

Estudo da Estimativa de Precipitação Utilizando diferentes Métricas de Interpolação Espacial para a Combinação de dados de Satélite e Observações *in situ*

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

Igor Frassoni Guedes dos Santos (UNIP, Bolsista PIBIC/CNPq)
E-mail: igor.frassoni@cptec.inpe.br

João Gerd Zell de Mattos (DMD/CPTEC/INPE, Orientador)
E-mail: joao.gerd@cptec.inpe.br

Julho de 2014

Sumário

| | |
|----------------------------------|----|
| Resumo..... | 3 |
| Introdução | 3 |
| Desenvolvimento | 4 |
| Métodos de Coleta de Dados | 4 |
| Softwares e linguagens | 5 |
| Métodos de Interpolação | 5 |
| Implementação do Algoritmo | 9 |
| Resultados de Discussões | 9 |
| Conclusão | 10 |
| Referências Bibliográficas..... | 10 |
| Lista de Anexos | 10 |

1.0. RESUMO

O projeto de pesquisa apresentado tem como objetivo realizar um estudo dos diferentes métodos de interpolação utilizados na combinação dos dados utilizados na estimativa de precipitação e, a partir deste estudo, desenvolver um produto para a geração de precipitação, baseado na combinação dos dados do satélite TRMM (*Tropical Rainfall Measuring Mission*) e de dados de estações meteorológicas.

No primeiro ano da pesquisa, o bolsista focou-se na compreensão dos métodos utilizados para a obtenção e armazenamento dos dados de precipitação, por meio da leitura de artigos científicos e desenvolvimento de rotinas em diferentes linguagens de programação (como fortran, GrADS e shell scripts), sendo avaliado através de apresentações e resumos.

No segundo ano, o bolsista iniciou o desenvolvimento da rotina computacional com finalidade de desenvolver o produto citado anteriormente.

Embora, o bolsista tenha se familiarizado com os dados de precipitação, compreendendo as formas de coleta, tratamento, armazenamento e disseminação dos dados de precipitação, bem como ter compreendido razoavelmente bem o tema os objetivos propostos, não foi possível concluir o projeto, pois o bolsista apresentou algumas deficiências para o desenvolvimento da metodologia e aplicação das técnicas empregadas no projeto para a elaboração da rotina computacional.

2.0. INTRODUÇÃO

O conhecimento da distribuição da precipitação é de grande importância para o planejamento de diversas atividades tanto econômicas como sociais, tais como agricultura, pecuária, geração de energia, entre outras. A baixa densidade de plataformas de coleta de dados meteorológicas, em algumas regiões, compromete significativamente os estudos relacionados aos sistemas meteorológicos. Sendo assim, variados produtos de precipitação obtidos por meio da combinação de estimativas de satélite e estações meteorológicas convencionais são gerados e disponibilizados a comunidade para o suprimento desta carência (veja Viltá et. al.). Entretanto, vários destes produtos apresentam problemas, que geralmente são associados aos métodos de interpolação utilizados. É comum verificar nas áreas de montanha um espalhamento horizontal incoerente nos campos de precipitação. Também, notam-se em algumas regiões, com uma rede esparsa de observações convencionais, áreas de intensa precipitação. Esse tipo de problema é causado principalmente pela metodologia de interpolação utilizada no processo. Em vista disso, o corrente projeto propôs um estudo sobre os métodos de interpolação utilizados na combinação dos dados de satélite e observações terrestres, com a finalidade de desenvolver um novo produto de precipitação onde os campos espaciais não apresentassem tais problemas.

3.0. DESENVOLVIMENTO

Como visto, o projeto iniciou-se no ano de 2012, com o período de finalização original em 2013. Este foi renovado até 2014. No primeiro ano do projeto, foram efetuadas diversas elaborações de resumos sobre assuntos relacionados a precipitação, mais especificamente sobre as formas de coleta de dados de precipitação. Este processo teve por finalidade a compreensão dos diferentes instrumentos utilizados para a coleta de precipitação, bem como os motivos pelos quais a rede de observações é esparsa e também os tipos de erros que podem estar presentes.

