

OTIMIZAÇÃO DO SISTEMA DE ASSIMILAÇÃO DE DADOS ATMOSFÉRICOS PSAS PARA O MODELO GLOBAL DO CPTEC

Rosângela Cintra, Dirceu Herdies, Jonas N.Tamaoki

Centro de Previsão de Tempo e Estudos Climáticos - CPTEC/INPE
Rodovia Presidente Dutra, km 40
12630-000 -Cachoeira Paulista - SP
e-mail:rcintra@cptec.inpe.br,dirceu@cptec.inpe.br,tamaoki@cptec.inpe.br

ABSTRACT

A Physical-space Statical Analysis System (PSAS) has been developed at Data Assimilation Office (DAO), NASA/GSFC. PSAS has been designed as an incremental improvement over the current Optimal Interpolation (OI) based in Data Assimilation System (DAS), and provide a framework to test advanced forecast error covariance models; a global analysis PSAS resulting for the CPTEC/COLA model, is being developed at Centro de Previsão de Tempo e Estudos Climáticos (CPTEC) The complexity of PSAS solution is still computationally expensive. For operational feasibility, several physical approximations have been implemented in PSAS, and the code has been parallelized to utilize a shared memory multi-processor NEC-SX4 system, using OpenMP programming, it would become possible to provide weather forecasts initial field in a timely fashion.

1. INTRODUÇÃO

Assimilação de Dados atmosféricos gera uma estimativa do estado da atmosfera num determinado tempo, através da combinação de uma previsão, fornecida por um modelo e de observações do estado da atmosfera no mesmo instante de tempo. Vem sendo o principal meio para fornecer condições iniciais para modelos de Previsão Numérica de Tempo (PNT). Matematicamente, a assimilação de dados pode ser descrita por teoria de estimação. Existem esquemas tradicionais como o da Interpolação Ótima (OI), bem como métodos baseados em algoritmos estatísticos usando a teoria do filtro de Kalman; o PSAS possui um esquema que suporta um filtro de Kalman sub-ótimo.

O Centro de Previsão de Tempo e Estudos Climáticos (CPTEC) está implantando um sistema de assimilação de dados que combina através de métodos de análise objetiva, todo o componente ar-terra-mar em determinada escala de tempo e espaço para gerar as condições iniciais para seus modelos de PNT, utiliza um sistema desenvolvido pela Divisão de Assimilação de Dados (DAO) da NASA/ Goddard Space Flight Center-EUA, o “Physical-Space Statistical Analysis System” (PSAS) que permite trabalhar com vínculos realistas. O PSAS relaxa as aproximações locais e as restrições na seleção de dados usadas na OI, mas ainda tem o custo computacional alto. Atualmente o CPTEC executa modelo global e regional que utilizam como condição inicial para as suas rodadas análises provenientes do National Center for Environmental Prediction (NCEP). A introdução deste sistema de assimilação de dados permitirá a obtenção de análise própria que providenciará a condição inicial dos modelos global de PNT do CPTEC.

Além de introduzir o PSAS no CPTEC, para que o centro tenha um sistema moderno de assimilação de dados globais, gerando análises operacionalmente viáveis, o objetivo principal deste trabalho é otimizar o sistema para que torne possível o desempenho das análises globais de modo detalhado e preciso em tempo hábil de suprir os modelos de previsão de tempo com seu campo de inicialização. A implementação inicial foi feita no computador NEC-SX4 utilizando OpenMP para sistema de memória compartilhada, com intuito de obter o melhor desempenho operacional do sistema, verificando o ganho com a paralelização e posteriormente aperfeiçoaremos seu desempenho ajustando-o à máquina vetorial. A introdução deste sistema de assimilação de dados eficiente trará benefícios diretos na qualidade da previsão de tempo gerada pelo centro e abrirá também importante linha de pesquisa no país.

2. METODOLOGIA

2.1 Sistema de Análise Estatística em Espaço-Físico - PSAS

O PSAS é um sistema baseado em Interpolação Ótima (OI) (Pfaendtner et al. 1995), embora formule as equações de análise de uma maneira diferente, são matematicamente equivalentes (GUO e da Silva, 1995/ da Silva et al. , 1995). Devido ao tamanho da solução, o PSAS relaxa as aproximações locais através de um método numérico: o algoritmo gradiente conjugado em forma de blocos para resolver a equação iterativamente, e remove as restrições da seleção de dados para simplificar a computação. Possui estrutura avançada para testar erros de covariância da previsão com suporte do Filtro de Kalman.

