



Ministério da
Ciência e Tecnologia



sid.inpe.br/mtc-m19/2011/06.03.18.08-MAN

**SISTEMA PARA PREVISÃO OPERACIONAL DA
DINÂMICA DA IONOSFERA BASEADO NO MODELO
SUPIM**

Adriano Petry
Jonas Rodrigues de Souza
Haroldo Fraga de Campos Velho

URL do documento original:
<<http://urlib.net/8JMKD3MGP7W/39QK26P>>

INPE
São José dos Campos
2011

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):**Presidente:**

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Membros:

Dr^a Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr^a Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr^a Regina Célia dos Santos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Dr. Ralf Gielow - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr. Wilson Yamaguti - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr. Horácio Hideki Yanasse - Centro de Tecnologias Especiais (CTE)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Deicy Farabello - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Vivéca Sant´Ana Lemos - Serviço de Informação e Documentação (SID)



Ministério da
Ciência e Tecnologia



sid.inpe.br/mtc-m19/2011/06.03.18.08-MAN

**SISTEMA PARA PREVISÃO OPERACIONAL DA
DINÂMICA DA IONOSFERA BASEADO NO MODELO
SUPIM**

Adriano Petry
Jonas Rodrigues de Souza
Haroldo Fraga de Campos Velho

URL do documento original:
<<http://urlib.net/8JMKD3MGP7W/39QK26P>>

INPE
São José dos Campos
2011

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 1 – Estrutura de diretórios utilizada para organização do sistema	2
Figura 2 – Estrutura de diretórios utilizada para organizar as simulações realizadas	4
Figura 3 – Tabela de agendamento de execuções para o sistema SUPIM-DAVS.....	6
Figura 4 – Parâmetros configuráveis da simulação.....	8
Figura 5 – Técnica “explosão de perfil” de primeira (a), segunda (b) e terceira ordem (c) aplicada para geração de mapas de conteúdo eletrônico total	12
Figura 6 – Atrasos temporais dos pontos simulados antes do ajuste de tempo.....	13
Figura 7 – Atrasos temporais dos pontos simulados após ajuste de tempo.....	13
Figura 8 – Exemplo de imagem para simulação do conteúdo eletrônico total para América do Sul	14
Figura 9 – Exemplo de imagem para simulação do conteúdo eletrônico total para Globo.....	15
Figura 10 – Geração de imagens para América do Sul: (a) com todas 28 rodadas do SUPIM presentes, (b) com 1 rodada faltante, (c) com 2 rodadas adjacentes faltantes, (d) com 3 rodadas adjacentes faltantes, (e) com 4 rodadas adjacentes faltantes, (f) com 5 rodadas adjacentes faltantes, (g) com 6 rodadas adjacentes faltantes, (h) com 7 rodadas adjacentes faltantes	20
Figura 11 – Geração de imagens para Globo: (a) com todas 61 rodadas do SUPIM presentes, (b) com 1 rodada faltante.....	20
Figura 12 – Parâmetros de configuração do DAVS.....	22
Figura 13 – Geração de mapas de TEC para Globo sem dados de simulação do dia anterior.....	23
Figura 14 – Geração de mapas de TEC para América do Sul sem dados de simulação do dia anterior.....	23

LISTA DE TABELAS

	<u>Pág.</u>
Tabela 1: Funcionalidade dos diretórios do sistema.....	3
Tabela 2: Conteúdo dos diretórios de simulação do sistema.....	5

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
2 ESTRUTURA E ORGANIZAÇÃO DO SISTEMA.....	2
3 EXECUÇÃO E OPERAÇÃO	6
3.1 Configuração dos parâmetros da simulação.....	7
3.2 Inicialização da simulação.....	7
3.3 Obtenção dos dados de fluxo solar.....	9
3.4 Execução paralela do código SUPIM.....	9
3.5 Execução paralela do DAVS.....	11
3.6 Geração de imagens	14
3.7 Finalização da simulação	15
4 POSSÍVEIS PROBLEMAS E SOLUÇÕES	16
4.1 Reinicialização do cluster.....	16
4.2 Nós do cluster com problema	17
4.3 Falta de espaço em disco.....	17
4.4 Ajuste da hora do sistema.....	18
4.5 Simulação iniciada com cluster ocupado	18
4.6 Arquivo com dados de fluxo solar ausentes	19
4.7 Falta de convergência do SUPIM	19
4.8 Falta de memória durante processamento do DAVS	21
4.9 Falta dos dados da simulação do dia anterior.....	22
4.10 Demora excessiva durante processamento.....	24
4.11 Escala de cores incompatível	24
4.12 Falha ao enviar imagens ao destino final.....	25
5 CONCLUSÃO.....	26
REFERÊNCIAS BIBLIOGRÁFICAS.....	27
APÊNDICE A – INSTALAÇÃO DE FERRAMENTAS DE SOFTWARE	29

1 INTRODUÇÃO

Este manual foi desenvolvido com o propósito de documentar todo o processo de instalação e operação de um sistema desenvolvido para previsão da dinâmica ionosférica, baseado no modelo SUPIM [1-2]. Esse sistema, batizado de SUPIM-DAVS, foi exaustivamente testado e está em operação desde março de 2011 rodando em um cluster de processadores, disponível junto ao Centro Regional Sul de Pesquisas Espaciais (CRS/INPE). As imagens de conteúdo eletrônico total (TEC) são disponibilizadas diariamente no sítio do Programa de Clima Espacial do INPE - <http://www.inpe.br/climaespacial/index.php>.

Inicialmente, o capítulo 2 mostra como o sistema foi estruturado, identificando a geração de diretórios e localização dos vários arquivos de configuração e de resultados dos subsistemas constituintes. No capítulo 3 é detalhado o funcionamento completo do sistema SUPIM-DAVS, desde sua configuração, geração e obtenção de arquivos, processamentos paralelos e seriais executados, geração de imagens e finalização. A seguir, o capítulo 4 identifica diversos problemas e dificuldades enfrentados durante o período de testes do sistema, apontando possíveis soluções. Esses problemas devem ser cuidadosamente analisados pela equipe de operação, já que muitos deles certamente irão ocorrer novamente ao longo do tempo. Ao final, conclusões e trabalhos futuros são apontados. O apêndice deste manual também inclui os procedimentos adotados para correta instalação e configuração das principais ferramentas de software necessárias à operação do sistema.

2 ESTRUTURA E ORGANIZAÇÃO DO SISTEMA

O sistema SUPIM-DAVS descrito neste documento foi construído separando em diretórios distintos os arquivos desenvolvidos e utilizados para instalação e “setup” do sistema (diretório “install”), dos arquivos que são utilizados e gerados na execução das simulações (diretório “runs”), como ilustra a figura 1.

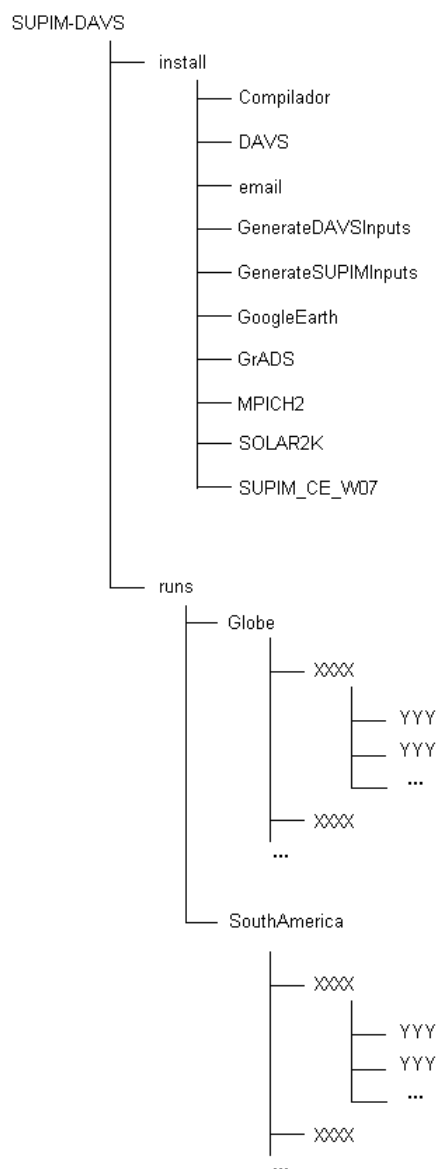


Figura 1 – Estrutura de diretórios utilizada para organização do sistema

A funcionalidade de cada diretório mostrado na figura 1 varia, e é detalhada na tabela 1.