Basicamente, no primeiro ano de projeto, o bolsista voltou seus estudos primeiramente para a compreensão dos processos que englobam o processo da análise, coleta e tratamento de dados. Para tal, variados estudos foram realizados, tais como leitura de artigos científicos, elaboração de relatórios e resumos sobre o assunto. Neste ano, o bolsista também trabalhou com os dados, de forma a organizá-los, visando um contato inicial.

No segundo ano, o bolsista voltou seus esforços para a área do projeto que engloba o desenvolvimento da rotina computacional visando criar um novo produto de precipitação. Para tal, o bolsista utilizou-se de variados softwares, como por exemplo, Fortran, o uso da plataforma de desenvolvimento Matlab, e o OpenGrads, para visualização dos dados utilizados, bem como a interação de ambos.

A seguir, podem-se encontrar mais detalhadamente os assuntos estudados no decorrer do projeto.

3.1. Métodos de coleta de dados

Métodos de coleta de dados são basicamente a forma de como os dados são coletados, quais os instrumentos são utilizados. Este assunto foi estudado no primeiro ano do projeto, e pode ser visto com mais detalhes a seguir.

3.1.1. Pluviômetros/pluviógrafos

São basicamente instrumentos destinados a medida de precipitação ocorrida em determinado local. Diferem-se entre si apenas no fato de armazenamento dos dados coletados, onde o pluviógrafo possui a capacidade de gravar os dados coletados, já o pluviômetro não possui tal artifício. Estes instrumentos são amplamente utilizados ao redor do mundo, por serem baratos e de fácil manuseio, porém, esse sistema de coleta de dados torna-se muito dependente da quantidade de pluviômetros e de seu espalhamento em torno da região a ser feita a estimativa.

Além de depender da quantidade e de seu espalhamento na região, estes tipos de equipamentos possuem uma taxa de erro associada a eles, visto que são equipamentos onde geralmente utiliza-se de contato humano para a coleta desses dados (exceto quando os dados são enviados automaticamente para o banco de dados, PCD's). Há também os erros associados ao equipamento, que pode estar descalibrado, ou sofre interferência de algum fator externo do ambiente, como catástrofes naturais, onde este tipo de equipamento pode sofrer danos.

3.1.2. Satélites

A precipitação por satélite pode ser estimada a partir de radares, sensores infravermelhos, micro-ondas, entre outros. Este é um método de estimativa indireta, pois o satélite não detecta exatamente as gotas de chuva, o que ele faz é utilizar-se de algoritmos que associam a espessura e/ou a temperatura de brilho da nuvem com a taxa de chuva de nuvens precipitantes. Um dos dados utilizados neste projeto foi proveniente do produto de estimativa de precipitação produzida pelo TRMM (Tropical Rainfall Measuring Mission), o produto 3B42RT. Este produto utiliza-se da combinação de sensores de micro-ondas e infravermelho para estimar a precipitação.

O erro associado a este tipo de equipamento pode ser gerado de diferentes formas, por ser um equipamento do satélite, este pode sofrer danos no trajeto da Terra ao espaço, além de muitos outros tipos de interferência que possam vir a ocorrer no espaço, tanto físicas como magnéticas.

3.2. Softwares e Linguagens

No decorrer de todo o projeto, foi necessária a utilização de alguns softwares de visualização, cálculo e de programação, como o GrADS (*Grid Analysis and Display System*), Matlab, Fortran e Shell Script.

3.2.1. GrADS

GrADS (Grid Analysis and Display System) é uma ferramenta de manipulação e visualização de dados relacionados a ciência terrestre. Esta ferramenta foi utilizada para a introdução do bolsista aos dados de precipitação.