2.1.1 O algoritmo

2.1.1.1 A solução PSAS global

Um esquema de análise estatística procura obter uma estimativa ótima, ou análise, do estado do sistema dinâmico pela combinação das observações da atmosfera com uma primeira previsão de modelo, Temos $w^f \in \mathfrak{R}^n$ simboliza o vetor representando a primeira previsão de tempo, definida em grade, e temos $w^t \in \mathfrak{R}^n$ simplificando o estado discreto verdadeiro aproximado por w^f ,

$$w^f = w^t + \varepsilon^f \quad (1)$$

onde $\varepsilon^f \in \mathfrak{R}^n$ significa o erro da previsão. Um índice de tempo é omitido nesta equação e em outras para ter a notação simplificada. Temos $w^o \in \mathfrak{R}^o$ que significa o vetor de p observações disponíveis no instante da análise, assumindo aqui como linearmente relacionada à variáveis de estado,

$$w^o = Hw^t + \varepsilon^o \quad (2)$$

aqui $H \in \mathfrak{R}^{p \times n}$ é o operador observação, ou operador de interpolação generalizado que transforma variáveis do modelo em variáveis observacionais, e $\varepsilon^o \in \mathfrak{R}^p$ significa os erros das observações, que é a soma dos erros dos instrumentos e o erro de representatividade (Lorenç,1986; Cohn,1997). Tipicamente, o número de graus de liberdade do modelo n (número de pontos de grade \times variáveis) é aproximadamente de 10^6 e o sistema de observações atual p (número de observações meteorológicas) é de aproximadamente 10^5 . A análise da variância mínima $w^a \in \mathfrak{R}^n$ é obtida requerendo $\langle (w^a - w^t)^T S (w^a - w^t) \rangle$ para ser o mínimo de todas matrizes $S \in \mathfrak{R}^n \times \mathfrak{R}^n$, e sob algumas supostas simplificações dadas pelas equações de análise.

$$w^a = w^f + K(w^o - Hw^f) \quad (3)$$

$$K = P^f H^T (HP^f H^T + R)^{-1} \quad (4)$$

A matriz K é chamada ganho de Kalman ou os pesos de inovação, que depende da matriz de erros de covariância da previsão,

$$P^f \equiv \langle (\varepsilon^f - \langle \varepsilon^f \rangle)(\varepsilon^f - \langle \varepsilon^f \rangle)^T \rangle \in \mathfrak{R}^{n \times n} \quad (5)$$

e da matriz de erros de covariância da observação.

$$R \equiv \langle (\varepsilon^o - \langle \varepsilon^o \rangle)(\varepsilon^o - \langle \varepsilon^o \rangle)^T \rangle \in \mathfrak{R}^{p \times p} \quad (6)$$

Ambas são simétricas e positivas semi-definidas por definição; R é de fato positiva definida sob a suposição de que nenhuma combinação não linear é perfeita. Estas matrizes são definidas como acima, mas na prática precisam ser modeladas. O algoritmo PSAS resolve as equações (3) e (4) de maneira direta. Primeiro, um sistema linear $p \times p$ é resolvido para a quantidade y ,

$$(HP^f H^T + R)y = w^o - Hw^f \quad (7)$$

e então o estado analisado w^a é obtido da equação

$$w^a = w^f + P^f H^T y \quad (8)$$

As equações (7) e (8) serão referenciadas como as equações do PSAS. A matriz de covariância de inovação é simétrica positiva definida,

$$M \equiv HP^f H^T + R \quad (9)$$

o algoritmo pré-condicionado gradiente conjugado (CG) [1] é utilizado como método de escolha para resolver o grande sistema linear, resolve a equação iterativamente com uma série de aproximações físicas e numéricas para simplificar a computação. Para o sistema de observações atual, resolver o sistema linear (7) representa metade do

esforço computacional do PSAS, e envolve cálculos no espaço de observação: $M \in \mathfrak{R}^{pxp}$ e $y \in \mathfrak{R}^p$. A outra metade do custo computacional é dada pelo passo (8) que transfere a solução y para o espaço do vetor estado: $P^f H^T y \in \mathfrak{R}^n$. A complexidade da solução PSAS global requer $\sim O(p^3)$ de operações de ponto flutuante de pelo método direto, e $\sim O(np^2)$ para resolver as equações iterativamente. O algoritmo no PSAS consiste em resolver um sistema linear pxp para um quantidade y que consiste de uma multiplicação matriz-vetor mais uma adição de vetor, sem requerer iteração. O vetor intermediário y será referenciado como as inovações de peso parciais.

Esta versão do sistema PSAS analisa globalmente pressão ao nível do mar e ventos próximos à superfície do mar, também analisa altura geopotencial, vetor vento e razão de mistura do vapor d'água em níveis constantes de pressão. A análise de ar-superior altura/vento e de superfície pressão ao nível do mar/ventos à superfície são multivariados usando modelos de erros de covariância vento-massa. A análise de umidade é desempenhada usando um algoritmo estatístico nos níveis de 1000 hPa a 300 hPa. A configuração básica consiste de 2° de latitude por $2,5^\circ$ de longitude e 18 níveis verticais (0.4, 1, 5, 7, 10, 30, 50, 70, 100, 150, 200, 250, 300, 400, 500, 700, 850, 1000 hPa).

