

Boletim Técnico da Escola Politécnica da USP
Departamento de Engenharia de Computação e
Sistemas Digitais

ISSN 1413-215X

BT/PCS/0225

Avaliação do Desempenho de
Aplicações Distribuídas sob Duas
Velocidades de Rede

Amilcar Rosa Pereira
Geraldo Lino de Campos

São Paulo - 2002

1298790

O presente trabalho é um resumo da dissertação de mestrado apresentada por Amílcar Rosa Pereira, sob orientação do Prof. Dr. Geraldo Lino de Campos: "Uma Avaliação de Desempenho de Sistemas Distribuídos sobre Redes de Alta Velocidade" defendida em 03/09/2002, na EPUSP.

A íntegra da dissertação encontra-se à disposição com o autor e na biblioteca de Engenharia de Eletricidade da Escola Politécnica da USP.

FICHA CATALOGRÁFICA

Pereira, Amílcar Rosa

Avaliação do desempenho de aplicações distribuídas sob duas velocidades de rede / A.R. Pereira, G.L. de Campos. – São Paulo : EPUSP, 2002.

12 p. – (Boletim Técnico da Escola Politécnica da USP, Departamento de Engenharia de Computação e Sistemas Digitais, BT/PCS/0225)

1. Sistemas distribuídos (Sistema operacional) 2. Redes de computadores I. Campos, Geraldo Lino de II. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais III. Título IV. Série

ISSN 1413-215X

CDD 005.4476

004.6

Avaliação do desempenho de aplicações distribuídas sob duas velocidades de rede.

Amilcar Rosa Pereira¹, Geraldo Lino de Campos

amilcar@poli.usp.br

geraldo@linodecampos.net

Departamento de Computação e Sistemas Digitais

Escola Politécnica de São Paulo

RESUMO

Este trabalho avalia o comportamento de um sistema operacional distribuído quando este sofre alteração da velocidade de sua rede de interconexão. Com a evolução das redes locais para velocidades de até 1 Gbps surge a dúvida: qual a vantagem que aplicações distribuídas realmente obtém com o aumento da velocidade da rede do sistema? A análise aqui realizada compara o desempenho de um sistema distribuído operando sob uma rede Fast Ethernet e sob uma rede Ethernet com a finalidade de responder a esta questão.

Os dados aqui apresentados foram obtidos através do uso do sistema operacional distribuído Amoeba e da utilização de suas primitivas de comunicação baseadas no protocolo de rede FLIP. Os números mostram que a simples evolução da tecnologia de rede, sem um trabalho paralelo no protocolo de rede, não implica numa melhora significativa da performance das aplicações distribuídas.

ABSTRACT

This work evaluates the behavior of a distributed operating system by changing its internetworking speed. With the evolution of the local network technology to speeds up to 1 Gbps the following question arises: Do distributed applications perform better using faster networks? The analysis presented here compares the performance of a distributed operating system working with a Fast Ethernet network and an Ethernet network to answer this question.

We have got the numbers presented here by using the Amoeba distributed operating system and its communications primitives based on the FLIP network protocol. The results show that the simple improvement of the network speed, without a parallel work in the network protocol, does not lead to a significant enhancement in the distributed applications performance.

1. Introdução

Nos últimos anos, uma nova tendência tem sido notada no mercado de computadores. A relação preço/performance das estações de trabalhos e computadores pessoais tem se tornado bem menor que a relação dos computadores de grande porte. A razão para esta

¹ Suporte Financeiro da CAPES

redução nos custos dos computadores de pequeno porte é razoavelmente simples. Em qualquer produto manufaturado os custos de desenvolvimento são divididos pelo volume produzido. O preço/performance dos computadores está melhorando 80% ao ano, enquanto que o preço performance dos supercomputadores está melhorando apenas de 20% a 30% ao ano (Anderson; Culler; Patterson, 1997).

Os sistemas baseados em cluster são uma alternativa interessante para aproveitar esta tendência, uma vez que a velocidade das redes locais tem conseguido atingir um patamar muito próximo a velocidade dos barramentos internos, principalmente com as redes Gigabit. Por exemplo, um barramento com tamanho de palavra de 4 bytes, funcionando a 66 Mhz, atinge uma taxa de transferência máxima de 250 Mbytes/s. Uma rede funcionando a 1 Gbit/s consegue atingir uma taxa de transferência máxima de 128 Mbytes/s. A velocidade do barramento interno é portanto apenas duas vezes maior que a velocidade da rede local. Com a próxima padronização de rede a 10 Gbit/s, este número ultrapassará 1000 Mbytes/s. Assim um sistema baseado em rede local pode atingir um desempenho muito próximo ao desempenho de um sistema semelhante baseado em barramentos proprietários, com a vantagem de ser bem mais barato.

Com este cenário formado surge uma dúvida. Aumentar a velocidade da rede de interconexão de um sistema distribuído realmente fornece uma melhora em seu desempenho ao resolver problemas e aplicações reais? É baseado nesta pergunta que se define o principal objetivo deste estudo, avaliar como se comporta um sistema distribuído depois de ter a velocidade de sua rede aumentada.

Algumas escolhas foram feitas com o objetivo de se definir o ambiente no qual se realizará este estudo. A partir do cluster montado em nosso laboratório formado por doze microcomputadores AMD k6 233Mhz², com 64 Mb de memória cada um, interligados por interfaces Fast Ethernet, através de um HUB 3Com, percebemos que um excelente sistema distribuído para avaliação de nossa experiência seria o sistema Amoeba (Tanenbaum et.al, 1991), que possui licença livre e código fonte aberto para Universidades. O Amoeba se encaixa perfeitamente em nosso cluster, uma vez que ele também foi desenvolvido para plataforma x86.

2. Sistema Operacional Amoeba

O projeto do sistema operacional Amoeba (Renesse; Staveren; Tanenbaum, 1989); (Tanenbaum et.al, 1990) e (Tanenbaum et.al, 1991) teve como foco principal a construção de um sistema totalmente distribuído. Procurou-se criar um Sistema Operacional em que as aplicações ficam espalhadas pelas diversas máquinas do sistema, mas o usuário tem a impressão de estar utilizando um poderoso sistema de multiprocessamento centralizado.

O software do Amoeba está dividido basicamente em duas partes: um *kernel*, que roda em todas as máquinas do sistema, e uma coleção de servidores especializados que fornecem a maioria das funcionalidades comuns a todos os sistemas operacionais. O *kernel* do Amoeba possui apenas as funcionalidades básicas: entrada e saída simples, gerenciamento de memória, controle de processos e mecanismos de troca de mensagens entre processos. Por esta razão esse tipo de *kernel* é chamado de *microkernel*.