3.2.2. Matlab

Matlab é um software de alta performance voltado para o cálculo numérico. Foi utilizado na elaboração de rotinas computacionais a fim de gerar o produto proposto. Sua utilização pode ser vista no *Anexo 1*.

3.2.3. Fortran

Fortran (Formula Translator) é uma linguagem de programação voltada especialmente para cálculo numérico, e foi utilizada para criar rotinas que seriam usadas no projeto.

3.2.4. Shell Script

Shell script é uma linguagem de script utilizada no sistema operacional GNU/Linux, com diferentes interpretadores de comando. No presente projeto, foi utilizada para a implementação de rotinas computacionais a fim de atualizar o banco de dados do produto de precipitação proveniente do TRMM. Um exemplo pode ser visto no *Anexo 2*.

3.3. Métodos de Interpolação

Após o entendimento dos dados, sua forma de coleta, e um breve estudo dos softwares disponíveis para a visualização e manipulação dos dados de precipitação, iniciaram-se os estudos a respeito das técnicas de interpolação existentes. Este assunto foi visto no primeiro ano, mas sua utilização foi necessária em todo o decorrer do projeto.

3.3.1. Método dos Polígonos de Thiessen Este método assume que em qualquer ponto da bacia a precipitação é igual a medida do posto mais próximo. Nesse método é definida a área de influência de cada posto pluviométrico dentro da bacia hidrográfica. Esta técnica permite determinar, através da construção de polígonos, qual a área mais próxima de um ponto numa área, ou seja, unem-se todos os pares de pontos com segmentos de reta, e após, traça-se a perpendicular a cada segmento de reta. Forma-se assim um polígono que delimita uma área em que todos os pontos possuem o mesmo valor. Um exemplo de como este método funciona pode ser visto na *figura 1*. Este é um método básico, exato, e que não leva em conta os erros associados a interpolação, porém, de simples montagem e entendimento.

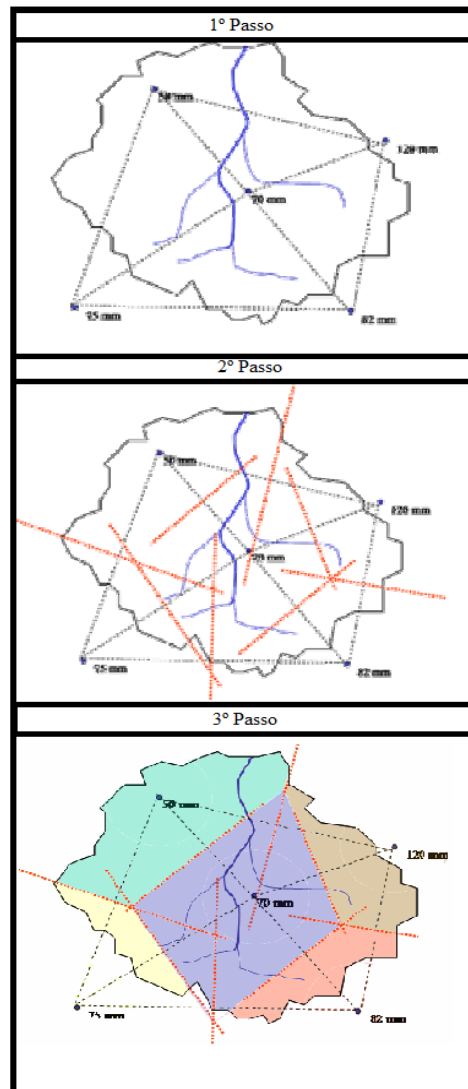


Figura 1: Exemplificação do método dos Polígonos de Thiessen

- Passo 1: Traçam-se linhas que unem os postos pluviométricos mais próximos entre si;
- Passo 2: Traçam-se linhas médias perpendiculares às linhas que unem os postos pluviométricos;
- Passo 3: Define-se a região de influência de cada posto pluviométrico a partir da área formada pela intersecção das mediatrizes das linhas que unem os postos pluviométricos;

3.3.2. Ponderação do Inverso das Distâncias

Este método utiliza-se do pressuposto que estações próximas exercem maior influência no ponto que se deseja interpolar. Assim, uma estação próxima possui maior peso no cálculo do que uma estação mais distante.