2.1.1.2 Considerações Computacionais

Algumas considerações foram implementadas visando a eficiência do algoritmo do PSAS:

- Para introduzir alguma esparsividade em M e por isso salvar algum esforço computacional, a esfera é dividida em N regiões, e os blocos matriciais associados com regiões separadas por mais de 6000 Km são nulos, tal que o produto parcial matriz-vetor para as regiões pode ser pulado, estes blocos nunca entram nos cálculos de CG. A esparsividade das matrizes é suportada por funções de correlação horizontal. O mesmo procedimento é aplicado para a matriz P^f na equação (8).
- Para o problema de interpolação o PSAS implementa um candidato natural ao pré-condicionador para o algoritmo gradiente conjugado (CG), é uma aproximação do tipo OI, em que o problema é resolvido separadamente para cada uma das N regiões em que os dados foram divididos.
- Nesta implementação o globo foi dividido em 80 regiões de áreas iguais usando uma grade icosaedral (Pfaendtner, 1996) (4 regiões triangulares esféricas para cada um dos 20 lados de um icosaedro). O vetor inovação $w_0 - Hw_f$ para as observações locais permite multiplicações matriz-vetor para executar em forma de bloco.
- Uma grade de área quase-igual é usada para reduzir o custo computacional da derivação do incremento de análise. Diferentes definições de longitudes são usadas para diferentes latitudes, para guardar a densidade do ponto de grade sobre o globo, todavia a redundância em uma grade de intervalo igual de regiões de latitudes mais altas é removida. O resultado são futuras interpolações para requerer igual grade de intervalo latitude-longitude.
- A implementação serial do PSAS, a matriz M é passada para o esquema do CG global como um operador, e os elementos das matriz são recalculados a cada iteração do CG.

2.1.1.3 Características do algoritmo PSAS

O PSAS foi implementado em Fortran 90 e o "driver" de resolução das suas equações utilizando o método número **Gradiente Conjugado** pré-condicionado que será o objeto de nosso estudo, tem a seguinte estrutura:

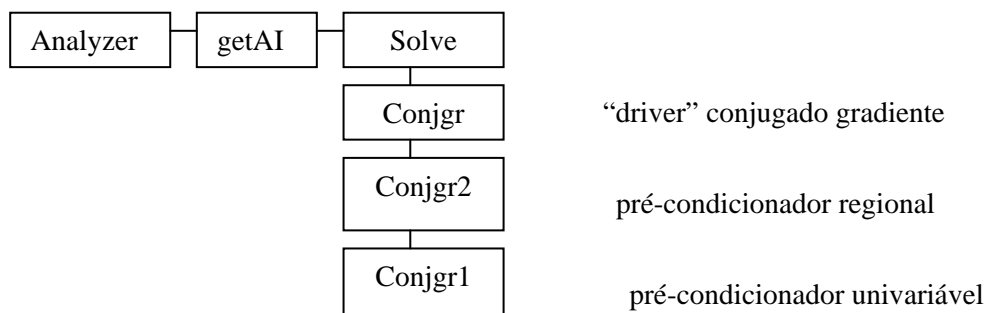


Figura 1 – Diagrama de Blocos dos módulos que compõe o “driver” que resolve os Incrementos de Análise.

Com $p \sim 100.000$ observações e $N \sim 80$ regiões, cada problema regional tem em média mais que 1.000 observações, que é muito para um pré-condicionador eficiente. Então, o problema regional é resolvido por outro algoritmo pré-condicionador gradiente conjugado, referenciado aqui como *CG nível 2*. Para o pré-condicionador do *CG nível 2* o mesmo problema é resolvido univariado para cada tipo de dados, (ex. observações de vento, altura geopotencial, etc.) . Portanto, estes problemas univariados são também grandes para serem resolvidos eficientemente por métodos diretos então ainda outro método iterativo é usado na solução, o *CG nível 1* . Como pré-condicionador do o *CG nível 1* são usadas rotinas da biblioteca LAPACK (Anderson et al., 1992) para realizar diretamente a fatorização de Cholesky dos blocos diagonais da sub-matriz de correlação de nível 1, estes blocos são tipicamente de tamanho 32 e são cuidadosamente escolhidos para incluir perfis verticais completos, uma característica desejável para a implementação de novos tipos de dados. Estes gradientes conjugados pré-condicionados aninhados são ilustrados na Figura 2. O ponto de partida é quando o vetor “inovação” (O-F) está disponível, após o controle de qualidade, e calcula parcialmente os pesos da inovação y (7).

2.1.1.3.1 O “driver” principal do PSAS

A rotina *getAI* inicia processando o número de observações ingeridas da seção do sistema global de observação, uma das tarefas da rotina é dividir estas observações em regiões e classificá-las, utilizando as seguintes chaves: região, tipo de dado (temperatura, vento, etc - kt), fonte do dado (synop, temp, tovs, etc - kx), latitude, longitude, nível. As rotinas abaixo assumem os dados nesta ordem de classificação.