Outros serviços comuns à maioria dos outros sistemas operacionais são disponibilizados no Amoeba através de servidores específicos espalhados pelos diversos nós do sistema. Dentre estes inúmeros servidores destacaríamos como mais importantes os servidores de arquivo, servidores de diretório, servidores de data, e os servidores de tcp/ip.

² Equipamentos financiados pela Finep-Brasil

Na Figura 1 podemos identificar os elementos principais que formam a arquitetura básica do Amoeba. Todo o poder de processamento do sistema fica concentrado no pool de processadores, disponível a todos os usuários do sistema. O acesso ao sistema é fornecido através das estações de trabalho. A finalidade desta organização é aproveitar ao máximo a capacidade dos processadores do sistema.

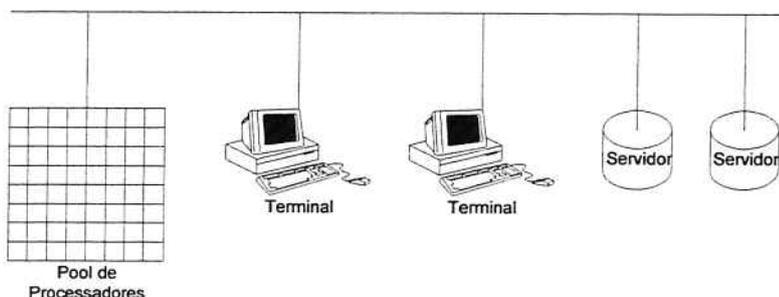


Figura 1 – Organização do Amoeba

O Amoeba é baseado em um protocolo de rede chamado FLIP (Kaashoek et.al, 1993). Este protocolo foi desenvolvido em conjunto com o sistema operacional Amoeba. O FLIP é um protocolo de rede sem conexão, mas com um aumento de funcionalidade para suportar chamadas remotas de procedimento (RPC), ao invés da transferência de uma simples seqüência de bytes (como o protocolo TCP/IP). Além de primitivas RPC, o Amoeba oferece um protocolo de comunicação de grupo para auxiliar no desenvolvimento de aplicações distribuídas que necessitam de primitivas de *broadcast*.

3. Aplicações

Nesta seção serão apresentados os desempenhos de algumas aplicações distribuídas para avaliar o comportamento das mesmas quando submetidas a uma alteração na velocidade da rede física. Foram escolhidos três problemas conhecidos cujas soluções baseadas em algoritmos paralelos são utilizadas na análise de desempenho de sistemas baseados em cluster. Os problemas escolhidos são: multiplicação de matrizes, ASP “All-Pairs Shortest Paths” e TSP (Problema do Caixeiro Viajante) (Wikipedia, 2002).

3.1. Multiplicação de Matrizes

A paralelização do algoritmo seqüencial que efetua a multiplicação de duas matrizes A e B é conhecida como um dos exemplos mais perfeitos de distribuição de tarefas entre N processadores. A operação de multiplicação de duas matrizes é executada através da fórmula abaixo:

$$A \times B = C \mid C_{ij} = \sum_{k=1}^n A_{ik} \times B_{kj}$$

Na implementação que foi feita p processadores são utilizados. Um dos processadores é utilizado para coordenar a operação. Inicialmente o processador mestre divide a matriz C em $p-1$ conjunto de linhas e designa cada um destes conjuntos para cada um dos outros processadores. Cada um dos conjuntos contém uma seqüência de linhas da matriz C . O processador mestre contém também as matrizes de entrada A e B , e no fim da computação deverá conter a matriz resultante C .

Quando um dos processadores recebe um conjunto de linhas de C para calcular, realizando uma série de RPCs para o processador mestre com a finalidade de obter as linhas de A necessárias para a operação e todas as linhas da matriz B. As linhas de A necessárias para a operação são exatamente as linhas de C que deverão ser calculadas. Após calcular todos os elementos de C sob sua responsabilidade, o processador devolve estes elementos através de outra seqüência de RPCs para o processador mestre.

No gráfico presente na Figura 2 é possível verificar o *speedup* desta operação para duas matrizes quadradas de 1500 elementos nas duas configurações de rede sob análise neste trabalho. É importante notar no gráfico que o conceito de *speedup* sofreu uma pequena distorção nesta análise, pois o processador que está coordenando a computação não foi considerado para simular o *speedup* perfeito. Nos dados apresentados se nota que a linearidade do *speedup* para a rede operando a 10 Mbps é bem menor que a 100 Mbps. Isto indica que a aplicação consegue tirar maior proveito do aumento do número de processadores com a rede mais rápida, conseguindo manter um crescimento do *speedup* mais constante. O desempenho desta aplicação foi medido utilizando o relógio interno da máquina onde o coordenador executou.

