



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE SOBRAL
CURSO DE ENGENHARIA ELÉTRICA

ACÉLIO LUNA MESQUITA

**DESENVOLVIMENTO DE UMA FERRAMENTA COMPUTACIONAL PARA
ANÁLISE DO POTENCIAL EÓLICO E ESTIMATIVA DE GERAÇÃO DE
ENERGIA EÓLICA**

SOBRAL
2015

ACÉLIO LUNA MESQUITA

**DESENVOLVIMENTO DO PROGRAMA PARA ESTUDO DE CASO PARA
GERAÇÃO DE ENERGIA EÓLICA**

Trabalho de Conclusão de Curso submetido à Coordenação do Curso de Engenharia Elétrica, da Universidade Federal do Ceará, como requisito parcial para obtenção do título de Bacharel em Engenharia Elétrica. Áreas de Concentração: Fontes Alternativas de Energia.

Orientador: Prof. Dr. Vandilberto Pereira Pinto.

SOBRAL

2015

ACÉLIO LUNA MESQUITA

**DESENVOLVIMENTO DO PROGRAMA PARA ESTUDO DE CASO PARA
GERAÇÃO DE ENERGIA EÓLICA**

Trabalho de Conclusão de Curso submetido à
Coordenação do Curso de Engenharia Elétrica,
da Universidade Federal do Ceará, como
requisito parcial para obtenção do título de
Bacharel em Engenharia Elétrica. Áreas de
Concentração: Fontes Alternativas de Energia.

Aprovada em: 18/02/2016.

BANCA EXAMINADORA

Vandilberto Pereira Pinto

Prof. Dr. Vandilberto Pereira Pinto (Orientador)

Universidade Federal do Ceará (UFC)

Nilena Brito Maciel Dias

Prof. Ma. Nilena Brito Maciel Dias

Universidade Federal do Ceará (UFC)

Washington Luís Araújo Siqueira

Eng. Washington Luís Araújo Siqueira

Universidade Federal do Ceará (UFC)

A Deus, Triuno.

A meus pais, Carlos Alberto e Elza Fernandes.

A meus irmãos, Alice e Arcelino.

A minha tia materna Vanderleide

AGRADECIMENTOS

A Deus, por estar sempre presente em todos os momentos de minha vida, me auxiliando e me dando forças para superar todos os desafios que passaram e que estão por vir. Sem Ele, nada disto seria possível.

A minha família, pelo apoio incondicional durante todos esses anos.

A minha irmã, Alice Luna, por sempre acreditar no meu potencial até mesmo em momentos que eu mesmo não acreditava.

Ao Prof. Dr. Vandilberto Pereira Pinto pela orientação, auxílio, paciência e disponibilização, dado durante todo o desenvolvimento do software e do trabalho de conclusão do curso.

Aos amigos professores, Nilena Dias, Eber Diniz, Marcus Rogerio, André Lima, por toda a paciência e elucidação tanto dentro como fora de aula.

Ao engenheiro participante da banca examinadora, Washington Luís Araujo Siqueira, por toda a ajuda durante todos os anos de graduação e auxílio na elaboração do trabalho de conclusão do curso.

Aos amigos e colegas de graduação, Douglas Aquino, Rafael Lima, Pedro Servolu, Aldenir Araújo, Rodolfo Marques, Roberto Bruno e Jean Carlos pelas horas, noites e finais de semanas de estudos e companheirismo, com os quais suportar o fardo pesado e o karma do curso de Engenharia Elétrica tornou-se mais agradável.

Ao não só meu, mas também amigo de todos os graduandos de engenharia, Germano Araújo da Silva pelo seu companheirismo e pelas palavras sábias nos momentos difíceis.

RESUMO

A energia eólica é uma das principais fontes de energia alternativa, renovável e limpa para a produção de energia elétrica. Hoje a energia eólica se encontra como uma energia em alta em quase todo o mundo devido a sua competitividade econômica, já que sua fonte é inesgotável. Neste trabalho foi desenvolvido um estudo de caso para as regiões de Icapuí, Sobral e Ubajara assim como o desenvolvimento de um programa para o auxílio do estudo das regiões através da interface gráfica do software Matlab “GUI”, que tem como objetivo o cálculo de toda a parte probabilística da região como a distribuição de Weibull e a lei da potência de Hellman. O estudo desenvolvido mostrou qual a melhor cidade e a melhor turbina eólica para a implantação de um parque eólico fictício, de vinte turbinas eólicas, tendo como critério principal o fator de capacidade para a escolha da turbina eólica escolhida.

Palavras-chave: Energia eólica, Vento, Matlab, Distribuição de Weibull.

ABSTRACT

Wind energy is one of the main sources of alternative energy, renewable and clean for the production of electricity. Today wind power is as high in power in most of the world due to its economic competitiveness, since its source is inexhaustible. In this work a case study for the regions of Icapuí, Sobral and Ubajara well as the development of a program to aid the study of the regions through the graphical interface of Matlab software "GUI", which aims to calculate all the probability of the region as the Weibull distribution and the law of power Hellman. The study conducted showed that the best city and the best wind turbine for the implementation of a fictitious wind farm, twenty wind turbines, the main criterion the capacity factor for the choice of the selected wind turbine.

Keywords: Wind Energy, Wind, Matlab, Weibull distribution.

LISTA DE ILUSTRAÇÕES

Figura 1 – Gráfico com o crescimento da energia eólica no mundo	2
Figura 2 – Porcentagem de cada geração no Brasil	4
Figura 3 – Porcentagem dos empreendimentos em construção	5
Figura 4 – Gráfico com as porcentagens dos empreendimentos que ainda não iniciaram as construções	5
Figura 5 – Mapa do potencial eólico brasileiro	6
Figura 6 – Comparativo nos valores do MWh nos leilões de energia	7
Figura 7 – Variação da intensidade de turbulência pela altura.....	10
Figura 8 – Seção transversal de formato cilíndrico imaginário para o cálculo da potência eólica.....	11
Figura 9 – Parque eólico offshore de Thornton Bank na Bélgica com capacidade de 325MW.....	14
Figura 10 – Parque eólico onshore de Osório Rio Grande do Sul com 150MW instalados..	14
Figura 11 – Histograma da velocidade do vento com a curva de Weibull retirado do programa Acelio_wind.....	16
Figura 12 – Curva de potência de turbinas eólicas.....	17
Figura 13 – Programa Acelio_wind.....	20
Figura 14 – Programa Acelio_wind após a entrada dos dados da velocidade dos ventos.....	21
Figura 15 – Gráfico da rosa dos ventos no programa Acelio_wind.....	22
Figura 16 – Ajuste da curva de Weibull no programa Acelio_wind.....	22
Figura 17 – Valores de performance do aerogerador.....	23
Figura 18 – Local para mudança da altura da planilha.....	24
Figura 19 – Simulação em uma altura de 10m para a cidade de Sobral, para uma turbina E33-330kW.....	26
Figura 20 – Simulação em uma altura de 50m com expoente de Hellman de 0,3 para a cidade de Sobral para a turbina eólica E33-330 kW.....	27
Figura 21 – Simulação em uma altura de 10m para a cidade de Sobral, para uma turbina E82-2000kW.....	28
Figura 22 – Simulação em uma altura de 108 m com expoente de Hellman de 0,3 para a cidade de Sobral para a turbina eólica E33-330 kW.....	29
Figura 23 – Simulação em uma altura de 10m para a cidade de Sobral, para uma turbina	

	E126-7500kW.....	30
Figura 24 –	Simulação em uma altura de 135 m com expoente de Hellman de 0,3 para a cidade de Sobral para a turbina eólica E33-330 kW.....	31
Figura 25 –	Simulação em uma altura de 10m para a cidade de Icapuí, para uma turbina E33-330kW.....	32
Figura 26 –	Simulação em uma altura de 50m com expoente de Hellman de 0,15 para a cidade de Icapuí para a turbina eólica E33-330 kW.....	33
Figura 27 –	Simulação em uma altura de 10m para a cidade de Icapuí, para uma turbina E82-2000kW.....	34
Figura 28 –	Simulação em uma altura de 108 m com expoente de Hellman de 0,15 para a cidade de Icapuí para a turbina eólica E82-2000 kW.....	35
Figura 29 –	Simulação em uma altura de 10m para a cidade de Icapuí, para uma turbina E126-7500kW.....	36
Figura 30 –	Simulação em uma altura de 135 m com expoente de Hellman de 0,15 para a cidade de Icapuí para a turbina eólica E126-7500 kW.....	37
Figura 31 –	Simulação em uma altura de 10m para a cidade de Ubajara, para uma turbina E33-330kW.....	38
Figura 32 –	Simulação em uma altura de 50m com expoente de Hellman de 0,25 para a cidade de Ubajara, para a turbina eólica E33-330 kW.....	39
Figura 33 –	Simulação em uma altura de 10m para a cidade de Ubajara, para uma turbina E82-2000kW.....	40
Figura 34 –	Simulação em uma altura de 108m com expoente de Hellman de 0,25 para a cidade de Ubajara, para a turbina eólica E82-2000 kW.....	41
Figura 35 –	Simulação em uma altura de 10m para a cidade de Ubajara, para uma turbina E126-7500 kW.....	42
Figura 36 –	Simulação em uma altura de 135m com expoente de Hellman de 0,25 para a cidade de Ubajara, para a turbina eólica E126-7500 kW.....	43

LISTA DE SÍMBOLOS

CGH	Central Geradora Hidrelétrica
CGU	Central Geradora Undi-elétrica
EOL	Central Geradora Eólica
PCH	Pequena Central Hidrelétrica
UFV	Central Geradora Solar Fotovoltaica
UHE	Usina Hidrelétrica
UTE	Usina Termelétrica
UTN	Usina Termonuclear
WWEA	World Wind Energy Association
I	Intensidade da turbulência
σ	Desvio padrão
V_m	Velocidade média
P	Potência do vento
v_1	Velocidade do vento que incide na turbina eólica
ρ	Massa específica do ar
A	Área da seção transversal
V_2	Velocidade na altura desejada
V_1	Velocidade na altura da medição
h	Altura desejada
h_1	Altura da medição
α	Coefficiente de atrito
k	Fator de forma
c	Fator de escala
Γ	Função gama
v_p	Velocidade de partida
v_n	Velocidade nominal
v_c	Velocidade de corte
FC	Fator de capacidade

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Motivação	1
1.2	Objetivo geral	7
1.2.1	Objetivos específicos	8
1.3	Estrutura do Trabalho.....	8
2	O VENTO	9
2.1	Turbulência	9
3	BASE DE DADOS E TRATAMENTO ESTATÍSTICO	11
3.1	Potência eólica	11
3.2	Lei da potência de Hellman	12
3.3	Distribuição de Weibull	14
3.4	Curva de potência da turbina eólica.....	16
3.5	Fator de capacidade.....	17
4	PROGRAMA ACELIO_WIND	20
4.1	Consideração inicial	20
4.2	Entrada de dados.....	20
4.3	Ajuste da Distribuição de Weibull.....	22
4.4	Seleção da turbina eólica.....	23
4.5	Mudança de altura da planilha.....	24
5	ANALISE DOS DADOS DAS REGIÕES	25
5.1	Cidade de Sobral	25
5.1.1	<i>Simulação para a turbina E33-330 kW</i>	<i>25</i>
5.1.2	<i>Simulação para a turbina E82-2000 kW</i>	<i>27</i>
5.1.3	<i>Simulação para a turbina E126-7500 kW</i>	<i>29</i>
5.2	Cidade de Icapuí.....	31
5.2.1	<i>Simulação para a turbina E33-330 kW</i>	<i>32</i>
5.2.2	<i>Simulação para a turbina E82-2000 kW</i>	<i>33</i>
5.2.3	<i>Simulação para a turbina E126-7500 kW</i>	<i>35</i>
5.3	Cidade de Ubajara	37
5.3.1	<i>Simulação para a turbina E33-330 kW</i>	<i>38</i>
5.3.2	<i>Simulação para a turbina E82-2000 kW</i>	<i>39</i>

5.3.3	<i>Simulação para a turbina E126-7500 kW</i>	41
5.4	Comparação entre as regiões	43
6	CONCLUSÃO	45
6.1	Trabalhos futuros	45
	REFERÊNCIAS	46
	APÊNDICE A - CÓDIGO DO PROGRAMA NO MATLAB	48

1 INTRODUÇÃO

1.1 Motivação

Hoje o mundo tem como discussão os temas sustentabilidade e energia limpa. A sustentabilidade é tratada com seriedade em vários países pelo mundo e novos meios de captação de energia limpa e renovável entra em destaque para os investimentos dos mesmos.

Desde a assinatura do protocolo de Kyoto, em 1997, alguns países se comprometeram a reduzir as emissões de gases causadores do efeito estufa como o CO₂ (PROTOCOLO DE KYOTO, s/d).

Assim como, o documentário “Uma verdade inconveniente” mostrou as consequências do aquecimento global para a humanidade, o mesmo rendeu o Prêmio Nobel da paz para Al Gore em 2007 (WALTER, 2007).

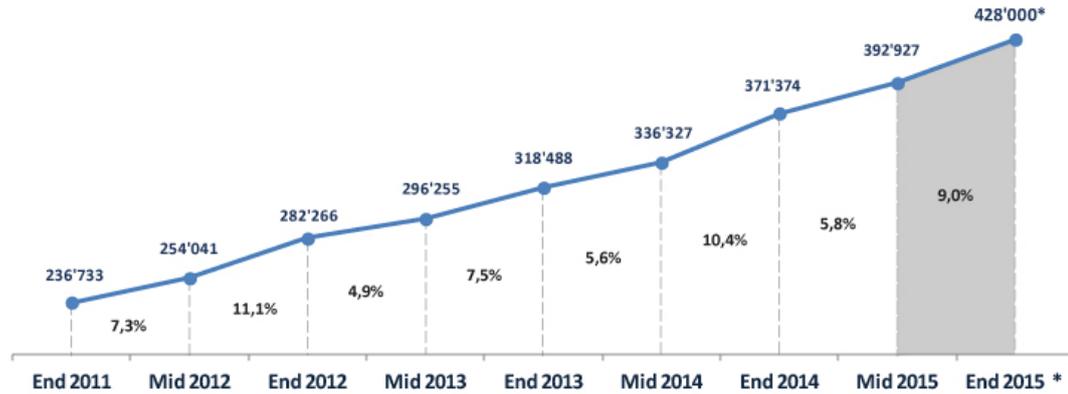
Dessa forma, a maior preocupação com a forma de produção de energia está ligada aos impactos ambientais decorrentes da mesma.

Sendo a energia hoje um item indispensável para a sobrevivência diária e a sua disponibilidade um fator fundamental para o desenvolvimento futuro, novos meios de produção de fontes de energia que sejam confiáveis tendem a aumentar.

Dois tipos de energias renováveis entram em destaque por sua produção ser limpa e renovável, são elas a energia fotovoltaica (energia solar) e a energia eólica (energia dos ventos) (CALDAS, 2002).

O investimento nessas energias no Brasil tem crescido muito, principalmente a energia eólica, superando e muito o crescimento dos outros países mundo afora (CALDAS, 2010). Segundo o relatório anual da WWEA (WORLD WIND ENERGY ASSOCIATION, 2015), o crescimento da energia eólica no mundo vem batendo recordes de instalação com uma totalização de 392.927 MW até metade de 2015 como mostra o gráfico da Figura 1.

Figura 1 – Gráfico com o crescimento da energia eólica no mundo



* Estimated

Total installed capacity: Includes all installed wind capacity, connected and not-connected to the grid.

© WWEA - 2015

Fonte: World Wind Energy Association

Hoje os países com maior produção e capacidade instalada em energia eólica são a China e os Estados Unidos, A China com uma capacidade instalada de 124.710MW e os Estados Unidos com 67.870MW. O Brasil vem mostrando um crescimento expressivo nos últimos anos, mais do que dobrando a sua capacidade instalada que era de 2.788MW em junho de 2013 para 6.800MW em junho de 2015. Com esses dados o Brasil ocupa a décima colocação no ranking de instalação de energia eólica mundial, como mostra a Tabela 1.

Tabela 1 - Ranking dos países com maior capacidade instalada em energia eólica

Position	Country/Region	Total capacity June 2015 [MW]	Added capacity H1 2015 [MW]	Total capacity end 2014 [MW]	Added capacity H1 2014 [MW]	Total capacity end 2013 [MW]	Total capacity June 2013 [MW]
1	China	124'710	10'101	114'763	7'175	91'413	80'827
2	United States	67'870	1'994	65'754	835	61'108	59'884
3	Germany	42'367	1'991	40'468	1'830	34'658	32'458
4	India *	23'762	1'297	22'465	1'112	20'150	19'564
5	Spain	22'987	0	22'987	0	22'959	22'918
6	United Kingdom	13'313	872	12'440	649	10'531	9'776
7	Canada	10'204	510	9'694	723	7'698	6'578
8	France	9'819	523	9'296	338	8'254	7'697
9	Italy	8'787	124	8'663	30	8'551	8'417
10	Brazil	6'800	838	5'962	1'301	3'399	2'788
11	Sweden	5'582	157	5'425	354	4'470	4'271
12	Denmark	4'959	76	4'883	83	4'772	4'578
13	Portugal *	4'953	0	4'953	105	4'724	4'547
14	Turkey	4'193	431	3'763	466	2'958	2'619
15	Poland	4'117	283	3'834	337	3'390	2'798
16	Australia	4'006	200	3'806	699	3'049	3'059
	Rest of the World	34'600	2'400	32'219	1576	26'493	23'802
	Total	392'927	21'678	371'374	17'613	318'577	296'581

Fonte: World Wind Energy Association

O aumento na capacidade instalada não aumenta só a produção de energia elétrica para um país, o investimento na mesma vem atraindo o mercado e gerando mais competitividade.

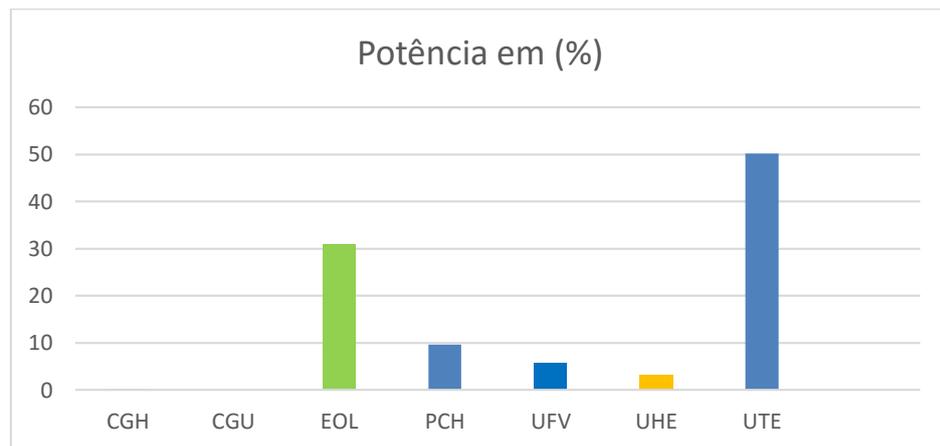
Segundo o secretário geral da WWEA Stefan Gsänger, “O mercado mundial em energia eólica está crescendo como nunca antes, e nós esperamos um novo recorde de novas instalações para 2015” (WORD WIND ENERGY ASSOCIATION, 2015). Ele ainda complementa que o mercado chinês ainda continua sendo o maior “O principal mercado continua sendo a China, com um crescimento surpreendente de mais de 10GW dentro de 6 meses” (Loc. Cit., 2015). O mercado brasileiro também chega a ser mencionado como um dos que teve maior crescimento “O Brasil apresentou a maior taxa de crescimento de todos os principais mercados, o país aumentou a sua capacidade de energia eólica em 14% desde o início deste ano” (Loc. Cit., 2015).

