

Trabalho 01 - Simulação de Memória - Amnesia

1st Andre Santos

Ciência da Computação

andre.alves.1330374@sga.pucminas.br

2nd Bernardo Bartholomeu

Ciência da Computação

barbartholomeu@sga.pucminas.br

3rd Gabriel Fernandes

Ciência da Computação

gabriel.fernandes.1378675@sga.pucminas.br

4th Luis Henrique Guedes

Ciência da Computação

lhguedes@sga.pucminas.br

5th Nathan de Araujo

Ciência da Computação

email address or ORCID

***Index Terms*—Hierarquia de memória, localidade espacial, localidade temporal, simulador Amnesia.**

I. INTRODUÇÃO

A eficiência dos sistemas computacionais modernos está fortemente atrelada ao desempenho da hierarquia de memória. Nesse contexto, estratégias como localidade espacial e temporal desempenham um papel essencial na redução de latências e na otimização do acesso aos dados. Este estudo utiliza o simulador Amnesia como ferramenta para investigar, de forma prática e didática, o comportamento da hierarquia de memória sob diferentes configurações.

Através de experimentos controlados, o trabalho explora o impacto de parâmetros como tamanho de cache, nível de associatividade, políticas de escrita e substituição de blocos, além do uso de caches multiníveis e TLBs. O objetivo é compreender como essas variáveis influenciam métricas cruciais, como a taxa de acertos em cache e o tempo médio de acesso à memória, fornecendo uma base comparativa para avaliar a eficácia de diferentes arquiteturas de memória.

A. Objetivos

O objetivo deste trabalho é analisar o impacto de distintas configurações na hierarquia de memória, considerando parâmetros como tamanho das caches, níveis de associatividade, políticas de escrita e algoritmos de substituição. Busca-se compreender como essas variáveis influenciam o desempenho do sistema, em particular a taxa de acertos na cache e o tempo médio de acesso à memória. A pesquisa propõe uma avaliação comparativa dos resultados obtidos para identificar configurações mais eficientes sob diferentes cenários de execução.

II. TRABALHOS RELACIONADOS

Diversos estudos têm explorado o uso de ferramentas educacionais para facilitar o entendimento da hierarquia de memória e seus impactos no desempenho de sistemas computacionais. Dentre eles, destaca-se o artigo “Aprendendo Hierarquia de Memória e a Exploração das Localidades Espacial e Temporal com o Simulador Amnesia”, que propõe uma abordagem didática utilizando o simulador Amnesia. Através de um

ambiente interativo, os autores permitem que estudantes experimentem diferentes estratégias de substituição e configurações de cache, promovendo um aprendizado prático sobre o comportamento da memória.

Outro trabalho relevante é o artigo “Amnésia: um Objeto de Aprendizagem para o Ensino de Organização e Arquitetura de Computadores”, que descreve o desenvolvimento do simulador como um objeto de aprendizagem voltado para o ensino de conceitos fundamentais como cache, memória principal e políticas de substituição. A interface visual e os recursos interativos do Amnesia demonstraram grande eficácia no processo de ensino-aprendizagem, apresentando boa aceitação entre os usuários e contribuindo significativamente para a assimilação dos conteúdos.

Esses estudos reforçam a importância de abordagens práticas e visuais no ensino de arquitetura de computadores, evidenciando o potencial do simulador Amnesia como uma ferramenta didática valiosa. O presente trabalho expande essa perspectiva ao utilizar o simulador não apenas como recurso educacional, mas como meio de investigação técnica e análise comparativa entre diferentes configurações de hierarquia de memória.

2. O artigo “Amnésia: um Objeto de Aprendizagem para o Ensino de Organização e Arquitetura de Computadores” descreve o desenvolvimento de um objeto de aprendizagem interativo para auxiliar no ensino dos conceitos de memória principal e hierarquia de memória. A interface visual permite a experimentação de diversos parâmetros, como tamanho de cache e políticas de substituição, contribuindo significativamente para o entendimento dos conceitos e demonstrando boa aceitação pelos usuários.

