



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m19@80/2010/07.19.13.43-NTC

ANÁLISE E PARALELIZAÇÃO DO CÓDIGO SUPIM

Adriano Petry

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/37SMSTS> >

INPE
São José dos Campos
2010

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

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m19@80/2010/07.19.13.43-NTC

ANÁLISE E PARALELIZAÇÃO DO CÓDIGO SUPIM

Adriano Petry

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/37SMSTS> >

INPE
São José dos Campos
2010

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 1: Simplificação do código SUPIM.....	11
Figura 2: Aplicação de diretivas OpenMP para otimização de um laço de execução	14
Figura 3: Região abrangida por duas simulações do código SUPIM	16

LISTA DE TABELAS

	<u>Pág.</u>
Tabela 1: Tempos de execução do código original variando otimizações automáticas ...	8
Tabela 2: Sub-rotinas críticas	12
Tabela 3: Sub-rotinas críticas ordenadas pelo tempo médio consumido em cada chamada	13
Tabela 4: Tempos de execução do código com sub-rotinas críticas paralelizadas através de OpenMP e junção de laços comuns	15
Tabela 5: Tempos de execução do código para 3 simulações distintas, usando MPI e OpenMP	18

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	5
2 VISÃO GERAL DO SUPIM	6
3 ANÁLISE DO PERFIL DE EXECUÇÃO DO CÓDIGO SUPIM.....	7
3.1 Temporização	7
3.2 Execução em duas rodadas	9
4 ATIVIDADES DE PARALELIZAÇÃO.....	14
4.1 Paralelização usando OpenMP	14
4.2 Paralelização usando MPI.....	15
4.3 Paralelização hibrida.....	17
5 CONCLUSÃO.....	19
REFERÊNCIAS BIBLIOGRÁFICAS	20
APÊNDICE A – PERFIL DE EXECUÇÃO DA PRIMEIRA RODADA	21
APÊNDICE B – PERFIL DE EXECUÇÃO DA SEGUNDA RODADA.....	28

1 INTRODUÇÃO

Este documento tem por objetivo inicial analisar o funcionamento de um software escrito na linguagem de programação FORTRAN 77. A seguir, são mostradas as técnicas de paralelização aplicadas ao código, que reduzem o tempo de processamento requerido para sua execução. Essas técnicas são usadas em um cluster de processadores disponível no Centro Regional Sul de Pesquisas Espaciais (CRS/INPE). O software utilizado, denominado Sheffield University Plasmasphere-ionosphere Model (SUPIM), resolve numericamente um sistema de equações da continuidade, movimento e do equilíbrio de energia para os principais íons constituintes da ionosfera, fornecendo uma previsão para seu comportamento dinâmico no tempo.

2 VISÃO GERAL DO SUPIM

O SUPIM [1-2] vem sendo desenvolvido há mais de 20 anos pelo professor Graham John Bailey, da Universidade de Sheffield, Reino Unido, e colaboradores. SUPIM é um modelo físico que descreve a distribuição de ionização que ocorre ao longo do tempo na ionosfera terrestre, entre altitudes de aproximadamente 130Km ao longo de linhas fechadas de campo magnético [3]. O modelo fornece como saída as densidades, velocidades alinhadas ao campo magnético e temperaturas para os elétrons e para os íons O^+ , H^+ , He^+ , N_2^+ , O_2^+ e NO^+ . Esses dados são fornecidos em função da latitude e altura para um intervalo de tempo determinado, a partir de uma data e hora pré-estabelecidas. Vários processos físicos e químicos são considerados no modelo, como:

- produção iônica devida à radiação solar;
- produção e perdas iônicas em razão de reações químicas entre ions e gases neutros;
- difusão térmica e ambipolar;
- colisões entre ions e entre ions e elementos neutros;
- ventos termosféricos;
- deriva eletrodinâmica vertical;
- condução térmica;
- aquecimento fotoelétrico;
- aquecimento em razão da fricção.

3 ANÁLISE DO PERFIL DE EXECUÇÃO DO CÓDIGO SUPIM

Neste capítulo são descritos os procedimentos realizados com o código original do programa SUPIM a fim de analisar seu funcionamento e identificar a melhor estratégia de paralelização a ser adotada.

3.1 Temporização

Os tempos de execução e as análises de chamadas para sub-rotinas foram obtidos com base em uma execução-padrão convergente do programa, que apresenta resultados conhecidos e fisicamente aceitáveis. Os núcleos de processamento usados consistem em processadores Dual-Core AMD Opteron rodando a frequência de 1 GHz, sistema operacional SUSE Linux, disponível no cluster junto ao Centro Regional Sul de Pesquisas Espaciais (CRS/INPE).

O código SUPIM é compilado e executado duas vezes, onde cada compilação define um nome diferente dentro do código para o arquivo de entrada. Assim, os arquivos executáveis gerados são invocados de forma consecutiva para produzir a resposta final. Cada execução utiliza um arquivo de entrada diferente, que contém os parâmetros que são utilizados na rodada. O resultado da execução da primeira rodada é mantido em arquivo textual para utilização na rodada seguinte. A tabela 1 mostra os tempos de execução das rodadas e seus percentuais relativos ao tempo total, quando utilizado um único núcleo de processamento. Os tempos mostrados na tabela 1 foram obtidos a partir da execução do código original utilizando a instrução “time” para marcação do tempo de execução. O código foi compilado usando-se o compilador Fortran da Intel para 4 tipos de otimizações automáticas: nível 0 (opção -O0), nível 1 (opção -O1), nível 2 (opção -O2) e nível 3 (opção -O3). As otimizações mais agressivas de níveis 2 e 3 resultaram em saídas absurdas e não foram consideradas. A saída para a otimização automática nível 1 também apresentou pequenas diferenças quando comparada com a saída sem otimização (nível 0). No entanto, essas diferenças limitaram-se a erros de arredondamento de valores, persistindo a consistência dos dados de saída.

Tabela 1: Tempos de execução do código original variando otimizações automáticas

Otimizações automáticas	1ª Rodada	2ª Rodada	Total
Nível 0 (desabilitadas)	1110,9 s (54,2 %)	938,3 s (45,8 %)	2049,2 s
Nível 1 (máxima velocidade)	523,06 s (55,0 %)	427,72 s (45,0 %)	950,78 s

Para entender melhor a distribuição da carga de processamento dentro das sub-rotinas e funções do código, foram gerados perfis de execução para cada rodada. Esses perfis são mostrados nos anexos A e B, referentes à primeira e segunda rodadas, respectivamente.

Na primeira e segunda rodadas, o “flat profile” indica que as sub-rotinas críticas são basicamente as mesmas. As sub-rotinas críticas, que durante essas rodadas consomem ao menos 1% do tempo total de processamento, são mostradas na tabela 2.

As sub-rotinas identificadas na tabela 2 são sem dúvida as mais relevantes para implementação do paralelismo, já que ocupam a maior parte do tempo total de processamento. Contudo, essa carga de processamento pode ser devido à demora para execução das instruções da própria sub-rotina, ou devido ao número elevado de invocações à sub-rotina. No primeiro caso, a estratégia de paralelização deve focar no código da sub-rotina em si. Mas se o tempo gasto com a sub-rotina for devido à grande quantidade de chamadas, a estratégia de paralelização mais adequada deve tratar as rotinas chamadoras. A tabela 3 foi construída baseada nos perfis de execução dos anexos A e B. Ela mostra, para cada uma das sub-rotinas identificadas pela tabela 2, a soma do tempo gasto durante as rodadas, assim como o número de vezes que a

sub-rotina é invocada e o tempo médio de execução de cada chamada. Claramente as sub-rotinas que apresentam os maiores tempos médios por chamada devem ter seu código interno de execução otimizado. Já as sub-rotinas que apresentam o tempo médio por chamada muito pequeno deverão ter a otimização concentrada nas rotinas chamadoras.

