

Processamento Paralelo Aplicado à Análise de Contingências de Sistemas Elétricos de Potência

Marco Aurélio de Souza Birchal +

Luciano de Errico*

Maria Helena Murta Vale *

+ DCC- Departamento de Ciência da Computação
PUC Minas – Pontifícia Universidade Católica de Minas Gerais
Av. Dom José Gaspar, 500 – Prédio 34 – Coração Eucarístico
CEP. 30535-610 – Belo Horizonte, MG
tel: 55-31-319-4007
e-mail: birchal@pucminas.br

* UFMG- CPDEE – Universidade Federal de Minas Gerais
Av. Antônio Carlos 6 627 – Pampulha
CEP 31270.901 – Belo Horizonte – Brasil
Tel: 55-31-499-4872
e-mail: {mhelena, l.errico}@cpdee.ufmg.br

Resumo

Este trabalho tem por objetivo apresentar os resultados das investigações realizadas pelos autores relativas à implementação da função *Análise de Contingências de Sistemas Elétricos de Potência* utilizando técnicas de alto desempenho. O elevado grau de paralelismo da função e o grande esforço computacional envolvido na sua solução, através de algoritmos sequenciais, motivam a aplicação de processamento paralelo. A versão paralela do algoritmo foi desenvolvida em uma *Máquina Paralela Virtual* apresentando ganhos bastante significativos, abrindo perspectivas para novos desenvolvimentos na área. Para que se obtivesse o melhor desempenho da aplicação, foi elaborado todo um estudo comparativo entre diferentes plataformas usando os *softwares* PVM e HPVM em dois sistemas operacionais. Os resultados relativos a esta análise estão apresentados no artigo.

Palavras-chave: processamento paralelo, alto desempenho, operação de sistemas elétricos, análise de contingências.

1. Introdução

A equipe de Sistemas Elétricos de Potência da UFMG tem investigado a aplicação da computação de alto desempenho na solução de problemas relacionados à análise e ao controle de sistemas elétricos [1,2,3]. Pacotes computacionais paralelos vêm sendo desenvolvidos no *LACAD* (Laboratório de Computação de Alto Desempenho - Engenharia Elétrica UFMG¹) e, também, no *CENAPAD MG/CO* (Centro Nacional de Processamento de Alto Desempenho Minas Gerais/Centro Oeste). Diversas funções de supervisão e controle vêm sendo avaliadas, com respeito à utilização da lógica paralela. Dentre elas destaca-se a função *Análise de Contingências*, tema deste trabalho, devido ao seu elevado grau de paralelismo e ao grande esforço computacional requerido para execução em tempo real.

O uso de técnicas de paralelismo traz novas perspectivas para o desenvolvimento de *software* aplicado à engenharia elétrica. No entanto, muitas vezes, tal desenvolvimento requer, para que seus ganhos sejam efetivos, conhecimentos próprios da área de computação. Nesta direção, a busca do melhor desempenho do aplicativo *Análise de Contingências* envolveu um estudo comparativo entre diferentes plataformas usando os *softwares* PVM e HPVM em dois sistemas operacionais, na montagem de máquinas paralelas virtuais. Assim, a investigação explorou não apenas o tema elétrico, como também trouxe informações relevantes para a própria concepção do ambiente computacional, conforme é mostrado neste artigo.

Neste contexto, o *objetivo* deste artigo é apresentar os resultados do trabalho conjunto de uma equipe multidisciplinar (engenharia elétrica e computação) na investigação da *Análise de Contingências de Sistemas de Energia Elétrica utilizando Máquinas Paralelas Virtuais*. Para cumprir tal objetivo, este texto está estruturado da seguinte maneira: após esta introdução, o problema elétrico envolvido na aplicação é apresentado no item 2; o item 3 descreve o ambiente paralelo de desenvolvimento; o item 4 se dedica à caracterização da estratégia de paralelismo adotada; o item 5 apresenta os resultados das simulações bem como sua análise; finalmente, o item 6 traz as conclusões e perspectivas de continuidade do trabalho.