Tabela 1: Funcionalidade dos diretórios do sistema

Diretório	Descrição
Compilador	Armazenamento e instalação dos compiladores Fortran e C da Intel
DAVS	Arquivos-fonte em linguagem C++ do Data Assimilation and Visualization System (DAVS), para interpolação, assimilação e visualização de dados
email	Programa em JAVA e bibliotecas utilizados para notificação via email de eventos durante operação
GenerateDAVSInputs	Programa em JAVA utilizado para gerar arquivos de configuração do DAVS, além de também gerar um script utilizado para geração das imagens pelo GrADS
GenerateSUPIMInputs	Programa em JAVA capaz de gerar arquivos de entrada necessários às rodadas do SUPIM, além de também gerar um script usado para rodar o SUPIM usando MPI com o gerenciador de recursos OAR
GoogleEarth	Programa em JAVA para gerar arquivo KMZ para visualização das imagens através do Google Earth
GrADS	Contém arquivos para instalação do software GrADS e os scripts “scriptPrintImage.gs” e “cbar.gs”, usados para geração das imagens e visualização da escala de cores
MPICH2	Armazenamento e instalação da implementação do MPI versão 2
SOLAR2K	Armazenamento e instalação do sistema Solar 2000, e programa escrito em linguagem C para automatização do processo de consulta aos dados solares
SUPIM_CE_W07	Programa SUPIM escrito em linguagem Fortran, usado para simular a dinâmica ionosférica
runs e subdiretórios	Cada rodada completa do sistema gera arquivos que são guardados em subpastas a partir de “runs>Globe” ou “runs>SouthAmerica”, onde XXXX equivale ao ano e YYY ao dia juliano simulado

A cada simulação do sistema e na medida em que vão sendo usados, os programas são compilados e apenas os arquivos executáveis ou resultados de execução são copiados para os diretórios “YYY” das simulações. A única exceção refere-se ao programa escrito em linguagem C utilizado para automatizar o processo de consulta aos dados solares do Solar 2000 [7], que não é compilado no momento da simulação. A compilação não é feita porque a versão utilizada do Solar 2000 é a “SYSTEM GRADE V2.36”, chamada a partir do programa em C, e que requer que o Interactive Data Language (IDL) [8] esteja rodando. No entanto, apenas a versão 32 bits funciona corretamente no cluster do CRS (que dispõe de processadores 64 bits). Assim, o programa em C é compilado em processador 32 bits e o arquivo executável gerado é transferido ao cluster para execução.

A estrutura de subdiretórios que é criada a partir de cada pasta YYY é mostrada na figura 2. Essa estrutura foi concebida para armazenar de forma organizada os diversos arquivos que são gerados a cada rodada.

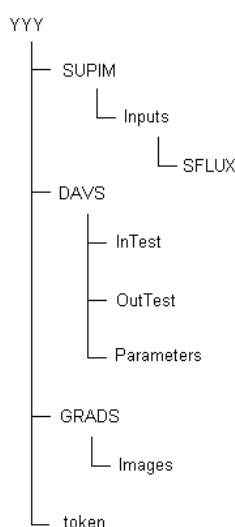


Figura 2 – Estrutura de diretórios utilizada para organizar as simulações realizadas

A finalidade e o conteúdo de cada subdiretório da estrutura mostrada na figura 2 é detalhado na tabela 2.

Tabela 2: Conteúdo dos diretórios de simulação do sistema

Diretório	Descrição
SUPIM	Armazenamento dos arquivos gerados durante a execução do programa SUPIM. Alguns desses arquivos são transferidos para “DAVS>InTest” para tratamento pelo DAVS
Inputs	Armazenamento dos arquivos de configuração utilizados na execução do programa SUPIM
SFLUX	Contém informações de fluxo solar utilizadas na execução do programa SUPIM
DAVS	Contém arquivos de configuração e gerados durante a execução do DAVS
InTest	Armazena arquivos com informações dos pontos de simulação, que são gerados pela execução do programa SUPIM e transferidos para este diretório para serem utilizados pelo DAVS
OutTest	A execução do DAVS gera arquivos de saída neste diretório, que são posteriormente transferidos para o diretório GRADS
Parameters	Contém arquivos de configuração para a execução do DAVS
GRADS	Os arquivos de saída do DAVS, gerados em “DAVS>OutTest”, são transferidos para este diretório para geração dos mapas
Images	Os mapas gerados pela execução do GrADS são armazenados neste diretório, e posteriormente transferidos para uma localidade na internet
token	Alguns arquivos existentes em “DAVS>InTest” são transferidos para este diretório para uso na simulação do dia seguinte

3 EXECUÇÃO E OPERAÇÃO

O sistema SUPIM-DAVS executa diariamente, gerando mapas com a previsão da dinâmica da ionosfera para um período futuro. A sua execução é agendada utilizando o serviço CRON do Linux. O arquivo mostrado na figura 3, obtido através do comando “crontab –e”, ilustra a tabela de execução de simulações diárias para a América do Sul e para o Globo, que são disparadas diariamente às 21:50h e 22:00h, respectivamente.

```
50 21 * * * /home/supim-davs/SUPIM-DAVS/runs/SouthAmerica/run_SUPIM-
DAVS_v2.4.sh > /home/supim-davs/SUPIM-DAVS/runs/SouthAmerica/log
00 22 * * * /home/supim-davs/SUPIM-DAVS/runs/Globe/run_SUPIM-DAVS_v2.4.sh
> /home/supim-davs/SUPIM-DAVS/runs/Globe/log
```

Figura 3 – Tabela de agendamento de execuções para o sistema SUPIM-DAVS

A figura 3 indica também que as saídas textuais para a execução das simulações são direcionadas ao diretório “runs/SouthAmerica” para a simulação para a América do Sul, e “runs/Globe”, para o Globo. Nos mesmos diretórios também estão presentes o script de execução que é invocado, chamado “run_SUPIM-DAVS_v2.4.sh”. Nele, estão presentes os parâmetros e instruções utilizados durante uma execução do sistema, que pode ser decomposta em sete etapas distintas. A primeira delas refere-se à “configuração dos parâmetros da simulação”, que define a forma, técnicas e algoritmos utilizados. A seguir, a etapa de “inicialização”, que abrange a verificação de simulações em curso, avaliação de recursos disponíveis e execução de procedimentos básicos. Antes do código SUPIM [1-2] iniciar seu funcionamento paralelo, a etapa de “obtenção dos dados de fluxo solar” garante que as informações de fluxo e atividade solar para o dia simulado estejam disponíveis e válidas. Em seguida, a “execução paralela do código SUPIM” irá executar de forma paralelizada várias rodadas do código, em longitudes distintas, de forma a fornecer a quantidade suficiente de pontos de simulação dentro de uma área de cobertura pré-definida. Já a “execução paralela do DAVS” faz uso das saídas geradas pela etapa anterior de forma a

obter valores de concentração eletrônica e/ou iônica dentro de uma grade homogeneamente espaçada, que varia em latitude, longitude, altitude e tempo. Os resultados obtidos são gravados em arquivos no formato apropriado para que a etapa de “geração de imagens” possa gerar mapas bi-dimensionais que contenham a informação desejada a cada espaço de tempo. Por fim, a etapa de “finalização da simulação” encarrega-se de enviar os mapas gerados ao destino apropriado, compactar os arquivos gerados e notificar a conclusão do processo. Cada uma das etapas mencionadas é detalhada a seguir.

3.1 Configuração dos parâmetros da simulação

Cada simulação pode ser executada de formas variadas utilizando recursos e técnicas diferenciadas. As informações constantes no início do script de execução definem esses parâmetros. A figura 4 mostra os parâmetros que podem ser configurados para esta versão do sistema. Cada parâmetro apresenta um comentário explicativo da sua funcionalidade e possíveis valores que podem assumir.

3.2 Inicialização da simulação

A simulação inicia identificando a data (dia, mês, ano e dia juliano) que deve rodar. A seguir, verifica se o sistema já está rodando outra simulação do mesmo usuário (supim-davs). Se estiver, fica aguardando por até 8 horas a conclusão dessa simulação. Simulações concorrentes poderão interferir umas nas outras, ocasionando falha. Assim, apenas uma simulação irá rodar por vez. Quando nenhuma outra simulação estiver rodando, é verificado se existem processos ativos do usuário supim-davs no cluster. Se existirem, eles serão eliminados, já que nenhuma simulação está aguardando sua conclusão. A seguir, a estrutura de diretórios é criada, o código SUPIM é compilado e são gerados os arquivos para que ele rode da forma estabelecida pelos parâmetros de configuração.

```

##### Configuration data #####
#
# Number of SUPIM simulations to be used (12 - initial, 19 - median
# resolution, 28 - better resolution)
numberSimulations=28
#
# Starting longitude (-85 - default)
startingLongitude=-85
#
# Longitude step (5 - initial, 3 - median resolution, 2 - better resolution)
longitudeStep=2
#
# Date generation: use current day (0), next day (1 - default) or pre-defined date (-1)
dateGeneration=1
#
# Pre-defined date values (ignored if dateGeneration equals to 1 or 0)
PreDefinedJulianDay=74
PreDefinedDAY=15
PreDefinedMONTH=3
PreDefinedYEAR=2011
#
# Define if Solar 2000 file will be downloaded and checked (1 - default)
# or if local file in install directory will be used without check (0)
SolarFile=1
#
# Define if DEBUG mode is off (0 - default) or on (1). If DEBUG mode
# is on, no cleaning is performed on files.
DEBUGmode=0
#
# Define if SUPIM outputs will (1 - default) or not (0) be post
# processes to adjust timing in simulation points
timeAdjust=1
#
# Destination server for images
destinationServer="200.132.24.196:/var/lib/tomcat6/webapps/ROOT/site/supim"
#
# Define if email will be sent (1 - default) or not (0)
notifyStatus=1
#
# Define if files generated by SUPIM and used by DAVS will be (1 - default)
# or will not be (0) compressed or (-1) will be deleted after simulation concludes
compressFiles=1
#
# System complete path (default /home/supim-davs/DAVS_operacional/ )
DAVS_path="/home/supim-davs/SUPIM-DAVS"
#
# Output directory
outputDirectory="/home/supim-davs/SUPIM-DAVS/runs/SouthAmerica"
#
# Use OAR with all simulations together (0),
# or MPICH2 with all simulations together (1),
# or OAR with independent simulations (2 - default)
clusterUse=2
#
# Define number of processes per node to be used: (1 - default), 2 or 4 if
# clusterUse is 2. Otherwise it is ignored.
processesPerNode=4
#
# Define APIndex used on SUPIM input files (4 - default or 10)
# for calm days and geomagnetic storms, respectively
APIndex=4
#
# Ground Station latitude (-23 - default for South America)
GSLAT=-23
#
# DAVS configuration - default values for south america
# are 24, 0, -90, -25, 2, -65, 20, 2, 90000, 1000000, 10000, 2
numberTimeSteps=24
initialTime=0
lonBegin=-90
lonEnd=-25
lonStep=2
latBegin=-65
latEnd=20
latStep=2
elevBegin=90000
elevEnd=1000000
elevStep=10000
explosion=2
#
# SSH key directory
sshDir="/home/supim-davs/.ssh"
#
# Set up environment variables
# MPICH2 bin path
export PATH=/home/supim-davs/mpich2/bin:$PATH
source /etc/profile
#
scriptGrADS="scriptPrintImageSouthAmerica.gs"
#
#####