3.2 Execução em duas rodadas

O código SUPIM é executado duas vezes consecutivas. O código que é executado é exatamente o mesmo em cada uma das duas rodadas, variando-se unicamente o nome do arquivo de entrada (“Input”) a ser usado. Neste arquivo de entrada, o valor indicado em “JDAY” identifica qual rodada será executada, e esse parâmetro, lido para uma variável de mesmo nome, é usado em desvios condicionais para executar uma ou outra seqüência de código. A passagem de informações entre as rodadas é feita exclusivamente através de um arquivo textual, cujo nome é identificado como “FILE3” no arquivo de entrada (“Input”). Os demais arquivos gerados pelo SUPIM nunca são lidos durante sua execução. O código utiliza extensivamente variáveis globais para troca de dados entre sub-rotinas e o programa principal. Além disso, ele está estruturado de forma pouco amigável, com o uso de instruções de desvio de execução “goto”. A figura 1 ilustra de forma simplificada o funcionamento do código SUPIM, que originalmente possui o total de 8466 linhas com códigos e comentários. Foram omitidas as implementações e chamadas para sub-rotinas, além de trechos de código irrelevantes para este estudo, que foram substituídos por “...”. Nessa figura foram salientadas as partes que são executadas condicionalmente, de acordo com a rodada: primeira (JDAY igual a 0) ou segunda (JDAY igual a 2). Essas partes são testadas em função da variável “JDAY”. Também foram evidenciados na figura 1 os principais laços de execução, os desvios “goto”, assim como leituras e escritas no arquivo FILE3. A análise da figura 1 permite-nos verificar que os dados são gerados de forma iterativa. No caso da convergência não ser verificada, a instrução “goto 1000”

faz com que os cálculos da próxima iteração sejam iniciados. Para a primeira rodada, os dados finais (a serem lidos pela rodada seguinte) somente são gravados em FILE3 quando satisfeitos os critérios de convergência e imediatamente antes do término da rodada, que se dá pela instrução STOP 'NORMAL END 3'. A segunda rodada realiza o carregamento das informações de FILE3 já no laço que termina na linha identificada pelo número 100, antes do início dos cálculos principais (linha identificada pelo número 1000). Essa análise de geração de resultados iterativa e passagem de dados entre rodadas praticamente inviabiliza o paralelismo entre rodadas distintas, já que a segunda rodada só poderá iniciar após a convergência e conclusão da rodada anterior.


```

PROGRAM SUPIM
...
{
  IF(JDAY.GT.1) THEN
  ...
  ELSE
  ...
  ENDF
}
...
{
  IF(JDAY.EQ.2) THEN
  ...
  ENDF
}
DO 100 IFL=1,NFL
{
  IF(JDAY.GT.0) THEN
  1 leitura FILE3, format 901
  ...
  1 leitura FILE3, format 902
  ...
  ELSE
  ...
  ENDF
}
...
{
  IF(JDAY.GT.0) THEN
  (2+3xNIONS) leituras FILE3, format 903
  ...
  GOTO 101
  ENDF
}
... BLOCO INITIAL PLASMA ...
101 ...
100 CONTINUE
1000 ...
{
  IF(JDAY.EQ.2) THEN
  ...
  ENDF
}
{
  IF(JDAY.EQ.2.AND.(xxx)) THEN
  ...
  ENDF
}
DO 410 IFL=1,NFL
...
IF(xxx) GOTO 410
...
410 CONTINUE
IF(xxx) GOTO 600
DO 700 IFL=1,NFL
...
IF(xxx) THEN
...
DO 721 I=INEB,ISEB
DO 722 II=IINEB,IISEB
{
  IF(xxx) THEN
  ...
  GOTO 721
  ELSEIF(xxx) THEN
  ...
  GOTO 721
  ENDF
}
722 CONTINUE
STOP 'UNABLE .. POINT 2'
721 CONTINUE
...
DO 731 I=INEB,ISEB
DO 732 II=IINEB,IISEB
{
  IF(xxx) THEN
  ...
  GOTO 731
  ELSEIF(xxx) THEN
  ...
  GOTO 731
  ENDF
}
732 CONTINUE
STOP 'UNABLE .. POINT 1'
731 CONTINUE
...
ELSE
...
ENDF
700 CONTINUE
...

```

```

600 IF(JDAY.EQ.2) THEN
...
ENDF
IF(JDAY.LT.2) THEN
IF(xxx) THEN
...
DO 616 IFL=1,NFL
{
  IF(xxx) THEN
  ...
  GOTO 617
  ENDF
}
616 CONTINUE
617 IF(xxx) THEN
DO 618 IFL=1,NFL
...
1 escrita FILE3, format 901
1 escrita FILE3, format 902
(2+3xNIONS) escritas FILE3, format 903
...
CONTINUE
...
STOP 'NORMAL END 3'
ELSE
GOTO 1000
ENDF
ELSE
GOTO 1000
ENDF
ELSE
IF(xxx) THEN
...
STOP 'NORMAL END 4'
ELSEIF(xxx) THEN
...
GOTO 1000
ELSE
GOTO 1000
ENDF
ENDF
END

```

Figura 1: Simplificação do código SUPIM

Tabela 2: Sub-rotinas críticas

Sub-rotina	Tempo (%) 1ª Rodada	Tempo (%) 2ª Rodada
temp	23,04%	20,22%
ionhr	13,01%	11,43%
globe5	7,60%	7,09%
differ	4,89%	4,37%
main	0,47%	8,69%
elhr	4,30%	3,77%
denss	3,71%	3,71%
globw4	2,97%	2,91%
concen	2,58%	2,20%
tridi	2,14%	1,90%
glbw4e	1,93%	1,77%
thd	1,74%	1,53%
vsgws4	1,69%	1,50%
lpgws4	1,26%	1,19%
prod	1,33%	1,10%

Tabela 3: Sub-rotinas críticas ordenadas pelo tempo médio consumido em cada chamada

Sub-rotina	Tempo total gasto (s)	Número de chamadas	Tempo médio por chamada (μ s)
main	71,97	2	35985000,00
temp	371,07	67632	5486,60
prod	20,93	67958	307,98
concen	41,05	405792	101,16
ionhr	209,72	3043440	68,91
elhr	69,24	1014480	68,25
thd	28,11	608688	46,18
globe5	125,72	7036336	17,87
differ	79,31	6285388	12,62
tridi	34,69	4260816	8,14
denss	63,33	18470382	3,43
glbw4e	31,69	23163447	1,37
vsgws4	27,36	23163447	1,18
lpgws4	20,97	23163447	0,91
globw4	50,22	62068319	0,81

4 ATIVIDADES DE PARALELIZAÇÃO

Após a análise do perfil de execução do código SUPIM, é possível avaliar as estratégias de paralelização possíveis de serem adotadas. Inicialmente concentramos nossa atenção às sub-rotinas críticas do código, utilizando paralelismo através de uma arquitetura de memória compartilhada, através da OpenMP Application Programming Interface (API) [4] e avaliando os resultados. A seguir, o código foi adaptado para fazer uso de paralelismo usando uma arquitetura de memória distribuída, através do Message Passing Interface (MPI) [5]. As atividades de paralelização resultaram em um código hibridamente paralelizado, com técnicas de memória compartilhada e distribuída, que fornece maior eficiência em sua execução.

4.1 Paralelização usando OpenMP

O código usado nas sub-rotinas críticas foi analisado, visando sua paralelização através de técnicas de memória compartilhada. Assim, diretivas OpenMP foram utilizadas para distribuir o processamento em núcleos diferentes do mesmo nó. A figura 2 ilustra a aplicação de diretivas OpenMP em um laço da sub-rotina “differ”.

```
!$omp parallel do private(W1,W2,W3) if (IS1-IN1.gt.300)
  DO 10 I=IN1,IS1
    W1=DX(I,IFL)*DX(I,IFL)
    W2=DX(I-1,IFL)*DX(I-1,IFL)
    W3=DX(I,IFL)*DX(I-1,IFL)*(DX(I,IFL)+DX(I-1,IFL))
    DF(I)=XETA(I,IFL)*(W1*F(I-1)+(W2-W1)*F(I)-W2*F(I+1))/W3
  10 CONTINUE
!$omp end parallel do
```

Figura 2: Aplicação de diretivas OpenMP para otimização de um laço de execução

Além das diretivas, alguns dos laços de repetição também puderam ser agrupados, reduzindo sobrecarga de processamento desnecessária. A tabela 4 mostra os tempos de execução do código para cada rodada, considerando a paralelização conjunta das sub-rotinas críticas através de OpenMP, para diferentes quantidades de threads utilizadas. Como esperado, a configuração

que utiliza 4 threads para executar o paralelismo através de diretivas OpenMP mostrou o melhor ganho de desempenho – Speedup igual a 1,32, uma vez que cada processador dispõe de 4 núcleos de processamento. As otimizações para máxima velocidade (opção -O1) foram usadas durante a compilação do código.