O valor “p” é determinado minimizando o erro médio quadrático, que é a estatística calculada por um procedimento de validação cruzada, neste procedimento, por sua vez, tem-se que cada ponto observado é removido e comparado com o predito para aquele local.

Este método de interpolação é exato, onde poucas decisões são tomadas em relação aos parâmetros do modelo. Porém, nele não é realizada uma avaliação da predição dos erros, que pode gerar o efeito “olho de boi” em torno da localização do dado, esse efeito, é caracterizado como um espalhamento incoerente da precipitação no campo ao redor do dado, um exemplo pode ser visto na *figura 2*.

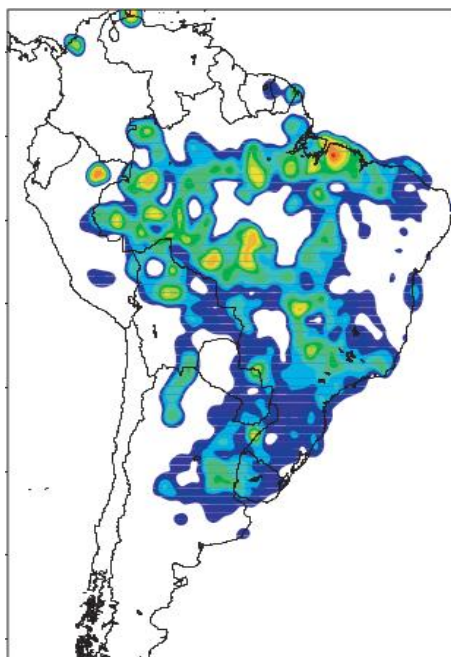


Figura 2: Exemplo de espalhamento incoerente da precipitação, próximo ao Norte do Estado do Pará

3.3.3. Interpolação Polinomial

A interpolação polinomial define que uma equação global seja adequada ao estudo da área de interesse, usando uma função polinomial algébrica ou trigonométrica. Este método de interpolação possui como desvantagem, em alguns casos, dependendo da ordem da função polinomial, o excessivo uso computacional necessário para sua aplicação, e possui como vantagem a remoção dos efeitos de tendências globais (i.e. Efeitos de vento, poluição, etc) de longa duração, criando superfícies que descrevem processos físicos.

3.3.4. Interpolação Multi-quadrática

Este método de interpolação define que a influência de cada amostra é representada por cones quadráticos como uma função das coordenadas desses pontos. A estimativa de certo ponto é então obtida pela soma das contribuições de todos os cones quadráticos.

3.3.5. Interpolação Ótima

Este método de interpolação determina que os pesos aplicados as estações são determinados a partir da minimização da variância do erro da interpolação. Ou seja, faz com que diminuam as possibilidades do valor interpolado ser distante do esperado. O cálculo utilizado neste projeto, se dá a partir das matrizes de covariância de erros, como o de background (B, valores de entrada, do satélite), da Observação (R, dados de estações). A Equação que define o processo de interpolação Ótima pode ser visto a seguir:

$$W = B * H' * inv(R + H * B * H') \quad (1)$$

$$d = x_o - x_b \quad (2)$$

$$x_a = x_b + W * d \quad (3)$$

Onde x_a é o resultado da interpolação, d é o vetor Inovação, e W é a matriz de ganho. R e B são as covariâncias dos erros descritos anteriormente.

