



Ministério da
Ciência e Tecnologia



PARALELIZAÇÃO E ASSIMILAÇÃO DE DADOS PARA O SUPIM

RELATÓRIO FINAL DE ATIVIDADES DE INICIAÇÃO CIENTÍFICA PIBIC/INPE – CNPq/MCT

Cristiano Reis dos Santos
Bolsista PIBIC/INPE
Email: creis@inf.ufsm.br

Dr. Adriano Petry (Orientador)
Email: dr.adriano.petry@crs.inpe.br

CRS/INPE – MCT
Santa Maria, julho de 2011

DADOS DE IDENTIFICAÇÃO:

Bolsista:

Cristiano Reis dos Santos

Acadêmico do curso de Ciência da Computação – Centro de Tecnologia

Universidade Federal de Santa Maria – UFSM

Orientador:

Dr. Adriano Petry

Centro Regional Sul de Pesquisas Espaciais – CRS/INPE

Colaborador:

André Grahl Pereira

Acadêmico do curso de Ciência da Computação – Centro de Tecnologia

Universidade Federal de Santa Maria – UFSM

AGRADECIMENTOS

Agradeço à minha família pelo apoio e o amparo nos momentos difíceis. Mãe, irmã e minha linda sobrinha Vitória. À minha amada Ana Maria, pelo apoio e incentivo recebido todos os dias. Aos colegas pelo apoio e contribuições dadas ao longo do ano, em especial ao André e à Miriam. Ao Instituto Nacional Pesquisa Espacial (INPE) pela infraestrutura e oportunidade de pesquisa a mim proporcionada. Ao CNPq pelo apoio financeiro. Ao meu orientador, Dr. Adriano Petry pelo incentivo, conhecimento compartilhado e a oportunidade de pesquisa proporcionada, que é de grande importância para minha formação acadêmica.

RESUMO

Alguns sistemas físicos podem ser descritos por modelos matemáticos. Esses modelos podem ser usados para prever a dinâmica esperada para um determinado sistema real, a partir de um estado inicial. Muitas vezes, no entanto, o comportamento do sistema modelado acaba divergindo do comportamento do sistema real. Nesse momento, dados observacionais oriundos de instrumentos de medição podem ser apresentados ao modelo matemático, que os utiliza para que sua dinâmica seja aproximada ao verdadeiro estado do sistema. Esse processo é conhecido como Assimilação de Dados. O trabalho atual trata da codificação de métodos de assimilação de dados. Foram prototipados dois métodos de assimilação de dados: O método de Cressman e o Best Linear Unbiased Estimator (BLUE). Foram desenvolvidas versões em diferentes linguagens de programação. A primeira abordagem foi realizada com MATLAB, um software interativo de alto desempenho voltado para o cálculo numérico. A segunda abordagem utilizou a linguagem de programação C++. Além disso, um software foi desenvolvido para acessar uma base de dados e obter arquivos com informações sobre densidade eletrônica, para que esses dados possam ser utilizados pelo sistema de Assimilação de Dados.

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 2.1 – Representação abstrata do processo de assimilação de dados. .	13
Figura 2.2 – Representação gráfica do processo da análise de Cressman	15
Figura 2.3 – Representação da área de abrangência da IO.	18
Figura 3.1 – Análise de Cressman com observações a cada espaço de tempo.	21
Figura 3.2 – Análise de Cressman com observações a cada 4 espaços de tempo.	22
Figura 3.3 – Análise de Cressman com observações aleatoriamente espaçadas	23
Figura 3.4 – BLUE com observações a cada espaço de tempo.....	24
Figura 3.5 – BLUE com observações a cada 4 espaços de tempo	25
Figura 3.6 – BLUE com observações aleatoriamente espaçadas	26
Figura 3.7 – Somatório do erro quadrático.	28
Figura 3.8 – Arquivo de dados observacionais.....	30
Figura 3.9 – Densidade eletrônica de uma ionosonda localizada em Porto Rico e a previsão do SUPIM para uma região próxima.....	31

LISTA DE SIGLAS E ABREVIATURAS

INPE	Instituto Nacional de Pesquisas Espaciais
PNT	Previsão Numérica de Tempo
LABCCE	Laboratório de Computação Para o Clima Espacial
CRS	Centro Regional Sul de Pesquisas Espaciais
BLUE	Best Linear Unbiased Estimator
OI	Interpolação Ótima
NOAA	National Oceanic and Atmospheric Administration
SPIDR	Space Interactive Data Resource
SUPIM	Sheffield University Plasmasphere Ionosphere

LISTA DE SÍMBOLOS

x	<i>Model state vector</i>
xb	<i>Background model state</i>
xt	<i>True state</i>
y	<i>Observation vector</i>
Δx	Incremento da análise
B	Matriz de covariância dos erros de <i>background</i>
H	Operador de observação
K	Matriz de ganho
R	Matriz de covariância dos erros de observação

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	9
1.1. Motivação	10
1.2. Objetivo	10
1.3. Organização do texto	11
2 FUNDAMENTAÇÃO TEÓRICA	12
2.1. Assimilação de dados	12
2.2. Métodos de assimilação	13
2.2.1. Análise de Cressman	14
2.2.2. Best Linear Unbiased Estimator	16
2.2.3. Interpolação Ótima	17
3 ATIVIDADES DESENVOLVIDAS DURANTE O PERÍODO.....	19
3.1. Protótipo em MATLAB.....	19
3.1.1. Funcionamento do programa.....	20
3.1.2. Resultados.....	20
3.2. Protótipo em C++	26
3.2.1. Funcionamento do programa.....	27
3.2.2. Resultados preliminares	27
3.3. Aquisição de dados observacionais	29
3.3.1. Funcionamento do programa.....	30
3.3.2. Resultados Preliminares	31
4 CONCLUSÃO	32
4.1. Trabalhos futuros	33
REFERÊNCIAS BIBLIOGRÁFICAS.....	34
APÊNDICE A – PROTÓTIPO EM MATLAB: ANÁLISE DE CRESSMANN	35
APÊNDICE B - PROTÓTIPO EM MATLAB: BLUE.....	37

1 INTRODUÇÃO

Alguns sistemas podem ser realisticamente descritos por modelos matemáticos. Esses modelos são usados para prever o comportamento esperado para um determinado sistema, a partir de um estado inicial. A representação perfeita da realidade não é passível de representação, então, o que tem se tentado fazer é descrever a melhor representação possível de ser gerada matematicamente e computacionalmente.

O modelo e a condição inicial contêm incertezas em suas representações. Por esta razão, observações provenientes de diversas fontes são utilizadas com a finalidade de ajustar o modelo de previsão, diminuindo os erros e aumentando a exatidão. Isso é conhecido atualmente como assimilação de dados, que vem sendo desenvolvida e aprimorada a cada dia.

Uma boa previsão depende de dois fatores importantes: observações de boa qualidade, que nos dirão como o sistema parece estar se comportando, e um modelo matemático confiável, responsável por combinar as observações com o estado passado do ambiente em análise, para gerar a provável situação futura (Lahozet et.al., 2010).

Um dos grandes desafios encontrados para realizar a assimilação com uma grande quantidade de dados é a complexidade matemática envolvida nos cálculos presentes nos modelos. A complexidade aumenta consideravelmente de modelos mais simples para modelos mais complexos. É necessário um grande poder computacional para que seja possível realizar uma previsão numérica com qualidade e eficiência aceitáveis. Grandes centros computacionais, privados e governamentais dispõem de grande quantidade de hardware para processamento, como clusters e supercomputadores. Mesmo que esses centros possuam um alto poder de processamento, novas técnicas estão sendo pesquisadas e desenvolvidas com o objetivo de aperfeiçoar e aumentar a exatidão da informação gerada.

Como a maioria dos ambientes simulados se comporta de maneira pouco previsível, modificando-se constantemente, técnicas mais avançadas, que de alguma maneira consigam se adaptar a essa incerteza presente na informação utilizada, seriam de grande importância para os centros de pesquisa.

1.1. Motivação

Muitas áreas são altamente dependentes do poder de previsão da assimilação de dados. Entre elas esta a Previsão Numérica de Tempo (PNT) que utiliza modelos de previsão e detalhes sobre a atmosfera para simular seu estado de algumas horas até varias semana à frente. A Previsão Numérica de Tempo está diretamente relacionada à economia. A capacidade de prever acontecimentos climáticos pode evitar grandes perdas na área agropecuária. Governos realizam ações preventivas baseados na previsão de seca ou chuvas em grandes quantidades. Muitas escolhas para o futuro são baseadas na previsão que se espera encontrar no futuro.