Segundo a Agencia Nacional de Energia Elétrica (ANEEL, 2015) o Brasil possui no total 4.394 empreendimentos de geração de energia elétrica em operação, totalizando 139.924.122 kW de potência instalada e está previsto para os próximos anos uma adição de

40.461.406 kW na capacidade de geração do país, proveniente de 230 empreendimentos atualmente em construção e mais 633 ainda não iniciados.

A principal fonte de energia elétrica do Brasil está na geração hidrelétrica hoje com uma produção de 61,41% de toda a energia do país, logo atrás está a energia termelétrica com 28,15% e em terceiro lugar a energia eólica com 5,28% da produção nacional, como mostrados pela Figura 2 e pela Tabela 2.

Figura 2 – Porcentagem de cada geração no Brasil



Fonte: ANEEL

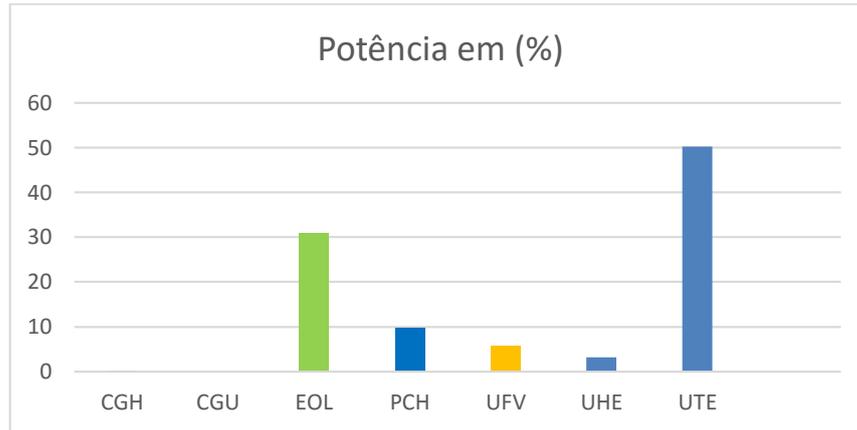
Tabela 2 – Geração de cada fonte de energia no Brasil

Empreendimentos em Operação				
Tipo	Quantidade	Potência Outorgada (kW)	Potência Fiscalizada (kW)	%
CGH	538	393.123	395.026	0,28
EOL	308	7.480.673	7.381.102	5,28
PCH	465	4.842.727	4.822.273	3,45
UFV	33	25.336	21.336	0,02
UHE	197	89.518.401	85.929.188	61,41
UTE	2.851	41.261.551	39.385.197	28,15
UTN	2	1.990.000	1.990.000	1,42
Total	4.394	145.511.811	139.924.122	100

Fonte: ANEEL

E os empreendimentos que hoje estão em construção, estão ilustrados pela Figura 3 e pela Tabela 3.

Figura 3 – porcentagem dos empreendimentos em construção



Fonte: ANEEL

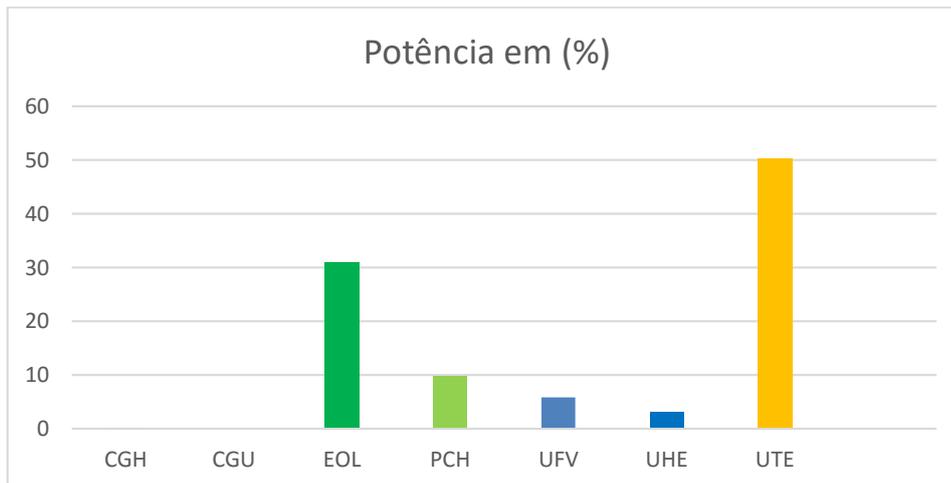
Tabela 3 – Capacidade dos empreendimentos em construção no Brasil

Empreendimentos em Construção			
Tipo	Quantidade	Potência Outorgada (kW)	%
CGH	1	848	0
EOL	167	4.020.204	19,45
PCH	32	411.218	1,99
UHE	10	13.449.342	65,08
UTE	19	1.434.639	6,94
UTN	1	1.350.000	6,53
Total	230	20.666.251	100

Fonte: ANEEL

E os empreendimentos que ainda não estão com as construções iniciadas, estão elucidados pela Figura 4 e pela Tabela 4.

Figura 4 – Gráfico com as porcentagens dos empreendimentos que ainda não iniciaram as construções.



Fonte: ANEEL

Tabela 4 – Empreendimentos que ainda não iniciaram as construções

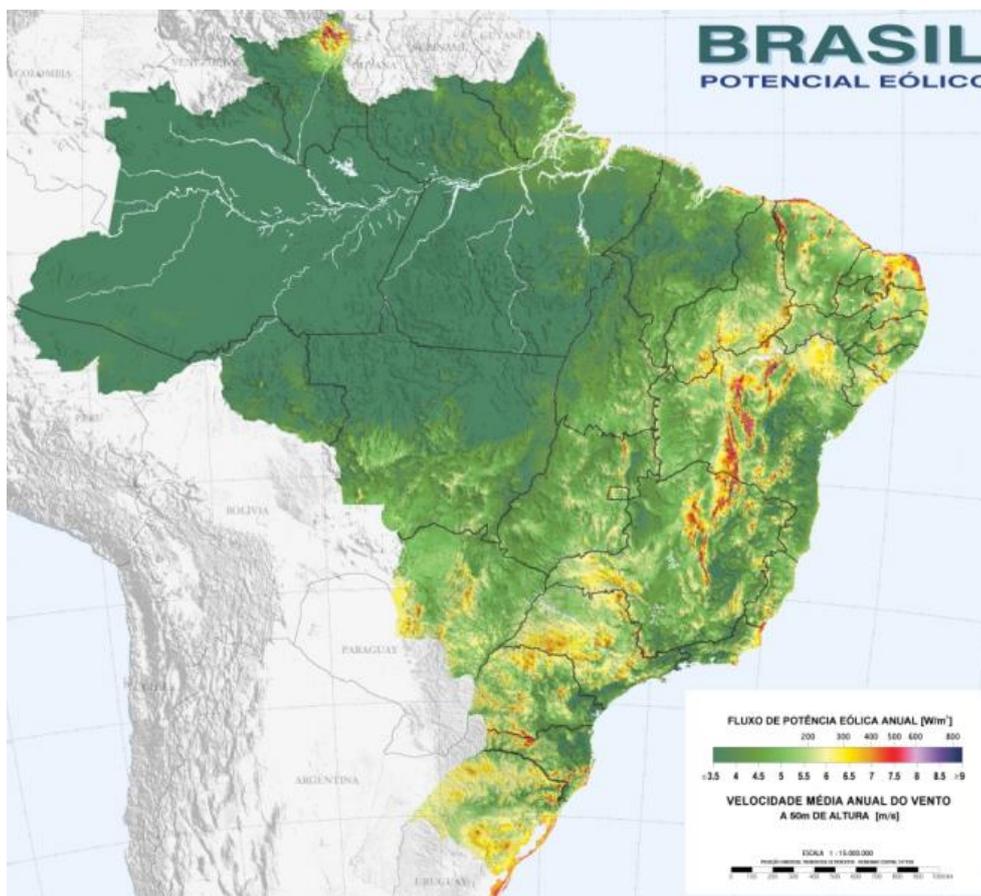
Empreendimentos com Construção não iniciada			
Tipo	Quantidade	Potência Outorgada (kW)	%
CGH	41	30.279	0,15
CGU	1	50	0
EOL	261	6.133.650	30,99
PCH	134	1.914.569	9,67
UFV	40	1.142.975	5,77
UHE	6	629.000	3,18
UTE	150	9.944.632	50,24
Total	633	19.795.155	100

Fonte: ANEEL

A partir dos dados mostrados acima fica claro o grau de investimento que o Brasil vem fazendo para o crescimento na área da energia eólica no país.

Segundo o atlas eólico Brasileiro publicado desde 2001 pelo Ministério de Minas e Energia (AMARANTE *et. al.*, 2001), o Brasil tem um potencial para geração de energia eólica de 150GW a 50m de altura e 250GW a 100m de altura, elucidado pela Figura 5.

Figura 5 – Mapa do potencial eólico brasileiro.



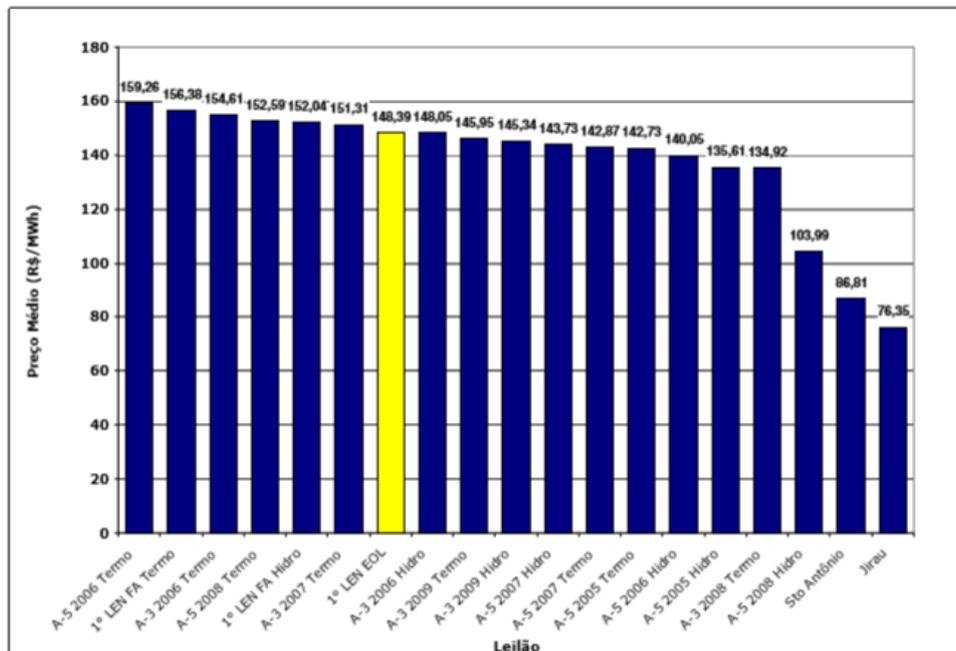
Fonte: Ministério de Minas e Energia, CEPTEL. 2001

Visando esse grande potencial energético, o governo promoveu o primeiro leilão específico para a energia eólica em dezembro de 2009 (CALDAS, 2010).

A procura pelos investidores tanto estatais como privados foi enorme, com mais de 10GW de projetos habilitados tecnicamente voltados para essa energia. Ao fim do leilão foi arrematado 1.800MW com tarifas que variaram entre de R\$ 131,00 a R\$ 152,00 por MWh (CALDAS, 2002).

Com esse resultado foi mostrado que a energia eólica entrou de forma competitiva no mercado comparada com as outras fontes de energia elétrica no Brasil, como mostrado pela Figura 6.

Figura 6 – Comparativo dos valores do MWh nos leilões de energia



Fonte: Revista Brasil Energia.

1.2. Objetivo Geral

O objetivo geral do trabalho é analisar um estudo do potencial eólico de uma determinada localidade e estimativa da produção de energia eólica utilizando um programa desenvolvido a partir da interface gráfica do Matlab “GUI”. Dessa forma, será possível utilizar o mesmo base de dados para o levantamento do estudo e podendo evitar softwares pagos como o WAsP, software dinamarquês mais utilizado no mercado para esse tipo de estudo.

1.2.1 Objetivos Específicos

- 1- Apontar o potencial eólico de uma região;
- 2- Elaborar estudo de caso das regiões de Sobral, Icapuí e Ubajara, que foram disponibilizados pela Fundação Cearense de Meteorologia e Recursos Hídricos (FUNCEME);
- 3- Identificar qual a melhor turbina para a região em estudo;
- 4- Ilustrar a implantação de 20 turbinas para região em estudo.

1.3. Estrutura do trabalho

Este trabalho foi dividido de forma a elucidar a metodologia utilizada no trabalho do cálculo do potencial eólico.

No segundo capítulo, abordaremos todas as informações referentes aos ventos, suas formas de medição e aerogeradores.

No terceiro capítulo, abordaremos todo o embasamento matemático e probabilístico necessário para o estudo do potencial eólico de uma região.

Dando continuidade ao estudo, o quarto capítulo elucidará a criação do programa Acelio_wind e sua facilitação nos estudos de um potencial eólico tal como a possível escolha do aerogerador.

No quinto capítulo, contém todos os resultados do estudo dos locais assim como o potencial financeiro da implantação de um parque eólico.

No último capítulo, contém a conclusão do estudo assim como os possíveis trabalhos futuros.

2 O VENTO

“O vento nada mais é que o ar em movimento.” (PINTO, p. 46, 2014)

A formação do vento ocorre devido à radiação solar que não se distribui de forma igual na superfície da terra, essa não uniformidade no aquecimento da atmosfera ocorre, entre outros fatores, devido à orientação dos raios solares e aos movimentos da Terra, fazendo com que haja porções na atmosfera que são mais aquecidas, tornando-se menos densas e com menor pressão, ocupando assim as camadas superiores, e outras menos aquecidas que são mais densas e com maior pressão, que ocupam as regiões inferiores, formando as correntes de vento (PINTO, 2014).

2.1 Turbulência

O estudo da turbulência tem grande importância para o estudo da produção da energia eólica, pois o mesmo é a causa de flutuações e variações na potência gerada na turbina eólica, além de causar estresse na estrutura da turbina. A turbulência é causada por obstáculos, pela rugosidade e pela orografia do terreno e ao seu entorno (MUELLER, 2004).

Uma das principais causas de turbulência é o atrito com a superfície da terra e as massas de ar que se movem verticalmente devido a efeitos térmicos, assim gerando variações tanto de temperatura como de densidade do ar (MUELLER, 2004).

A vários meios estatísticos de calcular a turbulência dependendo da aplicação desejada.

A intensidade da turbulência pode ser calculada para representar a turbulência, a mesma é elucidada pela equação 1.

$$I = \frac{\sigma}{V_m} \quad (1)$$

Onde:

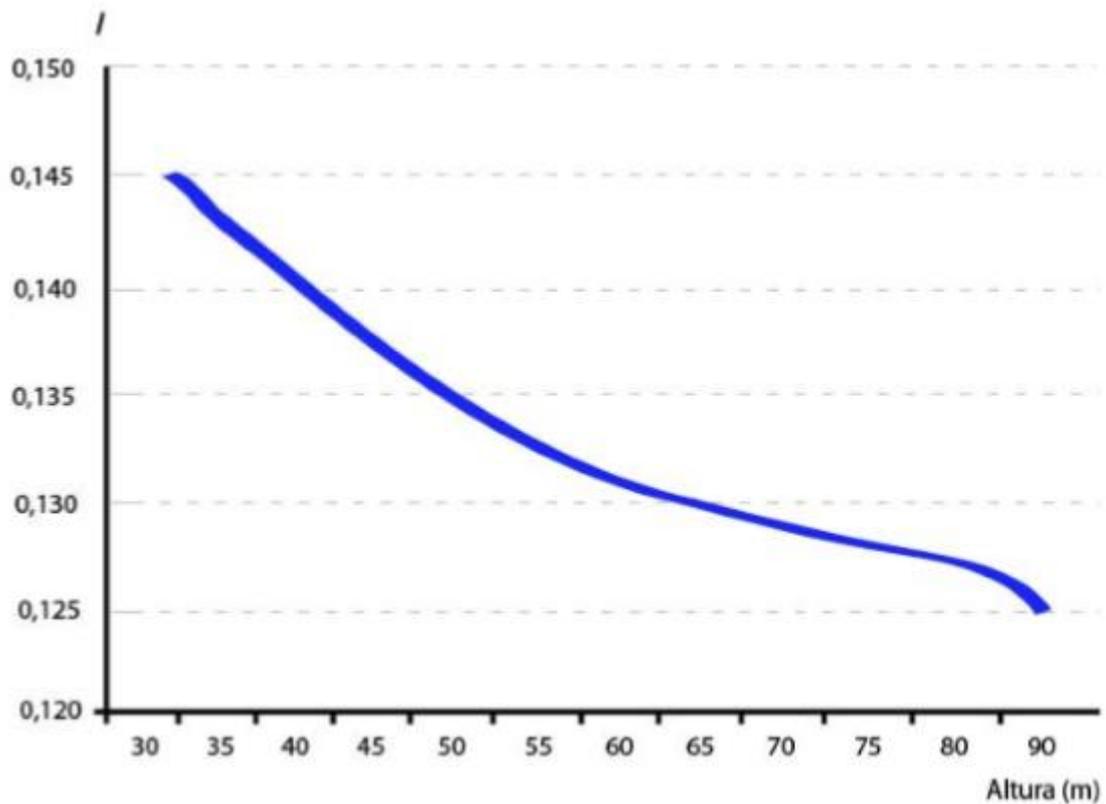
I = intensidade da turbulência

σ = desvio padrão

V_m = velocidade média

A intensidade da turbulência diminui com o aumento da altura, o mesmo está relacionado com a rugosidade do terreno. A Figura 7 elucida a variação da intensidade de turbulência “ I ” com a altura.

Figura 7: Variação da intensidade de turbulência pela altura.



Fonte: MUELLER, 2004.

A Tabela 5 mostra os valores indicados para a intensidade de turbulência para diferentes tipos de regiões.

Tabela 5: Valores de intensidade de turbulência.

Tipo de Terreno	Intensidade de turbulência – I
Água aberta (mares, lagos, etc)	0,10
Plano ou levemente ondulado	0,15
Morros	0,20
Pequenas Montanhas	0,25
Grandes Montanhas	0,30

Fonte: MUELLER, 2004.

3 BASE DE DADOS E TRATAMENTO ESTATÍSTICO

Para um estudo da energia eólica o engenheiro necessita do conhecimento de algumas equações físicas e alguns conhecimentos em cálculos básicos, para uma total compreensão dos mesmos.

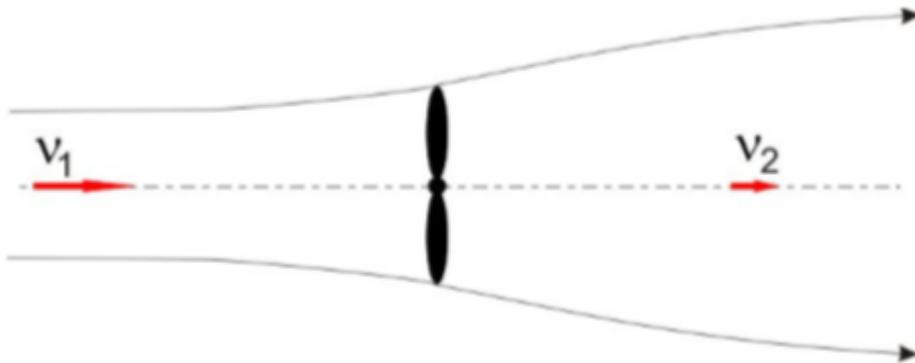
Inicialmente teremos que elucidar os fatores principais para a geração da energia eólica, ou seja, o vento que necessita de uma medição no local a ser implantado o parque eólico tanto pela sua direção como pela sua velocidade. A partir dessas necessidades tornasse de extrema importância aparelhos de medições de confiança (PINTO, 2014).

