



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

INPE-13018-PRE/8295

ANÁLISE MULTIESCALA DE SINGULARIDADES EM CAMPOS METEOROLÓGICOS

Paulo Mauricio Vital*

*Bolsista INPE

Relatório Final de Projeto de Iniciação Científica (PIBIC/CNPq/INPE), orientado pelos
Drs. Margarete Oliveira Domingues e Chou Sin Chan

INPE
São José dos Campos
2005



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS



RELATÓRIO FINAL DE PROJETO DE INICIAÇÃO CIENTÍFICA
(PIBIC/CNPq/INPE)

Análise multiescala de singularidades em campos meteorológicos

Paulo Mauricio Vital (INPE, Bolsista PIBIC/CNPq)
e-mail: pvital@cptec.inpe.br

Dra. Margarete Oliveira Domingues (LAC/INPE, Orientadora)
e-mail: mo.domingues@lac.inpe.br

Dra. Chou Sin Chan (CPTEC/INPE, Orientadora)
e-mail: chou@cptec.inpe.br

Março a Junho de 2005

1. INTRODUÇÃO

A transformada *wavelet* originalmente foi empregada para análise de sinais sísmicos na geofísica, embora tais idéias já constituíssem um centro de convergência dos desenvolvimentos de físicos, matemáticos, engenheiros, entre outros.

O uso das técnicas *wavelet* na área de análise de dados vem crescendo rapidamente, pois elas representam uma síntese de idéias antigas aliadas a resultados matemáticos recentes e eficientes algoritmos computacionais de interesse de várias comunidades.

Nas análises de sinais meteorológicos, em particular, duas linhas mestras de utilização dessa técnica têm sido seguidas: uma delas é a análise de singularidade e a outra é a análise *wavelet* de variância ou espectro de *wavelet*. No estudo deste projeto é utilizada a segunda técnica na avaliação de séries temporais observacionais e séries temporais obtidas do modelo Eta/CPTec em um ponto de grade específico associado a posição de coleta do dado observacional.

2. Metodologia e Dados

2.1. Análise Wavelet

Ao nome wavelet associa-se a idéia de “pequenas ondas”, esse termo está associado a ondas localizadas, ondas que crescem e decaem em um período de tempo. Para que uma função seja denominada wavelet, denotada pela letra ψ , ela deve obedecer as seguintes propriedades:

1) A integral da função wavelet de ser zero, i.e.,

$$\int_{-\infty}^{\infty} \psi(t) dt = 0.$$

Isso garante que a função wavelet tenha uma forma do tipo onda. Essa condição é conhecida como condições de admissibilidade.

2) A função wavelet deve ter energia unitária, i.e.,

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt = 1.$$

Isso garante que a função wavelet possua suporte compacto, ou com um decaimento rápido de amplitude, garantindo a localização espacial (Domingues et al., 2003).

As funções wavelet possuem a propriedade de dupla localização: em frequência e em tempo, com um compromisso entre elas. A localização temporal ocorre devido à função wavelet localizar-se em um intervalo finito. Dessa forma, à medida que a escala aumenta, as funções wavelet dessas escalas ficam localizadas em intervalos de comprimento cada vez menores. Em cada nível de escala, todas as funções wavelet possuem a mesma forma, só mudando seus pontos de localização, i.e., transladando. A localização em frequência deve-se à transformada de Fourier da função wavelet poder ser interpretada como um filtro passa-banda. Devido à propriedade de dupla localização das funções wavelet, a transformada wavelet é dita do tipo local em tempo-frequência, com resolução temporal e em frequência, inversamente proporcional (Domingues, et al. 2003).

A transformada wavelet é uma transformada linear que pode ser utilizada na análise de sinais não estacionários para extrair informações das variações em frequência desses sinais e para detectar suas estruturas temporalmente e/ou espacialmente localizadas.

A transformada wavelet de uma série temporal f é definida pela transformada integral

$$\mathfrak{W}_f^\psi(a, b) = \int_{-\infty}^{\infty} f(u) \bar{\psi}_{a,b}(u) du \quad a > 0,$$

em que

$$\psi_{a,b}(u) = \frac{1}{\sqrt{a}} \psi \left(\frac{u - b}{a} \right)$$

representa a família de funções wavelet escolhida, denominada wavelet-mãe. O parâmetro a se refere à escala, b é um parâmetro de translação ou localização da função wavelet-mãe e $\bar{\psi}_{a,b}(u)$ é o complexo conjugado de $\psi_{a,b}(u)$.

A variação do parâmetro a tem o efeito de dilatação (quando $a > 1$) e de contração (quando $a < 1$) da função wavelet-mãe. Com isto, pode-se analisar respectivamente os aspectos de longo ou curto período na série. À medida que b varia, a função f é analisada localmente em torno desse ponto. Esse comportamento pode ser analisado na figura 1.

Neste trabalho utiliza-se a transformada contínua de Morlet para calcular o espectro wavelet ou simplesmente escalograma.

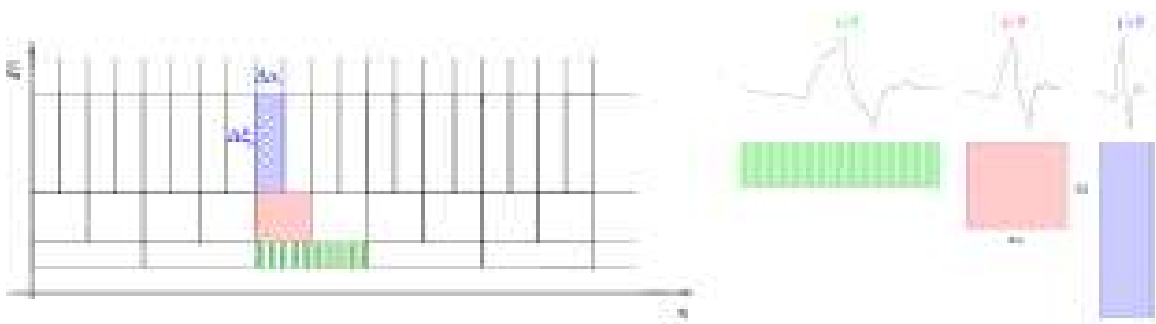


Figura 1 Esquema do diagrama tempo X pseudo freqüência (escala) e da dilatação da função wavelet-mãe nas escalas.

A função wavelet-mãe de Morlet é uma função complexa, cuja a parte real está apresentada na Figura 2.