Para aprimorar a capacidade do Laboratório de Computação para o Clima Espacial (LABCCE) do Centro Regional Sul de Pesquisas Espaciais (CRS – INPE) de realizar previsões de conteúdo eletrônico total, jugou-se necessário acrescentar dados assimilados na obtenção da previsão do conteúdo eletrônico total.

1.2. Objetivo

O principal objetivo deste trabalho de iniciação científica é auxiliar no desenvolvimento de uma ferramenta de assimilação de dados para melhorar a previsão do conteúdo eletrônico total (TEC), gerado atualmente pelo LABCCE, gerado atualmente pelo LABCCE, que é baseada no modelo físico Sheffield University Plasmasphere Ionosphere Model (SUPIM, 1970), computacionalmente implementado em código Fortran.

Os métodos de assimilação de dados que estarão presentes na ferramenta primeiramente serão desenvolvidos na linguagem de computação de alto nível

MATLAB (MATLAB, 2009). MATLAB apresenta um ambiente interativo que permite a visualização dos resultados em forma de gráficos. Seus gráficos serão uma das formas de apresentação e validação da assimilação.

Após o desenvolvimento dos métodos de assimilação de dados em MATLAB, os mesmos serão transcritos para a linguagem de programação C++. Isso será realizado para que o módulo de assimilação de dados possa ser acoplado ao sistema já desenvolvido, chamado SUPIM-DAVS (Petry, Souza e Velho, 2011).

Uma ferramenta para obtenção de dados de densidade eletrônica também foi desenvolvida, para que as informações geradas atualmente pelo LABCCE possam ser validadas, e os dados assimilados.

1.3. Organização do texto

O texto a está organizado em quatro capítulos. O capítulo 1 apresenta a ideia do projeto, assim como a motivação e o objetivo de trabalho de iniciação científica.

No capítulo 2 é realizada uma breve introdução sobre assimilação de dados. Após são mostrados os métodos de assimilação, utilizados para desenvolver a aplicação.

No capítulo 3 descreve se a metodologia utilizada no projeto de iniciação científica, e são apresentados alguns resultados obtidos até o presente momento. Resultados provenientes das duas abordagens utilizadas para codificar os algoritmos são mostrados. Uma alternativa para a falta de dados para teste é sugerida.

No capítulo 4 é realizada a conclusão e são sugeridos trabalhos futuros, para continuação do projeto.

2 FUNDAMENTAÇÃO TEÓRICA

2.1. Assimilação de dados

A assimilação de dados tem por finalidade combinar observações com um modelo de sistema dinâmico para estimar o atual e futuro estado do sistema (Lahozet et.al., 2010).

A assimilação de dados tenta manter o modelo o mais realístico possível. O comportamento caótico visto na atmosfera, por exemplo, nos impede de confiar em uma previsão inicial. Por esse motivo as observações são obtidas por diversos meios e combinadas com a versão mais nova do modelo gerando uma nova previsão mais condizente com a realidade do estado. Essa nova previsão é utilizada futuramente como o novo estado do modelo para realizar uma nova previsão.

O estado da atmosfera é representado como uma matriz coluna chamada de *state vector* x . O estado verdadeiro da atmosfera chamado de x_t , ou *truestate*, não é passível de representação. Então, o que realmente é utilizado como estado da atmosfera é a melhor representação possível da realidade. Outra importante informação utilizada na realização de uma assimilação de dados é o x_b , o *background* ou *firstguess*. O x_b representa a melhor estimativa que se tem disponível no momento da análise.

Para realizar uma análise, necessita-se de valores observados do ambiente. Isso é conhecido como *observation vector* y . O y tem por finalidade ajustar o modelo de previsão. A Figura 2.1 é uma representação superficial de como a assimilação é realizada.

Atualmente os modelos de assimilação de dados trabalham com uma grande quantidade de dados. A dimensão utilizada em modelos de previsão meteorológicos é de aproximadamente da ordem de 10^7 para a matriz coluna

x , definida anteriormente como o *state vector*. E o número de observações está na ordem de 10^5 (Lahoz et al., 2010). Os dados não são distribuídos de forma uniforme no espaço, geralmente existem buracos onde não há observações, ocasionando erros de previsão mais visíveis.

Todo sistema de assimilação faz uma previsão com uma margem de erros. A minimização dos erros é muito importante, levando muitos centros de pesquisa a desenvolver diferentes abordagens de tratamento de erros.

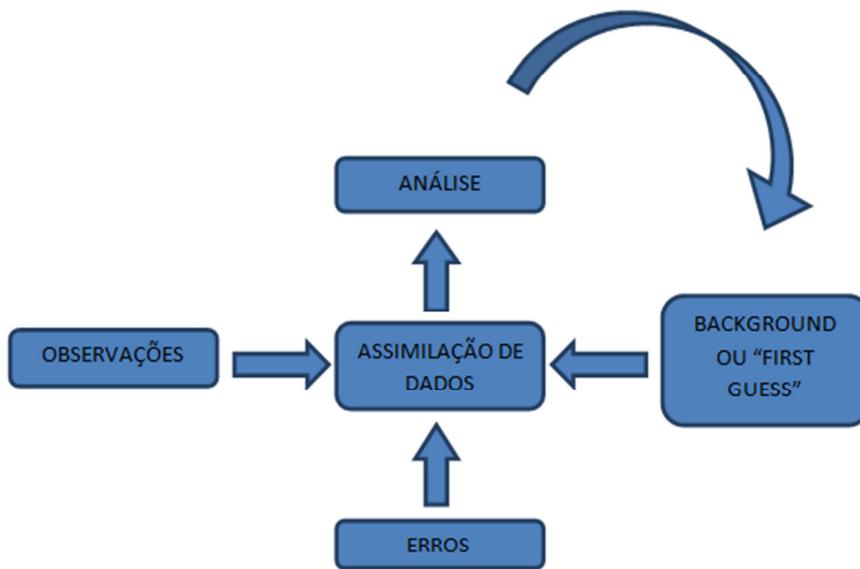


Figura 2.1 – Representação abstrata do processo de assimilação de dados.

2.2. Métodos de assimilação

Os métodos de assimilação de dados são baseados em técnicas de interpolação, que é um processo que utiliza valores de dados conhecidos para estimar valores de dados desconhecidos. Várias técnicas de interpolação são utilizadas atualmente nas ciências atmosféricas e oceânicas. Entre esses métodos podemos encontrar desde abordagens simples a métodos muito complexos de estimativa de valores. Entre as técnicas mais simples está a

Interpolação Linear, que requer o conhecimento de dois pontos e uma constante de variação entre eles. Com essas informações podem-se interpolar valores em qualquer região entre os pontos conhecidos. Outro método um pouco mais complexo, mas também simples de ser entendido é conhecido como Análise de Cressman.

Entre as técnicas mais avançadas para interpolação de dados encontram-se o Best Linear Unbiased Estimator (BLUE), a Interpolação Ótima (OI), o Filtro de Kalman, o 3D-VAR e o 4D-VAR, todos descritos em (Bouttier; Courtier, 1999). Esses métodos permitem que atualmente seja possível prever fenômenos climáticos com um alto grau de exatidão, que não seria possível apenas utilizando equipamentos de medição, seja pelo valor para aquisição desses equipamentos, pela grande área que deveria ser coberta por eles ou pelo tempo para analisar a informação e distribuí-la para os interessados com um alto grau de confiança.

2.2.1. Análise de Cressman

A Análise de Cressman foi desenvolvida em 1959 por George Cressman. É uma técnica de interpolação de estações de dados definidas através de altitude e longitude. Muitos passos são realizados através de dados em grade utilizando um raio para influenciar no aumento da precisão (Lahozet et.al., 2010).