Após as medições necessárias da velocidade e da direção dos ventos, para uma melhor confiabilidade dos dados os mesmos devem ter sido medidos por no mínimo um ano.

3.1 Potência eólica

Considerando um fluxo de ar que se move a uma velocidade v_1 através da seção transversal de um tubo imaginário de formato cilíndrico (turbina eólica), como mostrado na Figura 8.

Figura 8: Seção transversal de formato cilíndrico imaginário para o cálculo da potência eólica



Fonte: MUELLER, 2004

Segundo Pinto (2014), a potência disponível pode ser demonstrada pelo vento que passa pela seção A, transversal ao fluxo de ar que é dado por:

$$P = \frac{1}{2} \rho A v_1^3 \quad (1)$$

No qual:

P = potência do vento (W);

v_1 = velocidade do vento que incide na turbina eólica (m/s);

ρ = massa específica do ar (kg/m^3);

A = área da seção transversal (m^2).

Analisando a expressão 1 podemos verificar que a potência eólica dependerá de três fatores, que são a massa específica do ar, com o cubo da velocidade do vento e com área da seção transversal por onde o ar irá fluir. Com isso torna-se de vital importância uma correta análise desses fatores na região a ser implantado o parque eólico, pois, os mesmos influenciaram de forma significativa no valor final da potência eólica disponível.

3.2 Lei da potência de Hellman

Devemos começar analisando sua velocidade média do local, uma simples média aritmética. Logo em seguida devemos analisar a altura em que foi feita as análises do vento, geralmente os dados utilizados pela FUNCEME tem uma medição a uma altura de 10m a partir disso devemos utilizar cálculos estatísticos a partir da equação conhecida como a lei da potência ou lei da altitude de Hellman para uma aproximação de uma velocidade em uma altura desejada, já que a mesma varia com a altura (PESSANHA, 2010):

$$V_2 = V_1 \left(\frac{h}{h_1} \right)^\alpha \quad (2)$$

No qual:

V_2 = velocidade na altura desejada;

V_1 = velocidade na altura da medição;

h = altura desejada;

h_1 = altura da medição;

α = coeficiente de atrito.

O coeficiente de atrito (α) ou também chamado de coeficiente de Hellman é uma função do terreno onde o vento sopra, o mesmo é de vital importância para os cálculos de especulação da velocidade dos ventos e pode ser calculado a partir da medição da velocidade dos ventos em duas alturas da mesma região a partir da fórmula:

$$\alpha = \frac{\ln(V_{h_2} / V_{h_1})}{\ln(h_2 / h_1)} \quad (3)$$

Onde;

V_{h_1} = velocidade na altura de medição 1.

V_{h_2} = velocidade na altura de medição 2.

h_1 = altura da medição 1.

h_2 = altura da medição 2.

A partir da Tabela 6 mostrada abaixo podemos observar o valor do coeficiente de atrito para alguns tipos de terrenos.

Tabela 6: Coeficiente de atrito para vários tipos de terrenos

Características do terreno	Coeficiente de atrito (α)
Calma superfície aquática ou solo suave	0,10
Gramma alta ao nível do solo	0,15
Arbustos e cercas	0,20
Áreas rurais com muitas árvores	0,25
Pequenas cidades com árvores e arbustos	0,30
Grandes cidades com prédios elevados	0,40

Fonte: PINTO, 2014.

Como mostrado pela Tabela 7, podemos observar que quanto mais plano o terreno menor será o valor para o coeficiente de atrito α e devido a isso uma melhor região para a implantação de parques eólicos são regiões litorâneas tanto *offshore* (no mar) como *onshore* (em terra), elucidados pelas Figuras 9 e 10.

Figura 9: Parque eólico *offshore* de Thornton Banck na Belgica com capacidade de 325MW



Fonte: <http://www.portal-energia.com/inaugurado-maior-parque-eolico-offshore-mar-norte-belga/>.

Figura 10: Parque eólico *onshore* de Osório Rio Grande do Sul com 150MW instalados



Fonte: <https://eco4u.wordpress.com/tag/parque-eolico/>

3.3 – Distribuição de Weibull

Após os ajustes com a altura da velocidade dos ventos o responsável pelo estudo deve então utilizar esses dados e representá-los em uma distribuição da velocidade do vento utilizando a equação 4:

$$V = \int_0^{\infty} f(v)dv \quad (4)$$

Com a construção do histograma deve utilizar técnicas estatísticas no caso a função de densidade de probabilidade e a que mais se ajusta para uma distribuição dos ventos é a distribuição de Weibull (PINTO, 2014).

A função de Weibull é comumente utilizada como um ponto de partida para caracterizar as estatísticas da velocidade do vento, dada pela equação abaixo:

$$F(v) = \left(\frac{k}{c}\right) \cdot \left(\frac{v}{c}\right)^{k-1} e^{-\left(\frac{v}{c}\right)^k} \quad (5)$$

Onde:

v = velocidade do vento (m/s)

k = fator de forma (m/s)

c = fator de escala (adimensional)

Os coeficientes de forma (k) e de escala (c) podem ser calculados de diversas formas, porém, a forma sugerida foi o método da velocidade média e desvio padrão, para o cálculo do fator de forma “ k ”, a formula está elucidada abaixo:

Fator de forma k :

$$k = \left(\frac{\sigma}{V_m}\right)^{-1,086} \quad (6)$$

Onde:

σ = desvio padrão;

V_m = velocidade média.

O fator de escala “ c ” pode ser calculado a partir da relação entre a função Gama e o fator de forma “ k ” de Weibull, pela equação 7 mostrada abaixo:

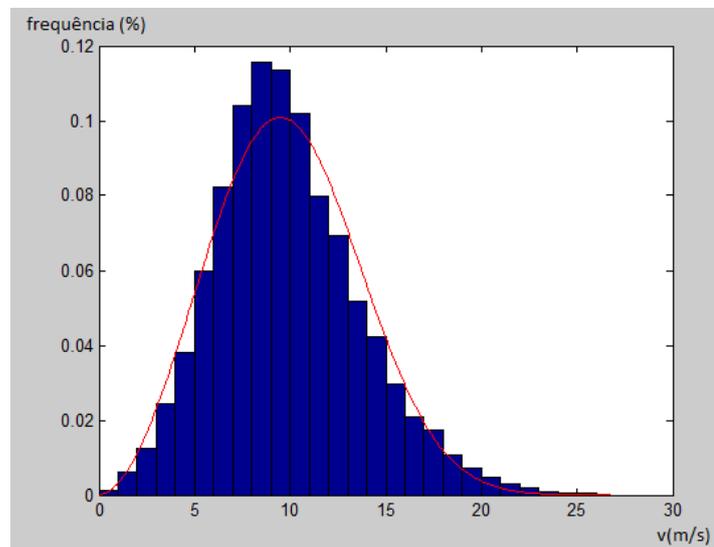
Fator de escala c :

$$c = \frac{V_m}{\Gamma\left(1 + \left(\frac{1}{k}\right)\right)} \quad (7)$$

O fator de escala “c” está relacionado com a velocidade média do local estudado, sendo expresso por unidades de velocidade.

Com um ajuste adequado para a curva de Weibull a mesma é traçada em um histograma para uma melhor visualização, como mostrada na Figura 11.

Figura 11: Histograma da velocidade do vento com a curva de Weibull retirado do programa Acelio_wind.



Fonte: Do autor.

3.4 Curva de potência da turbina eólica

A potência elétrica gerada por uma turbina eólica varia com o cubo da velocidade do vento transversal a área varrida pelas pás da turbina, como mostra a equação 8.

$$P = \frac{1}{2} \cdot C_p \cdot \eta_m \cdot \eta_g \cdot A \cdot v^3 \quad (8)$$

Onde:

P = potência elétrica;

v = velocidade do vento;

A = área varrida pelas pás;

η_m = é o rendimento mecânico da caixa de transmissão;

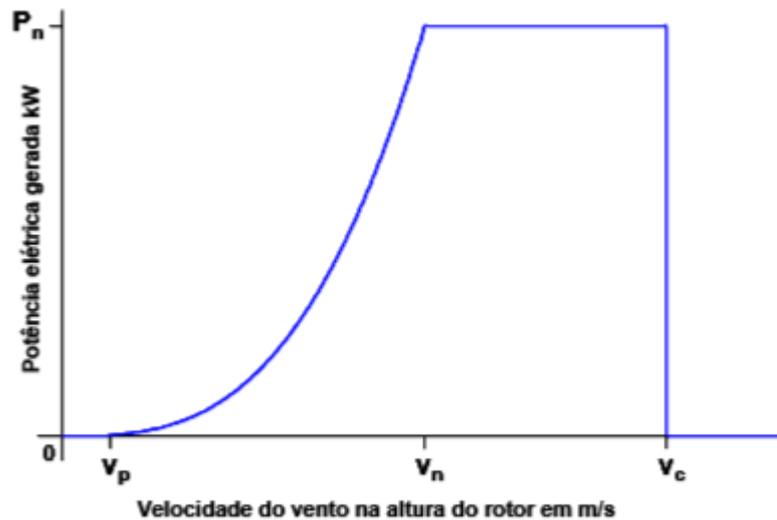
η_g = é o rendimento do gerador;

C_p = é o coeficiente de potência da turbina.

O coeficiente de potência da turbina expressa a parcela percentual (%) da potência disponível no vento que é extraída pela turbina.

A relação entre a potência elétrica gerada pela turbina e a velocidade do vento é normalmente apresentada na forma da curva de potência ilustrada na Figura 12.

Figura 12: Curva de potência de turbinas eólicas



Fonte: XLIISBPO Bento Gonçalves.

3.5 Fator de capacidade

Há vários modelos de turbinas disponíveis no mercado, cada uma com suas próprias especificações, altura e velocidades características. Portanto, na implantação de um aproveitamento eólico é fundamental identificar a turbina eólica cujas velocidades características v_p , v_n e v_c proporcionem a extração mais econômica da potência disponível no vento. Tradicionalmente, a escolha da turbina mais adequada baseia-se no fator de capacidade, definido pela razão entre a produção média da turbina e a sua produção potencial, se operada constantemente a plena capacidade (PESSANHA, 2010).

$$P(v) = \begin{cases} 0 & v < v_p \\ P = \frac{1}{2} \cdot Cp \cdot \eta_m \cdot \eta_g \cdot A \cdot v^3 & v_p \leq v < v_n \\ P = \frac{1}{2} \cdot Cp \cdot \eta_m \cdot \eta_g \cdot A \cdot v_n^3 & v_n \leq v < v_c \\ 0 & v > v_c \end{cases} \quad (9)$$

Onde:

v_p = velocidade de partida;

v_n = velocidade nominal;

v_c = velocidade de corte.

Assim, o valor da potência gerada em uma turbina eólica pode ser definida pela equação (10).

$$P_{média} = \int_0^{\infty} P(V) \cdot f(V) dV \quad (10)$$

Onde $f(V)$ é a função de distribuição de Weibull.

Considerando os limites de velocidade da curva de potência da turbina eólica temos a equação (11) abaixo.

$$P_{média} = \int_{v_p}^{v_n} \frac{1}{2} \cdot Cp \cdot \eta_m \cdot \eta_g \cdot \rho \cdot A \cdot v^3 \cdot f(V) \cdot dV + \int_{v_n}^{v_c} \frac{1}{2} \cdot Cp \cdot \eta_m \cdot \eta_g \cdot \rho \cdot A \cdot v_n^3 \cdot f(V) dV \quad (11)$$

A potência é máxima quando a velocidade do vento alcança a velocidade nominal v_n , assim para o cálculo da potência máxima fica elucidado pela equação (12).

$$P_{média} = \frac{1}{2} \cdot Cp \cdot \eta_m \cdot \eta_g \cdot \rho \cdot A \cdot v_n^3 \quad (12)$$

O fator de capacidade (FC) é definido pela razão entre a potência média e a potência máxima, como mostra a equação (13) a seguir.

$$FC = \frac{P_{m\acute{e}dia}}{P_{m\acute{a}xima}} \quad (13)$$

Substituindo as equações 11 e 12 em 13 obtemos a seguinte equação:

$$FC = \frac{1}{v_n^3} \int_{v_p}^{v_m} V^3 f(V) dV + \int_{v_n}^{v_c} f(V) dV \quad (14)$$

É importante frisar que o fator de capacidade depende apenas das velocidades características da curva de potência da turbina eólica (v_p , v_n e v_c) e da distribuição de Weibull da velocidade do vento na altura do rotor da turbina.

A energia gerada em kWh ao longo de um ano pode ser estimada com base na equação (15), elucidada a seguir.

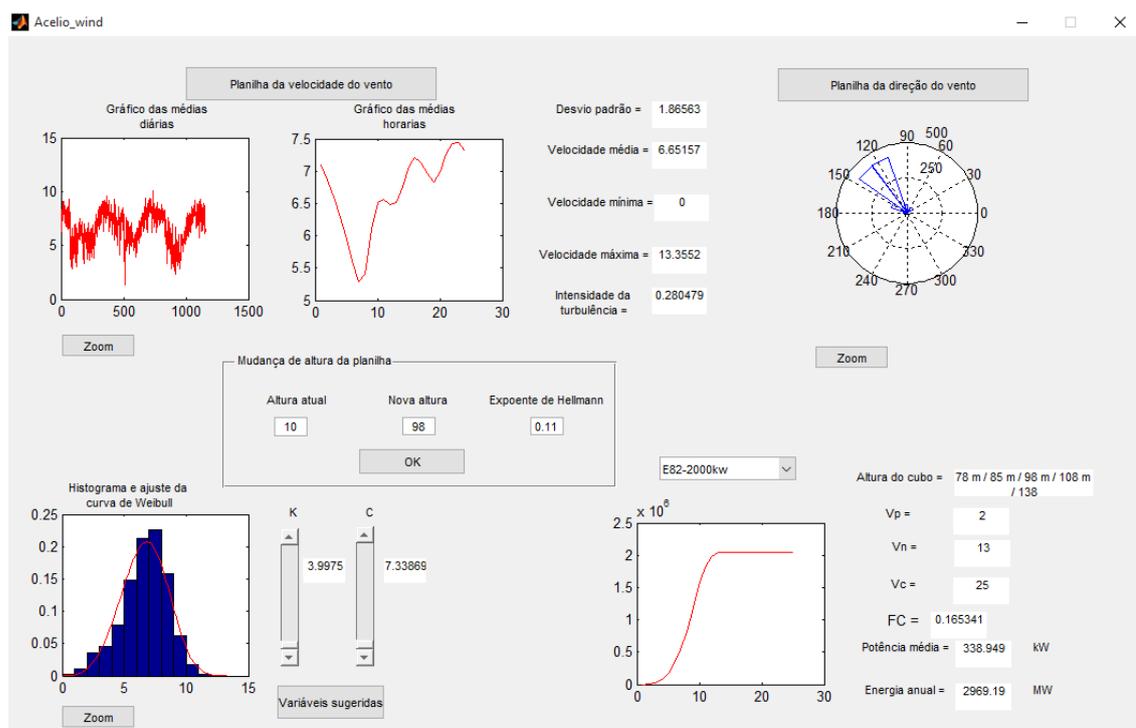
$$Energia = FC \cdot P_{m\acute{a}xima} \cdot 8760 \quad (15)$$

4 PROGRAMA ACELIO_WIND

4.1. Consideração inicial

Através dos dados apresentados no capítulo anterior, foi desenvolvido um programa pela interface gráfica do software Matlab “GUI” para auxiliar os cálculos do local a ser implantado o aerogerador, a Figura 13 mostra o programa batizado de Acelio_wind.

Figura 13 – Programa Acelio_wind.



Fonte: Do autor.

4.2. Entrada de dados

O programa tem como entrada de dados a velocidade e a direção dos ventos do local a ser estudado. Todos os dados devem estar em um arquivo de formato.xls contendo apenas os dados a serem estudados. Os dados dos ventos devem estar no formado em que as colunas são os números de medições feitas durante um dia e as linhas a quantidade de dias. Para a planilha de direção, os dados devem estar em formato .xls contendo apenas os dados, assim como a velocidade a direção também deve estar da mesma forma no sentido de linhas e colunas.

Após a entrada dos dados da velocidade dos ventos o programa irá plotar o gráfico da velocidade média diária, a velocidade média horária, o histograma dos valores da planilha,

indicará a velocidade média da planilha, o valor do desvio padrão, a velocidade máxima e a velocidade mínima de toda a planilha. Esses dados estão elucidados em destaque de amarelo pela Figura 14.

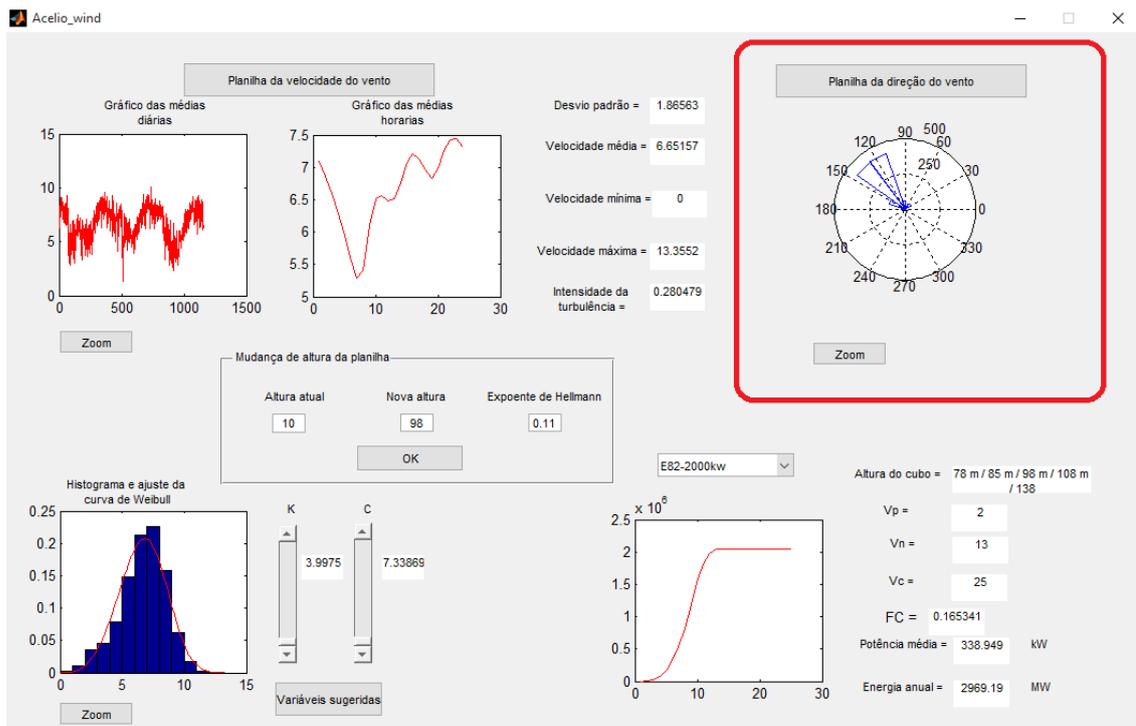
Figura 14 – Programa Acelio_wind após a entrada dos dados da velocidade dos ventos.



Fonte: Do autor.

Para a entrada dos dados da direção dos ventos o programa irá plotar uma rosa dos ventos para uma melhor visualização da direção do mesmo, mostrado em destaque de vermelho pela Figura 15.

Figura 15: Gráfico da rosa dos ventos no programa Acelio_wind.