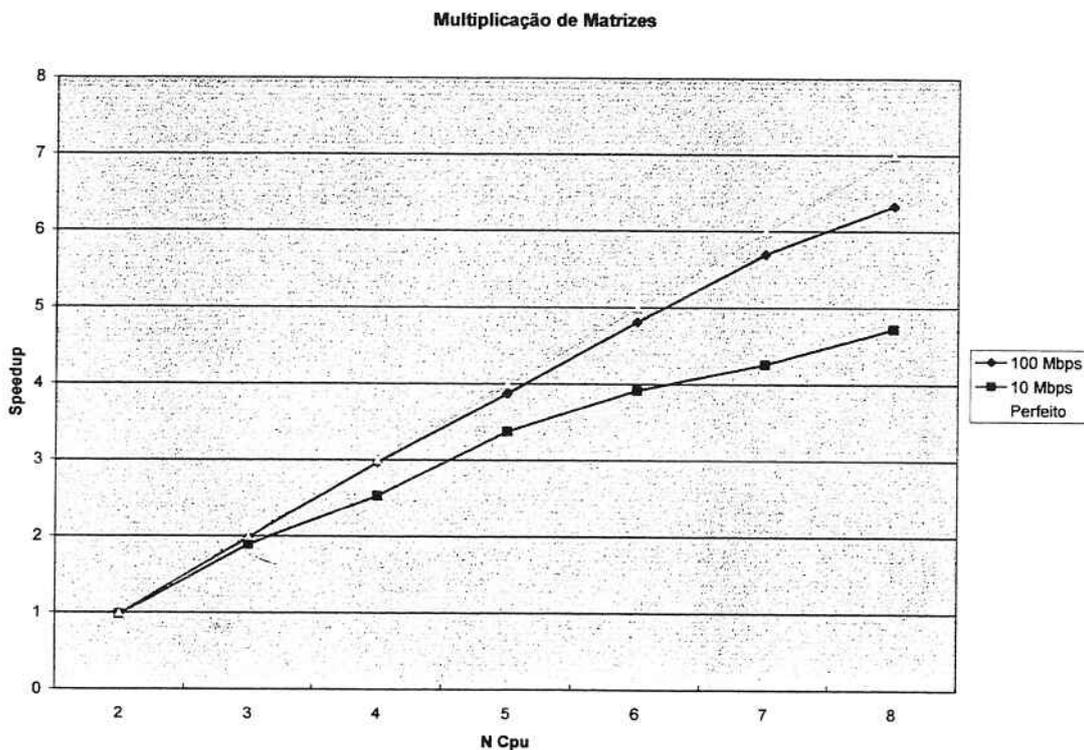


Figura 2 – Speedup multiplicação de matrizes

3.2. TSP

O problema do caixeiro viajante é um dos problemas mais antigos da ciência da computação. Este problema é um dos clássicos da teoria da complexidade de algoritmos e já foi provado ser um problema da classe NP-Completo (não pode ser decidido deterministicamente em tempo polinomial).

O problema pode ser colocado da seguinte maneira: dado um conjunto N de cidades e o custo de viajar de uma cidade para outra, qual o menor caminho necessário para passar por

todas as cidades? É fácil perceber que uma solução ótima para este problema envolve a procura de praticamente todas as rotas possíveis, e isto pode levar a um algoritmo de complexidade $N!$. Assim, para um número pequeno de cidades, a quantidade de passos para achar uma solução ótima para este problema pode se tornar muito grande. Desta maneira os algoritmos que são utilizados para resolver este problema baseiam-se em soluções heurísticas.

Neste projeto utilizou-se a técnica de ramificar e limitar o problema (“branch-and-bound”). Esta técnica consiste em gerar um conjunto inicial de rotas com as n primeiras cidades. Para cada um destes ramos é feito um cálculo heurístico de um limite mínimo para o resto da rota, com a influência neste limite de cada cidade que ainda não está na rota. Em seguida cada uma destas rotas passa por um processo recursivo de busca de uma solução ótima, onde cada recursão verifica se a rota calculada até o momento mais o limite mínimo é melhor que a menor solução já encontrada. O cálculo deste limite mínimo é portanto fundamental para o algoritmo, pois através dele é possível reduzir o número de recursões realizadas para cada sub-rota.

Na implementação do algoritmo, um processador mestre gera todas as sub-rotas iniciais, colando-as numa fila de tarefas. Cada processador escravo realiza chamadas RPC para o processador mestre com a finalidade de retirar as tarefas da fila. Sempre que um processador escravo encontra uma rota completa mais curta baseado no algoritmo descrito no parágrafo anterior, ele efetua uma chamada RPC para o processador mestre informando a nova rota. Para os testes de desempenho apresentados Figura 3, foi usado um grafo simétrico de 22 cidades com as subrotas sendo geradas pelo processador mestre com 4 cidades inicialmente.

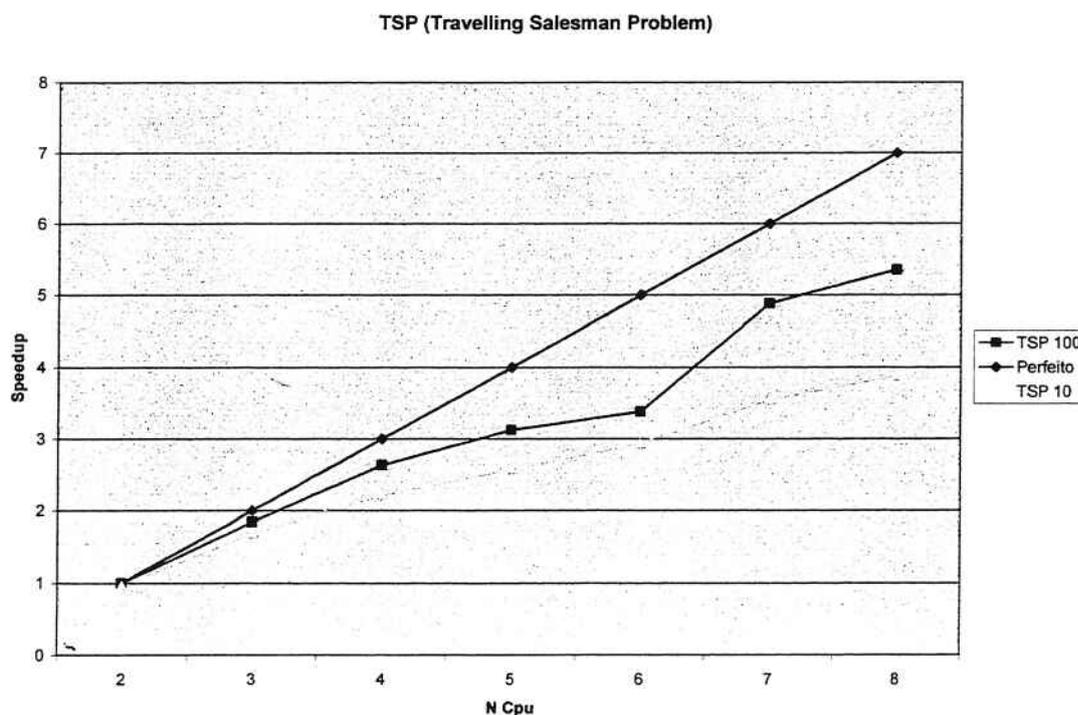


Figura 3 – Speedup TSP

Esta é uma aplicação que trabalha trafegando uma quantidade grande de mensagens pequenas. Como na multiplicação de matrizes, o aumento da velocidade da rede permite um *speedup* real mais próximo do *speedup* perfeito com o aumento do número de processadores. A sensação que se tem ao observar o gráfico é que com 8 processadores, o sistema está muito

perto de seu limite com a rede funcionando a 10 Mbps. A rede funcionando a 100 Mbps demonstra um fôlego maior para o acréscimo de novos processadores no sistema.