O raio de influência é definido como o máximo raio de um ponto da grade para uma estação de dados ao qual foi observado um valor. Estações de dados além do raio de influência não tem qualquer influência no valor da grade. Para cada ponto da grade um novo valor é calculado baseado em um fator de correção. Esse fator de correção é determinado analisando cada estação dentro do raio de influência. Observações próximas ao ponto de grande carregam maior peso. Com a distância aumentando o peso de influência diminui.

$$x_{a(j)} = x_{b(i)} + \frac{\sum_{i=1}^n w(i,j)\{y(i) - x_{b(i)}\}}{\sum_{i=1}^n w(i,j)} \quad (2.1)$$

$$w(i,j) = \max\left(0, \frac{R^2 - d_{ij}^2}{R^2 + d_{ij}^2}\right) \quad (2.2)$$

A Equação 2.1 (Lahozet al., 2010), representa a análise de Cressman, onde x_a representa a nova análise. Assume-se que x_b é o *firstguess* ou *background*. $y(i)$ são os valores das observações e $w(i,j)$ representado pela Equação 2.2 definida em (Lahoz et al., 2010), é o peso da função que dependerá da distância entre os pontos de grade e a observação. R é uma constante definida pelo usuário, conhecido como raio de influência e $d(i,j)$ é a distância entre os pontos de grade e a observação. A figura 2.2 ilustra o resultado esperado para a análise de Cressmann aplicada a um *background* ilustrativo.

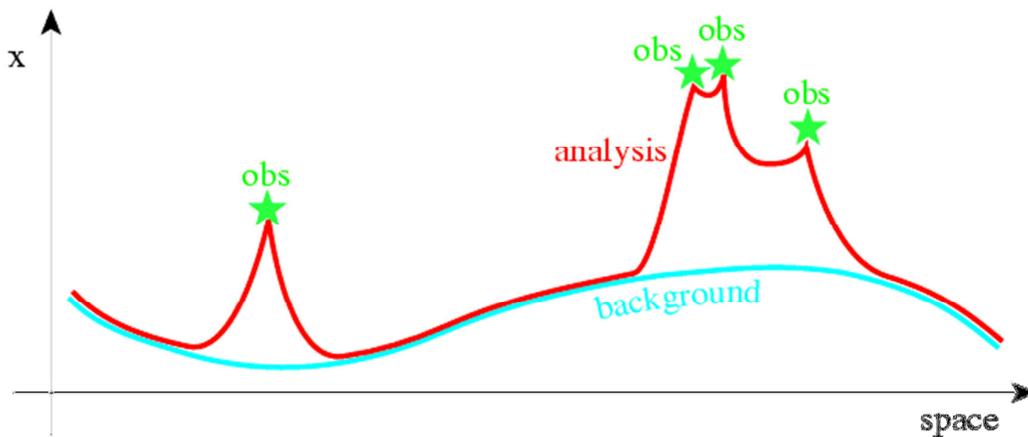


Figura 2.2 – Representação gráfica do processo da análise de Cressman
 Fonte: Adaptada de (Bouttier; Courtier, 1999).

2.2.2. Best Linear Unbiased Estimator

O BLUE, também conhecido como estimador dos mínimos quadrados, fornece a melhor ponderação estatística possível considerando os erros de observação e de background. O BLUE realiza a análise considerando todos os pontos presentes no modelo. Sua utilização para sistemas que contêm muitos dados pode ser proibitiva, pois o processamento requerido é muito elevado.

Serão apresentadas a seguir algumas notações utilizadas pelo BLUE, e pequenos comentários sobre essas notações, que foram adaptadas de (Bouttier; Courtier, 1999).

Nas equações 2.3 a 2.9 o x_t é a melhor representação possível da realidade. O x_b é denominado como *background*. O x_b representa a melhor estimativa que se tem disponível no momento da análise. O resultado de uma nova análise é x_a , que representa a nova previsão para o tempo atual e informações utilizadas na análise. A análise é obtida através da Equação 2.8. Para realizar uma análise, necessita-se de um número de valores observados do ambiente. Isso é conhecido como vetor de observações y . O vetor de observações y tem por finalidade ajustar o modelo de previsão. A matriz H é conhecida como operador de observação. Ele tem por finalidade conduzir o espaço do modelo para a mesma dimensão das observações. O operador K representado na Equação 2.9 é chamado de ganho, ou matriz de peso da análise. As matrizes B e R representam os erros do modelo. B é uma matriz de covariância que representa os erros de *background*. R é a matriz de covariância que representa os erros de observação. As matrizes são definidas a partir da Equação 2.3 e Equação 2.5, respectivamente.

$$B = \overline{(e_b - \bar{e}_b)(e_b - \bar{e}_b)^T} \quad (2.3)$$

$$e_b = x_b - x_t \quad (2.4)$$

$$R = \overline{(e_o - \bar{e}_o)(e_o - \bar{e}_o)^T} \quad (2.5)$$

$$e_o = y - H(x_t) \quad (2.6)$$

O problema da análise é encontrar uma correção δ_x (ou incremento da análise) que aproxime x_a o mais possível de x_t (Bouttier; Courtier, 1999), como mostrado na Equação 2.7.

$$x_a = x_b + \delta_x \quad (2.7)$$

$$x_a = x_b + K(y - H[x_b]) \quad (2.8)$$

$$K = BH^T(HBH^T + R)^{-1} \quad (2.9)$$

2.2.3. Interpolação Ótima

A Interpolação Ótima é uma variação Best Linear Unbiased Estimator (BLUE). Ela se diferencia dos mínimos quadrados por utilizar apenas observações que se encontram em uma região próxima dos pontos de análise. A principal hipótese na Interpolação Ótima é: para cada variável do modelo, somente algumas observações são importantes na determinação do incremento da análise (Bouttier; Courtier, 1999).

A grande vantagem da Interpolação Ótima é o menor custo computacional, decorrente da utilização de algumas observações selecionadas, e não todas como ocorre com o BLUE, como podemos visualizar na Figura 2.3.

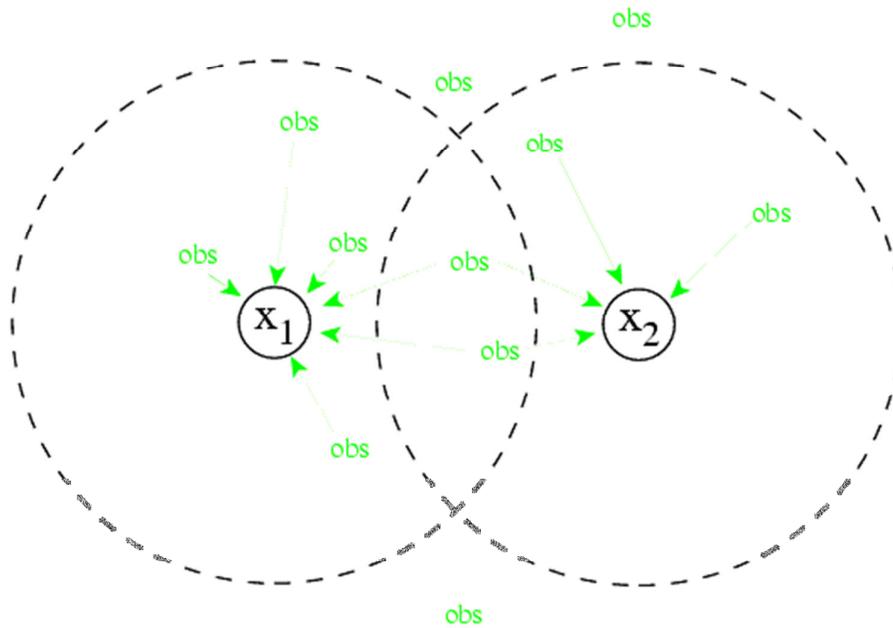


Figura 2.3 – Representação da área de abrangência da IO.
Fonte: Adaptada de (Bouttier; Courtier, 1999).

3 ATIVIDADES DESENVOLVIDAS DURANTE O PERÍODO

As atividades realizadas até o presente momento foram o desenvolvimento de um protótipo no ambiente MATLAB, C++ e uma ferramenta para obtenção de dados de densidade eletrônica com vistas a assimilação.

A seguir serão apresentadas mais detalhadamente essas atividades, e alguns testes que demonstram seus resultados.

3.1. Protótipo em MATLAB

Na primeira etapa do desenvolvimento da ferramenta de assimilação de dados, foi utilizada a linguagem MATLAB (MATLAB, 2009). Essa linguagem é do tipo interpretada, não necessita de compilação para executar uma aplicação desenvolvida em seu ambiente de programação. O ambiente MATLAB integra análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos em um ambiente fácil de usar, onde problemas e soluções são expressos somente como eles são descritos matematicamente. A facilidade na resolução de sistemas numéricos, acrescentada com o seu poder de visualização dos resultados são atrativos para o uso do MATLAB.