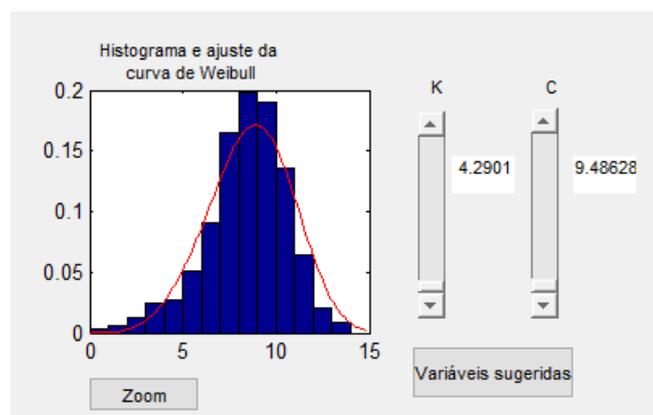


Fonte: Do autor.

4.3 Ajuste da Distribuição de Weibull

Após a entrada dos dados podemos ajustar a curva de Weibull de duas formas no programa, a primeira através dos dois *sliders* de ajuste de valores uma para o valor do fator de forma “*k*” e outro para o fator de escala “*c*” e a segunda é pelo botão abaixo dos *sliders* do qual irá calcular esses valores através da formula da velocidade média e desvio padrão, a Figura 16 mostra o ajuste no programa Acelio_wind.

Figura 16: Ajuste da curva de Weibull no programa Acelio_wind.



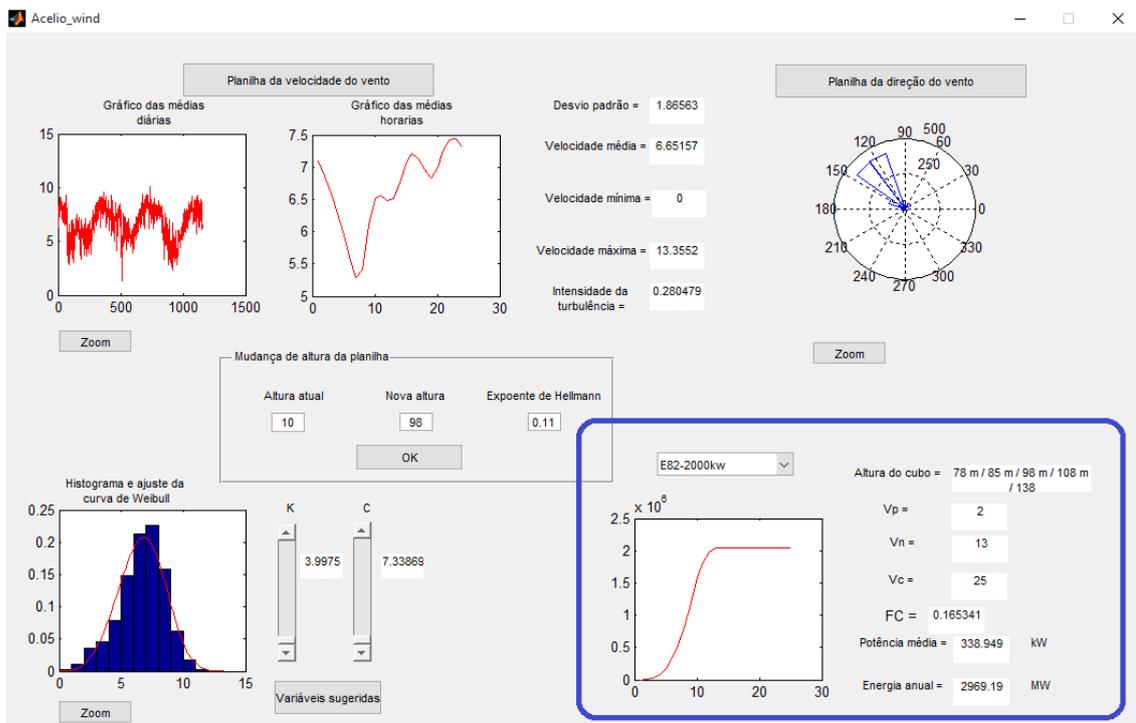
Fonte: Do autor.

As formulas na parte de variáveis sugeridas foram elucidadas no capítulo anterior utilizando a equação 5 para o fator de forma “ k ” e a equação 6 para o fator de escala “ c ” assim automaticamente ao apertar o botão os respectivos valores serão utilizados.

4.4 Seleção da turbina eólica

Com todos os ajustes prontos, podemos escolher a turbina eólica mais adequada para o lugar estudado. O programa conta com uma lista com 10 turbinas eólicas e ao escolher uma delas o mesmo irá mostrar os resultados da performance esperada para aquela região. Ao escolher a turbina eólica o programa irá mostrar os seguintes dados; curva de potência, altura em que o cubo da turbina deve ser instalada, velocidade de partida “ V_p ”, velocidade nominal “ V_n ”, velocidade de corte “ V_c ”, potência média, energia anual e o fator de capacidade “ FC ”. A Figura 17 mostra em azul todos esses valores no programa.

Figura 17: Valores de performance do aerogerador.



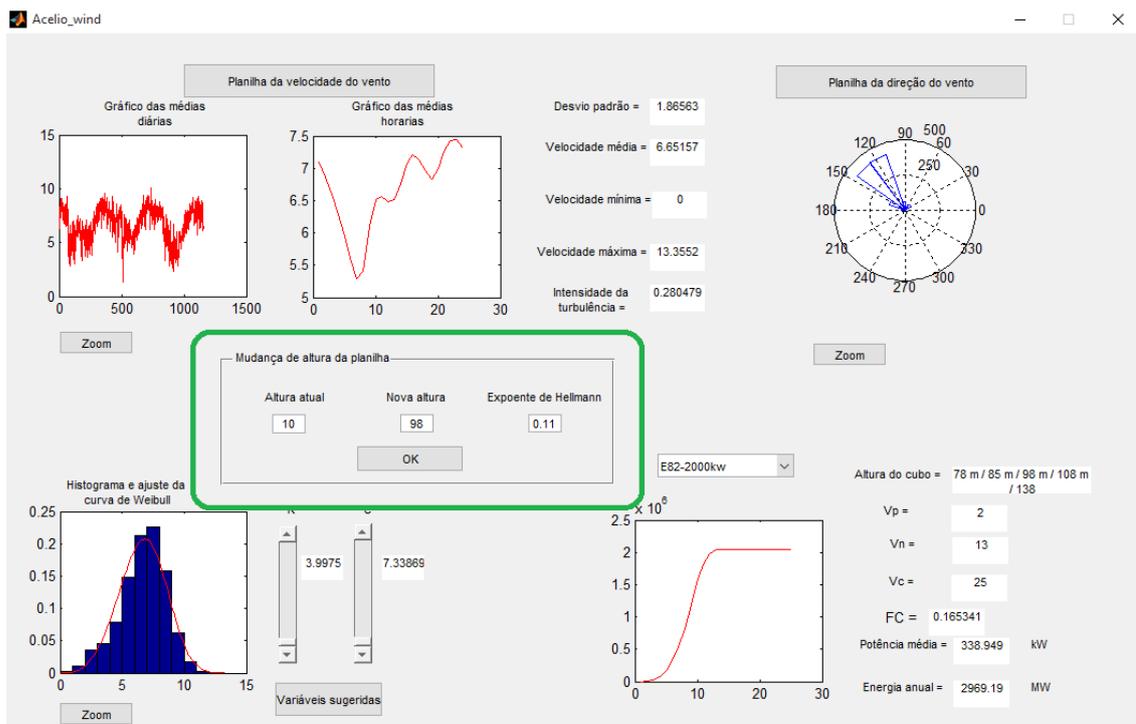
Fonte: Do autor.

A partir dos valores mostrados podemos escolher a turbina mais adequada a região que deve ser estudada, escolhendo principalmente uma turbina que demonstre um fator de capacidade “ FC ” elevado, assim, gerando uma energia anual maior.

4.5 Mudança de altura na planilha

Na parte central do programa podemos mudar a altura da planilha para uma desejada desde que o usuário possua os dados da altura em que a velocidade do vento foi medida e o valor do coeficiente de Hellman que foi elucidado no capítulo anterior através da equação 2. A região designada para mudança de altura no programa pode ser vista pela Figura 18 abaixo.

Figura 18: Local para mudança da altura da planilha.



Fonte: Do autor.

Com todos os dados inseridos no programa o usuário deve apertar o botão “OK”. Com isso, todos os dados iniciais serão atualizados de forma automática para os valores da altura desejada, porém, o ajuste da curva de Weibull deve ser feita novamente, assim como a escolha da turbina para que ocorra a atualização dos valores da mesma.

É importante frisar que os dados inseridos no programa já devem estar tratados de forma correta pois o mesmo não possui função para correção das planilhas.

5 ANALISE DOS DADOS DAS REGIÕES

A partir das informações elucidadas nos capítulos anteriores, utilizaremos o programa Acelio_wind para simular a implantação de 20 turbinas eólicas em três regiões, sendo elas Sobral, Icapuí e Ubajara. As regiões estudadas possuem diferentes valores de rugosidade. Assim, poderemos observar diferentes valores para o expoente de Hellman e observar a mudança dos valores em diferentes regiões, seguindo os valores do mesmo para os valores elucidados pela Tabela 7 no capítulo 3.

Todas as planilhas usadas para o estudo foram disponibilizadas pela FUNCEME, onde nelas contém os dados da direção do vento, precipitação pluviométrica, pressão atmosférica, radiação incidente total, temperatura do ar, umidade relativa do ar e a velocidade do vento, porém, utilizaremos apenas os valores da direção do vento e da velocidade do vento nesse estudo. Todos os dados foram medidos a 10 metros de altura.

Para todas as regiões utilizaremos três turbinas eólicas da empresa ENERCON, sendo elas E33-330kW, E82-2000kW e E126-7500kW.

5.1 Cidade de Sobral

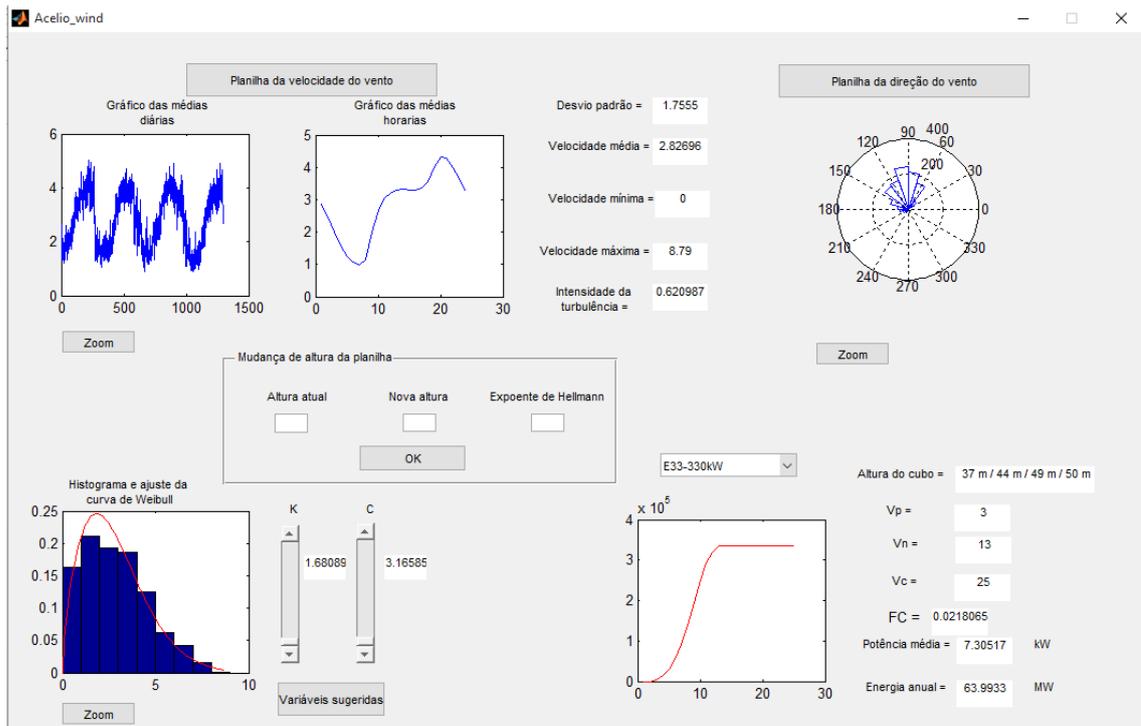
A primeira região a ser estudada será a cidade de Sobral localizada no interior da região norte do Ceará, para essa região devemos utilizar o valor para o expoente de Hellman de 0,3, pois, o mesmo se localiza numa região com um grau de rugosidade elevado.

Os dados utilizados para essa região foram obtidos pela FUNCEME através de medições com intervalos de uma hora, totalizando vinte e quatro medições diárias, durante o período de 24/04/2003 até 31/12/2006, gerando 31.080 medições.

5.1.1 Simulação para a turbina E33-330kW

Para a primeira simulação utilizaremos a turbina E33-330kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 19.

Figura 19: Simulação em uma altura de 10m para a cidade de Sobral, para uma turbina E33-330kW.

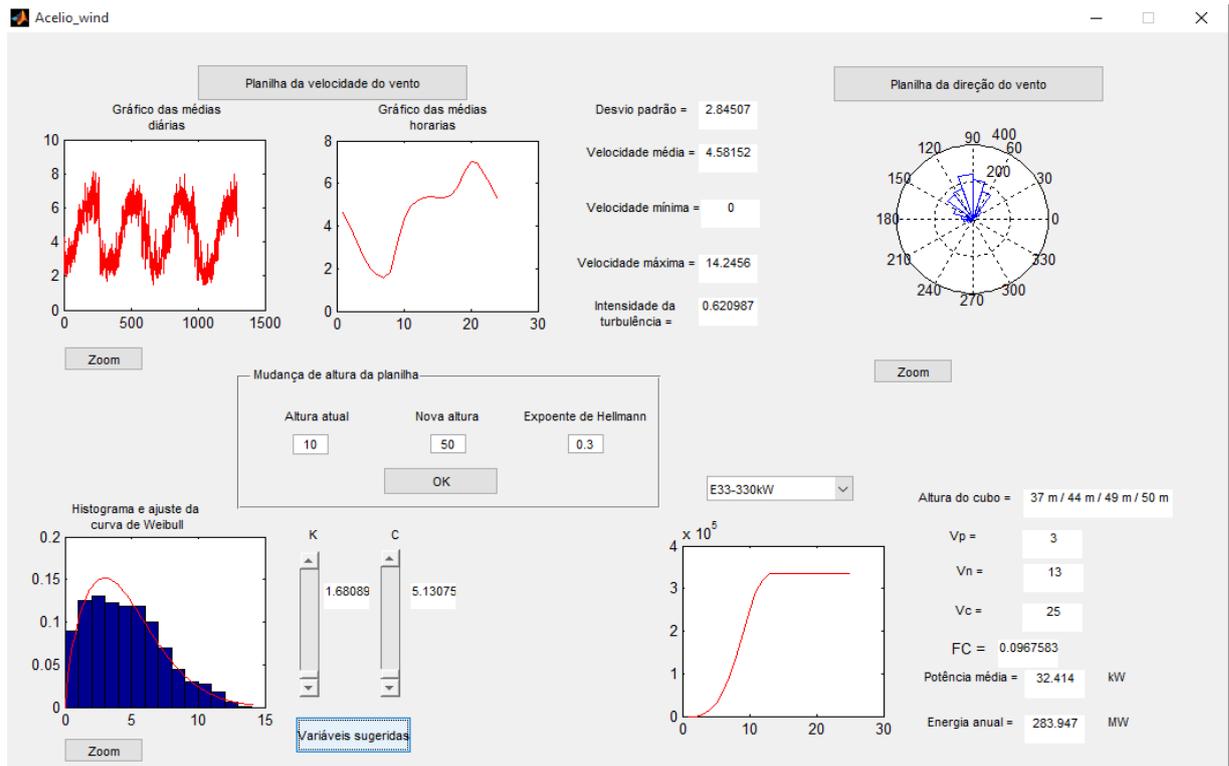


Fonte: Do autor.

Com a entrada dos dados podemos observar que a turbina deve ser instalada a uma altura entre 37 a 50 metros e que a altura da planilha deve ser corrigida, então utilizaremos a altura de 50 metros para a correção com um valor do expoente de Hellman de 0,3 já que a região se encontra como uma pequena cidade descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 20.

Figura 20: Simulação em uma altura de 50m com expoente de Hellman de 0,3 para a cidade de Sobral para a turbina eólica E33-330 kW.



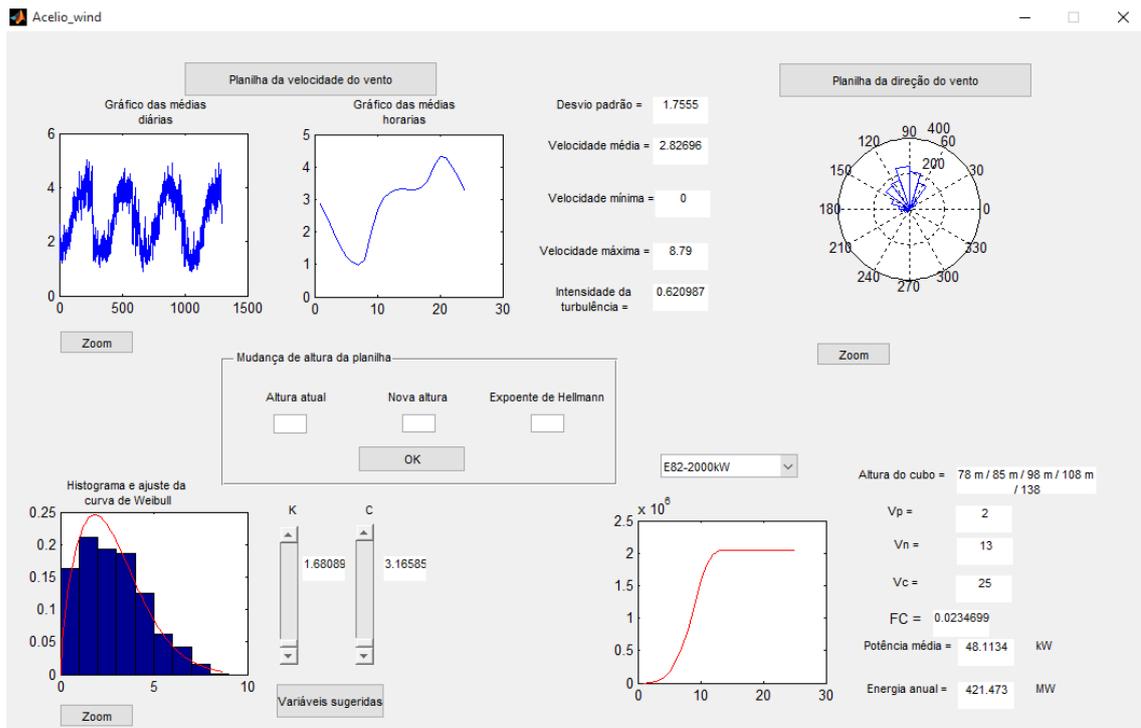
Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha todos os dados foram atualizados para um desvio padrão de 2,84507, uma velocidade média de aproximadamente 4,58, uma potência média de 32,414kW e uma energia anual de 283,947kW. Porém, mesmo com a correção dos valores o fator de capacidade (FC) continua com um valor extremamente baixo, já que antes o valor do mesmo era de aproximadamente 0,022 e passou para aproximadamente 0,097.

5.1.2 Simulação para a turbina E82-2000kW

Para a primeira simulação, utilizaremos a turbina E82-2000kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 21.

Figura 21: Simulação em uma altura de 10m para a cidade de Sobral, para uma turbina E82-2000kW.