```

Figura 4 – Parâmetros configuráveis da simulação

3.3 Obtenção dos dados de fluxo solar

As informações de fluxo solar são obtidas a partir da “Solar Irradiance Platform” versão “System Grade” (SIP_System_Grade) da empresa “Space Environment Technologies”. Detalhes dessa plataforma podem ser obtidos em [7]. Essa versão da plataforma permite realizar download via internet dos dados desejados, através de uma função disponibilizada em Interactive Data Language (IDL), da empresa “ITT Visual Information Solutions”. Detalhes dessa linguagem podem ser obtidos em [8]. Essa função pode ser invocada a partir de programas escritos em outras linguagens de programação. Havendo uma licença IDL válida, os dados de fluxo solar podem ser obtidos via internet e utilizados através do arquivo textual gerado. Para o sistema descrito neste documento, foi escrito um programa em linguagem C que invoca a função em IDL que obtém via internet o arquivo com dados de fluxo solar para uma data pré-determinada em outro arquivo texto (gerado automaticamente antes do *download*). Para datas passadas, o arquivo de fluxo solar recebido será completo. Para datas futuras, o arquivo vem com falta de algumas informações essenciais. Assim, foi desenvolvido outro programa em linguagem C que avalia se os dados obtidos estão ou não completos. A simulação somente dará continuidade ao processamento após receber todas as informações de fluxo solar necessárias. Verificou-se que em torno de 01:00h UT de cada dia, os dados de fluxo solar daquele dia já estão disponíveis. Depois, em torno de 23:00h UT esses dados são revistos e atualizados. Assim, para realizar a previsão futura da dinâmica ionosférica para um determinado dia, somente após a 01:00h UT todo o processamento poderá ser iniciado.

3.4 Execução paralela do código SUPIM

Na versão atualmente disponível do sistema SUPIM-DAVS, o código SUPIM [1-3] pode ser executado paralelamente de três formas distintas: através da técnica Message Passing Interface (MPI) versão 2 [5] aplicada diretamente aos nós do cluster, através da técnica MPI submetida ao gerenciador de recursos

OAR, ou apenas disparando os processamentos paralelos diretamente através do gerenciador de recursos OAR [9] (sem uso de MPI). Diferentemente das duas primeiras alternativas de execução paralela, a última alternativa vem se mostrando mais adequada para melhorar a estabilidade das simulações, já que a falta de convergência em alguma rodada do código SUPIM não afeta a execução das demais rodadas. Como a simulação da dinâmica ionosférica para a América do Sul é capaz de gerar os mapas completos de conteúdo eletrônico mesmo havendo algumas falhas em rodadas do código SUPIM, a execução concorrente e independente das rodadas apresenta, portanto, um caráter de tolerância a falhas. Além disso, existem alguns parâmetros do código SUPIM que podem ser alterados, caso haja problemas de convergência. Essa alteração é independente para cada rodada, no caso de utilização do gerenciador OAR diretamente, sem o uso do MPI. Nesta abordagem, os processos são submetidos um a um, sendo que um novo processo somente será submetido se o anterior já estiver em execução. Após a submissão de um processo, o SUPIM-DAVS aguarda por um período de tempo pré-determinado (atualmente 60 segundos) o processo iniciar sua execução. Caso isso não aconteça, por exemplo por falta de nós disponíveis, o mesmo processo é submetido novamente. Caso contrário o próximo processo é submetido, até a finalização de todas as submissões. A seguir, o sistema permanece verificando a cada minuto se todos os processamentos foram concluídos, sejam eles bem sucedidos ou não. Tanto durante a submissão de processos, quanto durante a espera pela conclusão dos mesmos, foi implementado um mecanismo de proteção relacionado ao tempo já decorrido. Esse mecanismo finaliza a execução da simulação caso se tenha aguardado, sem sucesso, por mais de 8 horas pelos resultados.

3.5 Execução paralela do DAVS

O DAVS objetiva, basicamente, utilizar os resultados gerados pelas várias rodadas do código SUPIM em uma simulação para construir uma grade homogeneamente espaçada em latitude, longitude e altitude geográficas com estimação de valores de concentração iônica/eletrônica em cada ponto. Além disso, o DAVS pretende também realizar a assimilação de dados de observações para a grade gerada, fazendo com que ela represente mais fielmente as informações medidas. No estágio atual do desenvolvimento do DAVS, a assimilação de dados ainda não está concluída.

O processo de interpolação dos dados das rodadas do código SUPIM poderia ser feito normalmente utilizando alguma técnica, como Inverse Distance Weighting [10-11] aplicada a cada ponto da grade. No entanto, verificou-se que a complexidade de processamento inviabilizaria a geração da grade em tempo razoável. Assim, foi desenvolvida uma técnica chamada “explosão de perfil” [13] na qual os pontos simulados são associados aos pontos da grade próximos a ele. A ponderação para estimar a concentração para cada ponto da grade então utilizará apenas alguns dos pontos simulados identificados como próximos, de acordo com o alcance desejado (relacionado a “ordem de explosão”). Essa abordagem possibilitou a redução considerável do tempo de processamento requerido, mantendo contudo a coerência no resultado final. A partir da grade tridimensional gerada pelo DAVS, podem ser calculados mapas bi-dimensionais com a estimativa do conteúdo eletrônico total (TEC), acumulando os valores de concentração eletrônica em altura. A figura 5 ilustra a técnica utilizada, para diferentes “ordens de explosão”, resultando em mapas de conteúdo eletrônico total com diferentes coberturas.

O DAVS é aplicado paralelamente para cada hora simulada, gerando ao total 24 mapas com a previsão do conteúdo eletrônico total. O paralelismo utilizado se deu de forma semelhante ao utilizado pelo código SUPIM, utilizando o gerenciador de recursos OAR para disparar cada um dos processos, sem o uso de MPI.

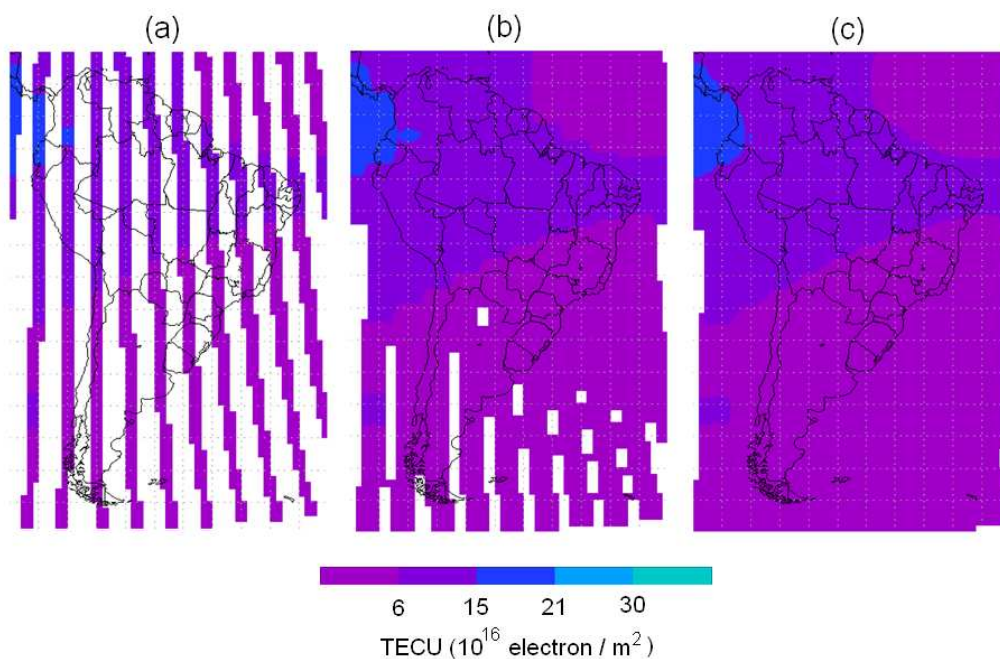


Figura 5 – Técnica “explosão de perfil” de primeira (a), segunda (b) e terceira ordem (c) aplicada para geração de mapas de conteúdo eletrônico total