III. PROPOSTA DE ARQUITETURA

A definição da arquitetura inicial é um passo fundamental para a análise dos efeitos das estratégias de localidade espacial e temporal na hierarquia de memória. Neste trabalho, foi estabelecida uma configuração base, utilizada como referência para as simulações no simulador Amnesia.

Foram selecionados parâmetros importantes como o tamanho da cache, associatividade, tamanho das linhas e palavras, além das políticas de coerência e substituição. A presença da TLB (*Translation Lookaside Buffer*) também foi

considerada, permitindo observar seu impacto na eficiência da hierarquia de memória.

A Tabela I apresenta a configuração base adotada:

TABLE I: Configuração Base do Sistema

Parâmetro	Valor
Tamanho da linha	2 bytes
Número de vias (associatividade)	1 via (direta)
Tamanho da palavra	4 bytes
Tamanho total da cache	64 bytes
Política de coerência	Write-Back
Política de substituição	FIFO
Tamanho da memória principal	1024 bytes

Essa configuração inicial visa proporcionar uma base controlada e simples, permitindo observar com clareza os impactos de cada ajuste individual nos testes posteriores. A partir dela, diferentes cenários foram simulados com alterações nos parâmetros para avaliar o desempenho em termos de tempo de execução, taxa de acertos na cache e comportamento da TLB.

IV. PROPOSTA DE TRACE

A definição do *trace* é um elemento crucial na simulação, uma vez que representa a sequência de operações realizadas pelo processador, como leituras, gravações e buscas de instruções. No simulador *Amnesia*, cada linha do arquivo de entrada é composta por uma dupla: um rótulo (representado em decimal) e um endereço (em hexadecimal). Os rótulos são interpretados da seguinte forma:

- Rótulo “0”: leitura de dados;
- Rótulo “1”: gravação de dados;
- Rótulo “2”: busca de instrução;
- Rótulo “3”: registro *escape* (acesso desconhecido);
- Rótulo “4”: registro *escape* (cache *flush*).

A escolha do *trace* influencia diretamente os resultados da simulação, uma vez que padrões de acesso distintos afetam de maneira diferente o comportamento da hierarquia de memória. Traços com acessos contíguos tendem a se beneficiar de blocos de cache maiores, devido à exploração da localidade espacial. Em contrapartida, traços com acessos aleatórios geralmente não tiram proveito de blocos maiores, podendo inclusive gerar mais falhas por capacidade.

Além disso, a aplicação de políticas de substituição como a *Least Recently Used* (LRU) mostra-se mais eficaz em cenários com alta localidade temporal, ao priorizar a retenção de dados recentemente acessados. Assim, a escolha criteriosa do *trace* é essencial para simulações representativas e para uma análise comparativa entre diferentes configurações de memória.

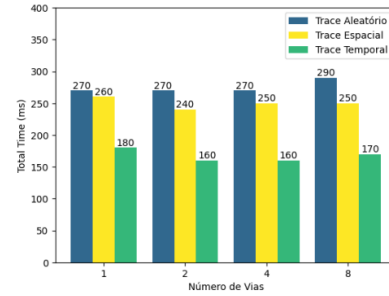
V. AVALIAÇÃO DE RESULTADOS: MEMÓRIA CACHE

Para a análise dos experimentos com a memória cache, os resultados foram organizados em três cenários distintos.