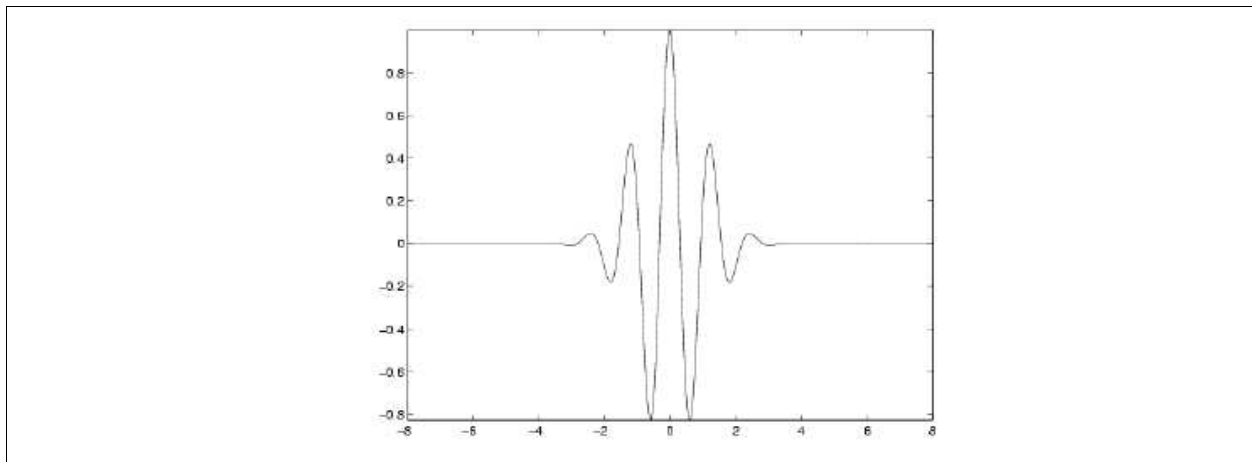


Figura 2 Gráfico da parte real da função wavelet-mãe de Morlet.

2.2. Scilab

O Scilab (www.scilab.org) é um software livre desenvolvido pela INRIA (Institut de Recherche en Informatique et en Automatique). Este software é um ambiente utilizado para facilitar a resolução de problemas numéricos, de simples acesso a inúmeras bibliotecas numéricas como: cálculo matricial, integração numérica, otimização, equações diferenciais. Além disso, o Scilab possui uma sintaxe simples. Por essas facilidades os programas em Scilab são compactos e de fácil entendimento, frequentemente muito mais curtos que programas em linguagens compiladas. Para o cálculo da transformada wavelet foi utilizado o pacote Fraclab (www.scilab.org, contribuições).

2.3. Modelo Regional Eta

O modelo regional operacional do CPTEC denominado Eta, é proveniente do “National Centers for Environmental Prediction” (NCEP). O modelo utiliza como condições de contorno as previsões do modelo global CPTEC, estas últimas são atualizadas a cada 6 horas de integração. A resolução horizontal utilizada é de 40km e a vertical de 38 camadas. As parametrizações incluídas no modelo são: troca vertical turbulenta pelo esquema de Mellor-Yamada, radiação de onda longa segundo esquema de Lacis e Hansen, radiação de onda curta pelo esquema de Fels e Schwarzkopf, água no solo segue o

esquema de Chen, as chuvas convectivas são produzidas pelo esquema Betts-Miller-Janjic. Atualmente o domínio do modelo abrange a maior parte da América do Sul (Josiane et al., 2005).

Dois tipos de dados provenientes do modelo Eta são utilizados nos casos de estudo: dados de uma rodada longa e dados de rodadas curtas. Os dados de uma rodada longa são os dados retidos da integração contínua, neste caso, do dia 15 de janeiro com previsão até 30 de maio de 2002. Os dados das rodadas curtas são as previsões de tempo em si, que são retirados das integrações de 12 em 12 horas, neste caso a partir do dia 15 de janeiro até o dia 30 de maio de 2002. Neste caso são utilizadas previsões de 30 a 48 horas. As variáveis analisadas são: temperatura do ar, umidade relativa, precipitação e pressão no ponto de grade correspondente a Passo do Lontra(57W, 19.56S), Miranda (MS).

2.4. Dados da Torre (Observacionais)

Os dados observacionais são da torre micrometeorológica do INPE/UFMS de resposta lenta (cada trinta minutos) e dados de resposta rápida (~8 Hz). São analisados os dados de temperatura do ar, umidade relativa, precipitação e pressão. No total esses dados compoem uma base de mais de 2Gbytes de informação.

São avaliados os dados de resposta lenta da torre com os dados retirados de integrações de 6 em 6 horas do modelo Eta, em particular nesta etapa são analisados os dados a 2 metros de altura. O mesmo período de dados anterior foi utilizado nesse estudo, ie, 15 de janeiro a 30 de maio de 2002.

2.5. OpenDX

O OpenDX (www.opendx.org) é um programa gratuito com código aberto, semelhante e derivado do Data Explorer da IBM (DX,2005) e construído por um conjunto de ferramentas de manipulação, processamento, visualização e animação gráfica. Além disso, ele possui um grande potencial de transformação de dados que é utilizado em diversas áreas científicas como Ciências Espaciais, Meteorologia, Oceanografia e entre outras.

Esse visualizador gráfico, também foi desenvolvido pelas equipes da IBM Visualization Systems.

Ele apresenta uma versão para diversos ambientes operacionais, com GNU/Linux, Solaris, Microsoft Windows, entre outros.

O programa contém diversos ambientes de programação que permite ao usuário ter uma grande flexibilidade no momento de criar uma visualização. Esses ambientes se dividem em um editor de programas visuais (VPE), que é o ambiente gráfico do programa; e um módulo constituído por um script, que permite ao usuário ter controle do programa de visualização por meio de um prompt. O script do programa possibilita que o usuário estenda as funcionalidades do programa adicionando módulos por meio de bibliotecas, assim como, criando seus próprios módulos (Thompson et al., 2001; Ibañez et al, 2005; Blaz et al, 2001).

3. Resultados

As séries temporais observacionais possuem alguns problemas como falta de dados e dados com erros de gravação. Em alguns casos os dados observacionais e do modelo são obtidos em variáveis ou em unidades diferentes. Por exemplo, na torre se dispõem de medidas de umidade relativa e no modelo de umidade específica, e os dados de temperatura do ar na torre estão em Celsius e os do modelo estão em Kelvin.

Os dados da torre possuem um formato diferente dos do modelo, então foi necessário substituir vírgula por espaço e converter datas. Para lidar com esses desafios foram feitas algumas correções, conversões e novas formatações nos dados em programas em C, descritos no Apêndice A.

Em alguns tempos, os dados de pressão atmosférica da torre de observação micrometeorológica possuíam os valores muito altos (1053.5) ou valores muito baixos constantes (599.88, 600, 612.22, 673.73), indicando falha do sensor e/ou armazenamento dos dados. Com a finalidade de corrigir esse problema, como isso ocorria em apenas uma medida isolada, foi feita uma média entre o dado anterior e o dado posterior. Como as comparações foram feitas de 6 em 6 horas apenas em 8 casos houve a necessidade de se preencher as falhas. Nos dados do Modelo Regional Eta não haviam falhas. Os dados do Modelo Regional Eta foram extraídos por meio de um script desenvolvido em Grads.