3.3. ASP

A terceira aplicação distribuída apresentada neste capítulo consiste de um algoritmo paralelo para resolver o problema de encontrar todos os menores caminhos entre dois vértices de um grafo. Neste problema o grafo G é mapeado para uma matriz de distâncias, com a distância do vértice V_i para o vértice V_k armazenada no elemento G_{ik} da matriz de distâncias.

O algoritmo utilizado para solucionar este problema é uma versão paralela do algoritmo de Floyd. Neste algoritmo, cada processador computa parte da matriz resultante com a menor distância de cada par de vértices do grafo. A cada iteração do algoritmo todos os processadores devem receber a linha da matriz que foi calculada naquela iteração. Isto resulta em uma quantidade grande de sincronização e de troca de mensagens entre os processadores.

Inicialmente o grafo G é particionado entre todos os processadores, da mesma maneira que no algoritmo de multiplicação de matrizes indicado anteriormente. Um processador é eleito para gerar o grafo G e distribuí-lo para todos os outros processadores. Ao final do processamento este mesmo processador deverá conter a matriz de distâncias resultante. Numa primeira versão deste algoritmo foi usado o mecanismo de RPC para realizar todas as trocas de mensagem. Quando cada processador inicia, ele realiza uma série de RPCs para o processador principal requisitando todas as linhas do grafo G . Após a distribuição de G , todos os processadores iniciam o algoritmo, e ao início de cada iteração do mesmo, o processador responsável pela linha daquela iteração (definida no particionamento do grafo), repassa a mesma para todos os outros processadores, que em seguida utilizam esta linha para atualizar todas as linhas sob sua responsabilidade.

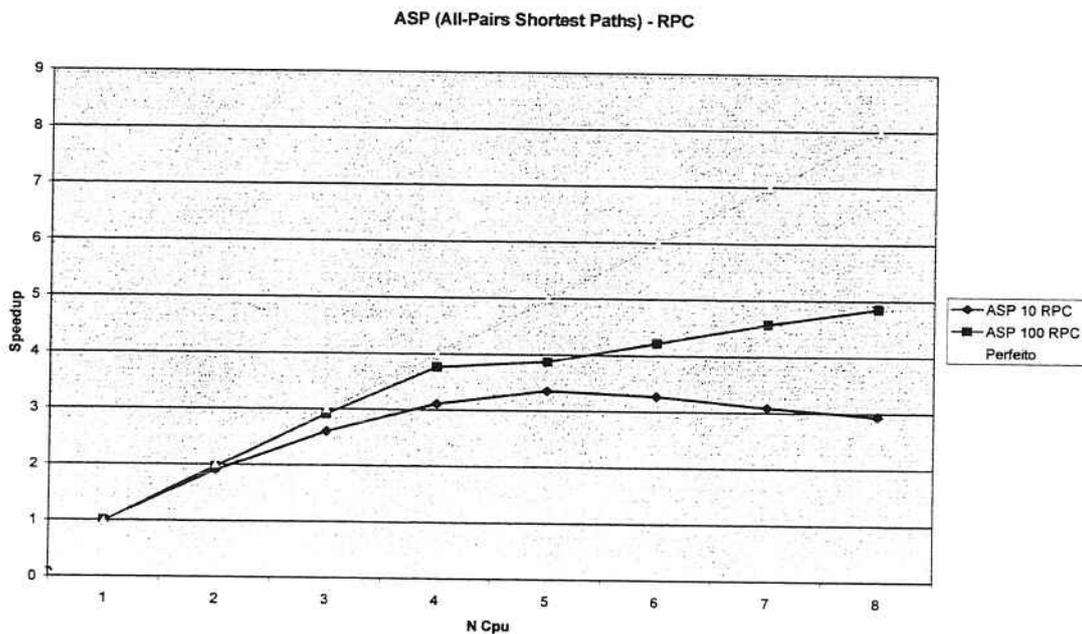


Figura 4 – Speedup ASP RPC

No gráfico presente na Figura 4 seguem os resultados de performance deste algoritmo para um grafo de 2000 cidades. A quantidade de comunicação deste algoritmo é muito grande (8000 bytes para cada linha transmitida), e a carga de ocupação da rede é muito grande. Isto pode ser notado principalmente para a rede funcionando a 10 Mbps, que após cinco processadores começa a ter uma inversão na curva do *speedup*. Isto significa que a capacidade

da rede atingiu seu pico. O *speedup* para 100 Mbps não é perfeito porém o crescimento do *speedup* não parou até a quantidade de 8 processadores.

Para efetuar uma segunda análise deste algoritmo na rede do Amoeba, foi feita uma outra implementação do mesmo utilizando as primitivas de comunicação de grupo. Nesta nova versão, um grupo é criado com todos os processadores sendo membros do mesmo. Inicialmente o processador principal envia mensagens para o grupo com a finalidade de distribuir o grafo G.

Cumprida esta etapa todos os processadores (inclusive o principal) iniciam a computação da matriz de distância. Ao início de cada iteração, o processador responsável pela linha daquela iteração envia a mesma para o grupo. Desta maneira a sincronização entre os membros fica simplificada porque o protocolo de comunicação de grupo garante a ordem entre mensagens.

Na Figura 5 se encontra o gráfico com a performance do algoritmo nas duas redes em estudo neste trabalho. Nota-se que a performance fica linear para ambas as velocidades de rede, e as diferenças que aparecem na versão RPC ficam bem mais reduzidas. Isto em parte se deve a problemas com o controle de congestão existente no protocolo FLIP para mensagens de *multicast*. Este sistema de controle de congestão foi feito para uma velocidade de rede de 10 Mbps, e não se comporta adequadamente com a alteração da velocidade da rede para 100 Mbps.