Antes de aplicar o DAVS para geração da grade homogênea de concentração eletrônica a cada hora, foi verificado que o código SUPIM gerava pontos de simulação em coordenadas magnéticas conforme International Geomagnetic Reference Field (IGRF) [12] que, ao serem passados para coordenadas geográficas, acarretavam um deslocamento longitudinal muitas vezes considerável. Esse deslocamento, decorrente da inclinação apresentada pelas linhas de campo magnético, está associado a um atraso ou adiantamento temporal nos pontos simulados, que em algumas regiões poderia ser superior a 30 minutos. A figura 6 ilustra esses atrasos em minutos para o globo. É possível perceber algumas regiões chegam a ter atrasos e adiantamentos maiores que uma hora do local simulado (pontos ao longo do equador magnético). Para reduzir esse efeito na geração de mapas a cada hora, pontos de simulação atrasados ou adiantados mais de 30 minutos foram transferidos para a simulação da hora anterior ou seguinte. O efeito desse ajuste é mostrado na figura 7, na qual é possível verificar que o máximo atraso ou adiantamento apresentado é menor que 30 minutos.

Ajuste Temporal da Simulacao

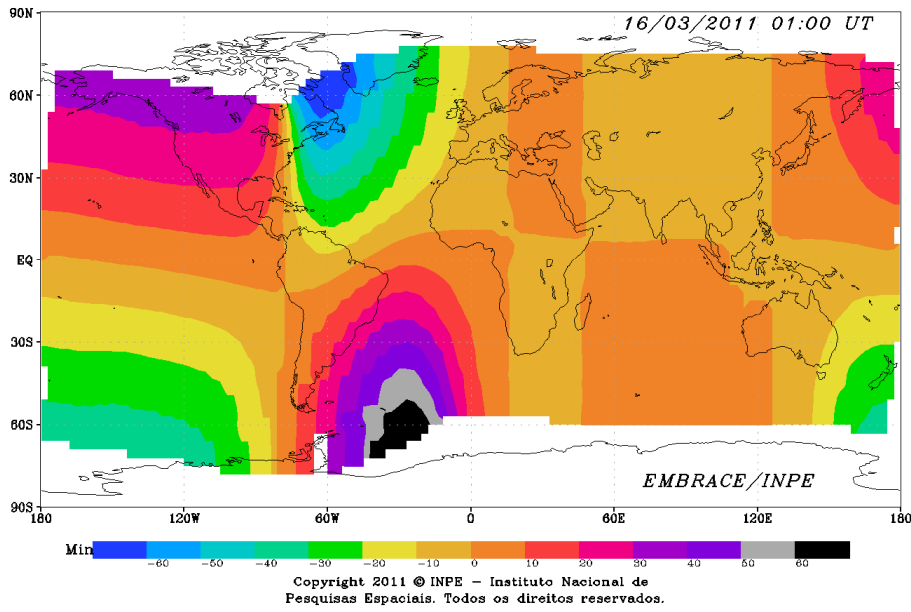


Figura 6 – Atrasos temporais dos pontos simulados antes do ajuste de tempo

Ajuste Temporal da Simulacao

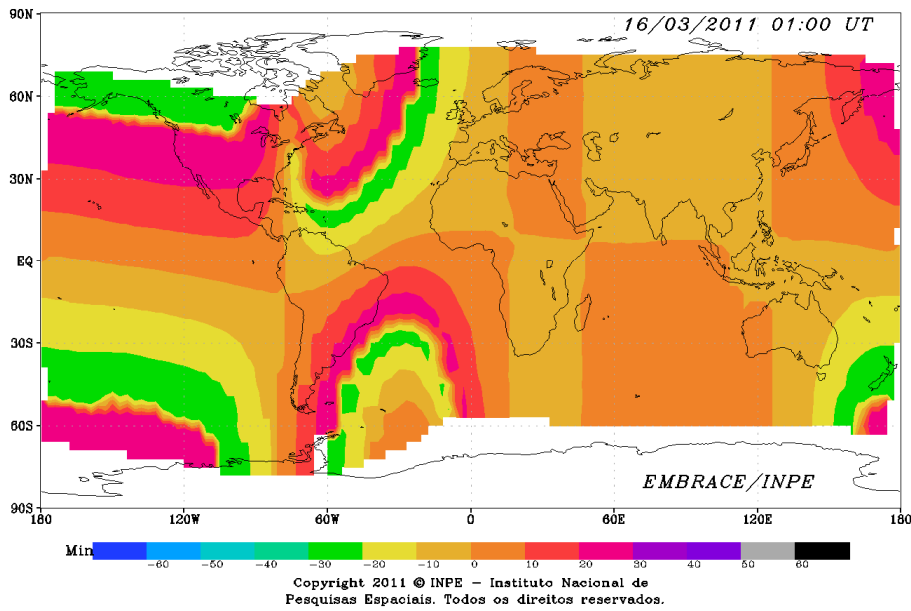


Figura 7 – Atrasos temporais dos pontos simulados após ajuste de tempo

3.6 Geração de imagens

A execução do DAVS gera arquivos em formato que pode ser utilizado pelo software GrADS [6]. Assim, foi desenvolvido um script utilizando a GrADS scripting language para automaticamente gerar mapas para a América do Sul ou mapas Globais de acordo com um formato pré-definido de escala de cores, resoluções e textos. As figuras 8 e 9 ilustram o formato adotado até o momento para os mapas horários de conteúdo eletrônico total. Salienta-se, no entanto, que os dados gerados pelo DAVS fornecem dados tridimensionais. Assim, seria possível sem maiores dificuldades gerar, por exemplo, o perfil vertical de concentração eletrônica em uma determinada coordenada geográfica, ou um mapa contendo a seção transversal para a concentração eletrônica fixando-se uma latitude ou uma longitude.

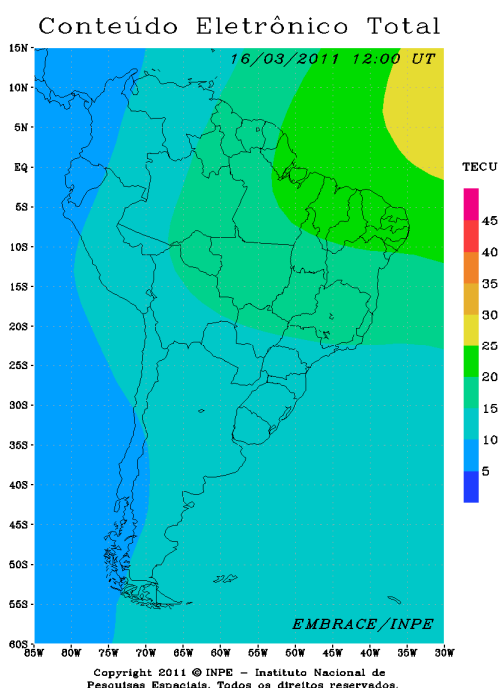


Figura 8 – Exemplo de imagem para simulação do conteúdo eletrônico total para América do Sul

Conteúdo Eletrônico Total

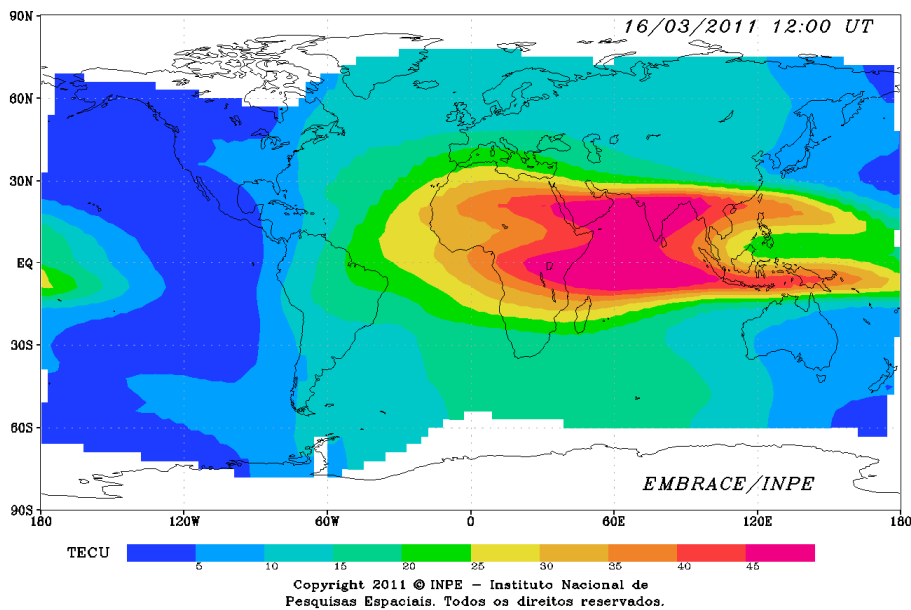


Figura 9 – Exemplo de imagem para simulação do conteúdo eletrônico total para Globo

3.7 Finalização da simulação

Após a geração dos mapas, a simulação realiza algumas operações antes de concluir. Basicamente, as imagens são copiadas para um servidor de destino, onde ficarão disponíveis para acesso externo. A seguir, os arquivos gerados durante as rodadas do código SUPIM são mantidos como estão, comprimidos (como padrão), ou excluídos. Por fim, um email de notificação é enviado para contas pré-definidas, contendo a cópia do arquivo de “log” da simulação.

