

PARALELIZAÇÃO DA ASSIMILAÇÃO DE DADOS ATMOSFÉRICOS PARA O MODELO GLOBAL DO CPTEC UTILIZANDO OPEN_MP

Rosângela Cintra¹, Dr. Airam J. Preto², Dr. Stephan Stephany², Dr. Haroldo F. de Campos Velho²

¹Centro 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

²Laboratório de Computação e Matemática Aplicada- LAC/INPE
Av. dos Astronautas,1758 – Jd. Da Granja -12227-010–São José dos Campos- SP
e-mail:airam@lac.inpe.br,stephan@lac.inpe.br,haroldo@lac.inpe.br.

Resumo

O Centro de Previsão de Tempo e Estudos Climáticos (CPTEC) está implantando um sistema de assimilação de dados, o *Physical-space Statistical Analysis System (PSAS)* que gerará análises objetivas para o Modelo de Circulação Geral (MCG) Global CPTEC/COLA. Esse sistema de assimilação baseado em teoria da estimação, implica num alto custo computacional para a sua versão atual, devido à complexidade de sua solução. O software foi desenvolvido para máquinas com processadores escalares, dispendo de otimizações para acesso à memória cache. Para viabilizar o uso operacional do PSAS no CPTEC, este foi paralelizado utilizando o padrão OpenMP para o NEC-SX4, uma máquina vetorial de memória compartilhada com 8 processadores. Detalhes do algoritmo, da paralelização e da implementação do OpenMP no NEC-SX4 são abordados e os resultados da paralelização em termos de speed-up/eficiência são expostos, bem como sua análise e considerações sobre trabalhos futuros.

I. 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 de Circulação Geral (MCG) 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; o PSAS possui modelos de erros de covariância avançados. 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 o 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 numéricas locais, utiliza restrições na seleção de dados mesmo com estas simplificações tem o custo computacional alto. Para que o CPTEC tenha este sistema moderno de assimilação de dados, gerando análises globais precisas e detalhadas, operacionalmente viáveis, é necessário otimizar o sistema e obter análises globais em tempo hábil de suprir os modelos de previsão de tempo com seu campo de inicialização. A implementação inicial foi feita verificando o algoritmo com sua complexidade matemática e aplicando a paralelização do código em máquina de memória compartilhada de arquitetura vetorial e paralela, o NEC/SX4, utilizando o padrão OpenMP, procurou-se obter o melhor desempenho operacional, com o ganho da paralelização, foi utilizado vetorização básica fornecida pelo fabricante.

II. METODOLOGIA

A. 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 e resolve as equações iterativamente, remove as restrições de seleção de dados para simplificar a computação. Possui estrutura avançada para testar erros de covariância da previsão com suporte do Filtro de Kalman sub-ótimo.

A.1 O algoritmo - 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 ,

$$w^f = w^t + \mathcal{E}^f \quad (1)$$

onde $\mathcal{E}^f \in \mathfrak{R}^n$ 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,

$$w^o = Hw^t + \mathcal{E}^o \quad (2)$$

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 $\mathcal{E}^o \in \mathfrak{R}^p$ 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 requer $\langle (w^a - w^t)^T S (w^a - w^t) \rangle$ 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,

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

e da matriz de erros de covariância da observação.

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

Ambas são simétricas e positivas semi-definida 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 pxp 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, utiliza o algoritmo pré-condicionado gradiente conjugado (CG) [1] é utilizado como método de escolha para resolver o grande sistema linear, resolve as equações iterativamente com uma série de aproximações físicas e numéricas para simplificar a computação. 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, ventos e razão de mistura do vapor d'água em níveis constantes de pressão.

A.2 Considerações Computacionais

Algumas considerações foram implementadas visando eficiência no algoritmo do PSAS:

- Para introduzir alguma esparsividade na matriz de covariância da inovação e para 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 CG, é uma aproximação, 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.3 Características do algoritmo PSAS

O PSAS foi implementado em Fortran 90 e o *driver* de resolução das suas equações utiliza o método número **Gradiente Conjugado** pré-condicionado [1]. 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 CG pré-condicionados aninhados são ilustrados na Figura 1. 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).

