CSCC69

Yinan Gong, Chentao Tang, Tarunkumar Reddy Yellu

March 2017

1 Comparion Between Algorithms

We compared program outputs for running the 5 different algorithms: Exact LRU, OPT, CLOCK, FIFO, and RAND, on 4 trace files: simpleloop, matmul, blocked, and is-palindrome.

We first analyze the hit accuracy results for different algorithms, in general we noticed:

1. The RAND algorithm performed the worst in most cases, except in 2 edge cases where one had significantly and other marginally better performances
2. The FIFO algorithm came second last in the general comparison, it was significantly better than RAND, but only marginally worse than the other 3 algorithms.
3. The CLOCK algorithm always performed better than the FIFO algorithm by a reasonable amount, and is only surpassed by LRU by an insignificant amount
4. The LRU algorithm was almost identical to the CLOCK algorithm, only exceeding it by a minuscule number.
5. The OPT algorithm ran the best, and it performed better than the other algorithms by a considerable margin.

The above conclusions are well within our expectations, and we now look at the reasons behind it.

The OPT algorithm is effectively "predicting the future", and as a result it can perform the most optimized replacements at anytime during the program execution. However this is not realistic in practical applications, since these replacement algorithms differ in terms of a trade off between hit accuracy vs resource, and in the face of OPT, it is completely skewed to hit accuracy, while sacrificing computing resources.

We notice that CLOCK and LRU are almost identical, this is also normal since the algorithms are fundamentally similar and run in a comparable fashion.

Another interesting behaviour we noticed is that the hit rate for algorithms grow logarithmic pattern in relation to the memory size, as in, the impact on the hit rate when going from $150- > 200$ is much lower than from $50- > 100$.

2 LRU algorithm impact on Memory

For the LRU algorithm, as the memory size increases, the hit rate increases as well (log order), while the miss count, eviction count, dirty eviction count and clean eviction count decreases. This is reasonable since the bigger the memory size, the better for LRU as the algorithm gains increased space to store information. And since this algorithm is based on the least recently used, this means it can store more least recently used memory address, enabling the algorithm to perform better.

3 Result Tables ordered by Algorithm

LRU

simpleloop

memsize	50	100	150	200
hit rate	74.1562	75.1484	75.1484	75.1484
hit count	7997	8104	8104	8104
miss count	2787	2680	2680	2680
overall eviction count	2737	2580	2530	2480
clean eviction count	97	0	0	0
dirty eviction count	2640	2580	2530	2480

blocked

memsize	50	100	150	200
hit rate	99.7937	99.8485	99.8486	99.8536
hit count	2520046	2521429	2521434	2521558
miss count	5210	3817	3822	3698
overall eviction count	5160	3727	3672	3498
clean eviction count	2814	2648	2613	2439
dirty eviction count	2346	1079	1059	1059

matmul

memsize	50	100	150	200
hit rate	64.9228	66.0946	98.8923	98.8926
hit count	1927167	1961952	2935518	2935529
miss count	1041233	1006448	32882	32871
overall eviction count	1041183	1006348	32732	32671
clean eviction count	1040080	1005271	31655	31594
dirty eviction count	1103	1077	1077	1077

is_palindrome

memsize	50	100	150	200
hit rate	99.9976	99.9986	99.9986	99.9986
hit count	8817646	8817730	8817731	8817731
miss count	210	126	125	125
overall eviction count	160	26	0	0
clean eviction count	52	0	0	0
dirty eviction count	108	26	0	0

Opt

simpleloop

memsize	50	100	150	200
hit rate	75.2318	75.5471	75.5471	75.5471
hit count	8113	8147	8147	8147
miss count	2671	2637	2637	2637
overall eviction count	2621	2537	2487	2437
clean eviction count	26	0	0	0
dirty eviction count	2595	2537	2487	2437

blocked

memsize	50	100	150	200
hit rate	99.8519	99.8799	99.8996	99.9095
hit count	2521516	2522224	2522721	2522971
miss count	3740	3032	2535	2285
overall eviction count	3690	2932	1385	2085
clean eviction count	2607	1859	1315	1020
dirty eviction count	1083	1073	1070	1065

matmul

memsize	50	100	150	200
hit rate	80.2097	96.8739	99.1035	99.3511
hit count	2380945	2875606	2941789	2949139
miss count	587455	92794	26611	19261
overall eviction count	587405	92694	26461	19061
clean eviction count	586324	91614	25382	17982
dirty eviction count	1081	1080	1079	1079

is_palindrome

memsize	50	100	150	200
hit rate	99.9983	99.9986	99.9986	99.9986
hit count	8817710	8817731	8817731	8817731
miss count	146	125	125	125
overall eviction count	96	25	0	0
clean eviction count	11	0	0	0
dirty eviction count	85	25	0	0