2. Análise de Segurança de Sistemas de Energia Elétrica

Para que um Sistema Elétrico de Potência (SEP) cumpra a finalidade de atender a demanda *segundo critérios de qualidade, segurança e economia*, é executado um grande espectro de atividades, envolvendo etapas de planejamento e operação [4]. Grande parte das tarefas envolvidas nessas atividades está relacionada a simulações computacionais do comportamento estático e dinâmico do SEP. Tais simulações têm exigido cada vez mais um maior esforço computacional, devido a diversas razões, dentre elas as elevadas dimensões do SEP e a exigência cada vez maior de modelos mais detalhados para os componentes de rede elétrica.

Durante a operação do sistema, são executadas tarefas relacionadas à *Supervisão e ao Controle em tempo real*. Dentre elas, encontram-se as chamadas *Funções de Segurança*, que procuram avaliar a habilidade do sistema elétrico em enfrentar perturbações (perda de linha de transmissão, por exemplo), sem passar para o estado de emergência, estado este que caracteriza um grande problema para o abastecimento de energia elétrica. Esta avaliação da “robustez” do SEP é feita, de forma preventiva, através da chamada “Análise de Contingências”, que consiste na

¹ Laboratório financiado pela FINEP/ RECOPE - Rede Cooperativa de Pesquisa - Subrede Alto Desempenho

simulação de perturbações (contingências) no sistema elétrico e na análise da reação do mesmo frente a cada uma delas. Uma vez conhecido o comportamento do sistema previamente, ações de controle podem ser efetuadas, procurando evitar situações de emergência.

Para que essa análise de segurança do SEP seja efetiva, um *grande número de perturbações diferentes deve ser simulado*. O conjunto de contingências a serem testadas inclui desde perdas simples e múltipla de linha de transmissão até a variação da geração. O esforço computacional exigido para essa análise é extremamente pesado, mesmo para sistemas elétricos de porte médio.

Um outro aspecto de grande importância é que tais simulações devem ser executadas em *tempo real*, isto é, a avaliação da segurança do SEP, de forma preventiva, só tem sentido se feita para cada instante de operação do sistema, ou seja, a intervalos curtos de tempo. Sendo assim, a análise de contingências se caracteriza por *um grande número de simulações de perturbações previamente selecionadas, independentes umas das outras, executadas a curtos intervalos de tempo*.

Os algoritmos de análise de contingências, disponíveis nos Centros de Supervisão e Controle, utilizam processamento seqüencial e as contingências são simuladas seqüencialmente, uma após a outra. Tal tipo de processamento tem se mostrado inadequado às características do problema a ser solucionado. Por outro lado, o processamento paralelo apresenta excelentes perspectivas na busca da diminuição do esforço computacional. No caso específico da Análise de Contingências, as simulações poderiam ser executadas ao mesmo tempo, em diferentes processadores, sem a necessidade de comunicação entre eles durante as simulações individuais.

Da importância da Análise de Contingências para o controle em tempo real do SEP e da possibilidade de sua utilização mais efetiva, através do processamento paralelo, surge a motivação para a elaboração deste projeto.

3. Ambiente Paralelo de Desenvolvimento da Aplicação

Máquinas Paralelas Virtuais

Conforme visto anteriormente, as características da função Análise de Contingências favorecem o uso das chamadas Máquinas Paralelas Virtuais, como ambiente de simulação.

Uma **Máquina Paralela Virtual** constitui-se em um conjunto de máquinas heterogêneas ligadas em rede, de forma a compartilhar seus recursos e se mostrando como uma entidade única para o usuário. Cada computador membro da máquina virtual é chamado de *host*. O usuário tem acesso a todos os *hosts* que participam da máquina virtual de forma transparente, podendo executar aplicações paralelas sobre os mesmos. A tecnologia destas máquinas paralelas virtuais, assim, permite que grupos de computadores convencionais ligados a uma rede local possam ser uma alternativa às máquinas paralelas atuais, nem sempre disponíveis e muitas vezes extremamente especializadas. Além disto, como a arquitetura de redes é cada vez mais comum em todas as áreas, inclusive no setor de energia elétrica, o uso destas máquinas virtuais vem trazer uma ampla perspectiva para a solução de problemas de grande porte, sob a ótica paralela.

