını göz önüne alarak, bu değerler 90 ile 100 metre arasında değiştiği için 90 metre olarak kabul ediyoruz.

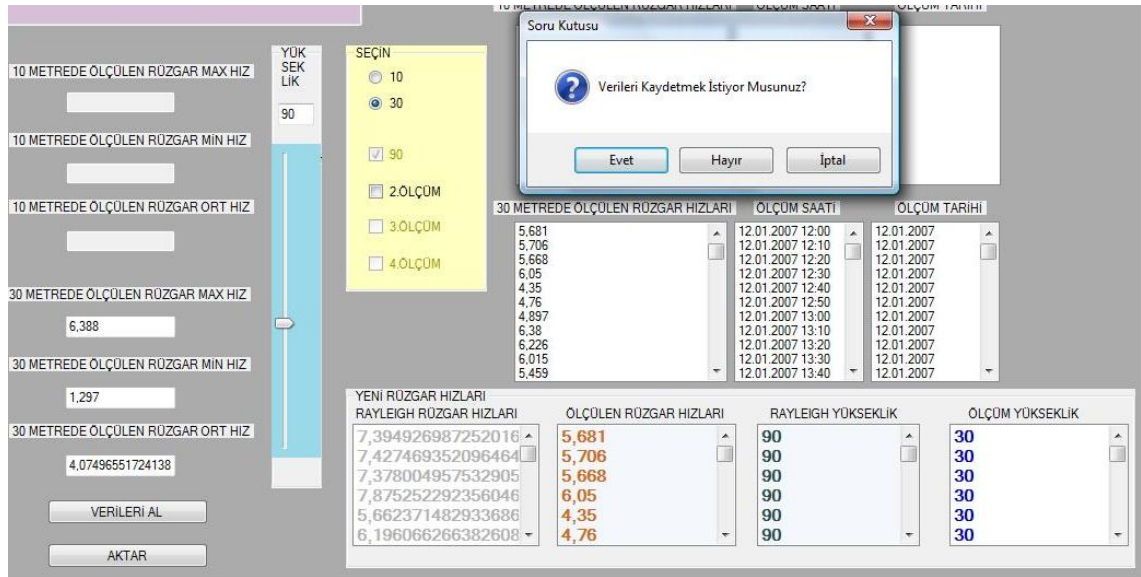


Resim 6.8: Yükseklik seçim çubuğu.

“Seçin” grubunda yer alan ölçüm yüksekliği seçim butonu olan “30” değeri seçilerek, birinci ölçüm kutusu doldurup, 90 metreye yükseltgenmiş ölçüm değerleri bulunabilir.



Resim 6.9: Seçim kutuları.



Resim 6.10: İndirgenmiş rüzgâr hızı değerleri ve kayıt onay ekranı.

Bu şekilde kaydedilen veriler, programın sonraki aşamalarında diğer arayüzlerde (güç analiz ekranı, türbin analiz ekranı, maliyet analiz ekranı) kullanılabilir ve veriler arası ilişkilendirme yapılabilir. Programın fonksiyonu appendix.2’de belirtilmiştir.

“RÜZGÂR ÖLÇÜM KOŞULLARI” ekranında , ölçülen rüzgar hızı değerlerinin taşıdıkları güçleri hesaplayabilmek için öncelikle hava yoğunluğunu  $\rho$  (yoğunluk) hesaplamamız gerekmektedir. Programın fonksiyonu appendix.3’de belirtilmiştir.

Resim 6.11: İndirgenmiş rüzgâr hızı değerleri ve kayıt onay ekranı.

Bundan sonraki aşamada; enerji analizi ekranına geçilip, Betz teoremi yardımıyla indirgenmiş rüzgâr hızı değerlerinin anlık güç karşılıkları bulunabilecektir. Ortalama rüzgâr hızı değeri yardımıyla kayıplarda hesap edilerek, rüzgâr enerjisi optimizasyonu için önemli bir seçim kriteri olan gerçek güç faktörü değeri ve dolayısıyla gerçek güç değerlerine ulaşılabilecektir.

## 6.2. ENERJİ ANALİZ EKLANINDA VERİLERİN İŞLETİLMESİ

Bundan sonraki işlem adımımız “TÜRBİN KATALOG DEĞERLERİNİ KULLAN” seçim kutusundan (Programımız içinde bulunan Vestas firmasına ait rüzgar türbinlerinden) istediğimiz türbin tipini seçerek, Betz teoremi doğrultusunda kullanabileceğimiz türbin tipine göre değişen “SÜPÜRÜLEN RÜZGÂR ALANI”’nı, türbinin kataloğunda belirtilmiş “GÜÇ FAKTÖRÜ HESABI” ve “YAVAŞLATMA FAKTÖRÜ HESABI” değerlerini otomatik olarak hesaplayacağız, bu arayüzümüzde programımızın sahip olduğu algoritma çerçevesinde “TÜRBİN RÜZGÂR HIZLARINA GÖRE ORT GÜÇ HESABI”, “TÜRBİN RÜZGÂR HIZINA GÖRE GÜÇ HESABI”, “YARARLANILAN RÜZGÂR HIZI YÜZDESİ” değerleri bulunacaktır. (Programın fonksiyonu appendix.4’de belirtilmiştir.)

**ENERJİ ANALİZİ**

DENEYSEL    AYLIK    EXCEL    HELLMAN

TÜRBİNE GELEN RÜZGAR HIZI      TÜRBİNDEN ÇIKAN RÜZGAR HIZI       Ortalama Rüzgar Hızına Göre

Rüzgar Giriş Hızı  m/s      Rüzgar Çıkış Hızı  m/s

**GÜÇ FAKTÖRÜ HESABI**

PT\PR=

**YAVAŞLATMA FAKTÖRÜ HESABI**

VK\VR=

**DİĞER FAKTÖRLER**

HAVANIN YOĞUNLUĞU  kg/m<sup>3</sup>

SÜPÜRÜLEN RÜZGAR ALANI  m<sup>2</sup>

TÜRBİN İÇİNDEKİ RÜZGAR HIZI

**YARARLANILAN RÜZGAR HIZI YÜZDESİ**

**KAYIP RÜZGAR HIZI YÜZDESİ**

Resim 6.12: Enerji analizi ekranı yoğunluk, güç faktörü, yavaşlatma faktörü hesabı.

YILLIK ORTALAMA RÜZGAR HIZI:

SEZONLUK ORTALAMA RÜZGAR HIZI:

AYLIK ORTALAMA RÜZGAR HIZI:

GÜNLÜK ORTALAMA RÜZGAR HIZI:

**VESTAS V90-1,8MW**

**GÜÇ HESAPLARI**

ORTALAMA RÜZGAR HIZINA GÖRE GÜÇ HESABI

TÜRBİN RÜZGAR HIZLARINA GÖRE ORTALAMA GÜÇ HESABI

m/s

**TÜRBİN RÜZGAR HIZINA GÖRE GÜÇ HESABI**

TÜRBİN ORTALAMA RÜZGAR HIZINA GÖRE

6,127	
6,154	
6,113	
6,525	
4,691	
5,133	
5,281	
6,88	
6,714	
6,487	

GÜÇ FAKTÖRÜYLE TÜRBİN RÜZGAR HIZLARI

5,753	
5,778	
5,74	
6,127	
4,405	
4,82	
4,959	
6,461	
6,305	
6,091	
5,528	

**TÜRBİN ORTALAMA RÜZGAR GÜCÜ**

885,494	
897,235	
879,429	
1069,491	
397,538	
520,873	
567,154	
1254,218	
1165,57	
1051,037	

GÜÇ FAKTÖRÜYLE TÜRBİN GÜCÜ

733,217	
742,939	
728,195	
885,572	
329,174	
431,299	
469,622	
1038,532	
965,129	
870,292	
650,575	

Resim 6.13: Güç hesaplamaları.

Türbin katalog bilgilerine dayanılarak hazırlanan kayıp değerleriyle ilgili veriler kayıplar ekrandaki yerlere işlenmiştir. Programın fonksiyonu appendix.6'de belirtilmiştir. Bu değerler kayıp hesaplamaları yapılması için ve de gerçek güç faktörü değerinin bulunarak, bir sonraki ekran olan türbin analiz ekranındaki verilerin işlenmesinde kullanılacaktır.



KAYIPLAR

PROFİL KAYBI HESABI:	0,279	
UÇ KAYBI HESABI:	0,952	
GİRDAP KAYBI HESABI:	0,6	
$\lambda$ A(Dizayn Devirlik Sayısı)	12,883	
$\epsilon$ (Kayma Sayısı)	17,857	
$V_{\epsilon}$ (Çevresel Hız)	2049,889	
R(Yarıçap)	45	
CA(Kaldırma Kuvveti Katsayısı)	20	KAYIPLARI HESAPLA
CW(Direnç Kuvveti Katsayısı)	1,12	
Z(Kanat Sayısı)	3	
$n_d$ (Devir Sayısı)	14,5	
$V_r$ (Ortalama Rüzgar Hızı)	5,304	m/s
GERÇEK GÜÇ FAKTÖRÜ:	0,159	
MAKSİMUM GÜÇ:	352,402	W
GERÇEK ORT. GÜÇ:	91,478	W

Resim 6.14: Kayıp hesap grubu ve hesaplanmış kayıp değerleri.

Rüzgârın esme saati, esme yüzdesi, yıllık esme saati verileri program içinde kullanılan Rayleigh dağılım fonksiyonu sayesinde hesaplanabilir. Programın fonksiyonu appendix.7’de belirtilmiştir.

ESME SAATI			
RÜZGAR HIZLARI	ESME SAATI	RÜZGARIN ESME YÜZDESİ	RÜZGARIN YILLIK ESME SAATI
7,395	0,09	8,97	786,39
7,427	0,089	8,89	779,41
7,378	0,09	9,01	790,1
7,875	0,078	7,79	682,49
5,662	0,129	12,92	1132,36
6,196	0,118	11,85	1038,39
6,374	0,114	11,45	1003,55
8,305	0,068	6,76	592,71
8,104	0,072	7,23	634,14
7,83	0,079	7,9	692,11
7,106	0,097	9,69	849,43
6,585	0,11	10,96	960,58
7,651	0,083	8,34	730,67
6,214	0,118	11,81	1034,94
6,251	0,117	11,72	1027,78
5,232	0,136	13,6	1192,54

VERİLERİ KAYDET

Resim 6.15: Esmeye saati verileri hesap grubu.

### 6.3. TÜRBİN ANALİZİ EKRAİNDA VERİLERİNİN İŞLETİLMESİ

Bu kısımda katalog bilgileri program içinde bulunan VESTAS firmasına ait türbin çeşitlerinden faydalanılacaktır. Diğer türbin çeşitlerine ait katalog bilgileri de programa dahil edilerek daha geniş kapsamlı bir çalışma yapılabilir. Tezimiz kapsamında yapacağımız çalışmada VESTAS grubuna ait; V80-2MW, V82-1.65MW, V90-1,8MW, V90-3MW ve V112-3MW(OFFSHORE) türbinleri kullanacağız.

Gelen ekranda “TÜRBİN TİPLERİ” grubundan ilk önce “V90-1,8MW” VESTAS tip türbini seçerek işlemlerimize başlıyoruz. Bu kısımda işlem yapılacak türbin ile ilgili teknik bilgilere de yer verilmiştir. Kullanıcı istediği tip türbin bilgisini buradaki seçim kutusunu kullanarak seçebilir ve istediği karşılaştırmaları da yapabilmektedir.

Ayrıca sistem maliyetini etkileyen, kullanılması gereken optimum türbin sayısı bilgisi de, programın bu kısmında hesaplanmaktadır.

$$\text{Türbin Sayısı} = \frac{\text{Bir Günlük Ortalama Elektrik Enerjisi İhtiyacı}}{\text{Ortalama Rüzgâr Hızında Üretilen Bir Günlük Enerji}} \quad (6.1)$$

ÜRETİLMESİ GEREKEN ENERJİ

YILLIK  
 DÖNEMLİK  
 AYLIK  
 GÜNLÜK  
 SAATLİK

77000

Resim 6.16: Bir günlük ortalama elektrik enerjisi ihtiyacı.

Ortalama rüzgâr hızında üretilen bir günlük enerji;

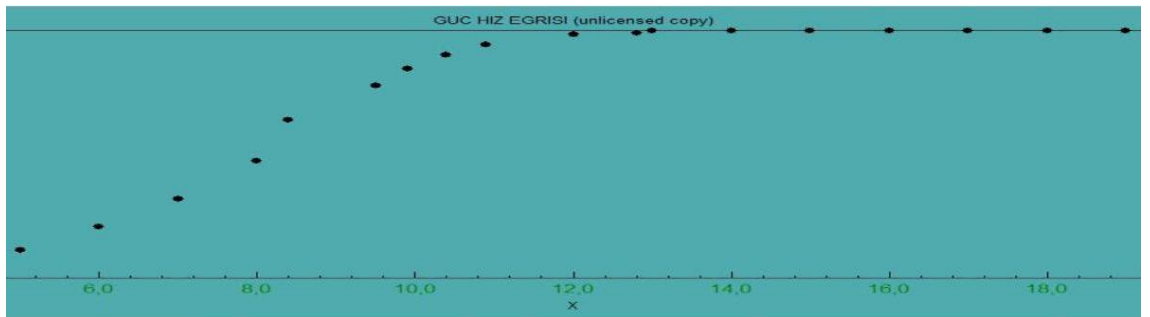
$$375 * 24 = 9000 \text{ kwh'tır.}$$

Türbin sayısı hesap formülünden;

$$\text{Optimum türbin sayısı} = 77000 / 9000$$

$$= 8 \text{ olarak bulunur.}$$

Her bir türbin için ayrı olarak hazırlanmış ve rüzgâr hızı değerlerine göre türbinlerin üretecekleri güçlerin hesaplamasında kullanılan güç-hız eğrileri güç analizi ekranının altında bulunmaktadır.



Resim 6.17: Güç-hız eğrisi

“ÜRETİLEN ENERJİ” grubunda bulunan “ÜRETİLMESİ GEREKEN TOPLAM ENERJİ” kısmına ilgili değer girilerek, “İHTİYAÇ FAZLASI ENERJİ” veya “DIŞARDAN TALEP EDİLEN ENERJİ” kısımlarına ayrıca “TÜRBİN GRUBUNUN YILLIK ENERJİ ÜRETİMİ”, “ÜRETİLMESİ GEREKEN YILLIK ENERJİ”, “YILLIK

*İHTİYAÇ FAZLASI ENERJİ*”, “*YILLIK DIŞARDAN TALEP EDİLEN ENERJİ*” ve “*TÜRBİN GRUBUNUN YILLIK TOPLAM ENERJİ ÜRETİMLERİ*” kısımlarına da ulaşıp, buradaki veriler gözlenebilir.

**TÜRBİN SAĞLAYICILARI**

**FİRMALAR**

**VESTAS**  
 **NORDEX**  
 **SIEMENS**  
 **DEWIND**

**TÜRBİN TİPLERİ**

**V52-850KW**  
**V80-2MW**  
**V82-1,65MW**  
**V90-1,8MW**  
**V90-3MW**  
**V112-3MW(OFFSHORE)**

**RÜZGAR HIZLARI (m/s)**

**7,395**  
**7,427**  
**7,378**  
**7,875**  
**5,662**  
**6,196**  
**6,374**

**TÜRBİN KATALOG GÜÇ DEĞERLERİ (kw)**

**850**  
**850**  
**850**  
**850**  
**375**  
**575**  
**575**

**5,30433793103448**      **m/s**

**375**      **kw**

Resim 6.18: Firmalar, türbin tipleri, rüzgâr hızı verileri.

ÜRETİLEN ENERJİ	
BİR TÜRBİNİN ÜRETTİĞİ ENERJİ:	16233,333 kWh
TÜRBİN GRUBUNUN ÜRETTİĞİ ENERJİ:	129866,66 kWh
ÜRETİLMESİ GEREKEN TOPLAM ENERJİ:	100000
İHTİYAÇ FAZLASI ENERJİ:	29866,66 kWh
DIŞARDAN TALEP EDİLEN ENERJİ:	
<b>TÜRBİN GRUBUNUN YILLIK ENERJİ ÜRETİMİ</b>	
	47401333,3333334 kWh
<b>ÜRETİLMESİ GEREKEN YILLIK ENERJİ</b>	36500000 kWh
<b>YILLIK İHTİYAÇ FAZLASI ENERJİ</b>	10901333,3333335 kWh
<b>YILLIK DIŞARDAN TALEP EDİLEN ENERJİ</b>	
<b>DIŞARDAN TALEP EDİLEN ENERJİ İHT</b>	
<b>TÜRBİN GRUBUNUN YILLIK TOPLAM ENERJİ ÜRETİMİ</b>	47431200.0000001 kWh

Resim 6.19: Üretilmesi gereken enerji.

