New Visual Characterization
Graphs for Memory System
Analysis and Evaluation

Edson T. Midorikawa
edson.midorikawa@poli.usp.br

Hugo Henrique Cassettari
hugohc@terra.com.br

LAHPC - PCS
Escola Politécnica da USP
Topics

• Introduction

• Previous work

• New Proposed Graphs

• Case study: performance of adaptive page replacement algorithms

• Conclusion and Future works
Introduction

- Memory systems (still) are one of the most important components of modern computer systems.

- They work based on the *Principle of Locality of References*.

- Workload characterization is essential for memory management research.
Previous Work

• WSO 2004
  – **LRU-WAR** (*LRU with Working Area Restriction*)

• WSO 2005
  – **3P** (*Three Pointers*)

• Performance comparison among algorithms:
  – FIFO, LRU, Clock, ...
  – FBR, 2Q, ...
  – SEQ, EELRU, LIRS, ARC, CAR, CART
A simple question

Why algorithm A has better performance than algorithm B for program P under condition C?
Previous Work

Memory access map

Locality surface
Objective

• Introduce new visual resources to study memory access behavior, workload characterization, and performance prediction.

• Present a case study with the proposed visual characterization graphs.
New Proposed Graphs

- We introduce 6 new visual characterization graphs:
  - Memory access surface
  - IRG graph
  - IRR graph
  - IRR surface
  - IRR histogram
  - IRR cumulative graph
Case Study

- We present a case study of *performance evaluation of page replacement algorithms* for modern virtual memory systems
  - *Traditional*: FIFO, LRU, FBR, 2Q
  - *Adaptive*: ARC, EELRU, LIRS, LRU-WAR
  - *Online*: Clock, CAR, CART, 3P

- Applications (trace files):
  - *Compress*: file compression utility.
  - *Espresso*: a circuit simulator.
  - *Grobner*: a formula-rewrite program.
  - *Sprite*: from the Sprite network file system.
Case Study

- Simulation results

<table>
<thead>
<tr>
<th>Trace</th>
<th>FIFO</th>
<th>LRU</th>
<th>FBR</th>
<th>2Q</th>
<th>ARC</th>
<th>EELRU</th>
<th>LIRS</th>
<th>LRU-WAR</th>
<th>CLOCK</th>
<th>CAR</th>
<th>CART</th>
<th>3P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compress</td>
<td>191.97%</td>
<td>121.07%</td>
<td>96.24%</td>
<td>94.78%</td>
<td>120.81%</td>
<td>72.63%</td>
<td>122.41%</td>
<td>79.48%</td>
<td>145.99%</td>
<td>127.91%</td>
<td>103.95%</td>
<td>94.66%</td>
</tr>
<tr>
<td>Espresso</td>
<td>174.85%</td>
<td>67.73%</td>
<td>62.88%</td>
<td>80.27%</td>
<td>68.19%</td>
<td>62.24%</td>
<td>66.84%</td>
<td>86.88%</td>
<td>114.98%</td>
<td>114.14%</td>
<td>102.99%</td>
<td>93.60%</td>
</tr>
<tr>
<td>Gnobner</td>
<td>233.98%</td>
<td>139.12%</td>
<td>115.68%</td>
<td>116.53%</td>
<td>139.17%</td>
<td>72.91%</td>
<td>113.86%</td>
<td>125.40%</td>
<td>139.20%</td>
<td>139.34%</td>
<td>118.21%</td>
<td>108.06%</td>
</tr>
<tr>
<td>Sprite</td>
<td>33.29%</td>
<td>17.70%</td>
<td>17.00%</td>
<td>25.21%</td>
<td>22.93%</td>
<td>19.04%</td>
<td>40.72%</td>
<td>21.36%</td>
<td>18.67%</td>
<td>23.68%</td>
<td>26.27%</td>
<td>25.13%</td>
</tr>
</tbody>
</table>
Memory access surface
Case Study

• Grobner
  – It uses almost all pages in the virtual address space.
  – High access density:
    • Sequential access pattern.
    • Temporal reuse of low address pages.
  – Good results for page replacement algorithms that detect sequential patterns.
Definitions

- **IRG (inter-reference gap)**
  - Time distance between successive references of a memory page.

- **IRR (inter-reference recency)**
  - reuse distance or recency
  - number of other unique pages accessed between two consecutive references to the same memory page.
IRG Graph
IRR Graph

![Graph showing Inter-Reference Record (IRR) over Virtual Time (memory accesses).]
IRR Surface
Case Study

• Sprite

  - **IRG graph**: most of accesses occur in virtual time intervals between 1 and 5,000 references.
  - **IRR graph + IRR surface**: most accesses occur in the first 1,000 positions in the LRU stack (15% of the footprint).

  - It presents pages with strong temporal locality being alternating with less accessed pages:
    - Not favorable to adaptive algorithms.
IRR Histograms

- IRR Histogram
- Cumulative Histogram

WSO 2006
Case Study

• Compress
  
  – A memory size equal to 8 pages is enough to maintain the page fault rate as low as 18.7%.
  – With 64 pages: 1.45%.

  – It presents high temporal locality;
  – It provides more accurate distinction among different processing phases by adaptive algorithms:
    • Good performance of adaptive algorithms.
Conclusion

• In this paper we introduced six new graphs for studying locality of references.

• Visual aspects:
  – Memory access patterns;
  – Temporal and spatial localities;
  – Real distribution of memory accesses;
  – Reuse frequency;
  – LRU stack position.

• The performance of page replacement algorithms was analyzed using these new graphs.
Future Work

• More case studies
  – Parallel applications

• New metric: \textbf{IRR-n} (number of distinct pages referenced among n+1 consecutive accesses to the same page).

• Integration and enhancement of the tools available in \textit{Elephantools}.
Thank you.

Questions?
Extra slides
Locality Surface

- Technique to quantify temporal and spatial locality in programs
- Introduced to cache memory studies.
- Characterize the memory accesses taking into account the complete program:
  - Z-axis: number of occurrences of specific delay (stack distance) and stride (difference between memory addresses) values obtained among inter-referenced pages
  - X-axis:
  - Y-axis:
Case Study

- Simulation experiments

<table>
<thead>
<tr>
<th>Trace</th>
<th>Number of Distinct Pages</th>
<th>Memory Sizes Considered in Simulations</th>
<th>Number of Simulations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compress</td>
<td>396</td>
<td>10, 15, 20, ..., 385, 390, 395</td>
<td>78</td>
</tr>
<tr>
<td>Espresso</td>
<td>77</td>
<td>10, 11, 12, ..., 73, 74, 75</td>
<td>66</td>
</tr>
<tr>
<td>Grobner</td>
<td>67</td>
<td>10, 11, 12, ..., 63, 64, 65</td>
<td>56</td>
</tr>
<tr>
<td>Sprite</td>
<td>7075</td>
<td>100, 200, 300, ..., 6800, 6900, 7000</td>
<td>70</td>
</tr>
</tbody>
</table>