O *driver* principal do PSAS (*Analyzer*), possui uma rotina *getAI* que 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, fonte do dado, latitude, longitude, nível. As rotinas que seguem 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 nível). As coordenadas cartesianas são usadas pelos modelos de covariâncias para calcular as distâncias horizontais. Calcula o índice das sondagens (da Silva and Redder 1995), os índices e pesos da interpolação, 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 *CG* 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.

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. 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 contem os perfis verticais. Um solucionador direto do método de Cholesky é executado nestes blocos usando LAPACK [2].

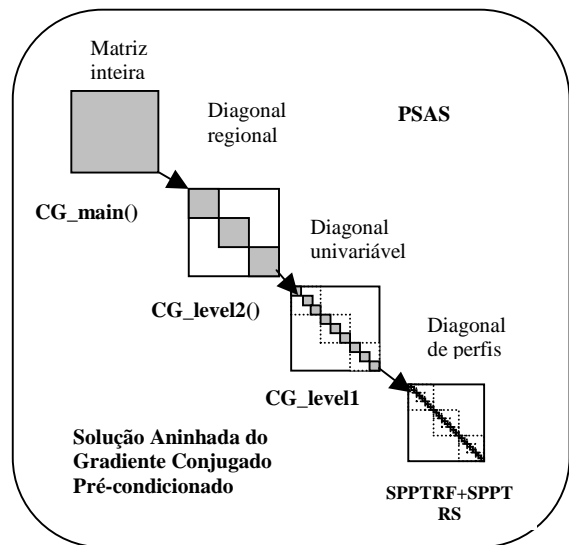


Figura 1 - Solução aninhada usando pré-condicionado gradiente conjugado. A rotina *CG_main()* contém o condutor principal do gradiente conjugado. A rotina é

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.

B. 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 verticais.

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), interpolado para os pontos com observações (O) para as variáveis correspondentes e calculada as diferenças para calibração dos erros usados no cálculo das matrizes de covariância. 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, razão de mistura do vapor d'água e pressão ao nível do mar. 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.

C. A Paralelização

O PSAS foi paralelizado com a tecnologia OpenMP do sistema NEC/SX-4, de arquitetura paralela vetorial que possui um compilador OpenMP Fortran que usa o modelo de *fork-join* na execução paralela. A configuração da máquina do CPTEC/INPE consiste num sistema com 8 processadores e de memória compartilhada. A capacidade de memória principal é de 8 Gbytes e a máxima taxa de processamento por processador é de 2 Gflops. O tamanho dos registradores vetoriais é de 256. As diretivas são uma extensão do modelo de programação sequencial Fortran com uma construção SPMD (*single program multiple data*), permitem sincronização, construção *work-sharing* e suporte também para compartilhamento e privação de

dados. Inicialmente existe um simples processo chamado master thread que é executado sequencialmente até encontrar a primeira construção paralela. Quando isso ocorre é criado um grupo de threads e a master thread passa a ser o processo mestre dessas threads. Existem vários tipos de construções paralelas num programa e meios permitir seu controle e sincronização.

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 sequenciais 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 sequencial.

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.

C.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 *threads*. 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 contém as diretivas do NEC (figura 2).

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.

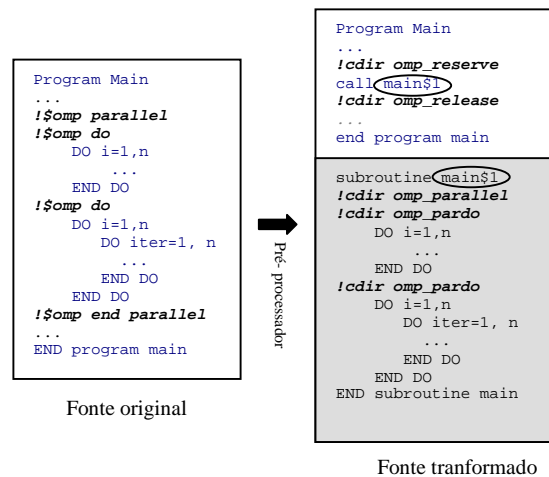


Figura 2 - Pré-processador OpenMP NEC/SX4

III. 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.

Número de Processadores	1	2	4	6	8
Tempo (seg)	2115.08	1642.46	1399.05	1432.94	1315.05
Eficiência		64.39%	37.79%	24.60%	20.10%
Speedup	1.00	1.29	1.51	1.48	1.61

Tabela 2 - Tempo de CPU (em segundos) para execução do programa de análise do PSAS, a eficiência da paralelização é dada pelo *speedup*.