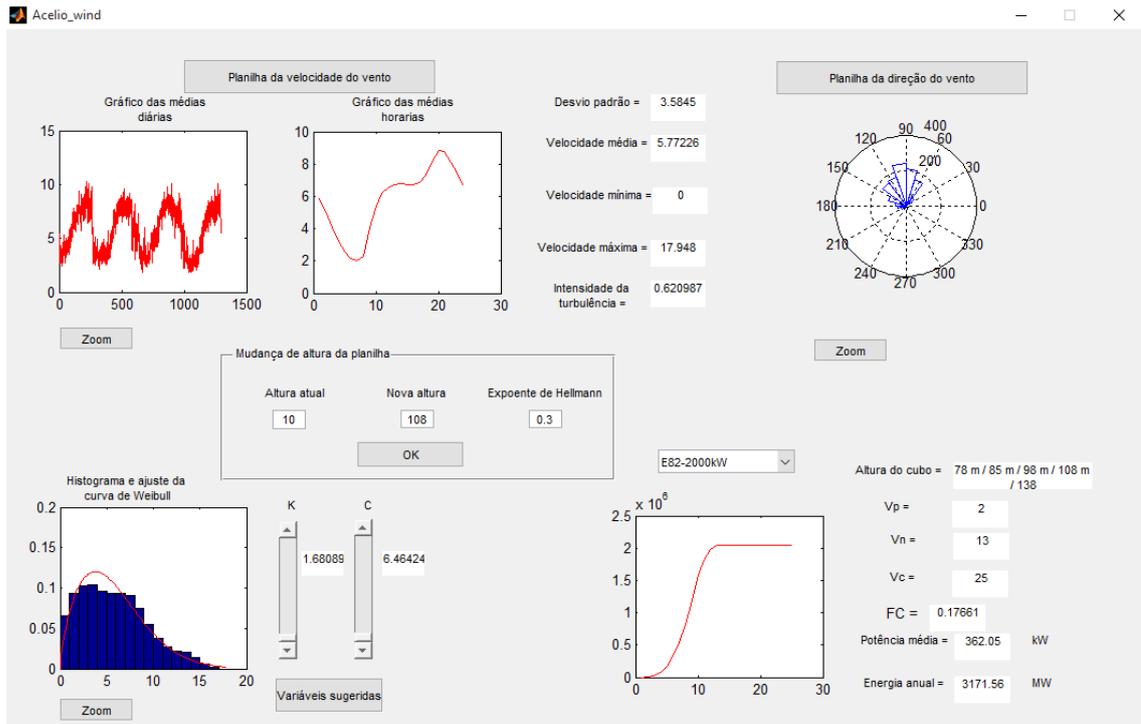


Fonte: Do autor.

Com a entrada dos dados, podemos observar que a turbina deve ser instalada a uma altura entre 78 a 138 metros e que a altura da planilha deve ser corrigida, então utilizaremos a altura de 108 metros para a correção com um valor do expoente de Hellman de 0,3 já que a região se encontra como uma pequena cidade descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 22.

Figura 22: Simulação em uma altura de 108 m com expoente de Hellman de 0,3 para a cidade de Sobral para a turbina eólica E33-330 kW.



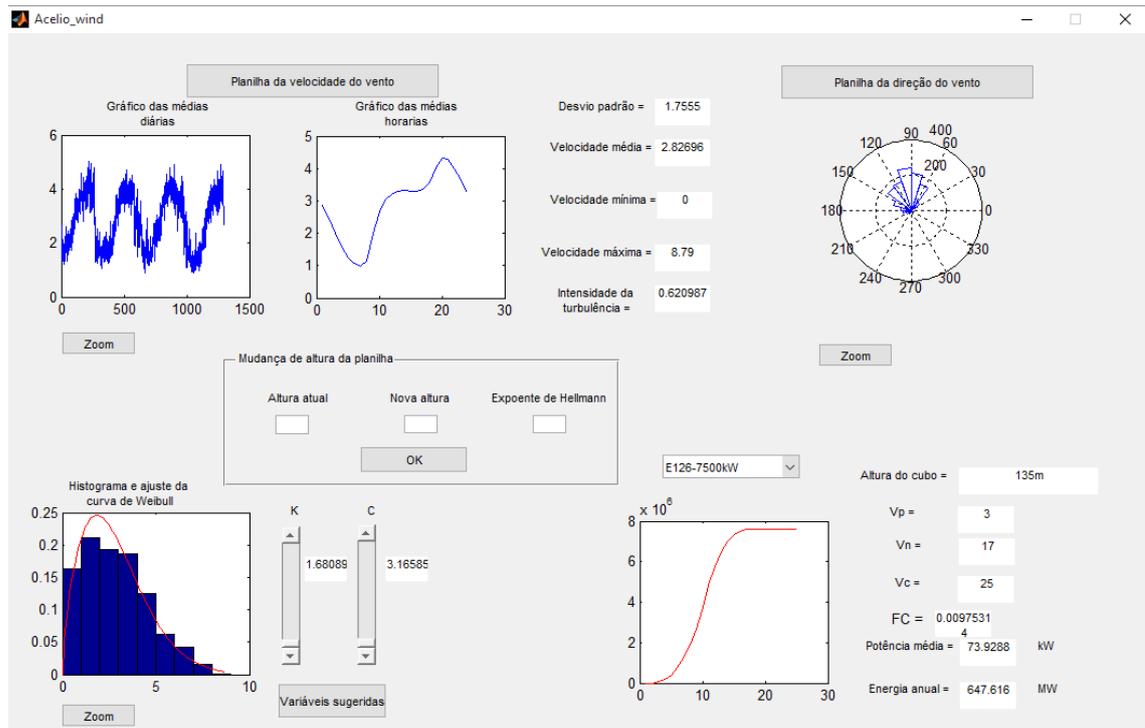
Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha todos os dados foram atualizados para um desvio padrão de 3,5845, uma velocidade média de aproximadamente 5,77, uma potência média de 362,05kW e uma energia anual de 3.171,56kW. Porém, mesmo com a correção dos valores, o fator de capacidade (FC) continua com um valor extremamente baixo, já que antes o valor do mesmo era de aproximadamente 0,0235 e passou para aproximadamente 0,18.

5.1.3 Simulação para a turbina E126-7500kW

Para a primeira simulação, utilizaremos a turbina E126-7500kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 23.

Figura 23: Simulação em uma altura de 10m para a cidade de Sobral, para uma turbina E126-7500kW.

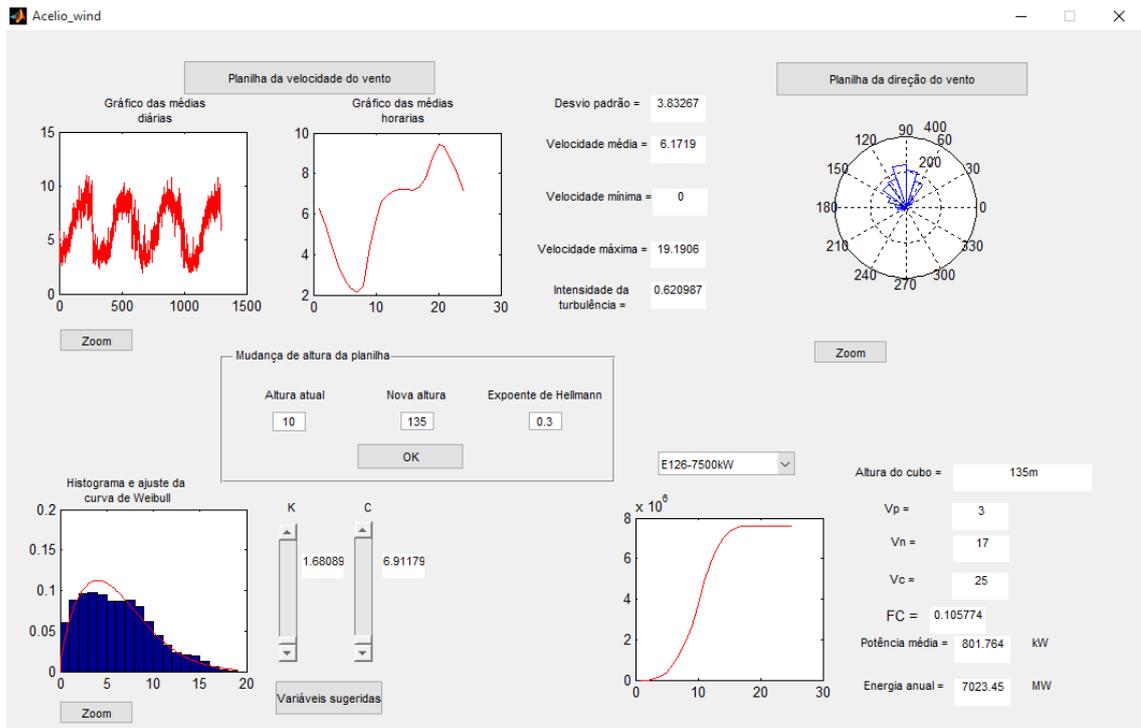


Fonte: Do autor.

Com a entrada dos dados podemos observar que a turbina deve ser instalada a uma altura de 135 metros e que a altura da planilha deve ser corrigida. Então, utilizaremos a correção para um valor do expoente de Hellman de 0,3 já que a região se encontra como uma pequena cidade descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 24.

Figura 24: Simulação em uma altura de 135 m com expoente de Hellman de 0,3 para a cidade de Sobral para a turbina eólica E33-330 kW.



Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha todos os dados foram atualizados para um desvio padrão de 3,83267, uma velocidade média de aproximadamente 6,17, uma potência média de 801,764 kW e uma energia anual de 7.023,45 kW. Porém, mesmo com a correção dos valores o fator de capacidade (FC) continua com um valor extremamente baixo, já que antes o valor do mesmo era de aproximadamente 0,0098 e passou para aproximadamente 0,1058.

5.2 Cidade de Icapuí

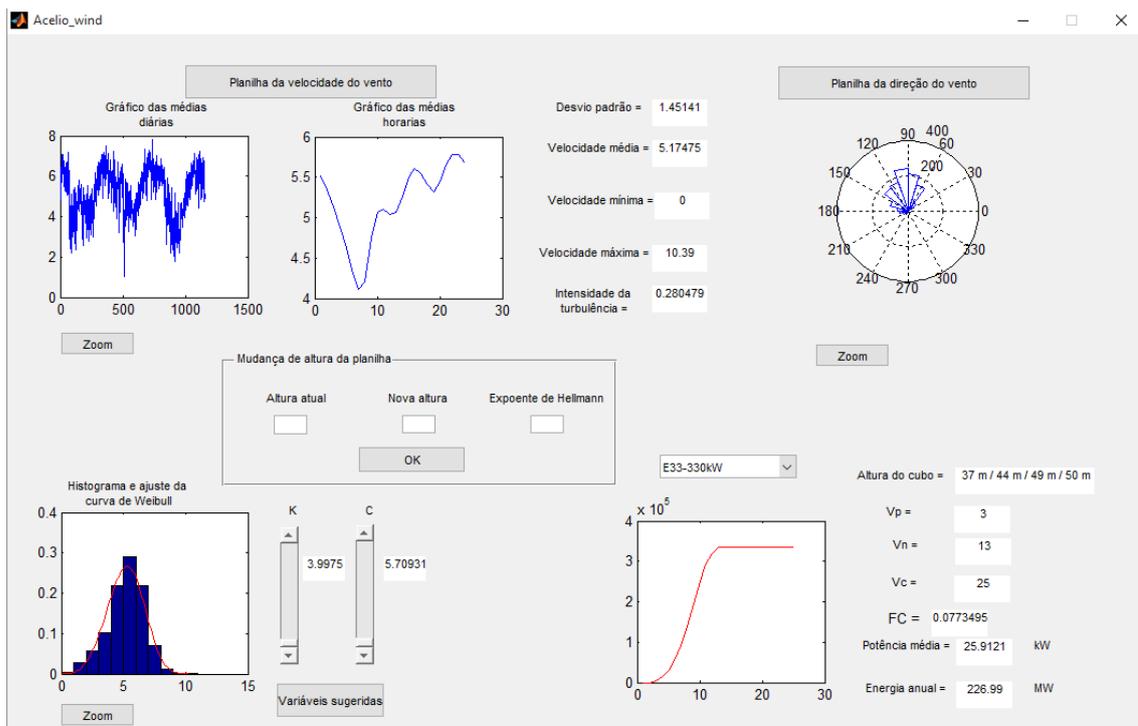
A segunda região a ser estudada será a cidade de Icapuí, localizada no interior da região norte do Ceará. Para essa região devemos utilizar o valor para o expoente de Hellman de 0,25, pois, a mesma se localiza numa região com um grau de rugosidade mediano.

Os dados utilizados para essa região foram obtidos pela FUNCEME através de medições com intervalos de uma hora, totalizando vinte e quatro medições diárias, durante o período de 01/11/2003 até 31/12/2006, gerando 27.768 medições.

5.2.1 Simulação para a turbina E33-330kW

Para a primeira simulação, utilizaremos a turbina E33-330kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 25.

Figura 25: Simulação em uma altura de 10m para a cidade de Icapuí, para uma turbina E33-330kW.

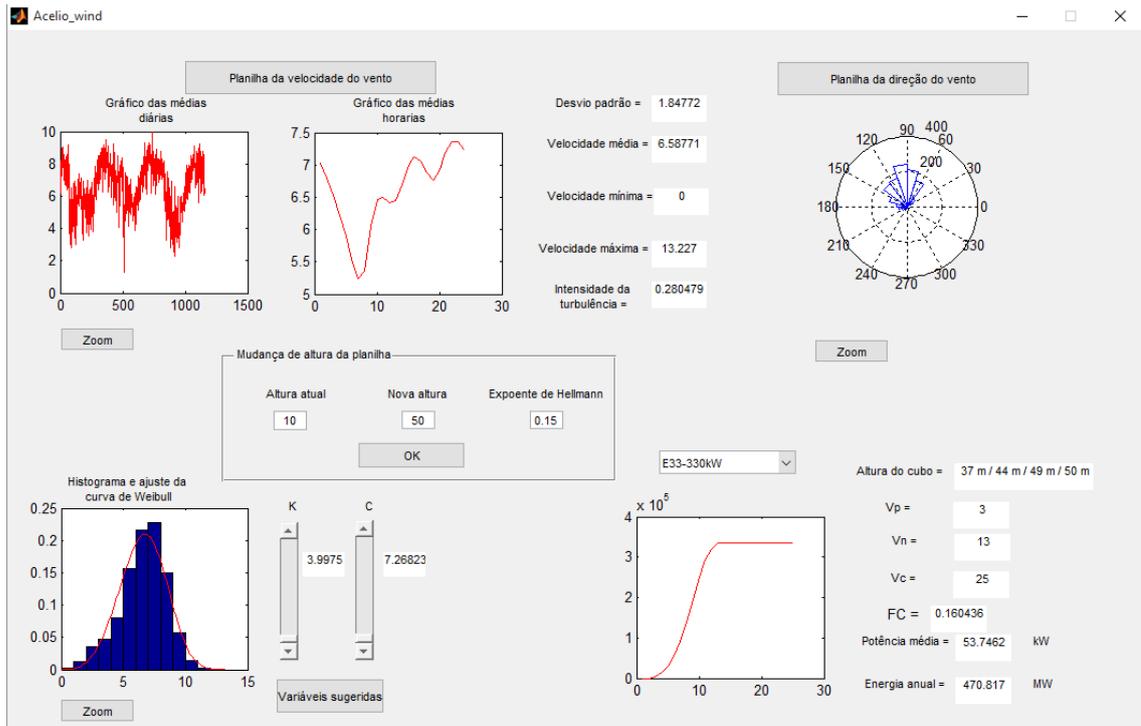


Fonte: Do autor.

Com a entrada dos dados, podemos observar que a turbina deve ser instalada a uma altura entre 37 a 50 metros e que a altura da planilha deve ser corrigida, então utilizaremos a altura de 50 metros para a correção com um valor do expoente de Hellman de 0,15 já que a região se encontra no litoral e usaremos a descrição de grama alta ao nível do solo descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 26.

Figura 26: Simulação em uma altura de 50m com expoente de Hellman de 0,15 para a cidade de Icapuí para a turbina eólica E33-330 kW.



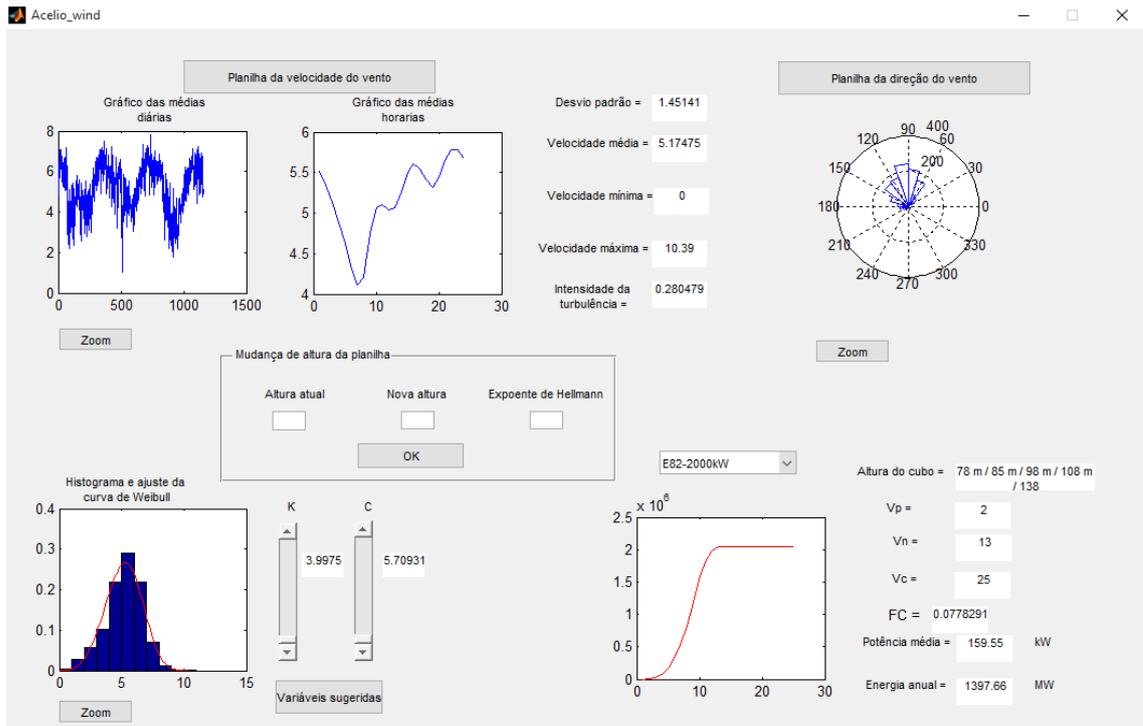
Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha todos os dados foram atualizados como o desvio padrão para 1,84772, uma velocidade média de aproximadamente 6,59, uma potência média de 53,746kW e uma energia anual de 470,817kW. Porém, mesmo com a correção dos valores, o fator de capacidade (FC) continua com um valor baixo, já que antes o valor do mesmo era de aproximadamente 0,0773 e passou para aproximadamente 0,1604.

5.2.2 Simulação para a turbina E82-2000kW

Para a primeira simulação utilizaremos a turbina E82-2000kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 27.

Figura 27: Simulação em uma altura de 10m para a cidade de Icapuí, para uma turbina E82-2000kW.