A rotina *solve* prepara os dados para o gradiente conjugado. Calcula coordenadas cartesianas (x,y,z) na unidade esférica correspondente a cada ponto de observação (lat,lon). As coordenadas cartesianas são usadas pelos modelos de covariâncias para calcular as distâncias horizontais. Calcula o índice das sondagens (da Silva and Redder 1995). Coloca os índices e pesos da interpolação, e normaliza os desvios padrões dos erros da previsão e observações. Dado o vetor inovação (observação menos previsão), a rotina retorna a solução y do sistema linear de equações (7). Note que y é definido na localização da observação, o algoritmo gradiente conjugado é chamado para resolver este sistema na forma $Cx = b$.

A rotina *do gradiente conjugado principal (conjgr)* implementa o método direto do gradiente conjugado pré-condicionado já explicado anteriormente e resolve o sistema $Cx = b$ onde C é a matriz de correlação de inovação, e b é o conjunto de múltiplos RHS. Quando desempenha a análise global com PSAS, o RHS é simplesmente o (O-F) normalizado pelo desvio padrão. Os múltiplos RHS são necessários para estimar as variâncias do erro da análise. O algoritmo do Gradiente Conjugado pré-condicionado [1] foi reproduzido como segue:

```

k = 0; x0 = 0; r0 = b
while r_k ≠ 0
  solve Cz_k = r_k ⇒ call conjgr2()
  k = k + 1
  if k = 1 { p1 = z0 }
  else { β_k = r_{k-1}^T z_{k-1} / r_{k-2}^T z_{k-2}
        p_k = z_{k-1} + β_k p_{k-1} }
  q_k = Cp_k ⇒ call sym_Cxpy()
  α_k = z_{k-1}^T r_{k-1} / p_k^T q_k
  x_k = x_{k-1} + α_k p_k
  r_k = r_{k-1} - α_k q_k
end

```

A biblioteca “Basic Linear Algebra Subprograms” (BLAS) é utilizada para desempenhar as operações de álgebra linear como normas, adições de vetores, etc. O custo maior desta rotina é a multiplicação matriz-vetor de correlação global (rotina *sym_Cxpy*).

O pré-condicionador nível 2 (*conjgr2*) tem a estrutura similar ao *conjgr()*, a principal diferença é a invocação do pré-condicionador pela rotina *conjgr1* que age para cada um destes vetores de tipo de dados (univariável). Resolve similarmente o sistema linear de equações $\tilde{C}x = b$ onde \tilde{C} é uma versão simplificada da matriz de covariância da inovação e b é o conjunto de múltiplos RHS.

A estrutura geral do pré-condicionador nível 1 (*conjgr1*) é também similar ao *conjgr()* principal. Todavia, neste nível os blocos de sub-matrizes de correlação são calculados e armazenados. O pré-condicionador resolve

diretamente os sub-blocos diagonais (sondagens agrupadas). A rotina identifica os blocos que contêm os perfis verticais. Um solucionador direto do método de Cholesky é executado nestes blocos usando LAPACK [2].