4 POSSÍVEIS PROBLEMAS E SOLUÇÕES

O sistema desenvolvido apresentou bom funcionamento e estabilidade durante um período de mais de um mês de testes diários. No entanto, pelo fato de ser bastante complexo e depender de sub-sistemas de software e hardware, alguns problemas poderão ocorrer durante sua operação. A seguir são descritos alguns problemas que já ocorreram durante o processo de desenvolvimento, e as possíveis soluções encontradas. Antes de se tentar executar algum procedimento é conveniente avaliar os arquivos de “log” que são gerados a cada simulação. Eles são capazes de auxiliar na identificação da falha e facilitar a solução. Estes arquivos textuais, nomeados simplesmente “log”, são gerados durante a execução de uma simulação no diretório correspondente ao tipo de simulação em execução (“/home/supim-davs/SUPIM-DAVS/runs/Globe” ou “/home/supim-davs/SUPIM-DAVS/runs/SouthAmerica”), e ao final são transferidos para o diretório correspondente ao dia juliano simulado.

4.1 Reinicialização do cluster

A reinicialização do cluster, ou ao menos do nó zero pode vir a ser necessária futuramente, para possíveis atualizações de sistema operacional ou outras atividades. Esse procedimento deve ser feito cuidadosamente para evitar problemas com a execução das simulações. Um problema importante que poderá ocorrer é a falha na execução do gerenciador de dispositivos OAR [9], que poderá deixar de funcionar, possivelmente devido a um problema com banco de dados instalado. Essa situação já ocorreu no passado, devido a falta de compatibilidade do banco de dados mysql instalado. Assim, deve-se verificar a funcionalidade do OAR, utilizando as instruções “oarstat” e “oarnodes -s”. Além do possível problema com o OAR, provavelmente ocorrerá um problema com a licença de ativação do IDL [8], necessária para rodar o programa de download automático dos arquivos de fluxo solar. Neste caso, a

solução é executar os seguintes passos para ativação da licença IDL usando o gerenciador de licenças "lmgrd":

```
"killall lmgrd"
```

```
"/usr/local/itt/idl/idl80/bin/lmgrd -l /var/log/lmgrd.log -c  
/usr/local/itt/license/license.dat &"
```

4.2 Nós do cluster com problema

Depois de intensa utilização do cluster através do gerenciador de recursos OAR [9], já ocorreu de alguns dos nós serem classificados como "Suspected" pelo OAR. Essa classificação indica uma possível falha de hardware, a ser averiguada. Nós em estado "Suspected" não recebem processos pelo OAR e poderão fazer falta durante as simulações. Caso após algum tempo esses nós não voltarem ao estado "Alive" deve-se logar como usuário ROOT em cada nó "Suspected" ir para o diretório "/etc/init.d" e rodar "./oar-node restart". Para verificação do estado dos nós e do sucesso do procedimento, pode ser utilizado a instrução "oarnodes -s".

4.3 Falta de espaço em disco

A existência de espaço para armazenamento dos arquivos gerados em cada simulação é crítica, já que o volume de dados é muito elevado. Em simulações globais, é usual o espaço em disco ocupado pelos arquivos chegar a 13GB. Para simulações da América do Sul, o espaço alcança em torno de 5,4GB. Assim, um dos parâmetros da simulação indica se os arquivos localizados no diretório "DAVS/InTest" serão ou não compactados ao final. Caso sejam, o espaço de armazenamento necessário se reduz para 3,5GB em simulações globais e 1,6GB em simulações para América Latina. Caso não haja espaço em disco suficiente na partição "/home" de no mínimo 28GB, a simulação simplesmente não será executada com sucesso. Deve-se avaliar continuamente o espaço existente para garantir a operação do sistema. É possível monitorar o espaço disponível usando a instrução "df -h /home". Para

averiguar o espaço ocupado por diretórios, usamos “du -s -h <diretório avaliado>”. Caso o espaço em disco atingir um valor crítico próximo a 28GB, sugere-se utilizar o parâmetro “compressFiles” igual a “-1”, de forma a não mais manter os arquivos de simulação armazenados.

4.4 Ajuste da hora do sistema

O Brasil adota anualmente horário de verão durante alguns meses do ano. Isso gera a necessidade de atualização da hora do sistema, que poderá normalmente ser feito utilizando o comando “date”. No entanto, a mudança na hora do sistema gera a desativação do serviço CRON do Linux. Essa desativação fará com que as simulações não sejam mais disparadas automaticamente. Para corrigir esse problema, basta reinicializar o serviço através da instrução “/etc/init.d/cron restart”.

4.5 Simulação iniciada com cluster ocupado

O cluster disponível para as simulações é compartilhado com outras atividades. Caso uma simulação inicie com alguns dos nós do cluster ocupados, ela irá utilizar os nós livres e ficará verificando periodicamente a liberação de outros nós. Isso ocasionará um atraso nos resultados, e caso esse atraso seja superior a um limite pré-estabelecido (atualmente de 8 horas), ocasionará a finalização da simulação. Caso outra simulação disparada pelo mesmo usuário “supim-davs” esteja ativa, a simulação recentemente disparada aguardará a conclusão da simulação corrente para iniciar a ocupação dos nós com o processamento requerido. Isso porque a execução simultânea de mais de uma simulação causa o mau funcionamento de ambas, já que o usuário que dispara os processos é o mesmo, e processos de uma simulação poderão ser mal interpretados por outra.

4.6 Arquivo com dados de fluxo solar ausentes

Antes de todo processamento da simulação iniciar, é necessária a disponibilização de dados relativos ao fluxo solar para o dia simulado. Essas informações são guardadas em “/home/supim-davs/SUPIM-DAVS/runs/TTTT/XXXX/YYYY/SUPIM/Inputs/SFLUX/”, onde “TTTT” refere-se ao tipo de simulação executada (“Globe” ou “SouthAmerica”), “XXXX” é o ano avaliado, e “YYY” o dia juliano daquele ano. Para datas passadas, normalmente não haverá dificuldades em obter os dados de fluxo solar. No entanto, isso não acontece para datas futuras. Normalmente, os dados de fluxo solar para um determinado dia só estarão disponíveis por volta de 01:00h UT daquele dia, e os valores são ajustados por volta das 23:00h UT do mesmo dia. Quando disparada, a simulação ficará aguardando que informações válidas estejam disponíveis, fazendo o download dos dados a cada minuto. Somente quando as informações de fluxo solar estiverem disponíveis o resto do processamento é iniciado.

4.7 Falta de convergência do SUPIM

Atualmente cada simulação utiliza 28 rodadas do SUPIM linearmente espaçadas a cada 2 graus de longitude para a América do Sul e 61 rodadas espaçadas a cada 6 graus de longitude para geração dos mapas globais. A interpolação de dados que o DAVS realiza posteriormente é capaz de absorver a ausência de até 3 rodadas do SUPIM em longitudes adjacentes para a América do Sul, como pode ser visto na figura 10. Outras falhas poderão ocorrer mantendo o mapa final completo, desde que não adjacentes em longitude. Para maiores falhas adjacentes durante a execução do código SUPIM, os mapas gerados não formam imagens completas para a região avaliada. Nesta versão do sistema, não são averiguadas a qualidade da imagem gerada pelo sistema. Já para o caso da geração de mapas globais, como as rodadas do SUPIM são executadas com um distanciamento maior, a ausência de apenas 1 rodada do SUPIM já ocasiona a ausência de regiões

com valores calculados para o conteúdo eletrônico nos mapas, como pode ser vista na figura 11.