A. Cenário 1 – Impacto da Associatividade

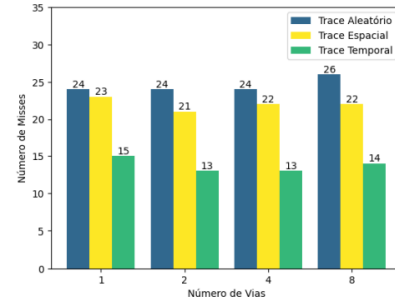
Neste cenário, avaliou-se o efeito do aumento do número de vias na cache. No **Trace 1**, o tempo total de execução se mantém constante até quatro vias; com oito vias, há um aumento do tempo, decorrente das frequentes substituições de

blocos, uma vez que os endereços aleatórios não exploram localidade. Esse traço apresenta oito *cache misses* compulsórios no primeiro acesso, além de misses por conflito e capacidade (a cache comporta 8 palavras enquanto o traço acessa 16). Por outro lado, o **Trace 2**, com endereços sequenciais, mostra uma leve melhoria, ainda que os misses sejam altos. Em contraste, o **Trace 3**, que favorece a localidade temporal, apresenta o melhor desempenho, devido à repetição frequente de endereços que gera alta taxa de *cache hits*. Nota-se que a configuração completamente associativa (8 vias) com política FIFO ocasionou substituições prematuras.



(a) Cenário 1: Aumento da Associatividade.

(a) Tempo de execução e misses (Trace 1).



(b) Misses cenário 1

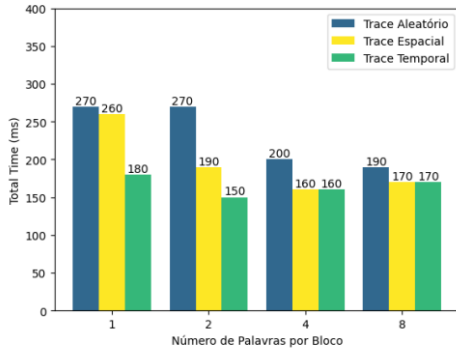
Fig. 1. Memória Cache: Cenário 1.

(b) Substituição de blocos (Trace 1).

Fig. 1: Resultados do Cenário 1 – Impacto da Associatividade.

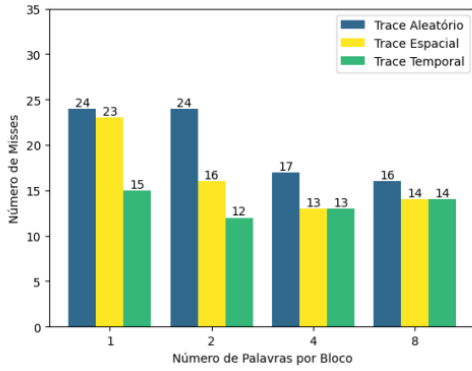
B. Cenário 2 – Influência do Tamanho do Bloco

Neste cenário, analisou-se o efeito do número de palavras por bloco na cache. Conforme a Figura 2a, para o **Trace 1**, o tempo total de execução diminuiu gradativamente com o aumento do tamanho do bloco, devido à melhor exploração da localidade espacial (maior chance de reutilização dos dados já carregados, reduzindo os misses, conforme Figura 2b). No **Trace 2**, o melhor desempenho ocorreu com blocos de quatro palavras, enquanto blocos maiores elevaram os custos de substituição. Já no **Trace 3**, blocos de duas palavras apresentaram desempenho satisfatório, e blocos de 8 palavras aumentaram os misses por capacidade.



(a) Cenário 2: Aumento no número de palavras por bloco.

(a) Tempo total (Trace 1).



(b) Misses cenário 2

Fig. 2. Memória Cache: Cenário 2.

(b) Número de misses (Trace 1).

Fig. 2: Resultados do Cenário 2 – Influência do Tamanho do Bloco.