Inicialmente, para analisar e comparar séries temporais de dados é preciso visualizá-la e comparar qualitativamente suas características mais marcantes. Isso foi realizado no programa Scilab. Após isso, aplicou-se a Transformada Wavelet Contínua de Morlet nos dados do Modelo Regional Eta e nos dados da torre de observação e calculou-se os escalogramas. Isso foi realizado por um programa na linguagem Scilab utilizando funções do Fraclab. Este programa encontra-se no Apêndice B.

Assim, a seguir são apresentadas as séries temporais estudadas e seus respectivos escalogramas.

Observando as Figuras 3-6 geradas com base nos dados de temperatura e umidade da torre de observação e do modelo regional Eta/CPTEC, percebe-se que a frequência de variação nos dados da torre de observação é maior que a frequência de variação dos dados do Modelo Regional Eta. Outra observação preliminar interessante é que quando ocorreram sistemas meteorológicos característicos multiescala também foram verificadas nesses escalogramas

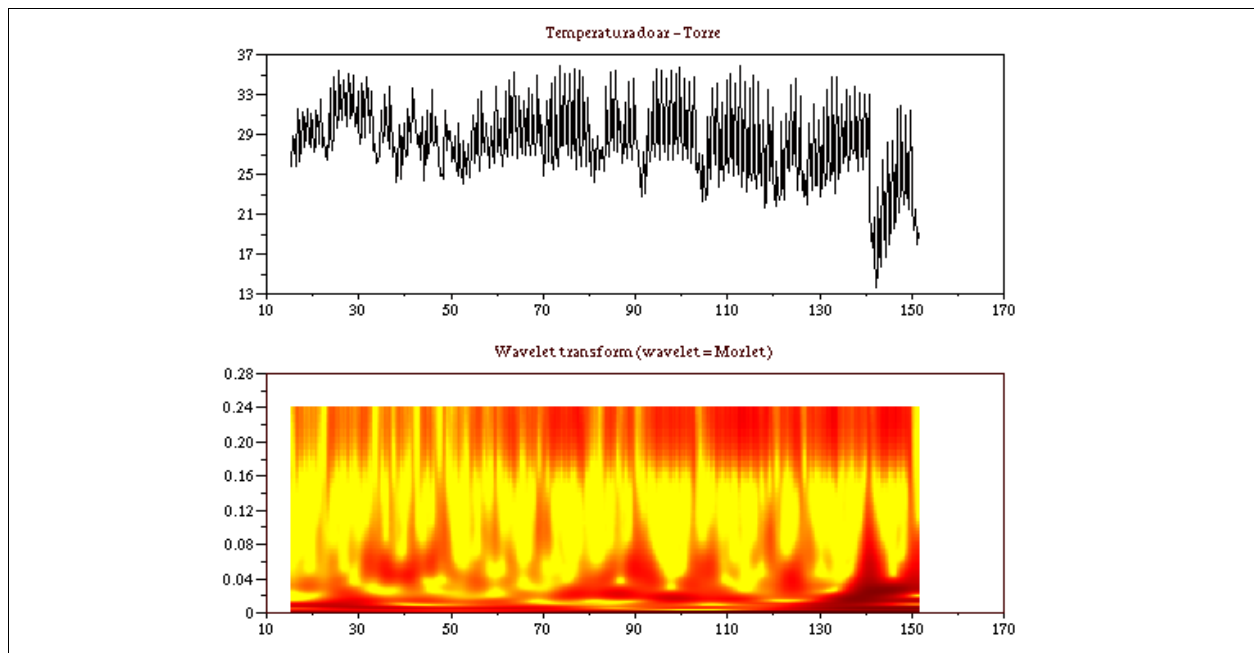


Figura 3 Série temporal de 6 em 6 horas de temperatura do ar a 2 m (em graus Celsius), na torre micrometeorológica de Passo do Lontra, Miranda (MS) e seu respectivo escalograma (tempo x pseudo-escala) com a wavelet de Morlet.

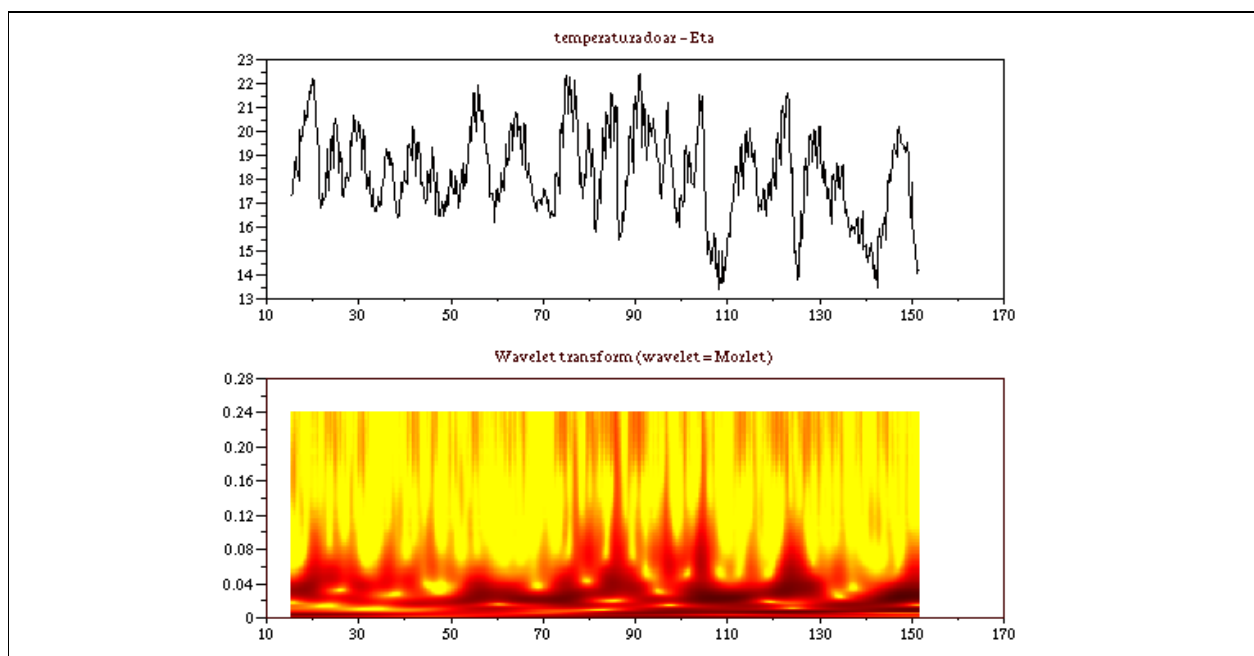


Figura 4 Série temporal de 6 em 6 horas de temperatura do ar a superfície (graus Celsius) proveniente da previsão da rodada longa do Eta/CPTEC e seu respectivo escalograma (tempo x pseudo-escala) com a wavelet de Morlet.