3.4. Implementação do Algoritmo

No segundo ano, foi iniciado os trabalhos na área que compreende a implementação do produto proposto no início do projeto. Para este processo, a ferramenta de cálculo numérico de alta capacidade Matlab foi utilizada. Esta ferramenta foi escolhida devido ao fato de este possuir alta performance no processamento de dados numéricos, como matrizes, que é basicamente a forma utilizada dos dados de precipitação.

O algoritmo criado possui o intuito de funcionamento da seguinte maneira:

- Carregar o arquivo de precipitação de estações
- Carregar o arquivo descritor (.ctl) para os dados do satélite
- Coletar as variáveis necessárias dos arquivos (dados de precipitação, coordenadas)
- Realizar os cálculos de covariância das matrizes (Erro de background, Erro de Observação), do Operador Observacional, a Matriz de ganho, o Inovação e o Resultado final da interpolação

Este processo pode ser visualizado no *Anexo 1*. Este, inicialmente possui a função de executar um único conjunto de arquivos, o que posteriormente seria agregado a uma versão 2, a qual já estaria pronta para executar vários arquivos.

3.5. Resultados e Discussões

O projeto no segundo ano, basicamente teve o objetivo de desenvolver o algoritmo proposto no início do projeto, deve-se considerar também que ocorreram também outras atividades voltadas na compreensão de algumas etapas a serem realizadas no algoritmo, como apresentações e relatórios.

O que pôde-se notar, é que houve um avanço considerável comparado com o primeiro ano de projeto, visto que o bolsista já foi capaz de desenvolver em parte, a rotina computacional. O algoritmo não foi concluído, devido ao fato de este se mostrar de alta complexidade para o bolsista, o qual falhou ao tentar desenvolver uma etapa, onde seria necessário criar uma rotina para diminuir o tempo de execução do script. Tal etapa deveria criar um filtro, como forma de selecionar estações a serem consideradas no momento do cálculo de covariâncias, que se torna necessário, devido ao fato de a rotina gerar matrizes consideravelmente grandes, de forma a diminuir o tempo de execução do cálculo dessas matrizes, e conseqüentemente diminuir o tempo de execução do script em geral.

Com base nos estudos realizados, pode-se notar é que variados métodos foram utilizados neste estudo, e o que se pode notar é que eles se assemelham em sua forma de determinar

o local a ser interpolado, possuindo variações apenas na determinação dos pesos a serem aplicados às estações.

A partir destes métodos, pode-se perceber que eles funcionam de determinadas formas, e que podem ou não ser úteis, o que vai depender será a aplicação destinada a eles. Por exemplo, caso haja uma área com dados a serem interpolados, e ainda, esse processo seja somente para exemplificação, pode-se utilizar um método que menos dispendioso, de fácil execução, como o método dos Polígonos de Thiessen, por exemplo. Em suma, o que define a usabilidade de um método é sua aplicação.

Entre esses métodos, devido as características funcionais, o que se destaca para a finalidade deste projeto, é o método de interpolação ótima, visto que este minimiza a variância do erro da interpolação, ou seja, diminui as possibilidades do valor interpolado ser distante do esperado. É claro que não deve-se descartar os outros métodos, visto que possuem suas particularidades, de modo que se tornam úteis dependendo da sua utilização.

O projeto não foi finalizado, faltando concluir alguns detalhes na rotina para a implementação do produto final.

4.0. CONCLUSÃO

O projeto apresentado foi um grande desafio proposto ao bolsista, visto que este não possuía conhecimento algum na área tratada, meteorologia e assimilação de dados, além dos softwares, este pôde aprender a utilizar diversos softwares como Fortran, Grads, além do sistema operacional GNU/Linux.

No segundo ano, o bolsista já adquiriu um vasto conhecimento a respeito de diversos fatores, tanto na área em que o estudo foi dirigido, como no ambiente global de trabalho.