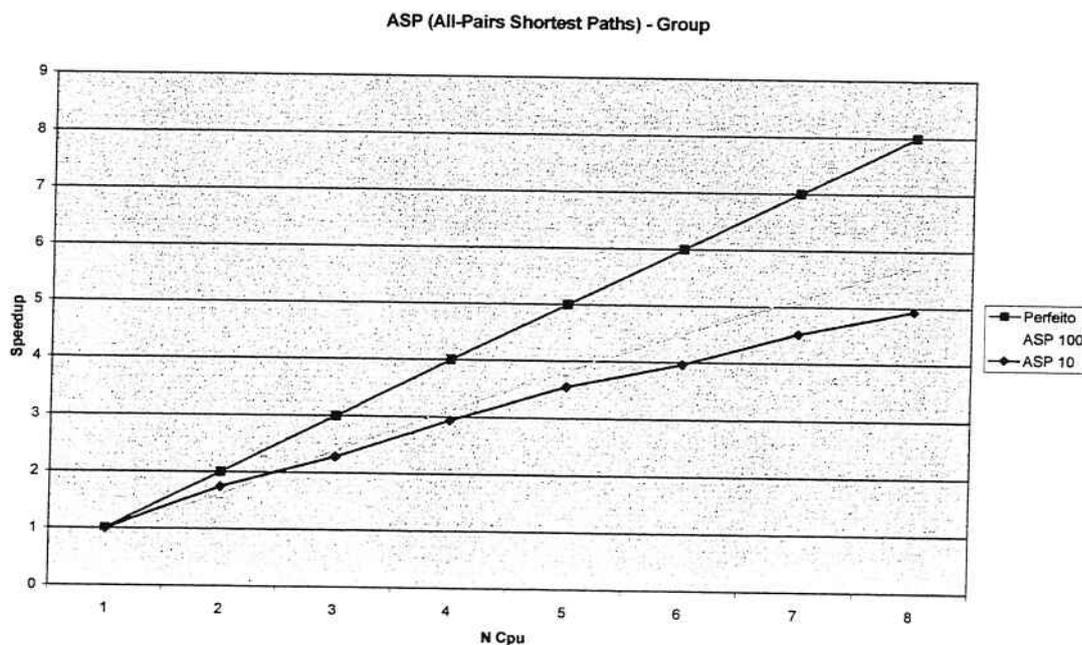


Figura 5 – Speedup ASP group

4. Considerações Finais

Neste trabalho fizemos uma análise do comportamento de um sistema distribuído ao variar a velocidade da rede de interconexão do mesmo. Ao alterar a velocidade da rede de 10 Mbps para 100 Mbps não foi obtido um ganho de performance do protocolo de rede desta mesma ordem.

Uma das explicações para este fato é a seguinte. Os mecanismos de controle de fluxo e controle de congestão implementados no protocolo FLIP, protocolo de comunicação de grupo

e protocolo RPC do Amoeba são dependentes da velocidade da rede. Estes mecanismos foram calibrados para uma rede ethernet de 10 Mbps e não funcionam satisfatoriamente quando a rede está operando a 100 Mbps. Fica claro, portanto, que somente o aumento da velocidade da rede não acarreta numa melhora do sistema como um todo. Para que este aumento de velocidade seja realmente repassado para as aplicações, é necessário que o protocolo de rede esteja preparado para esta nova velocidade.

Porém, apesar das observações acima, ficou evidente nos estudos efetuados com as aplicações distribuídas que o uso de uma velocidade de rede alta permite manter um ganho de performance ao acrescentar um número maior de processadores. Além disto o FLIP mostrou ser um protocolo eficiente e que em condições normais, juntamente com o protocolo de comunicação de grupo, facilita bastante o desenvolvimento de aplicações distribuídas.

5. Bibliografia

ANDERSON, T.; CULLER, D.; PATTERSON, D. A Case for NOW (Networks of Workstations). **IEEE Micro**, fev. 1995.

KAASHOEK, M.F. et al. FLIP: an internetwork protocol for supporting distributed systems. **ACM Trans. Computer Systems**, v.11, n.1, p.77-106, fev. 1993.

RENESE, R.V.; STAVAREN, H.V.; TANENBAUM, A.S. The Performance of the Amoeba Distributed Operating System. **Software - Practice and Experience**, v.19, n.3, p.223-234, mar. 1989.

TANENBAUM A.S. et al. Experiences with the Amoeba Distributed Operating System. **Communications of the ACM**, v.33, n.12, 1990.

TANENBAUM, A.S. et al. The Amoeba distributed operating system - a status report. **Computer Communications**, v.14, n.6, p.324-335, ago. 1991.

WIKIPEDIA – THE FREE ENCYCLOPEDIA. Seção Computer Science. Disponível em: <<http://www.wikipedia.com>>. Acesso em: 20 de mar. 2002.