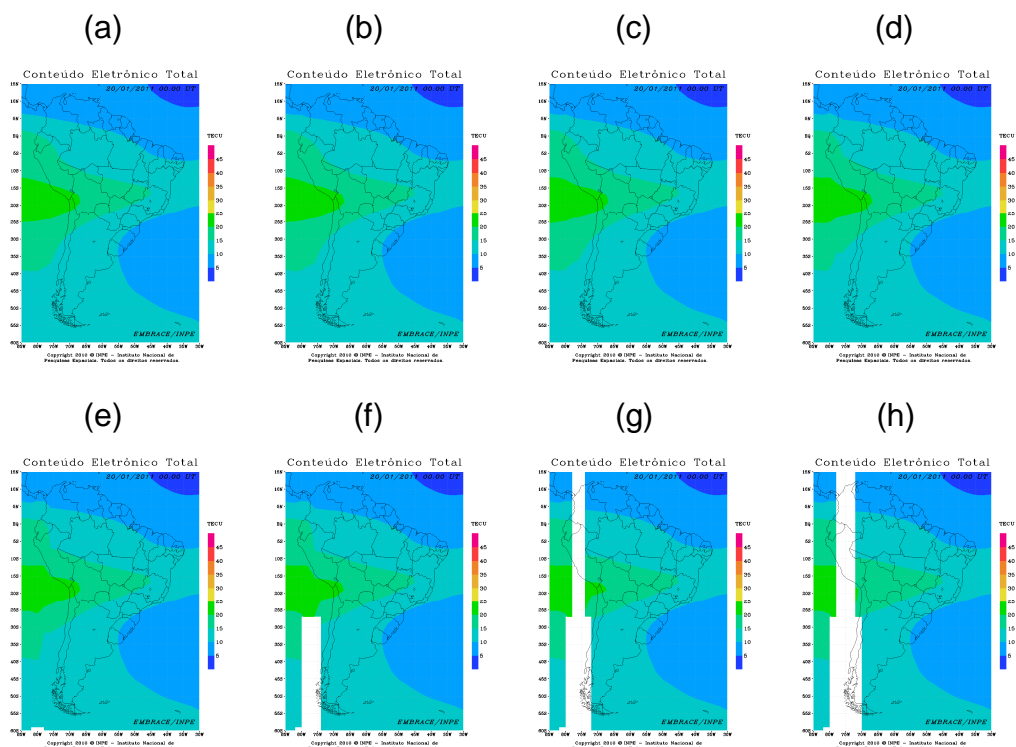


Figura 10 – Geração de imagens para América do Sul: (a) com todas 28 rodadas do SUPIM presentes, (b) com 1 rodada faltante, (c) com 2 rodadas adjacentes faltantes, (d) com 3 rodadas adjacentes faltantes, (e) com 4 rodadas adjacentes faltantes, (f) com 5 rodadas adjacentes faltantes, (g) com 6 rodadas adjacentes faltantes, (h) com 7 rodadas adjacentes faltantes

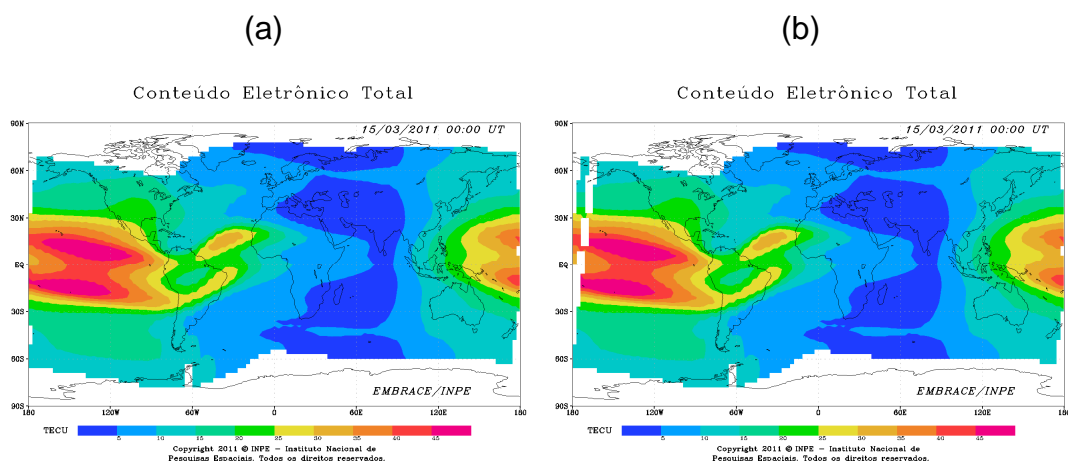


Figura 11 – Geração de imagens para Globo: (a) com todas 61 rodadas do SUPIM presentes, (b) com 1 rodada faltante

4.8 Falta de memória durante processamento do DAVS

O DAVS [13] gera arquivos com dados tridimensionais homoganeamente espaçados da concentração iônica e/ou eletrônica resultante de uma técnica de interpolação desenvolvida especialmente para os pontos de simulação gerados pelo código SUPIM, assim como dados bidimensionais com estimativa do conteúdo eletrônico total sobre esses dados. Essa técnica requer grande quantidade de memória primária, e poderá resultar em erro quando a região analisada for muito grande, quando a resolução utilizada for muito grande, ou quando a “explosão” aplicada aos dados de simulação for muito elevada. Para resolver o problema de falta de memória durante o processamento do DAVS, é necessário diminuir a quantidade de rodadas do SUPIM a serem utilizadas, ou alterar os parâmetros configuráveis da simulação, mostrados na figura 12, que determinam a forma como os dados serão gerados pelo DAVS. Para esta alternativa, as opções seriam o aumento dos valores de “lonStep”, “latStep” ou “elevStep” (ou seja, diminuir a resolução da interpolação resultante), ou a redução do valor de “explosion” (ou seja, reduzir a sobreposição entre pontos de simulação). Além dessas alternativas, é possível também diminuir a área avaliada.

```
# DAVS configuration - default values for south america
# are 24, 0, -90, -25, 2, -65, 20, 2, 90000, 1000000, 10000, 2
numberTimeSteps=24
initialTime=0
lonBegin=-90
lonEnd=-25
lonStep=2
latBegin=-65
latEnd=20
latStep=2
elevBegin=90000
elevEnd=1000000
elevStep=10000
explosion=2
```

Figura 12 – Parâmetros de configuração do DAVS

4.9 Falta dos dados da simulação do dia anterior

O processo de ajuste temporal da simulação, descrito no item “4.5 Execução paralela do DAVS” ocasiona um remanejamento dos pontos de simulação para grades geradas para diferentes horas. Por exemplo, os pontos “cedidos” para uma grade correspondente a hora seguinte são também “recebidos” da grade da hora anterior. Isso faz com que os mapas permaneçam completos em sua cobertura, a exceção do primeiro e último mapa. No caso do primeiro mapa (0 UT), não existem pontos simulados adiantados do mapa da hora anterior (23 UT do dia anterior) para serem “recebidos”. Esse fenômeno gera mapas incompletos, como ilustrado nas figuras 13 e 14. Para resolver esse problema, os dados da hora 23 UT de cada simulação são disponibilizados para a simulação seguinte, de forma a manter o mapa da hora 0 UT completo. Esses dados são copiados para o diretório “token” de cada simulação. Caso os dados no diretório “token” não estiverem disponíveis, não haverá como gerar mapas completos para a hora 0 UT. No caso do último mapa (23 UT), o código SUPIM gera pontos de simulação para hora 24 UT, que são utilizados para o preenchimento do mapa relativo a hora 23 UT.

Conteúdo Eletrônico Total

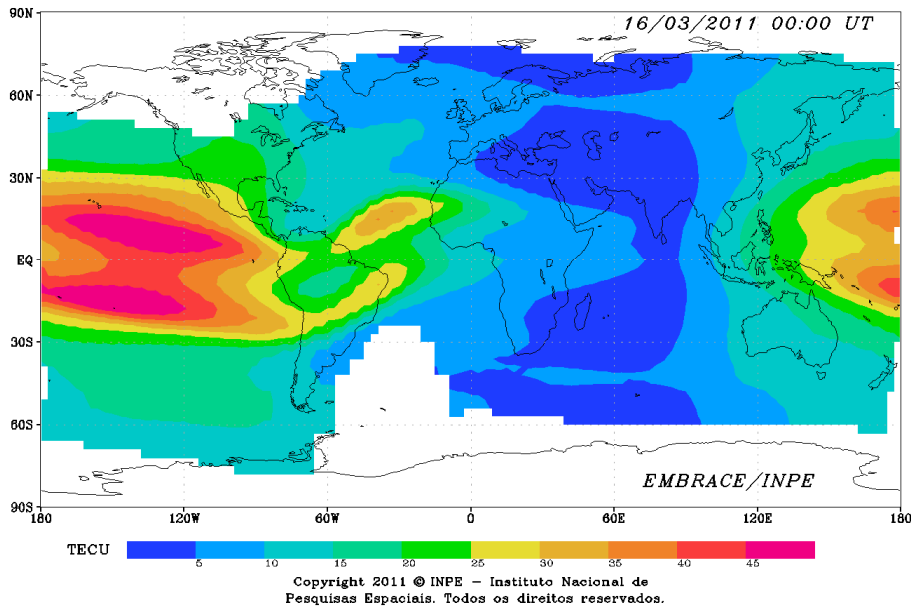


Figura 13 – Geração de mapas de TEC para Globo sem dados de simulação do dia anterior

Conteúdo Eletrônico Total

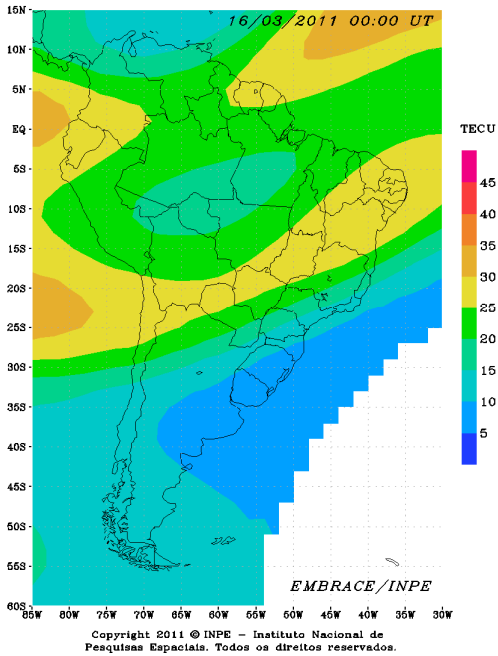


Figura 14 – Geração de mapas de TEC para América do Sul sem dados de simulação do dia anterior

4.10 Demora excessiva durante processamento

Após a submissão de processos executados paralelamente, a simulação aguarda durante um intervalo de tempo pré-estabelecido pelos resultados. Atualmente esse tempo equivale a 8 horas. Após esse período, a simulação é encerrada automaticamente e email é enviado contendo o “log” da simulação. Ocorreram casos em que o número de nós disponíveis para processamento era bem inferior ao necessário, devido à ocupação do cluster com outros processamentos. Isso ocasionou um atraso significativo para conclusão dos processos. Assim, é importante que seja verificado se o atraso esperado para conclusão da simulação poderá ou não ser retardado por um intervalo de tempo longo, ajustando a temporização do sistema se necessário.