A grande capacidade de processamento das máquinas convencionais, aliada a redes locais cada vez mais rápidas, tem permitido que se emule máquinas paralelas em ambientes de rede. Desenvolvedores de *software* para sistemas distribuídos investem cada vez mais nesta possibilidade e o aparecimento de soluções como o PVM (*Parallel Virtual Machine*) [5] e o HPVM (*High-Performance Virtual Machine*) [6] têm permitido a solução de problemas, sob a ótica paralela, de forma bastante eficiente.

No trabalho de pesquisa elaborado pelos autores, referência [7], é apresentado, em detalhes, o estudo comparativo entre duas soluções em *software* para a implementação de máquinas virtuais em redes locais. Foram selecionadas duas implementações representativas, que implementassem o paradigma de passagem de mensagens em redes locais de computadores, e que pudessem ser executadas em dois dos mais difundidos sistemas operacionais para este tipo de arquitetura: o Windows NT e o Linux. Embora existam trabalhos anteriores que visam estabelecer comparações de desempenho entre pacotes para programação paralela em redes locais [8], a pesquisa traz o estudo do comportamento de uma mesma solução em sistemas operacionais distintos, além da comparação de uma implementação clássica, o PVM, com uma implementação recente, o HPVM. Com o PVM (que emprega *sockets* para comunicação entre processos) e o HPVM (que implementa uma camada de rede proprietária chamada *Fast Messages*) foram implementados testes de *benchmark* em linguagem C de forma a medir o desempenho dos mesmos em diversas situações de troca de mensagens entre processos.

Bibliotecas Utilizadas: PVM e HPVM

O PVM (*Parallel Virtual Machine*) [5] é uma API que implementa um dispositivo de passagem de mensagens sobre uma rede heterogênea de computadores, criando uma máquina paralela virtual, com a intenção de se executar programas paralelos sobre um ambiente distribuído. As aplicações do PVM são constituídas de componentes (ou processos) cada qual representando um sub-algoritmo da aplicação. Cada componente é um código SPMD que pode ser multi-instanciado, tanto num mesmo processador quanto em processadores distintos. A troca de mensagens e o sincronismo entre os componentes são obtidos através de chamadas às primitivas de programação do PVM. Se todos os componentes da aplicação são idênticos, tem-se um paralelismo por dados. Se existem componentes distintos que executam funções específicas da aplicação (entrada de dados, processamento e saída de dados), tem-se paralelismo funcional. O PVM suporta ambos modelos de paralelismo, inclusive em conjunto.

O HPVM ou *High Performance Virtual Machines* [6] é um conjunto de APIs desenvolvido na Universidade de Illinois por Andrew Chien, e tem por objetivo fornecer um ambiente de computação de alto desempenho sobre um *cluster* de *workstations*. O HPVM define um conjunto de interfaces de alto e baixo níveis, em forma de camadas, que o torna rápido e portátil. Para alcançar esta proposta, o HPVM define uma API de baixo nível - uma interface de comunicação de rede baseada no conceito de *Active Messages* de *Berkeley* que faz a conexão entre o *hardware* da rede e as APIs de *software* - o FM ou *Fast Messages*. O *Fast Messages* é a principal API do HPVM, sendo responsável por criar uma abstração eficiente do *hardware*, sobre a qual as APIs de passagem de mensagens convencionais de alto nível serão montadas (MPI, *Global Array* ou *shared-memory*).