Türbinlerin karşılaştırılmasında dolayısıyla rüzgâr enerjisi optimizasyonunda kullanılan önemli bir değer “*KARŞILAŞTIRMA ORANI*” dır. Bu değer; “*TÜRBİN GRUBUNUN YILLIK ENERJİ ÜRETİMİ*” nin “*ÜRETİLMESİ GEREKEN YILLIK ENERJİYE*” oranıdır. V90-1.8MW için bu değer *1.6177*’dir.

Türbinlerin karşılaştırılmasında kullanılacak önemli diğer bir veri ise, “*KAPASİTE FAKTÖRÜ*” dır. Kapasite faktörü, bir türbinin bir yılda ürettiği enerjinin aynı türbinin üretebileceği maksimum enerjiye bölünmesi sonucu bulunmaktadır. Bir türbinin üretebileceği maksimum enerji, o türbinin nominal gücünün 8760 saat ile çarpılması ile bulunur. V90-1.8MW için bu değer *0,517* olarak bulunmuştur. Bu son derece iyi bir değerdir. Çünkü kapasite faktörü analizinde dünya ortalaması % 20 civarındadır. Bunun üstündeki yerler enerji üretimi açısından verimli bölgelerdir. Programın fonksiyonu appendix.8’de belirtilmiştir.

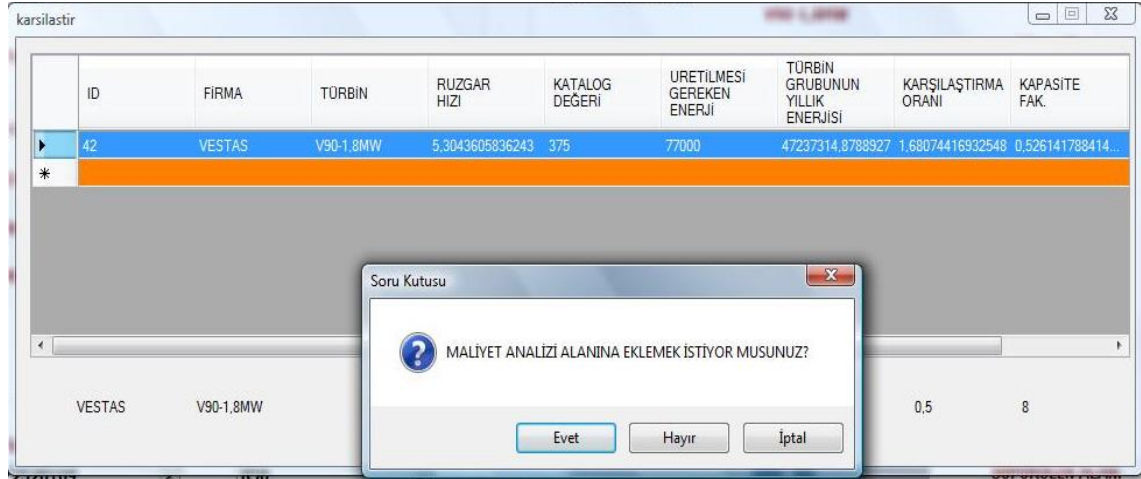
ÜRETİLEN ENERJİ	
BİR TÜRBİNİN ÜRETTİĞİ ENERJİ:	16177,162 <small>kwh</small>
TÜRBİN GRUBUNUN ÜRETTİĞİ ENERJİ:	129417,30 <small>kwh</small>
ÜRETİLMESİ GEREKEN TOPLAM ENERJİ:	80000
İHTİYAÇ FAZLASI ENERJİ:	49417,30 <small>kwh</small>
DIŞARDAN TALEP EDİLEN ENERJİ:	
TÜRBİN GRUBUNUN YILLIK ENERJİ ÜRETİMİ	
	47237314,8788927 <small>kwh</small>
ÜRETİLMESİ GEREKEN YILLIK ENERJİ	29200000 <small>kwh</small>
YILLIK İHTİYAÇ FAZLASI ENERJİ	18037314,8788926 <small>kwh</small>
YILLIK DIŞARDAN TALEP EDİLEN ENERJİ	0
TÜRBİN GRUBUNUN YILLIK TOPLAM ENERJİ ÜRETİMİ	47286732,1799308 <small>kwh</small>
KARŞILAŞTIRMA ORANI	
bul	1,61771626297578

Resim 6.20: Karşılama oranı hesabı.

KAPASİTE FAKTÖRÜNE BAĞLI DEĞERLER	
KAPASİTE FAKTÖRÜ:	0,517461232859
ÜRETİLEN ENERJİ:	8159328,719723
ÜRETİLEBİLECEK MAX.ENERJİ:	15768000

Resim 6.21: Kapasite faktörü hesabı.

V90-1,8MW türbinine ait değerler, karşılaştırma alanına eklenir. Bu kısımda da eğer elde daha başka türbinler ve onlara ait işletim değerleri varsa karşılaştırılıp, son analiz kısmı olan maliyet analizi kısmına iletilirler.



Resim 6.22: Türbin karşılaştırma ekranı.

#### 6.4. MALİYET ANALİZİ EKLANINDA VERİLERİN İŞLETİLMESİ

Son ve en önemli ekran olan maliyet analizi ekranında, türbin analiz ekranında seçilen türbinler ve onlara ait kritik değerler kullanılacaktır.

“*TÜRBİN MARKASI*” seçim kutusu kullanılarak maliyet analizi yapılacak olan türbin tipi seçilerek işlemlere başlanır.

İlgili türbin tipi seçildikten sonra, daha önceki ekranlarda veri tabanına aktarılmış olan, “*TÜRBİN KATALOG GÜÇ DEĞERİ*”, “*TÜRBİN SAYISI*” ve “*TOPLAM ÜRETİLEN ENERJİ*” verileri ekranda gözükecektir.

Daha sonra, Vestas’ın internet sitesinden veya katalog bilgileri vasıtasıyla ulaşabileceğimiz türbin satış bilgilerinden “*TÜRBİN FABRİKA ÇIKIŞ FİYATI*” bilgisi girilerek ve “*TÜRBİN YAN MALİYETİ*” ekranında bulunan ilgili değerler kalem kalem girilerek ya da yüzde olarak belirtilerek toplam türbin maliyeti hesaplanacaktır.

“*ORTALAMA EKONOMİK ÖMÜR*” alanına türbinin katalogunda Vestas firması tarafından belirtilen ya da kullanıcı tarafından işletilmesi düşünülen ömür süresi

girilerek, *birim enerji maliyeti* hesaplanacaktır. Bu maliyet türbin seçiminde ve santral kurulumunda önemli bir değer olup, türbine yapılan yatırımın geri dönüş süresinin bulunmasında kullanılacaktır.

$$\text{Birim Enerji Maliyeti} = \frac{\text{Toplam Maliyet}}{\text{Toplam Üretilen Enerji}} \quad (6.2)$$

*Programda gerekli işlemleri gerçekleştirdiğimizde V90-1,8MW tipinde kullanacağımız türbinimizin birim enerji maliyeti 0,042 \$/kwh olarak bulunur.*

Resim 6.23: Birim enerji maliyeti hesabı.

Son aşamada da sistemin geri ödeme süresinin bulunması vardır. Geri ödeme süresini hesaplamak için; birim satış fiyatından (TEDAŞ birim satış fiyatı) birim enerji maliyeti çıkarılarak, elde edilen rakam üretilen yıllık enerji miktarıyla çarpılmış ve yıllık net kar oluşturulmuştur. Geri ödeme süresi, türbin fiyatının bu kara bölünmesiyle yıl bazında hesaplanmıştır. Programın fonksiyonu appendix.9'de belirtilmiştir.



Resim 6.24: Tedaş birim fiyat ekranı.

Tedaş birim fiyatının Dolar olarak karşılığı yaklaşık 0,08\$/kWh'tir.

TÜRBİN	V90-1,8MW
ÜRETİLEN ENERJİ(kwh)	47401333,333
BİRİM SATIŞ FİYATI(\$/kwh)	0,08
BİRİM MALİYET(\$/kwh)	0,042
TÜRBİN FİYATI(\$)	8000000
GERİ ÖDEME SÜRESİ(YIL)	4,441

Resim 6.25: Geri ödeme süresinin bulunması.

## 7. BULGULAR

Tez çalışmasında, verilerin işlenmesinde kullanılan rüzgâr optimizasyon programı aracılığı ile VESTAS V90 - 1,8 MW'lık türbin tipinin seçimiyle, Malzeme Ve Yöntem kısmında yapılan analizler, VESTAS'ın diğer türbin tipleri de seçilerek aynı koşullar altında yapılmıştır. Bu şekilde elde edilen türbinlere ait analiz sonuçları bu kısımda ele alınacaktır.

ID	FIRMA	TURBIN	RÜZGAR HIZI	KATALOG DEĞERİ	ÜRETİLMESİ GEREKEN ENERJİ	TÜRBİN GRUBUNUN YILLIK ENERJİSİ	KARŞILAŞTIRMA ORANI	KAPASİTE FAK.
63	VESTAS	V80-1,65MW	RÜZGAR HIZI (m/s)		77000	51162770,7006373	1,75214968152867	0,505919063243...
64	VESTAS	V90-3MW	5,29448189655181	225	77000	102284374,542795	3,50288953913682	0,47664913319632
65	VESTAS	V80-2MW	5,29495123152709	200	77000	77948025,6283891	2,66945293247908	0,451969360933...
62	VESTAS	V90-1,8MW	5,30315229885058	375	77000	47212319,7239794	1,61686026451984	0,517064937277...
**	-1							

Resim 7.1: Türbin karşılaştırma ekranı.

Resim 7.1'in ve Tablo 7.1'in incelenmesi sonucunda kapasite faktörüne bakıldığında, en verimli türbinin, %51'lik oranla Vestas V90-1,8MW olduğu görülmektedir. Ondan sonra %50'lik oranla Vestas V82-1,65MW'lık türbin gelmektedir. Karşılaştırma oranlarına bakıldığında tüm türbinlerin fazlasıyla istenen kapasiteyi karşıladığı görülmektedir. Kapasite faktörü bir türbinin bir yılda ürettiği enerjinin, aynı türbinin üretebileceği maksimum enerjiye bölünmesi ile ortaya çıkmaktadır. Bir türbinin bir yılda üretebileceği maksimum enerji, o türbinin nominal gücünün 8760 saat ile çarpılması sonucunda bulunur.

Kapasite faktörü analizinde dünya ortalaması %20 civarındadır. Bu oranın üzerindeki kapasite faktörüne sahip yerler, enerji üretimi açısından verimlidir.

Tablo 7.1: Karşılaştırma tablosu

<b>Firma</b>	<b>Türbin</b>	<b>Rüzgâr Hızı (m/s)</b>	<b>Katalog Değeri (kw)</b>	<b>Üretilmesi Gereken Enerji (kwh)</b>	<b>Türbin Grubunun Yıllık Enerjisi (kwh)</b>	<b>Karşılama Oranı</b>	<b>Kapasite Faktörü</b>	<b>Türbin Sayısı</b>
Vestas	V90-3MW	5,304	225	77000	102093541	3,411	0,47	14
Vestas	V82-1,65MW	5,304	300	77000	51039166,66	1,705	0,50	10
Vestas	V80-2MW	5,304	200	77000	77769333,33	2,598	0,45	16
Vestas	V90-1,8MW	5,304	375	77000	47401333,33	1,617	0,51	8

Tablo 7.1'in incelenmesi ile Vestas V90-1,8MW'in en verimli türbin olduğu görülebilmektedir (%51). Vestas V80-2 MW'lık türbinin de %45'lik oranla diğer incelenen türbinlere kıyasla en verimsiz türbin olmasına karşın, yine de dünya ortalamasının üstünde bir kapasite faktörü değerine sahip olduğu görülmektedir. Ayrıca bu oranlar bize, rüzgar analizini yaptığımız bölgenin rüzgar enerjisi bakımından ne kadar verimli bir bölge olduğunu da göstermektedir.

Maliyet analizlerine bakmadan önce şunu belirtmemiz gerekmektedir; birim enerji maliyeti, enerji sistemlerinin değerlendirilmesinde en önemli parametredir. Bizim çalışmamızda da, önerilen sistemin diğer sistemlerle kıyaslanabilmesi için, birim enerji maliyeti hesabı yapılmıştır. Kıyaslamamız, şehir elektrik şebekesi kWh ücreti ile üretilen birim enerjinin maliyeti arasında yapılacaktır.

Karşılaştırma değerlerimizi görebilmemiz için, maliyet analizi arayüzümüzü kullanarak, ilgili veriler ışığında türbinlerimiz için geri ödeme sürelerini her biri için tek tek hesaplayacağız.

TÜRBİN	V90-1,8MW
ÜRETİLEN ENERJİ(kwh)	47212319,724
BİRİM SATIŞ FİYATI(\$/kwh)	0,08
BİRİM MALİYET(\$/kwh)	0,042
TÜRBİN FİYATI(\$)	8000000
GERİ ÖDEME SÜRESİ(YIL)	4,459

Resim 7.2: V90-1,8MW geri ödeme süresi.

TÜRBİN	V90-3MW
ÜRETİLEN ENERJİ(kwh)	102093541,667
BİRİM SATIŞ FİYATI(\$/kwh)	0,08
BİRİM MALİYET(\$/kwh)	0,051
TÜRBİN FİYATI(\$)	21000000
GERİ ÖDEME SÜRESİ(YIL)	7,093

Resim 7.3: V90-3MW geri ödeme süresi.

TÜRBİN	V80-1,65MW
ÜRETİLEN ENERJİ(kwh)	51039166,667
BİRİM SATIŞ FİYATI(\$/kwh)	0,08
BİRİM MALİYET(\$/kwh)	0,049
TÜRBİN FİYATI(\$)	10000000
GERİ ÖDEME SÜRESİ(YIL)	6,32

Resim 7.4: V80-1,65MW geri ödeme süresi.

<b>TÜRBİN</b>	V80-2MW
<b>ÜRETİLEN ENERJİ(kwh)</b>	77769333,333
<b>BİRİM SATIŞ FİYATI(\$/kwh)</b>	0,08
<b>BİRİM MALİYET(\$/kwh)</b>	0,064
<b>TÜRBİN FİYATI(\$)</b>	20000000
<b>GERİ ÖDEME SÜRESİ(YIL)</b>	16,073

Resim 7.5: V80-2MW geri ödeme süresi.

Herbir türbin tipi için verilen tablolarda, geri ödeme süresi hesaplanırken; birim satış fiyatından birim enerji maliyeti düşülmüş, çıkan rakam yıllık enerji miktarı ile çarpılarak yıllık net kar oluşturulmuştur. Geri ödeme süresi, türbin fiyatlarının bu kara bölünmesiyle yıl bazında hesaplanmıştır. Buna göre V90-1,8 MW türbin tipi 4 sene 4 ay gibi süre zarfında kendisine yapılan maliyetleri çıkarabilmektedir. Diğer türbinlerle kıyaslandığında en kısa süre zarfında kendini amorti eden türbinin V90-1.8MW olduğu görülmektedir.

Kapasite faktörü, karşılama oranı ve geri ödeme süreleri dikkate alındığında Vestas V90-1,8 MW tipli türbinin belirlediğimiz bölge için en uygun türbin olduğu görülmektedir.

## 8. TARTIŞMA VE SONUÇ

Şüphesizki enerji, insan hayatında tartışılmaz bir öneme sahiptir. Enerjiyi insanoğlunun ilk kullanması, ateşin bulunması ile başlamıştır. Enerji kelimesinin kullanılmaya başlaması 17. yüzyılda buhar makinalarının icadıyla olmuştur. Bu tarihten itibaren kömür, gaz ve yağ sıkça kullanılan kimyasal maddeler olmuşlar, ülkelerin endüstriyel gelişimlerinde belirleyici rol oynamışlardır. Elektriğin bulunuşu ve evlerimizin içine kadar girmesiyle enerji konusu günlük hayatın bir parçası haline gelmeye başlamıştır. Toplumların gelişmesinde enerji vazgeçilmez unsurlardan biri haline gelmiştir. Özellikle elektrik enerjisi, tüm enerji formları arasında günlük hayatta en çok kullanılan ve ihtiyaç duyulan enerji çeşididir.

