

Extensão de um elemento do artigo *Accelerating Decoupled Look-ahead via Weak Dependence Removal: A Metaheuristic Approach*

Gustavo Ciotto Pinton



Universidade Estadual de Campinas - UNICAMP
MO601B - Arquitetura de Computadores

18 de Novembro de 2016

Sumário

Arquitetura Decoupled Look-Ahead

Projeto 3

Projeto 4

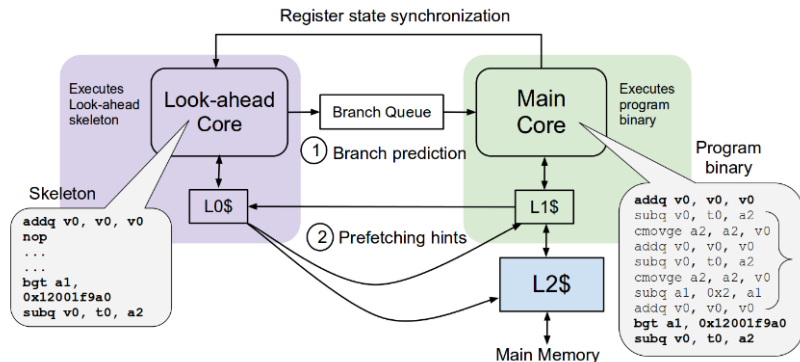
Conclusões

Referências

Arquitetura *Decoupled Look-Ahead*

Relembrando...

- ▶ *Parser* que transforma o binário do programa principal em uma versão reduzida, somente para procurar *misses*.
- ▶ Versão esqueleto roda em *core* separado, anteriormente ao programa principal.
- ▶ Os resultados de saltos condicionais são transmitidos através de uma fila para o *core*.



Proposta do artigo

Relembrando...

- ▶ Constatou-se que a *thread* auxiliar, isto é, a *look ahead thread* se tornou o novo limite de velocidade do sistema.
- ▶ A corretude da *look-ahead thread* não é exigida, permitindo várias otimizações
 - ▶ Dependências fracas: instruções que contribuem marginalmente para o resultado e, portanto, podem ser retiradas.
- ▶ O artigo propõe uma maneira de otimizá-la a partir de algoritmos genéticos:
 - ▶ Identificação dos pontos desnecessários que poderiam ser retirados desta *thread*.
 - ▶ Caracterização de genes e cromossomos.

Projeto 3

Relembrando...

- ▶ Reprodução das curvas *ideal* e *single-thread* da figura 3.

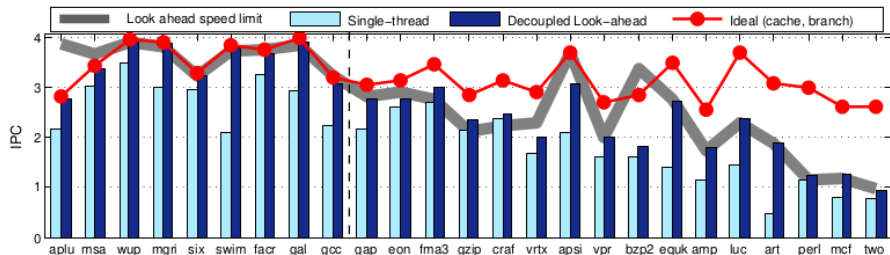


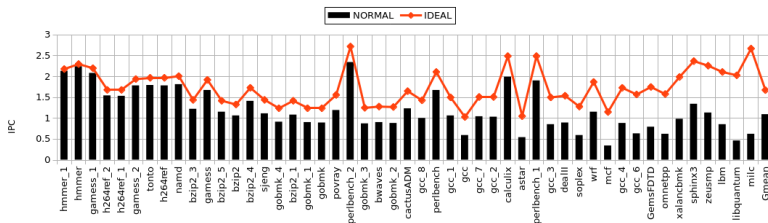
Figure 3. Performance comparison of 4 configurations. Shown in the bars are baseline single core (left) and a decoupled look-ahead system (right). Two upper-bounds are shown: the performance of a single core with idealized branch predictions and perfect cache accesses (curve with circles), and the approximate speed limit of the look-ahead thread (gray wide curve indicating approximation). The applications are sorted with increasing performance gap between the decoupled look-ahead system and the prediction- and accesses-idealized single-core system.