Tabela 4: Tempos de execução do código com sub-rotinas críticas paralelizadas através de OpenMP e junção de laços comuns

Número de Threads OpenMP	1 ^a Rodada	2 ^a Rodada	Total	Speedup	Eficiência
1	534,46 s	437,69 s	972,15 s	0,98	0,245
2	445,89 s	373,52 s	819,41 s	1,16	0,290
3	394,28 s	336,39 s	730,67 s	1,30	0,325
4	388,25 s	332,71 s	720,96 s	1,32	0,330
5	507,79 s	430,23 s	938,02 s	1,01	0,253
6	520,50 s	433,87 s	954,37 s	1,00	0,250
7	526,88 s	442,63 s	969,51 s	0,98	0,245

4.2 Paralelização usando MPI

Para que possamos utilizar os recursos de paralelismo de memória distribuída definidos pela Interface de Passagem de Mensagens (MPI) no SUPIM, inicialmente fizemos com que as tarefas de leitura e escrita de arquivos em disco fossem executadas de forma concorrente entre os processos. Para isso, foi utilizado no cluster o sistema de arquivos NFS (Network File System).

Assim, vários processos puderam ser disparados em nós diferentes do cluster, chegando todos aos mesmos resultados. Com isso foi possível implementar a paralelização do código e troca de informações entre nós. No entanto, o ganho pela paralelização aplicada em laços internos do código não compensou a sobrecarga de processamento intrínseca ao MPI.

O SUPIM realiza a simulação para uma longitude magnética fixa, variando-se a latitude e altitude das linhas de campo magnético consideradas. A abordagem utilizada para aproveitar o poder de processamento dos vários nós do cluster através do MPI basicamente consistiu em dividir a região de análise em longitudes magnéticas diferentes, e rodar o SUPIM para cada longitude em um nó diferente. A figura 3 ilustra os locais simulados em duas execuções do código para longitudes magnéticas distintas. A execução do código em cada longitude é independente das demais execuções, e pode ser realizada paralelamente.

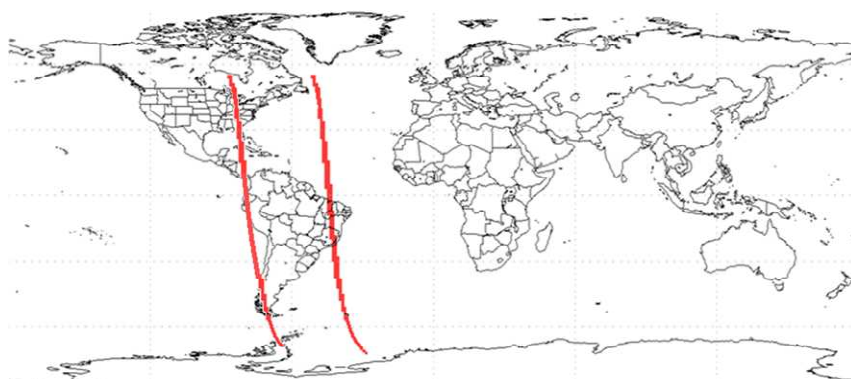


Figura 3: Região abrangida por duas simulações do código SUPIM

Assim, o conjunto de resultados obtidos em execuções para várias longitudes magnéticas diferentes mostra a simulação da dinâmica da ionosfera para uma região inteira, tendo sido computados os resultados de cada longitude individualmente em nós distintos do cluster. Essa abordagem possibilitou a redução do tempo de processamento total proporcionalmente ao número de nós utilizados, limitando o número de nós utilizados ao número de execuções distintas do código.

4.3 Paralelização híbrida

A forma mais eficiente para paralelização do código SUPIM consistiu na utilização simultânea de técnicas de paralelização de memória compartilhada usando OpenMP e de memória distribuída, usando MPI. A paralelização com OpenMP foi extensivamente utilizada nas estruturas existentes nas sub-rotinas críticas do código. Já o MPI foi utilizado para facilitar a execução simultânea, em nós diferentes, de simulações em localidades geográficas distintas. A tabela 5 ilustra os tempos de execução obtidos com esta abordagem, para a execução de 3 simulações independentes, utilizando 1, 2 ou 3 nós de processamento. O número de threads OpenMP utilizado foi fixado em 4. Note-se que, utilizando a metodologia descrita, o acréscimo de mais nós de processamento não reduzirá o tempo total de execução, já que cada simulação distinta é executada em apenas 1 nó. Para o exemplo ilustrado na tabela 5, a utilização de 3 nós de processamento forneceu speedup igual a 3,96, enquanto a eficiência ficou em 0,330.

Tabela 5: Tempos de execução do código para 3 simulações distintas, usando MPI e OpenMP

Número de nós de processamento utilizados	Tempo Total de execução	Speedup	Eficiência
1	2162,88 s	1,32	0,330
2	1441,92 s	1,98	0,247
3	720,96 s	3,96	0,330

5 CONCLUSÃO

O objetivo principal deste documento foi analisar o código SUPIM, para previsão da dinâmica da ionosfera e documentar a utilização de técnicas de paralelismo para reduzir o tempo total de processamento requerido. Foi verificado inicialmente que a utilização de algumas otimizações automáticas na compilação do código original resultou em erro nos arquivos de saída. Além disso, verificou-se que o paralelismo entre rodadas de execução diferentes não é possível de ser implementado, assim como, o paralelismo em laços iterativos, muito presentes no código. Para as principais sub-rotinas, a dependência dos dados e uso de variáveis globais dificulta muito o trabalho de paralelização de maior granularidade. O perfil de execução do código também foi analisado, e as sub-rotinas críticas foram identificadas. Nas atividades de paralelização, foram utilizadas duas abordagens: paralelização através de técnicas de memória compartilhada (usando OpenMP) e memória distribuída (usando MPI). A aplicação das técnicas de paralelismo com OpenMP resultou em speedup igual a 1,32, quando utilizadas 4 threads em um nó. A MPI foi utilizada para facilitar a distribuição da execução de várias simulações entre os nós de processamento disponíveis. Foram implementadas ambas técnicas de paralelização no código. Para um caso hipotético de simulação para 3 localidades diferentes, essa abordagem híbrida reduziu o tempo de execução necessário de 2852,34 segundos (SUPIM serial) para 720,96 segundos (SUPIM paralelo), quando usados 3 nós de execução rodando 4 threads cada um. Esse ganho equivale a um speedup igual a 3,96 e eficiência 0,330.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BAILEY, G. **Informações sobre o modelo SUPIM**. Disponível em <<http://gbailey.staff.shef.ac.uk/supim.html>>. Acesso em abril de 2010.
- [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, São José dos Campos. 1997.
- [3] SANTOS, P. M. T. **Estudo da dinâmica do sistema termosfera - ionosfera por modelo teórico e observações experimentais**. 2005. 134 p. (INPE-13270-TDI/1032). Tese (Doutorado em Geofísica Espacial) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos. 2005. Disponível em: <<http://urlib.net/sid.inpe.br/iris@1913/2005/04.28.13.11>>. Acesso em: 20 jul. 2010.
- [4] CHANDRA, R.; DAGUM, L.; KOHR, D.; MAYDAN, D.; McDONALD, J.; MENON, R. **Parallel programming in OpenMP**. San Diego: Academic Press, 2001.
- [5] SNIR, M. **MPI - the complete reference MPI-2**. Massachusetts: MIT Press 1998.

APÊNDICE A – PERFIL DE EXECUÇÃO DA PRIMEIRA RODADA

Flat profile:

Each sample counts as 0.01 seconds.