Kalkınmakta ve nüfusu hızla büyümekte olan ülkemizin, enerjiye bilhassa elektrik enerjisine olan ihtiyacı her geçen gün artmaktadır. Kalkınmasını gerçekleştirebilmek için ülkemiz, elektrik enerji ihtiyacını karşılamak zorundadır. Şu an itibariyle elektrik ihtiyacının büyük çoğunluğunu; petrol, doğalgaz ve kömür gibi birincil enerji kaynaklarından sağlamaktadır. Bu kaynaklardan özellikle petrol ve doğalgazda tam anlamıyla dışa bağımlılık söz konusudur. Sahip olduğumuz özkaynaklarımızın kullanımı, konvansiyonel enerji kullanımına oranla çok düşük düzeydedir. Özellikle ülkemizin zengin su kaynakları, yoğun oranda güneş alan bölgeleri, zengin yer altı kaynakları ve rüzgar açısından verimli bölgeleri bulunmaktadır. Tüm bu yenilenebilir çatı altında tanımlayabileceğimiz kaynaklarımızı etkin şekilde kullanmamız, enerji ve ona bağlı kalkınma problemimizin çözümü olma özelliği taşımaktadır.

Sadece rüzgar enerjisi potansiyelimiz teorik olarak 83 GW'tır. Ege, Marmara ve Doğu Akdeniz kıyılarımız rüzgar enerjisi bakımından zengin bölgelerdir.

Ülkemizin önündeki bir başka ve önemli problem de; zaten yetersiz olan elektrik enerjisi karşılama potansiyelimizin, kayıplarla, kaçaklarla ve uygunsuz santral kurulum ve işletmesiyle daha da çıkmaz bir hal almış olmasıdır. Sadece hat kaybımız, toplam üretimimizin % 21,4'ü gibi çok yüksek bir rakamdır. Dünya standartlarında bu oran % 5'ler civarındadır. Buradan çıkarılacak önemli sonuçlardan biri de; sahip olduğumuz

santrallerimizde ve elektrik alt yapımızda revize uygulamalar yapmamız gerektiğidir. İşte bu uygulamalardan bir tanesi de tezimizin konusu olan optimizasyondur.

Optimizasyon, mevcut koşullar altında en iyiyi arama sürecidir. Bu sürecin mevcut elektrik sistemlerimize uygulanmasıyla; mevcut durumdaki santrallerimizin, elektrik alt yapımızın işletme problemleri düzelecek, üretim süreci daha sistematik olacak, hat kayıplarımız önlenecek, kullanacağımız enerjimizin kalitesi artacaktır. Yeni santrallerin kurulumu aşamasında bu sürecin uygulanması ile, uygun bölgeye uygun santralin, daha az maliyetle kurulmasını, teknik anlamda daha az problemlili olmasını ve daha uzun ömürlü olmasını sağlayacaktır.

Tezimizde optimizasyon sürecini oluşturduğumuz optimizasyon programı sayesinde analiz ederek, yenilenebilir enerji kaynaklarından günümüzde en çok tercih edilen, ucuz, temiz ve bol olan rüzgar enerjisini, bu optimizasyon süreci içerisinde ele aldık. Vardığımız sonuçları şu şekilde özetleyebiliriz:

- Bu tez çalışmasında, Çorlu Ulaş mevkinde konumlu, Can Enerji Üretim A.Ş.'ye ait rüzgâr ölçüm direğinden alınan rüzgâr hızı verileri ve oluşturulan optimizasyon programı kullanılarak ilgili analizler (rüzgâr hızı analizi, güç analizi, türbin analizi ve maliyet analizi) yapılmış ve veriler eşliğinde karşılaştırılarak, bölge için uygun türbin seçimi yapılmaya çalışılmıştır.
- Programda, santralin besleyeceği bölgenin elektrik ihtiyacı yaz ve kış sezonu olarak iki ayrı dönem içinde düşünülmüş; yaz dönemi için (Mayıs, Haziran, Temmuz, Ağustos, Eylül) puant elektrik enerjisi ihtiyacı 3500 kW, kış sezonu içinse (Ekim, Kasım, Aralık, Ocak, Şubat, Mart, Nisan) 3000 kW olarak alınmıştır. Buradan bölgenin bir günlük elektrik enerjisi ihtiyacı hesaplanmış ve toplam 77000 kWh olarak bulunmuştur.
- Rüzgar hızı ölçüm cihazının excel kayıtları, program içine alınıp ilk önce rüzgar hızı analizi yapılarak, 30 metrede ölçülen maksimum rüzgar hızı değeri (6,388 m/s), minimum rüzgar hızı değeri (1,297 m/s) ve ortalama rüzgar hızı (4,079 m/s) değerleri bulunarak, Hellman yükseltme bağıntısı uygulanıp 90 metredeki rüzgar hızı değerlerine yükseltgenmiştir.



- 90 metredeki hız değerlerine yükseltgenen rüzgar hızı değerlerinin, güç analiz ekranında, program içinde bulunan Vestas firmasının türbinlerinin (V90-3MW, V90-1,8MW, V82-1,65MW, V80-2MW) katalog bilgileri kullanılarak ve Betz teoremi uygulanarak, taşıdıkları güç değerleri tespit edilmiş, ortalama rüzgar hızında sağlanabilecek ortalama güç değeri tespit edilmiş, gerçek güç faktörleri ve türbin kayıpları bulunmuştur.
- Türbin analiz ekranında, Vestas firmasına ait türbinler için, bulunan rüzgar hızı değerlerinde üretebilecekleri güç değerleri bulunmuş, enerji ihtiyaçlarını karşılama oranları ve kapasite faktörleri de bulunarak, karşılaştırma alanına eklenmişlerdir.
- Maliyet analizi ekranında karşılaştırılması yapılan türbinlerin, birim enerji maliyetleri oluşturulmuş ve türbin kurulumu ve bakımı için olan maliyetlerin geri dönüş süreleri incelenmiştir.
- Tüm bu veriler ışığında yapılan çalışmada ;

Firma	Türbin	Rüzgâr Hızı (m/s)	Katalog Değeri (kw)	Üretilmesi Gereken Enerji (kwh)	Türbin Grubunun Yıllık Enerjisi (kwh)	Karşılama Oranı	Kapasite Faktörü	Türbin Sayısı
Vestas	V90-3MW	5,304	225	77000	102093541	3,411	0,47	14
Vestas	V82-1,65MW	5,304	300	77000	51039166,66	1,705	0,50	10
Vestas	V80-2MW	5,304	200	77000	77769333,33	2,598	0,45	16
Vestas	V90-1,8MW	5,304	375	77000	47401333,33	1,617	0,51	8

Tüm türbin tiplerinin, kapasiteyi fazlasıyla karşıladığı görülmüştür (Karşılama Oranı). Türbin seçiminde önemli bir kriter olan kapasite faktörü değerlerine bakıldığında ise Vestas V90-1,8MW tipindeki türbinin % 51'lik oranla en uygun

türbin olduğu, onu % 50'lik oranla Vestas V82-1,65MW'lık türbinin takip ettiği gözlenmiştir.

Türbinler arası maliyet ve geri dönüşüm oranlarına bakıldığında da, birim maliyeti 0,042 \$/kwh olan Vestas V90-1,8MW tipindeki türbinin en düşük birim maliyete sahip türbin olduğu ve geri dönüş süresinin de yine 4 yıl 4 ay gibi bir süreyle, en erken bu türbinde olduğu saptanmıştır.

Tüm bu veriler ışığında, Vestas V90-1,8 MW'lık türbin tipinin, bu bölge için tasarlanan rüzgar santraline en uygun türbin tipi olduğu ve işletilmesinde en optimum sonuçları vereceği gözlenmektedir.

Günümüzde enerjinin dengeli kullanımı, sürdürülebilir politikalarla desteklenmesi, yenilenebilir ve çevre dostu olması büyük önem taşımaktadır. Ülkelerin gelişmişlik düzeylerinin belirlenmesinde, bunlar belirleyici birer faktör olmaktadır.

Enerjide dışa bağımlılığı yenebilen, istikrarlı politikalarla her yönden gelişimini sürdürebilen ve çağdaş gereksinimlerin karşılanabilmesi için bilgi ve teknolojik ilerlemelere açık uluslar, geleceğin dünyasına şekil vereceklerdir. Bu yönden baktığımız zaman ülkemizin de vakit kaybetmeden bu belirtilenleri hayata geçirmesi bir zorunluluktur.

Sonuç itibarıyla; rüzgar hızları bakımından verimli olan yerlere kurulacak rüzgar enerjisi santralleri ile verimli, ucuz, temiz ve yenilenebilir bir enerji ile elektrik üretimi sağlanabilir. Hazırladığımız tez çalışmamızda elde ettiğimiz veriler ve bulgulara göre şunu söyleyebiliriz ki; enerji üretim tesisinin seçiminde dikkat edilecek en önemli husus, sistemin kurulacağı bölgenin şartlarına en uygun enerji tipinin saptanarak, bu doğrultuda optimum enerji üretebilecek santral kurulması gerekliliğidir.

## KAYNAKLAR

- A. Janssen S. F. Lienin, F. Gassmann and W. Alexander “ Model aided policy development for the market penetration of natural gas vehicles in Switzerland, *Transportation Research Book Part A 40 (2006) 316–333, Zurich*
- A.Şenkal, N. Çetin “Türkiye’de Kurulu Olan Büyük Güçlü Rüzgâr Santrallerinin Kapasite Faktörlerine Genel Bir Bakış, *Ege Bölgesi Enerji Forumu” 12-13 Ekim 2009, Denizli.*
- Bayrakçı, C. H. "Türkiye'de Rüzgâr Enerjisi ve Potansiyel Belirleme Çalışmaları. " *Mühendis ve Makine dergisi Cilt 48 Sayı 569: 78-80 s (2007).*
- Ç.Şen (2003) .“Gökçeada’nın Elektrik İhtiyacının Rüzgâr Enerjisi İle Sağlanması”, *Dokuz Eylül Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Enerji Bölümü, Ağustos 2003, İzmir.*
- E. ACAR, A. DOĞAN, VII. Ulusal Temiz Enerji Sempozyumu, “UTES’2008 Potansiyeli Ve Çevresel Etkilerinin Değerlendirilmesi” *Araştırması, 17-19 Aralık 2008, İstanbul*
- E. Demirci, İ. Şenlik Ondokuz Mayıs Üniversitesi Yerleşkesi Rüzgâr Enerjisi Potansiyelinin Dönemsel Değerlendirilmesi. *Ondokuz Mayıs Üniversitesi, Samsun MYO Elektrik Programı, Çalışma Raporu, 2009, Samsun.*
- EMD International, WindPRO Yazılımı Çeşitli Katalog Bilgileri, *Anonim, 2010.*
- E. Şekerci, M. Elfarra, C. Albasan, Rüzgâr Türbini Tahmini Yıllık Enerji Üretim Hesaplama Yöntemleri Ve Araziye Uygun Rüzgâr Türbini Seçimi. *RÜGES 2009 2.Rüzgar Sempozyumu, 4-5 Haziran 2009, Ondokuz Mayıs Üniversitesi, Samsun.*
- M. A. Aktacir vd., “Fotovoltaik-Rüzgâr Hibrid Güç Sistemi Uygulaması”, *Yeni Enerji Dergisi , Yıl 1, Sayı 3, 2008.*
- M. N. Sastri, *Introduction to Nuclear Science Book, Affiliated East-West Press Private Limited, New Delhi (1983).*
- M.Ragheb “*Wind Energy and its Extraction through Wind Mills” book, 2010*

M. TOLAY, H. YAMANKARADENİZ, VII. Ulusal Temiz Enerji Sempozyumu, “UTES’2008 Hayvansal Atıklardan Biyogaz Üretimi” Araştırması, 17-19 Aralık 2008, İstanbul

N. Kılıç, *Dünyanın Önemli Doğal Kaynağı “Rüzgâr Enerjisi” Ar&Ge Bülteni (2009).*

N. Nurbay, A. Çınar, Rüzgâr Türbinlerinin Çeşitleri ve Birbirleriyle Karşılaştırılması Özdamar, A., (2001). “Wind Energy Utilization in a House in Izmir, Turkey”, *International Journal of Energy Research 2001; 25:253-261.*

Özdamar, A., (2002). “Rüzgâr Enerjisi Tesisleri”, Ders Notları, Yayınlanmamış, Ege Üniversitesi, Güneş Enerjisi Enstitüsü, İzmir.

Ö. Kara, A. Özdamar, N. Özbalta, “Yükselen Hava Akımlı Rüzgâr Türbinleri”, Ege Üniversitesi Güneş Enerjisi Enstitüsü Makina Mühendisliği Bölümü Ders Notları, 2003

Pure Power Wind energy targets for 2020 and 2030, *The European Wind Energy Association Report, EWEA (2009)*

R.M. Zavadil, “Wind Generation Technical Characteristics for the NYSERDA Wind Impacts Study” report, November 24 2003

R. Narayan and B. Viswanathan, “Chemical and Electrochemical energy system” *Research, Universities Press, 1998, pp 28-35.*

Wind In Power 2009 European Statistics, *The European Wind Energy Association Report, EWEA ( 2009 )*

Y. Malkoç, “Türkiye’de Elektrik Enerjisi İhtiyacının Karşılmasında Rüzgâr Enerjisinin Yeri”, *Yeni Enerji Dergisi Yıl 1 Sayı 4, 2008.*

[http://en.wikipedia.org/wiki/Nuclear\\_power](http://en.wikipedia.org/wiki/Nuclear_power)

[http://en.wikipedia.org/wiki/Natural\\_gas](http://en.wikipedia.org/wiki/Natural_gas)

<http://en.wikipedia.org/wiki/Petro>

<http://yenilenebilirenerjikaynaklari.ws.tc>

[www.danishwind.com](http://www.danishwind.com)

[www.dewind.de](http://www.dewind.de)

[www.epkd.gov.tr](http://www.epkd.gov.tr)

[www.gencistatistik.com](http://www.gencistatistik.com)

[www.greenenergy.org.uk](http://www.greenenergy.org.uk)

[www.newhomewindpower.com](http://www.newhomewindpower.com)

[www.powered.org.uk](http://www.powered.org.uk)

[www.siemens.com](http://www.siemens.com)

[www.turkcebilgi.com](http://www.turkcebilgi.com)

[www.vestas.com](http://www.vestas.com)

[www.windpower.org](http://www.windpower.org)

**EKLER****LOGGER CİHAZI 12.01.2007-13.01.2007 ARASI ÇORLU CAN ENERJİ ELEKTRİK ÜRETİM SANTRALİ RÜZGAR ÖLÇÜM DEĞERLERİ:**

ID	Verituru	Olcumyukseklk	OrtHiz	MinHiz	MaxHiz	RuzHizlari	OlcumSaati
1	Aylık	30	5.681	0	15,6	5,681	12.01.2007 12:00
2	Aylık	30	12,5	4,33	6,41	5,706	12.01.2007 12:10
3	Aylık	30	14,35	3,79	6,61	5,668	12.01.2007 12:20
4	Aylık	30	12,5	2,79	4,847	6,05	12.01.2007 12:30
5	Aylık	30	14,35	2,75	5,63	4,35	12.01.2007 12:40
6	Aylık	30	8,25	3,03	6,51	4,76	12.01.2007 12:50
7	Aylık	30	8,25	3,17	6,19	4,897	12.01.2007 13:00
8	Aylık	30	8,25	4,65	7,13	6,38	12.01.2007 13:10
9	Aylık	30	8,25	4,25	7,23	6,226	12.01.2007 13:20

10	Aylık	30	8,25	3,85	6,93	6,015	12.01.2007 13:30
11	Aylık	30	8,25	2,91	7,35	5,459	12.01.2007 13:40
12	Aylık	30	8,25	2,81	6,61	5,059	12.01.2007 13:50
13	Aylık	30	8,25	4,55	6,71	5,878	12.01.2007 14:00
14	Aylık	30	8,25	2,81	6,27	4,774	12.01.2007 14:10
15	Aylık	30	8,25	2,65	5,63	4,802	12.01.2007 14:20
16	Aylık	30	8,25	1,61	5,29	4,019	12.01.2007 14:30
17	Aylık	30	8,25	2,45	5,05	4,32	12.01.2007 14:40
18	Aylık	30	8,25	2,35	4,91	3,915	12.01.2007 14:50
19	Aylık	30	8,25	2,31	4,63	3,675	12.01.2007 15:00
20	Aylık	30	8,25	2,35	5,65	4,138	12.01.2007 15:10
21	Aylık	30	8,25	2,53	5,55	4,009	12.01.2007 15:20
22	Aylık	30	8,25	2,69	4,93	4,106	12.01.2007 15:30
23	Aylık	30	8,25	2,53	5,13	3,834	12.01.2007 15:40
24	Aylık	30	8,25	2,09	4,31	3,388	12.01.2007 15:50
25	Aylık	30	8,25	1,69	4,11	2,529	12.01.2007 16:00
26	Aylık	30	8,25	1,75	3,25	2,227	12.01.2007 16:10
27	Aylık	30	8,25	1,13	3,29	1,694	12.01.2007 16:20
28	Aylık	30	8,25	1,07	4,39	1,574	12.01.2007 16:30
29	Aylık	30	8,25	2,55	4,39	3,694	12.01.2007 16:40
30	Aylık	30	8,25	2,27	4,79	3,47	12.01.2007 16:50
31	Aylık	30	8,25	1,95	4,95	3,745	12.01.2007 17:00