C. Cenário 3 – Caches Multiníveis

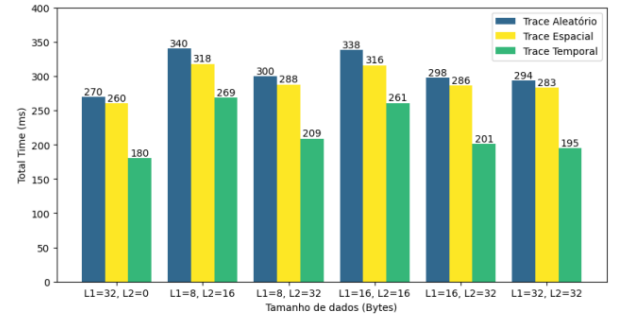
Este cenário teve como objetivo avaliar o impacto da utilização de caches multiníveis (níveis L1 e L2) no desempenho do sistema. Essa abordagem reflete a arquitetura adotada por processadores modernos, que utilizam múltiplos níveis de cache para mitigar a diferença de velocidade entre o processador e a memória principal.

Durante os testes, foram variadas as capacidades da cache L1 e L2 para observar os efeitos no tempo de execução e na taxa de *cache misses*. Observou-se que, quando a cache L1 possui tamanho suficiente para armazenar os dados mais utilizados (por exemplo, 8 palavras), há uma significativa redução no tempo total de execução. Isso ocorre porque a maioria dos acessos é resolvida no nível mais rápido da hierarquia.

No entanto, quando a cache L1 é pequena e o volume de dados acessados é elevado, o número de misses em L1 cresce, e os dados precisam ser buscados na cache L2 ou, eventualmente, na memória principal. Nesse caso, a presença da cache L2 funciona como uma camada intermediária que suaviza o impacto dos misses da L1, resultando em desempenho superior comparado a uma hierarquia com apenas um nível de cache.

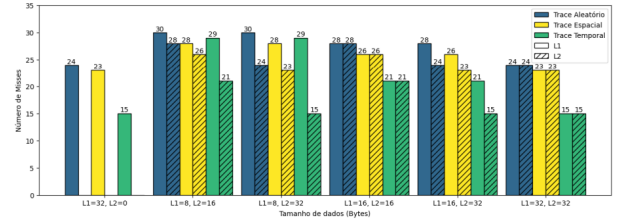
Na Figura 3a, nota-se que o tempo de execução diminui quando a cache L1 consegue capturar os dados com alta localidade temporal. Entretanto, ao aumentar o tamanho da cache L2 sem ajuste proporcional na L1, o tempo pode voltar a subir levemente, em função do aumento da latência de consulta nos dois níveis.

A Figura 3b mostra a comparação entre o número de misses nos diferentes níveis. Observa-se que, apesar da presença da L2 reduzir o número total de acessos à memória principal, um dimensionamento inadequado entre os níveis pode limitar os ganhos esperados. Estratégias de mapeamento eficazes e uma boa política de substituição são essenciais para maximizar os benefícios das caches multiníveis.



(a) Cenário 3: Caches Multiníveis.

(a) Tempo de execução em diferentes configurações de caches multiníveis.



(b) Misses cenário 3

Fig. 3. Memória Cache: Cenário 3.

(b) Comparação do número de misses em L1, L2 e Memória Principal.

Fig. 3: Resultados do Cenário 3 – Análise da hierarquia com múltiplos níveis de cache.

Em resumo, os resultados demonstram que o desempenho da hierarquia de memória está fortemente relacionado à configuração dos parâmetros de cache, sendo essencial a análise cuidadosa do tamanho, associatividade e número de níveis para alcançar um bom balanceamento entre custo e desempenho.

VI. CONCLUSÕES E AVALIAÇÃO DE RESULTADOS

Este trabalho apresentou uma análise aprofundada sobre o impacto de diferentes configurações da hierarquia de memória no desempenho de sistemas computacionais, utilizando o simulador *Amnesia* como ferramenta de experimentação. Foram

explorados parâmetros como o tamanho e associatividade das caches, políticas de substituição, tamanho da TLB e presença de caches multiníveis, avaliando seus efeitos diretos na taxa de acertos (hits), falhas (misses) e no tempo médio de acesso à memória.