time	% cumulative	seconds	self	calls	self	total	name
		seconds	seconds	s/call	s/call	s/call	
23.04	213.79	213.79		36336	0.01	0.01	temp_
13.01	334.57	120.77	1635120	0.00	0.00	0.00	ionhr_
7.60	405.10	70.53	3780248	0.00	0.00	0.00	globe5_
5.66	457.66	52.56					exp.A
4.89	503.01	45.35	3343238	0.00	0.00	0.00	differ_
4.53	545.03	42.02					cos.A
4.46	586.38	41.35					sqrt
4.30	626.28	39.90	545040	0.00	0.00	0.00	elhr_
3.71	660.73	34.45	9923151	0.00	0.00	0.00	denss_
2.97	688.32	27.59	34883040	0.00	0.00	0.00	globw4_
2.58	712.22	23.90	218016	0.00	0.00	0.00	concen_
2.38	734.29	22.07					pow.A
2.14	754.19	19.90	2289168	0.00	0.00	0.00	tridi_
1.93	772.13	17.94	13243440	0.00	0.00	0.00	glbw4e_
1.74	788.32	16.19	327024	0.00	0.00	0.00	thd_
1.69	803.99	15.67	13243440	0.00	0.00	0.00	vsgws4_
1.59	818.77	14.78					sin.A
1.57	833.31	14.54					__powr8i4
1.33	845.67	12.36	36499	0.00	0.00	0.00	prod_
1.26	857.35	11.68	13243440	0.00	0.00	0.00	lpgws4_
0.98	866.45	9.10	13294839	0.00	0.00	0.00	optic_
0.88	874.58	8.13	2362655	0.00	0.00	0.00	glob5l_
0.73	881.37	6.79	109008	0.00	0.00	0.00	thdiff_
0.52	886.20	4.83					sin
0.47	890.58	4.38	1	4.38	719.17	719.17	MAIN__
0.37	894.03	3.45	36499	0.00	0.00	0.00	nprod_
0.35	897.27	3.24					log.A
0.35	900.49	3.22	472531	0.00	0.00	0.00	gts5_
0.30	903.26	2.77	655026	0.00	0.00	0.00	reac_
0.26	905.70	2.44	472531	0.00	0.00	0.00	atmos_
0.23	907.88	2.18					log10.A
0.23	910.01	2.13					cos
0.20	911.90	1.89	4252779	0.00	0.00	0.00	ccor_
0.19	913.62	1.72	36499	0.00	0.00	0.00	qphe_
0.17	915.23	1.61					exp.L
0.16	916.68	1.45	3307717	0.00	0.00	0.00	dnet_
0.14	918.02	1.34					cosh
0.13	919.19	1.17					cos.L
0.12	920.34	1.15	13243440	0.00	0.00	0.00	gws4_
0.11	921.32	0.98	26486880	0.00	0.00	0.00	wprof4_
0.10	922.21	0.89					exp
0.09	923.05	0.84					acos
0.08	923.76	0.71	3358176	0.00	0.00	0.00	splinw_
0.07	924.40	0.64	36336	0.00	0.00	0.00	wind_
0.07	925.04	0.64					__powi4i4
0.06	925.61	0.57					sin.L
0.05	926.08	0.47					sinh
0.02	926.29	0.21	3358176	0.00	0.00	0.00	splint_
0.02	926.47	0.18					cvtas_t_to_a
0.02	926.64	0.17					pow.L
0.02	926.79	0.15					matherr
0.02	926.94	0.15					matherrl
0.01	927.06	0.12					for_write_seq_fmt_xmit
0.01	927.18	0.12					for__interp_fmt
0.01	927.28	0.10					log10
0.01	927.36	0.08					cvt_ieee_t_to_text_ex
0.01	927.44	0.08					pow
0.01	927.50	0.06					for__format_value
0.01	927.55	0.05	51399	0.00	0.00	0.00	mg2geo_
0.00	927.59	0.04	163	0.00	0.00	0.00	coord_
0.00	927.63	0.04					for__acquire_lun
0.00	927.67	0.04					log.L
0.00	927.70	0.03					for__desc_ret_item
0.00	927.72	0.02	163	0.00	0.00	0.00	mdirec_
0.00	927.74	0.02					_intel_fast_memset
0.00	927.76	0.02					cvtas_string_to_nan_x
0.00	927.78	0.02					llround
0.00	927.80	0.02					log10.L
0.00	927.82	0.02					prmsg5_
0.00	927.83	0.01	163	0.00	0.00	0.00	mgfldl_
0.00	927.84	0.01					__libm_setusermatherrl
0.00	927.85	0.01					_intel_fast_memcpy.A

0.00	927.86	0.01				_intel_fast_memcpy.J
0.00	927.87	0.01				asinh.A
0.00	927.88	0.01				atan2
0.00	927.89	0.01				cvt_ieee_t_to_text
0.00	927.90	0.01				cvtas_s_to_a
0.00	927.91	0.01				euvacd_
0.00	927.92	0.01				for_put_sf
0.00	927.93	0.01				for_write_output
0.00	927.94	0.01				for_write_seq_fmt
0.00	927.95	0.01				for_write_seq_lis
0.00	927.96	0.01				g_
0.00	927.97	0.01				log
0.00	927.98	0.01				meters_
0.00	927.99	0.01				sinh.A
0.00	927.99	0.00	59426	0.00	0.00	nraphs_
0.00	927.99	0.00	36499	0.00	0.00	writ04_
0.00	927.99	0.00	978	0.00	0.00	magreb_
0.00	927.99	0.00	164	0.00	0.00	magreq_
0.00	927.99	0.00	1	0.00	0.00	dipole_
0.00	927.99	0.00	1	0.00	0.00	euvac_
0.00	927.99	0.00	1	0.00	0.00	geo2mg_
0.00	927.99	0.00	1	0.00	0.00	igrf_
0.00	927.99	0.00	1	0.00	0.00	tsgws4_

Call graph:

granularity: each sample hit covers 2 byte(s) for 0.00% of 927.99 seconds

index	% time	self	children	called	name
[1]	77.5	4.38	714.79	1/1	main [2]
		4.38	714.79	1	MAIN__ [1]
		213.79	212.88	36336/36336	temp_ [3]
		23.90	145.84	218016/218016	concen_ [4]
		0.64	75.93	36336/36336	wind_ [8]
		12.36	12.55	36499/36499	prod_ [20]
		0.19	9.24	36499/472531	atmos_ [5]
		6.41	0.00	472368/3343238	differ_ [12]
		0.92	0.00	218994/655026	reac_ [33]
		0.05	0.00	51399/51399	mg2geo_ [60]
		0.04	0.00	163/163	coord_ [61]
		0.02	0.00	163/163	mdirec_ [65]
		0.01	0.00	163/163	mgfldl_ [71]
		0.01	0.00	163/36499	qphe_ [38]
		0.00	0.00	36499/36499	writ04_ [89]
		0.00	0.00	978/978	magreb_ [90]
		0.00	0.00	163/164	magreq_ [91]
		0.00	0.00	1/1	dipole_ [92]
		0.00	0.00	1/1	geo2mg_ [94]
					<spontaneous>
[2]	77.5	0.00	719.17		main [2]
		4.38	714.79	1/1	MAIN__ [1]
		213.79	212.88	36336/36336	MAIN__ [1]
[3]	46.0	213.79	212.88	36336	temp_ [3]
		120.77	0.00	1635120/1635120	ionhr_ [6]
		39.90	0.00	545040/545040	elhr_ [16]
		31.55	0.00	2325504/3343238	differ_ [12]
		18.95	0.00	2180160/2289168	tridi_ [23]
		1.71	0.00	36336/36499	qphe_ [38]
		23.90	145.84	218016/218016	MAIN__ [1]
[4]	18.3	23.90	145.84	218016	concen_ [4]
		2.25	110.43	436032/472531	atmos_ [5]
		6.79	16.19	109008/109008	thdiff_ [21]
		7.39	0.00	545040/3343238	differ_ [12]
		1.84	0.00	436032/655026	reac_ [33]
		0.95	0.00	109008/2289168	tridi_ [23]
		0.19	9.24	36499/472531	MAIN__ [1]
		2.25	110.43	436032/472531	concen_ [4]
[5]	13.2	2.44	119.67	472531	atmos_ [5]
		3.22	116.45	472531/472531	gts5_ [7]
		120.77	0.00	1635120/1635120	temp_ [3]
[6]	13.0	120.77	0.00	1635120	ionhr_ [6]
		3.22	116.45	472531/472531	atmos_ [5]
[7]	12.9	3.22	116.45	472531	gts5_ [7]

```

70.53 0.00 3780248/3780248 globe5_ [10]
34.45 0.00 9923151/9923151 denss_ [17]
8.13 0.00 2362655/2362655 glob5l_ [29]
1.89 0.00 4252779/4252779 ccor_ [37]
1.45 0.00 3307717/3307717 dnet_ [40]
-----
[8] 8.3 0.64 75.93 36336/36336 MAIN_ [1]
0.64 75.93 36336 wind_ [8]
1.15 74.78 13243440/13243440 gws4_ [9]
-----
[9] 8.2 1.15 74.78 13243440/13243440 wind_ [8]
1.15 74.78 13243440 gws4_ [9]
17.94 27.35 13243440/13243440 glbw4e_ [13]
27.59 0.00 34883040/34883040 globw4_ [18]
0.98 0.92 26486880/26486880 wprof4_ [36]
-----
[10] 7.6 70.53 0.00 3780248/3780248 gts5_ [7]
70.53 0.00 3780248 globe5_ [10]
-----
[11] 5.7 52.56 0.00 <spontaneous>
exp.A [11]
-----
0.00 0.00 326/3343238 coord_ [61]
6.41 0.00 472368/3343238 MAIN_ [1]
7.39 0.00 545040/3343238 concen_ [4]
[12] 4.9 31.55 0.00 2325504/3343238 temp_ [3]
45.35 0.00 3343238 differ_ [12]
-----
[13] 4.9 17.94 27.35 13243440/13243440 gws4_ [9]
17.94 27.35 13243440 glbw4e_ [13]
15.67 11.68 13243440/13243440 vsgws4_ [19]
0.00 0.00 1/1 tsgws4_ [96]
-----
[14] 4.5 42.02 0.00 <spontaneous>
cos.A [14]
-----
[15] 4.5 41.35 0.00 <spontaneous>
sqrt [15]
-----
[16] 4.3 39.90 0.00 545040/545040 temp_ [3]
39.90 0.00 545040 elhr_ [16]
-----
[17] 3.7 34.45 0.00 9923151/9923151 gts5_ [7]
34.45 0.00 9923151 denss_ [17]
-----
[18] 3.0 27.59 0.00 34883040/34883040 gws4_ [9]
27.59 0.00 34883040 globw4_ [18]
-----
[19] 2.9 15.67 11.68 13243440/13243440 glbw4e_ [13]
15.67 11.68 13243440 vsgws4_ [19]
11.68 0.00 13243440/13243440 lpgws4_ [27]
-----
[20] 2.7 12.36 12.55 36499/36499 MAIN_ [1]
12.36 12.55 36499 prod_ [20]
9.10 0.00 13294839/13294839 optic_ [28]
3.45 0.00 36499/36499 nprod_ [31]
0.00 0.00 1/1 euvac_ [93]
-----
[21] 2.5 6.79 16.19 109008/109008 concen_ [4]
6.79 16.19 109008 thdiff_ [21]
16.19 0.00 327024/327024 thd_ [24]
-----
[22] 2.4 22.07 0.00 <spontaneous>
pow.A [22]
-----
0.95 0.00 109008/2289168 concen_ [4]
18.95 0.00 2180160/2289168 temp_ [3]
[23] 2.1 19.90 0.00 2289168 tridi_ [23]
-----
[24] 1.7 16.19 0.00 327024/327024 thdiff_ [21]
16.19 0.00 327024 thd_ [24]
-----
[25] 1.6 14.78 0.00 <spontaneous>
sin.A [25]
-----
[26] 1.6 14.54 0.00 <spontaneous>
__powr8i4 [26]
-----
[27] 1.3 11.68 0.00 13243440/13243440 vsgws4_ [19]
11.68 0.00 13243440 lpgws4_ [27]
-----