32	Aylık	30	8,25	2,85	4,91	4,262	12.01.2007 17:10
33	Aylık	30	8,25	3,79	4,85	4,672	12.01.2007 17:20
34	Aylık	30	8,25	3,89	5,25	5,021	12.01.2007 17:30
35	Aylık	30	8,25	4,09	5,37	4,976	12.01.2007 17:40
36	Aylık	30	8,25	3,65	4,93	4,542	12.01.2007 17:50
37	Aylık	30	8,25	4,03	4,91	4,514	12.01.2007 18:00
38	Aylık	30	8,25	4,65	5,33	5,187	12.01.2007 18:10
39	Aylık	30	8,25	4,45	5,63	5,379	12.01.2007 18:20
40	Aylık	30	8,25	5,09	5,71	5,584	12.01.2007 18:30
41	Aylık	30	8,25	4,55	5,55	5,309	12.01.2007 18:40
42	Aylık	30	8,25	4,25	6,07	5,259	12.01.2007 18:50
43	Aylık	30	8,25	1,01	4,89	3,078	12.01.2007 19:00
44	Aylık	30	8,25	0,49	4,65	2,056	12.01.2007 19:10
45	Aylık	30	8,25	2,05	5,17	3,296	12.01.2007 19:20
46	Aylık	30	8,25	3,43	5,03	4,531	12.01.2007 19:30
47	Aylık	30	8,25	2,39	4,47	3,412	12.01.2007 19:40
48	Aylık	30	8,25	3,25	4,33	3,804	12.01.2007 19:50
49	Aylık	30	8,25	3,69	5,37	4,869	12.01.2007 20:00
50	Aylık	30	8,25	4,89	5,73	5,526	12.01.2007 20:10
51	Aylık	30	8,25	4,89	5,63	5,473	12.01.2007 20:20
52	Aylık	30	8,25	3,53	4,81	3,975	12.01.2007 20:30
53	Aylık	30	8,25	3,47	4,31	3,716	12.01.2007 20:40



54	Aylık	30	8,25	0	6,49	5,681	12.01.2007 20:50
55	Aylık	30	8,25	4,432	6,606	5,706	12.01.2007 21:00
56	Aylık	30	8,25	4,203	7,063	5,668	12.01.2007 21:10
57	Aylık	30	8,25	2,83	6,377	4,847	12.01.2007 21:20
58	Aylık	30	8,25	3,059	5,919	4,35	12.01.2007 21:30
59	Aylık	30	8,25	3,174	6,72	4,764	12.01.2007 21:40
60	Aylık	30	8,25	3,631	6,72	4,897	12.01.2007 21:50
61	Aylık	30	8,25	4,775	7,292	6,388	12.01.2007 22:00
62	Aylık	30	8,25	4,432	7,75	6,226	12.01.2007 22:10
63	Aylık	30	8,25	3,059	5,919	4,35	12.01.2007 22:20
64	Aylık	30	8,25	3,174	6,72	4,64	12.01.2007 22:30
65	Aylık	30	8,25	2,83	7,178	5,059	12.01.2007 22:40
66	Aylık	30	8,25	4,89	7,063	5,878	12.01.2007 22:50
67	Aylık	30	8,25	1,229	5,347	4,32	12.01.2007 23:00
68	Aylık	30	8,25	2,258	5,233	3,915	12.01.2007 23:10
69	Aylık	30	8,25	2,258	5,347	4,32	12.01.2007 23:20
70	Aylık	30	8,25	2,373	5,118	3,675	12.01.2007 23:30
71	Aylık	30	8,25	2,945	6,148	4,138	12.01.2007 23:40
72	Aylık	30	8,25	2,602	5,919	4,009	12.01.2007 23:50
73	Aylık	30	8,25	2,83	5,233	4,106	12.01.2007 00:00
74	Aylık	30	8,25	2,716	5,347	3,834	13.01.2007 00:10
75	Aylık	30	8,25	1,801	4,546	3,388	13.01.2007 00:20

76	Aylık	30	8,25	0	3,974	2,529	13.01.2007 00:30
77	Aylık	30	8,25	0	3,288	2,27	13.01.2007 00:40
78	Aylık	30	8,25	0	3,402	1,694	13.01.2007 00:50
79	Aylık	30	8,25	0	4,546	1,574	13.01.2007 01:00
80	Aylık	30	8,25	2,716	4,661	3,694	13.01.2007 01:10
81	Aylık	30	8,25	2,144	5,233	3,47	13.01.2007 01:20
82	Aylık	30	8,25	0	5,118	3,745	13.01.2007 01:30
83	Aylık	30	8,25	3,174	5,462	4,262	13.01.2007 01:40
84	Aylık	30	8,25	4,203	5,004	4,672	13.01.2007 01:50
85	Aylık	30	8,25	4,432	5,233	5,021	13.01.2007 02:00
86	Aylık	30	8,25	4,546	5,462	4,976	13.01.2007 02:10
87	Aylık	30	8,25	4,203	4,89	4,542	13.01.2007 02:20
88	Aylık	30	8,25	4,203	4,89	4,514	13.01.2007 02:30
89	Aylık	30	8,25	4,775	5,462	5,187	13.01.2007 02:40
90	Aylık	30	8,25	5,188	5,69	5,379	13.01.2007 02:50
91	Aylık	30	8,25	5,233	5,805	5,584	13.01.2007 03:00
92	Aylık	30	8,25	4,661	5,69	5,309	13.01.2007 03:10
93	Aylık	30	8,25	4,546	6,148	5,259	13.01.2007 03:20
94	Aylık	30	8,25	0	5,004	3,078	13.01.2007 03:30
95	Aylık	30	8,25	0	4,203	2,056	13.01.2007 03:40
96	Aylık	30	8,25	1,686	5,118	3,296	13.01.2007 03:50
97	Aylık	30	8,25	3,517	5,233	4,531	13.01.2007 04:00

98	Aylık	30	8,25	2,144	4,432	3,033	13.01.2007 04:10
99	Aylık	30	8,25	2,373	3,174	4,461	13.01.2007 04:20
100	Aylık	30	8,25	3,174	4,546	3,804	13.01.2007 04:30
101	Aylık	30	8,25	3,517	5,576	4,869	13.01.2007 04:40
102	Aylık	30	8,25	0	6,51	3,465	13.01.2007 04:50
103	Aylık	30	8,25	4,15	6,19	4,789	13.01.2007 05:00
104	Aylık	30	8,25	3,61	6,55	4,389	13.01.2007 05:10
105	Aylık	30	8,25	2,97	5,71	3,709	13.01.2007 05:20
106	Aylık	30	8,25	2,33	5,61	3,654	13.01.2007 05:30
107	Aylık	30	8,25	2,33	6,21	4,009	13.01.2007 05:40
108	Aylık	30	8,25	3,19	8,59	5,024	13.01.2007 05:50
109	Aylık	30	8,25	4,57	7,19	5,345	13.01.2007 06:00
110	Aylık	30	8,25	3,41	6,99	3,981	13.01.2007 06:10
111	Aylık	30	8,25	3,65	6,61	4,763	13.01.2007 06:20
112	Aylık	30	8,25	2,75	6,85	4,025	13.01.2007 06:30
113	Aylık	30	8,25	2,41	6,11	3,946	13.01.2007 06:40
114	Aylık	30	8,25	4,23	6,61	4,025	13.01.2007 06:50
115	Aylık	30	8,25	2,49	5,95	3,023	13.01.2007 07:00
116	Aylık	30	8,25	2,29	5,53	3,482	13.01.2007 07:10
117	Aylık	30	8,25	1,19	5,01	3,672	13.01.2007 07:20
118	Aylık	30	8,25	2,27	5,35	3,712	13.01.2007 07:30
119	Aylık	30	8,25	2,33	5,02	3,134	13.01.2007 07:40

120	Aylık	30	8,25	2,05	4,59	3,091	13.01.2007 07:50
121	Aylık	30	8,25	2,15	5,55	3,732	13.01.2007 08:00
122	Aylık	30	8,25	2,87	5,47	4,231	13.01.2007 08:10
123	Aylık	30	8,25	2,37	4,73	3,587	13.01.2007 08:20
124	Aylık	30	8,25	1,85	5,47	2,992	13.01.2007 08:30
125	Aylık	30	8,25	1,87	4,19	2,547	13.01.2007 08:40
126	Aylık	30	8,25	1,73	4,27	2,391	13.01.2007 08:50
127	Aylık	30	8,25	1,23	3,09	1,632	13.01.2007 09:00
128	Aylık	30	8,25	1,13	3,99	1,297	13.01.2007 09:10
129	Aylık	30	8,25	0,81	4,09	2,143	13.01.2007 09:20
130	Aylık	30	8,25	1,91	4,29	2,012	13.01.2007 09:30
131	Aylık	30	8,25	2,07	4,61	2,409	13.01.2007 09:40
132	Aylık	30	8,25	2,23	4,77	2,865	13.01.2007 09:50
133	Aylık	30	8,25	2,43	4,89	3,871	13.01.2007 10:00
134	Aylık	30	8,25	3,03	4,69	3,209	13.01.2007 10:10
135	Aylık	30	8,25	3,19	4,29	3,503	13.01.2007 10:20
136	Aylık	30	8,25	2,97	4,51	3,734	13.01.2007 10:30
137	Aylık	30	8,25	2,57	4,35	3,231	13.01.2007 10:40
138	Aylık	30	8,25	2,57	4,05	2,887	13.01.2007 10:50
139	Aylık	30	8,25	2,81	4,57	3,445	13.01.2007 11:00
140	Aylık	30	8,25	3,27	4,75	3,647	13.01.2007 11:10
141	Aylık	30	8,25	3,47	5,69	3,991	13.01.2007 11:20

142	Aylık	30	8,25	2,95	5,63	3,224	13.01.2007 11:30
143	Aylık	30	8,25	3,17	5,61	3,834	13.01.2007 11:40
144	Aylık	30	8,25	0,59	4,23	2,114	13.01.2007 11:50
145	Aylık	30	8,25	0,57	3,31	1,653	13.01.2007 12:00

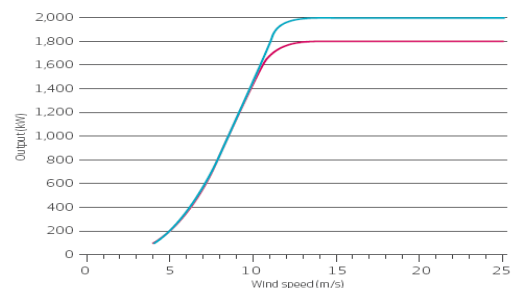
## VESTAS RÜZGAR TÜRBİNİ KATALOG DEĞERLERİ (2010)

### TECHNICAL DATA FOR V90-1.8/2.0 MW

<b>Power regulation</b>	pitch regulated with variable speed		<b>Main dimensions</b>
<b>Operating data</b>	<b>IEC IIA</b>	<b>IEC IIIA</b>	<b>Blade</b>
Rated power	1,800 kW	2,000 kW	Length 4.4 m
Cut-in wind speed	4 m/s		Max. chord 3.5 m
Rated wind speed	12 m/s		Weight 6,700 kg
Cut-out wind speed	25 m/s		<b>Nacelle</b>
Frequency	50 Hz/60 Hz		Height for transport 4 m
Operating temperature	standard range -20°C to 40°C		Height installed (including CoolerTop™) 5.4 m
	low temperature option -30°C to 40°C		Length 10.4 m
			Width 3.4 m
			Weight 70 metric tonnes
<b>Sound power</b>			<b>Hub</b>
(1.0 m above ground, hub height 80 m air density 1,225 kg/m <sup>3</sup> )			Max. diameter 3.3 m
4 m/s	94.4 dB(A)		Max. width 4 m
5 m/s	99.4 dB(A)		Length 4.2 m
6 m/s	102.5 dB(A)		Weight 1.8 metric tonnes
7 m/s	103.6 dB(A)		<b>Tower</b>
> 8 m/s	104 dB(A)		<b>80 m</b>
<b>Rotor</b>			Weight 1.48 metric tonnes
Rotor diameter	90 m		<b>95 m</b>
Swept area	6,362 m <sup>2</sup>		Weight 206 metric tonnes
Nominal revolutions	14.5 rpm		<b>105 m</b>
Operational interval	9.3 - 16.6 rpm		Weight 245 metric tonnes
Air brake	full blade feathering with 3 pitch cylinders		<b>125 m</b>
			Weight 335 metric tonnes
<b>Tower</b>			<b>Power curve V90-1.8/2.0 MW</b>
Type	tubular steel tower		Noise reduced sound power modes are available.
Hub heights	80 m, 95 m and 105 m (IEC IIA) 95 m, 105 m and 125 m (DIBt II)		
<b>Generator</b>	<b>50 Hz</b>	<b>60Hz</b>	
Type	4-pole asynchronous with variable speed	6-pole asynchronous with variable speed	
Nominal output	1,800 kW/ 2,000 kW	1,800 kW	
<b>Gearbox</b>			
Type	3-stage planetary/helical		

**Copyright Notice**  
This document is created by Vestas Wind Systems A/S and contains copyrighted material, trademarks, and other proprietary information. All rights reserved. No part of the documents may be reproduced or copied in any form or by any means such as graphic, electronic, or mechanical, including photocopying, taping, or information storage and retrieval systems without the prior written permission of Vestas Wind Systems A/S

All specifications are for informational purposes and are subject to change without notice. Vestas does not make any representations or extend any warranties, expressed or implied, as to the adequacy or accuracy of this information.  
VCS turbines are not available in the USA or Canada.



## TECHNICAL DATA FOR V90-3.0 MW

<b>Power regulation</b>	pitch regulated with variable speed
<b>Operating data</b>	
Rated power	3,000 kW
Cut-in wind speed	3.5 m/s
Rated wind speed	15 m/s
Cut-out wind speed	25 m/s
Wind Class - IEC	IEC IA (high wind) and IEC IIA (medium wind)
Operating temperature range	standard range -20°C to 40°C. low temperature option -30°C to 40°C
<b>Sound power</b>	
(10 m above ground, hub height 80 m, standard air density 1,225 kg/m <sup>3</sup> )	
4 m/s	97.9 dB(A)
5 m/s	100.9 dB(A)
6 m/s	104.2 dB(A)
7 m/s	106.1 dB(A)
8 m/s	107.0 dB(A)
9 m/s	106.9 dB(A)
<b>Rotor</b>	
Rotor diameter	90 m
Swept area	6,362 m <sup>2</sup>
Nominal revolutions	16.1 rpm
Operational interval	8.6 - 18.4 rpm
Air brake	full blade feathering with three pitch cylinders
<b>Tower</b>	
Type	tubular steel tower
Hub heights	80 m and 90 m (IEC IA) 105 m (IEC IIA)
<b>Electrical</b>	
Frequency	50 Hz/60 Hz
Generator type	4-pole asynchronous with variable speed
<b>Gearbox</b>	
Type	two planetary stages and one helical stage

### Copyright Notice

This document was prepared by Vestas Wind Systems A/S and contains copyrighted material, trademarks, and other proprietary information. All rights reserved. No part of the document may be reproduced or copied in any form or by any means such as graphic, electronic, or mechanical, including photocopying, taping, or information storage and retrieval systems without the prior written permission of Vestas Wind Systems A/S.

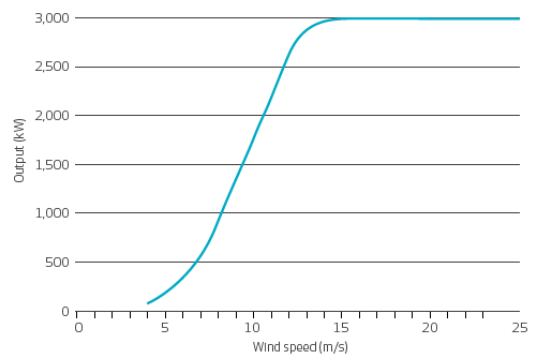
All specifications are for informational purposes and are subject to change without notice. Vestas does not make any representations or extend any warranties, expressed or implied, as to the adequacy or accuracy of this information.