Por mais que o projeto de iniciação científica não tenha sido finalizado, este foi de grande valia para o bolsista, visto que este adquiriu conhecimentos sobre determinados assuntos que são de grande valia para a sua vida acadêmica. Além disso, o projeto ajudou a ampliar a visão de trabalho, a qual engloba diversos fatores antes desconhecidos em relação ao ambiente de trabalho de um instituto de pesquisa.

5.0. REFERÊNCIAS BIBLIOGRÁFICAS

5.1. Tabios, G. Q. And Salas, J. D. (1985), **A COMPARATIVE ANALYSIS OF TECHNIQUES FOR SPATIAL INTERPOLATION OF PRECIPITATION**. JAWRA Journal of the American Water Resources Association, 21: 365-380. doi: 10.1111/j.1752-1688.1985.tb00147.x

5.2. Vila, Daniel A., Luis Gustavo G. De Gonçalves, David L. Toll, José Roberto Rozante, 2009: **Statistical Evaluation of Combined Daily Gauge Observations and Rainfall Satellite Estimates over Continental South America**. J. Hydrometeor, 10, 533-543. doi: <http://dx.doi.org/10.1175/2008JHM1048.1>

5.3. Rozante, J. R. ; Moreira, D. S. ; de Gonçalves, L.G.G. ; Vila, Daniel. **Combining TRMM and Surface Observation Precipitation: Technique and Validation over South American**. *Weather and Forecasting*, 2010.

5.4. **Site de discussões com questões referentes ao software**
(Disponível em: <http://stackoverflow.com/questions/tagged/matlab>)

6.0. LISTA DE ANEXOS

6.1. Anexo 1: Script criado no Matlab, para a implementação do produto de precipitação

6.2. Anexo 2: Rotina em Shell Script para atualização de dados de precipitação

Anexo 1

```
% Interpolação Ótima - versão 1.0
% Teste em 1 Único arquivo

close all; clear all;
%% Leitura dos arquivos (.bin - TRMM e .dat - ESTAÇÕES)
addpath /opt/matlab/arquivos/SA_daily/
addpath /opt/matlab/arquivos/2004_BR/

read_grads('01012004.ct1');

load prec0101200412.dat;

%% Atribuição de valores

xa = prec;

xb = xa;

xo = prec0101200412(:, [4]);

xax = header.XDEF.vec; % coordenadas dos pontos de grade
xay = header.YDEF.vec;
tamx = header.XDEF.num;
tamy = header.YDEF.num;

xox = prec0101200412(:, [2]); % coordenadas das observações
xoy = prec0101200412(:, [3]);

xa = xa';
xo = xo';
xb = xb';

Ny = 246; % Tamanho em Y (pontos de grade)
Nx = 190; % Tamanho em X (pontos de grade)

NxNy = Nx * Ny;
Nobs = numel(xo);

for i=1:Nobs % Posição dos dados de Observação
    pos0(i) = sqrt((xox(i))^2+(xoy(i))^2);
end

for i=1:tamx % Posição dos dados de Análise
    for j=1:tamy
        k = i*j;
        posA(k) = sqrt((xax(i))^2+(xay(j))^2);
    end
end
end
```

```

%% Matriz de Covariância do Erro de Background (B)
Ny = 246;
Nx = 190;
NxNy = Nx*Ny; % =46740

% procedimento para calcular as distâncias em torno do ponto desejado

NY9 = Ny*9;
B = sparse(Nx, NY9);
val1 = 1;
val2 = 1;
conta2=0;
media = mean2(xa);

% calculo do desvio padrão, usado posteriormente no cálculo de B
for nx=1:Ny;
    for ny=1:Nx;
        conta = (xa(nx, ny)-media)^2;
        conta2 = conta+conta2;
    end
end

desvioP = sqrt(conta2/(NxNy-1));

% -----
tic
for nx=1:Nx;

    for ny=1:Ny;

        inicioX=nx-1;
        fimX=nx+1;

        inicioY=ny-1;
        fimY=ny+1;

        if nx==1
            inicioX=nx;
        end
        if nx==Nx
            fimX=Nx;
        end
        if ny==1
            inicioY=ny;
        end
        if ny==Ny
            fimY=Ny;
        end
    end
end