Plataforma de Desenvolvimento

A plataforma de *hardware* utilizada para os testes de avaliação foi composta por computadores Pentium e placas de rede *Ethernet*. Os sistemas operacionais utilizados foram o Windows NT Workstation 4.0 (com memória virtual de 50 a 150 Mbytes) e o Linux, versão de *kernel* 2.0.34 (com *swap* de 64 Mbytes). Os compiladores utilizados foram o Microsoft Visual C++ 5.0 para Windows NT e o gcc para Linux.

Benchmarks Utilizados nos Estudos de Desempenho

Os estudos para avaliação do desempenho das plataformas de processamento paralelo se basearam nos seguintes parâmetros de interesse: a latência e o *throughput*. A latência de comunicação é definida como o tempo total gasto por um processo no envio ou no recebimento de uma mensagem. O *throughput* (também denominado *bandwidth* ou taxa efetiva) é a taxa média de comunicação, geralmente em bits por segundo, vista pela aplicação que se comunica através do recurso de comunicação disponível (por exemplo uma rede local).

Nos próximos itens são apresentados os resultados obtidos, até o momento, com relação à aplicação do paralelismo à função Análise de Contingências. Resultados relativos a outras situações podem ser encontrados em [7].

4. Análise de Contingências Sob o Enfoque Paralelo

Conforme dito anteriormente, a função Análise de Contingências em Tempo Real caracteriza-se pela execução de um elevado número de simulações independentes em pequeno intervalo de tempo. Na análise estática, apropriada para a avaliação da segurança em tempo real, cada contingência corresponde à execução de um cálculo de Fluxo de Potência.

Analisando o grau de paralelismo desta tarefa, verifica-se que a exploração do paralelismo pode se dar em diversos níveis: na execução simultânea de contingências ou no cálculo interno de cada uma delas (paralelismo das potências ativa e reativa ou até mesmo na solução do sistema linear presente no Método de Newton-Raphson do fluxo de potência).

Com relação ao paralelismo interno ao cálculo do fluxo de potência, este vem sendo investigado. Porém, o paralelismo ao nível das contingências tem apresentado, até o momento, melhores perspectivas para o aumento do desempenho.

Algoritmo Paralelo Desenvolvido

A filosofia adotada para a implementação do algoritmo paralelo consiste na elaboração de um *programa mestre* que dispara um número de *processos escravos* que executam, cada qual, uma contingência do conjunto selecionado a ser estudado. Ao final da execução, o escravo escreve um arquivo de saída com o resultado da análise.

O fluxograma apresentado na figura 1 detalha os mecanismos de entrada e saída de dados, bem como os de passagem de mensagens. O programa mestre lê um arquivo que possui os dados iniciais (número de contingências a serem simuladas e quantidade de escravos a serem disparados) e os parâmetros das contingências (tipo e localização), que são lidos seqüencialmente, à medida que existam escravos disponíveis. Este arquivo deve estar presente na mesma máquina em que se encontra o mestre.

Cada escravo lê o arquivo de configuração do sistema elétrico, que reflete as características do mesmo, sobre o qual se está trabalhando, e gera um novo arquivo temporário, produto da modificação causada pela contingência. Este arquivo é sobrescrito a cada nova contingência calculada. Terminado o cálculo, o escravo grava o resultado em um arquivo de saída. Deve haver um arquivo de configuração do sistema em cada máquina em que se execute um escravo.