A aplicação desenvolvida utilizando MATLAB possibilitou que pudesse ser feita uma comparação entre os resultados disponibilizados pelo processo de assimilação de dados, e os valores que representam o estado verdadeiro. Por meio de seus gráficos é possível visualizar a exatidão ou a discrepância da assimilação realizada. Assim, é possível validar os dados gerados, e também identificar quando o programa não está realizando suas computações como deveria, facilitando para o desenvolvedor a busca por erros na aplicação.

O ambiente MATLAB disponibiliza funções matemáticas em sua biblioteca padrão que facilitam a codificação dos métodos de assimilação de dados, possibilitando assim um desenvolvimento mais ágil.

3.1.1. Funcionamento do programa

O algoritmo desenvolvido no ambiente MATLAB funciona da seguinte maneira:

- É definida a posição das observações;
- Gera-se o xt utilizando uma função conhecida;
- Gera-se o xb utilizando xt mais um valor de erro aleatório;
- Geram-se as observações com base em suas posições;
- No caso do BLUE gera-se as matrizes B, R e H;
- É realizada a assimilação de dados utilizando o método de Cressman ou o BLUE;
- Os resultados são mostrados na forma de um gráfico, para visualização e comparação dos resultados;

O código em MATLAB para o método de Cressman pode ser encontrado no APÊNDICE A, e o código para o BLUE no APÊNDICE B.

3.1.2. Resultados

A aplicação desenvolvida nos proporcionou a possibilidade de visualizar os resultados obtidos pelos métodos de assimilação de uma forma ágil, onde foi possível verificar o quão próximo eles estavam dos supostos valores reais desses dados.

Os dados usados para realizar os testes foram obtidos de uma função seno. Utilizou-se os valores disponibilizados pela função seno para representar xt . Para gerar valores que representassem xb , um pequeno erro foi adicionado aos valores gerados para xt . As observações foram geradas similarmente a geração do xt , mas o erro inserido foi menor, com o objetivo de que as observações encontrassem-se próximas da realidade, auxiliando assim no ajuste da previsão.

Para demonstrar os resultados obtidos no ambiente MATLAB, usou-se um raio de influência de 4 passos de tempo para Cressman. Utilizou-se observações a cada 1, 2, e 4 passos de tempo. Utilizou-se também uma geração aleatória de observações.

Com os testes realizados com a ferramenta utilizado um raio de influência para as observações a cada 4 passos de tempo, e com observações a cada espaço de tempo observou-se que o método de Cressman realizou uma assimilação muito próxima da realidade. Isso é uma característica desse método (Bouttier; Courtier, 1999), onde observações próximas do ponto de análise tem grande influência no resultado final. A Figura 3.1 demonstra uma visualização do resultado final por meio de um gráfico. Note-se que o x_t não faz parte da formulação de Cressman, mas está presente na visualização dos resultados, para efeito de comparação.

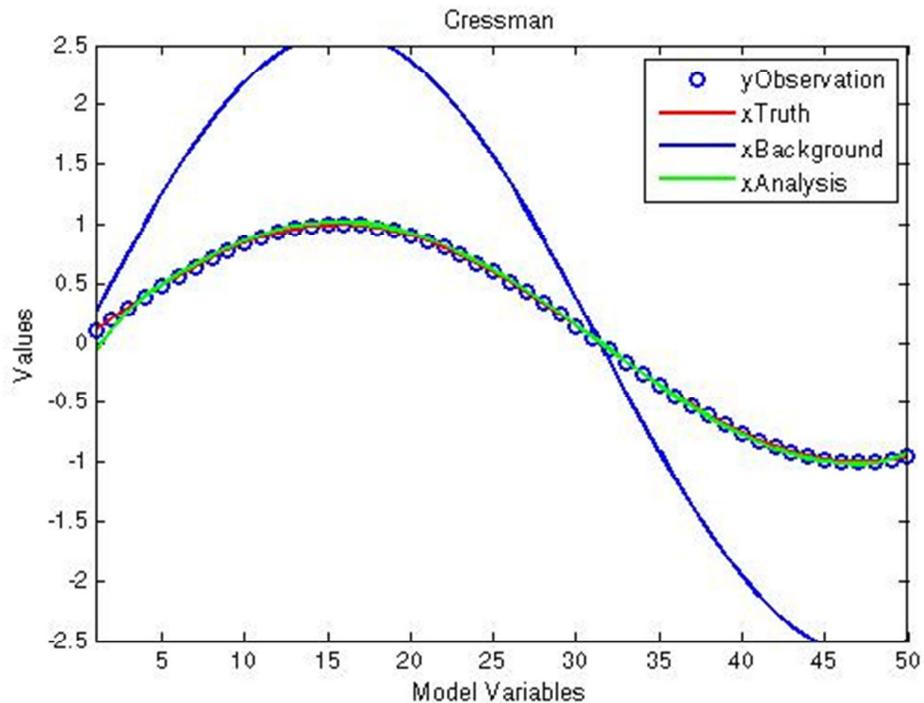


Figura 3.1 – Análise de Cressman com observações a cada espaço de tempo.

A Figura 3.2 representa os testes realizados utilizando um raio de influência a cada 4 passos de tempo, e com observações a cada 4 passos de tempo. A análise gerada ainda está muito próxima da realidade. Pois o método de Cressman ainda encontra observações em seu raio de busca para realizar a assimilação, que influenciam consideravelmente na geração dos dados pelo processo de assimilação.

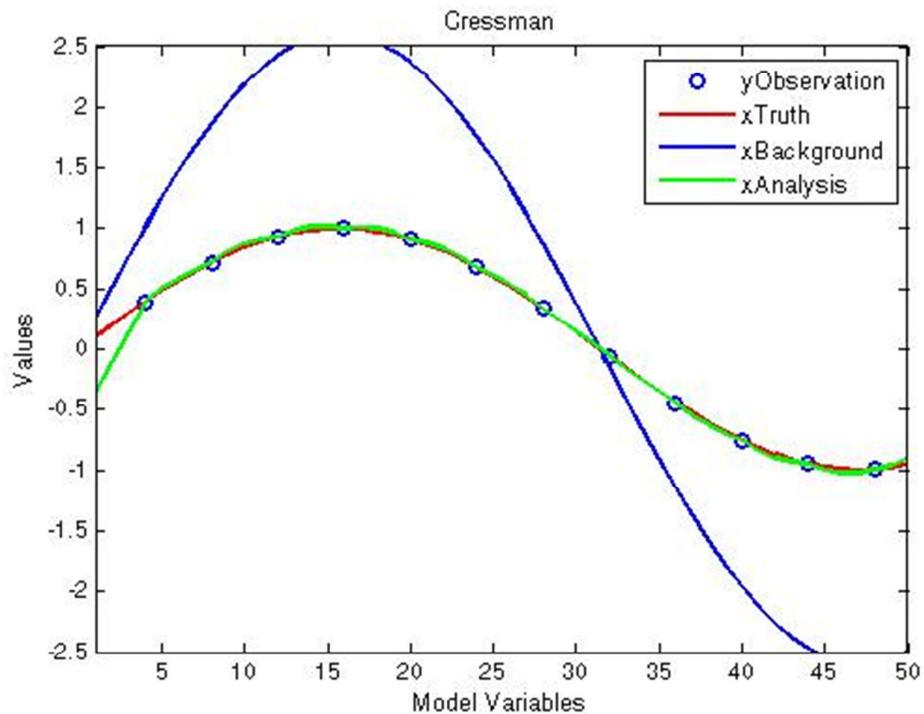


Figura 3.2 – Análise de Cressman com observações a cada 4 espaços de tempo.

A Figura 3.3 representa uma forma diferente para geração dos dados observacionais. Foi utilizada uma forma de geração aleatória de observações, com o objetivo de demonstrar que quando há falta de observações, os resultados do processo de assimilação divergem consideravelmente da realidade. Sendo assim influenciadas pela última análise realizada, o x_b .