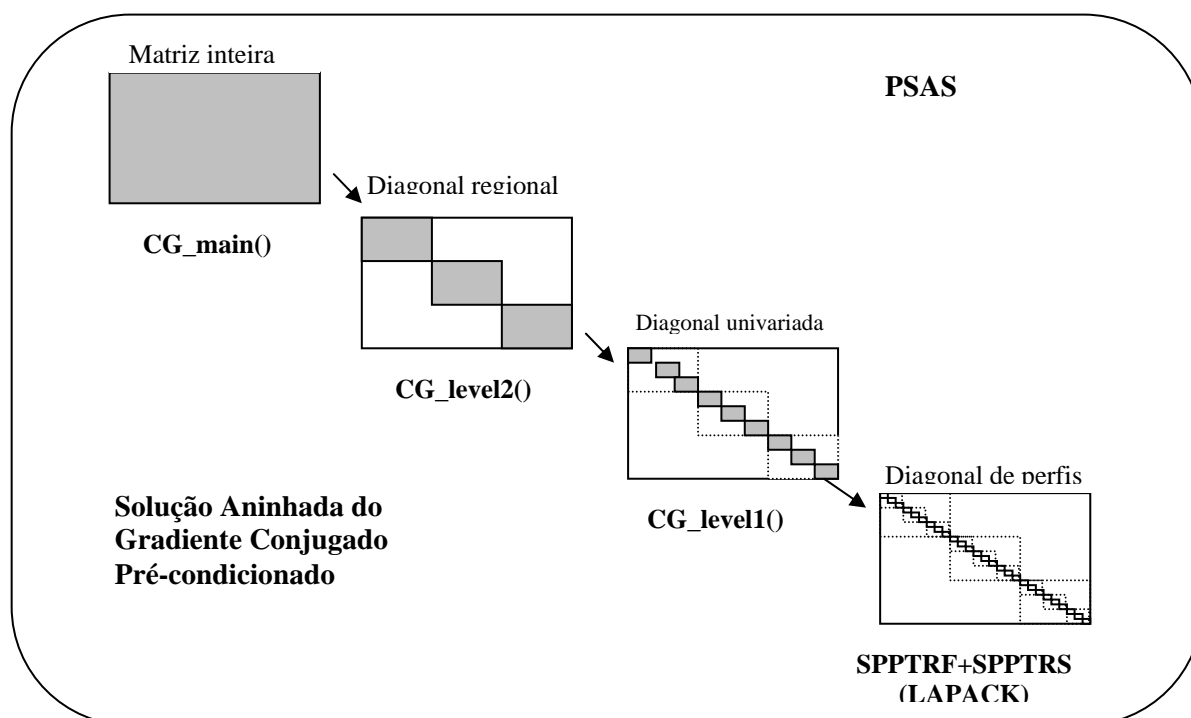


Figura 2-Solução aninhada usando pré-condicionado gradiente conjugado. A rotina CG_main() contém o condutor principal do gradiente conjugado. A rotina é pré-condicionada pelo CG_level2(), que resolve o problema similar para cada região. A rotina é novamente pré-condicionada pelo CG_level1() que resolve o sistema linear univariavelmente.

2.2 Fluxo de dados para o PSAS

O fluxo de dados a ser utilizado pelo PSAS consiste de outros grandes dois sistemas: um *modelo de circulação atmosférica* CPTEC/COLA com grade gaussiana de resolução T126L28 com $1^{\circ} \times 0,93^{\circ}$ (latitude por longitude) na horizontal e 28 níveis sigma na vertical, e um sistema de *observações meteorológicas* provenientes do Global Telecommunication System (GTS/WMO), mas também poderão ser introduzidos dados gerados localmente, como os obtidos dos satélites, PCD, etc. As observações são decodificadas, selecionadas e submetidas a um controle de qualidade. E o modelo global de previsão gera a estimativa inicial (F) proveniente de uma previsão (6h), é interpolada para os pontos com observações (O) para as variáveis correspondentes. A seguir é feita a subtração O-F. Estas diferenças são armazenadas para calibração dos erros usados para o cálculo das matrizes de covariância. O sistema PSAS (baseado em interpolação ótima) produz os incrementos analisando as diferenças (O-F) que são então interpolados para a grade do modelo e ser adicionada à estimativa F. O sistema PSAS tem a grade horizontal na resolução $2^{\circ} \times 2,5^{\circ}$ e 28 níveis verticais de pressão, as variáveis analisadas incluem altura geopotencial e ventos com modelos de função de correlação multivariada, razão de mistura do vapor d'água com função de correlação univariada, bem como pressão ao nível do mar e ventos à superfície via um modelo de correlação balanceada. A análise da pressão ao nível do mar é usado para produzir altura a 1000 hPa complementando a análise de altos níveis. Com as restrições de seleção de dados e aproximação local, o sistema seleciona no máximo 75 observações por cilindro de raio igual a 1600 Km, com níveis verticais variando de 1 aos 2 níveis significativos acima e abaixo do "mini-volume". Os pontos de grade são agrupados em ~12.000 "mini-volumes", para todos pontos de grade em um mesmo "mini-volume" são utilizadas as mesmas observações selecionadas. O resultado será a análise em ponto de grade que será transformada nas variáveis em coeficientes espectrais que formam a condição inicial do modelo global (logaritmo natural da pressão à superfície, temperatura virtual, divergência horizontal, vorticidade vertical e umidades específica). Os dados necessários estão disponíveis no CPTEC que tem rotinas operacionais diárias para sua aquisição. São eles: o modelo T126L28 e dados globais observados do GTS, provenientes do INMET nos horários sinóticos (00,06,12 e 18 UTC); dados de PCD existentes no INPE e dados gerados localmente no CPTEC derivados de imagens de satélite em horários dependentes da passagem dos satélites.

2.3 As estratégias de Paralelização

Utilizando o sistema multiprocessados de memória compartilhada NEC/SX-4, o PSAS foi paralelizado utilizando a tecnologia OpenMP deste sistema. Para se paralelizar um algoritmo em um computador paralelo com memória compartilhada é necessário identificar partes do programa que podem ser executadas concomitantemente, e também explicitamente cuidar da alocação dos dados do problema através da máquina de forma a permitir a exploração do paralelismo existente no algoritmo. Para esta aplicação foi identificado o “gargalo” do procedimento com rodadas seqüenciais com medidas de tempo, o maior custo aparece na rotina com multiplicações matriz-vetor em bloco do algoritmo PSAS. (veja tabela 1). A melhor estratégia foi paralelizar o laço mais externo, onde invoca o pré-condicionador do gradiente conjugado **conjgr1**, ou seja, o laço contém o número de linhas da primeira divisão da matriz em regiões.