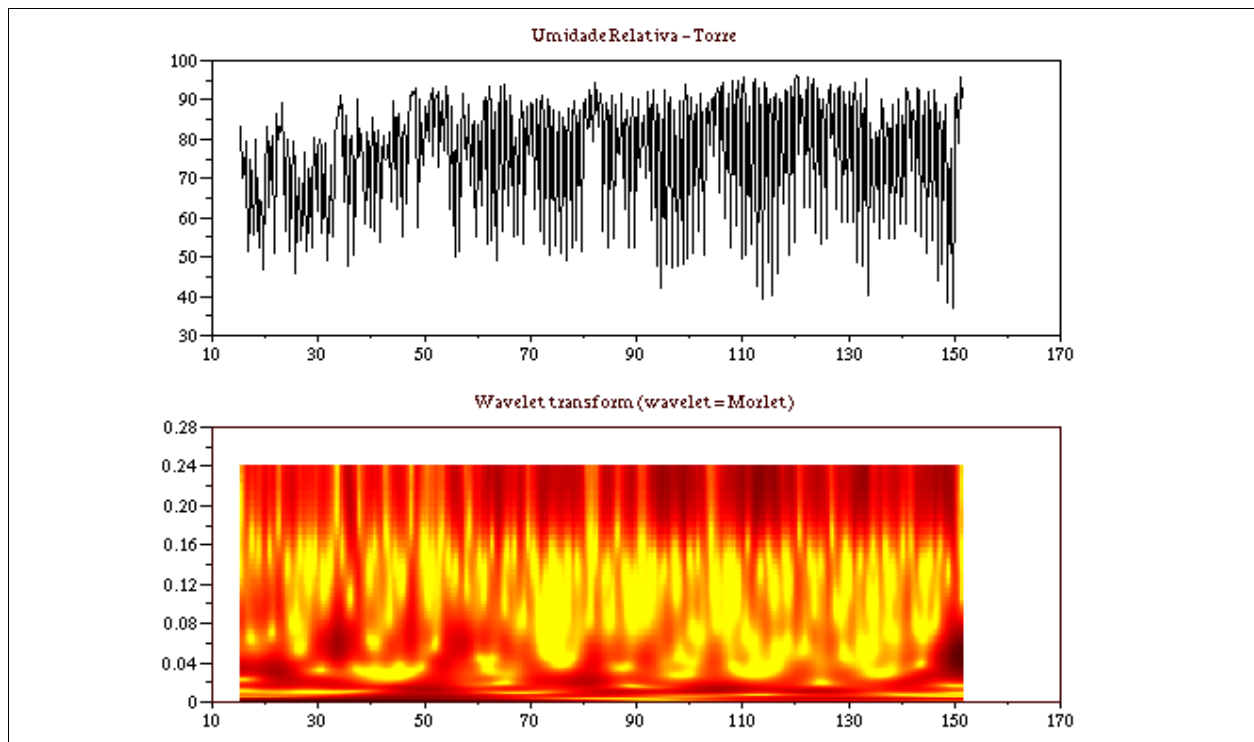


Figura 5 Série temporal de 6 em 6 horas de umidade relativa a 2 metros (em %), na torre micrometeorológica de Passo do Lontra, Miranda (MS) e seu respectivo escalograma (tempo x pseudo-escala) com a wavelet de Morlet.

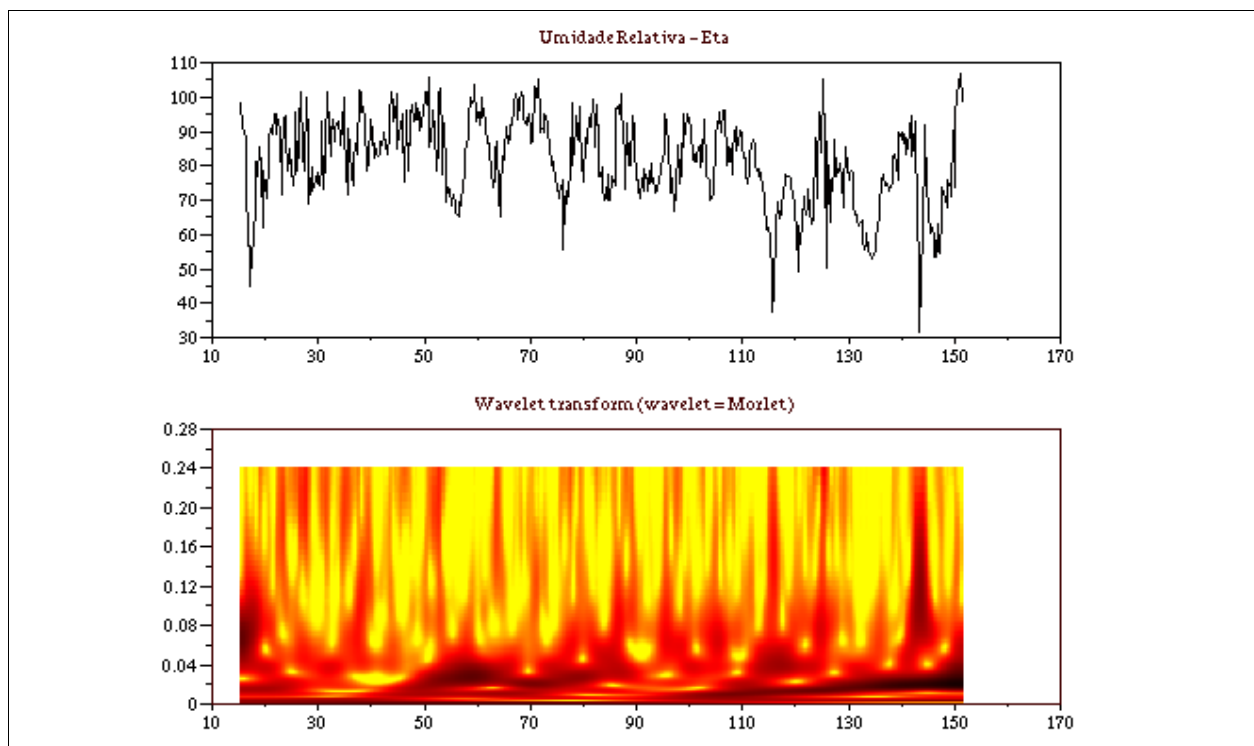


Figura 6 Série temporal de 6 em 6 horas de umidade relativa a superfície (em %) proveniente da previsão da rodada longa do Eta/CPTEC e seu respectivo escalograma (tempo x pseudo-escala) com a wavelet de Morlet.

Observando as Figuras 7-8 geradas com base nos dados de pressão atmosférica da torre de observação e do modelo regional Eta/CPTEC, percebe-se que ambos os dados apresentam variações nas baixas frequência muito parecidas no escalograma.