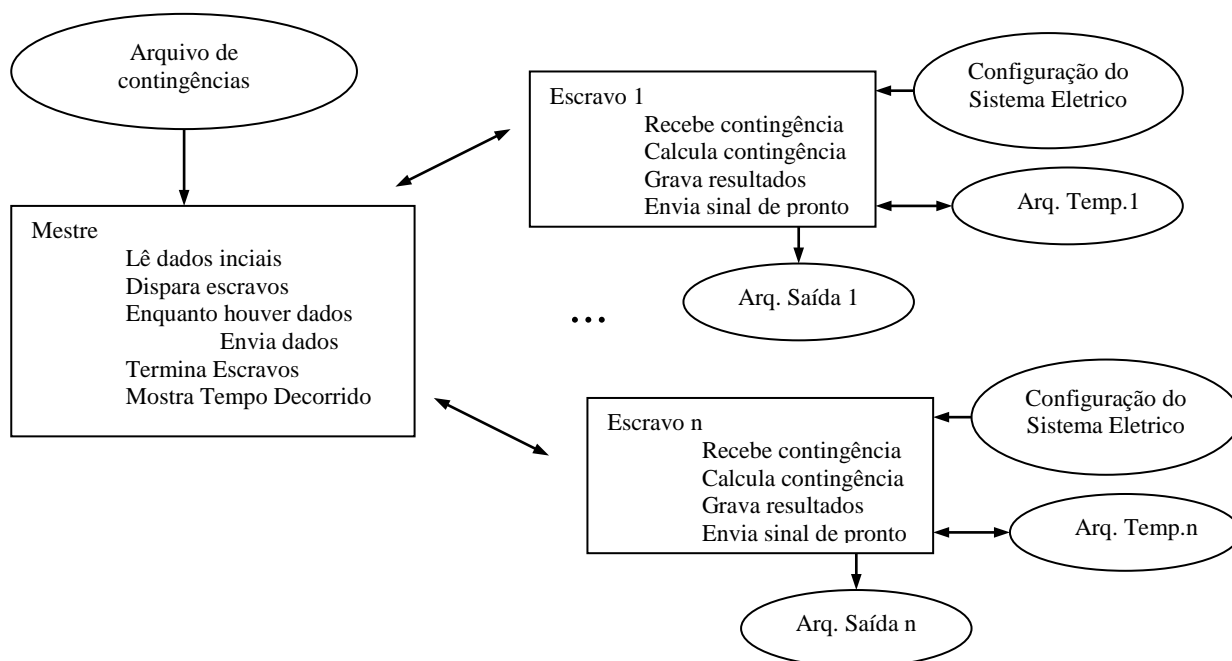


Figura 1 - Diagrama da Abordagem Paralela Adotada

Foram incluídas no programa mestre – o qual é responsável pela distribuição das contingências entre os escravos, podendo ainda ser concorrente aos mesmos – duas rotinas de medição de tempo. A rotina principal mede o tempo total gasto pela aplicação para o cálculo de n contingências. Uma segunda rotina mede o tempo gasto com envio de mensagens. Este é o tempo de utilização de rede pelo sistema.

A rotina de envio (implementada no programa mestre) e a rotina de recebimento de mensagens (implementada no programa escravo) podem ser executadas um número $\#num_de_vezes$ para cada mensagem enviada (ou recebida).

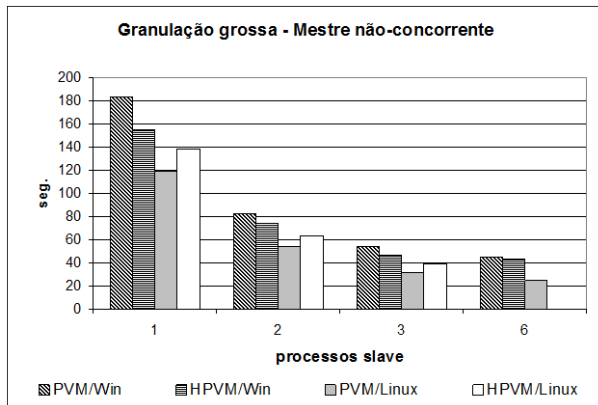
Quando $\#num_de_vezes$ vale 1, tem-se o sistema executado da forma convencional, onde cada valor de contingência é entregue somente uma vez ao escravo e este a computa. Quando $\#num_de_vezes$ vale 5, o *loop* de envio/recebimento é executado 5 vezes para cada contingência.