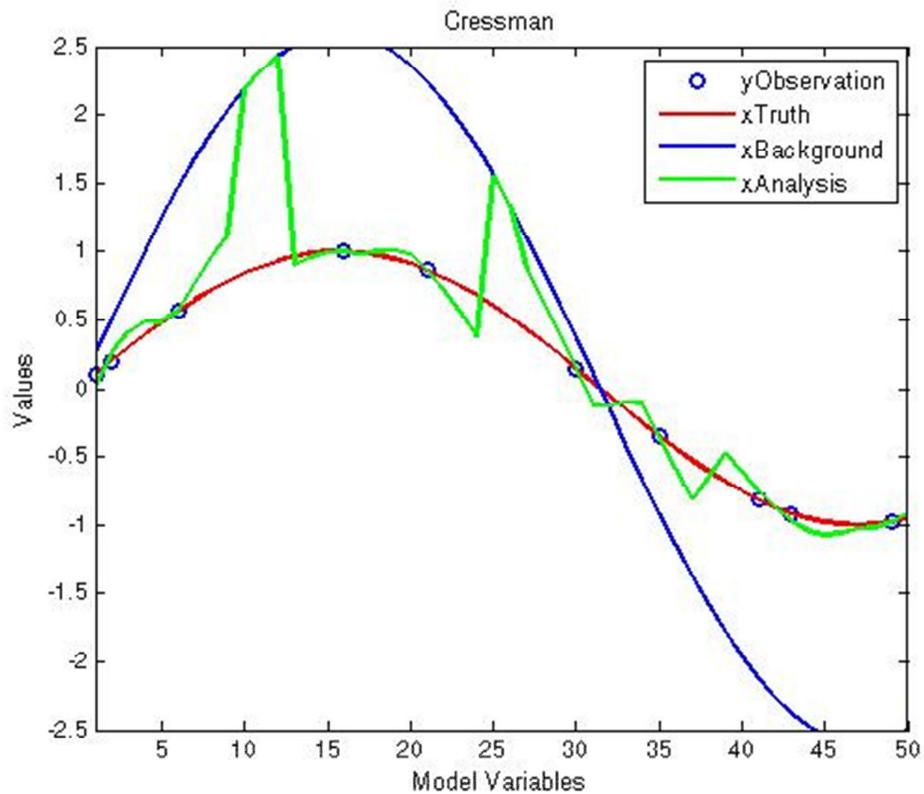


Figura 3.3 – Análise de Cressman com observações aleatoriamente espaçadas

A Figura 3.4 representa uma assimilação realizada utilizando o BLUE. Foram utilizadas observações a cada espaço de tempo. Pode se perceber que com um número elevado de observações a previsão é influenciada para a direção das observações.

Assim, se as observações disponíveis foram de boa qualidade e representem um número significativo à previsão tende a ser próxima da realidade.

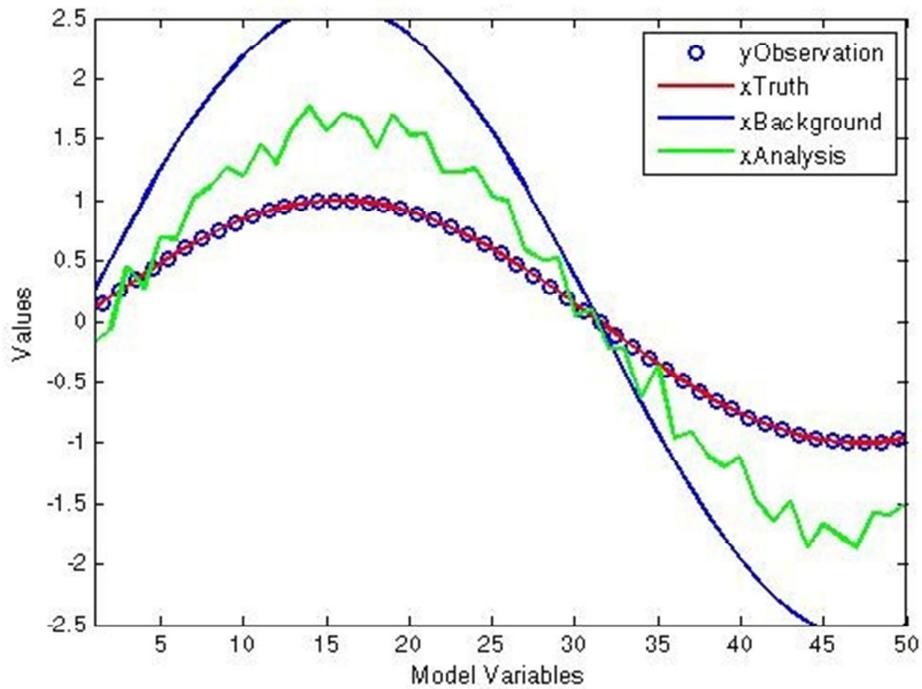


Figura 3.4 – BLUE com observações a cada espaço de tempo

A Figura 3.5 representa outra assimilação realizada utilizando o BLUE. Nessa visualização foram utilizadas observações a cada quatro espaços de tempo. Percebe-se que com um número menor de observações a previsão não é mais tão influenciada pelas observações, sendo um meio termo entre as observações e a previsão anterior, o x_b .

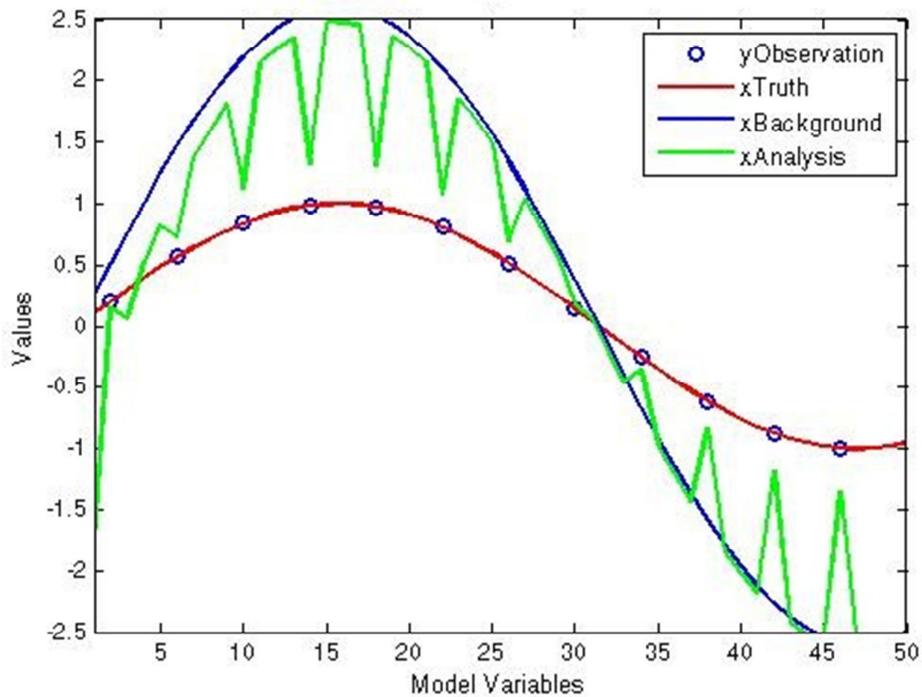


Figura 3.5 – BLUE com observações a cada 4 espaços de tempo

A Figura 3.6 representa uma assimilação utilizando o BLUE, onde foram utilizadas observações aleatoriamente espaçadas. Pode se notar que a assimilação é fortemente influenciada em direção das observações quando essas ocorrem em sua proximidade e para o x_b quando não há observações em uma região próxima.