ROTINAS	TEMPO REAL (segundos)	m:s	%
ana	1824.7	30:25	100.00%
dyn_get	1.9	00:02	0.10%
ods_get	3.7	00:04	0.20%
Observer	61.4	01:01	3.40%
simulator	10	00:10	0.50%
soqcs	32.1	00:32	1.80%
PSAS_Sigs	12.7	00:13	0.70%
Analyzer	1733.7	28:54:00	95.00%
superObs	2.1	00:02	0.10%
getAI	1691.9	28:12:00	92.70%
Ana2Dyn	3.4	00:03	0.20%
ods_put	11.3	00:11	0.60%
dyn_put	2.7	00:03	0.20%

Tabela 1 - Medida de tempo real (wall time) feita em execução do PSAS seqüencial.

Todavia, problemas de contenção de memória poderiam ocorrer se múltiplos processadores forem programados para atualizar o mesmo bloco do vetor produto das multiplicações matriz-vetor paralelizadas. Reconhecendo que a matriz da equação (7) é assimétrica, e a matriz da equação (8) é simétrica, o cálculo das duas equações é paralelizado com dois esquemas diferentes, ambos esquemas prevêm que dois processadores não atualizem a mesma seção da memória no mesmo segmento paralelizado. A *multiplicação matriz-vetor assimétrico* da equação (7) pôde ser paralelizada pela divisão do vetor de incremento de análise em blocos de acordo com o tipo de variáveis e os perfis verticais e seus elementos. Como a esparsividade da matriz depende de vários fatores, a carga computacional entre as partições paralelas pode não estar balanceada. Um balanço razoável da carga de processamento pode ser obtido para uma computação dinâmica (cíclica). Uma granularidade menor pode ser alcançada também, dividindo a matriz C em blocos menores de acordo com os índices regionais e suas colunas. A equação (8) é resolvida iterativamente e em cada iteração envolve *multiplicações da matriz simétrica* $HP^f H^T + R$ (ou matrizes pré-condicionadas) e o vetor y . As multitarefas da multiplicação é implementada com um rearranjo na ordem nas operações de adição. A matriz $C = HP^f H^T + R$ é simétrica e dividida em blocos $\{C_{ij}\}$ simetricamente correspondente ao bloco do vetor y , que é particionado para $\{y_i\}$ de acordo com os índices regionais e os tipos de variáveis e seus elementos. Todo $\{C_{ij}\}$ tem uma imagem simétrica $C_{ij} = C_{ij}^T$. Como poderiam ser gasto mais computações para gerar novamente a matriz, é altamente desejável computar $C_{ij}^T y_i$ ao mesmo tempo que uma matriz bloco $\{C_{ij}\}$ tenha sido criada para $C_{ij} y_j$. Todavia, a multiplicação é dividida em três multiplicações parciais. Um simples componente da expressão representa uma partição. Um esquema similar é usado na criação de uma segunda área vetorial e a ligação das partições para seus blocos de matrizes transpostas. Com este esquema, nenhuma partição enviará suas saídas para a mesma localização de memória no segmento do programa paralelizado. Na expressão mais à direita há apenas uma partição virtual para ser computada. Todavia, no caso real com a própria ordem regional, a somatória será terminada antes que esta chegue

ao fim da expressão mais a direita, pois a matriz C é normalmente em banda, e seus blocos a esta altura são zeros, assumem a compressão suportada pelas funções de correlação horizontal.

2.4 A arquitetura do computador NEC

O CPTEC possui um supercomputador NEC/SX4-8A com arquitetura paralela vetorial, que consiste num sistema de um nó vetorial com oito processadores com memória compartilhada. A capacidade de memória principal é de 8 Gbytes por nó com taxa de processamento máxima por processador de 2 Gflops, este computador totaliza no máximo 16 Gflops de processamento; a memória cache tem capacidade de 128 Kbytes e o registrador vetorial é de 256. A otimização desta aplicação de assimilação de dados inicialmente tem o objetivo de explorar as peculiaridades da arquitetura aplicando as estratégias de paralelização estudadas e citadas acima, utilizando o desempenho vetorial que o compilador oferece. Futuramente uma análise será feita e poderá ser melhorada a vetorização da aplicação, bem como melhorada sua paralelização para se alcançar uma maior meta de desempenho.

2.4.1 OpenMP e o NEC/SX4

O OpenMP é um modelo de programação paralela que utiliza o compartilhamento de memória e é considerado um padrão para aplicações paralelas de baixa complexidade de programação. O compilador F90/OpenMP do computador NEC/SX4 funciona como um pré-processador e transforma as diretivas OpenMP em diretivas paralelas privativas do NEC e que são mapeadas em um conjunto de micro-tarefas, baseado no padrão de programação *POSIX pthreads*. Tarefas são geradas assim que são criadas as regiões paralelas em OpenMP e são liberadas quando essas regiões são fechadas. O compilador transforma o código e trata cada região paralela como uma subrotina onde contem as diretivas do NEC (figura 4).