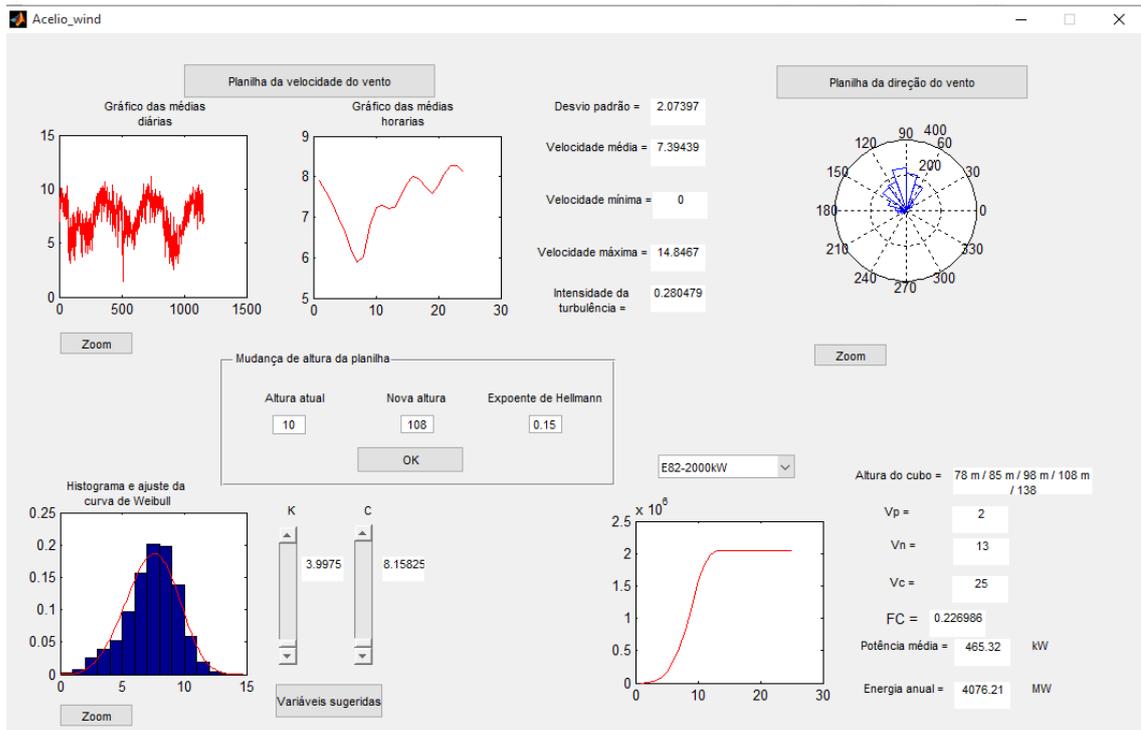


Fonte: Do autor.

Com a entrada dos dados, podemos observar que a turbina deve ser instalada a uma altura entre 78 a 138 metros e que a altura da planilha deve ser corrigida, então utilizaremos a altura de 108 metros para a correção com um valor do expoente de Hellman de 0,15 já que a região se encontra no litoral e usaremos a descrição de grama alta ao nível do solo descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 28.

Figura 28: Simulação em uma altura de 108 m com expoente de Hellman de 0,15 para a cidade de Icapuí para a turbina eólica E82-2000 kW.



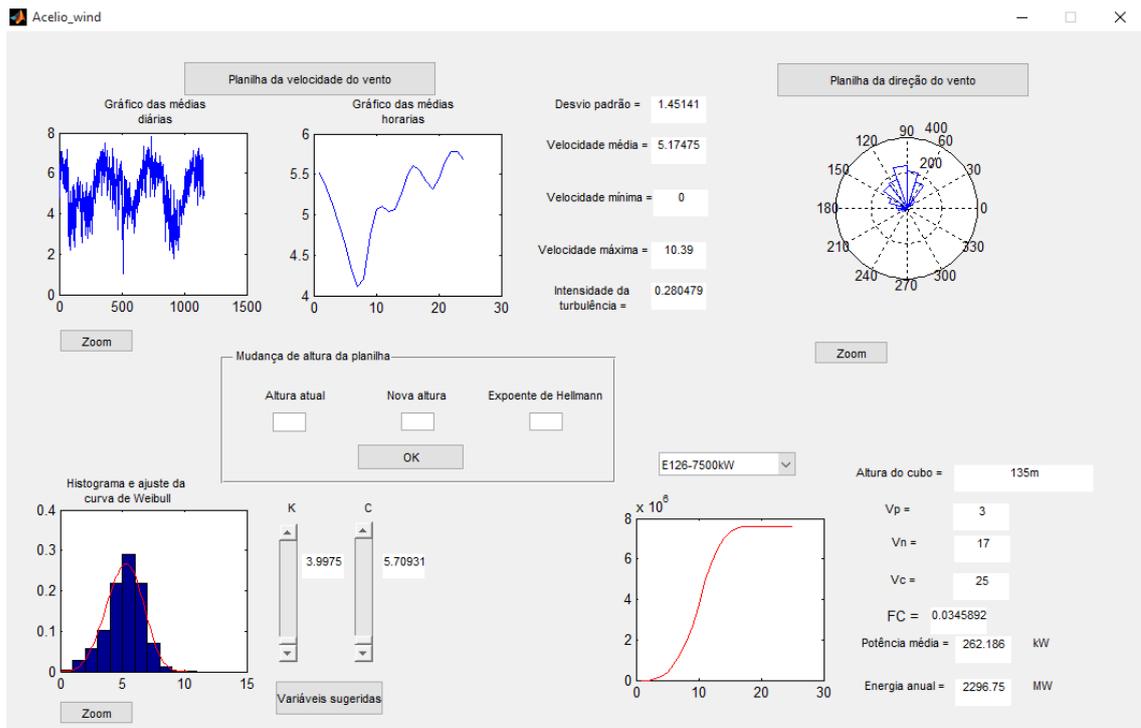
Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha todos os dados foram atualizados para um desvio padrão de 2,07397, uma velocidade média de aproximadamente 7,39, uma potência média de 465,32 kW e uma energia anual de 4.076,21 kW. Porém, mesmo com a correção dos valores o fator de capacidade (FC) continua com um valor baixo, já que antes o valor do mesmo era de aproximadamente 0,0778 e passou para aproximadamente 0,227.

5.2.3 Simulação para a turbina E126-7500kW

Para a primeira simulação utilizaremos a turbina E126-7500kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 29.

Figura 29: Simulação em uma altura de 10m para a cidade de Icapuí, para uma turbina E126-7500kW.

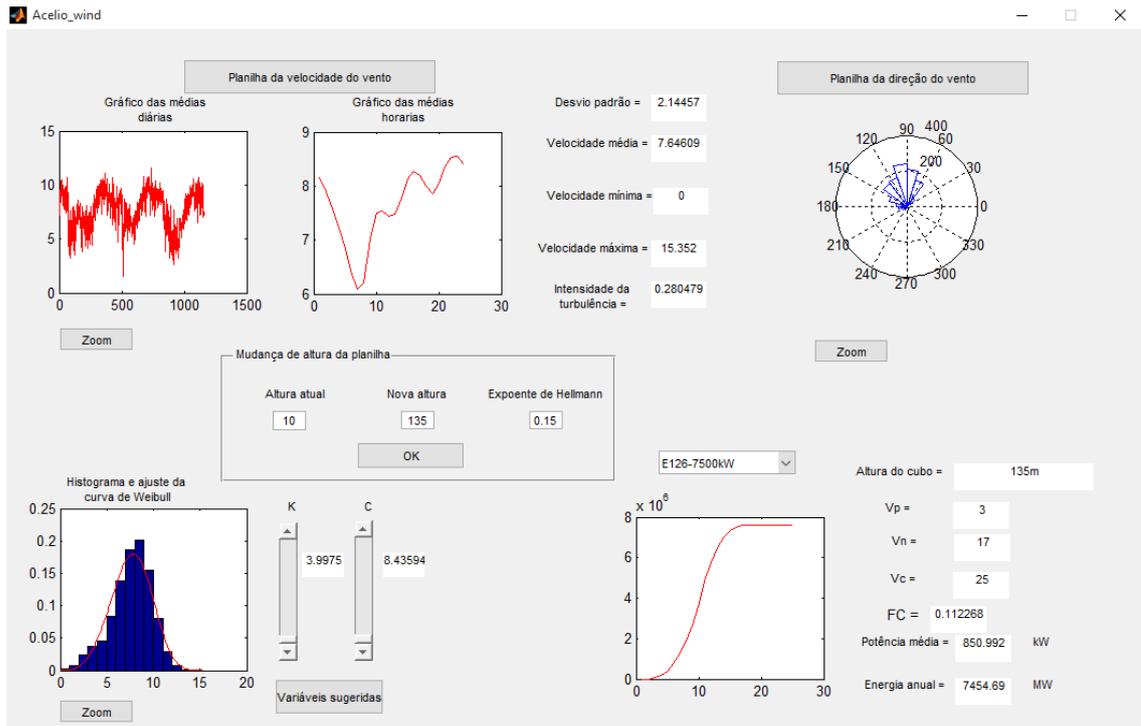


Fonte: Do autor

Com a entrada dos dados, podemos observar que a turbina deve ser instalada a uma altura de 135 metros e que a altura da planilha deve ser corrigida, então utilizaremos a correção para um valor do expoente de Hellman de 0,15 já que a região se encontra no litoral e usaremos a descrição de grama alta ao nível do solo descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 30 abaixo:

Figura 30: Simulação em uma altura de 135 m com expoente de Hellman de 0,15 para a cidade de Icapuí para a turbina eólica E126-7500 kW.



Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha todos os dados foram atualizados para um desvio padrão de 2,145, uma velocidade média de aproximadamente 7,65, uma potência média de 850,992 kW e uma energia anual de 7.454,69 kW. Porém, mesmo com a correção dos valores o fator de capacidade (FC) continua com um valor extremamente baixo, já que antes o valor do mesmo era de aproximadamente 0,0346 e passou para aproximadamente 0,1123.

5.3 Cidade de Ubajara

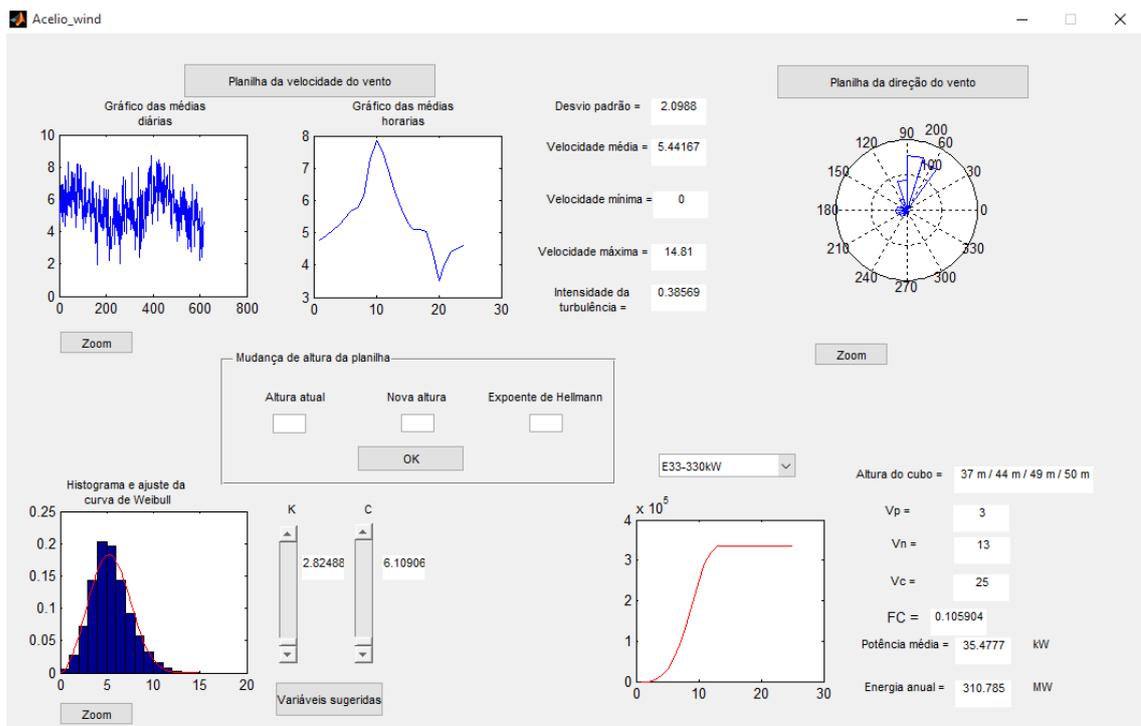
A segunda região a ser estudada será a cidade de Ubajara localizada na microrregião da Ibiapaba no noroeste do Ceará, para essa região devemos utilizar o valor para o expoente de Hellman de 0,25, pois, a mesma se localiza numa região com um grau de rugosidade mediano.

Os dados utilizados para essa região foram obtidos pela FUNCEME através de medições com intervalos de uma hora, totalizando vinte e quatro medições diárias, durante o período de 16/08/2004 até 24/04/2006, gerando 14.808 medições.

5.3.1 Simulação para a turbina E33-330kW

Para a primeira simulação utilizaremos a turbina E33-330kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 31.

Figura 31: Simulação em uma altura de 10m para a cidade de Ubajara, para uma turbina E33-330kW

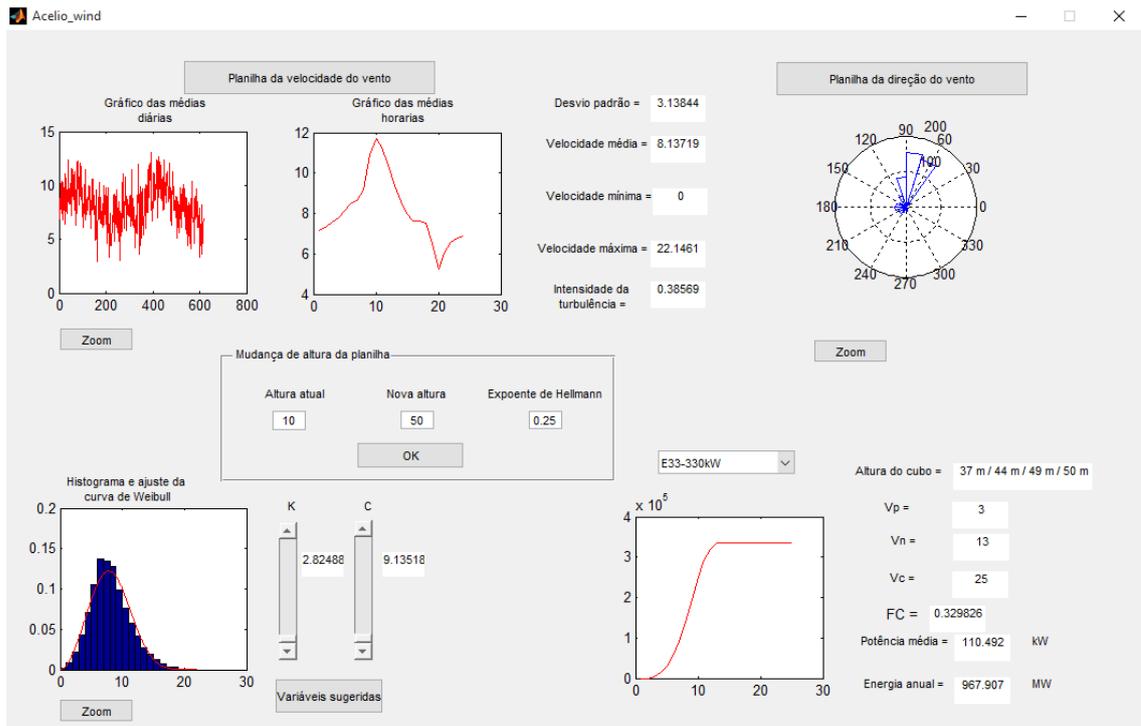


Fonte: Do autor.

Com a entrada dos dados podemos observar que a turbina deve ser instalada a uma altura entre 37 a 50 metros e que a altura da planilha deve ser corrigida, então utilizaremos a altura de 50 metros para a correção com um valor do expoente de Hellman de 0,25 já que a região se encontra numa região serrana e se enquadra na região de áreas rurais com muitas árvores descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 32:

Figura 32: Simulação em uma altura de 50m com expoente de Hellman de 0,25 para a cidade de Ubajara, para a turbina eólica E33-330 kW.



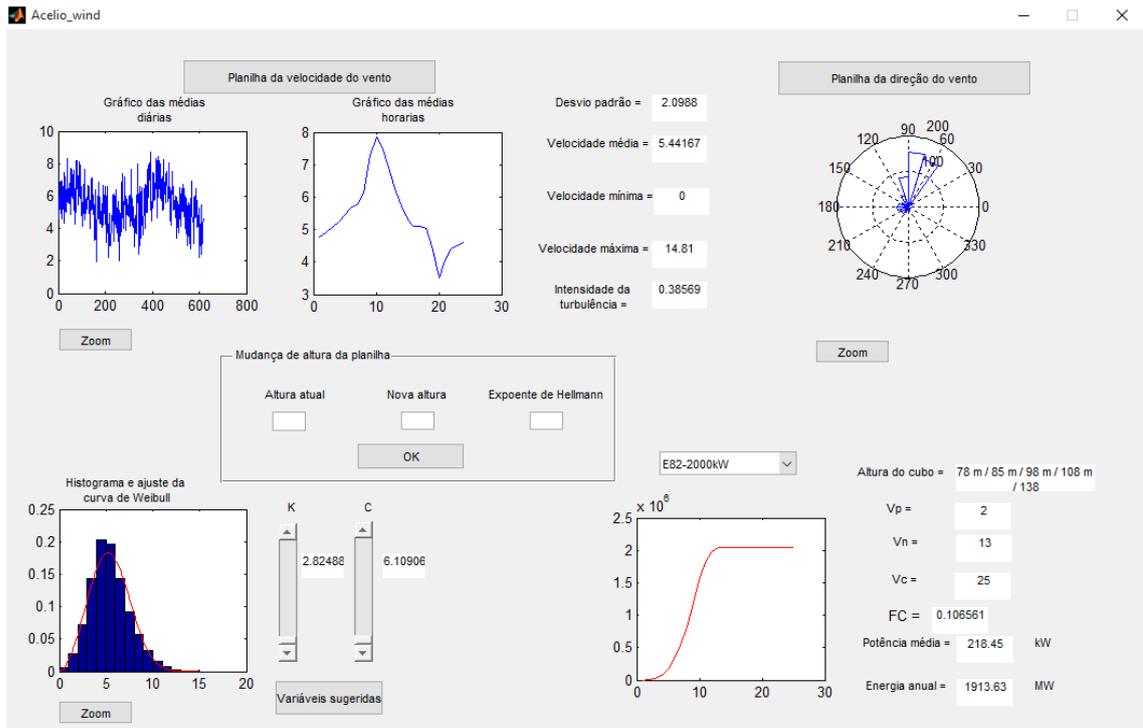
Fonte: Do autor.

Como podemos observar após a correção da altura da planilha todos os dados foram atualizados como o desvio padrão para 3,138, uma velocidade média de aproximadamente 8,14, uma potência média de 110,492 kW e uma energia anual de 967,907 kW. Para os valores mostrados anteriormente e com o fator de potência obtido de 0,3298 mostra uma relação satisfatória para uma altura de 50 m.

5.3.2 Simulação para a turbina E82-2000kW

Para a primeira simulação utilizaremos a turbina E82-2000kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 33.

Figura 33: Simulação em uma altura de 10m para a cidade de Ubajara, para uma turbina E82-2000kW.