Os resultados obtidos demonstram que o ajuste adequado da cache pode reduzir significativamente o número de acessos à memória principal, otimizando o tempo de resposta do sistema. A ampliação da TLB também se mostrou eficiente na diminuição de falhas de tradução, contribuindo para a melhoria global do desempenho. Adicionalmente, a adoção de caches multiníveis (L1 e L2) proporcionou um equilíbrio mais eficiente entre latência e capacidade, especialmente em cenários com alta localidade temporal.

Constatou-se que não existe uma configuração única ideal, mas sim um conjunto de boas práticas que devem considerar o perfil de acesso da aplicação e o tipo de *workload* executado. A combinação entre políticas eficientes de substituição, blocos de tamanho adequado e caches bem dimensionadas é essencial para alcançar um bom desempenho com o menor custo possível.

Esses achados reforçam a importância do entendimento teórico aliado à experimentação prática no ensino e na otimização de arquiteturas computacionais. Como proposta futura, sugere-se expandir o estudo para incluir a análise de caches de três níveis (L3) e a integração com *prefetching* e técnicas de especulação de memória, visando uma simulação ainda mais próxima da realidade de arquiteturas modernas.

Além disso, investigações futuras podem considerar a aplicação de cargas de trabalho reais e benchmarks padronizados, permitindo uma avaliação mais precisa e contextualizada do desempenho do sistema. Outra possibilidade relevante é explorar arquiteturas multicore, analisando os efeitos da coerência entre caches, o compartilhamento de recursos e as estratégias de sincronização, aspectos fundamentais em sistemas contemporâneos de alto desempenho.

REFERENCES

- [1] Moraes, M. P.; SOUZA, P. S. L.; Bruschi, S.M. Usando Arquivos de Rastro no Projeto Amnesia. São Carlos: IX Simpósio Internacional de Iniciação Científica da Universidade de São Paulo (SIIC/USP), ICMC – USP 2011. (scientific initiation abstract).
- [2] Santos, J.H.; SOUZA, P. S. L.; Bruschi, S.M.; Souza, S.R.S. Amnesia: Um Simulador de Hierarquia de Memória – Módulo Processador. São Carlos: Grupo PET do ICMC/USP, 2007 (scientific initiation abstract).
- [3] L. Yen et al., "LogTM-SE: Decoupling Hardware Transactional Memory from Caches," 2007 IEEE 13th International Symposium on High Performance Computer Architecture, Scottsdale, AZ, USA, 2007, pp. 261-272, doi: 10.1109/HPCA.2007.346204.
- [4] Jih-Kwon Peir, Shih-Chang Lai, Shih-Lien Lu, Jared Stark, and Konrad Lai. Bloom Filtering Cache Misses for Accurate Data Speculation and Prefetching. In Proc. of the 2002 International Conference on Supercomputing, pages 189-198, June 2002.
- [5] Oliveira, B.H.; Santos, J.H.; SOUZA, P. S. L.; Bruschi, S.M.; Souza, S.R.S. Amnesia: Um Simulador de Hierarquia de Memória. In: International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD) / Workshop sobre Educação em Arquitetura de Computadores (WEAC), Campo Grande. Proceedings of 20th SBAC-PAD. IEEE Computer Society, 2008. v. 1. p. 13-16.

- [6] Buzo, G.R.; Bruschi, S.M.; SOUZA, P. S. L.; Souza, S.R.S. Amnesia: Um Simulador de Hierarquia de Memória – Módulo Memória Virtual. São Carlos: Grupo PET do ICMC/USP, 2007 (scientific initiation abstract).
- [7] TIOSSO, F.; Bruschi, S.M.; SOUZA, P. S. L.; BARBOSA, E. F. Amnesia: um Objeto de Aprendizagem para o Ensino de Hierarquia de Memória. Proceedings of the 25º Simpósio Brasileiro de Informática na Educação (SBIE 2014), Dourados, Sociedade Brasileira de Computação, 2014. v. 1. p. 1-10.