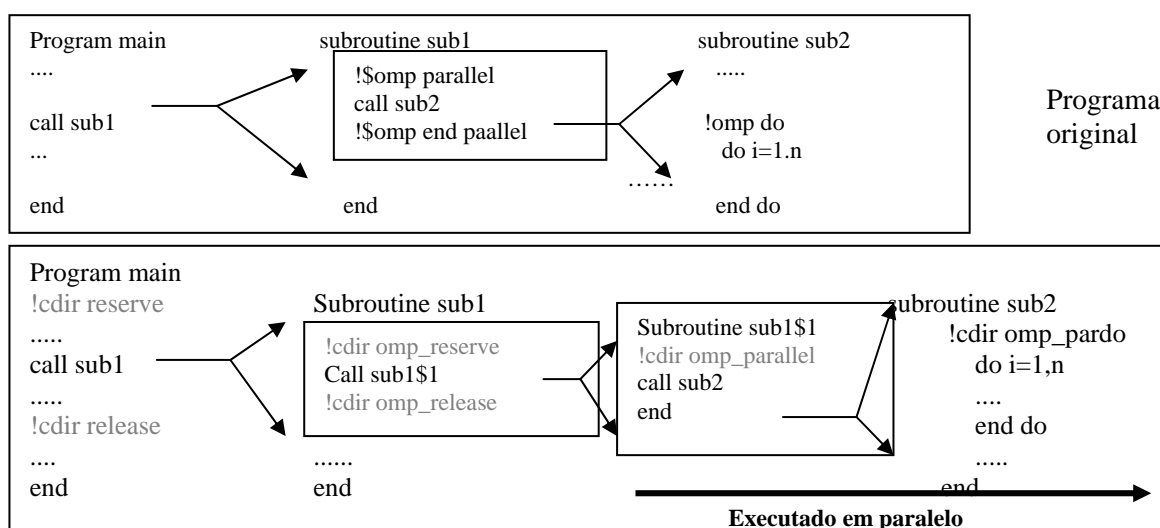


Figura 4 - Transformação feita pelo compilador f90/OpenMP do Nec.

Devido à criação de tarefas a cada região paralela e a reserva de recursos (memória, Processadores, etc) para cada sub-rotina criada, fica ineficiente abrir e fechar regiões paralelas a cada micro-tarefa, o que pode gerar um grande "overhead" de chamadas às bibliotecas de multitarefas OpenMP. Conclui-se que se o número de sub-rotinas criadas pelo compilador e também a alocação de recursos for mínima, a eficiência aumenta para esse tipo de implementação, pois o número de regiões paralelas deve ser minimizado sempre.

3. RESULTADOS PRELIMINARES E PARCIAIS

Utilizamos utilitários para medidas de tempo em tempo real (wall time), mas devido aos testes serem realizados em máquina não dedicada, apresento aqui uma estimativa de ganho de paralelização com o tempo de CPU, ou seja, o tempo em que o programa esteve na máquina. A utilização do OpenMP com esta estratégia de paralelização não foi eficiente, o que necessitará de uma análise mais profunda às dependências de variáveis, a pureza das sub-rotinas, o alto índice de falha de memória cache, etc. Verificou-se também a necessidade prioritária de melhorar a vetorização para realmente obter o desempenho desejável nesta aplicação.

Número de Processadores	1	2	4	6	8
Tempo (seg)	2115.085	1642.467	1399.053	1432.943	1315.053
Eficiência		64.39%	37.79%	24.60%	20.10%
Speedup	1.00	1.29	1.51	1.48	1.61

Tabela 1 - Tempo de CPU (em segundos) para execução do programa de análise do PSAS, a eficiência da paralelização é dada pelo *speedup*.

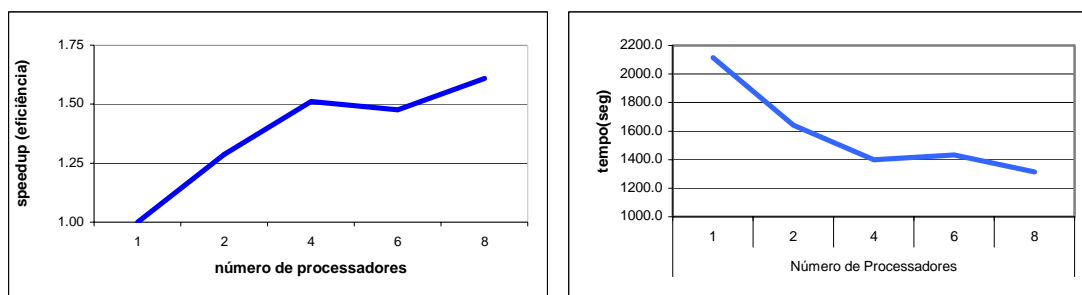


Figura 6 - Curva de eficiência por processadores. e ganho de tempo verificado por numero de processadores