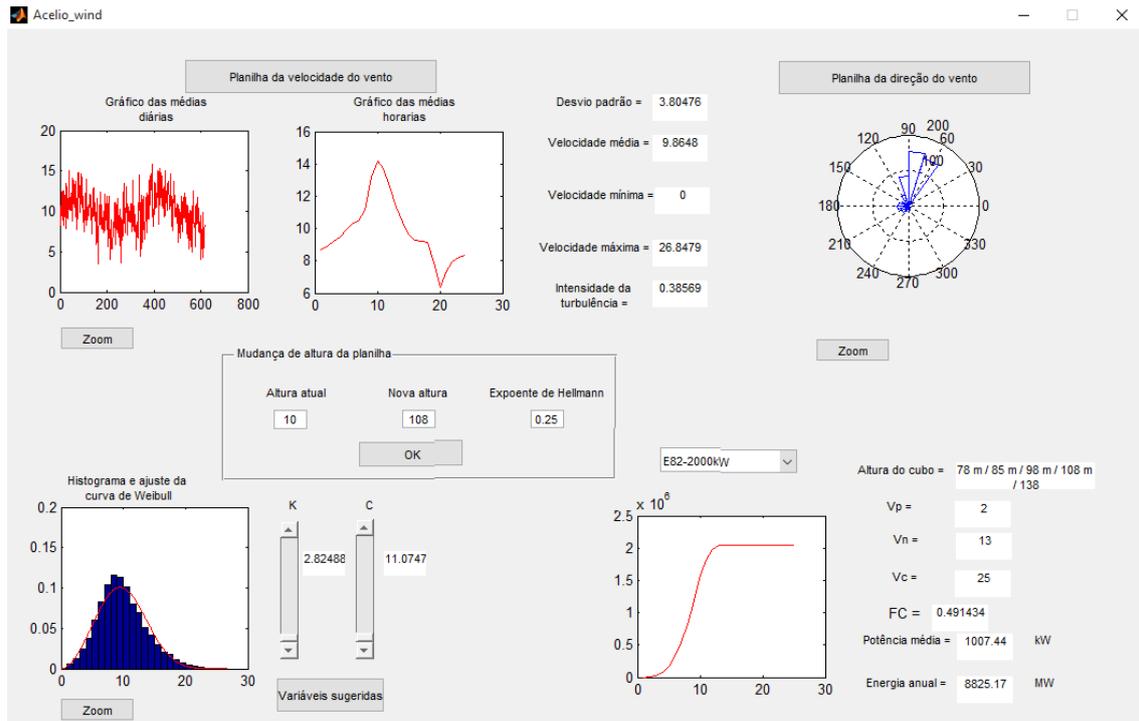


Fonte: Do autor.

Com a entrada dos dados, podemos observar que a turbina deve ser instalada a uma altura entre 78 a 138 metros e que a altura da planilha deve ser corrigida. Então, utilizaremos a altura de 108 metros para a correção com um valor do expoente de Hellman de 0,25 já que a região se encontra numa região serrana e se enquadra na região de áreas rurais com muitas árvores descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 34:

Figura 34: Simulação em uma altura de 108m com expoente de Hellman de 0,25 para a cidade de Ubajara, para a turbina eólica E82-2000 kW.



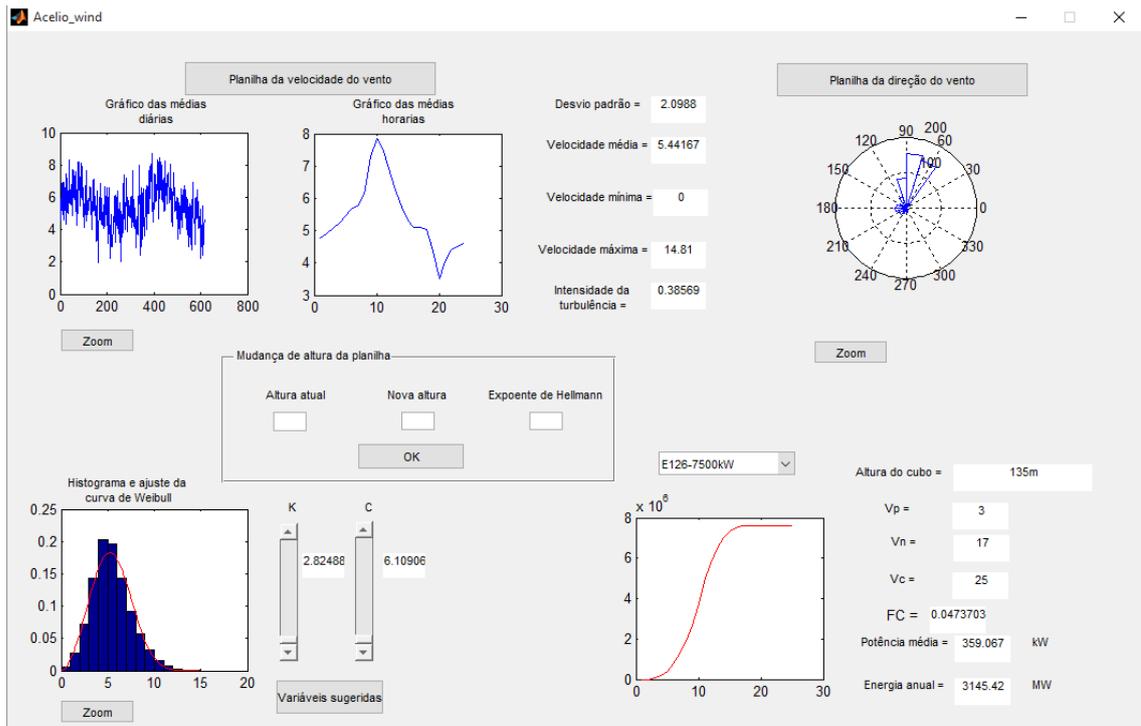
Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha todos os dados foram atualizados para um desvio padrão de 3,80476, uma velocidade média de aproximadamente 9,86, uma potência média de 1.007,44 kW e uma energia anual de 8.825,17 kW. Para os valores mostrados anteriormente e com o fator de potência obtido de 0,4914 mostra uma relação satisfatória para uma altura de 108 m.

5.3.3 Simulação para a turbina E126-7500kW

Para a primeira simulação utilizaremos a turbina E126-7500kW a uma altura de 10 metros, já que todos os valores iniciais foram disponibilizados nessa altura, como mostrado na Figura 35.

Figura 35: Simulação em uma altura de 10m para a cidade de Ubajara, para uma turbina E126-7500 kW.

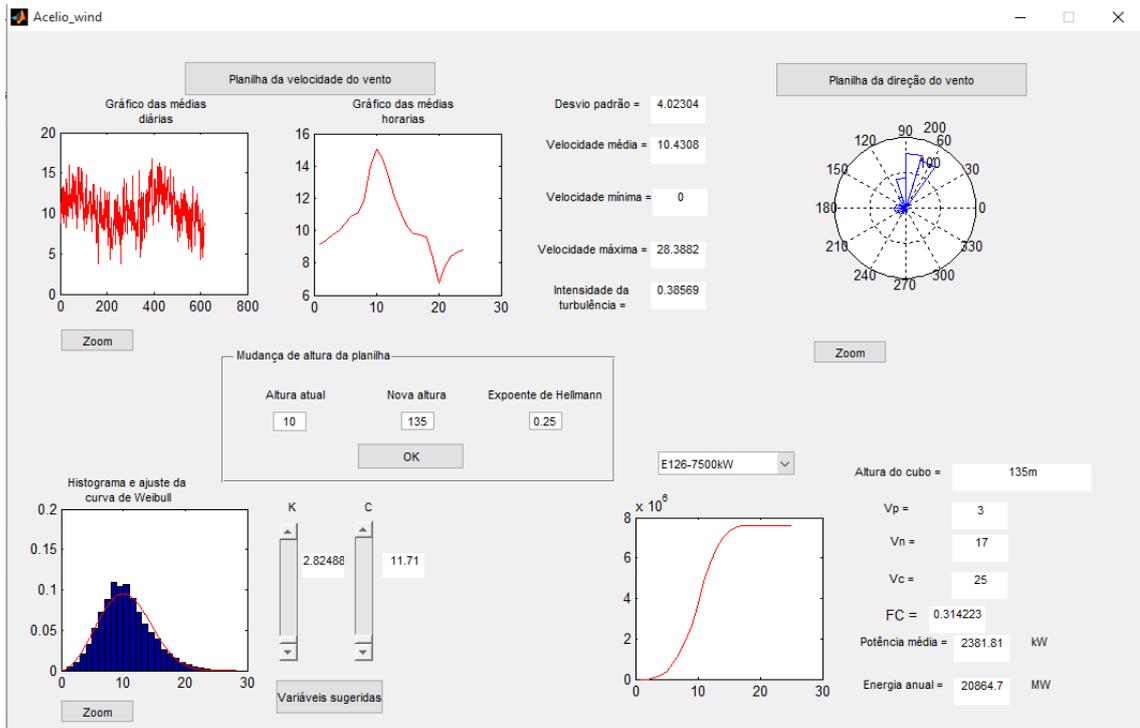


Fonte: Do autor.

Com a entrada dos dados podemos observar que a turbina deve ser instalada a uma altura de 135 metros e que a altura da planilha deve ser corrigida, então utilizaremos a correção para um valor do expoente de Hellman de 0,25 já que a região se encontra numa região serrana e se enquadra na região de áreas rurais com muitas árvores descrita na Tabela 7 no capítulo 3.

Utilizando o programa para a correção dos valores teremos os valores elucidados pela Figura 36:

Figura 36: Simulação em uma altura de 135m com expoente de Hellman de 0,25 para a cidade de Ubajara, para a turbina eólica E126-7500 kW.



Fonte: Do autor.

Como podemos observar, após a correção da altura da planilha, todos os dados foram atualizados para um desvio padrão de 4,02304, uma velocidade média de aproximadamente 10,43, uma potência média de 2.381,81 kW e uma energia anual de 20.864,7 kW. Para os valores mostrados anteriormente e com o fator de potência obtido de 0,3142 mostra uma relação quase satisfatória para uma altura de 135 m.

5.4 Comparação entre as regiões

Com as simulações anteriores, feitas entre os tópicos 5.1 a 5.3, podemos decidir qual a melhor região para a implantação do nosso parque fictício e analisarmos qual a melhor turbina eólica a ser utilizada.

Para uma melhor visualização da comparação dos dados da turbina E33-330 kW, foi criada a Tabela 7.

Tabela 7: dados comparativos das regiões para a turbina E33-330 kW a 50m.

Região	Desvio padrão	Velocidade média (m/s)	Potência média (kW)	Energia anual (MW)	Fator de Capacidade
Sobral	2,84507	4,58	32,414	283,947	0,097
Icapuí	1,84772	6,59	53,746	470,817	0,1604
Ubajara	3,138	8,14	110,492	967,907	0,3298

Fonte: Do autor.

Seguindo o mesmo princípio, foi desenvolvida a Tabela 8 e 9 para a comparação dos valores das turbinas E82-2000 kW e a E126-7500 kW respectivamente, elucidadas abaixo.

Tabela 8: dados comparativos das regiões para a turbina E82-2000 kW a 108m.

Região	Desvio padrão	Velocidade média (m/s)	Potência média (kW)	Energia anual (MW)	Fator de Capacidade
Sobral	3,5845	5,77	362,05	3.171,56	0,18
Icapuí	2,07397	7,39	465,32	4.076,21	0,227
Ubajara	3,80476	9,86	1.007,44	8.825,17	0,4914

Fonte: Do autor.

Tabela 9: dados comparativos das regiões para a turbina E126-7500 kW a 135m.

Região	Desvio padrão	Velocidade média (m/s)	Potência média (kW)	Energia anual (MW)	Fator de Capacidade
Sobral	3,83267	6,17	801,764	7.023,45	0,1058
Icapuí	2,145	7,65	850,992	7.454,69	0,1123
Ubajara	4,02304	10,43	2.381,81	20.864,70	0,3142

Fonte: Do autor.

6 Conclusão

A partir dos dados elucidados pelas Tabelas 7, 8 e 9, pode-se definir qual a melhor turbina e qual a melhor região, a ser instalado o parque eólico. Através do critério do fator de capacidade (FC), o melhor fator de capacidade estudado se encontra na Tabela 8, para a região de Ubajara, e a turbina eólica E82-2000 kW, o mesmo possui um valor de 0,4914.

Com a instalação de um parque eólico fictício de 20 (vinte) turbinas eólicas do modelo E82-2000 kW é possível estimar a quantidade de energia anual do parque e observarmos o retorno do investimento do mesmo, segundo o ministério de minas e energias o último leilão de energia reserva para a energia eólica cotou o valor do MW no valor de R\$ 177,00 (cento e setenta e sete reais) (EMPRESA DE PESQUISA ENERGETICA, 2015). Com esses dados teremos um parque com uma produção de energia anual no valor de 176.503,4 MW e um valor de retorno do investimento de R\$ 31.241.101,80.

6.1 Trabalhos Futuros

Como trabalhos futuros podemos sugerir algumas ferramentas que poderiam facilitar a utilização do programa para o usuário como a entrada automática de outras formas de cálculo dos parâmetros da curva de Weibull, um método de correção automática dos dados das planilhas com a entrada de dados e uma complementação com a entrada de mais turbinas eólicas na lista de turbinas.

REFERÊNCIAS

AMARANTE, O. A. C. et al. **Atlas do Potencial Eólico Brasileiro**, Ministério de Minas e Energia, Eletrobrás, CEPEL, 2001.

ANEEL. BIG: **Banco de Informações de Geração. Capacidade de Geração do Brasil**. Disponível em < <http://www.aneel.gov.br/aplicacoes/capacidadebrasil/capacidadebrasil.cfm>>. Acesso em 22 de Dezembro de 2015.

CALDAS, D. M. **Estudo do potencial eólico e estimativa de geração de energia de um projeto eólico na cidade do Rio de Janeiro utilizando o WindPro e o WAsP**. Projeto submetido ao corpo docente do departamento de engenharia elétrica da escola politécnica da Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2010.

EMPRESA DE PESQUISA ENERGETICA. Leilão de energia alternativa 2015. **Leilão FA 2015 negocia energia de 11 usinas com deságio de 1,96%**. Abril, 2015. Disponível em: <<http://www.epe.gov.br/leiloes/Paginas/Leil%C3%A3o%20de%20Energia%20de%20Fontes%20Alternativas%202015/Leil%C3%A3oFA2015negociaenergiade11usinascomdes%C3%A1giode1,96.aspx?CategoriaID=6976>>. Acesso em 12 de Outubro de 2015./

MULLER, C. B., **Seleção de um aerogerador para utilização em uma torre de transmissão de dados em Taquari – RS**. Monografia apresentada ao Departamento de Engenharia Mecânica da Escola de Engenharia da Universidade Federal do Rio Grande do Sul. Porto Alegre, 2004.

PESSANHA J. F. M., BARCELOS, G. F. B., FARIA, A. V. C., & FERREIRA, V. M. F. (2010) Análise Estatística de Registros Anemométricos e Seleção de Turbinas Eólicas: Um Estudo de Caso. *Proceedings (Anais do XLII SBPO, Bento Gonçalves – RS, 2010)*.

PINHEIRO, L. M. G., **Estudos da geração eólio-elétrica levando em consideração as características dos ventos, seus aspectos probabilísticos e influência do terreno**. Monografia submetida à Universidade Federal do Ceará como parte dos requisitos para graduação em Engenharia Elétrica. Fortaleza, 2009.

PINTO, M., **Fundamentos de Energia Eólica**, Editora LTC, Rio de Janeiro, 2014.

PROTOCOLO DE KYOTO. **Protocolo de Kyoto**. S/A.S/D. Disponível em:<<http://protocolo-de-kyoto.info/>>. Acesso em 8 de Outubro de 2015.

WALTER, M. K. C. Mudanças Climáticas: Uma verdade inconveniente. Revista Multiciência. Campinas. Edição no. 8. Mudanças Climáticas. Maio, 2007.

WORD WIND ENERGY ASSOCIATION. **Worldwide Wind Market Booming Like Never Before: Wind capacity over 392 gigawatts**. s/a. 9 de setembro de 2015. Disponível em: <<http://www.wwindea.org/hyr2015/>>. Acesso em 12 de Outubro de 2015

APÊNDICE A – CÓDIGO DO PROGRAMA NO MATLAB

```

function varargout = Acelio_wind(varargin)
% ACELIO_WIND MATLAB code for Acelio_wind.fig
%     ACELIO_WIND, by itself, creates a new ACELIO_WIND or raises the
existing
%     singleton*.
%
%     H = ACELIO_WIND returns the handle to a new ACELIO_WIND or the
handle to
%     the existing singleton*.
%
%     ACELIO_WIND('CALLBACK',hObject,eventData,handles,...) calls the
local
%     function named CALLBACK in ACELIO_WIND.M with the given input
arguments.
%
%     ACELIO_WIND('Property','Value',...) creates a new ACELIO_WIND or
raises the
%     existing singleton*. Starting from the left, property value pairs
are
%     applied to the GUI before Acelio_wind_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to Acelio_wind_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help Acelio_wind

% Last Modified by GUIDE v2.5 05-Jan-2016 01:36:39

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
'gui_Singleton',  gui_Singleton, ...
'gui_OpeningFcn', @Acelio_wind_OpeningFcn, ...
'gui_OutputFcn',  @Acelio_wind_OutputFcn, ...
'gui_LayoutFcn',  [], ...
'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Acelio_wind is made visible.
function Acelio_wind_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure

```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% varargin command line arguments to Acelio_wind (see VARARGIN)

% Choose default command line output for Acelio_wind
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes Acelio_wind wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = Acelio_wind_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject handle to slider1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
% get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
cla(handles.axes2, 'reset');

handles.slider1 = get(hObject, 'Value');
tex.slider1 = handles.slider1;
set(handles.text19, 'String', tex.slider1);
k = handles.slider1;
c = handles.slider2;
frecuencia = handles.frecuencia;
vento=handles.vento;

weibull=wblpdf(vento,c,k);

bar(handles.axes2, [0.5:length(frecuencia)], frecuencia, 1);
hold(handles.axes2);

plot(handles.axes2, vento, weibull, 'r');
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject handle to slider1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

```

```

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Executes on slider movement.
function slider2_Callback(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
cla(handles.axes2,'reset');

handles.slider2 = get(hObject,'Value');
tex.slider2 = handles.slider2;
set(handles.text20,'String',tex.slider2);

k = handles.slider1;
c = handles.slider2;
frecuencia = handles.frecuencia;
vento = handles.vento;

weibull = wblpdf(vento,c,k);

bar(handles.axes2,[0.5:length(frecuencia)],frecuencia,1);
hold (handles.axes2);

plot(handles.axes2,vento,weibull,'r');
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function slider2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

%limpar os graficos

```

```

cla(handles.axes1, 'reset');
cla(handles.axes2, 'reset');
cla(handles.axes10, 'reset');

handles.fileName = uigetfile('*.xls');
handles.fileName = xlsread(handles.fileName);

matriz = handles.fileName;
vetor_media = [];
vetor_hora = [];
tam = size(matriz);

for i = 1 : tam(1)
    media = sum(matriz(i,:));
    media = media/tam(2);
    vetor_media = [vetor_media, media];
end

%calculado o vetor da media horaria

    mediahora = sum(matriz,1);
    mediahora = mediahora/tam(1);

%gerando o vetor frequencia
x=0;
n=0;
y=1;
frequencia = [];
while y < max(max(matriz))+2
for i= 1:tam(1)
for j = 1:tam(2)
if (x <=matriz(i,j)) & (matriz(i,j)<y);
        n = n + 1;
end
end
end
    frequencia = [frequencia,n];
    n=0;
    x=x+1;
    y=y+1;
end

%tirando a curva de weibull
k = handles.slider1;
c = handles.slider2;

frequencia = frequencia/(tam(1)*tam(2));
vento = [0:0.1:(max(max(matriz)))];
weibull = wblpdf(vento,c,k);

%calculando a velocidade media diaria
tam = size(vetor_media);
velocidade_media = (sum(vetor_media))/tam(2);

%calculando intensidade da turbulência e desvio padrão

```

```

desvio = std2(matriz);

turbulencia = (desvio/velocidade_media);

%calculado valor maximo da matriz
maximo1 = max(matriz);
maximo = max(maximo1);