Clock

simpleloop

memsize	50	100	150	200
hit rate	74.0912	75.1205	75.1391	75.1391
hit count	7990	8101	8103	8103
miss count	2794	2683	2681	2681
overall eviction count	2744	2583	2531	2481
clean eviction count	103	2	0	0
dirty eviction count	2641	2581	2531	2481

blocked

memsize	50	100	150	200
hit rate	99.7708	99.8353	99.8485	99.8718
hit count	2519469	2521097	2521430	2522018
miss count	5787	4159	3826	3238
overall eviction count	5737	4059	3676	3038
clean eviction count	3326	2665	2616	1970
dirty eviction count	2411	1394	1060	1068

matmul

memsize	50	100	150	200
hit rate	64.9225	66.2516	98.8316	98.8922
hit count	1927160	1966613	2933718	2935516
miss count	1041240	1001787	34682	32884
overall eviction count	1041190	1001687	34532	32684
clean eviction count	1040086	1000609	33454	31607
dirty eviction count	1104	1078	1078	1077

is_palindrome

memsize	50	100	150	200
hit rate	99.9975	99.9985	99.9986	99.9986
hit count	8817635	8817725	8817731	8817731
miss count	221	131	125	125
overall eviction count	171	31	0	0
clean eviction count	56	0	0	0
dirty eviction count	115	31	0	0

FIFO

simpleloop

memsize	50	100	150	200
hit rate	70.9848	74.5364	75	75.0742
hit count	7655	8038	8088	8096
miss count	3129	2746	2696	2688
overall eviction count	3079	2646	2546	2488
clean eviction count	418	76	32	24
dirty eviction count	2661	2570	2514	2464

blocked

memsize	50	100	150	200
hit rate	98.4211	99.7589	99.8208	99.8393
hit count	2485384	2519167	2520732	2521199
miss count	39872	6089	4524	4057
overall eviction count	39822	5989	4374	3857
clean eviction count	36555	3948	2711	2532
dirty eviction count	3267	2041	1663	1325

matmul

memsize	50	100	150	200
hit rate	62.0241	63.4974	98.8408	98.8584
hit count	1841124	1884856	2933991	2934513
miss count	1127276	1083544	34409	33887
overall eviction count	1127226	1083444	34259	33687
clean eviction count	1083237	1061225	32946	32435
dirty eviction count	43989	22219	1313	1252

is_palindrome

memsize	50	100	150	200
hit rate	99.9968	99.9984	99.9986	99.9986
hit count	8817572	8817711	8817731	8817731
miss count	284	145	125	125
overall eviction count	234	45	0	0
clean eviction count	89	0	0	0
dirty eviction count	145	45	0	0

Rand

simpleloop

memsize	50	100	150	200
hit rate	72.0883	74.4622	74.8053	74.9073
hit count	7774	8030	8067	8078
miss count	3010	2754	2717	2706
overall eviction count	2960	2654	2567	2506
clean eviction count	271	49	22	18
dirty eviction count	2689	2605	2545	2488

blocked

memsize	50	100	150	200
hit rate	99.6708	99.7931	99.8263	99.8456
hit count	2516942	2520030	2520869	2521356
miss count	8314	5226	4387	3900
overall eviction count	8264	5126	4437	3700
clean eviction count	5767	3406	2764	2369
dirty eviction count	2497	1720	1473	1331

matmul

memsize	50	100	150	200
hit rate	66.4608	89.1236	96.7645	98.096
hit count	1972821	2645544	2872357	2911882
miss count	995579	322856	96043	56518
overall eviction count	995529	322756	95893	56318
clean eviction count	956271	315329	93552	54702
dirty eviction count	39258	7427	2341	1616

is_palindrome

memsize	50	100	150	200
hit rate	99.9964	99.9984	99.9986	99.9986
hit count	881739	8817714	8817731	8817731
miss count	317	142	125	125
overall eviction count	267	42	0	0
clean eviction count	123	4	0	0
dirty eviction count	144	38	0	0