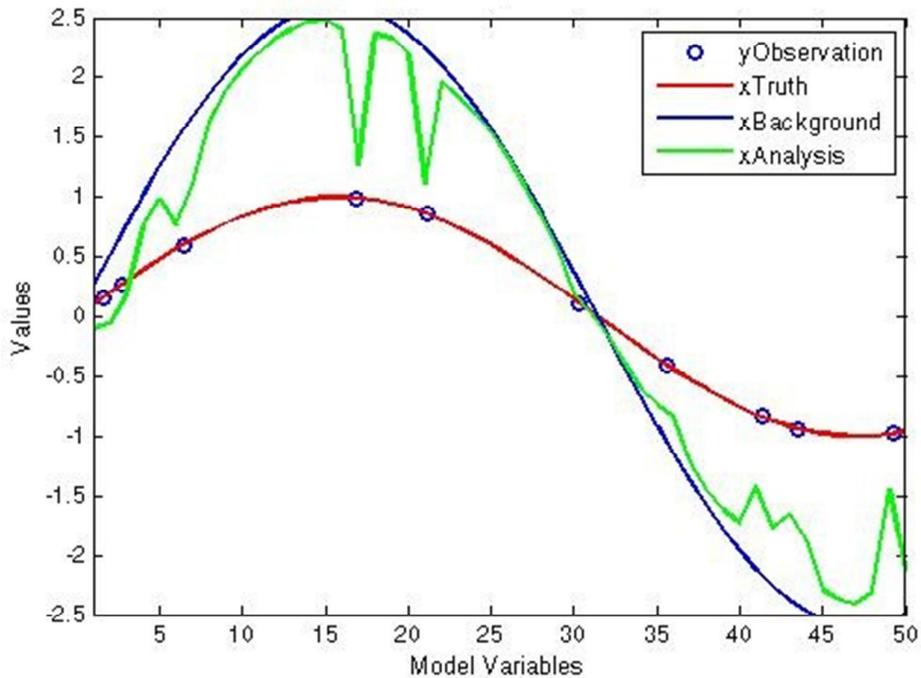


Figura 3.6 – BLUE com observações aleatoriamente espaçadas

3.2. Protótipo em C++

Na segunda etapa do desenvolvimento da ferramenta de assimilação de dados, as ideias anteriormente desenvolvidas em MATLAB foram utilizadas para codificação de uma versão da aplicação utilizando a linguagem de programação C++. A linguagem de programação orientada a objetos C++ é uma das mais populares linguagens de programação utilizadas. Ela dispõe de uma ampla variedade de bibliotecas com funções para os mais variados tipos de aplicações.

A abordagem utilizando a linguagem C++ também se deve ao fato da posterior integração da ferramenta de assimilação de dados com o sistema SUPIM-DAVS (Petry, Souza e Velho, 2011), que foi desenvolvido utilizando tal linguagem.

A popularidade da linguagem C++ faz com que empresas e grupos de pesquisa desenvolvam suas próprias bibliotecas para esta linguagem. Assim uma biblioteca chamada uBLAS (uBLAS, 2011) para álgebra linear, presente no conjunto de bibliotecas Boost (Boost, 2011) foi utilizado para facilitar os cálculos envolvendo matrizes, que são a base dos métodos de assimilação de dados.

3.2.1. Funcionamento do programa

O algoritmo do programa desenvolvido utilizando C++ funciona da seguinte maneira:

- Leitura do arquivo com os dados do Atrator de Lorenz para representar o xt ;
- Gera-se o xb acrescentando um pequeno erro nos valores de xt ;
- No caso do BLUE gera-se as matrizes B, R e H;
- É realizada a assimilação de dados utilizando o método de Cressman ou o BLUE;
- É calculado o somatório do erro;

3.2.2. Resultados preliminares

A ferramenta codificada em C++ ainda está em processo de desenvolvimento. Os métodos anteriormente codificados em MATLAB já foram codificados em C++. Espera-se que a versão estável da aplicação esteja pronta para ser utilizada nas atividades realizadas pelo LABCCE dentro de algumas semanas.

Utilizou-se uma abordagem diferente para geração da informação utilizada para testar a ferramenta. Os dados foram gerados pelo modelo caótico tridimensional conhecido como *Atrator de Lorenz*.

A partir de um conjunto de dados iniciais gerados pelo *Atrator de Lorenz*, foram acrescentados erros para obter o xb e as observações.

A Figura 3.7 apresenta o somatório do erro quadrático para assimilações realizadas com o método de Cressman, com o BLUE e a simples diferença de valores do xb em relação ao xt . A diferença de cada variável X, Y e Z em relação ao xt foi calculado, para que o erro quadrático fosse gerado e o somatório desses erros fosse feito. Para Cressman foram realizadas três assimilações, com raios de influencia de 3, 5 e 8 passos de tempo.

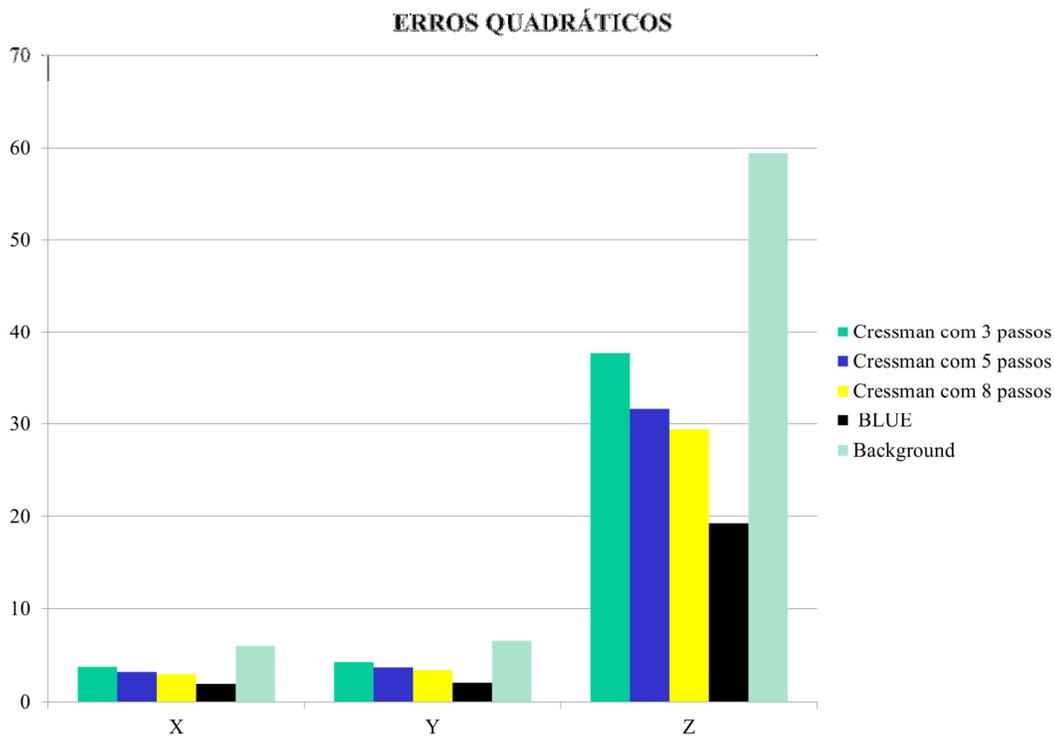


Figura 3.7 – Somatório do erro quadrático.

Os resultados preliminares demonstraram que o BLUE gerou dados que aproximam-se mais dos valores atribuídos como o estado verdadeiro. Notou-se que todas as assimilações realizadas, encontram-se mais próximas da realidade, do que o estado hipotético usado como xb .

3.3. Aquisição de dados observacionais

Uma das grandes dificuldades encontradas no processo de assimilação é a disponibilidade de dados para serem utilizados. Não se encontram muitas fontes de dados com acesso livre, que contenham dados reais medidos por equipamentos confiáveis.

Os testes realizados anteriormente foram todos baseados em algum tipo de função matemática, no caso a função trigonométrica seno e o *Atrator de Lorenz*. Com base nessa restrição buscou-se locais na internet que disponibilizam informações periodicamente, e de livre acesso. Essas informações foram encontradas no site da *National Oceanic and Atmospheric Administration* (NOAA), mais especificamente no projeto *Space Interactive Data Resource* (SPIDR), um projeto que permite acesso a uma base de dados histórica de dados físicos do espaço, implementada na linguagem de programação Java (Java, 1991) e disponibilizado pelo *World Wide Web*. Essa informação é disponibilizada em forma de arquivos como pode ser visualizado na Figura 3.5.

Esses arquivos disponibilizam a densidade eletrônica para determinadas altitudes, além de outras informações. No primeiro momento, apenas essa informação de densidade eletrônica é relevante para o projeto.