Projeto 3

Resultados

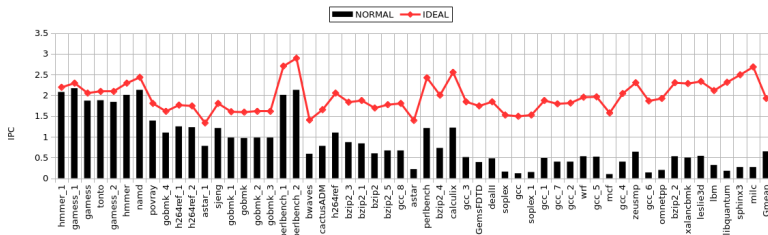
IPC dos benchmarks do SPEC CPU2006

Configuração do Arquivo de Configuração Padrão Gaiwestown



IPC dos benchmarks do SPEC CPU2006

Configuração do Artigo



Projeto 4

Introdução

- ▶ Parâmetro modificado: tamanho das páginas de memória:
 - ▶ 4KB e 4MB.
 - ▶ Projeto 2: número de acessos adicionais à memória.
 - ▶ Projeto 4: efeitos da alteração no IPC.
- ▶ Modificações em relação ao projeto 3:
 - ▶ *Pinballs* com 100M *warmup* e 30M região detalhada: recálculo de todos os resultados anteriores.
 - ▶ Problema com a propriedade `perf_model/dram/latency`: *sniper* a interpreta como nanossegundos e não como ciclos.

Projeto 4

Implementação

- ▶ Classe TLB do *sniper* modificada:
 - ▶ 3 constantes importantes: `SIM_PAGE_SHIFT`, `SIM_PAGE_SIZE` e `SIM_PAGE_MASK`.
 - ▶ `SIM_PAGE_SIZE = (1 « SIM_PAGE_SHIFT)`
 - ▶ `SIM_PAGE_MASK = ~ (SIM_PAGE_SHIFT - 1)`
 - ▶ Por padrão: `SIM_PAGE_SHIFT = 12` (4KB)
 - ▶ Alterações no construtor da classe.
- ▶ Criação de uma nova opção `perf_model/tlb/page_size_bits`:
 - ▶ Classe `MemoryManager` lê essa propriedade e instancia TLBs.
- ▶ Estrutura das *caches*:
 - ▶ 2 níveis.
 - ▶ I-TLB: 128 entradas e *4-way associative*.
 - ▶ D-TLB: 64 entradas e *4-way associative*.
 - ▶ STLB: TLB compartilhada com 512 entradas e *4-way associative*.