Tais análises são denominadas *granulação grossa* e *granulação fina*, respectivamente, por simularem processos de programação paralela em ambiente distribuído de tais granulações.

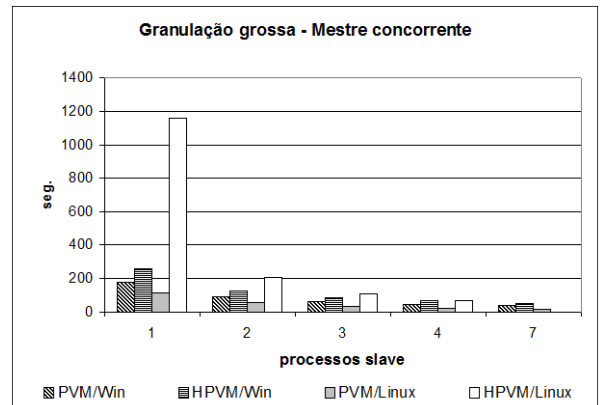
5. Resultados das Simulações

Os resultados obtidos mostram o tempo gasto para os casos de granulação grossa, utilizando-se 3000 contingências, e de granulação fina, com 2000 contingências, com relação ao número de processos escravos executados em paralelo. Para cada uma dessas situações, analisaram-se, também, as possibilidades de mestre concorrente e não-concorrente, simuladas em pares API/SO distintos.

Através da análise dos gráficos da figura 2, que representam as respostas das APIs em granulação grossa, percebe-se que o HPVM apresenta melhores desempenhos em Windows NT que em Linux. Chegando mesmo a não se obterem resultados para HPVM/Linux quando o número de processos aumenta.



(a)

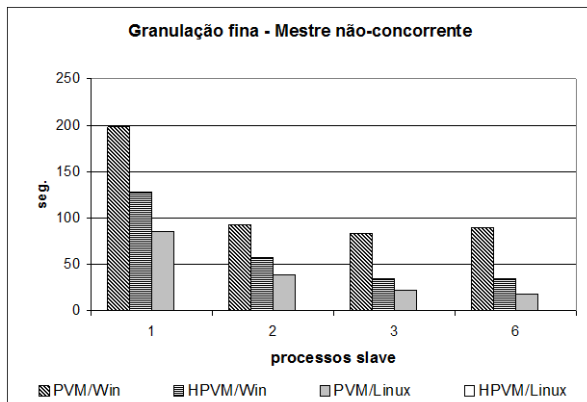


(b)

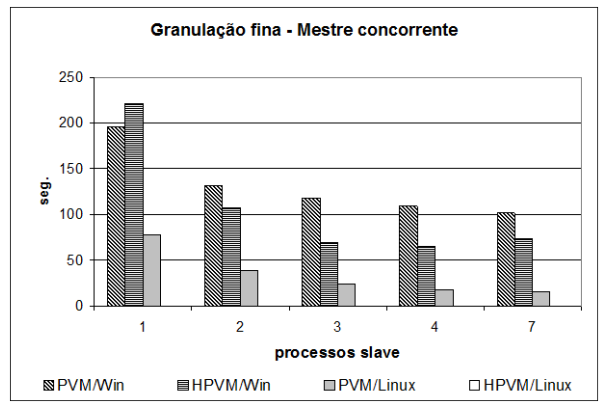
Figura 2 - Granulação grossa

Para granulação fina, representada nos gráficos da figura 3, observa-se que o HPVM/Linux não conseguiu concluir nenhum dos testes.

O PVM/Linux se mostrou como a escolha de melhor desempenho com relação ao tempo de processamento. O PVM se mostrou estável em todas as combinações implementadas, em ambos sistemas operacionais.



(a)



(b)

Figura 3 - Granulação fina

As investigações mostram diversos pontos interessantes, nos quais se baseiam as conclusões sobre a plataforma mais adequada para o aumento do desempenho das funções de Controle de Segurança de SEP.