4.11 Escala de cores incompatível

Os mapas de conteúdo eletrônico gerados obedecem uma escala de cores pré-estabelecida. Inicialmente essa escala está configurada para um limite de 50 unidades de TEC (TECU – equivalente a 1016 elétrons / m²). No entanto, com o aumento da atividade solar, os valores estimados para o TEC poderão ultrapassar 200 TECU, como ocorreu durante o último período de máximo solar. Assim, haverá necessidade de ajustar a escala de cores, de forma a abranger uma faixa de valores mais elevados. Isso pode ser feito alterando a instrução 'set clevs 5 10 15 20 25 30 35 40 45' dos arquivos escritos em GrADS scripting language, nomeados “scriptPrintImageSouthAmerica.gs” e “scriptPrintImageGlobe.gs”, contidos dentro do diretório “install”, subdiretório “GrADS”. Assim, por exemplo, para aumentar os valores abrangidos para atingir até 100 TECU, mantendo a distribuição de cores inalterada, a instrução mencionada deveria conter a informação 'set clevs 10 20 30 40 50 60 70 80 90'.

4.12 Falha ao enviar imagens ao destino final

Após a geração dos mapas de conteúdo eletrônico, as imagens geradas são copiadas ao servidor de destino para acesso externo. No entanto, não há garantias que o envio será bem sucedido. Possíveis problemas durante essa transmissão envolvem a infra-estrutura de rede disponível no cluster, no servidor de destino ou no usuário/senha cadastrados no servidor de destino. Caso o envio dos arquivos não seja feito com sucesso, a simulação é interrompida e um email de notificação é enviado, contendo dados de “log” da simulação.

5 CONCLUSÃO

Este manual documentou a estrutura utilizada para rodar o sistema desenvolvido para previsão da dinâmica ionosférica dentro do Programa de Clima Espacial do INPE, batizado de SUPIM-DAVS. Detalhes de operação e uso do sistema foram mostrados. A seguir, foram relacionados os principais problemas e soluções encontrados durante o período de testes ao qual o sistema foi submetido. Em anexo, os principais processos utilizados para instalação e configuração de ferramentas de software necessárias ao funcionamento do sistema foram documentados.

Esperamos que este manual sirva como guia para a equipe responsável pela operação diária do sistema de previsão da dinâmica ionosférica para a região da América do Sul e Globo. Alguns ajustes e refinamentos poderão ser feitos ao sistema, mas acreditamos que a sua estrutura principal será mantida.

Como trabalhos futuros, podemos destacar a aquisição, armazenamento e utilização de dados observacionais, oriundos de estações GPS ou ionossondas, para alimentar um sistema de assimilação de dados, atualmente em desenvolvimento.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] BAILEY, G. **Supim sheffield university plasmasphere ionosphere Model**. Disponível em <<http://gbailey.staff.shef.ac.uk/supim.html>>. Acesso em março de 2011.

[2] SOUZA, J. R. **Modelagem ionosferica em baixas latitudes no Brasil**. 1997. 182 p. (INPE-6395-TDI/611). Tese (Doutorado em Geofísica Espacial) - Instituto Nacional de Pesquisas Espaciais, Sao Jose dos Campos. 1997.

[3] INTEL CORPORATION. **Compilers and libraries**. Disponível em <<http://software.intel.com/en-us/intel-sdp-home/>>. Acesso em março de 2011.

[4] JAVA. **O que é o Java?**. Disponível em <http://www.java.com/pt_BR/>. Acesso em março de 2011.

[5] SNIR, M. **MPI - the complete reference MPI-2**. Cambridge, Massachusetts: MIT Press, 1998.

[6] THE GRID ANALYSIS AND DISPLAY SYSTEM (GrADS). **Overview of GrADS**. Disponível em <<http://www.iges.org/grads/>>. Acesso em março de 2011.

[7] SOLAR IRRADIANCE PLATFORM. **Downloading and Installing the Solar Irradiance Platform**. Disponível em: <<http://www.spacewx.com/solar2000.html>>. Acesso em março de 2011.

[8] Scientific data visualization software from ITTVIS. Disponível em <<http://www.itvis.com/language/en-US/ProductsServices/IDL.aspx>>. Acesso em março de 2011.

[9] OAR Documentation User Guide. Disponível em <http://oar.imag.fr/users/user_documentation.html>. Acesso em março de 2011.

[10] THOMPSON, J. F.; SONI, B. K.; WEATHERILL, N. P. (eds.) **Handbook of grid generation**. Boca Raton: CRC Press, 1999.

[11] GEORGE Y. LU; WONG, D. W. An adaptive inverse-distance weighting spatial interpolation technique. **Computers & Geosciences**, v. 34, n. 9, Sept. p. 1044-1055, 2008.

[12] INTERNATIONAL GEOMAGNETIC REFERENCE FIELD (IGRF-11). Disponível em <swdcwww.kugi.kyoto-u.ac.jp/igrf/index.html>. Acesso em março de 2011.

[13] PETRY, A.; PEREIRA, A. G.; VIERO, F.; SOUZA, J. R. **Image generation and visualization system for ionosphere dynamics**. São José dos Campos: INPE, 2011-05-31. (INPE ePrint sid.inpe.br/mtc-m19/2011/05.30.13.07). Disponível em: <<http://urlib.net/8JMKD3MGP7W/39PTGAS>>. Acesso em: 09 jun. 2011.. Aceito para publicação no 12th International Congress of the Brazilian Geophysical Society, Rio de Janeiro, Brasil, 15-18 de agosto de 2011.

APÊNDICE A – INSTALAÇÃO DE FERRAMENTAS DE SOFTWARE

O sistema SUPIM-DAVS foi instalado e funciona através de uma conta criada para este fim, que pode ser acessada a partir da rede interna do INPE utilizando número IP: 200.132.24.245 e username: supim-davs. Antes do desenvolvimento e operação do sistema, foi necessário realizar a instalação e configuração das seguintes ferramentas de software:

- compilador C e Fortran [3];
- ambiente JAVA [4];
- implementação do Message Passing Interface (MPI) versão 2 [5];
- software Grid Analysis and Display System (GrADS) [6].

A instalação e configuração de cada uma das ferramentas mencionadas é detalhada a seguir, assim como a metodologia para geração de chaves para conexão a servidores através de Secure Shell (SSH).

Compilador C e Fortran

O sistema desenvolvido é composto por sub-sistemas que foram escritos em linguagens de programação diferentes. Dentre essas linguagens há programas escritos em linguagem C e Fortran. Assim, para que esses programas possam ser compilados e executados, deve-se instalar e configurar apropriadamente software compiladores. Para o compilador Intel [3], os passos necessários a sua correta instalação e configuração são:

a) Copiar os arquivos de instalação do compilador para uma pasta local. Foi utilizada a pasta /home/supim-davs/SUPIM-DAVS/install/Compilador/intel

b) Descompactar os arquivos usando os programas gzip e tar:

Para o compilador Fortran as instruções executadas foram: "gzip -d I_fc_p_10.1.013_intel64.tar.gz" e "tar -xf I_fc_p_10.1.013_intel64.tar".

Para o compilador C as instruções executadas foram: "gzip -d I_cc_p_10.1.013_intel64.tar.gz" e "tar -xf I_cc_p_10.1.013_intel64.tar".

c) Se necessário, instalar um compilador C++ usando: "sudo apt-get install build-essential", e também instalar o pacote contendo "libstdC++5". Este pacote pode ser instalado usando o "Gerenciador de Pacotes Synaptics".

d) Rodar o script de instalação de dentro do diretório descompactado: "cd I_fc_x_10.1.013_intel64" e após "./install.sh". Escolher a opção 3 (instalar como usuário corrente), e depois opção 1 (install) fornecendo um número serial para instalação. A seguir, executar a instalação típica (opção 1). Fazer o mesmo para o diretório do compilador C "I_cc_x_10.1.013_intel64".

e) Ajustar as variáveis de ambiente: "source <diretório_instalação>/intel/fce/10.1.013/bin/ifortvars.sh" para Fortran e "source <diretório_instalação>/intel/cce/10.1.013/bin/iccvars.sh" para C++. Após isso, os executáveis "icc" e "ifort" poderão ser invocados de qualquer lugar.

f) Colocar o ajuste de variáveis de ambiente do passo anterior no arquivo ".bashrc" para automatizar o processo a cada login. Também incluir as seguintes linhas no arquivo ".bashrc" para definir as variáveis de ambiente:

```
export f90=ifort
export F90=ifort
export f77=ifort
export F77=ifort
export CC=icc
```

Java Runtime Environment (JRE)

Alguns programas foram escritos em linguagem de programação JAVA [4]. Assim, é importante que o ambiente Java Runtime Environment (JRE) esteja instalado e funcionando corretamente. Para isso, os passos necessários são:

- a) A partir de uma janela de console, rodar o comando “sudo apt-get install sun-java6-bin sun-java6-jre”. Fornecer a senha de root para executar a instrução e confirmar as alterações solicitadas.
- b) Automaticamente os arquivos serão obtidos da internet e instalados no sistema operacional. Aceite a licença de software mostrada.