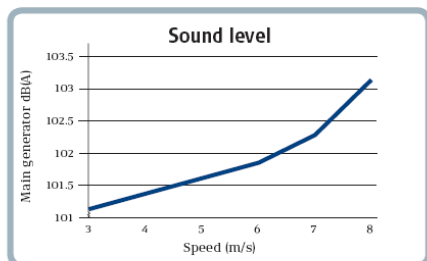
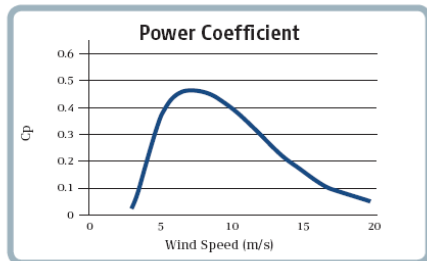
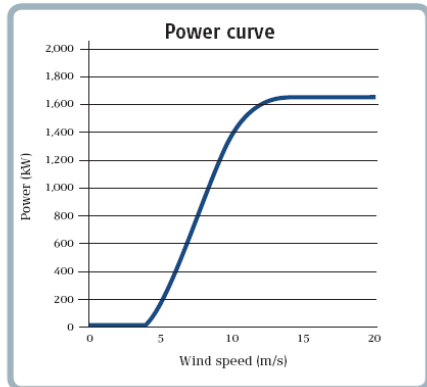
### Main dimensions

<b>Blade</b>	
Length	44 m
Max. chord	3.5 m
Weight	6,700 kg
<b>Nacelle</b>	
Height for transport	4 m
Length	9.65 m
Width	3.65 m (3.85 m installed)
Weight	70 metric tonnes
<b>Hub incl. nose cone</b>	
Diameter	3.6 m
Max. width	4.2 m
Length	4.4 m
Weight	22 metric tonnes
<b>Tower</b>	
<b>80 m (IEC IA)</b>	
Weight (50 Hz)	145 metric tonnes
Weight (60 Hz)	155 metric tonnes
<b>90 m (IEC IA)</b>	
Weight (50 Hz)	205 metric tonnes
<b>105 m (IEC IIA)</b>	
Weight (50 Hz/60 Hz)	255 metric tonnes
<b>Offshore</b>	site specific towers

### Power curve V90-3.0 MW

Noise reduced sound power modes are available.





### Rotor

Diameter: 82 m  
 Area swept: 5,281 m<sup>2</sup>  
 Nominal revolutions: 14.4 rpm  
 Number of blades: 3  
 Power regulation: Active-Stall®  
 Air brake: Full blade pitch by three separate hydraulic pitch cylinders.

### Tower

50Hz, 230V: Hub height (approx.) 78 m  
 60Hz, 110V: Hub height (approx.) 70 m, 80 m

### Operational data

Cut-in wind speed: 3.5 m/s  
 Nominal wind speed: 13 m/s  
 Cut-out wind speed (10 minutes): 20 m/s

### Generator

Type: Asynchronous water cooled  
 Nominal output: 1,650 kW  
 Operational data: 50/60 Hz 690/600V

### Gearbox

Type: Planetary/helical stages

### Control

Type: Microprocessor-based monitoring of all turbine functions with the option of remote monitoring. Output regulation and optimisation via Active-Stall®.

### Weight

Nacelle: 52 t  
 Rotor: 43 t

Towers:  
 50Hz, 230V  
 Hub height: IEC IIA  
 78 m 115 t

60Hz, 110V  
 Hub height: IEC IIA  
 70 m 105 t  
 80 m 125 t

*t = metric tonnes.*

All specifications subject to change without notice.

## APPENDIX

### 1. Kişi başı elektrik tüketiminin program içindeki fonksiyonu :

```

void kisibasi()
{
    double yazdonemi = 0;
    double kisdonemi = 0;
    if (checkBox2.Checked == true)
    {
        if (textBox1.Text == "")
        {
            MessageBox.Show(this, "Lütfen Yaz Dönemi Kişi Başı
Elektrik Tüketim Miktarını Girin");
        }
        else
        {
            if (checkedListBox3.GetItemChecked(0) == true)
            {
                yazdonemi += Convert.ToDouble(textBox1.Text);
            }
            if (checkedListBox3.GetItemChecked(1) == true)
            {
                yazdonemi += Convert.ToDouble(textBox1.Text);
            }
            if (checkedListBox3.GetItemChecked(2) == true)
            {
                yazdonemi += Convert.ToDouble(textBox1.Text);
            }
            if (checkedListBox3.GetItemChecked(3) == true)
            {
                yazdonemi += Convert.ToDouble(textBox1.Text);
            }
            if (checkedListBox3.GetItemChecked(4) == true)
            {
                yazdonemi += Convert.ToDouble(textBox1.Text);
            }
            radioButton22.Checked = true;
        }
    }
    if (checkBox1.Checked == true)
    {
        if (textBox10.Text == "")
        {
            MessageBox.Show(this, "Lütfen Yaz Dönemi Kişi Başı
Elektrik Tüketim Miktarını Girin");
        }
    }
}

```



```

    }
    else
    {
        if (checkedListBox2.GetItemChecked(0) == true)
        {
            k1sdonemi += Convert.ToDouble(textBox10.Text);
        }
        if (checkedListBox2.GetItemChecked(1) == true)
        {
            k1sdonemi += Convert.ToDouble(textBox10.Text);
        }
        if (checkedListBox2.GetItemChecked(2) == true)
        {
            k1sdonemi += Convert.ToDouble(textBox10.Text);
        }
        if (checkedListBox2.GetItemChecked(3) == true)
        {
            k1sdonemi += Convert.ToDouble(textBox10.Text);
        }
        if (checkedListBox2.GetItemChecked(4) == true)
        {
            k1sdonemi += Convert.ToDouble(textBox10.Text);
        }
        if (checkedListBox2.GetItemChecked(5) == true)
        {
            k1sdonemi += Convert.ToDouble(textBox10.Text);
        }
        if (checkedListBox2.GetItemChecked(6) == true)
        {
            k1sdonemi += Convert.ToDouble(textBox10.Text);
        }
        radioButton22.Checked = true;
    }
}
label72.Text = ((yazdonemi + k1sdonemi) * 24) /
12).ToString();
textBox20.Text = ((yazdonemi + k1sdonemi) * 24) /
12).ToString();
if ((checkBox1.Checked == true) && (checkBox2.Checked ==
true))
{
    radioButton8.Checked = true;
    radioButton22.Checked = false;
}
}

```

## 2. Hellman arayüzü rüzgar hızı verileri analiz fonksiyonu :

```

if (radioButton2.Checked == true)
{
    checkBox6.Text = textBox7.Text;
    checkBox6.Enabled = false;

    sayi = listBox1.Items.Count;

```

```

vrolc1 = new double[sayi];
sayi1 = listBox2.Items.Count;
vrolc2 = new double[sayi1];
vrist1 = new double[sayi];
vrist2 = new double[sayi1];
hist = Convert.ToInt32(checkBox5.Text);
alfa = 0.24;
if (radioButton4.Checked == true)
{
    holc = 10;
    for (olcum = 0; olcum <= sayi - 1; olcum++)
    {
        vrolc1[olcum] =
Convert.ToDouble(listBox1.Items[olcum]);

        vrist1[olcum] = vrolc1[olcum] *
(Math.Pow((hist / holc), alfa));

        leighortalama += vrolc1[olcum];
listBox9.Items.Add(vrist1[olcum]);
listBox10.Items.Add(vrolc1[olcum]);
listBox11.Items.Add(hist);
listBox12.Items.Add(holc);

    }
    esmesayisil = sayi - 1;
    leighortalamasonuc = leighortalama / esmesayisil;

    for (olcum = 0; olcum <= sayi - 1; olcum++)
    {

        rayleighesmesure[olcum] = (Math.PI / 2) *
Math.Pow((vrist1[olcum] / leighortalamasonuc), 2) *
Math.Pow(2.71828183, -(Math.PI / 4) * (Math.Pow((vrist1[olcum] /
leighortalamasonuc), 2)));
        listBox3.Items.Add(rayleighesmesure[olcum]);
        sonucu += (vrist1[olcum] *
rayleighesmesure[olcum]);
        katsayi += rayleighesmesure[olcum];
    }
    ortalamasonucu = sonucu / katsayi;

    degerlendir = MessageBox.Show(this, "Verileri
Kaydetmek Istiyor Musunuz?", "Soru Kutusu", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
    if (degerlendir == DialogResult.Yes)
    {
        kaydet();
        MessageBox.Show(this, "Kayit İşlemi
Gerçekleştirildi");
        listBox3.Items.Clear();
    }
    else
    {

```

```

        MessageBox.Show(this, "Kayıt İşlemi
Gerçekleştirilmedi.");
        listBox3.Items.Clear();
        listBox9.Items.Clear();
        listBox10.Items.Clear();
        listBox11.Items.Clear();
        listBox12.Items.Clear();
    }
}
else
{
    holc = 30;
    for (olcum = 0; olcum <= sayil - 1; olcum++)
    {
        vrolc2[olcum] =
Convert.ToDouble(listBox2.Items[olcum]);
        vrist2[olcum] = vrolc2[olcum] *
(Math.Pow((hist / holc), alfa));
        leighortalama += vrolc2[olcum];
        listBox9.Items.Add(vrist2[olcum]);
        listBox10.Items.Add(vrolc2[olcum]);
        listBox11.Items.Add(hist);
        listBox12.Items.Add(holc);
    }
    esmesayisi2 = sayil ;
    leighortalamasonuc = leighortalama / esmesayisi2;

    for (olcum = 0; olcum <= sayil - 1; olcum++)
    {

        rayleighesmesure[olcum] = (Math.PI / 2) *
Math.Pow((vrist2[olcum] / leighortalamasonuc), 2) *
Math.Pow(2.71828183, -(Math.PI / 4) * (Math.Pow((vrist2[olcum] /
leighortalamasonuc), 2)));
        listBox6.Items.Add(rayleighesmesure[olcum]);
        sonucu += (vrist2[olcum] *
rayleighesmesure[olcum]);
        katsayi += rayleighesmesure[olcum];
    }
    ortalamasonucu = sonucu / katsayi;

    degerlendir = MessageBox.Show(this, "Verileri
Kaydetmek İstiyor Musunuz?", "Soru Kutusu", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
    if (degerlendir == DialogResult.Yes)
    {
        kaydet();
        MessageBox.Show(this, "Kayıt İşlemi
Gerçekleştirildi");
        listBox6.Items.Clear();
    }
}
else
{
    MessageBox.Show(this, "Kayıt İşlemi
Gerçekleştirilmedi.");
    listBox6.Items.Clear();
    listBox9.Items.Clear();
}

```

```

        listBox10.Items.Clear();
        listBox11.Items.Clear();
        listBox12.Items.Clear();
    }
}
else
{
    checkBox6.Text = textBox7.Text;
    checkBox6.Enabled = true;

    sayi = listBox1.Items.Count;
    vrolc1 = new double[sayi];
    sayi1 = listBox2.Items.Count;
    vrolc2 = new double[sayi1];
    vrist1 = new double[sayi];
    vrist2 = new double[sayi1];
    hist = Convert.ToInt32(checkBox6.Text);
    alfa = 0.24;
    if (radioButton4.Checked == true)
    {
        holc = 10;
        for (olcum = 0; olcum <= sayi - 1; olcum++)
        {
            vrolc1[olcum] =
Convert.ToDouble(listBox1.Items[olcum]);

            vrist1[olcum] = vrolc1[olcum] *
(Math.Pow((hist / holc), alfa));

            leighortalama += vrolc1[olcum];
            listBox9.Items.Add(vrist1[olcum]);
            listBox10.Items.Add(vrolc1[olcum]);
            listBox11.Items.Add(hist);
            listBox12.Items.Add(holc);
            sonuc1 += vrist1[olcum];

        }
        esmesayisi1 = sayi ;

        ortalamasonuc1 = sonuc1 / esmesayisi1;

        degerlendir = MessageBox.Show(this, "Verileri
Kaydetmek İstiyor Musunuz?", "Soru Kutusu", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
        if (degerlendir == DialogResult.Yes)
        {
            kaydet();
            MessageBox.Show(this, "Kayıt İşlemi
Gerçekleştirildi");
        }
    }
}
else
{

```

```

        MessageBox.Show(this, "Kayıt İşlemi
Gerçekleştirilmedi.");

        listBox9.Items.Clear();
        listBox10.Items.Clear();
        listBox11.Items.Clear();
        listBox12.Items.Clear();

    }
}
else
{
    holc = 30;
    for (olcum = 0; olcum <= sayil - 1; olcum++)
    {
        vrolc2[olcum] =
Convert.ToDouble(listBox2.Items[olcum]);
        vrist2[olcum] = vrolc2[olcum] *
(Math.Pow((hist / holc), alfa));
        leighortalama += vrolc2[olcum];
        listBox9.Items.Add(vrist2[olcum]);
        listBox10.Items.Add(vrolc2[olcum]);
        listBox11.Items.Add(hist);
        listBox12.Items.Add(holc);
        sonucu2 += vrist2[olcum];
    }
    esmesayisi2 = sayil;

    ortalamasonucu2 = sonucu2 / esmesayisi2;

    degerlendir = MessageBox.Show(this, "Verileri
Kaydetmek İstiyor Musunuz?", "Soru Kutusu", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
    if (degerlendir == DialogResult.Yes)
    {
        kaydet();
        MessageBox.Show(this, "Kayıt İşlemi
Gerçekleştirildi");
    }
    else
    {
        MessageBox.Show(this, "Kayıt İşlemi
Gerçekleştirilmedi.");

        listBox9.Items.Clear();
        listBox10.Items.Clear();
        listBox11.Items.Clear();
        listBox12.Items.Clear();

    }
}
}
}
}

```

### 3. Hava yoğunluğu ölçüm fonksiyonu:

```

{
    double pl = 0;

    double pl0, g, z, rl, tl, rd, t, statik;
    pl0 = 101320;
    g = 9.81;
    z = Convert.ToDouble(maskedTextBox2.Text);
    rl = rd = 287;
    tl = t = (Convert.ToDouble(maskedTextBox1.Text) + 273);
    statik = Math.Exp((-g * z) / (rl * tl));
    pl = (pl0 * statik) - 4300;
    yogunluk = pl / (rd * t);
    if ((maskedTextBox1.Text == "") || (maskedTextBox2.Text ==
""))
    {
        MessageBox.Show(this, "Lütfen Bir Değer Girin");
        maskedTextBox1.Focus();
    }
    else
    {
        maskedTextBox3.Text = yogunluk.ToString();
        maskedTextBox5.Text = pl.ToString();
        comboBox3.Enabled = true;
    }
}

```

### 4. Rüzgar gücü hesabı kodları:

```

textBox29.Visible = true;
textBox30.Visible = true;

int i, adet;
double deger, yararliyuzde, kayipyuzde;
double toplam = 0;
if ((textBox28.Text == "") || (maskedTextBox3.Text == ""))
|| (maskedTextBox4.Text == "")
{
    MessageBox.Show(this, "Lütfen Bir Değer Giriniz");
}
else
{
    deger = Convert.ToDouble(textBox28.Text);

    DialogResult dugmesec;

    if ((deger < 0.333333333) || (deger > 0.9))
    {
        dugmesec = MessageBox.Show(this, "Lütfen 0,33333
ile 0,9 Arası Bir değer Girin", "Onaylama Kutusu",
MessageBoxButtons.OK, MessageBoxIcon.Question);
        textBox28.Focus();
        textBox28.ResetText();
    }
}

```

```

    }
    else
    {
        yuzey = Convert.ToDouble(maskedTextBox4.Text);
        adet = comboBox1.Items.Count;
        icersi = new double[adet];
        kullanilan = new double[adet];
        siraliguc = new double[adet];
        ilkhizlar = new double[adet];
        faktorguctoplam = new double[adet];
        faktorguchiz = new double[adet];
        textBox4.Text = deger.ToString();
        gucfaktoru = (0.5 * (1 + deger) * (1 -
Math.Pow(deger, 2)));
        yararliyuzde = deger * 100;
        kayipyuzde = 100 - yararliyuzde;
        label43.Visible = true;
        label44.Visible = true;
        textBox30.Text = yararliyuzde.ToString();
        textBox29.Text = kayipyuzde.ToString();
        textBox2.Text = gucfaktoru.ToString();
        for (i = 0; i <= adet - 1; i++)
        {
            ilkhizlar[i] =
Convert.ToDouble(comboBox1.Items[i]);
            icersi[i] =
(Convert.ToDouble(comboBox1.Items[i])) * deger;
            comboBox2.Items.Add(icersi[i]);
            kullanilan[i] =
((Convert.ToDouble(comboBox1.Items[i])) +
(Convert.ToDouble(comboBox2.Items[i]))) / 2;
            siraliguc[i] = (0.5 * yogunluk * yuzey *
Math.Pow(kullanilan[i], 3));
            toplam += siraliguc[i];
            ortalamatoplam = toplam / adet - 1;
            faktorguctoplam[i] = (0.5 * yogunluk * yuzey *
Math.Pow(ilkhizlar[i], 3)) * gucfaktoru;
            faktorguchiz[i] = Math.Pow((2 *
faktorguctoplam[i]) / (yogunluk * yuzey),
0.33333333333333333333333333333333);
            listBox9.Items.Add(faktorguctoplam[i]);
            listBox8.Items.Add(faktorguchiz[i]);
            listBox3.Items.Add(kullanilan[i]);
            listBox4.Items.Add(siraliguc[i]);
            textBox36.Text = ortalamatoplam.ToString();
            listBox10.Enabled = true;
            listBox10.Items.Add(ilkhizlar[i]);
        }
        button12.Visible = true;
        comboBox1.Enabled = true;
        comboBox2.Enabled = true;
        checkBox15.Enabled = true;
        label122.Show();
        label123.Show();

        label125.Show();
        label126.Show();
        label128.Show();
        label129.Show();