```

[28]	1.0	9.10	0.00	13294839/13294839	prod_ [20]
		9.10	0.00	13294839	optic_ [28]
[29]	0.9	8.13	0.00	2362655/2362655	gts5_ [7]
		8.13	0.00	2362655	glob51_ [29]
[30]	0.5	4.83	0.00		<spontaneous> sin [30]
[31]	0.4	3.45	0.00	36499/36499	prod_ [20]
		3.45	0.00	36499	nprod_ [31]
[32]	0.3	3.24	0.00		<spontaneous> log.A [32]
[33]	0.3	0.92	0.00	218994/655026	MAIN_ [1]
		1.84	0.00	436032/655026	concen_ [4]
		2.77	0.00	655026	reac_ [33]
[34]	0.2	2.18	0.00		<spontaneous> log10.A [34]
[35]	0.2	2.13	0.00		<spontaneous> cos [35]
[36]	0.2	0.98	0.92	26486880/26486880	gws4_ [9]
		0.98	0.92	26486880	wprof4_ [36]
		0.71	0.00	3358176/3358176	splinw_ [45]
		0.21	0.00	3358176/3358176	splint_ [49]
[37]	0.2	1.89	0.00	4252779/4252779	gts5_ [7]
		1.89	0.00	4252779	ccor_ [37]
[38]	0.2	0.01	0.00	163/36499	MAIN_ [1]
		1.71	0.00	36336/36499	temp_ [3]
		1.72	0.00	36499	qphe_ [38]
[39]	0.2	1.61	0.00		<spontaneous> exp.L [39]
[40]	0.2	1.45	0.00	3307717/3307717	gts5_ [7]
		1.45	0.00	3307717	dnet_ [40]
[41]	0.1	1.34	0.00		<spontaneous> cosh [41]
[42]	0.1	1.17	0.00		<spontaneous> cos.L [42]
[43]	0.1	0.89	0.00		<spontaneous> exp [43]
[44]	0.1	0.84	0.00		<spontaneous> acos [44]
[45]	0.1	0.71	0.00	3358176/3358176	wprof4_ [36]
		0.71	0.00	3358176	splinw_ [45]
[46]	0.1	0.64	0.00		<spontaneous> __powi4i4 [46]
[47]	0.1	0.57	0.00		<spontaneous> sin.L [47]
[48]	0.1	0.47	0.00		<spontaneous> sinh [48]
[49]	0.0	0.21	0.00	3358176/3358176	wprof4_ [36]
		0.21	0.00	3358176	splint_ [49]
[50]	0.0	0.18	0.00		<spontaneous> cvtas_t_to_a [50]
[51]	0.0	0.17	0.00		<spontaneous> pow.L [51]
[52]	0.0	0.15	0.00		<spontaneous> matherr [52]
[53]	0.0	0.15	0.00		<spontaneous> matherrl [53]

[80]	0.0	0.01	0.00		<spontaneous> for_write_seq_lis [80]
[81]	0.0	0.01	0.00		<spontaneous> g_ [81]
[82]	0.0	0.01	0.00		<spontaneous> log [82]
[83]	0.0	0.01	0.00		<spontaneous> meters_ [83]
[84]	0.0	0.01	0.00		<spontaneous> sinh.A [84]
[85]	0.0	0.01	0.00		<spontaneous> __libm_setusermatherrl [85]
[86]	0.0	0.01	0.00		<spontaneous> _intel_fast_memcpy.A [86]
[87]	0.0	0.01	0.00		<spontaneous> _intel_fast_memcpy.J [87]
[88]	0.0	0.00	0.00	59426/59426	mgfldl_ [71]
[89]	0.0	0.00	0.00	36499/36499	nraps_ [88]
[90]	0.0	0.00	0.00	978/978	MAIN__ [1]
[91]	0.0	0.00	0.00	978	writ04_ [89]
[92]	0.0	0.00	0.00	1/164	MAIN__ [1]
[93]	0.0	0.00	0.00	163/164	magreb_ [90]
[94]	0.0	0.00	0.00	164	coord_ [61]
[95]	0.0	0.00	0.00	1/1	MAIN__ [1]
[96]	0.0	0.00	0.00	1	magreq_ [91]
[97]	0.0	0.00	0.00	1/1	MAIN__ [1]
[98]	0.0	0.00	0.00	1	dipole_ [92]
[99]	0.0	0.00	0.00	1/1	igrf_ [95]
[100]	0.0	0.00	0.00	1/1	prod_ [20]
[101]	0.0	0.00	0.00	1	euvac_ [93]
[102]	0.0	0.00	0.00	1/1	MAIN__ [1]
[103]	0.0	0.00	0.00	1	geo2mg_ [94]
[104]	0.0	0.00	0.00	1/1	dipole_ [92]
[105]	0.0	0.00	0.00	1	igrf_ [95]
[106]	0.0	0.00	0.00	1/1	glbw4e_ [13]
[107]	0.0	0.00	0.00	1	tsgws4_ [96]

Index by function name

[1] MAIN__	[39] exp.L	[65] mdirec_
[85] __libm_setusermatherrl	[62] for__acquire_lun	[83] meters_
[46] __powi4i4	[64] for__desc_ret_item	[60] mg2geo_
[26] __powr8i4	[59] for__format_value	[71] mgfldl_
[86] _intel_fast_memcpy.A	[55] for__interp_fmt	[31] nprod_
[87] _intel_fast_memcpy.J	[77] for__put_sf	[88] nraps_
[70] _intel_fast_memset	[78] for__write_output	[28] optic_
[44] acos	[79] for_write_seq_fmt	[58] pow
[72] asinh.A	[54] for_write_seq_fmt_xmit	[22] pow.A
[73] atan2	[80] for_write_seq_lis	[51] pow.L
[5] atmos_	[81] g_	[69] prmsg5_
[37] ccor_	[94] geo2mg_	[20] prod_
[4] concen_	[13] glbw4e_	[38] qphe_
[61] coord_	[29] glob51_	[33] reac_
[35] cos	[10] globe5_	[30] sin
[14] cos.A	[18] globw4_	[25] sin.A
[42] cos.L	[7] gts5_	[47] sin.L
[41] cosh	[9] gws4_	[48] sinh
[74] cvt_ieee_t_to_text	[95] igrf_	[84] sinh.A
[57] cvt_ieee_t_to_text_ex	[6] ionhr_	[49] splint_
[75] cvtas_s_to_a	[67] llround	[45] splinw_
[66] cvtas_string_to_nan_x	[82] log	[15] sqrt
[50] cvtas_t_to_a	[32] log.A	[3] temp_
[17] denss_	[63] log.L	[24] thd_
[12] differ_	[56] log10	[21] thdiff_