```

```

for nnx=inicioX:fimX

    for nny=inicioY:fimY

        dx = xax(nx) - xax(nnx);
        dy = xay(ny) - xay(nny);
        dist = sqrt(dx^2 + dy^2);
        conta = (desvioP * exp((-dist/300)));
        B(val1, val2) = conta;

        val2=val2+1;

    end

end

end

val2=1;

val1=val1+1;
end

toc

% -----

% {
prec_vec = reshape(prec', 1, NxNy); %transformando a matriz em vetor

% while para criar a linha 2 com as coordenadas X
nn=1;
while nn <= NxNy

for nx=1:Nx;
    prec_vec(2, nn)=xax(nx);
    nn=nn+1;
end

end

% while para criar a linha 3 com as coordenadas Y
nn=1;
while nn <= NxNy

for ny=1:Ny;
    prec_vec(3, nn)=xay(ny);
    nn=nn+1;
end

end
end

```

```

nd=1;

NXNY = NxNy*7; % 3 + 1 + 3
dx = sparse(1, NXNY);

tic
for n1x=1:NxNy;

    n1xminus=n1x-3; % Raio 3
    n1xplus=n1x+3;

    if (n1xminus <= 0)
        n1xminus=n1x;
    end

    if (n1xplus >= (NxNy-3))
        n1xplus=NxNy;
    end

    for n2x=n1xminus:n1xplus;

        dx(1, nd)=prec_vec(2, n1x)-prec_vec(2, n2x);
        nd=nd+1;
    end

end
toc

%}

%[
% Mesmo esquema utilizado no script de Exemplo (ana_t500)
% Cria a matriz B

Ny = 246;
Nx = 190;
NxNy = Nx*Ny; % =46740

B = sparse(NxNy, NxNy) % aqui ocorre um erro (Maximum variable size allowed by the program is
exceeded.)

% PS: arrumei o erro que dava da memória
% utilizando a função sparse()

for n1x = 1:Nx;
for n2x = 1:Nx;
    dx12(n1x, n2x) = xax(n1x) - xax(n2x);
end
end

for n1y = 1:Ny;

```

```

for n2y = 1:Ny;
    dy12(n1y, n2y) = xay(n1y) - xay(n2y);
end
end

r12 = sqrt(dx12.^2+dy12.^2);

B(n1, n2) = 0.8 * exp((-1/2)*(r12./300));

*****
%{
dx2 = repmat(xax, length(xax), 1);
dy2 = repmat(xay, length(xay), 1);

dx2 = (dx2 - dx2').^2 / 90000;
dy2 = (dy2 - dy2').^2 / 90000;

n1x = 1:Nx;
n2x = 1:Nx;
for n1y = 1:Ny
    Y1 = (n1y-1)*Nx;
    for n2y = 1:Ny;

        Y2 = (n2y-1)*Nx;
        DY = dy2(n1y, n2y);
        n1 = Y1 + n1x;
        n2 = Y2 + n2x;
        r12 = sqrt(dx2 + DY);
        B(n1, n2) = 0.8 * exp((-1/2)*(r12));

    end
end
%}
return
%}

%% Matriz de Covariância do Erro de Observação (R)

Ny = numel(xox); % = 898
Nx = numel(xoy);
NxNy = Nx*Ny; % =806404

R = sparse(NxNy, NxNy);
dx12=zeros(Nx, Nx);
dy12=zeros(Ny, Ny);

for n1x = 1:Nx;
for n2x = 1:Nx;
    dx12(n1x, n2x) = xox(n1x) - xox(n2x);
end
end

