

Processamento de imagens em cluster openMosix

(Image processing in openMosix cluster)

Alexsander Martini¹; Lucas de Paula Souza²; Lucimar Sasso Vieira³

¹G – Faculdade de Tecnologia (FATEC) de São José do Rio Preto-SP
alexmartini@gmail.com

²G – Faculdade de Tecnologia (FATEC) de São José do Rio Preto-SP
lucaspaulasouza@gmail.com

³Faculdade de Tecnologia (FATEC) de São José do Rio Preto-SP
lucimarsasso@ieee.com

Abstract. *This work aims to describe the potential of a Linux Cluster based on openMosix distribution for digital image processing in three dimension, using Povray software. The objective is to present the increase in processing speed by adding more computers (nodes) in the cluster, and in wich point occurs the the saturation (the addition of more nodes doesn't improve the performance).*

Keywords: *image processing; cluster; Linux; openMosix; high performance computer; parallel computer.*

Resumo. *Este trabalho tem o objetivo de explorar as potencialidades do uso de um cluster Linux baseado na distribuição openMosix na renderização de imagens digitais em terceira dimensão, utilizando o software Povray. O objetivo é apresentar o aumento na velocidade de processamento com a adição de mais computadores (nós) no cluster, e em que ponto ocorre a saturação (a adição de mais nós causa pouca ou nenhuma melhoria no desempenho).*

Palavras-chave. *processamento de imagens; cluster; Linux; openMosix; computação de alto desempenho; computação paralela.*

1. Introdução

Atualmente vive-se em uma era onde o máximo de desempenho computacional nunca é o suficiente. A demanda por processamento cada vez mais rápido cresce a todo dia, juntamente com evoluções na arquitetura dos computadores. Com o barateamento dos preços dos componentes utilizados nos computadores pessoais e melhorias significativas em seu desempenho, surgiu a idéia de se utilizar computadores de uso pessoal interligados em rede como um só computador de grande desempenho. A esse agrupamento de computadores interligados dá-se o nome de *cluster*. Nosso trabalho explora a utilização de uma implementação específica de *cluster* juntamente com um software de renderização de imagens em terceira dimensão que demanda alto poder de processamento.

2. Metodologia

Para a realização dos testes foi utilizado o ClusterKnoppix versão 3.4, uma distribuição Linux baseada no Knoppix, mas que já vem com o openMosix instalado. Pelo fato de rodar diretamente do CD, propicia um ambiente de testes flexível e rápido de se implementar.

O software de renderização de imagens utilizado foi o Povray versão 3.0, que é um software de uso livre que permite, a partir da especificação de formas, cores, texturas e iluminação, a criação de imagens de realismo fotográfico. Para facilitar a execução do Povray foi utilizada uma GUI (Gaphical user inteface), o Povmosix versão 1.0, que possibilita definir a quantidade de processos que devem ser gerados para a renderização da cena.

No presente trabalho foram utilizados 14 computadores pessoais para a montagem do *cluster*, sendo que todos os computadores possuem a seguinte configuração: processador Pentium 4 2.8 GHz, 512 MB de memória, disco rígido de 80 GB e placa de rede Gigabit, interligados por um switch D-Link Fast-Ethernet 100 Mbits.

A implementação de *cluster* utilizado é o openMosix, que utiliza a migração de processos entre os nós do *cluster* como forma de balanceamento de carga. Essa forma transparente de funcionamento permite que qualquer programa, mesmo não otimizado para a execução distribuída, se beneficie com a execução nesse *cluster*.

3. Resultados

Para o teste foi utilizada a cena chess2.pov, que é um exemplo integrante do pacote do software Povray. A cena foi renderizada na resolução 1600x1200 pixels e pode ser vista na figura 1:



Figura 1 – Cena chess2.pov renderizada

A seqüência de teste foi executada três vezes, de forma a aumentar a confiabilidade dos dados. O tempo consumido com esse procedimento foi alto (aproximadamente cinco horas de testes), mas gerou resultados bastante consistentes. Os tempos de renderização das seqüências de testes podem ser vistos abaixo:

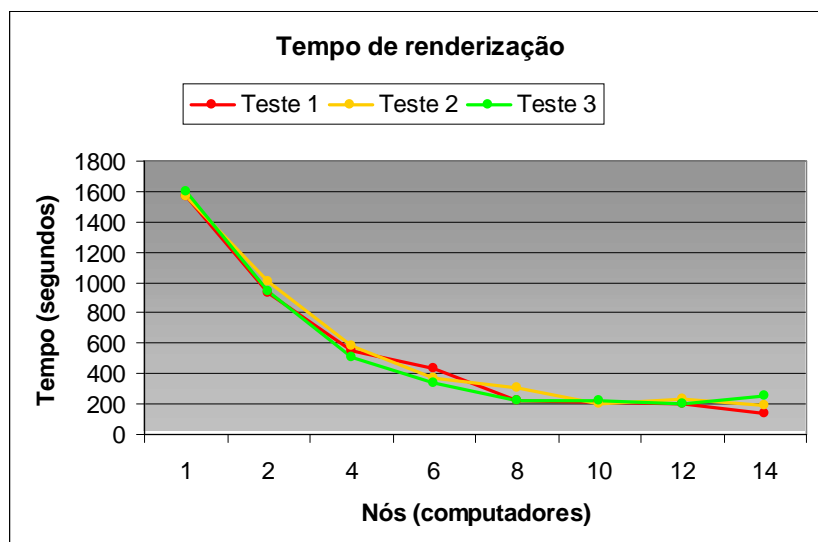


Gráfico 1 – Seqüências de testes executados
 Fonte: Produção destes autores

A tabela dos desvios padrão pode ser observada abaixo:

nós (computadores)	Teste 1	Teste 2	Teste 3	Desvio Padrão	%
1	1567	1570	1594	14,80	0,938%
2	933	1004	946	37,80	3,934%
4	553	578	512	33,32	6,081%
6	433	367	336	49,54	13,072%
8	222	308	224	49,08	19,556%
10	208	197	219	11,00	5,288%
12	203	231	197	18,15	8,642%
14	139	190	259	60,22	30,727%

Tabela 1 – Desvios padrões observados
 Fonte: Produção destes autores

4. Discussão e Conclusões

Pode-se observar que existe um grande ganho de desempenho com a adição de uma máquina (diminuição de até 40%, do tempo de renderização) em relação ao tempo anterior, no qual se utilizou uma máquina apenas. Com a adição de mais duas máquinas (total de quatro agora), obteve-se em média um processamento até 43% mais rápido. Com a adição de mais dois computadores (total de seis), observa-se um processamento em média 31% mais rápido, em relação ao tempo anterior. Com a adição de mais dois computadores (total de oito), reduziu-se 34% do tempo, em média, com relação a seis computadores. A partir de dez computadores, essa taxa passa a cair, sendo em média de apenas 17% em relação ao tempo anterior. Com a adição de mais dois computadores, totalizando doze, têm-se um aumento do

tempo de processamento, que demora, na média dos tempos, 1% a mais para ser executado. Esse resultado indesejável mostrar que o *cluster* chegou ao ponto de saturação, onde a adição de mais computadores causará pouca ou nenhuma melhoria de desempenho. Quando se chegou ao total de catorze computadores, voltou-se a ter uma redução de tempo, com um processamento 7% mais rápido em média, mais um ganho pouco significativo.

Segundo Pitantaga (2004), essa limitação à adição de mais nós nesse *cluster*, pode ser causada principalmente pela estrutura de rede utilizada, que foi a Fast-Ethernet (IEEE 802.3u) com velocidade de 100 Mbps. A utilização de uma estrutura de rede baseada no padrão Gigabit Ethernet (IEEE 802.3z) aumentaria consideravelmente a largura de banda (1.000 Mbps teórica), possibilitando assim a utilização de mais computadores nesse processamento, mas essa hipótese não pode ser verificada na prática.

Esses testes mostraram também que o ambiente do *cluster* openMosix é uma plataforma ideal para a renderização de cenas complexas, devido à ótima relação custo benefício, além da alta escalabilidade e flexibilidade dessa solução. Comenta-se também o fato da possibilidade de utilização de estações sem disco rígido, pois todo o processo de inicialização e carregamento do Linux nas estações foi remoto, utilizando o LTSP (Linux Terminal Server Project) que já se encontra disponível no ClusterKnoppix.

5. Referências

- ClusterKnoppix. Disponível em: <<http://clusterknoppix.sw.be/>>. Acessado em: 15 agosto 2006.
- Linux Terminal Server Project. Disponível em: < <http://www.ltsp.org> >. Acessado em: 15 agosto 2006.
- openMosix. Disponível em: < <http://openmosix.sourceforge.net> >. Acessado em: 15 agosto 2006.
- PITANGA, M. **Computação em cluster**. 2. ed. Rio de Janeiro: Brasport, 2004.
- Povmosix. Disponível em: < <http://povmosix.sourceforge.net> >. Acessado em: 15 agosto 2006.
- Povray. Disponível em: < <http://www.povray.org> >. Acessado em: 15 agosto 2006.