[92] dipole_
[40] dnet_
[16] elhr_
[93] euvac_
[76] euvacd_
[43] exp
[11] exp.A

[34] log10.A
[68] log10.L
[27] lpgws4_
[90] magreb_
[91] magreq_
[52] matherr
[53] matherr1

[23] tridi_
[96] tsgws4_
[19] vsgws4_
[8] wind_
[36] wprof4_
[89] writ04_

APÊNDICE B – PERFIL DE EXECUÇÃO DA SEGUNDA RODADA

Flat profile:

Each sample counts as 0.01 seconds.

time	% cumulative	seconds	self	calls	self	s/call	total	s/call	name
20.22	157.28	157.28	31296	0.01	0.01	temp_			
11.43	246.23	88.95	1408320	0.00	0.00	ionhr_			
8.69	313.83	67.59	1	67.59	609.59	MAIN_			
7.09	369.02	55.19	3256088	0.00	0.00	globe5_			
5.20	409.50	40.48				exp.A			
4.39	443.62	34.12				cos.A			
4.37	477.58	33.96	2942150	0.00	0.00	differ_			
4.06	509.16	31.58				sqrt			
3.77	538.49	29.34	469440	0.00	0.00	elhr_			
3.71	567.37	28.88	8547231	0.00	0.00	denss_			
2.91	590.01	22.63	27185279	0.00	0.00	globw4_			
2.29	607.82	17.81				pow.A			
2.20	624.97	17.15	187776	0.00	0.00	concen_			
1.90	639.76	14.79	1971648	0.00	0.00	tridi_			
1.77	653.51	13.75	9920007	0.00	0.00	glbw4e_			
1.76	667.21	13.70				sin.A			
1.53	679.13	11.92	281664	0.00	0.00	thd_			
1.50	690.82	11.69	9920007	0.00	0.00	vsgws4_			
1.34	701.27	10.45				__powr8i4			
1.19	710.56	9.29	9920007	0.00	0.00	lpgws4_			
1.10	719.13	8.57	31459	0.00	0.00	prod_			
0.89	726.07	6.94	9920007	0.00	0.00	optic_			
0.85	732.70	6.63	2035055	0.00	0.00	glob5l_			
0.57	737.12	4.42	93888	0.00	0.00	thdiff_			
0.43	740.47	3.35				sin			
0.37	743.32	2.85				log.A			
0.32	745.81	2.49	31459	0.00	0.00	nprod_			
0.28	747.99	2.18	407011	0.00	0.00	gts5_			
0.26	749.99	2.00	407011	0.00	0.00	atmos_			
0.26	751.99	2.00	563328	0.00	0.00	reac_			
0.25	753.95	1.96	31459	0.00	0.00	coord_			
0.24	755.79	1.84				log10.A			
0.23	757.60	1.81				cos			
0.19	759.05	1.45	3663099	0.00	0.00	ccor_			
0.17	760.41	1.36	31459	0.00	0.00	qphe_			
0.17	761.75	1.34	2849077	0.00	0.00	dnet_			
0.17	763.07	1.32				exp.L			
0.14	764.16	1.09				acos			
0.14	765.23	1.07				cosh			
0.14	766.29	1.06	9896738	0.00	0.00	nraphs_			
0.13	767.34	1.05				cos.L			
0.12	768.26	0.92	31459	0.00	0.00	mdirec_			
0.11	769.12	0.86	9920007	0.00	0.00	gws4_			
0.10	769.93	0.81	9920007	0.00	0.00	mg2geo_			
0.10	770.67	0.74				asinh.A			
0.09	771.36	0.69				exp			
0.09	772.03	0.67	2938104	0.00	0.00	splinw_			
0.08	772.67	0.64				atan2			
0.08	773.28	0.61				__powi4i4			
0.07	773.82	0.54	19840014	0.00	0.00	wprof4_			
0.05	774.22	0.40				sinh			
0.05	774.58	0.36	31459	0.00	0.00	wind_			
0.05	774.94	0.36				sin.L			
0.04	775.22	0.28				atan.A			
0.03	775.49	0.27				cvtas_t_to_a			
0.03	775.75	0.26	31296	0.00	0.00	mgfld2_			
0.03	775.97	0.22	2938104	0.00	0.00	splint_			
0.03	776.18	0.21				for_write_seq_fmt_xmit			
0.02	776.37	0.19				cvtas_a_to_t			
0.02	776.50	0.13				for__interp_fmt			
0.02	776.63	0.13				pow			
0.02	776.75	0.12				pow.L			
0.01	776.86	0.11				cvt_ieee_t_to_text_ex			
0.01	776.97	0.11				matherrl			
0.01	777.07	0.10				matherr			
0.01	777.16	0.09				for__format_value			
0.01	777.24	0.08				log10			
0.01	777.30	0.06				for_read_seq_fmt_xmit			
0.01	777.35	0.05				for__close_proc			
0.01	777.40	0.05				for_write_seq_fmt			
0.01	777.45	0.05				log.L			
0.01	777.49	0.04	119808	0.00	0.00	bspl4t_			

0.01	777.53	0.04				for__desc_ret_item
0.01	777.57	0.04				for__put_sf
0.00	777.60	0.03	119808	0.00	0.00	bspl4l_
0.00	777.63	0.03				asinh.L
0.00	777.66	0.03				cvtas_string_to_nan_x
0.00	777.69	0.03				log
0.00	777.72	0.03				sinh.A
0.00	777.74	0.02	31459	0.00	0.00	writ04_
0.00	777.76	0.02	163	0.00	0.00	mgfld1_
0.00	777.78	0.02				cosh.A
0.00	777.80	0.02				for__write_output
0.00	777.81	0.01	15811	0.00	0.00	writ20_
0.00	777.82	0.01	192	0.00	0.00	sfvdm_
0.00	777.83	0.01				_intel_fast_memcpy
0.00	777.84	0.01				_intel_fast_memcpy.J
0.00	777.85	0.01				_intel_fast_memset.J
0.00	777.86	0.01				asinh
0.00	777.87	0.01				atan.L
0.00	777.88	0.01				cvt_ieee_t_to_text
0.00	777.89	0.01				cvt_integer_to_text
0.00	777.90	0.01				cvt_text_to_ieee_t
0.00	777.91	0.01				for__acquire_lun
0.00	777.92	0.01				for__flush_readahead
0.00	777.93	0.01				for__get_s
0.00	777.94	0.01				for__put_d
0.00	777.95	0.01				for__release_lun
0.00	777.96	0.01				llround
0.00	777.97	0.01				log10.L
0.00	777.98	0.01				prmsg5_
0.00	777.98	0.00	31460	0.00	0.00	magreq_
0.00	777.98	0.00	31296	0.00	0.00	exbv_
0.00	777.98	0.00	978	0.00	0.00	magreb_
0.00	777.98	0.00	192	0.00	0.00	g_
0.00	777.98	0.00	1	0.00	0.00	dipole_
0.00	777.98	0.00	1	0.00	0.00	euvac_
0.00	777.98	0.00	1	0.00	0.00	geo2mg_
0.00	777.98	0.00	1	0.00	0.00	igrf_
0.00	777.98	0.00	1	0.00	0.00	tsgws4_