BOLETINS TÉCNICOS - TEXTOS PUBLICADOS

- BT/PCS/9301 - Interligação de Processadores através de Chaves Ômicron - GERALDO LINO DE CAMPOS, DEMI GETSCHKO
- BT/PCS/9302 - Implementação de Transparência em Sistema Distribuído - LUÍSA YUMIKO AKAO, JOÃO JOSÉ NETO
- BT/PCS/9303 - Desenvolvimento de Sistemas Especificados em SDL - SIDNEI H. TANO, SELMA S. S. MELNIKOFF
- BT/PCS/9304 - Um Modelo Formal para Sistemas Digitais à Nivel de Transferência de Registradores - JOSÉ EDUARDO MOREIRA, WILSON VICENTE RUGGIERO
- BT/PCS/9305 - Uma Ferramenta para o Desenvolvimento de Protótipos de Programas Concorrentes - JORGE KINOSHITA, JOÃO JOSÉ NETO
- BT/PCS/9306 - Uma Ferramenta de Monitoração para um Núcleo de Resolução Distribuída de Problemas Orientado a Objetos - JAIME SIMÃO SICHMAN, ELERI CARDOSO
- BT/PCS/9307 - Uma Análise das Técnicas Reversíveis de Compressão de Dados - MÁRIO CESAR GOMES SEGURA, EDIT GRASSIANI LINO DE CAMPOS
- BT/PCS/9308 - Proposta de Rede Digital de Sistemas Integrados para Navio - CESAR DE ALVARENGA JACOBY, MOACYR MARTUCCI JR.
- BT/PCS/9309 - Sistemas UNIX para Tempo Real - PAULO CESAR CORIGLIANO, JOÃO JOSÉ NETO
- BT/PCS/9310 - Projeto de uma Unidade de Matching Store baseada em Memória Paginada para uma Máquina Fluxo de Dados Distribuído - EDUARDO MARQUES, CLAUDIO KIRNER
- BT/PCS/9401 - Implementação de Arquiteturas Abertas: Uma Aplicação na Automação da Manufatura - JORGE LUIS RISCO BECERRA, MOACYR MARTUCCI JR.
- BT/PCS/9402 - Modelamento Geométrico usando do Operadores Topológicos de Euler - GERALDO MACIEL DA FONSECA, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9403 - Segmentação de Imagens aplicada a Reconhecimento Automático de Alvos - LEONCIO CLARO DE BARROS NETO, ANTONIO MARCOS DE AGUIRRA MASSOLA
- BT/PCS/9404 - Metodologia e Ambiente para Reutilização de Software Baseado em Composição - LEONARDO PUJATTI, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9405 - Desenvolvimento de uma Solução para a Supervisão e Integração de Células de Manufatura Discreta - JOSÉ BENEDITO DE ALMEIDA, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/9406 - Método de Teste de Sincronização para Programas em ADA - EDUARDO T. MATSUDA, SELMA SHIN SHIMIZU MELNIKOFF
- BT/PCS/9407 - Um Compilador Paralelizante com Detecção de Paralelismo na Linguagem Intermediária - HSUEH TSUNG HSIANG, LÍRIA MATSUMOTO SAITO
- BT/PCS/9408 - Modelamento de Sistemas com Redes de Petri Interpretadas - CARLOS ALBERTO SANGIORGIO, WILSON V. RUGGIERO
- BT/PCS/9501 - Síntese de Voz com Qualidade - EVANDRO BACCI GOUVÊA, GERALDO LINO DE CAMPOS
- BT/PCS/9502 - Um Simulador de Arquiteturas de Computadores "A Computer Architecture Simulator" - CLAUDIO A. PRADO, WILSON V. RUGGIERO
- BT/PCS/9503 - Simulador para Avaliação da Confiabilidade de Sistemas Redundantes com Reparo - ANDRÉA LUCIA BRAGA, FRANCISCO JOSÉ DE OLIVEIRA DIAS
- BT/PCS/9504 - Projeto Conceitual e Projeto Básico do Nível de Coordenação de um Sistema Aberto de Automação, Utilizando Conceitos de Orientação a Objetos - NELSON TANOMARU, MOACYR MARTUCCI JUNIOR
- BT/PCS/9505 - Uma Experiência no Gerenciamento da Produção de Software - RICARDO LUIS DE AZEVEDO DA ROCHA, JOÃO JOSÉ NETO
- BT/PCS/9506 - MétodoOO - Método de Desenvolvimento de Sistemas Orientado a Objetos: Uma Abordagem Integrada à Análise Estruturada e Redes de Petri - KECHI HIRAMA, SELMA SHIN SHIMIZU MELNIKOFF
- BT/PCS/9601 - MOOPP: Uma Metodologia Orientada a Objetos para Desenvolvimento de Software para Processamento Paralelo - ELISA HATSUE MORIYA HUZITA, LÍRIA MATSUMOTO SATO
- BT/PCS/9602 - Estudo do Espalhamento Brillouin Estimulado em Fibras Ópticas Monomodo - LUIS MEREJE SANCHES, CHARLES ARTUR SANTOS DE OLIVEIRA
- BT/PCS/9603 - Programação Paralela com Variáveis Compartilhadas para Sistemas Distribuídos - LUCIANA BEZERRA ARANTES, LÍRIA MATSUMOTO SATO
- BT/PCS/9604 - Uma Metodologia de Projeto de Redes Locais - TEREZA CRISTINA MELO DE BRITO CARVALHO, WILSON VICENTE RUGGIERO

- BT/PCS/9605 - Desenvolvimento de Sistema para Conversão de Textos em Fonemas no Idioma Português - DIMAS TREVIZAN CHBANE, GERALDO LINO DE CAMPOS
- BT/PCS/9606 - Sincronização de Fluxos Multimídia em um Sistema de Videoconferência - EDUARDO S. C. TAKAHASHI, STEFANIA STIUBIENER
- BT/PCS/9607 - A importância da Completeza na Especificação de Sistemas de Segurança - JOÃO BATISTA CAMARGO JÚNIOR, BENÍCIO JOSÉ DE SOUZA
- BT/PCS/9608 - Uma Abordagem Paraconsistente Baseada em Lógica Evidencial para Tratar Exceções em Sistemas de Frames com Múltipla Herança - BRÁULIO COELHO ÁVILA, MÁRCIO RILLO
- BT/PCS/9609 - Implementação de Engenharia Simultânea - MARCIO MOREIRA DA SILVA, MOACYR MARTUCCI JÚNIOR
- BT/PCS/9610 - Statecharts Adaptativos - Um Exemplo de Aplicação do STAD - JORGE RADY DE ALMEIDA JUNIOR, JOÃO JOSÉ NETO
- BT/PCS/9611 - Um Meta-Editor Dirigido por Sintaxe - MARGARETE KEIKO IWAI, JOÃO JOSÉ NETO
- BT/PCS/9612 - Reutilização em Software Orientado a Objetos: Um Estudo Empírico para Analisar a Dificuldade de Localização e Entendimento de Classes - SELMA SHIN SHIMIZU MELNIKOFF, PEDRO ALEXANDRE DE OLIVEIRA GIOVANI
- BT/PCS/9613 - Representação de Estruturas de Conhecimento em Sistemas de Banco de Dados - JUDITH PAVÓN MENDONZA, EDIT GRASSIANI LINO DE CAMPOS
- BT/PCS/9701 - Uma Experiência na Construção de um Tradutor Inglês - Português - JORGE KINOSHITA, JOÃO JOSÉ NETO
- BT/PCS/9702 - Combinando Análise de "Wavelet" e Análise Entrópica para Avaliar os Fenômenos de Difusão e Correlação - RUI CHUO HUEI CHIOU, MARIA ALICE G. V. FERREIRA
- BT/PCS/9703 - Um Método para Desenvolvimento de Sistemas de Computacionais de Apoio a Projetos de Engenharia - JOSÉ EDUARDO ZINDEL DEBONI, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/9704 - O Sistema de Posicionamento Global (GPS) e suas Aplicações - SÉRGIO MIRANDA PAZ, CARLOS EDUARDO CUGNASCA
- BT/PCS/9705 - METAMBI-OO - Um Ambiente de Apoio ao Aprendizado da Técnica Orientada a Objetos - JOÃO UMBERTO FURQUIM DE SOUZA, SELMA S. S. MELNIKOFF
- BT/PCS/9706 - Um Ambiente Interativo para Visualização do Comportamento Dinâmico de Algoritmos - IZAURA CRISTINA ARAÚJO, JOÃO JOSÉ NETO
- BT/PCS/9707 - Metodologia Orientada a Objetos e sua Aplicação em Sistemas de CAD Baseado em "Features" - CARLOS CÉSAR TANAKA, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9708 - Um Tutor Inteligente para Análise Orientada a Objetos - MARIA EMÍLIA GOMES SOBRAL, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9709 - Metodologia para Seleção de Solução de Sistema de Aquisição de Dados para Aplicações de Pequeno Porte - MARCELO FINGUERMAN, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/9801 - Conexões Virtuais em Redes ATM e Escalabilidade de Sistemas de Transmissão de Dados sem Conexão - WAGNER LUIZ ZUCCHI, WILSON VICENTE RUGGIERO
- BT/PCS/9802 - Estudo Comparativo dos Sistemas da Qualidade - EDISON SPINA, MOACYR MARTUCCI JR.
- BT/PCS/9803 - The VIBRA Multi-Agent Architecture: Integrating Purposive Vision With Deliberative and Reactive Planning - REINALDO A. C. BIANCHI, ANNA H. REALI C. RILLO, LELIANE N. BARROS
- BT/PCS/9901 - Metodologia ODP para o Desenvolvimento de Sistemas Abertos de Automação - JORGE LUIS RISCO BECCERRA, MOACYR MARTUCCI JUNIOR
- BT/PCS/9902 - Especificação de Um Modelo de Dados Bitemporal Orientado a Objetos - SOLANGE NICE ALVES DE SOUZA, EDIT GRASSIANI LINO DE CAMPOS
- BT/PCS/9903 - Implementação Paralela Distribuída da Dissecção Cartesiana Aninhada - HILTON GARCIA FERNANDES, LIRIA MATSUMOTO SATO
- BT/PCS/9904 - Metodologia para Especificação e Implementação de Solução de Gerenciamento - SERGIO CLEMENTE, TEREZA CRISTINA MELO DE BRITO CARVALHO
- BT/PCS/9905 - Modelagem de Ferramenta Hiperídia Aberta para a Produção de Tutoriais Interativos - LEILA HYODO, ROMERO TORI
- BT/PCS/9906 - Métodos de Aplicações da Lógica Paraconsistente Anotada de Anotação com Dois Valores-LPA2v com Construção de Algoritmo e Implementação de Circuitos Eletrônicos - JOÃO I. DA SILVA FILHO, JAIR MINORO ABE
- BT/PCS/9907 - Modelo Nebuloso de Confiabilidade Baseado no Modelo de Markov - PAULO SÉRGIO CUGNASCA, MARCO TÚLIO CARVALHO DE ANDRADE
- BT/PCS/9908 - Uma Análise Comparativa do Fluxo de Mensagens entre os Modelos da Rede Contractual (RC) e Colisões Baseada em Dependências (CBD) - MÁRCIA ITO, JAIME SIMÃO SICHMAN