ROTINAS	1 Proc	2 Proc	4 Proc	6 Proc	8 Proc
	m:s	m:s	m:s	m:s	m:s
	%	%	%	%	%
ana	36:13	28:14	23:31	24:05	22:09
ana	100.00	100.00	100.00	100.00	100.00
dyn_get	00:02	00:02	00:02	00:02	00:02
dyn_get	0.10	0.10	0.10	0.10	0.10
ods_get	00:05	00:04	00:04	00:04	00:05
ods_get	0.20	0.30	0.30	0.30	0.40
Observer	00:58	00:58	00:58	00:58	00:58
Observer	2.70	3.40	4.10	4.00	4.40
simulator	00:09	00:10	00:09	00:09	00:09
simulator	0.40	0.60	0.70	0.60	0.70
soqcs	00:31	00:30	00:30	00:30	00:30
soqcs	1.40	1.80	2.20	2.10	2.30
PSAS_Sigs	00:12	00:12	00:12	00:12	00:12
PSAS_Sigs	0.60	0.70	0.90	0.80	0.90
Analyzer	34:46	26:50	22:08	22:43	20:44
Analyzer	96.00	95.10	94.10	94.30	93.60
superObs	00:02	00:02	00:02	00:02	00:02
superObs	0.10	0.10	0.20	0.20	0.20
getAI	34:03	26:08	21:26	22:01	20:02
getAI	94.00	92.60	91.10	91.40	90.50
Ana2Dyn	00:03	00:03	00:03	00:03	00:03
Ana2Dyn	0.20	0.20	0.20	0.20	0.30
ods_put	00:09	00:06	00:06	00:05	00:06
ods_put	0.40	0.40	0.40	0.40	0.40
dyn_put	00:03	00:03	00:03	00:03	00:04
dyn_put	0.10	0.20	0.20	0.20	0.30

Tabela 3 - Tempo real em min./segundos em máquina não dedicada com parcela de porcentagem de tempo gasto por cada rotina

4. CONCLUSÃO

O sistema de análise PSAS que está sendo implantado no CPTEC/INPE é de grande relevância para o centro, trará benefícios diretos na qualidade da previsão de tempo gerada pelo CPTEC, e abrirá novas linhas de pesquisa na área de assimilação de dados. Porém estas vantagens somente serão possíveis se a aplicação executar em tempo hábil, para que esta análise possa ser gerada quantas vezes forem necessárias. Daremos continuidade ao trabalho verificando e modificando a vetorização do programa, adaptando-o à máquina vetorial do CPTEC, a conclusão para esta estratégia usada é que foi considerada paralelização de máquina escalar, a pesquisa continuará em paralelização vetorial para concluir o trabalho de otimização do PSAS, alcançando o desempenho e o resultado desejado. Diferentes situações podem variar este desempenho e todas as situações, caso a caso, devem ser estudadas com o intuito de que as variações não alterem os fatores que aumentam o "overhead" e diminuam a velocidade de processamento, procurando respeitar as limitações da arquitetura da máquina e do software de paralelização. Todo trabalho de otimização somente obterá sucesso se todas as alterações e mudanças ocorridas não alterem o resultado meteorológico da análise objetiva para os modelos de PNT, e este cuidado está sendo verificado e será apresentado em outros trabalhos.

5. REFERÊNCIAS

- [1] Golub, G. H. and C. F. Van Loan, 1989: *Matrix Computations*, 2ª edição. The Johns Hopkins University Press, 642pp.
- [2] Anderson, E., Z. Bai, C. Bisshof, J. Demmel, J. Dongarra, J. da Cruz, A Greenbaun, S. Hammarling, A McKenney, S. Ostrouchov, D. Sorensen, 1995: *LAPACK. User's Guide*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 325pp.
- [3] Chandra, R. Dagum, L., Kohr, D., Maydan, D., McDonald, J., Menon, R., 2001: *Parallel Programming in Openmp*. Academic Press, Morgan Kaufmann Publishers, 230pp.
- [4] da Silva, A , J. Guo, 1996: *Documentation Of Physical-Space Statitital Analysis System (PSAS) - PART I: The Conjugate Gradient Solver*. DAO Offoce Note 96-02, NASA/ Goddard Space Flight Center 66pp.
- [5] da Silva, Arlindo , M. Tippet, J. Guo, 1998: *PSAS Usr's Manual*. DAO Offoce Note 96-02, NASA/ Goddard Space Flight Center, 25pp
- [6] da Silva, Arlindo, J. Guo, 1995: *Computacional Aspects of Goddard's Physycal-Space Staticical Analysis System*. DAO Offoce Note 96-02, NASA/ Goddard Space Flight Center, 8pp