Call graph:

granularity: each sample hit covers 2 byte(s) for 0.00% of 777.98 seconds

index	% time	self	children	called	name
		67.59	541.99	1/1	main [2]
[1]	78.4	67.59	541.99	1	MAIN__ [1]
		157.28	156.85	31296/31296	temp_ [3]
		17.15	113.92	187776/187776	concen_ [4]
		0.36	59.65	31459/31459	wind_ [8]
		8.57	9.43	31459/31459	prod_ [20]
		0.15	7.39	31459/407011	atmos_ [5]
		4.70	0.00	406848/2942150	differ_ [14]
		1.96	0.73	31459/31459	coord_ [32]
		0.26	1.05	31296/31296	mgfld2_ [42]
		0.92	0.00	31459/31459	mdirec_ [47]
		0.81	0.00	9920007/9920007	mg2geo_ [48]
		0.67	0.00	187776/563328	reac_ [34]
		0.00	0.08	31296/31296	exbv_ [68]
		0.02	0.01	163/163	mgfld1_ [83]
		0.02	0.00	31459/31459	writ04_ [84]
		0.01	0.00	15811/15811	writ20_ [87]
		0.01	0.00	163/31459	qphe_ [39]
		0.00	0.00	31459/31460	magreq_ [104]
		0.00	0.00	978/978	magreb_ [105]
		0.00	0.00	1/1	dipole_ [107]
		0.00	0.00	1/1	geo2mg_ [109]

[2]	78.4	0.00	609.59		<spontaneous>
		67.59	541.99	1/1	main [2]
					MAIN__ [1]

		157.28	156.85	31296/31296	MAIN__ [1]
[3]	40.4	157.28	156.85	31296	temp_ [3]
		88.95	0.00	1408320/1408320	ionhr_ [7]
		29.34	0.00	469440/469440	elhr_ [16]
		23.12	0.00	2002944/2942150	differ_ [14]
		14.09	0.00	1877760/1971648	tridi_ [23]
		1.35	0.00	31296/31459	qphe_ [39]

[4]	16.8	17.15 17.15 1.85 4.42 5.42 1.33 0.70	113.92 113.92 88.28 11.92 0.00 0.00 0.00	187776/187776 187776 375552/407011 93888/93888 469440/2942150 375552/563328 93888/1971648	MAIN_ [1] concen_ [4] atmos_ [5] thdiff_ [22] differ_ [14] reac_ [34] tridi_ [23]
[5]	12.6	0.15 1.85 2.00 2.18	7.39 88.28 95.67 93.49	31459/407011 375552/407011 407011 407011/407011	MAIN_ [1] concen_ [4] atmos_ [5] gts5_ [6]
[6]	12.3	2.18 2.18 55.19 28.88 6.63 1.45 1.34	93.49 93.49 0.00 0.00 0.00 0.00 0.00	407011/407011 407011 3256088/3256088 8547231/8547231 2035055/2035055 3663099/3663099 2849077/2849077	atmos_ [5] gts5_ [6] globe5_ [10] denss_ [17] glob5l_ [29] ccor_ [37] dnet_ [40]
[7]	11.4	88.95 88.95	0.00 0.00	1408320/1408320 1408320	temp_ [3] ionhr_ [7]
[8]	7.7	0.36 0.36 0.86	59.65 59.65 58.79	31459/31459 31459 9920007/9920007	MAIN_ [1] wind_ [8] gws4_ [9]
[9]	7.7	0.86 0.86 13.75 22.63 0.54	58.79 58.79 20.98 0.00 0.89	9920007/9920007 9920007 9920007/9920007 27185279/27185279 19840014/19840014	wind_ [8] gws4_ [9] glbw4e_ [12] globw4_ [18] wprof4_ [38]
[10]	7.1	55.19 55.19	0.00 0.00	3256088/3256088 3256088	gts5_ [6] globe5_ [10]
[11]	5.2	40.48	0.00		<spontaneous> exp.A [11]
[12]	4.5	13.75 13.75 11.69 0.00	20.98 20.98 9.29 0.00	9920007/9920007 9920007 9920007/9920007 1/1	gws4_ [9] glbw4e_ [12] vsgws4_ [19] tsgws4_ [111]
[13]	4.4	34.12	0.00		<spontaneous> cos.A [13]
[14]	4.4	0.73 4.70 5.42 23.12 33.96	0.00 0.00 0.00 0.00 0.00	62918/2942150 406848/2942150 469440/2942150 2002944/2942150 2942150	coord_ [32] MAIN_ [1] concen_ [4] temp_ [3] differ_ [14]
[15]	4.1	31.58	0.00		<spontaneous> sqrt [15]
[16]	3.8	29.34 29.34	0.00 0.00	469440/469440 469440	temp_ [3] elhr_ [16]
[17]	3.7	28.88 28.88	0.00 0.00	8547231/8547231 8547231	gts5_ [6] denss_ [17]
[18]	2.9	22.63 22.63	0.00 0.00	27185279/27185279 27185279	gws4_ [9] globw4_ [18]
[19]	2.7	11.69 11.69 9.29	9.29 9.29 0.00	9920007/9920007 9920007 9920007/9920007	glbw4e_ [12] vsgws4_ [19] lpgws4_ [27]
[20]	2.3	8.57 8.57 6.94 2.49 0.00	9.43 9.43 0.00 0.00 0.00	31459/31459 31459 9920007/9920007 31459/31459 1/1	MAIN_ [1] prod_ [20] optic_ [28] nprod_ [33] euvac_ [108]
[21]	2.3	17.81	0.00		<spontaneous> pow.A [21]
[22]	2.1	4.42 4.42	11.92 11.92	93888/93888 93888	concen_ [4] thdiff_ [22]

```

-----
                11.92    0.00  281664/281664    thd_ [25]
-----
                0.70    0.00   93888/1971648    concen_ [4]
                14.09    0.00  1877760/1971648    temp_ [3]
[23]    1.9    14.79    0.00  1971648    tridi_ [23]
-----
                <spontaneous>
                sin.A [24]
-----
                11.92    0.00  281664/281664    thdiff_ [22]
[25]    1.5    11.92    0.00  281664    thd_ [25]
-----
                <spontaneous>
                __powr8i4 [26]
-----
                9.29    0.00  9920007/9920007    vsgws4_ [19]
[27]    1.2    9.29    0.00  9920007    lpgws4_ [27]
-----
                6.94    0.00  9920007/9920007    prod_ [20]
[28]    0.9    6.94    0.00  9920007    optic_ [28]
-----
                6.63    0.00  2035055/2035055    gts5_ [6]
[29]    0.9    6.63    0.00  2035055    glob5l_ [29]
-----
                <spontaneous>
                sin [30]
-----
                <spontaneous>
                log.A [31]
-----
                1.96    0.73  31459/31459    MAIN__ [1]
[32]    0.3    1.96    0.73  31459    coord_ [32]
                0.73    0.00  62918/2942150    differ_ [14]
                0.00    0.00    1/31460    magreq_ [104]
-----
                2.49    0.00  31459/31459    prod_ [20]
[33]    0.3    2.49    0.00  31459    nprod_ [33]
-----
                0.67    0.00  187776/563328    MAIN__ [1]
[34]    0.3    1.33    0.00  375552/563328    concen_ [4]
                2.00    0.00  563328    reac_ [34]
-----
                <spontaneous>
                log10.A [35]
-----
                <spontaneous>
                cos [36]
-----
                1.45    0.00  3663099/3663099    gts5_ [6]
[37]    0.2    1.45    0.00  3663099    ccor_ [37]
-----
                0.54    0.89  19840014/19840014    gws4_ [9]
[38]    0.2    0.54    0.89  19840014    wprof4_ [38]
                0.67    0.00  2938104/2938104    splinw_ [51]
                0.22    0.00  2938104/2938104    splint_ [58]
-----
                0.01    0.00    163/31459    MAIN__ [1]
[39]    0.2    1.35    0.00  31296/31459    temp_ [3]
                1.36    0.00  31459    qphe_ [39]
-----
                1.34    0.00  2849077/2849077    gts5_ [6]
[40]    0.2    1.34    0.00  2849077    dnet_ [40]
-----
                <spontaneous>
                exp.L [41]
-----
                0.26    1.05  31296/31296    MAIN__ [1]
[42]    0.2    0.26    1.05  31296    mgfld2_ [42]
                1.05    0.00  9837312/9896738    nraphs_ [45]
-----
                <spontaneous>
                acos [43]
-----
                <spontaneous>
                cosh [44]
-----
                0.01    0.00  59426/9896738    mgfld1_ [83]
[45]    0.1    1.05    0.00  9837312/9896738    mgfld2_ [42]
                1.06    0.00  9896738    nraphs_ [45]
-----
                <spontaneous>