Com o objetivo de termos acesso aos dados disponibilizados pela NOAA, uma aplicação em C++ foi desenvolvida. A aplicação acessa os repositórios das ionossondas e faz o download da informação contida nela a partir de parâmetros passados para o programa.

hmf2	fof2	conf	nmf2	conf	lat	lon														
338.16	5.11	0.09	0.3237E+12	0.1133E+11	38.00	23.50	AQI=1	AT138_2011125000000	Problem	Flags1GH.	qualscan	V2009.10							
height	freq	f_conf	density	density_conf	ARTIST	FOLAN	ART	FOL	ART	FOL	ART	FOL	Slope	FOLAN	1/2					
264.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.08	F	5.1	5.1	348.4	338.2	38.7	36.6	140.0					
265.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.11	F												
266.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.14	F												
267.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.17	F												
268.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.20	F												
269.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.23	F												
270.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.27	F												
271.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.30	F												
272.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.34	F												
273.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.37	F												
274.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.41	F												
275.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.44	F												
276.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.47	F												
277.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.51	F												
278.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.54	F												
279.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.57	F												
280.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.61	F												
281.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.65	F												
282.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.68	F												
283.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.72	F												
284.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.76	F												
285.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.79	F												
286.00	-99.00	-99.00	-0.9900E+02	-0.9900E+02	0.00	c	3.83	F												
287.00	3.46	0.41	0.1506E+12	0.3521E+11	3.05	F2	3.87	F												
288.00	3.51	0.40	0.1546E+12	0.3497E+11	3.11	F2	3.91	F												
289.00	3.55	0.39	0.1586E+12	0.3473E+11	3.16	F2	3.95	F												
290.00	3.60	0.39	0.1627E+12	0.3446E+11	3.22	F2	3.99	F												
291.00	3.65	0.38	0.1668E+12	0.3418E+11	3.27	F2	4.03	F												
292.00	3.69	0.37	0.1710E+12	0.3387E+11	3.33	F2	4.06	F												
293.00	3.74	0.36	0.1752E+12	0.3354E+11	3.38	F2	4.10	F												
294.00	3.79	0.35	0.1794E+12	0.3318E+11	3.43	F2	4.14	F												
295.00	3.83	0.35	0.1836E+12	0.3280E+11	3.49	F2	4.18	F												

Figura 3.8 – Arquivo de dados observacionais.

Fonte: Adaptado de (SPIDR).

3.3.1. Funcionamento do programa

O algoritmo do programa funciona da seguinte maneira:

- É especificado o ano em que se deseja obter os arquivos;
- É especificado o intervalo de dias julianos em que se deseja obter os arquivos;
- Especificasse o horário inicial e final para obtenção dos arquivos;
- Especificasse o formato que deseja obter os arquivos;
- É estabelecida uma conexão via FTP com a base de dados;
- As bases de dados para cada ionosonda são validadas, com o objetivo de verificar se elas realmente contem informações;
- Os arquivos são construídos com base nas informações passadas para o programa;
- Arquivos são obtidos e armazenados em um diretório nomeado com o nome da localização da ionosonda;

3.3.2. Resultados Preliminares

Com as informações até o momento obtidas, do banco de dados das ionosondas, foi feita uma visualização da densidade eletrônica, representada pela Figura 3.9, para a região de Porto Rico (latitude 18.5, longitude 292.8). Uma previsão gerada pelo SUPIM para uma região próxima também pode ser visualizada na Figura 3.9.

O objetivo dessas visualizações é que seja possível validar as previsões feitas pelo SUPIM com base em medições reais feitas por equipamentos. Em um segundo momento, os dados de ionosondas serão utilizados para assimilação.

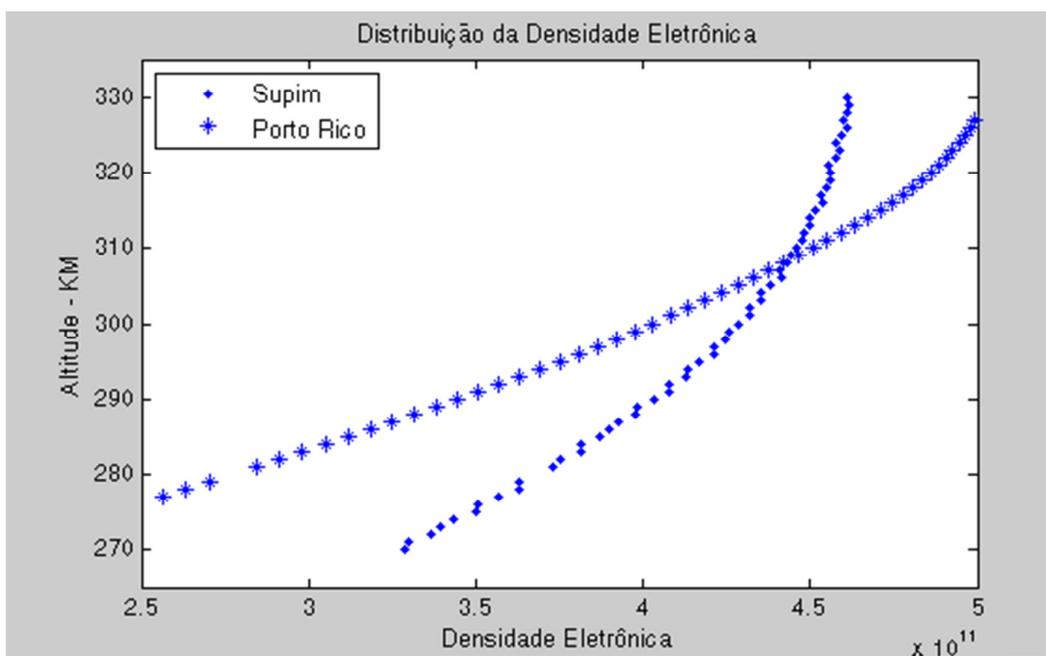


Figura 3.9 – Densidade eletrônica de uma ionosonda localizada em Porto Rico e a previsão do SUPIM para uma região próxima.

4 CONCLUSÃO

Durante o período de realização de trabalho, realizou-se um estudo sobre assimilação de dados e seus principais métodos. Muitas dificuldades práticas de implementação tiveram que ser superadas para construção de protótipos consistentes.

Considerou-se que as implementações tanto em MATLAB, quanto em C++ estão de acordo com os objetivos preliminares. As visualizações disponibilizadas pelos métodos codificados em MATLAB são úteis para realização de comparações.

A aquisição de informação da densidade eletrônica de fontes externas, que representam medições reais, será de grande utilidade para a continuação da pesquisa.

O bolsista teve a oportunidade de utilizar seus conhecimentos adquiridos durante seu curso de graduação, que muitas vezes ficam apenas na teoria da sala de aula, em uma área aplicada. O projeto de iniciação científica foi muito produtivo para a carreira acadêmica do orientando.

O acadêmico aprimorou seus conhecimentos em linguagens de programação utilizadas no projeto e conheceu novas linguagens nunca utilizadas, e que provavelmente não seriam exigidas no meio acadêmico.

Esse projeto de iniciação científica proposto pelo orientador Dr. Adriano Petry, proporcionou um ganho significativo aos conhecimentos do acadêmico. O incremento de conhecimento na formação acadêmica do orientando influenciará nas decisões tomadas futuramente.

4.1. Trabalhos futuros

O projeto ainda está em andamento. A ferramenta com os métodos de assimilação está em desenvolvimento. Nos últimos meses pesquisadores do INPE, que são especialistas na área de assimilação de dados tem contribuído com seus conhecimentos na pesquisa. As informações passadas por eles tem sanado muitas duvidas que estavam pendentes na pesquisa.

Os dados de medições reais obtidos através do site da NOAA devem ser usados para as próximas etapas do projeto.

Outros métodos de assimilação como o Filtro de Kalman (Evensen, 2006), devem ser estudados, prototipados e avaliados.

REFERÊNCIAS BIBLIOGRÁFICAS

- Basic Linear Álgebra, uBLAS, 2011. Disponível em <http://www.boost.org/doc/libs/1_40_0/libs/numeric/ublas/doc/index.htm> Acesso em janeiro de 2011.
- Boost C++ Libraries, 2011. Disponível em < <http://www.boost.org/>> Acesso em janeiro de 2011.
- European Centre for Medium-Range Weather Forecast, ECMWF. Bouttier, F; Courtier, P. Data assimilation concepts and methods. 1999. Disponível em: < http://www.ecmwf.int/newsevents/training/rcourse_notes/pdf_files/Assim_concepts.pdf >. Acesso em: junho 2011.
- Evensen, G. Data Assimilation: The Ensemble Kalman Filter. New York : Springer , 2006. 2New York : Springer , 2006. 77 p.
- Java, 2001. Disponível em: <<http://www.oracle.com/technetwork/java/>>. Acesso em abril de 2011.
- Lahoz, W; Khattatov, B; Ménard, R. Data Assimilation: Making Sense of Observations. New York: Springer, 2010. 718 p.
- MATLAB , 2009. Disponível em <<http://www.mathworks.com/products/matlab/>> Acesso em janeiro de 2011.
- Microsoft Visual Studio 2010, 2010. Disponível em <<http://www.microsoft.com/visualstudio/pt-br/>> Acesso em janeiro de 2011.
- NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION (NOAA). Disponível em < <http://www.noaa.gov/>>. Acesso em: abril de 2011.
- SPACE PHYSICS INTERACTIVE DATA RESOURCE (SPIDR). Disponível em <<http://spidr.ngdc.noaa.gov/spidr/home.do/>>. Acesso em: abril de 2011.
- SHEFFIELD UNIVERSITY PLASMASPHERE IONOSPHERE MODEL (SUPIM), 1970. Disponível em < <http://gbailey.staff.shef.ac.uk/supim.html>>. Acesso em junho de 2011.
- PETRY, A.; SOUZA, J. R.; VELHO, H. F. C. Sistema para previsão operacional da dinâmica da ionosfera baseado no modelo SUPIM. São José dos Campos: INPE, versão: 2011-06-03. Disponível em: <<http://urlib.net/8JMKD3MGP7W/39QK26P>>. Acesso em: 14 jul. 2011.