- BT/PCS/9909 – Otimização de Processo de Inserção Automática de Componentes Eletrônicos Empregando a Técnica de Times Assíncronos – CESAR SCARPINI RABAK, JAIME SIMÃO SICHMAN
- BT/PCS/9910 – MIISA – Uma Metodologia para Integração da Informação em Sistemas Abertos – HILDA CARVALHO DE OLIVEIRA, SELMA S. S. MELNIKOFF
- BT/PCS/9911 – Metodologia para Utilização de Componentes de Software: um estudo de Caso – KAZUTOSI TAKATA, SELMA S. S. MELNIKOFF
- BT/PCS/0001 – Método para Engenharia de Requisitos Norteado por Necessidades de Informação – ARISTIDES NOVELLI FILHO, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/0002 – Um Método de Escolha Automática de Soluções Usando Tecnologia Adaptativa – RICARDO LUIS DE AZEVEDO DA ROCHA, JOÃO JOSÉ NETO
- BT/PCS/0101 – Gerenciamento Hierárquico de Falhas – JAMIL KALIL NAUFAL JR., JOÃO BATISTA CAMARGO JR.
- BT/PCS/0102 – Um Método para a Construção de Analisadores Morfológicos, Aplicado à Língua Portuguesa, Baseado em Autômatos Adaptativos – CARLOS EDUARDO DANTAS DE MENEZES, JOÃO JOSÉ NETO
- BT/PCS/0103 – Educação pela Web: Metodologia e Ferramenta de Elaboração de Cursos com Navegação Dinâmica – LUISA ALEYDA GARCIA GONZÁLEZ, WILSON VICENTE RUGGIERO
- BT/PCS/0104 – O Desenvolvimento de Sistemas Baseados em Componentes a Partir da Visão de Objetos – RENATA EVANGELISTA ROMARIZ RECCO, JOÃO BATISTA CAMARGO JÚNIOR
- BT/PCS/0105 – Introdução às Gramáticas Adaptativas – MARGARETE KEIKO IWAI, JOÃO JOSÉ NETO
- BT/PCS/0106 – Automação dos Processos de Controle de Qualidade da Água e Esgoto em Laboratório de Controle Sanitário – JOSÉ BENEDITO DE ALMEIDA, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/0107 – Um Mecanismo para Distribuição Segura de Vídeo MPEG – CÍNTIA BORGES MARGI, GRAÇA BESSAN, WILSON VICENTE RUGGIERO
- BT/PCS/0108 – A Dependence-Based Model for Social Reasoning in Multi-Agent Systems – JAIME SIMÃO SICHMAN
- BT/PCS/0109 – Ambiente Multilinguagem de Programação – Aspectos do Projeto e Implementação – APARECIDO VALDEMIR DE FREITAS, JOÃO JOSÉ NETO
- BT/PCS/0110 – LETAC: Técnica para Análise de Tarefas e Especificação de Fluxo de Trabalho Cooperativo – MARCOS ROBERTO GREINER, LUCIA VILELA LEITE FILGUEIRAS
- BT/PCS/0111 – Modelagem ODP para o Planejamento de Sistemas de Potência – ANIRIO SALLES FILHO, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/0112 – Técnica para Ajuste dos Coeficientes de Quantização do Padrão MPEG em Tempo Real – REGINA M. SILVEIRA, WILSON V. RUGGIERO
- BT/PCS/0113 – Segmentação de Imagens por Classificação de Cores: Uma Abordagem Neural – ALEXANDRE S. SIMÕES, ANNA REALI COSTA
- BT/PCS/0114 – Uma Avaliação do Sistema DSM Nautilus – MARIO DONATO MARINO, GERALDO LINO DE CAMPOS
- BT/PCS/0115 – Utilização de Redes Neurais Artificiais para Construção de Imagem em Câmara de Cintilação – LUIZ SÉRGIO DE SOUZA, EDITH RANZINI
- BT/PCS/0116 – Simulação de Redes ATM – HSU CHIH WANG CHANG, WILSON VICENTE RUGGIERO
- BT/PCS/0117 – Application of Monoprocessed Architecture for Safety Critical Control Systems – JOSÉ ANTONIO FONSECA, JORGE RADY DE ALMEIDA JR.
- BT/PCS/0118 – WebBee – Um Sistema de Informação via WEB para Pesquisa de Abelhas sem Ferrão – RENATO SOUSA DA CUNHA, ANTONIO MOURA SARAIVA
- BT/PCS/0119 – Parallel Processing Applied to Robot Manipulator Trajectory Planning – DENIS HAMILTON NOMIYAMA, LÍRIA MATSUMOTO SATO, ANDRÉ RIYUITSU HIRAKAWA
- BT/PCS/0120 – Utilização de Padrão de Arquitetura de Software para a Fase de Projeto Orientado a Objetos – CRISITINA MARIA FERREIRA DA SILVA, SELMA SHIN SHIMIZU MELNIKOFF
- BT/PCS/0121 – Agilizando Aprendizagem por Reforço Através do uso de Conhecimento sobre o Domínio – RENÉ PEGORARO, ANNA H. REALI COSTA
- BT/PCS/0122 – Modelo de Segurança da Linguagem Java Problemas e Soluções – CLAUDIO MASSANORI MATAYOSHI, WILSON VICENTE RUGGIERO
- BT/PCS/0123 – Proposta de um Agente CNM para o Gerenciamento Web de um Backbone ATM – FERNANDO FROTA REDÍGOLO, TEREZA CRISTINA MELO DE BRITO CARVALHO
- BT/PCS/0124 – Um Método de Teste de software Baseado em Casos Teste – SÉRGIO RICARDO ROTTA, KECHI HIRAMA
- BT/PCS/0201 – A Teoria Nebulosa Aplicada a uma Bicicleta Ergométrica para Fisioterapia – MARCO ANTONIO GARMS, MARCO TÚLIO CARVALHO DE ANDRADE
- BT/PCS/0202 – Synchronization Constraints in a Concurrent Object Oriented Programming Model – LAÍS DO NASCIMENTO SALVADOR, LÍRIA MATSUMOTO SATO

