



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

**DESENVOLVIMENTO DO MODELO SUPIM-DAVS PARA A PLATAFORMA
NEC SX-AURORA TSUBASA**

Luiz Henrique Broch Lago

Relatório Final de Iniciação Científica do
programa PIBIC, orientada por Haroldo F.
de Campos Velho.

UFSM
Santa Maria, RS
2022

**DESENVOLVIMENTO DO MODELO SUPIM-DAVS PARA A PLATAFORMA
NEC SX-AURORA TSUBASA**

Luiz Henrique Broch Lago

Relatório Final de Iniciação Científica do
programa PIBIC, orientada por Haroldo F.
de Campos Velho.

UFSM
Santa Maria, RS
2022

RESUMO

Esse trabalho iniciado em setembro de 2021 teve como foco a continuação da migração do programa SUPIM para a plataforma de processamento vetorial do computador SX-Aurora Tsubasa NEC. Uma das atividades do programa de clima espacial do INPE (*) é a previsão de conteúdo eletrônico total (TEC: Total Electron Content) realizado por um modelo de dinâmica da ionosfera SUPIM (Sheffield University Plasmasphere-Ionosphere Model), adaptado para execução operacional diária**. O programa SUPIM é executado realizando cálculos de simulação até 24 horas, com saídas a cada hora do dia em diversos pontos de coordenadas, onde um processo iterativo é ativado até a convergência. Uma ação de permanente investigação é a execução deste tipo de modelo complexo de simulação de sistemas dinâmicos com uso de computação intensiva em diferentes arquiteturas de hardware. O objetivo do presente projeto é avaliar o desempenho do modelo SUPIM na nova arquitetura vetorial NEC SXTsubasa / Vector Engine e adapta-lo para conseguir fazer uso dos recursos a fim de ter um ganho de desempenho. O sistema NEC SX-Tsubasa apresenta ser uma boa solução pelo grande poder computacional disponível para a realização dos cálculos, uma alta banda de memória e com excelente relação de desempenho de processamento com baixa demanda de energia. Otimizações no código tiveram que ser feitas para obtenção de melhor desempenho com os recursos da arquitetura vetorial. Serão apresentados os resultados após cada iteração de desempenho no sistema computacional de operação do SUPIM no CRS-INPE, comparado ao desempenho na nova arquitetura de processamento. A versão do SUPIM no cluster operacional tem tempo de CPU de 4730 segundos, enquanto a versão do hardware vetorial da NEC foi executada com 1050 segundos - ou seja, um ganho de desempenho superior a 4,5 vezes da versão operacional atual do SUPIM.

Palavras-chaves: ionosfera, modelo SUPIM, conteúdo eletrônico total, processamento de alto desempenho, processamento vetorial

SUMÁRIO

1	INTRODUÇÃO	1
1.1	MODELO SUPIM	1
1.2	HARDWARE DE PROCESSAMENTO VETORIAL.....	2
1.3	OBJETIVO	2
2	ANÁLISE	2
2.1	HARDWARE NEC	2
2.2	TESTE DE DESEMPENHO NOS HARDWARES	3
2.3	MIGRAÇÃO DO SUPIM.....	3
2.4	EXECUÇÃO DO SUPIM.....	3
2.5	RESULTADOS PRELIMINARES	4
2.6	AÇÕES PARA MELHORIA DE DESEMPENHO	4
3	DESENVOLVIMENTO.....	5
3.1	PROCESSO DE VETORIZAÇÃO	5
3.2	FUNÇÕES ALVO	6
3.3	RESULTADO.....	6
4	CONCLUSÃO.....	6

1 INTRODUÇÃO

O presente relatório apresentará alguns aspectos na integração do software SUPIM em máquinas de processamento vetorização, que apresentam como um grande potencial para a execução de cálculos com ganhos de desempenho. O projeto foi desenvolvido via acordo entre o INPE e a NEC-Brasil, com ciência e concordância da NEC-Japão.

1.1 MODELO SUPIM

O principal produto da simulação com o modelo SUPIM é o cálculo da concentração de elétrons na ionosfera. As suas equações são resolvidas em pontos discretos sobre as linhas de campo magnético da terra, calculando o valor de concentração, fluxos de alinhamento de campo e temperatura. O arquivo de entrada do SUPIM contém as informações de local geográfico (latitude e longitude) e um arquivo com o fluxo solar. O SUPIM está configurado para realizar previsões de 24 horas, com saídas (arquivos) de simulação a cada hora.