APÊNDICE A – PROTÓTIPO EM MATLAB: ANÁLISE DE CRESSMANN

Uma das versões desenvolvidas no ambiente MATLAB para o método de Cressman.

```
theNumModelVariables=50;

%Define as posições e o numero de observações
%De forma homogenea
thePositionObservations= zeros(theNumModelVariables/4,1,'double');
for i =1:theNumModelVariables/2
thePositionObservations(i)= i*2;
end

%De forma arbitraria
%thePositionObservations = [1, 2, 6, 16, 21, 30, 35, 41, 43, 49];

%Numero de Observações
theNumObservations= length(thePositionObservations);
%Xt
xTrue= zeros(theNumModelVariables,1,'double');
%Xb
xBackground= zeros(theNumModelVariables,1,'double');
%Y
yObservations= zeros(theNumObservations,1,'double');

%Faz uma nova simulação
for i =1:theNumModelVariables
xTrue(i)= sin(i/10);
xBackground(i)= sin(i/10)*2.6;
end
for i =1:theNumObservations
yObservations(i)= sin(thePositionObservations(i)/10);
end

xAnalysis= zeros(theNumModelVariables,1);
theRadius=2;
theRadius2 =theRadius*theRadius;

for j =1:theNumModelVariables
theSumWeight=0;
theSum=0;
for i =1:theNumObservations
theDist= abs(thePositionObservations(i)- j);
theDist2 =theDist*theDist;
theWeight=(theRadius2 - theDist2)/(theRadius2 + theDist2);
if theWeight<0
theWeight=0;
```

```

        end
theSum=theSum+theWeight;
theSumWeight=theSumWeight+theWeight*(yObservations(i)-
xBackground(thePositionObservations(i)));
end
if(theSum~=0)
xAnalysis(j)=xBackground(j)+theSumWeight/theSum;
end
if(theSum==0)
xAnalysis(j)=xBackground(j);
end

end

h =plot(thePositionObservations,yObservations,'o','LineWidth',2);
set(gca,'YLim',[-2.52.5])
set(gca,'XLim',[1theNumModelVariables])
hold on
plot(xTrue,'r','LineWidth',2);
hold on
plot(xBackground,'b','LineWidth',2);
hold on
xlabel('Model Variables')
ylabel('Values')
title('Cressman - Raio 4')
legend('yObservation','xTruth','xBackground')
plot(xAnalysis,'g','LineWidth',2);
hold off
legend('yObservation','xBackground','xAnalysis')

```

APÊNDICE B - PROTÓTIPO EM MATLAB: BLUE

Uma das versões desenvolvidas no ambiente MATLAB para o BLUE.

```
%OI
theNumModelVariables=50;
theNumIterations=100000;
%Posição das Observações
thePositionObservations= zeros(theNumModelVariables,1,'double');
for i =1:theNumModelVariables/4
thePositionObservations(i)=(i -0.5)*4;
end

%thePositionObservations = [1.6, 2.76, 6.45, 16.89, 21.1, 30.34,
35.64, 41.34, 43.46, 49.23];

theNumObservations= length(thePositionObservations);
disp(theNumObservations)
%Xt
xTrue= zeros(theNumModelVariables,1,'double');
%Xb
xBackground= zeros(theNumModelVariables,1,'double');
%Y
yObservations= zeros(theNumObservations,1,'double');

%Eb - NumIterations
Eb=zeros(theNumModelVariables,theNumIterations,'double');
%Eo - NumIterations
Eo=zeros(theNumObservations,theNumIterations,'double');

%Calculando H com IDW
H =zeros(theNumObservations,theNumModelVariables,'double');
for i =1:theNumObservations
theSum=0;
theExact=0;
for j =1:theNumModelVariables
dist= abs(thePositionObservations(i)- j);
%Se em uma observação com distancia 0, coloca 1 como o peso e zera o
%restante da linha
if(dist==0)
for k =1:theNumModelVariables
H(i,k)=0;
end
H(i,j)=1;
theExact=1;
break;
end
H(i,j)=1/dist;
theSum=theSum+ H(i,j);
end
if(theExact==0)
for j =1:theNumModelVariables
H(i,j)= H(i,j)/theSum;
end
end
```

```

end
end
disp(size(H))
disp(H)
%Faz varias iterações para ter vários Eb e Eo
for k =1:theNumIterations
for i =1:theNumModelVariables
xTrue(i)= sin(i/10)+(0.1-0.0).*rand()-(0.1-0.0).*rand();
xBackground(i)= sin(i/10)*2.6+(0.25-0.0).*rand()-(0.25-0.0).*rand();
end

for i =1:theNumObservations
yObservations(i)= sin(thePositionObservations(i)/10)+(0.1-
0.0).*rand()-(0.1-0.0).*rand();
end

for j =1:theNumModelVariables
Eb(j,k)=xBackground(j)-xTrue(j);
end

vecTemp= H *xTrue;
for j =1:theNumObservations
Eo(j,k)=yObservations(j)-vecTemp(j);
end
end

%CalculaEbMedio e EoMedio
EbMedio=zeros(theNumModelVariables,1,'double');
EoMedio=zeros(theNumObservations,1,'double');
for k =1:theNumIterations
for j =1:theNumModelVariables
EbMedio(j)=EbMedio(j)+Eb(j,k);
end
vecTemp= H *xTrue;
for j =1:theNumObservations
EoMedio(j)=EoMedio(j)+Eo(j,k);
end
end
for j =1:theNumModelVariables
EbMedio(j)=EbMedio(j)/theNumIterations;
end
for j =1:theNumObservations
EoMedio(j)=EoMedio(j)/theNumIterations;
end

%Calculando B
B = zeros(theNumModelVariables,theNumModelVariables,'double');
for k =1:theNumIterations
EbCurrent=zeros(theNumModelVariables,1,'double');
for j =1:theNumModelVariables
EbCurrent(j)=Eb(j,k);
end
tempB=(EbCurrent-EbMedio)*((EbCurrent-EbMedio)');
B = B +tempB;
end
B = B /theNumIterations;

```

```

%Calculando R para a nova simulação
R = zeros(theNumObservations,theNumObservations,'double');
for k =1:theNumIterations
EoCurrent=zeros(theNumObservations,1,'double');
for j =1:theNumObservations
EoCurrent(j)=Eo(j,k);
end
tempR=(EoCurrent-EoMedio)*((EoCurrent-EoMedio)');
R = R +tempR;
end
R = R /theNumIterations;

%Faz uma nova simulação
for i =1:theNumModelVariables
xTrue(i)= sin(i/10);
xBackground(i)= sin(i/10)*2.6;
end
for i =1:theNumObservations
yObservations(i)= sin(thePositionObservations(i)/10);
end

K = B * H'/(H * B * H' + R);

Delta = K *(yObservations- H *xBackground);

xAnalysis=xBackground+ Delta;

h =plot(thePositionObservations,yObservations,'o','LineWidth',2);
set(gca,'YLim',[-2.52.5])
set(gca,'XLim',[1theNumModelVariables])
hold on
plot(xTrue,'r','LineWidth',2);
hold on
plot(xBackground,'b','LineWidth',2);
hold on
xlabel('Model Variables')
ylabel('Values')
title('OI')
legend('yObservation','xTruth','xBackground')
plot(xAnalysis,'g','LineWidth',2);
hold off
legend('yObservation','xTruth','xBackground','xAnalysis')

```