- BT/PCS/0203 – Construção de um Ambiente de Dados sobre um Sistema de Arquivos Paralelos – JOSÉ CRAVEIRO DA COSTA NETO, LIRIA MATSUMOTO SATO
- BT/PCS/0204 – Maestro: Um Middleware para Suporte a Aplicações Distribuídas Baseadas em Componentes de Software – CLÁUDIO LUÍS PEREIRA FERREIRA, JORGE LUÍS RISCO BECERRA
- BT/PCS/0205 - Sistemas de Automação dos Transportes (ITS) Descritos Através das Técnicas de Modelagem RM-OPD (ITU-T) e UML (OMG) – CLÁUDIO LUIZ MARTE, JORGE LUÍS RISCO BECERRA, JOSÉ SIDNEI COLOMBO
- BT/PCS/0206 – Comparação de Perfis de Usuários Coletados Através do Agente de Interface PersonalSearcher – GUSTAVO A. GIMÉNEZ LUGO, ANALÍA AMANDI, JAIME SIMÃO SICHMAN
- BT/PCS/0207 – Arquitetura Reutilizáveis para a Criação de Sistemas de Tutorização Inteligentes – MARCO ANTONIO FURLAN DE SOUZA, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/0208 – Análise e Predição de Desempenho de Programas Paralelos em Redes de Estações de Trabalho – LIN KUAN CHING, LIRIA MATSUMOTO SATO
- BT/PCS/0209 – Previsões Financeiras Através de Sistemas Neuronebulosos – DANIEL DE SOUZA GOMES, MARCO TÚLIO CARVALHO DE ANDRADE
- BT/PCS/0210 – Proposta de Arquitetura Aberta de Central de Atendimento – ANA PAULA GONÇALVES SERRA, MOACYR MARTUCCI JÚNIOR
- BT/PCS/0211 – Alternativas de Implementação de Sistemas Nebulosos em Hardware – MARCOS ALVES PREDEBON, MARCO TÚLIO CARVALHO DE ANDRADE
- BT/PCS/0212 – Registro de Imagens de Documentos Antigos – VALGUIMA VICTORIA VIANA ODAKURA MARTINEZ, GERALDO LINO DE CAMPOS
- BT/PCS/0213 – Um Modelo de Dados Multidimensional – PEDRO WILLEMSSENS, JORGE RADY DE ALMEIDA JUNIOR
- BT/PCS/0214 – Autômatos Adaptativos no Tratamento Sintático de Linguagem Natural – CÉLIA YUMI OKANO TANIWAKI, JOÃO JOSÉ NETO
- BT/PCS/0215 – Fatores e Subfatores para Avaliação da Segurança em Software de Sistemas Críticos – JOÃO EDUARDO PROENÇA PÁSCOA, JOÃO BATISTA CAMARGO JÚNIOR
- BT/PCS/0216 – Derivando um Modelo de Projeto a Partir de um Modelo de Análise, com Base em Design Patterns J2EE – SERGIO MARTINS FERNANDES, SELMA SHIN SHIMIZU MELNIKOFF
- BT/PCS/0217 – Domínios Virtuais para Redes Móveis Ad Hoc: Um Mecanismo de Segurança – LEONARDO AUGUSTO MARTUCCI, TEREZA CRISTINA DE MELO BRITO CARVALHO
- BT/PCS/0218 – Uma Ferramenta para a Formulação de Consultas Baseadas em Entidades e Papéis – ANDRÉ ROBERTO DORETO SANTOS, EDIT GRASSIANI LINO CAMPOS
- BT/PCS/0219 – Avaliação de Performance de Arquiteturas para Computação de Alto Desempenho – KARIN STRAUSS, WILSON VICENTE RUGGIERO
- BT/PCS/0220 – BGLsim: Simulador de Sistema Completo para o Blue Gene/L – LUÍS HENRIQUE DE BARROS CEZE, WILSON VICENTE RUGGIERO
- BT/PCS/0221 - μ P: Uma Solução de Micropagamentos – PEDRO ANCONA LOPEZ MINDLIN, TEREZA CRISTINA MELO DE BRITO CARVALHO
- BT/PCS/0222 - Modelamento de Roteadores IP para Análise de Atraso – MARCELO BLANES, GRAÇA BRESSAN
- BT/PCS/0223 - Uma Biblioteca de Classes Utilizando Java 3D para o Desenvolvimento de Ambientes Virtuais Multi-Usuários - RICARDO NAKAMURA, ROMERO TORI
- BT/PCS/0224 – Interactive 3D Physics Experiments Through the Internet – ALEXANDRE CARDOSO, ROMERO TORI