A execução do SUPIM demanda intenso processamento e recursos de memória principal e secundária (armazenamento de dados). O processamento paralelo é realizado via decomposição de domínio que, em coordenadas magnéticas, permite expressar o sistema como um modelo várias regiões 2D desacopladas. A execução do SUPIM em máquinas de processamento paralelo distribuído pode demandar várias horas, dependendo da configuração de hardware. Resultados de previsão diárias podem ser acessadas no sítio do Programa de Clima Espacial do INPE (Embrace): TEC-SUPIM¹. O SUPIM tem aproximadamente 12 mil linhas de programação. Originalmente, o modelo SUPIM foi desenvolvido para computadores sequenciais.

Uma ação permanente é investigar diferentes arquiteturas de hardware para aprimorar a execução do modelo SUPIM, com adaptação do código à arquitetura disponível/investigada. Este é o objetivo do presente projeto.

¹ TEC SUPIM: <http://www2.inpe.br/climaespacial/portal/tec-supim-previsao/>

1.2 HARDWARE DE PROCESSAMENTO VETORIAL

A NEC desenvolveu um novo hardware de processamento vetorial paralelo, o sistema SX-Aurora TSUBASA. O sistema permite escalonamento de hardware, com integração simples de novos módulos de processamento. Além da vantagem de expansão, esse hardware apresenta um desempenho de processamento de 2,45 TFlops e uma largura de banda de memória de 1.2 TBytes por segundo. Há compiladores específicos para as linguagens C/C++/Fortran e compatível com padrões OpenMP e MPI de processamento paralelo.

1.3 OBJETIVO

Migração de uma cópia do software SUPIM para o servidor SX-Aurora TSUBASA da NEC-Japão. Compilação utilizando diretivas apropriadas, com indicação de alteração do código, caso necessário e verificar o desempenho do software e possíveis adaptações para o novo hardware.

2 ANÁLISE

2.1 HARDWARE NEC

O acesso ao equipamento SX-Aurora TSUBASA da NEC-Japão foi via acesso SSH utilizando certificados criptografados. A estrutura dos servidores era baseada em um servidor gateway, com acesso à internet e, deste servidor, tem-se acesso máquina SX-Aurora TSUBASA que era composta por 4 módulos vetoriais (VE: Vector Engine). Cada módulo VE contém 64 registradores vetoriais, processador de 1.4Ghz, 8 núcleos de processamento e capacidade de memória de 48GB.

2.2 TESTE DE DESEMPENHO NOS HARDWARES

Preliminarmente, executou-se um “programa teste”, para ambientação com compiladores e padrões de paralelismo OpenMP e MPI do servidor SX-Aurora TSUBASA. Teste simples com produto interno de 2 vetores com milhões de elementos e o produto interno era recalculado 1 milhão de vezes. O resultado foi um tempo de processamento de (aproximadamente) 10 minutos no cluster multiprocessado do CRSINPE e de ~ 10 segundos no SX-Aurora TSUBASA. Alguns pontos a se destacar nesses testes foi que o hardware TSUBASA não apresentou um bom escalonamento ao utilizar diversas threads, o que pode ser justificado pela deficiência do algoritmo que utilizou de memória compartilhada para a execução.

2.3 MIGRAÇÃO DO SUPIM

A migração do SUPIN deve ser feita de uma cópia do SUPIM que estava armazenado no servidor do INPE e encaminhada para o servidor da NEC. Ambos os servidores possuem máquinas intermediárias como *gateways*.

Para fazer a transferência dos arquivos, utilizou-se o protocolo FTP para a transferência dos arquivos do servidor do INPE para o seu *gateway* e após para uma máquina intermediária. Não era possível fazer a transferência dos arquivos do *gateway* do INPE direto para o *gateway* da NEC pois a autenticação era feita utilizando o certificado digital, além de não ser possível o acesso ao *gateway* da NEC usando o protocolo FTP. Ao final o processo de enviar os arquivos que estavam na máquina intermediária foi utilizando o protocolo SFTP, que pode ser utilizado em conjunto com o SSH, e fornecesse uma camada de segurança para transferências de arquivos.

2.4 EXECUÇÃO DO SUPIM

Após feito a transferência de todos os arquivos necessários para a execução do SUPIM, algumas modificações nas *flags* do compilador disponibilizado pela NEC tiveram de serem feitas por conta do hardware diferente e a versão do Fortran no

compilador ser diferente e apresentar mensagens de saída na compilação diferentes que quando compilado no servidor do INPE.