```

```

        label1130.Show();
        label1131.Show();
        label1132.Show();
    }
}
}

```

### 5. Türbin tipine göre rüzgar gücü hesap kodları:

```

private void turbinegorehesapla()
{
    textBox29.Visible = true;
    textBox30.Visible = true;

    int i, adet;
    double deger, yararliyuzde, kayipyuzde;
    double toplam = 0;
    if ((textBox28.Text == "") || (maskedTextBox3.Text == ""))
|| (maskedTextBox4.Text == "")
    {
        MessageBox.Show(this, "Lütfen Bir Değer Giriniz");
    }
    else
    {
        deger = Convert.ToDouble(textBox28.Text);

        DialogResult dugmesec;

        if ((deger < 0.333333333) || (deger > 0.9))
        {

            dugmesec = MessageBox.Show(this, "Lütfen 0,33333
ile 0,9 Arası Bir değer Girin", "Onaylama Kutusu",
MessageBoxButtons.OK, MessageBoxIcon.Question);
            textBox28.Focus();
            textBox28.ResetText();

        }
        else
        {
            yuzey = Convert.ToDouble(maskedTextBox4.Text);
            adet = comboBox1.Items.Count;
            icersi = new double[adet];
            kullanim = new double[adet];
            siraliguc = new double[adet];
            ilkhizlar = new double[adet];
            faktorguctoplam = new double[adet];
            faktorguchiz = new double[adet];
            textBox4.Text = deger.ToString();
            gucfaktoru = (0.5 * (1 + deger) * (1 -
Math.Pow(deger, 2)));
            yararliyuzde = deger * 100;
            kayipyuzde = 100 - yararliyuzde;
            label43.Visible = true;
            label44.Visible = true;
        }
    }
}

```



```

        textBox30.Text = yararliyuzde.ToString();
        textBox29.Text = kayipyuzde.ToString();
        textBox2.Text = gucfaktoru.ToString();
        for (i = 0; i <= adet - 1; i++)
        {
            ilkhizlar[i] =
Convert.ToDouble(comboBox1.Items[i]);
            icersi[i] =
(Convert.ToDouble(comboBox1.Items[i])) * deger;
            comboBox2.Items.Add(icersi[i]);
            kullanim[i] =
((Convert.ToDouble(comboBox1.Items[i])) +
(Convert.ToDouble(comboBox2.Items[i]))) / 2;
            siraliguc[i] = (0.5 * yogunluk * yuzey *
Math.Pow(kullanim[i], 3));
            toplam += siraliguc[i];
            ortalamatoplam = toplam / adet - 1;
            faktorguctoplam[i] = (0.5 * yogunluk * yuzey *
Math.Pow(ilkhizlar[i], 3)) * gucfaktoru;
            faktorguchiz[i] = Math.Pow((2 *
faktorguctoplam[i]) / (yogunluk * yuzey),
0.33333333333333333333333333333333);

listBox9.Items.Add(Math.Round((faktorguctoplam[i]), 3));
listBox8.Items.Add(Math.Round
((faktorguchiz[i]), 3));
listBox3.Items.Add(Math.Round
((kullanim[i]), 3));
listBox4.Items.Add(Math.Round
((siraliguc[i]), 3));
        textBox36.Text = ortalamatoplam.ToString();
        listBox10.Enabled = true;

listBox10.Items.Add(Math.Round((ilkhizlar[i]), 3));
        }
        button12.Visible = true;
        comboBox1.Enabled = true;
        comboBox2.Enabled = true;
        checkBox15.Enabled = true;
        label122.Show();
        label123.Show();

        label125.Show();
        label126.Show();
        label128.Show();
        label129.Show();
        label130.Show();
        label131.Show();
        label132.Show();
    }
}
}

```

## 6. Kayıp hesabı kodları:

```
private void kayıp_Click(object sender, EventArgs e)
```

```

{
R = Convert.ToDouble(textBox31.Text);
nd = Convert.ToDouble(textBox32.Text);
Vr = Convert.ToDouble(textBox33.Text);
Vc = R * nd * Math.PI;
Vlamda = (Vc / (Vr * 30));
textBox11.Text = Math.Round(Vlamda, 3).ToString();
Ca = Convert.ToDouble(textBox14.Text);
Cv = Convert.ToDouble(textBox15.Text);
epsilon = Ca / Cv;
profilkeybi = (1 - (Vlamda / epsilon));
label18.Text = Math.Round(profilkeybi, 3).ToString();
textBox12.Text = Math.Round(Vc, 3).ToString();
textBox13.Text = Math.Round(epsilon, 3).ToString();
z = Convert.ToInt32(textBox16.Text);
uckaybi = (1 - (1.84 / (z * Vlamda)));
label19.Text = Math.Round(uckaybi, 3).ToString();
if (Vlamda >= 0 && Vlamda < 0.5)
{
    girdapkeybi = 0.238;
}
else if (Vlamda >= 5 && Vlamda < 1)
{
    girdapkeybi = 0.400;
}
else if (Vlamda >= 1 && Vlamda < 1.5)
{
    girdapkeybi = 0.475;
}
else if (Vlamda >= 1.5 && Vlamda < 2)
{
    girdapkeybi = 0.515;
}
else if (Vlamda >= 2 && Vlamda < 2.5)
{
    girdapkeybi = 0.531;
}
else if (Vlamda >= 2.5 && Vlamda < 3)
{
    girdapkeybi = 0.537;
}
else if (Vlamda >= 3 && Vlamda < 3.5)
{
    girdapkeybi = 0.538;
}
else if (Vlamda >= 3.5 && Vlamda < 4)
{
    girdapkeybi = 0.541;
}
else if (Vlamda >= 4 && Vlamda < 4.5)
{
    girdapkeybi = 0.544;
}
else if (Vlamda >= 4.5 && Vlamda < 5)
{
    girdapkeybi = 0.547;
}
else if (Vlamda >= 5 && Vlamda < 5.5)
{
    girdapkeybi = 0.550;
}
}

```

```

}
else if (Vlamda >= 5.5 && Vlamda < 6)
{
    girdapkeybi = 0.553;
}
else if (Vlamda >= 6 && Vlamda < 6.5)
{
    girdapkeybi = 0.556;
}
else if (Vlamda >= 6.5 && Vlamda < 7)
{
    girdapkeybi = 0.559;
}
else if (Vlamda >= 7 && Vlamda < 7.5)
{
    girdapkeybi = 0.562;
}
else if (Vlamda >= 7.5 && Vlamda < 8)
{
    girdapkeybi = 0.565;
}
else if (Vlamda >= 8 && Vlamda < 8.5)
{
    girdapkeybi = 0.568;
}
else if (Vlamda >= 8.5 && Vlamda < 9)
{
    girdapkeybi = 0.570;
}
else if (Vlamda >= 9 && Vlamda < 9.5)
{
    girdapkeybi = 0.572;
}
else if (Vlamda >= 9.5 && Vlamda < 10)
{
    girdapkeybi = 0.576;
}
else
{
    girdapkeybi = 0.6;
}
label20.Text = girdapkeybi.ToString();
button16.Enabled = true;

```

### ***7. Rayleigh dağılım fonksiyonu kodları:***

```

void rayleigh()
{
    listBox11.Enabled = true;
    listBox12.Enabled = true;
    listBox13.Enabled = true;
    button18.Enabled = true;
    double[] helmann;
    double[] essene;
    int satir, i;
    double ortalamadeger, katsayi;
    ortalamadeger = Convert.ToDouble(textBox33.Text);
    satir = listBox10.Items.Count;

```

```

        helmann = new double[satir];
        essene = new double[satir];
        for (i = 0; i <= satir - 1; i++)
        {
            helmann[i] = Convert.ToDouble(listBox10.Items[i]);
            katsayi = -(Math.PI) / 4 * Math.Pow((helmann[i] /
ortalamadeger), 2);
            essene[i] = ((Math.PI) / 2) * (helmann[i] /
Math.Pow(ortalamadeger, 2)) * (Math.Pow(2.71828, katsayi));
            listBox11.Items.Add(Math.Round(essene[i], 3));

            listBox12.Items.Add(Math.Round((essene[i] * 100), 2));
            listBox13.Items.Add(Math.Round((essene[i] * 8766),
2));
        }
    }
}

```

### 8. Türbin güç değerleri ve optimum türbin sayısı hesabı kodları:

```

private void listBox1_turbinanalizi_click(object sender, EventArgs e)
{
    double karsilastir = 0;
    double enerji = 0;
    double enerjigunluk = 0;
    double turbinsayisi = 0;
    double turbingrubuenerji = 0;
    double[] vestasruzhiz, vestasguc;
    int vestassatir, ves;
    vestassatir = listBox14.Items.Count;
    vestasruzhiz = new double[vestassatir];
    vestasguc = new double[vestassatir];
    label178.Show();
    label179.Show();
    label183.Show();

    groupBox19.Visible = true;
    if (textBox33.Text == "")
    {
        MessageBox.Show(this, "Daha Ortalama Rüzgar Hızı
Değeri Girilmemiş");
    }
    else
    {
        karsilastir = Convert.ToDouble(textBox33.Text);
    }

    if (listBox1.SelectedItem == "V52-850KW".ToString())
    {
        pictureBox2.ImageLocation =
("C:\\Users\\Prozac\\Desktop\\Yeni Klasör\\v52.jpg");
        pictureBox1.ImageLocation =
("C:\\Users\\Prozac\\Desktop\\Res\\V52.jpg");
        groupBox20.Enabled = true;
        label90.Text = "V52-850KW".ToString();
        radioButton19.Enabled = true;
        radioButton20.Enabled = true;
    }
}

```

```

radioButton21.Enabled = true;
radioButton19.Visible = true;
radioButton20.Visible = true;
radioButton21.Visible = true;
radioButton14.Text = "44";
radioButton18.Text = "48";
radioButton19.Text = "55";
radioButton20.Text = "65";
radioButton21.Text = "74";
label87.Text = "3 kanatlı";
label74.Text = "5m/s 100kw";
label75.Text = "10m/s 600kw";
label76.Text = "16m/s 850kw";
label78.Text = "52";
label80.Text = "2,124";
label82.Text = "26";
label84.Text = "4m/s";
label86.Text = "16m/s";
label113.Text = "850";
listBox15.Items.Clear();
textBox19.ResetText();
for (ves = 0; ves <= vestassatir - 1; ves++)
{
    vestasruzhiz[ves] =
Convert.ToDouble(listBox14.Items[ves]);
    if (vestasruzhiz[ves] == 4)
    {
        vestasguc[ves] = 50;
    }
    else if ((vestasruzhiz[ves] > 4) &&
(vestasruzhiz[ves] <= 5))
    {
        vestasguc[ves] = 80;
    }
    else if ((vestasruzhiz[ves] > 5) &&
(vestasruzhiz[ves] <= 6))
    {
        vestasguc[ves] = 100;
    }
    else if ((vestasruzhiz[ves] > 6) &&
(vestasruzhiz[ves] <= 7))
    {
        vestasguc[ves] = 150;
    }
    else if ((vestasruzhiz[ves] > 7) &&
(vestasruzhiz[ves] <= 8))
    {
        vestasguc[ves] = 200;
    }
    else if ((vestasruzhiz[ves] > 8) &&
(vestasruzhiz[ves] <= 9))
    {
        vestasguc[ves] = 300;
    }
    else if ((vestasruzhiz[ves] > 9) &&
(vestasruzhiz[ves] < 9.5))
    {
        vestasguc[ves] = 350;
    }
}

```

```

        else if (vestasruzhiz[ves] == 9.5)
        {
            vestasguc[ves] = 400;
        }
        else if ((vestasruzhiz[ves] > 9.5) &&
(vestasruzhiz[ves] <= 10))
        {
            vestasguc[ves] = 450;
        }
        else if ((vestasruzhiz[ves] > 10) &&
(vestasruzhiz[ves] <= 10.5))
        {
            vestasguc[ves] = 550;
        }
        else if ((vestasruzhiz[ves] > 10.5) &&
(vestasruzhiz[ves] <= 11))
        {
            vestasguc[ves] = 600;
        }
        else if ((vestasruzhiz[ves] > 11) &&
(vestasruzhiz[ves] <= 11.5))
        {
            vestasguc[ves] = 650;
        }
        else if ((vestasruzhiz[ves] > 11.5) &&
(vestasruzhiz[ves] <= 12))
        {
            vestasguc[ves] = 700;
        }
        else if ((vestasruzhiz[ves] > 12) &&
(vestasruzhiz[ves] <= 12.5))
        {
            vestasguc[ves] = 750;
        }
        else if ((vestasruzhiz[ves] > 12.5) &&
(vestasruzhiz[ves] <= 13))
        {
            vestasguc[ves] = 775;
        }
        else if ((vestasruzhiz[ves] > 13) &&
(vestasruzhiz[ves] <= 13.5))
        {
            vestasguc[ves] = 800;
        }
        else if ((vestasruzhiz[ves] > 13.5) &&
(vestasruzhiz[ves] <= 14))
        {
            vestasguc[ves] = 825;
        }
        else if ((vestasruzhiz[ves] > 14) &&
(vestasruzhiz[ves] < 15))
        {
            vestasguc[ves] = 840;
        }
        else if ((vestasruzhiz[ves] > 15) &&
(vestasruzhiz[ves] <= 16))
        {
            vestasguc[ves] = 850;
        }
        else

```

```

{
    vestasguc[ves] = 850;
}
listBox15.Items.Add(vestasguc[ves]);
enerji += (vestasguc[ves] / 6);
enerjigunluk = (enerji * 144) / ves;
textBox22.Text = enerjigunluk.ToString();
if (karsilastir == 4)
{
    vestasguc[ves] = 50;
}
else if ((karsilastir > 4) && (karsilastir <= 5))
{
    vestasguc[ves] = 80;
}
else if ((karsilastir > 5) && (karsilastir <= 6))
{
    vestasguc[ves] = 100;
}
else if ((karsilastir > 6) && (karsilastir <= 7))
{
    vestasguc[ves] = 150;
}
else if ((karsilastir > 7) && (karsilastir <= 8))
{
    vestasguc[ves] = 200;
}
else if ((karsilastir > 8) && (karsilastir <= 9))
{
    vestasguc[ves] = 300;
}
else if ((karsilastir > 9) && (karsilastir < 9.5))
{
    vestasguc[ves] = 350;
}
else if (karsilastir == 9.5)
{
    karsilastir = 400;
}
else if ((karsilastir > 9.5) && (karsilastir <=
10))
{
    vestasguc[ves] = 450;
}
else if ((karsilastir > 10) && (karsilastir <=
10.5))
{
    vestasguc[ves] = 550;
}
else if ((karsilastir > 10.5) && (karsilastir <=
11))
{
    vestasguc[ves] = 600;
}
else if ((karsilastir > 11) && (karsilastir <=
11.5))
{
    vestasguc[ves] = 650;
}
}

```

```

else if ((karsilastir > 11.5) && (karsilastir <=
12))
{
    vestasguc[ves] = 700;
}
else if ((karsilastir > 12) && (karsilastir <=
12.5))
{
    vestasguc[ves] = 750;
}
else if ((karsilastir > 12.5) && (karsilastir <=
13))
{
    vestasguc[ves] = 775;
}
else if ((karsilastir > 13) && (karsilastir <=
13.5))
{
    vestasguc[ves] = 800;
}
else if ((karsilastir > 13.5) && (karsilastir <=
14))
{
    vestasguc[ves] = 825;
}
else if ((karsilastir > 14) && (karsilastir < 15))
{
    vestasguc[ves] = 840;
}
else if ((karsilastir > 15) && (karsilastir <=
16))
{
    vestasguc[ves] = 850;
}
else
{
    vestasguc[ves] = 850;
}
label93.Text = karsilastir.ToString();
label94.Text = vestasguc[ves].ToString();
if (label72.Text == "")
{
    MessageBox.Show(this, "Lütfen Bir Günlük
Ortalama Elektrik Enerjisi İhtiyacını Hesaplayın");
}
else
{
    if (label72.Text == " ")
    {
        MessageBox.Show(this, "Lütfen Kişi Başına
Düşen Elektrik Enerjisi İhtiyacını Hesaplayın");
    }
    else
    {
        turbinsayisi =
Math.Floor((Convert.ToDouble(label72.Text) / (vestasguc[ves] * 24)));
    }
}