Implementação do Message Passing Interface (MPI) versão 2

A paralelização na execução de programas pode utilizar ferramentas distintas. Uma das técnicas mais importantes e utilizadas é através do Message Passing Interface [5], que pode ser instalada através dos passos:

- a) Copiar os arquivos de instalação para um diretório local e descompacta-los usando: "tar xzf mpich2-1.0.5p4.tar.gz". Os arquivos serão descompactados em um subdiretório “mpich2-1.0.5p4”.
- b) Criar a pasta de destino da instalação: "mkdir <diretório_instalação>". Entrar no diretório criado durante a descompactação "cd mpich2-1.0.5p4" e configurar o MPICH2 (a partir da pasta de descompactação): "./configure - prefix=<diretório_instalação>".
- c) Rodar a instrução "make" e depois "make install".

d) Para automatizar a busca pelo executável "mpiexec" (sem precisar dar o caminho completo até o executável), incluir no arquivo ".bashrc" a linha:

```
export PATH=<diretório_instalação>/bin:$PATH
```

e) Criar o arquivo ".mpd.conf" na pasta "home" do usuário, contendo exclusivamente o texto "secretword=<secretword>", onde <secretword> é uma palavra qualquer. A seguir, trocar as permissões do arquivo, usando a instrução "chmod 600 .mpd.conf".

f) Antes de disparar os programas com "mpiexec", deve-se inicializar os deamons do MPICH2 nos nós que serão usados. Para isso, crie na pasta "home" do usuário o arquivo texto "mpd.hosts" contendo em linhas separadas os nomes dos processadores que serão usados. Por exemplo, no cluster do CRS, pode-se utilizar o texto:

```
node0:4 ifhn=192.168.100.100
node1:4 ifhn=192.168.100.1
node2:4 ifhn=192.168.100.2
node4:4 ifhn=192.168.100.4
node5:4 ifhn=192.168.100.5
node6:4 ifhn=192.168.100.6
node7:4 ifhn=192.168.100.7
node8:4 ifhn=192.168.100.8
node9:4 ifhn=192.168.100.9
node10:4 ifhn=192.168.100.10
node12:4 ifhn=192.168.100.12
node13:4 ifhn=192.168.100.13
node14:4 ifhn=192.168.100.14
node15:4 ifhn=192.168.100.15
node16:4 ifhn=192.168.100.16
node17:4 ifhn=192.168.100.17
```

```
node18:4 ifhn=192.168.100.18
node19:4 ifhn=192.168.100.19
node20:4 ifhn=192.168.100.20
node11:4 ifhn=192.168.100.11
```

Note-se que a instrução "mpdtrace" ou "mpdtrace -l" mostra os processadores disponíveis para processamento paralelo. Para finalizar os deamons rodando nesses processadores, usar a instrução "mpdallexit". Outra forma para inicializar os deamons é utilizar `mpdboot -n <numero>`, onde <numero> representa a quantidade de processadores que serão inicializadas com o daemon.

Ao utilizar paralelismo híbrido MPI e OPENMP, foi verificado que a compilação com diretiva "-openmp" fazia com que o programa não fosse executado via "mpiexec" a partir de nós diferentes do nó 0. Assim, o nó 0 (que não deve ser usado na prática), é adicionado a lista de nós usados no mpich2, mas ao disparar "mpiexec" usa-se a diretiva "-machinefile <arq>", onde <arq> é um arquivo-texto que contém apenas os nós diferentes do 0, por exemplo:

```
node5:1
node6:1
...
```

Assim, apesar do nó 0 fazer parte do anel de nós disponíveis para o MPI2, ele na prática não é usado. A informação ":1" indica que cada processo rodará em um nó diferente. Assim, teremos o MPI2 rodando 1 processo em cada nó e, também em cada nó 4 threads OpenMP. Além desse detalhe, a utilização de threads com MPI requer que a inicialização seja feita através de "call `MPI_Init_thread(MPI_THREAD_MULTIPLE,provided,ierr)`", e não "`MPI_Init()`", de forma a minimizar as chances de problemas.

g) Após os deamons ativos nos processadores, é possível disparar a execução de um programa MPI2 chamando "mpiexec -n <procs> <file>", onde <procs> é o número de processos paralelos que irão rodar e <file> o nome do executável, incluindo o caminho. Se o executável estiver na pasta corrente, ainda assim deve-se fazer <file> ser "./file". É possível também forçar a execução em um conjunto pre-definido de nós no momento de invocar "mpiexec", usando o parâmetro -machinefile <file>, onde <file> é um arquivo texto contendo os processadores que irão rodar o arquivo.

h) Para utilização o MPICH2 através do gerenciador de recursos OAR [9], inicialmente deve-se solicitar os recursos desejados, por exemplo, "oarsub -l -l nodes=4,walltime=5:00:00", que solicita os recursos: 4 nós por 5 horas. A seguir, deve-se inicializar o MPICH2 nos nós que foram alocados, por exemplo, "mpdboot --rsh=oarsh --totalnum=4 --file=\$OAR_FILE_NODES", que inicializa 4 nós (descritos pelo arquivo determinado na variável \$OAR_FILE_NODES que foi automaticamente preenchida na solicitação dos recursos). Por fim, pode-se verificar se os nós foram disponibilizados: "mpdtrace -l" e rodar o mpi normalmente, usando por exemplo "mpiexec -n 4 ./prog.exe".

Para rodar scripts sem utilização do MPI, pode-se usar, por exemplo, "oarsub -l nodes=4,walltime=5:00:00 script.sh". Pode-se também verificar o status das execuções utilizando "oarstat" ou "oarstat -f" ou "oarstat -j X -f", onde X representa o número do job.

Software Grid Analysis and Display System (GrADS)

O GRADS [6] é um software livre para visualização de dados em mapas terrestres. Sua instalação pode ser realizada através das instruções:

a) Copiar os arquivos de instalação para um diretório local e descompactar os executáveis contidos no arquivo "grads-2.0.a8-bin-i686-pc-linux-gnu.tar.gz" executando: "cd <diretório_instalação>", "gzip -d grads-2.0.a8-bin-i686-pc-linux-

gnu.tar.gz", "tar -xf grads-2.0.a8-bin-i686-pc-linux-gnu.tar". Será criado o diretório "grads-2.0.a8" contendo os arquivos executáveis necessários.

b) Após a descompactação dos binários do GRADS, é importante também descompactar o arquivo "data.tar.Z" em /usr/local/lib/grads/. Para isso, deve-se primeiramente criar a pasta "grads" no local especificado: "cd /usr/local/lib", "sudo mkdir grads", "cd grads". A seguir, devemos descompactar o arquivo dentro da pasta criada: "cp <path>/data.tar.Z ./", "sudo uncompress data.tar.Z", "sudo tar xvf data.tar"

c) Para automatizar a busca pelo executável "grads" (sem precisar dar o caminho completo até o executável), incluir no arquivo ".bashrc" a linha:

```
export PATH=<diretório_instalação>/bin:$PATH
```

Chave de Conexão via SSH

Durante a operação do sistema, há necessidade de eventualmente logar remotamente em algum servidor a fim de, por exemplo, fazer transferência de dados. Assim, se for definida uma chave de conexão via SSH é possível fazer essa conexão de forma automatizada, sem a necessidade de intervenção humana. O procedimento para estabelecer a chave de conexão é:

a) Criar a pasta .ssh no home do computador cliente e servidor: "mkdir -p \$HOME/.ssh"

b) Alterar as permissões da pasta no computador cliente e servidor: "chmod 0700 \$HOME/.ssh"

c) Gerar chaves pública e privada apenas no computador cliente: "ssh-keygen -t rsa -f \$HOME/.ssh/id_rsa -P "" (atenção para duas aspas simples). As chaves geradas estarão nos arquivos id_rsa (privada) e id_rsa.pub (pública).

d) Copiar o arquivo id_rsa.pub para o servidor de destino, que também deverá ter a pasta .ssh, e após executar "cat id_rsa.pub >> \$HOME/.ssh/authorized_keys2" e "cat id_rsa.pub >> \$HOME/.ssh/authorized_keys".

e) Alterar as permissões dos arquivos no computador servidor usando "chmod 0600 \$HOME/.ssh/authorized_keys2" e "chmod 0600 \$HOME/.ssh/authorized_keys".

f) Pode-se testar o funcionamento fazendo "ssh -i \$HOME/.ssh/id_rsa <servidor>", onde <servidor> é o endereço do computador servidor de destino.