Inicialmente foi necessário corrigir alguns trechos de código que estavam gerando erros durante o processo de compilação. Alguns desses trechos eram relacionados a variáveis declaradas implicitamente no qual eram atribuídos tipos que se tornariam incompatíveis com os comandos a frente. Na primeira execução do SUPIM no hardware de vetorização da NEC, resultou em um erro na execução dos cálculos, que retornava um valor negativo para a temperatura do íon. A solução para esse problema foi a adição de uma *flag* no compilador que inicializava as variáveis com o valor zero por padrão. No servidor do INPE era utilizado o compilador fornecido pela Intel que já fazia essas inicializações

2.5 RESULTADOS PRELIMINARES

Na primeira versão compilada, não foi usada nenhuma diretiva de otimização. Em ações subsequentes, incorporou-se as diretivas de compilação com diferentes níveis de otimização: O1, O2, O3 e diretivas “inline”. A tabela abaixo mostra o tempo de processamento (horas:minutos:segundos) com os diferentes tipos de opções de compilação.

Tipos de otimizações	Tempo de processamento
Sem otimização:	01:18:50
Diretivas “-O3” e “inline”:	00:39:47

2.6 AÇÕES PARA MELHORIA DE DESEMPENHO

Como o modelo SUPIM foi desenvolvido para máquinas sequencias, utilizou-se o profiler residente no servidor da NEC-Japão para identificar as rotinas e partes do código SUPIM com maiores demandas de processamento. Foram identificadas as rotinas com mais chamadas e que não estão vetorizadas de forma adequada. Por demanda de uso no SUPIM as rotinas e percentual de tempo de CPU no SUPIM estão listadas abaixo:

- a) GLOBE7: 28.7%
- b) GTS7: 17.2%
- c) DENSU: 16.8%
- d) HWMupdate: 13.1%

3 DESENVOLVIMENTO

3.1 PROCESSO DE VETORIZAÇÃO

O processo de vetorização utilizado consistia em adaptar a estrutura do software de tal forma que o compilador fornecido pela NEC conseguisse utilizar diretivas e otimizações de código compatíveis com o hardware alvo. Para atingir esse objetivo, foi necessário a adaptação do código e parâmetros de diversas funções do código SUPIM.

Para ser possível a vetorização total pelo compilador do código, se faz necessário transformar as estruturas complexas do código fonte em estruturas mais simples como loops e acesso sequenciais em vetores. O excesso de chamadas de funções e procedimento pelo código do SUPIM também causava em uma perda substancial de desempenho, que também foi um foco na melhora do código fonte.

Após feito um levantamento das funções que tinham o maior tempo de execução, começou um processo de diminuição do excesso de chamadas de uma mesma função, que antes recebiam variáveis escalares, agora recebem vetores de valores no qual era feito o procedimento que a função fazia agora para cada posição no vetor. Tornando as estruturas internas dos procedimentos em menos chamadas de funções e a abstraindo as estruturas em simples loops de acessos as posições e efetuação do cálculo possibilitou que muitas funções fossem vetorizadas corretamente pelo compilador aumentando muito a taxa de operações vetorial o que é um bom indicativo do desempenho em um hardware vetorial como o utilizado.

3.2 FUNÇÕES ALVO

Com o levantamento das funções com maior tempo de processamento da etapa de análise, selecionou-se as funções que teriam maior impacto se a vetorização fosse aplicada, no qual se exclui somente a HWMupdate pois não traria um ganho tão substancial, além de sua estrutura não possibilitar a vetorização tão facilmente quanto as outras funções destacadas

3.3 RESULTADO

Após cada função ser vetorizada se realizou um teste para analisar o ganho de desempenho e a quantidade de gasta na função que foi vetorizada. Na tabela abaixo é mostrado o resultado dos testes e a função que a última função vetorizada antes do teste:

FUNÇÃO VETORIZADA	TEMPO EXCLUSIVO %	TEMPO TOTAL DA EXECUÇÃO
GLOBE7	0.6%	26min 6seg
GTS7	8.1%	23min 20seg
DENSU	2.7%	20min 25seg
SPLIN'S	2,4%	17min 30seg

O tempo exclusivo se traduz na porcentagem do tempo que a função executa em relação ao tempo total da aplicação. Para os resultados na questão de tempo, as funções vetorizadas mais abaixo também incluem as vetorizações das funções acima delas, ou seja, após a vetorização da função GTS7 o código já incluía a vetorização da GLOBE7 então o ganho na questão do tempo não se entende exclusivamente por conta da vetorização da função em questão.

4 CONCLUSÃO

Junto ao início da análise do processo, foi possível diminuir o tempo da execução do SUPIM utilizando diretivas de otimização além de fazer um levantamento das funções que necessitavam de uma melhorada no código.

Após o processo da vetorização das funções destacadas, se chegou a ter um ganho de 56% em relação ao tempo relativo a fase de análise e relativo a primeira execução o código conseguiu executar em apenas 21% do tempo total o que leva um ganho substancial para o levantamento dos dados gerados pelo software.