```



```

        if (turbinsayisi == 0)
        {
            textBox19.Text = "1";
            turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
            textBox23.Text =
turbingrubuenerji.ToString();
            label100.Text = (turbingrubuenerji *
365).ToString();
        }
        else
        {
            textBox19.Text = turbinsayisi.ToString();
            turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
            textBox23.Text =
turbingrubuenerji.ToString();
            label100.Text = (turbingrubuenerji *
365).ToString();
        }
    }
}
else if (listBox1.SelectedItem == "V80-2MW".ToString())
{
    pictureBox2.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Yeni Klasör\\v80.jpg";
    pictureBox1.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Res\\V802MW.jpg";
    groupBox20.Enabled = true;
    label90.Text = "V80-2MW".ToString();
    radioButton19.Enabled = false;
    radioButton20.Enabled = false;
    radioButton21.Enabled = false;
    radioButton19.Visible = false;
    radioButton20.Visible = false;
    radioButton21.Visible = false;
    radioButton14.Text = "67";
    radioButton18.Text = "80";
    label87.Text = "3 kanatlı";
    label74.Text = "5m/s 100kw";
    label75.Text = "10m/s 1200kw";
    label76.Text = "15m/s 2000kw";
    label78.Text = "80";
    label80.Text = "5,027";
    label82.Text = "16,7";
    label84.Text = "4m/s";
    label86.Text = "16m/s";
    label113.Text = "2000";
    listBox15.Items.Clear();
    textBox19.ResetText();
    for (ves = 0; ves <= vestassatir - 1; ves++)
    {
        vestasruzhiz[ves] =
Convert.ToDouble(listBox14.Items[ves]);
        if (vestasruzhiz[ves] == 4)
        {
            vestasguc[ves] = 100;

```

```

    }
    else if ((vestasruzhiz[ves] > 4) &&
(vestasruzhiz[ves] <= 5))
    {
        vestasguc[ves] = 150;
    }
    else if ((vestasruzhiz[ves] > 5) &&
(vestasruzhiz[ves] <= 6))
    {
        vestasguc[ves] = 200;
    }
    else if ((vestasruzhiz[ves] > 6) &&
(vestasruzhiz[ves] <= 7))
    {
        vestasguc[ves] = 350;
    }
    else if ((vestasruzhiz[ves] > 7) &&
(vestasruzhiz[ves] <= 7.5))
    {
        vestasguc[ves] = 400;
    }
    else if ((vestasruzhiz[ves] > 7.5) &&
(vestasruzhiz[ves] < 8))
    {
        vestasguc[ves] = 500;
    }
    else if ((vestasruzhiz[ves] >= 8) &&
(vestasruzhiz[ves] < 8.5))
    {
        vestasguc[ves] = 600;
    }

    else if ((vestasruzhiz[ves] >= 8.5) &&
(vestasruzhiz[ves] < 9))
    {
        vestasguc[ves] = 700;
    }
    else if ((vestasruzhiz[ves] >= 9) &&
(vestasruzhiz[ves] < 10))
    {
        vestasguc[ves] = 1000;
    }
    else if ((vestasruzhiz[ves] >= 10) &&
(vestasruzhiz[ves] < 11))
    {
        vestasguc[ves] = 1300;
    }
    else if ((vestasruzhiz[ves] > 11) &&
(vestasruzhiz[ves] <= 11.5))
    {
        vestasguc[ves] = 1600;
    }
    else if ((vestasruzhiz[ves] > 11.5) &&
(vestasruzhiz[ves] < 12))
    {
        vestasguc[ves] = 1800;
    }
    else if ((vestasruzhiz[ves] >= 12) &&
(vestasruzhiz[ves] < 13))
    {

```

```

        vestasguc[ves] = 1900;
    }
    else if ((vestasruzhiz[ves] >= 13) &&
(vestasruzhiz[ves] < 14))
    {
        vestasguc[ves] = 1950;
    }
    else if ((vestasruzhiz[ves] >= 14) &&
(vestasruzhiz[ves] <= 15))
    {
        vestasguc[ves] = 1975;
    }

    else
    {
        vestasguc[ves] = 2000;
    }
    listBox15.Items.Add(vestasguc[ves]);
    enerji += (vestasguc[ves] / 6);
    enerjigunluk = (enerji * 144) / ves;
    textBox22.Text = enerjigunluk.ToString();
    if (karsilastir == 4)
    {
        vestasguc[ves] = 100;
    }
    else if ((karsilastir > 4) && (karsilastir <= 5))
    {
        vestasguc[ves] = 150;
    }
    else if ((karsilastir > 5) && (karsilastir <= 6))
    {
        vestasguc[ves] = 200;
    }
    else if ((karsilastir > 6) && (karsilastir <= 7))
    {
        vestasguc[ves] = 350;
    }
    else if ((karsilastir > 7) && (karsilastir <=
7.5))
    {
        vestasguc[ves] = 400;
    }
    else if ((karsilastir > 7.5) && (karsilastir < 8))
    {
        vestasguc[ves] = 500;
    }
    else if ((karsilastir >= 8) && (karsilastir <
8.5))
    {
        vestasguc[ves] = 600;
    }

    else if ((karsilastir >= 8.5) && (karsilastir <
9))
    {
        vestasguc[ves] = 700;
    }
    else if ((karsilastir >= 9) && (karsilastir < 10))
    {
        vestasguc[ves] = 1000;
    }

```

```

    }
    else if ((karsilastir >= 10) && (karsilastir <
11))
    {
        vestasguc[ves] = 1300;
    }
    else if ((karsilastir > 11) && (karsilastir <=
11.5))
    {
        vestasguc[ves] = 1600;
    }
    else if ((karsilastir > 11.5) && (karsilastir <
12))
    {
        vestasguc[ves] = 1800;
    }
    else if ((karsilastir >= 12) && (karsilastir <
13))
    {
        vestasguc[ves] = 1900;
    }
    else if ((karsilastir >= 13) && (karsilastir <
14))
    {
        vestasguc[ves] = 1950;
    }
    else if ((karsilastir >= 14) && (karsilastir <=
15))
    {
        vestasguc[ves] = 1975;
    }

    else
    {
        vestasguc[ves] = 2000;
    }
    label93.Text = karsilastir.ToString();
    label94.Text = vestasguc[ves].ToString();
    if (label72.Text == "")
    {
        MessageBox.Show(this, "Lütfen Bir Günlük
Ortalama Elektrik Enerjisi İhtiyacını Hesaplayın");
    }
    else
    {
        turbinsayisi =
Math.Floor((Convert.ToDouble(label72.Text) / (vestasguc[ves] * 24)));

        if (turbinsayisi == 0)
        {
            textBox19.Text = "1";
            turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
            textBox23.Text =
turbingrubuenerji.ToString();
            label100.Text = (turbingrubuenerji *
365).ToString();
        }
        else
        {

```

```

        textBox19.Text = turbinsayisi.ToString();
        turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
        textBox23.Text =
turbingrubuenerji.ToString();
        label100.Text = (turbingrubuenerji *
365).ToString();
    }
}
}
else if (listBox1.SelectedItem == "V82-1,65MW".ToString())
{
    pictureBox2.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Yeni Klasör\\v82.jpg";
    pictureBox1.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Res\\V82.jpg";
    groupBox20.Enabled = true;
    label90.Text = "V80-1,65MW".ToString();
    radioButton19.Enabled = false;
    radioButton20.Enabled = false;
    radioButton21.Enabled = false;
    radioButton19.Visible = false;
    radioButton20.Visible = false;
    radioButton21.Visible = false;
    radioButton14.Text = "78";
    radioButton18.Text = "80";
    label87.Text = "3 kanatlı";
    label74.Text = "5m/s 200kw";
    label75.Text = "10m/s 1300kw";
    label76.Text = "15m/s 1650kw";
    label78.Text = "82";
    label80.Text = "5,281";
    label82.Text = "14,4";
    label84.Text = "3,5m/s";
    label86.Text = "13m/s";
    label113.Text = "1650";
    listBox15.Items.Clear();
    textBox19.ResetText();
    for (ves = 0; ves <= vestassatir - 1; ves++)
    {
        vestasruzhiz[ves] =
Convert.ToDouble(listBox14.Items[ves]);
        if ((vestasruzhiz[ves] > 4) && (vestasruzhiz[ves]
<= 4.5))
        {
            vestasguc[ves] = 100;
        }
        else if ((vestasruzhiz[ves] > 4.5) &&
(vestasruzhiz[ves] <= 5))
        {
            vestasguc[ves] = 200;
        }
        else if ((vestasruzhiz[ves] > 5) &&
(vestasruzhiz[ves] <= 6))
        {
            vestasguc[ves] = 300;
        }
        else if ((vestasruzhiz[ves] > 6) &&
(vestasruzhiz[ves] <= 7))

```

```

        {
            vestasguc[ves] = 350;
        }
        else if ((vestasruzhiz[ves] > 7) &&
(vestasruzhiz[ves] <= 7.5))
        {
            vestasguc[ves] = 400;
        }
        else if ((vestasruzhiz[ves] > 7.5) &&
(vestasruzhiz[ves] < 8))
        {
            vestasguc[ves] = 500;
        }
        else if ((vestasruzhiz[ves] >= 8) &&
(vestasruzhiz[ves] < 8.5))
        {
            vestasguc[ves] = 600;
        }

        else if ((vestasruzhiz[ves] >= 8.5) &&
(vestasruzhiz[ves] < 9))
        {
            vestasguc[ves] = 700;
        }
        else if ((vestasruzhiz[ves] >= 9) &&
(vestasruzhiz[ves] < 10))
        {
            vestasguc[ves] = 1000;
        }
        else if ((vestasruzhiz[ves] >= 10) &&
(vestasruzhiz[ves] < 11))
        {
            vestasguc[ves] = 1300;
        }
        else if ((vestasruzhiz[ves] > 11) &&
(vestasruzhiz[ves] <= 11.5))
        {
            vestasguc[ves] = 1600;
        }
        else if ((vestasruzhiz[ves] > 11.5) &&
(vestasruzhiz[ves] < 12))
        {
            vestasguc[ves] = 1800;
        }
        else if ((vestasruzhiz[ves] >= 12) &&
(vestasruzhiz[ves] < 13))
        {
            vestasguc[ves] = 1900;
        }
        else if ((vestasruzhiz[ves] >= 13) &&
(vestasruzhiz[ves] < 14))
        {
            vestasguc[ves] = 1950;
        }
        else if ((vestasruzhiz[ves] >= 14) &&
(vestasruzhiz[ves] <= 15))
        {
            vestasguc[ves] = 1975;
        }

```

```

else
{
    vestasguc[ves] = 2000;
}
listBox15.Items.Add(vestasguc[ves]);
enerji += (vestasguc[ves] / 6);
enerjigunluk = (enerji * 144) / ves;
textBox22.Text = enerjigunluk.ToString();
if ((karsilastir > 4) && (karsilastir <= 4.5))
{
    vestasguc[ves] = 100;
}
5)) else if ((karsilastir > 4.5) && (karsilastir <=
{
    vestasguc[ves] = 200;
}
else if ((karsilastir > 5) && (karsilastir <= 6))
{
    vestasguc[ves] = 300;
}
else if ((karsilastir > 6) && (karsilastir <= 7))
{
    vestasguc[ves] = 350;
}
7.5)) else if ((karsilastir > 7) && (karsilastir <=
{
    vestasguc[ves] = 400;
}
else if ((karsilastir > 7.5) && (karsilastir < 8))
{
    vestasguc[ves] = 500;
}
8.5)) else if ((karsilastir >= 8) && (karsilastir <
{
    vestasguc[ves] = 600;
}
9)) else if ((karsilastir >= 8.5) && (karsilastir <
{
    vestasguc[ves] = 700;
}
else if ((karsilastir >= 9) && (karsilastir < 10))
{
    vestasguc[ves] = 1000;
}
11)) else if ((karsilastir >= 10) && (karsilastir <
{
    vestasguc[ves] = 1300;
}
11.5)) else if ((karsilastir > 11) && (karsilastir <=
{
    vestasguc[ves] = 1600;
}

```

```

else if ((karsilastir > 11.5) && (karsilastir <
12))
{
    vestasguc[ves] = 1800;
}
else if ((karsilastir >= 12) && (karsilastir <
13))
{
    vestasguc[ves] = 1900;
}
else if ((karsilastir >= 13) && (karsilastir <
14))
{
    vestasguc[ves] = 1950;
}
else if ((karsilastir >= 14) && (karsilastir <=
15))
{
    vestasguc[ves] = 1975;
}

else
{
    vestasguc[ves] = 2000;
}
label93.Text = karsilastir.ToString();
label94.Text = vestasguc[ves].ToString();
if (label72.Text == "")
{
    MessageBox.Show(this, "Lütfen Bir Günlük
Ortalama Elektrik Enerjisi İhtiyacını Hesaplayın");
}
else
{
    turbinsayisi =
Math.Floor((Convert.ToDouble(label72.Text) / (vestasguc[ves] * 24)));

    if (turbinsayisi == 0)
    {
        textBox19.Text = "1";
        turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
        textBox23.Text =
turbingrubuenerji.ToString();
        label100.Text = (turbingrubuenerji *
365).ToString();
    }
    else
    {
        textBox19.Text = turbinsayisi.ToString();
        turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
        textBox23.Text =
turbingrubuenerji.ToString();
        label100.Text = (turbingrubuenerji *
365).ToString();
    }
}
}
}

```



```

else if (listBox1.SelectedItem == "V90-1,8MW".ToString())
{
    pictureBox2.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Yeni Klasör\\v90.jpg";
    pictureBox1.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Res\\V90_1,8MW.jpg";
    groupBox20.Enabled = true;
    label90.Text = "V90-1,8MW".ToString();
    radioButton19.Enabled = false;
    radioButton20.Enabled = false;
    radioButton21.Enabled = false;

    radioButton19.Visible = false;
    radioButton20.Visible = false;
    radioButton21.Visible = false;
    radioButton14.Text = "90";
    radioButton18.Text = "105";
    label87.Text = "3 kanatlı";
    label74.Text = "5m/s 200kw";
    label75.Text = "10m/s 1300kw";
    label76.Text = "15m/s 1800kw";
    label78.Text = "90";
    label80.Text = "6,362";
    label82.Text = "14,5";
    label84.Text = "4m/s";
    label86.Text = "12m/s";
    label113.Text = "1800";
    listBox15.Items.Clear();
    textBox19.ResetText();
    for (ves = 0; ves <= vestassatir - 1; ves++)
    {
        vestasruzhiz[ves] =
Convert.ToDouble(listBox14.Items[ves]);
        if ((vestasruzhiz[ves] > 4) && (vestasruzhiz[ves]
<= 4.5))
        {
            vestasguc[ves] = 100;
        }
        else if ((vestasruzhiz[ves] > 4.5) &&
(vestasruzhiz[ves] <= 5))
        {
            vestasguc[ves] = 200;
        }
        else if ((vestasruzhiz[ves] > 5) &&
(vestasruzhiz[ves] <= 6))
        {
            vestasguc[ves] = 375;
        }
        else if ((vestasruzhiz[ves] > 6) &&
(vestasruzhiz[ves] <= 7))
        {
            vestasguc[ves] = 575;
        }
        else if ((vestasruzhiz[ves] > 7) &&
(vestasruzhiz[ves] <= 8))
        {
            vestasguc[ves] = 850;
        }
        else if ((vestasruzhiz[ves] > 8) &&
(vestasruzhiz[ves] < 8.5))