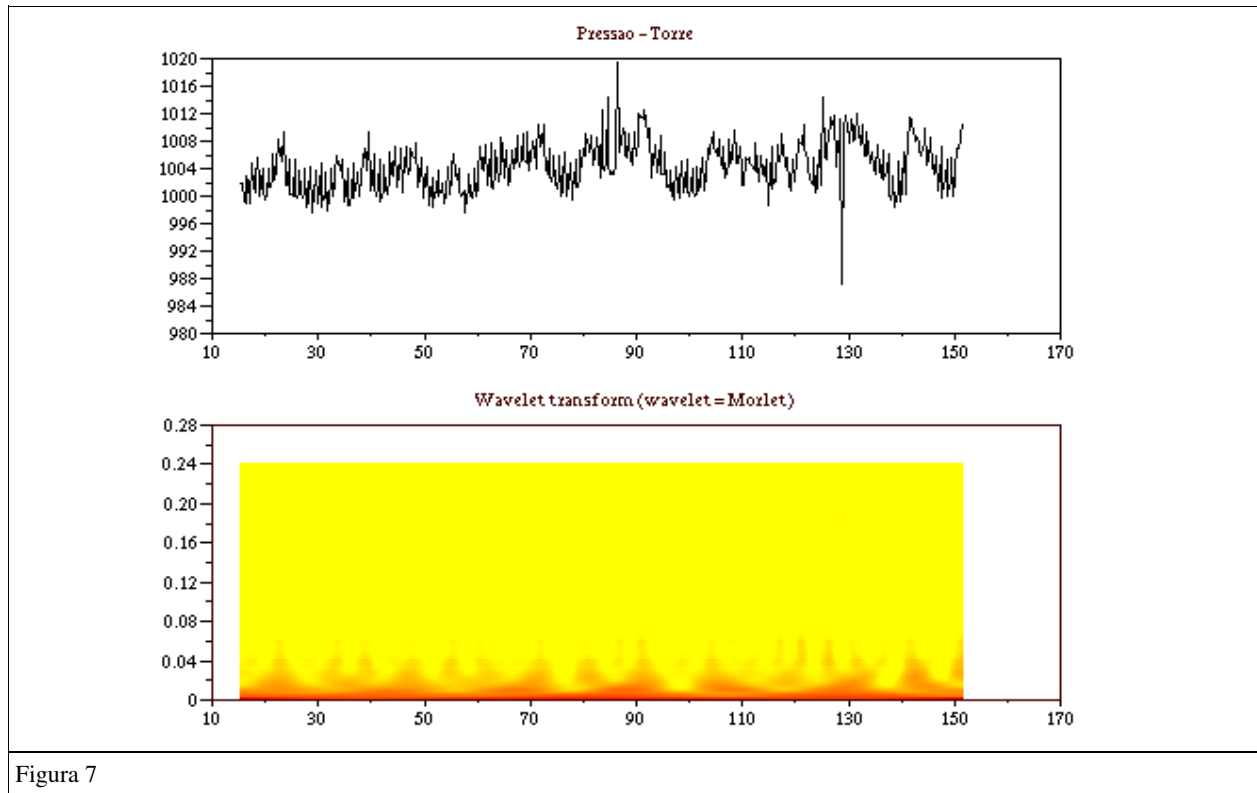


Figura 7

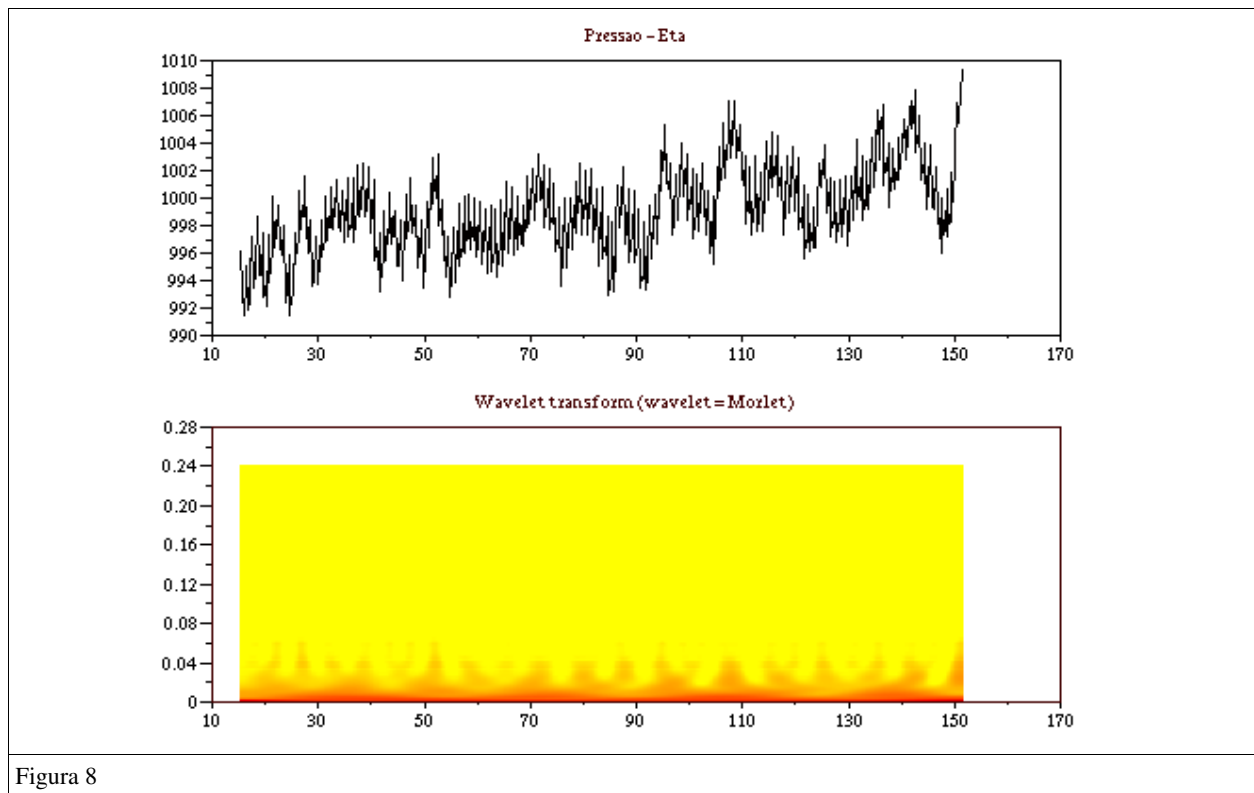


Figura 8

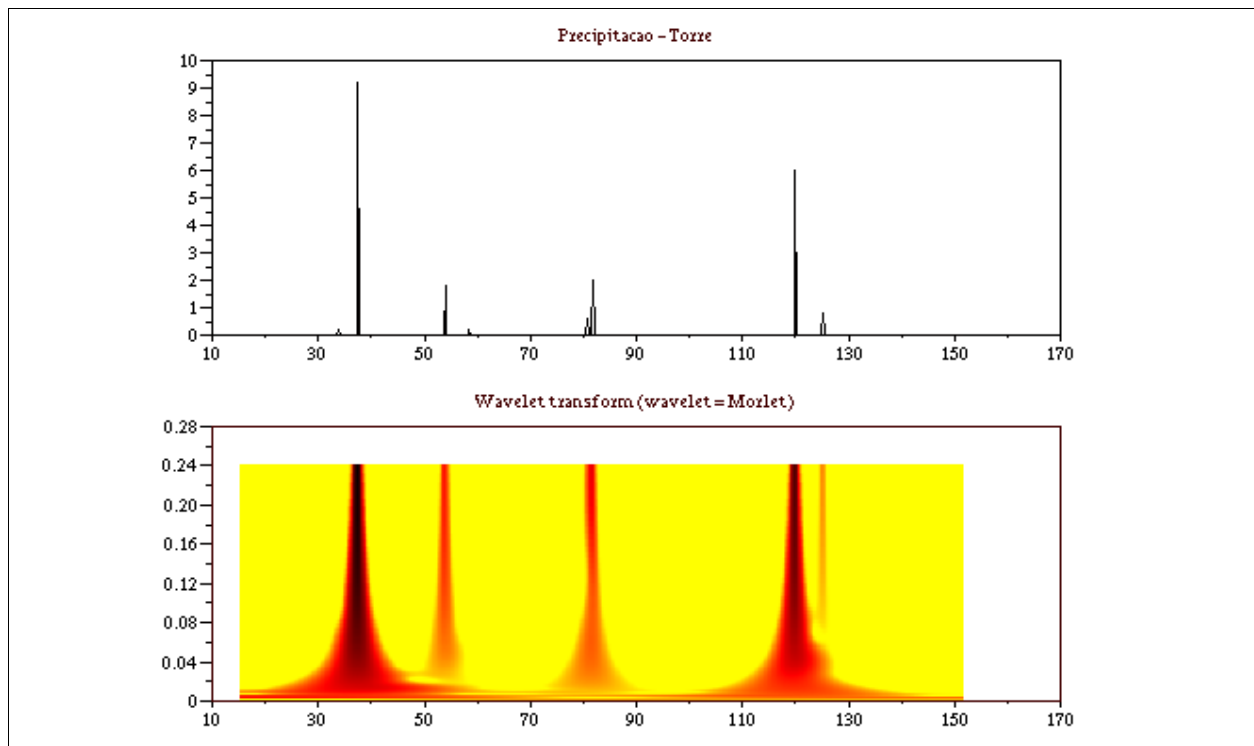


Figura 9 Série temporal de 6 em 6 horas de precipitação (mm), na torre micrometeorológica de Passo do Lontra, Miranda (MS) e seu respectivo escalograma (tempo x pseudo-escala) com a wavelet de Morlet.

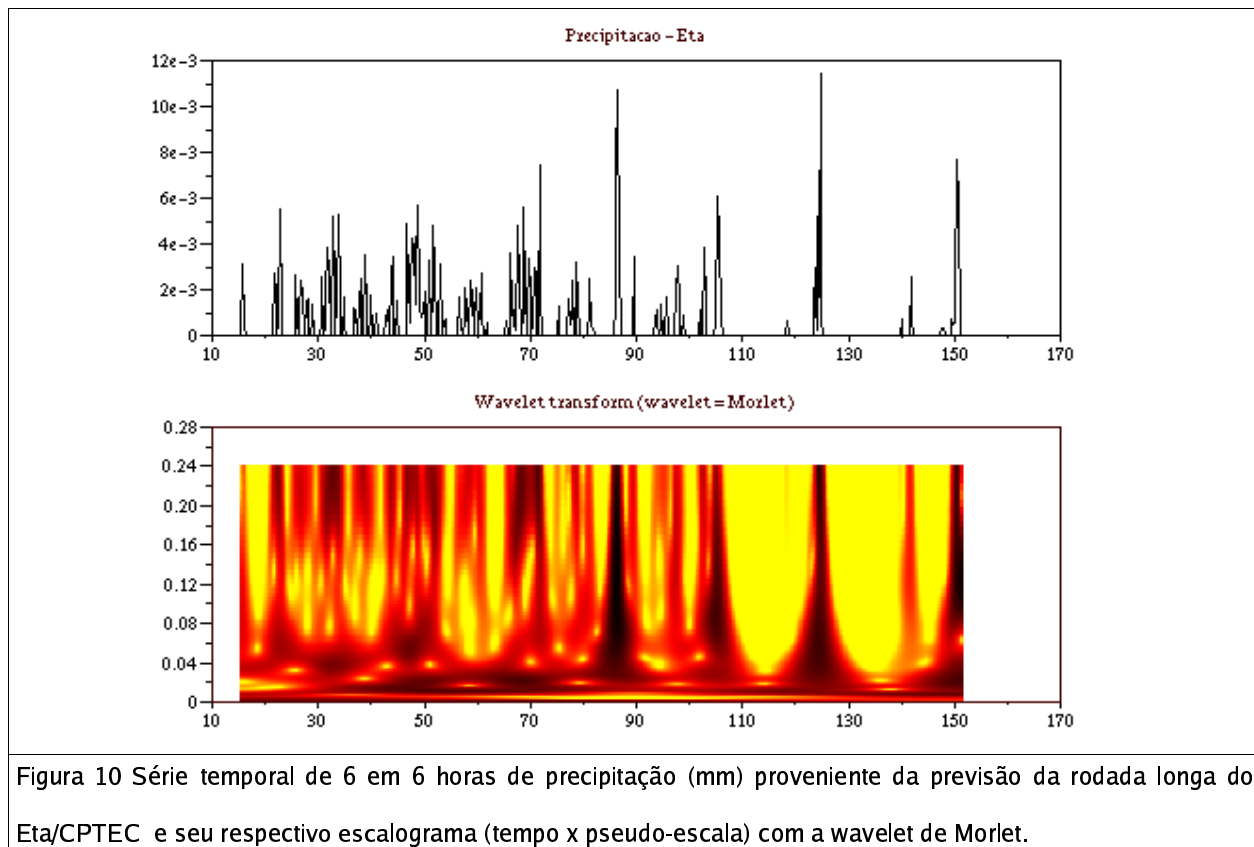


Figura 10 Série temporal de 6 em 6 horas de precipitação (mm) proveniente da previsão da rodada longa do Eta/CPTEC e seu respectivo escalograma (tempo x pseudo-escala) com a wavelet de Morlet.

Observando a Figura 9 e 10 geradas com base nos dados de precipitação do Modelo Regional Eta e nos dados de precipitação da torre de observação, percebe-se que a um maior volume e variação de precipitação no Modelo Regional Eta do que na torre de observação . Além disso os períodos de precipitação previsto no modelo não correspondem bem aos obidos na torre.

A visualização da transformada wavelet contínua de Morlet no Scilab/Fraclab não é satisfatória pois não é possível determinar a energia associada aos campos plotados e sendo assim dificulta a análise desses escalogramas de forma comparativa. Para solucionar esse desafio, o bolsista está aprendendo a visualizar os campos do escalograma na ferramenta openDX .