Projeto 4

Resultados

IPC dos benchmarks do CPU2006

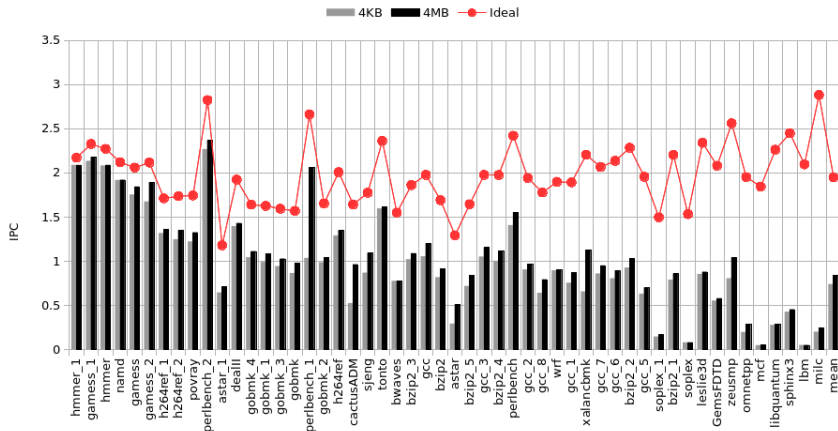


Figura: IPCs utilizando páginas de 4KB e 4MB

Projeto 4

Resultados

Diferença no IPC entre páginas de 4MB e 4KB

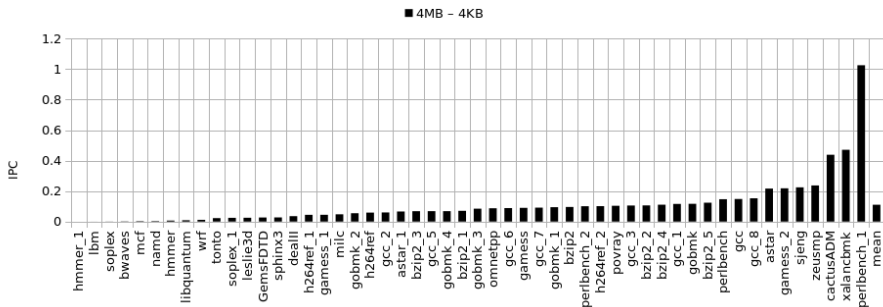


Figura: Diferença nos resultados obtidos nas páginas de 4KB e 4MB

Projeto 4

Resultados

Diferença no IPC para diferentes Pinballs

Curvas Ideias

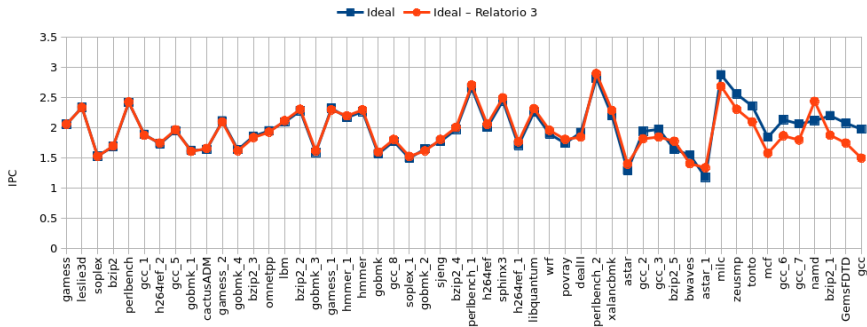


Figura: Diferença das curvas ideais calculadas em dois tipos de *pinballs*

Projeto 4

Resultados

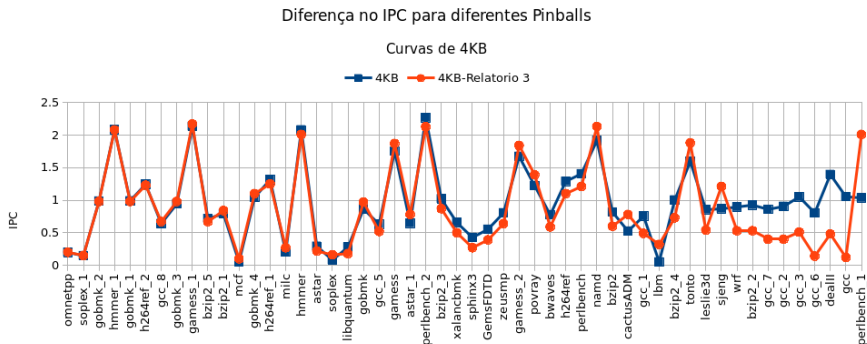


Figura: Diferença das curvas de 4KB calculadas em dois tipos de *pinballs*

Conclusões

- ▶ As curvas de 4KB e 4MB se distanciam da ideal para *benchmarks* com grande *memory footprints*.
- ▶ A modificação do tamanho das páginas da memória não é um fator determinante para o IPC: obtivemos um ganho, em média, de apenas 0.11.
- ▶ Diferença entre as execuções do *pinball*:
 - ▶ Caso ideal: pouca divergência, já que os mesmos intervalos de *cache misses* e *mispredictions* foram retirados.
 - ▶ 4KB: maior divergência, principalmente para os *benchmarks* com maiores *footprints* de memória.

Referências

- ▶ Carison, T. E. (2012). Interval simulation.
http://snipersim.org/w/Interval_Simulation.
- ▶ Carison, T. E. and Heirman, W. (2013). The Sniper User Manual.
- ▶ Parihar, R. and Huang, M. C. (2014). Accelerating decoupled look-ahead via weak dependence removal: A metaheuristic approach. International Symposium on High Performance Computer Architecture.
- ▶ Henning, J. L. (2007). SPEC CPU2006 Memory Footprint, ACM SIGARCH Computer Architecture News