```

[46]	0.1	1.05	0.00		cos.L [46]
[47]	0.1	0.92	0.00	31459/31459	MAIN__ [1] mdirec_ [47]
[48]	0.1	0.81	0.00	9920007/9920007	MAIN__ [1] mg2geo_ [48]
[49]	0.1	0.74	0.00		<spontaneous> asinh.A [49]
[50]	0.1	0.69	0.00		<spontaneous> exp [50]
[51]	0.1	0.67	0.00	2938104/2938104	wprof4_ [38] splinw_ [51]
[52]	0.1	0.64	0.00		<spontaneous> atan2 [52]
[53]	0.1	0.61	0.00		<spontaneous> __powi4i4 [53]
[54]	0.1	0.40	0.00		<spontaneous> sinh [54]
[55]	0.0	0.36	0.00		<spontaneous> sin.L [55]
[56]	0.0	0.28	0.00		<spontaneous> atan.A [56]
[57]	0.0	0.27	0.00		<spontaneous> cvtas_t_to_a [57]
[58]	0.0	0.22	0.00	2938104/2938104	wprof4_ [38] splint_ [58]
[59]	0.0	0.21	0.00		<spontaneous> for_write_seq_fmt_xmit [59]
[60]	0.0	0.19	0.00		<spontaneous> cvtas_a_to_t [60]
[61]	0.0	0.13	0.00		<spontaneous> for__interp_fmt [61]
[62]	0.0	0.13	0.00		<spontaneous> pow [62]
[63]	0.0	0.12	0.00		<spontaneous> pow.L [63]
[64]	0.0	0.11	0.00		<spontaneous> cvt_ieeee_t_to_text_ex [64]
[65]	0.0	0.11	0.00		<spontaneous> matherr1 [65]
[66]	0.0	0.10	0.00		<spontaneous> matherr [66]
[67]	0.0	0.09	0.00		<spontaneous> for__format_value [67]
[68]	0.0	0.00	0.08	31296/31296	MAIN__ [1] exbv_ [68]
[69]	0.0	0.01	0.07	192/192	sfvdm_ [69] exbv_ [68]
[70]	0.0	0.01	0.07	192	sfvdm_ [69] bspl4t_ [75]
[71]	0.0	0.04	0.00	119808/119808	bspl41_ [78]
[72]	0.0	0.03	0.00	119808/119808	bspl41_ [78]
[73]	0.0	0.00	0.00	192/192	g_ [106]
[74]	0.0	0.08	0.00		<spontaneous> log10 [70]
[75]	0.0	0.06	0.00		<spontaneous> for_read_seq_fmt_xmit [71]

[72]	0.0	0.05	0.00		<spontaneous> for__close_proc [72]
[73]	0.0	0.05	0.00		<spontaneous> for_write_seq_fmt [73]
[74]	0.0	0.05	0.00		<spontaneous> log.L [74]
[75]	0.0	0.04	0.00	119808/119808	sfvdm_ [69] bspl4t_ [75]
[76]	0.0	0.04	0.00		<spontaneous> for__desc_ret_item [76]
[77]	0.0	0.04	0.00		<spontaneous> for__put_sf [77]
[78]	0.0	0.03	0.00	119808/119808	sfvdm_ [69] bspl4t_ [78]
[79]	0.0	0.03	0.00		<spontaneous> asinh.L [79]
[80]	0.0	0.03	0.00		<spontaneous> cvtas_string_to_nan_x [80]
[81]	0.0	0.03	0.00		<spontaneous> log [81]
[82]	0.0	0.03	0.00		<spontaneous> sinh.A [82]
[83]	0.0	0.02	0.01	163/163	MAIN__ [1] mgfldl_ [83]
		0.01	0.00	59426/9896738	nraphs_ [45]
[84]	0.0	0.02	0.00	31459/31459	MAIN__ [1] writ04_ [84]
[85]	0.0	0.02	0.00		<spontaneous> cosh.A [85]
[86]	0.0	0.02	0.00		<spontaneous> for__write_output [86]
[87]	0.0	0.01	0.00	15811/15811	MAIN__ [1] writ20_ [87]
[88]	0.0	0.01	0.00		<spontaneous> asinh [88]
[89]	0.0	0.01	0.00		<spontaneous> atan.L [89]
[90]	0.0	0.01	0.00		<spontaneous> cvt_ieee_t_to_text [90]
[91]	0.0	0.01	0.00		<spontaneous> cvt_integer_to_text [91]
[92]	0.0	0.01	0.00		<spontaneous> cvt_text_to_ieee_t [92]
[93]	0.0	0.01	0.00		<spontaneous> for__acquire_lun [93]
[94]	0.0	0.01	0.00		<spontaneous> for__flush_readahead [94]
[95]	0.0	0.01	0.00		<spontaneous> for__get_s [95]
[96]	0.0	0.01	0.00		<spontaneous> for__put_d [96]
[97]	0.0	0.01	0.00		<spontaneous> for__release_lun [97]
[98]	0.0	0.01	0.00		<spontaneous> llround [98]

[99]	0.0	0.01	0.00		<spontaneous> log10.L [99]
[100]	0.0	0.01	0.00		<spontaneous> prmsg5_ [100]
[101]	0.0	0.01	0.00		<spontaneous> _intel_fast_memcpy [101]
[102]	0.0	0.01	0.00		<spontaneous> _intel_fast_memcpy.J [102]
[103]	0.0	0.01	0.00		<spontaneous> _intel_fast_memset.J [103]
[104]	0.0	0.00	0.00	1/31460 31459/31460 31460	coord_ [32] MAIN__ [1] magreq_ [104]
[105]	0.0	0.00	0.00	978/978 978	MAIN__ [1] magreb_ [105]
[106]	0.0	0.00	0.00	192/192 192	sfvdm_ [69] g_ [106]
[107]	0.0	0.00	0.00	1/1 1 1/1	MAIN__ [1] dipole_ [107] igrf_ [110]
[108]	0.0	0.00	0.00	1/1 1	prod_ [20] euvac_ [108]
[109]	0.0	0.00	0.00	1/1 1	MAIN__ [1] geo2mg_ [109]
[110]	0.0	0.00	0.00	1/1 1	dipole_ [107] igrf_ [110]
[111]	0.0	0.00	0.00	1/1 1	glbw4e_ [12] tsgws4_ [111]

Index by function name

[1] MAIN__	[68] exbv_	[104] magreq_
[53] __powi4i4	[50] exp	[66] matherr
[26] __powr8i4	[11] exp.A	[65] matherr1
[101] _intel_fast_memcpy	[41] exp.L	[47] mdirec_
[102] _intel_fast_memcpy.J	[93] for_acquire_lun	[48] mg2geo_
[103] _intel_fast_memset.J	[72] for_close_proc	[83] mgfld1_
[43] acos	[76] for_desc_ret_item	[42] mgfld2_
[88] asinh	[94] for_flush_readahead	[33] nprod_
[49] asinh.A	[67] for_format_value	[45] nraphs_
[79] asinh.L	[95] for_get_s	[28] optic_
[56] atan.A	[61] for_interp_fmt	[62] pow
[89] atan.L	[96] for_put_d	[21] pow.A
[52] atan2	[77] for_put_sf	[63] pow.L
[5] atmos_	[97] for_release_lun	[100] prmsg5_
[78] bspl4l_	[86] for_write_output	[20] prod_
[75] bspl4t_	[71] for_read_seq_fmt_xmit	[39] qphe_
[37] ccor_	[73] for_write_seq_fmt	[34] reac_
[4] concen_	[59] for_write_seq_fmt_xmit	[69] sfvdm_
[32] coord_	[106] g_	[30] sin
[36] cos	[109] geo2mg_	[24] sin.A
[13] cos.A	[12] glbw4e_	[55] sin.L
[46] cos.L	[29] glob51_	[54] sinh
[44] cosh	[10] globe5_	[82] sinh.A
[85] cosh.A	[18] globw4_	[58] splint_
[90] cvt_ieee_t_to_text	[6] gts5_	[51] splinw_
[64] cvt_ieee_t_to_text_ex	[9] gws4_	[15] sqrt
[91] cvt_integer_to_text	[110] igrf_	[3] temp_
[92] cvt_text_to_ieee_t	[7] ionhr_	[25] thd_
[60] cvtas_a_to_t	[98] llround	[22] thdiff_
[80] cvtas_string_to_nan_x	[81] log	[23] tridi_
[57] cvtas_t_to_a	[31] log.A	[111] tsgws4_
[17] denss_	[74] log.L	[19] vsgws4_
[14] differ_	[70] log10	[8] wind_
[107] dipole_	[35] log10.A	[38] wprof4_
[40] dnet_	[99] log10.L	[84] writ04_
[16] elhr_	[27] lpgws4_	[87] writ20_

[108] euvac_

[105] magreb_