Com relação aos valores numéricos de latência e *throughput*, para todos os conjuntos Biblioteca/Sistema Operacional: o HPVM mostrou desempenho inferior ao do PVM todas as vezes em que existem processos escravos concorrentes ao processo mestre, além de apresentar instabilidade com relação ao controle de envio e recebimento de mensagens, principalmente no sistema operacional Linux.

As investigações, até o momento, indicam que o HPVM pode oferecer, em algumas situações, menores latências e maiores velocidades de resposta que o PVM, notadamente quando não trabalha num regime de mestre concorrente em ambiente Windows. Em contrapartida, o HPVM apresentou instabilidade não suportando altas taxas de comunicação de dados (altos *throughputs*) com confiabilidade, podendo vir a gerar erros de processamento.

6. Conclusões e Perspectivas de Continuidade do Trabalho

O trabalho aqui apresentado significa um avanço às pesquisas relativas à aplicação de processamento paralelo na solução dos problemas da área de Sistemas de Energia Elétrica. Isto se deve, não apenas aos ganhos computacionais obtidos com a implementação efetiva da função Análise de Contingências, mas também à perspectiva de desenvolvimento de novas aplicações.

Em termos de continuidade dos estudos elétricos, vê-se que a abordagem de paralelismo adotada neste trabalho pode ser aplicada a outras funções, tais como a Seleção de Contingências, Cálculo de Queda de Tensão em Redes de Distribuição, Projeto de Aterramentos, dentre diversas outras. De imediato, já vem sendo desenvolvida a função Seleção de Contingências. Um outro aspecto a considerar é o estudo de sistemas elétricos com diferentes dimensões. Acredita-se que os ganhos serão ainda mais significativos.

Com relação às investigações em termos de desempenho de ambientes paralelos, os autores têm atuado no *CENAPAD – MG/CO* executando programas paralelos na máquina IBM RS-6000/SP com 48 processadores [9]. Em breve, estarão sendo divulgados resultados relativos às simulações efetuadas na máquina paralela deste centro e nas máquinas paralelas virtuais do *LACAD*.

Bibliografia

- [1] VALE, M.H.M, ARRUDA, L.N., MOREIRA, A.C., Aplicação de Processamento de Alto Desempenho em Sistemas Elétricos, Belo Horizonte, UFMG, 1999. (Inédito).
- [2] VALE, M.H.M., FALCÃO, D.M., KASZKUREWICZ, E., Electrical Power Network Decomposition for Parallel Computations, IEEE International Symposium on Circuits and Systems. vol. 6, pp. 2761-2764, San Diego, CA, USA, Maio de 1992.
- [3] VALE, M.H.M., FALCÃO, D.M., KASZKUREWICZ, E., A Network Decomposition Method for Block Iterative Solutions on Parallel Computers, aprovado para publicação no Electric Power System Research.
- [4] VALE, M.H.V., Centros Modernos de Supervisão e Controle de Sistemas de Energia Elétrica, Tese de Mestrado, COPPE-UFRJ, Rio de Janeiro, 1986.
- [5] GEIST, G. A., BEGUELIN, A., DONGARRA, J., JIANG, W., MANCHEK, R., SUNDERAM, V. PVM: Parallel Virtual Machine A User's Guide and Tutorial for Networked Computing. Cambridge: MIT Press, 1994.
- [6] PAKIN, S., BUCHANAM, M., CONNELLY, K. LAVERY, A., KOENIG, G., GIANNINI, L., PRUSAKOVA, J., HERMANN, G., CHIEN, A. HPVM 1.0 User Documentation, Urbana, 1997.
- [7] BIRCHAL, M. S., Uma Comparação entre duas Plataformas para Processamento Paralelo em Redes Locais, Dissertação de Mestrado, CPDEE – UFMG, 1999.

[8] RÍMOLO, G.S., Análise Comparativa de Duas Plataformas de Programação Paralela para Redes de Workstations, Belo Horizonte, 1996.

[9] <http://www.cenapad.ufmg.br>