```

```

        {
            vestasguc[ves] = 1150;
        }
        else if ((vestasruzhiz[ves] >= 8.5) &&
(vestasruzhiz[ves] < 9.5))
        {
            vestasguc[ves] = 1400;
        }

        else if ((vestasruzhiz[ves] >= 9.5) &&
(vestasruzhiz[ves] < 10))
        {
            vestasguc[ves] = 1525;
        }
        else if ((vestasruzhiz[ves] >= 10) &&
(vestasruzhiz[ves] < 10.5))
        {
            vestasguc[ves] = 1625;
        }
        else if ((vestasruzhiz[ves] >= 10.5) &&
(vestasruzhiz[ves] < 11))
        {
            vestasguc[ves] = 1700;
        }
        else if ((vestasruzhiz[ves] > 11) &&
(vestasruzhiz[ves] <= 12))
        {
            vestasguc[ves] = 1775;
        }
        else if ((vestasruzhiz[ves] > 12) &&
(vestasruzhiz[ves] < 13))
        {
            vestasguc[ves] = 1785;
        }

        else
        {
            vestasguc[ves] = 1800;
        }
        listBox15.Items.Add(vestasguc[ves]);
        enerji += (vestasguc[ves] / 6);
        enerjigunluk = (enerji * 144) / ves;
        textBox22.Text = enerjigunluk.ToString();
        if ((karsilastir > 4) && (karsilastir <= 4.5))
        {
            vestasguc[ves] = 100;
        }
        else if ((karsilastir > 4.5) && (karsilastir <=
5))
        {
            vestasguc[ves] = 200;
        }
        else if ((karsilastir > 5) && (karsilastir <= 6))
        {
            vestasguc[ves] = 375;
        }
        else if ((karsilastir > 6) && (karsilastir <= 7))
        {
            vestasguc[ves] = 575;
        }
    }
}

```

```

else if ((karsilastir > 7) && (karsilastir <= 8))
{
    vestasguc[ves] = 850;
}
else if ((karsilastir > 8) && (karsilastir < 8.5))
{
    vestasguc[ves] = 1150;
}
else if ((karsilastir >= 8.5) && (karsilastir <
9.5))
{
    vestasguc[ves] = 1400;
}

else if ((karsilastir >= 9.5) && (karsilastir <
10))
{
    vestasguc[ves] = 1525;
}
else if ((karsilastir >= 10) && (karsilastir <
10.5))
{
    vestasguc[ves] = 1625;
}
else if ((karsilastir >= 10.5) && (karsilastir <
11))
{
    vestasguc[ves] = 1700;
}
else if ((karsilastir > 11) && (karsilastir <=
12))
{
    vestasguc[ves] = 1775;
}
else if ((karsilastir > 12) && (karsilastir < 13))
{
    vestasguc[ves] = 1785;
}

else
{
    vestasguc[ves] = 1800;
}
label93.Text = karsilastir.ToString();
label94.Text = vestasguc[ves].ToString();
if (label72.Text == "")
{
    MessageBox.Show(this, "Lütfen Bir Günlük
Ortalama Elektrik Enerjisi İhtiyacını Hesaplayın");
}
else
{
    turbinsayisi =
Math.Floor((Convert.ToDouble(label72.Text) / (vestasguc[ves] * 24)));

    if (turbinsayisi == 0)
    {
        textBox19.Text = "1";
        turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);

```

```

        textBox23.Text =
turbingrubuenerji.ToString();
        label100.Text = (turbingrubuenerji *
365).ToString();
    }
    else
    {
        textBox19.Text = turbinsayisi.ToString();
        turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
        textBox23.Text =
turbingrubuenerji.ToString();
        label100.Text = (turbingrubuenerji *
365).ToString();
    }
}
}

else if (listBox1.SelectedItem == "V90-3MW".ToString())
{
    pictureBox2.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Yeni Klasör\\v90_3.jpg";
    pictureBox1.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Res\\V90_3.jpg";
    groupBox20.Enabled = true;
    label90.Text = "V90-3MW".ToString();
    radioButton19.Enabled = false;
    radioButton20.Enabled = false;
    radioButton21.Enabled = false;

    radioButton19.Visible = false;
    radioButton20.Visible = false;
    radioButton21.Visible = false;
    radioButton14.Text = "90";
    radioButton18.Text = "105";
    label87.Text = "3 kanatlı";
    label74.Text = "5m/s 150kw";
    label75.Text = "10m/s 1700kw";
    label76.Text = "15m/s 3000kw";
    label78.Text = "90";
    label80.Text = "6,362";
    label82.Text = "16,1";
    label84.Text = "3,5m/s";
    label86.Text = "15m/s";
    label113.Text = "3000";
    listBox15.Items.Clear();
    textBox19.ResetText();
    for (ves = 0; ves <= vestassatir - 1; ves++)
    {
        vestasruzhiz[ves] =
Convert.ToDouble(listBox14.Items[ves]);
        if ((vestasruzhiz[ves] > 4) && (vestasruzhiz[ves]
<= 4.5))
        {
            vestasguc[ves] = 100;
        }
        else if ((vestasruzhiz[ves] > 4.5) &&
(vestasruzhiz[ves] <= 5))

```

```

        {
            vestasguc[ves] = 175;
        }
        else if ((vestasruzhiz[ves] > 5) &&
(vestasruzhiz[ves] <= 6))
        {
            vestasguc[ves] = 225;
        }
        else if ((vestasruzhiz[ves] > 6) &&
(vestasruzhiz[ves] <= 7))
        {
            vestasguc[ves] = 575;
        }
        else if ((vestasruzhiz[ves] > 7) &&
(vestasruzhiz[ves] <= 8))
        {
            vestasguc[ves] = 850;
        }
        else if ((vestasruzhiz[ves] > 8) &&
(vestasruzhiz[ves] < 8.5))
        {
            vestasguc[ves] = 1000;
        }
        else if ((vestasruzhiz[ves] >= 8.5) &&
(vestasruzhiz[ves] < 9.5))
        {
            vestasguc[ves] = 1400;
        }

        else if ((vestasruzhiz[ves] >= 9.5) &&
(vestasruzhiz[ves] < 10))
        {
            vestasguc[ves] = 1875;
        }
        else if ((vestasruzhiz[ves] >= 10) &&
(vestasruzhiz[ves] < 10.5))
        {
            vestasguc[ves] = 2000;
        }
        else if ((vestasruzhiz[ves] >= 10.5) &&
(vestasruzhiz[ves] < 11))
        {
            vestasguc[ves] = 2300;
        }
        else if ((vestasruzhiz[ves] > 11) &&
(vestasruzhiz[ves] <= 11.5))
        {
            vestasguc[ves] = 2500;
        }
        else if ((vestasruzhiz[ves] > 11.5) &&
(vestasruzhiz[ves] < 12))
        {
            vestasguc[ves] = 2750;
        }
        else if ((vestasruzhiz[ves] >= 12) &&
(vestasruzhiz[ves] < 13))
        {
            vestasguc[ves] = 2850;
        }

```

```

else if ((vestasruzhiz[ves] >= 13) &&
(vestasruzhiz[ves] < 14))
{
    vestasguc[ves] = 2950;
}
else if ((vestasruzhiz[ves] >= 14) &&
(vestasruzhiz[ves] <= 15))
{
    vestasguc[ves] = 2975;
}

else
{
    vestasguc[ves] = 3000;
}
listBox15.Items.Add(vestasguc[ves]);
enerji += (vestasguc[ves] / 6);
enerjigunluk = (enerji * 144) / ves;
textBox22.Text = enerjigunluk.ToString();
if ((karsilastir > 4) && (karsilastir <= 4.5))
{
    vestasguc[ves] = 100;
}
else if ((karsilastir > 4.5) && (karsilastir <=
5))
{
    vestasguc[ves] = 175;
}
else if ((karsilastir > 5) && (karsilastir <= 6))
{
    vestasguc[ves] = 225;
}
else if ((karsilastir > 6) && (karsilastir <= 7))
{
    vestasguc[ves] = 575;
}
else if ((karsilastir > 7) && (karsilastir <= 8))
{
    vestasguc[ves] = 850;
}
else if ((karsilastir > 8) && (karsilastir < 8.5))
{
    vestasguc[ves] = 1000;
}
else if ((karsilastir >= 8.5) && (karsilastir <
9.5))
{
    vestasguc[ves] = 1400;
}

else if ((karsilastir >= 9.5) && (karsilastir <
10))
{
    vestasguc[ves] = 1875;
}
else if ((karsilastir >= 10) && (karsilastir <
10.5))
{
    vestasguc[ves] = 2000;
}

```

```

else if ((karsilastir >= 10.5) && (karsilastir <
11))
{
    vestasguc[ves] = 2300;
}
else if ((karsilastir > 11) && (karsilastir <=
11.5))
{
    vestasguc[ves] = 2500;
}
else if ((karsilastir > 11.5) && (karsilastir <
12))
{
    vestasguc[ves] = 2750;
}
else if ((karsilastir >= 12) && (karsilastir <
13))
{
    vestasguc[ves] = 2850;
}
else if ((karsilastir >= 13) && (karsilastir <
14))
{
    vestasguc[ves] = 2950;
}
else if ((karsilastir >= 14) && (karsilastir <=
15))
{
    vestasguc[ves] = 2975;
}

else
{
    vestasguc[ves] = 3000;
}
label93.Text = karsilastir.ToString();
label94.Text = vestasguc[ves].ToString();
if (label72.Text == "")
{
    MessageBox.Show(this, "Lütfen Bir Günlük
Ortalama Elektrik Enerjisi İhtiyacını Hesaplayın");
}
else
{
    turbinsayisi =
Math.Floor((Convert.ToDouble(label72.Text) / (vestasguc[ves] * 24)));
    if (turbinsayisi == 0)
    {
        textBox19.Text = "1";
        turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
        textBox23.Text =
turbingrubuenerji.ToString();
        label100.Text = (turbingrubuenerji *
365).ToString();
    }
    else
    {
        textBox19.Text = turbinsayisi.ToString();

```

```

        turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
        textBox23.Text =
turbingrubuenerji.ToString();
        label100.Text = (turbingrubuenerji *
365).ToString();
    }
}
}
else
{
    pictureBox2.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Yeni Klasör\\v112.jpg";
    pictureBox1.ImageLocation =
"C:\\Users\\Prozac\\Desktop\\Res\\V112_3MW.jpg";
    groupBox20.Enabled = true;
    label90.Text = "V112-3MW".ToString();
    radioButton19.Enabled = false;
    radioButton20.Enabled = false;
    radioButton21.Enabled = false;

    radioButton19.Visible = false;
    radioButton20.Visible = false;
    radioButton21.Visible = false;
    radioButton14.Text = "100";
    radioButton18.Text = "112";
    label87.Text = "3 kanatlı";
    label74.Text = "5m/s 150kw";
    label75.Text = "10m/s 1700kw";
    label76.Text = "15m/s 3000kw";
    label78.Text = "90";
    label80.Text = "6,362";
    label82.Text = "16,1";
    label84.Text = "3,5m/s";
    label86.Text = "15m/s";
    label113.Text = "3000";
    listBox15.Items.Clear();
    for (ves = 0; ves <= vestassatir - 1; ves++)
    {
        vestasruzhiz[ves] =
Convert.ToDouble(listBox14.Items[ves]);
        if ((vestasruzhiz[ves] > 3) && (vestasruzhiz[ves]
<= 4))
        {
            vestasguc[ves] = 50;
        }
        else if ((vestasruzhiz[ves] > 4) &&
(vestasruzhiz[ves] <= 5))
        {
            vestasguc[ves] = 375;
        }
        else if ((vestasruzhiz[ves] > 5) &&
(vestasruzhiz[ves] <= 6))
        {
            vestasguc[ves] = 550;
        }
        else if ((vestasruzhiz[ves] > 6) &&
(vestasruzhiz[ves] <= 7))
        {

```



```

        vestasguc[ves] = 1000;
    }
    else if ((vestasruzhiz[ves] > 7) &&
(vestasruzhiz[ves] <= 8))
    {
        vestasguc[ves] = 1475;
    }
    else if ((vestasruzhiz[ves] > 8) &&
(vestasruzhiz[ves] < 8.5))
    {
        vestasguc[ves] = 1900;
    }
    else if ((vestasruzhiz[ves] >= 8.5) &&
(vestasruzhiz[ves] < 9.5))
    {
        vestasguc[ves] = 2375;
    }

    else if ((vestasruzhiz[ves] >= 9.5) &&
(vestasruzhiz[ves] < 10.5))
    {
        vestasguc[ves] = 2700;
    }
    else if ((vestasruzhiz[ves] >= 10.5) &&
(vestasruzhiz[ves] < 11.5))
    {
        vestasguc[ves] = 2900;
    }
    else if ((vestasruzhiz[ves] >= 11.5) &&
(vestasruzhiz[ves] < 12))
    {
        vestasguc[ves] = 2975;
    }

    else
    {
        vestasguc[ves] = 3000;
    }
    listBox15.Items.Add(vestasguc[ves]);
    enerji += (vestasguc[ves] / 6);
    enerjigunluk = (enerji * 144) / ves;
    textBox22.Text = enerjigunluk.ToString();
    if ((karsilastir > 3) && (karsilastir <= 4))
    {
        vestasguc[ves] = 50;
    }
    else if ((karsilastir > 4) && (karsilastir <= 5))
    {
        vestasguc[ves] = 375;
    }
    else if ((karsilastir > 5) && (karsilastir <= 6))
    {
        vestasguc[ves] = 550;
    }
    else if ((karsilastir > 6) && (karsilastir <= 7))
    {
        vestasguc[ves] = 1000;
    }
    else if ((karsilastir > 7) && (karsilastir <= 8))
    {

```

```

        vestasguc[ves] = 1475;
    }
    else if ((karsilastir > 8) && (karsilastir < 8.5))
    {
        vestasguc[ves] = 1900;
    }
    else if ((karsilastir >= 8.5) && (karsilastir <
9.5))
    {
        vestasguc[ves] = 2375;
    }

    else if ((karsilastir >= 9.5) && (karsilastir <
10.5))
    {
        vestasguc[ves] = 2700;
    }
    else if ((vestasruzhiz[ves] >= 10.5) &&
(vestasruzhiz[ves] < 11.5))
    {
        vestasguc[ves] = 2900;
    }
    else if ((karsilastir >= 11.5) && (karsilastir <
12))
    {
        vestasguc[ves] = 2975;
    }

    else
    {
        vestasguc[ves] = 3000;
    }
    label93.Text = karsilastir.ToString();
    label94.Text = vestasguc[ves].ToString();
    if (label72.Text == "")
    {
        MessageBox.Show(this, "Lütfen Bir Günlük
Ortalama Elektrik Enerjisi İhtiyacını Hesaplayın");
    }
    else
    {
        turbinsayisi =
Math.Floor((Convert.ToDouble(label72.Text) / (vestasguc[ves] * 24)));
        if (turbinsayisi == 0)
        {
            textBox19.Text = "1";
            turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
            textBox23.Text =
turbingrubuenerji.ToString();
            label100.Text = (turbingrubuenerji *
365).ToString();
        }
        else
        {
            textBox19.Text = turbinsayisi.ToString();
            turbingrubuenerji = enerjigunluk *
Convert.ToDouble(textBox19.Text);
            textBox23.Text =
turbingrubuenerji.ToString();

```

```

label100.Text = (turbingrubuenerji *
365).ToString();
    }
}
}
}
}

```

### ***9. Geri ödeme süresinin bulunmasına dair kodlar:***

```

private void geriodemesuresi()
{
    double a,b,c,f;
    double d=0;
    double e = 0;
    a=Convert.ToDouble(label162.Text);
    b=Convert.ToDouble(label164.Text);
    c=Convert.ToDouble(label160.Text);
    d=(a-b)*c;
    f = Convert.ToDouble(label166.Text);
    e = f / d;
    label168.Text = Math.Round(e,3).ToString();
}

```

## ÖZGEÇMİŞ

Adı Soyadı : Gönenç AYDIN

Doğum Yeri : Kırklareli

Doğum Yılı : 1982

Eğitimi :

Yüksek Lisans 2005 -2010 İstanbul Üniversitesi Fen Bilimleri Elektrik-Elektronik Mühendisliği Ana Bilim Dalı

Lisans 2000-2005 Yıldız Teknik Üniversitesi Elektrik-Elektronik Fakültesi Elektrik Mühendisliği Bölümü

Lise 1997-2000 Kırklareli Anadolu Lisesi

Ortaokul 1994-1997 Kırklareli Anadolu Lisesi

Hazırlık 1993-1994 Kırklareli Anadolu Lisesi

İlkokul 1988-1993 Hamdi Helvacıoğlu İlkokulu

Yabancı Dil : İngilizce

İş Tecrübesi :

2010-..... YKS Tesis Yönetim Hizmetleri A.Ş.  
Elektrik Mühendisi, Enerji Yöneticisi, Operasyon Kısmı

2009-2010 Can Enerji Üretim A.Ş.  
Saha Mühendisliği, İşletme Kısmı

2007-2008 İstanbul Işık Yapı-Denetim A.Ş.  
Kontrol Mühendisi, Denetim Bölümü

2006-2007 Elektrik Mühendisleri Odası İstanbul Şubesi (E.M.O.)  
Denetçi Mühendis, Operasyon Bölümü

2005-2006 HoneyHome Otomasyon Ltd. Şti.  
Yazılım Mühendisi, Mühendislik Bölümü