%calculando o valor minimo da matriz
minimo1 = min(matriz);
minimo = min(minimo1);

handles.frequencia = frequencia;
handles.vento = vento;
handles.vetor_media = vetor_media;
handles.velocidade_media = velocidade_media;
handles.desvio = desvio;
handles.turbulencia = turbulencia;

set(handles.text24, 'String', desvio);
set(handles.text26, 'String', velocidade_media);
set(handles.text46, 'String', minimo);
set(handles.text48, 'String', maximo);
set(handles.text51, 'String', turbulencia);

%set(handles.text26, 'String', velocidade_media);

plot(handles.axes1, vetor_media)

plot(handles.axes10, mediahora)

bar(handles.axes2, [0.5:length(frequencia)], frequencia, 1);
hold(handles.axes2);
plot(handles.axes2, vento, weibull, 'r');
guidata(hObject, handles)

function edit2_Callback(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject, 'String') returns contents of edit2 as text
%        str2double(get(hObject, 'String')) returns contents of edit2 as a
double

% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.

```

```

%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%       str2double(get(hObject,'String')) returns contents of edit3 as a
double

% --- Executes during object creation, after setting all properties.
function edit3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)
% hObject    handle to edit4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit4 as text
%       str2double(get(hObject,'String')) returns contents of edit4 as a
double

Val = get(hObject,'String');
NewVal = str2double(Val);
handles.edit4 = NewVal;
guidata(hObject,handles);

% --- Executes during object creation, after setting all properties.
function edit4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

% --- Executes on button press in togglebutton1.
function togglebutton1_Callback(hObject, eventdata, handles)
% hObject    handle to togglebutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of togglebutton1

function edit5_Callback(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit5 as text
%        str2double(get(hObject,'String')) returns contents of edit5 as a
double

Val=get(hObject,'String');
NewVal = str2double(Val);
handles.edit5=NewVal;
guidata(hObject,handles);

% --- Executes during object creation, after setting all properties.
function edit5_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit6_Callback(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit6 as text
%        str2double(get(hObject,'String')) returns contents of edit6 as a
double

Val=get(hObject,'String');
NewVal = str2double(Val);
handles.edit6=NewVal;
guidata(hObject,handles);

% --- Executes during object creation, after setting all properties.
function edit6_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)

```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

function edit7_Callback(hObject, eventdata, handles)
% hObject handle to edit7 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit7 as text
% str2double(get(hObject,'String')) returns contents of edit7 as a
double

Val=get(hObject,'String');
NewVal = str2double(Val);
handles.edit7=NewVal;
guidata(hObject,handles);

% --- Executes during object creation, after setting all properties.
function edit7_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit7 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

function edit8_Callback(hObject, eventdata, handles)
% hObject handle to edit8 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit8 as text
% str2double(get(hObject,'String')) returns contents of edit8 as a
double

Val=get(hObject,'String');
NewVal = str2double(Val);
handles.edit8=NewVal;
guidata(hObject,handles);

% --- Executes during object creation, after setting all properties.
function edit8_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit8 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB

```

```

% handles      empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%      See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit9_Callback(hObject, eventdata, handles)
% hObject      handle to edit9 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit9 as text
%      str2double(get(hObject,'String')) returns contents of edit9 as a
double

Val=get(hObject,'String');
NewVal = str2double(Val);
handles.edit9=NewVal;
guidata(hObject,handles);

% --- Executes during object creation, after setting all properties.
function edit9_CreateFcn(hObject, eventdata, handles)
% hObject      handle to edit9 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%      See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton3 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

cla(handles.axes1,'reset');
cla(handles.axes2,'reset');

h = handles.edit7;
h1 = handles.edit8;
alfa = handles.edit9;
matriz = handles.fileName;
vetor_media = [];
matriz = matriz*((h1/h)^alfa);
tam = size(matriz);
for i = 1 : tam(1)
    media = sum(matriz(i,:));
    media = media/tam(2);
    vetor_media = [vetor_media, media];
end

```

```

end

%calculado o vetor da media horaria

    mediahora = sum(matriz,1);
    mediahora = mediahora/tam(1);

%gerando o vetor frequencia
x=0;
n=0;
y=1;
frequencia = [];
while y < max(max(matriz))
for i= 1:tam(1)
for j = 1:tam(2)
if (x <=matriz(i,j)) & (matriz(i,j)<y);
        n = n + 1;
end
end
end
    frequencia = [frequencia,n];
    n=0;
    x=x+1;
    y=y+1;
end

%tirando a curva de weibull
k = handles.slider1;
c = handles.slider2;

frequencia = frequencia/(tam(1)*tam(2));
vento=[0:0.1:(max(max(matriz)))];
weibull=wblpdf(vento,c,k);

bar(handles.axes2,[0.5:length(frequencia)],frequencia,1);
hold(handles.axes2)

tam = size(vetor_media);
velocidade_media = sum(vetor_media);
velocidade_media = velocidade_media/tam(2);
desvio = std2(matriz);

turbulencia = (desvio/velocidade_media);

handles.vento = vento;
handles.frequencia = frequencia;
handles.vetor_media = vetor_media;
handles.velocidade_media = velocidade_media;
handles.desvio = desvio;

%calculado valor maximo da matriz
maximol = max(matriz);
maximo = max(maximol);

%calculando o valor minimo da matriz
minimol = min(matriz);
minimo = min(minimol);

```

```

set(handles.text26, 'String', velocidade_media);
set(handles.text24, 'String', desvio);
set(handles.text46, 'String', minimo);
set(handles.text48, 'String', maximo);
set(handles.text51, 'String', turbulencia);

plot(handles.axes1, vetor_media, 'r');
plot(handles.axes10, mediahora, 'r');
plot(handles.axes2, vento, weibull, 'r');
guidata(hObject, handles);

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

cla(handles.axes3, 'reset')

fileName = uigetfile('*.xls');
fileName = xlsread(fileName);

plan = fileName;
vetor_dire = [];
tam = size(plan);
for i = 1 : tam(1)
    media = sum(plan(i, :));
    media = media/tam(2);
    vetor_dire = [vetor_dire, media];
end

vetor_dire = vetor_dire * pi/180;
handles.vetor_dire = vetor_dire;
rose(handles.axes3, vetor_dire);
guidata(hObject, handles)

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton8 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
figure(1);
vetor_media = handles.vetor_media;
plot(vetor_media);
guidata(hObject, handles)

% --- Executes on button press in pushbutton9.
function pushbutton9_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton9 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
figure(2);
k = handles.slider1;
c = handles.slider2;
frequencia = handles.frequencia;
vento=handles.vento;

```

```

weibull=wblpdf(vento,c,k);

bar([0.5:length(frequencia)],frequencia,1);
hold on;

plot(vento,weibull,'r');
guidata(hObject, handles)

% --- Executes on button press in pushbutton10.
function pushbutton10_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton10 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
figure(3);

vetor_dire = handles.vetor_dire;

rose(vetor_dire);

% --- Executes during object creation, after setting all properties.
function text24_CreateFcn(hObject, eventdata, handles)
% hObject    handle to text24 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

guidata(hObject, handles)

% --- Executes on button press in pushbutton12.
function pushbutton12_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton12 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes2, 'reset');

handles.slider1 = (handles.desvio/handles.velocidade_media)^(-1.090);
handles.slider2 =
((handles.velocidade_media)/gamma(1+(1/handles.slider1)));

tex.slider1 = handles.slider1;
set(handles.text19, 'String', tex.slider1);
tex.slider2 = handles.slider2;
set(handles.text20, 'String', tex.slider2);

k = handles.slider1;
c = handles.slider2;
frequencia = handles.frequencia;
vento=handles.vento;

weibull=wblpdf(vento,c,k);

bar(handles.axes2,[0.5:length(frequencia)],frequencia,1);
hold(handles.axes2);

plot(handles.axes2,vento,weibull,'r');

```

```
guidata(hObject, handles)
```

```
% --- Executes on selection change in listbox.
function listbox_Callback(hObject, eventdata, handles)
% hObject    handle to listbox (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns listbox contents
as cell array
%         contents{get(hObject,'Value')} returns selected item from listbox
set(hObject, 'BackgroundColor', 'white');
inf=get(hObject, 'Value');
gos=get(hObject, 'String');
cla(handles.axes7, 'reset');

    potencia = [];
switch inf
case 1 %E33
%         v = [1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25];
%         cp = [0.00 0.00 0.35 0.40 0.45 0.47 0.50 0.50 0.50 0.47 0.41
0.35 0.28 0.228 0.184 0.152 0.128 0.108 0.092 0.078 0.067 0.059 0.052 0.045
0.04];
%         a = 876;
%         for i = 1 : 25
%             potencia(1, i) = 0.6125 * a * v(1, i).^3 * cp(1, i);
%         end
    potencia = [0.0 0.0 5.0 13.7 30.0 55.0 92.0 138.0 196.0 250.0
292.8 320.0 335.0 335.0 335.0 335.0 335.0 335.0 335.0 335.0 335.0 335.0
335.0 335.0 335.0];
    potencia = potencia * 1000;

    vp = 3;
    vn = 13;
    vc = 25;
    k = handles.slider1;
    c = handles.slider2;

    F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
    F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
    FC1 = quad(F1, vp, vn);
    FC2 = quad(F2, vn, vc);
    FC = (1/(vn)^3)*FC1 + FC2;
    set(handles.text16, 'String', FC);

    PM = potencia(1, 24) * FC;
    PM = PM/1000;
    energia = potencia(1, 24) * FC * 8760;
    energia = energia/1000000;

    set(handles.text31, 'String', '37 m / 44 m / 49 m / 50 m');
    set(handles.text32, 'String', '3');
    set(handles.text35, 'String', '13');
    set(handles.text33, 'String', '25');
    set(handles.text39, 'String', PM);
    set(handles.text41, 'String', energia);
    plot(handles.axes7, potencia, 'r');
```

```

case 2 %E44
potencia = [0.0 0.0 4.0 20.0 50.0 96.0 156.0 238.0 340.0 466.0
600.0 710.0 790.0 850.0 880.0 905.0 910.0 910.0 910.0 910.0 910.0
910.0 910.0 910.0];
potencia = potencia * 1000;

vp = 3;
vn = 17;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
FC1 = quad(F1, vp, vn);
FC2 = quad(F2, vn, vc);
FC = (1/(vn)^3)*FC1 + FC2;
set(handles.text16, 'String', FC);

PM = potencia(1, 24) * FC;
PM = PM/1000;
energia = potencia(1, 24) * FC * 8760;
energia = energia/1000000;
set(handles.text31, 'String', '45 m/ 55 m/ 65 m');
set(handles.text32, 'String', '3');
set(handles.text35, 'String', '17');
set(handles.text33, 'String', '25');
set(handles.text39, 'String', PM);
set(handles.text41, 'String', energia);
plot(handles.axes7, potencia, 'r');

case 3 %E48
potencia = [0.0 0.0 5.0 25.0 60.0 110.0 180.0 275.0 400.0 555.0
671.0 750.0 790.0 810.0 810.0 810.0 810.0 810.0 810.0 810.0 810.0
810.0 810.0 810.0];
potencia = potencia * 1000;

vp = 3;
vn = 14;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
FC1 = quad(F1, vp, vn);
FC2 = quad(F2, vn, vc);
FC = (1/(vn)^3)*FC1 + FC2;
set(handles.text16, 'String', FC);

PM = potencia(1, 24) * FC;
PM = PM/1000;
energia = potencia(1, 24) * FC * 8760;
energia = energia/1000000;
set(handles.text31, 'String', '50 m / 60 m / 75 m / 76 m');
set(handles.text32, 'String', '3');
set(handles.text35, 'String', '14');
set(handles.text33, 'String', '25');
set(handles.text39, 'String', PM);

```

```

set(handles.text41,'String',energia);
plot(handles.axes7,potencia, 'r');

case 4 %E53
potencia = [0.0 2.0 14.0 38.0 77.0 141.0 228.0 336.0 480.0
645.0 744.0 780.0 810.0 810.0 810.0 810.0 810.0 810.0 810.0 810.0 810.0
810.0 810.0 810.0 810.0];
potencia = potencia * 1000;

vp = 2;
vn = 13;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
FC1 = quad(F1, vp, vn);
FC2 = quad(F2, vn, vc);
FC = (1/(vn)^3)*FC1 + FC2;
set(handles.text16,'String',FC);

PM = potencia(1, 24) * FC;
PM = PM/1000;
energia = potencia(1, 24) * FC * 8760;
energia = energia/1000000;
set(handles.text31,'String','60 m/ 73 m/ 75 m');
set(handles.text32,'String','2');
set(handles.text35,'String','13');
set(handles.text33,'String','25');
set(handles.text39,'String',PM);
set(handles.text41,'String',energia);
plot(handles.axes7,potencia, 'r');

case 5 %E70
potencia = [0.0 2.0 18.0 56.0 127.0 240.0 400.0 626.0 892.0
1223.0 1590.0 1900.0 2080.0 2230.0 2300.0 2310.0 2310.0 2310.0 2310.0
2310.0 2310.0 2310.0 2310.0 2310.0 2310.0];
potencia = potencia * 1000;

vp = 2;
vn = 14;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
FC1 = quad(F1, vp, vn);
FC2 = quad(F2, vn, vc);
FC = (1/(vn)^3)*FC1 + FC2;
set(handles.text16,'String',FC);

PM = potencia(1, 24) * FC;
PM = PM/1000;
energia = potencia(1, 24) * FC * 8760;
energia = energia/1000000;
set(handles.text31,'String','57 m / 64 m / 85 m / 98 m / 113
m');
set(handles.text32,'String','2');

```

```

set(handles.text35, 'String', '14');
set(handles.text33, 'String', '25');
set(handles.text39, 'String', PM);
set(handles.text41, 'String', energia);
plot(handles.axes7, potencia, 'r');

case 6 %E82
potencia = [0.0 3.0 25.0 82.0 174.0 321.0 532.0 815.0 1180.0
1580.0 1810.0 1980.0 2050.0 2050.0 2050.0 2050.0 2050.0 2050.0 2050.0
2050.0 2050.0 2050.0 2050.0 2050.0 2050.0];
potencia = potencia * 1000;

vp = 2;
vn = 13;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1) ./ (c^k)) .*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1) ./ (c^k)) .*exp(-(v/c).^k);
FC1 = quad(F1, vp, vn);
FC2 = quad(F2, vn, vc);
FC = (1/(vn)^3)*FC1 + FC2;
set(handles.text16, 'String', FC);

PM = potencia(1, 24) * FC;
PM = PM/1000;
energia = potencia(1, 24) * FC * 8760;
energia = energia/1000000;
set(handles.text31, 'String', '78 m / 85 m / 98 m / 108 m /
138');

set(handles.text32, 'String', '2');
set(handles.text35, 'String', '13');
set(handles.text33, 'String', '25');
set(handles.text39, 'String', PM);
set(handles.text41, 'String', energia);
plot(handles.axes7, potencia, 'r');

case 7 %E82
potencia = [0.0 3.0 25.0 82.0 174.0 321.0 532.0 815.0 1180.0
1580.0 1890.0 2100.0 2250.0 2350.0 2350.0 2350.0 2350.0 2350.0 2350.0
2350.0 2350.0 2350.0 2350.0 2350.0 2350.0];
potencia = potencia * 1000;

vp = 2;
vn = 14;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1) ./ (c^k)) .*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1) ./ (c^k)) .*exp(-(v/c).^k);
FC1 = quad(F1, vp, vn);
FC2 = quad(F2, vn, vc);
FC = (1/(vn)^3)*FC1 + FC2;
set(handles.text16, 'String', FC);

PM = potencia(1, 24) * FC;
PM = PM/1000;

```

```

energia = potencia(1, 24)* FC * 8760;
energia = energia/1000000;
set(handles.text31,'String','78 m / 85 m / 98 m / 108 m / 138
m');

set(handles.text32,'String','2');
set(handles.text35,'String','14');
set(handles.text33,'String','25');
set(handles.text39,'String',PM);
set(handles.text41,'String',energia);
plot(handles.axes7,potencia, 'r');

case 8 %E82
potencia = [0.0 3.0 25.0 82.0 174.0 321.0 532.0 815.0 1180.0
1580.0 1900.0 2200.0 2480.0 2700.0 2850.0 2950.0 3020.0 3020.0 3020.0
3020.0 3020.0 3020.0 3020.0 3020.0 3020.0];
potencia = potencia * 1000;

vp = 2;
vn = 17;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
FC1 = quad(F1,vp,vn);
FC2 = quad(F2,vn,vc);
FC = (1/(vn)^3)*FC1 + FC2;
set(handles.text16,'String',FC);

PM = potencia(1, 24)* FC;
PM = PM/1000;
energia = potencia(1, 24)* FC * 8760;
energia = energia/1000000;
set(handles.text31,'String','78 m / 85 m / 98 m / 108 m / 138
m');

set(handles.text32,'String','2');
set(handles.text35,'String','17');
set(handles.text33,'String','25');
set(handles.text39,'String',PM);
set(handles.text41,'String',energia);
plot(handles.axes7,potencia, 'r');

case 9 %E101
potencia = [0.0 3.0 37.0 118.0 258.0 479.0 790.0 1200.0 1710.0
2340.0 2867.0 3034.0 3050.0 3050.0 3050.0 3050.0 3050.0 3050.0
3050.0 3050.0 3050.0 3050.0 3050.0 3050.0];
potencia = potencia * 1000;

vp = 2;
vn = 13;
vc = 25;
k = handles.slider1;
c = handles.slider2;

F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
FC1 = quad(F1,vp,vn);
FC2 = quad(F2,vn,vc);
FC = (1/(vn)^3)*FC1 + FC2;

```

```

set(handles.text16, 'String', FC);

PM = potencia(1, 24)* FC;
PM = PM/1000;
energia = potencia(1, 24)* FC * 8760;
energia = energia/1000000;
set(handles.text31, 'String', '99 m / 135 m');
set(handles.text32, 'String', '2');
set(handles.text35, 'String', '13');
set(handles.text33, 'String', '25');
set(handles.text39, 'String', PM);
set(handles.text41, 'String', energia);
plot(handles.axes7, potencia, 'r');

case 10 %E126
    potencia = [0.0 0.0 55.0 175.0 410.0 760.0 1250.0 1900.0 2700.0
3750.0 4850.0 5750.0 6500.0 7000.0 7350.0 7500.0 7580.0 7580.0 7580.0
7580.0 7580.0 7580.0 7580.0 7580.0 7580.0];
    potencia = potencia * 1000;

    vp = 3;
    vn = 17;
    vc = 25;
    k = handles.slider1;
    c = handles.slider2;

    F1 = @(v) (k*v.^(3+k-1)./(c^k)).*exp(-(v/c).^k);
    F2 = @(v) (k*v.^(k-1)./(c^k)).*exp(-(v/c).^k);
    FC1 = quad(F1, vp, vn);
    FC2 = quad(F2, vn, vc);
    FC = (1/(vn)^3)*FC1 + FC2;
    set(handles.text16, 'String', FC);

    PM = potencia(1, 24)* FC;
    PM = PM/1000;
    energia = potencia(1, 24)* FC * 8760;
    energia = energia/1000000;
    set(handles.text31, 'String', '135m');
    set(handles.text32, 'String', '3');
    set(handles.text35, 'String', '17');
    set(handles.text33, 'String', '25');
    set(handles.text39, 'String', PM);
    set(handles.text41, 'String', energia);
    plot(handles.axes7, potencia, 'r');

end
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function listbox_CreateFcn(hObject, eventdata, handles)
% hObject    handle to listbox (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))

```

```
end
```

```
% --- Executes during object creation, after setting all properties.  
function text26_CreateFcn(hObject, eventdata, handles)  
% hObject    handle to text26 (see GCBO)  
% eventdata  reserved - to be defined in a future version of MATLAB  
% handles    empty - handles not created until after all CreateFcns called
```