



INF01113

Organização De Computadores B

Segundo Trabalho

Benchmarks de Organizações de Computadores

Bruno Alexandre Hofstetter Bourscheid (00550177)

Fernando Longhi de Andrade (00580366)

Luiz Augusto Ponzoni Schmidt (00580108)

Miguel Dutra Fontes Guerra (00342573)

Pedro Lubaszewski Lima (00341810)

Turma B

Bubble Sort

- Complexidade: $O(n^2)$;
- Algoritmo de ordenação simples;
- Muitas comparações e trocas;
- Baixo uso de memória;
- Pouco paralelismo (dependência entre as instruções).

Fast Fourier Transform (FFT)

- Complexidade: $O(n \log(n))$;
- Operações matemáticas complexas;
- Várias instruções de ponto flutuante;
- Acesso regular à memória;
- Alto paralelismo.

Multiplicação de Matrizes

- Complexidade: $O(n^3)$;
- Poucas instruções;
- Bastante acesso à memória contígua (depende da *cache*);
- Depende da latência de acesso à memória.

1

Tamanho da *Cache* L1

Representa a quantidade de dados mais próximos da *CPU*. Isso pode afetar o desempenho de programas que dependam de dados com localidade temporal e espacial, afetando o número de *cache misses*.

2

Associatividade da *Cache* L1

É o número de blocos de cada conjunto de memória *cache*. Também deve afetar a questão de programas com dados que apresentam localidade espacial, através de mais ou menos *cache misses*.

3

Tamanho do *Fetch Buffer*

Um maior tamanho de *fetch buffer* implica em mais instruções sendo processadas simultaneamente. Ou seja, deve afetar algoritmos que apresentem maior paralelismo de instruções.

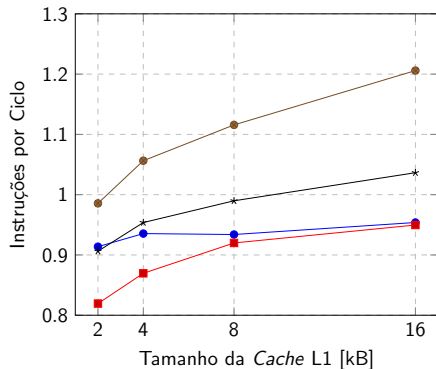
Como configuração fixa inicial, utilizou-se os seguintes parâmetros:

- Tamanho da *Cache* L1: 16kB;
- Associatividade da *Cache* L1: 8-way;
- Tamanho do *Fetch Buffer*: 64bytes;

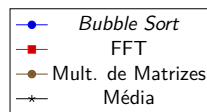
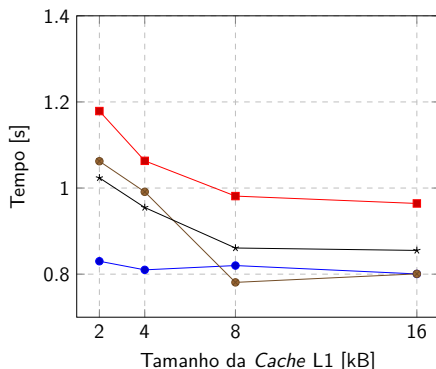
Os demais parâmetros de configuração da máquina foram mantidos.

Resultados mudando o tamanho de *cache* L1

IPC



Tempo de Execução



Análise do IPC:

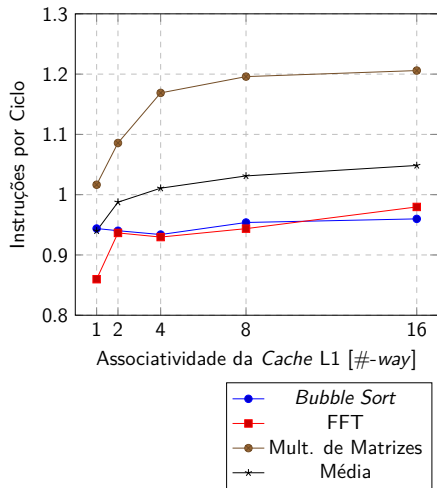
- Quase todos os algoritmos foram beneficiados ;
- Curva acentuada na mult. de matrizes (alta localidade temporal e espacial);
- FFT se beneficiou bastante também (boa localidade, porém menos previsível que a mult. de matrizes);
- *Bubble Sort* se beneficiou pouco (pouca localidade espacial);
- IPC médio entre as aplicações segue um crescimento aproximadamente logaritmico (mostrando um bom benefício do aumento do parâmetro).