4. Comentários Finais

4.1. Fase atual da Pesquisa

O bolsista estudou nos primeiros 6 meses de bolsa e já está familiarizado com a programação no ambiente Scilab/Fraclab, em Linguagem C e em Grads. O bolsista já domina bem a parte de usuário no GNU/LINUX e já processou os dados 1D, formatando tanto dados observacionais de resposta lenta e rápida quanto os do modelo Eta em rodadas longas para o ambiente Scilab/Fraclab. Estão sendo efetuadas análises e testes com os dados observacionais e do modelo no período da Campanha do Pantanal, na estação experimental da UFMS em Passo do Lontra (série 1D) e com modelo Eta em rodadas longas .

4.2. Trabalhos Futuros

Melhorar a visualização dos escalogramas gerados utilizando wavelet continua de Morlet no estudo dos campos meteorológicos e da evolução temporal deste no caso 1D, utilizando a ferramenta OpenDX. Analisar os resultados processados em rodadas longas e curtas.

Estudar uma transformada wavelet adequada no caso 2D aos dados a serem tratados. Efetuar a transformada wavelet 1D e 2D para avaliar as características multiescala do campo previsto em longas e curtas rodadas do modelo e os campos efetivamente observados, baseando-se na metodologia proposta por Biggs e Levine (1997). Inicialmente isso será realizado em pontos teste-chave, para se avaliar a multiescala temporal, e em campos espaciais para se avaliar a multiescala espacial.

5. Referências Bibliográficas

Blaz, R.; Domingues, M. O.; Mendes Jr., O. **Introdução à entrada de dados no Opendix: formatos ".DX", "General" e ".GRB"**. São José dos Campos: INPE, 2003. 62 p. (INPE-9558-NTC/351).

Biggs, W. M. and Levine. R. A. Wavelets an field forecast verification, **Monthly Weather Review**, 1997, 125(6), 1329-1341.

Domingues, M. O.; Mendes, O. Jr.; Costa, A. M., **Algumas aplicações wavelet na análise de sinais atmosféricos, Processamento de Imagens e Wavelet**, SBMAC, 2003, Segundo Congresso Temático de Dinâmica e Controle, <http://black.rc.unesp.br/dincon2003/>, ISBN: 85-86883-15-8.

Ibanez, M. M.; Stephany, S.; Domingues, M. O.; Mendes, O. J. **Introdução à visualização de dados atmosféricos utilizando OpenDX**. INPE, Brasil, 2005.

Josiane, F. B.; Sin Chan Chou; José, R. R.; Gomes, J. L.; Uma avaliação da previsão de tempo do modelo Eta para a América do sul, **Revista Brasileira de Meteorologia**, v.20, n.1, 59-70, 2005.

Thompson, D.; Braun, J.; Ford, R. **OpenDX Path to Visualization**. Missoula, MT: Visualization and Imagery Solution, Inc, 2001. 207p, <<http://www.vizsolutions.com>>. 1

Apêndice A

Neste apêndice são apresentados os programas utilizados para formatação dos dados: Script em Grads, com finalidade de gerar os dados do Modelo Regional Eta:

```
'reinit'  
'open /rede/etag01/eta2d/season/grb/2002011512/grbl40ganl2002011512.ctl'  
tempo=1  
while (tempo<546)  
  'set t 'tempo  
  'set lon 303'  
  'set lat -19.56'  
  'clear'  
  'q time'  
  res=subwrd(result,3)  
  hora=substr(res,1,2)  
  dia=substr(res,4,2)  
  mes=substr(res,6,3)  
  ano=substr(res,9,4)  
  hora=hora/24  
  'set display color white'  
  'd temp'  
  valort=subwrd(result,4)  
  'c'  
  'd uvel'  
  valoru=subwrd(result,4)  
  'c'  
  'd vvel'  
  valorv=subwrd(result,4)  
  'c'  
  'd umes'  
  valorq=subwrd(result,4)  
  'c'  
  'd psic'  
  valorps=subwrd(result,4)  
  'c'  
  'd prec'  
  valorpr=subwrd(result,4)  
  dataj=ano 'dia' 'mes' 'hora' 'valort' 'valoru' 'valorv' 'valorq' 'valorps' 'valorpr'  
  command=write('/tmp_mnt/home/pvital/eta_regional/gera_dados/dadoseta.dat',dataj,append)  
  tempo=tempo+1  
endwhile
```

Programa em linguagem C, com a finalidade de trocar a data dos dados gerados pelo script em Grads.

```
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
  
int main(int argc, char *argv[])  
{  
  char cano[40],cdia[40],cmes[40],chora[40],temp[40],uvel[40],vvel[40],umes[40],pslc[40],prec[40];  
  int d, diaaux=32, diaantfeb=1,diaantmar=1,diaantapr=1,diaantmay=1,dia;
```

```

float ddia=0,hora=0,mes=0,diafeb,diamar,diaapr,diamay;
FILE *fpin,
    *fpout;
if( (fpin=fopen("dadoseta.dat","r")) == NULL)
{
    printf("Erro ao abrir arquivo dadostorre.dat!");
    exit(1);
}
if( (fpout=fopen("dadosetadada.dat","w")) == NULL)
{
    printf("Erro ao abrir arquivo dadostorredada.dat!");
    exit(1);
}
printf("arquivos abertos\n");
diafeb=32;
diamar=60;
diaapr=91;
diamay=121;
while( fscanf( fpin, "%s%s%s%s%s%s%s%s%s%s%s", cano, cdia, cmes, chora, temp, uvel, vvel, umes,
pslc, prec) != EOF)
{
    mes= atoi(cmes);
    hora = atof(chora);
    dia = atoi(cdia);
    if (mes == 1)
    {
        ddia = dia + hora;
        fprintf(fpout,"%s %5.2f %s %s %s %s %s %s\n", cano, ddia, temp, uvel, vvel, umes, pslc,
prec);
    }
    if (mes == 2)
    {
        printf("dia=%d",dia);
        printf("diaantfeb=%d\n",diaantfeb);
        if (dia != diaantfeb)
        {
            diafeb = diafeb + 1;
            printf("diafeb=%f",diafeb);
        }
        ddia = diafeb + hora;
        fprintf(fpout,"%s %5.2f %s %s %s %s %s %s\n", cano, ddia, temp, uvel, vvel, umes, pslc,
prec);
        diaantfeb = atoi(cdia);
    }
    if (mes == 3)
    {
        printf("dia=%d",dia);
        printf("diaantmar=%d\n",diaantmar);
        if (dia != diaantmar)
        {
            diamar = diamar + 1;
            printf("diamar=%f",diamar);
        }
        ddia = diamar + hora;
        fprintf(fpout,"%s %5.2f %s %s %s %s %s %s\n", cano, ddia, temp, uvel, vvel, umes, pslc,
prec);
    }
}