```

```

for n1y = 1:Ny;
for n2y = 1:Ny;
    dy12(n1y, n2y) = xoy(n1y) - xoy(n2y);
end
end

r12 = sqrt(dx12.^2+dy12.^2);

R = 0.8 * exp((-1/2)*(r12./300));

size(R)

%{
for n1y = 1:Ny
for n1x = 1:Nx
    n1 = (n1y-1)*Nx+n1x;
for n2y = 1:Ny
for n2x = 1:Nx

    n2 = (n2y-1)*Nx+n2x;
    dx12 = xox(n1x) - xox(n2x);
    dy12 = xoy(n1y) - xoy(n2y);
    r12 = sqrt(dx12^2+dy12^2);
    R(n1, n2) = 0.8 * exp((-1/2)*(r12/300));

end
end

end
end

%}

%% Operador Observacional (H)

Ny = 246;
Nx = 190;
NxNy = Nx*Ny;

H = sparse(Nobs, NxNy);

for nobs=1:(Nobs-1)

    valor = (posA(nobs+nobs) - pos0(nobs))/(posA(nobs+nobs) - posA(nobs));
    H(nobs, nobs) = valor;

end

for nobs=1:Nobs

    valor = (pos0(nobs) - posA(nobs))/(posA(nobs+nobs) - posA(nobs));
    H(nobs, nobs+1) = valor;

```



```
end
```

```
%% Matriz de Ganho
```

```
W = B*H' *inv(R+H*B*H') ;
```

```
%% Inovação (d)
```

```
d = xo - xb
```

```
%% xa
```

```
xa = xb + W * d
```

```
toc
```

Anexo 2

```
#!/bin/bash
```

```
#
```

```
# Deverão ser passados 4 argumentos:
```

```
# 1 - Versao do TRMM [3B42RT/3B42_V6]
```

```
# 2 - Data Inicial [yyyymmddhh]
```

```
# 3 - Data Final [yyyymmddhh]
```

```
# 4 - Incremento de tempo [+ -ndnhnm]
```

```
TRMM=${1}
```

```
DataInicial=${2} #"1998010100"
```

```
DataFinal=${3} #"2010123121"
```

```
Incremento=${4} #" +3h"
```

```
DIR="/letkf/backup_tariba/saldas/jgerd/FORCING/PRECIPITACAO/TRMM/${TRMM}"
```

```
ftp="ftp://disc2.nascom.nasa.gov/data/TRMM/Gridded/${TRMM}"
```

```
while [ ${DataInicial} -ne ${DataFinal} ]; do
```

```
Ano=${DataInicial:0:4}
```

```
amd=${DataInicial:0:8}
```

```
Mes=${DataInicial:4:2}
```

```
Dia=${DataInicial:6:2}
```

```
hor=${DataInicial:8:2}
```

```
Ano2=${DataInicial:2:2}
```

```
amd2=${DataInicial:2:8}
```

```
case ${TRMM} in
```

```

B42RT)
if [ ${DataInicial:0:4} -lt 2008 ]; then
file="3B42RT.${Ano}${Mes}${Dia}${hor}.bin"
else
file='3B42RT.'${Ano}.'${Mes}.'${Dia}.'${hor}'z.bin'
fi
;;
3B42_V6)
file="3B42.${amd2}.${hor}z.6.precipitation.bin"
;;

esac

if [ -e ${DIR}/${Ano}/${amd}/${file} ];then
echo "${file} OK"
else

if [ ! -d "${TRMM}/${Ano}/${amd}"]; then mkdir -p ${TRMM}/${Ano}/${amd}; fi

wget -c -t0 -O ${DIR}/${Ano}/${amd}/${file} ${ftp}/${Ano}${Mes}/${file}

if [ ! -e ${DIR}/${Ano}/${amd}/${file} ]; then
echo "${file}" >> ${DIR}/${Ano}_notfound.txt
else
echo "${file} GET OK !!"
fi

fi

DataInicial=`/home/jgerd/bin/inctime ${DataInicial} ${Incremento} %y4%m2%d2%h2`
done

```