Análise do Tempo de Execução:

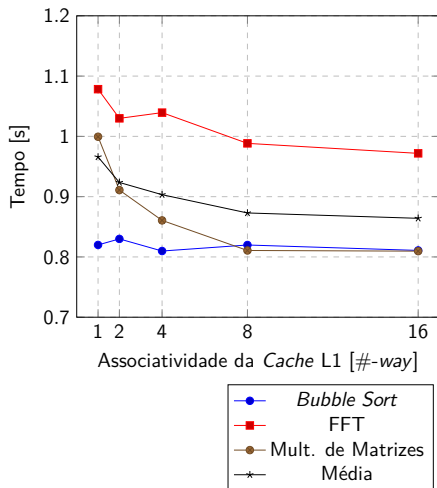
- Diminuição de *cache misses* (diminuição de tempo gasto com acesso à memória);
- Comportamento similar ao de IPC;
- Quase todos os algoritmos apresentam pelo menos um pouco de localidade espacial (mult. de matrizes mostra temporal também);
- A média dos tempos também segue um comportamento similar ao decrescimento da média do IPC.

Resultados mudando a associatividade da cache L1

IPC



Tempo de Execução



Análise do IPC:

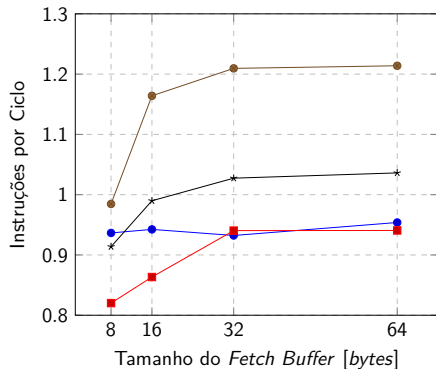
- Quase todos os programas foram beneficiados;
- Mult. de matrizes apresenta altos saltos nos primeiros aumentos e estabiliza em um platô (sofre com conflitos de *cache*);
- FFT apresentou os mesmos resultados que a mult. de matrizes, porém um pouco menos acentuados (mesmas justificativas);
- Já o *Bubble Sort* apresentou resultados mais sutis que a FFT, com uma curva mais plana (por conta da menor localidade do algoritmo);
- IPC médio entre as aplicações segue um crescimento aproximadamente logaritmico (mostrando um bom benefício do aumento do parâmetro).

Análise do Tempo de Execução:

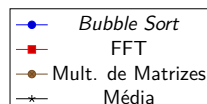
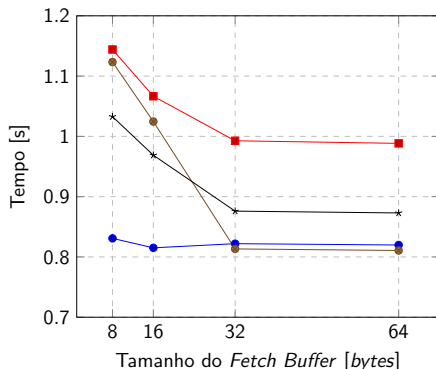
- Diminuição de *cache misses* devido a conflitos;
- Comportamento similar ao de IPC;
- Há algumas leves variações de 2-way para 8-way na FFT que podem se relacionar com uma menor previsibilidade de acessos à memória em comparação à mult. de matrizes;
- A média dos tempos segue um comportamento similar ao decrescimento da média do IPC.

Resultados mudando o tamanho do *fetch buffer*

IPC



Tempo de Execução



Análise do IPC:

- Aumento de desempenho quase generalizado (excluindo o *Bubble Sort*);
- Mult. de matrizes apresenta altos saltos nos primeiros aumentos e estabiliza a partir de um certo ponto (o problema passa a estar relacionado com a *cache*);
- FFT apresentou os mesmos resultados que a mult. de matrizes, mas variação total menor (mesmas justificativas);
- Já o *Bubble Sort* apresentou resultados mais sutis que a FFT, com uma curva mais estável (possui poucas instruções previsíveis e o *pipeline* básico consegue dar conta);
- IPC médio entre as aplicações segue um crescimento suave que estabiliza (mostrando um bom benefício do aumento do parâmetro).

Análise do Tempo de Execução:

- Benefício para programas com fluxo de controle linear (poucas ramificações);
- Comportamento bem parecido ao de IPC;
- A média dos tempos segue um comportamento similar ao decrescimento da média do IPC;
- Conclui-se que o comportamento em relação ao tempo de execução é proporcional ao inverso do comportamento do IPC.