```

```

        diaantmar = atoi(cdia);
    }
    if (mes == 4)
    {
        printf("dia=%d", dia);
        printf("diaantapr=%d\n", diaantapr);
        if (dia != diaantapr)
        {
            diaapr = diaapr + 1;
            printf("diaapr=%f", diaapr);
        }
        ddia = diaapr + hora;
        fprintf(fpout, "%s %5.2f %s %s %s %s %s %s\n", cano, ddia, temp, uvel, vvel, umes, pslc,
prec);
        diaantapr = atoi(cdia);
    }
    if (mes == 5)
    {
        printf("dia=%d", dia);
        printf("diaantmay=%d\n", diaantmay);
        if (dia != diaantmay)
        {
            diamay = diamay + 1;
            printf("diamay=%f", diamay);
        }
        ddia = diamay + hora;
        fprintf(fpout, "%s %5.2f %s %s %s %s %s %s\n", cano, ddia, temp, uvel, vvel, umes, pslc,
prec);
        diaantmay = atoi(cdia);
    }
    }
    fclose(fpin);
    fclose(fpout);
    return 0;
}

```

Programa desenvolvido em Linguagem C, com a finalidade de converter a temperatura de kelvin para Celsius:

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>

int main(int argc, char *argv[])
{
    char canoeta[40], cdataeta[40], ctemp[40], cuv[40], cvv[40], cq[40], cps[40], cpr[40];
    int d;
    float tempconv, temp, q, ps, pr;
    FILE *fpin,
        *fpout;
    if( (fpin=fopen("dadoseta.dat", "r")) == NULL)
    {
        printf("Erro ao abrir arquivo eta20020115.dat!");
        exit(1);
    }
}

```

```

    }
    if( (fpout=fopen("etaconvt.dat","w")) == NULL)
    {
        printf("Erro ao abrir arquivo etaconvt.dat!");
        exit(1);
    }
    printf("Arquivos Abertos");

    for (d=1;d<550;d++)
    {
        if (fscanf(fpin,"%s%s%s%s%s%s%s%s",canoeta,cdataeta,ctemp,cuv,cv,cq,cps,cpr) !=
EOF)
        {
            temp = atof(ctemp);
            q = atof(cq);
            ps = atof(cps);
            pr = atof(cpr);

            tempconv = temp-273.15;

            fprintf(fpout,"%s %s %5.2f %s %s %s %s %s\n", canoeta, cdataeta, tempconv,
cuv, cv, cq, cps, cpr);
        }
        fclose(fpin);
        fclose(fpout);
        return 0;
    }
}

```

Programa desenvolvido em linguagem C, com a finalidade de converter os umidade especifica em umidade relativa:

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main(int argc, char *argv[])
{
    char canoeta[40],cdiahoraeta[40],ctempeta[40],cumeseta[40],cpslceta[40];
    int d;
    float umes=0,pslc=0,temp=0,UR,e,es;
    FILE *fpin,
        *fpout;
    if( (fpin=fopen("dados_convert.dat","r")) == NULL)
    {
        printf("Erro ao abrir arquivo dados_convert.dat!");
        exit(1);
    }
    if( (fpout=fopen("eta_convert.dat","w")) == NULL)
    {
        printf("Erro ao abrir arquivo dadostorredata.dat!");
        exit(1);
    }
    printf("arquivos abertos\n");

    for (d=1;d<546;d++)

```

```

{
    if (fscanf(fpin, "%s%s%s%s%s", cano, diahora, cpslc, cumes, ctemp) != EOF)
    {
        pslc = atof(cpslc);
        umes = atof(cumes);
        temp = atof(ctemp);

        e = (umes*pslc)/0.622;
        es= (6.112)*(exp((17.67*temp)/(temp+243.5))); // Wexler formula (bolton 1980)
        UR= (e/es)*100;

        fprintf(fpout, "%s %s %5.2f \n", ano, diahora, UR);
        printf("DADOS GRAVADOS ! %darquivos abertos\n", d);

    }
}
fclose(fpin);
fclose(fpout);
// system ("pause");
return 0;
}

```

Apêndice B

Programa desenvolvido com a finalidade de calcular a Transformada Wavelet de Morlet no ambiente scilab/fraclab unidimensional.

```
//Carregendo as Bibliotecas Fraclab
;exec("/home/paulov/fraclab-2.0/builder.sce");
;exec("/home/paulov/fraclab-2.0/loader.sce");
//localizacao do arquivo de dados
DIRDAT = "/home/paulov/dados_tratados";
//diretorio dos dados
chdir(DIRDAT);
//leitura interativa do arquivo
printf("Informe o nome do arquivo...");
arqin = read(%io(1),1,1,'a');
xx=fscanfMat(arqin);
NN=size(xx);
N = NN(1);
smin=2;
smax=log2(N);
ns= 128;
WaveType = 'Morlet' ;
t = xx(:,2)+xx(:,3); // aqui voce deve alterar para visualizarmos a coluna dos dias no calen. juliano
if WaveType == 'Morlet' | WaveType == 'morlet'
    wave = 8*%i ;
elseif WaveType == 'Mexican' | WaveType == 'mexican'
    wave = 0 ;
else
    break;
end
//teste de que arquivo, que tipo de variavel está sendo feita a transformada
if arqin == "dadosT.dat"
    name=['Temperatura do ar - Torre','temperatura do ar - Eta'];
    namearq=['Ttorre','TEta'];
end
if arqin == "dadospr.dat"
    name=['Precipitacao - Torre','Precipitacao - Eta'];
    namearq=['prTorre','prEta'];
end
if arqin == "dadosps.dat"
    name=['Pressao - Torre','Pressao - Eta'];
    namearq=['psTorre','psEta'];
end
if arqin == "dadosUR.dat"
    name=['Umidade Relativa - Torre','Umidade Relativa - Eta'];
    namearq=['URTorre','UREta'];
end
t=xx(:,2);
for i=3:4
j=i-2;
x = xx(:,i);
[wt,scale,ffreq] = contwtmir(x,2^(-smax),2^(-smin),ns,wave) ;
xselect();xbasc();
xsetech([0,0,1,0.5]);
```

```
plot2d(t,x)
xtitle(name(j));
xsetech([0,0.5,1,0.5]);
Zinf = 10^(-28/10) ;
TheMat = (abs(wt).^2) ;
D=TheMat;
TheMat = TheMat./max(max(TheMat)) ;
TheMat = log10(max(ones(ns,N).*Zinf,TheMat)) ;
//viewmat(TheMat,t,linspace(smin,smax,ns)) ;
viewmat(TheMat,t,ffreq) ;
xtitle('Wavelet transform (wavelet = '+ WaveType+')');
xsetech([0 0 1 1])
    arq1=sprintf('SpectroWavelet1%c.dat',namearq(j));
    arq0=sprintf('SpectroWavelet0%c.dat',namearq(j));
fprintfMat(arq0,D,'%f');
fprintfMat(arq1,TheMat,'%f');
parada = read(%io(1),1,1,'(a)');
end
```