Verificou-se que a eficiência foi baixa nesta paralelização, devido à baixa taxa de vetorização, sabe-se

que o tamanho dos vetores são rearranjados pelo algoritmo, este que foi o enfoque explorado pela paralelização, diminuindo o maior tempo gasto no cálculo das equações; não explorou-se as peculiaridades da máquina vetorial, utilizamos a vetorização básica dada pelo compilador Fortran90 do fabricante, o que indica a necessidade prioritária de melhorar a vetorização para realmente obter o desempenho desejável nesta aplicação.

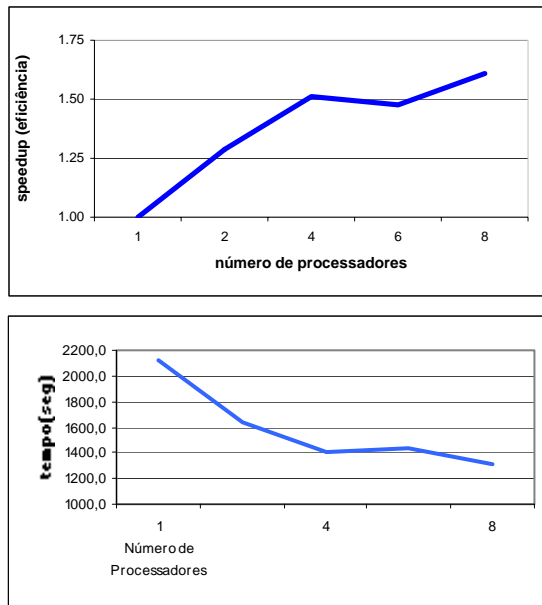


Figura 3 - Curva de eficiência por processadores. e ganho de tempo verificado por número de processadores

IV. CONCLUSÕES

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. Não obtivemos o desempenho desejado e para que a otimização do sistema seja completa daremos continuidade ao trabalho. Faremos a análise de alto desempenho do PSAS com as seguintes pesquisas:

- Utilizar padrão OpenMP em máquinas escalares com memória compartilhada, com a mesma estratégia de paralelização estudada.
- Utilizar padrão MPI em máquinas escalares com memória distribuída, estudando uma nova estratégia.
- Alterar o algoritmo para adaptar os tamanhos dos vetores à máquina vetorial do CPTEC, explorando o paralelismo mantendo a vetorização aplicada, e minimizar o número de regiões paralelas abertas, diminuindo *cache miss rate*.

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.

V. REFERÊNCIAS

- [1] Anderson, E.; Bai, Z.; Bisshof, C.; Demmel, J.; Dongarra, J.; Du Croz, J.; Greenbaun, A.; Hammarling, S.; McKenney, A.; Ostrouchov, S.; Sorensen, D. *LAPACK. User's Guide*, Society for Industrial and Applied Mathematics, Philadelphia, EUA, 1995. 325pp.
- [2] Chandra, R.; Dagum, L.; Kohr, D.; Maydan, D.; McDonald, J.; Menon, R. *Parallel Programming in Openmp*. Academic Press, Morgan Kaufmann Publishers, 2001. 230pp.
- [3] da Silva, A ; Guo, J. *Documentation Of Physical-Space Statitical Analysis System (PSAS) - PART I: The Conjugate Gradient Solver Version PSAS-1.00*. Greenbelt, Maryland: DAO,1995. 66p. (DAO Offoce Note 96-02)
- [4] da Silva, Arlindo; Tippet, M.; Guo, J. *PSAS User's Manual*. Greenbelt, Maryland: DAO,1995. 25p. (DAO Office Note 98-xx)
- [5] da Silva, Arlindo; Guo, J. ; *Computacional Aspects of Goddard's Physycal-Space Statical Analysis System*. Greenbelt, Maryland: DAO,1995. 66p. (DAO Office Note 96-03)
- [6] *Fortran90/SX – OpenMP* - NEC Corporation , 2000,11pp
- [7] Golub, G. H. ; Van Loan, C. F. *Matrix Computations*, v.2,The Johns Hopkins University Press,1989. 642pp.