Matrix experiment

Test program

The test program performs matrix transposition and evaluates given implementation:

- the cache-oblivious matrix transposition
- the trivial algorithm which transposed by definition.

It evaluates the implementation on simulated caches of m items organized in b-item blocks and matrices of different sizes n.

The simulated cache is fully associative and uses LRU replacement strategy.

- m1024-b16
- m8192-b64
- m65536-b256
- m65536-b4096

Performance

Performance is measured in terms of the average number of cache misses per item.

The diagonal items are not counted.

Plots

Each plot shows the dependence of the average number of misses on the matrix size n:

- the cache-oblivious algorithm: the green curve
- the trivial algorithm: the black curve

Theoretical facts

Number of memory transfers is:

- $O(n^2/b)$ for the cache-oblivious algorithm, $(m / b) < n$
- $O(n^2)$ for the trivial algorithm, $m > b^2$

The cache-oblivious algorithm

The average number of misses on the matrix size n: O(1/b).

- the number of items: n^2
- the number of transfers: $O(n^2/b)$
- $(n^2/b) / n^2 = (1 / b)$

The number of misses is 0 on the matrix size n, if $n^2 < M$ (the whole matrix can be stored in cache).

	1 / b (approx.)	n: $n = 2^{k,}$, $2^{2k} = m$
m1024-b16	0,0625	$2^{5,}$
m8192-b64	0.0156	2^6
m65536-b256	0.0039	2^8
m65536-b4096	0,0002	2^8

The experiment result:

- For n: $n^2 <= m$:
  The average number of misses is 0.

- For n: $n^2 > m$:
  - $n = 2^k$:
    The average number of misses on the matrix size n is 1 / b.

  - $2^k < n < 2^{k,}$:
    The average number of misses is smaller than $1 / 2^{\log(b)-1}$.

The trivial algorithm

The average number of misses on the matrix size n: O(1).

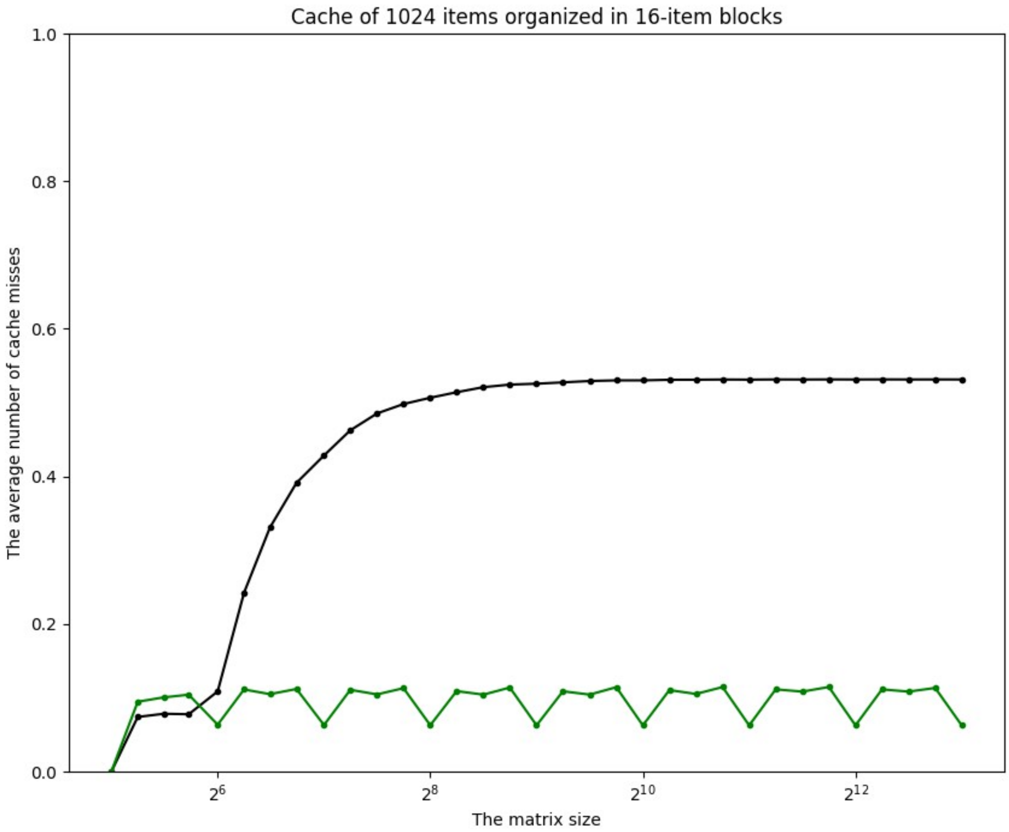
- the number of items: n^2
- the number of transfers: $O(n^2)$
- $n^2/ n^2 = 1$

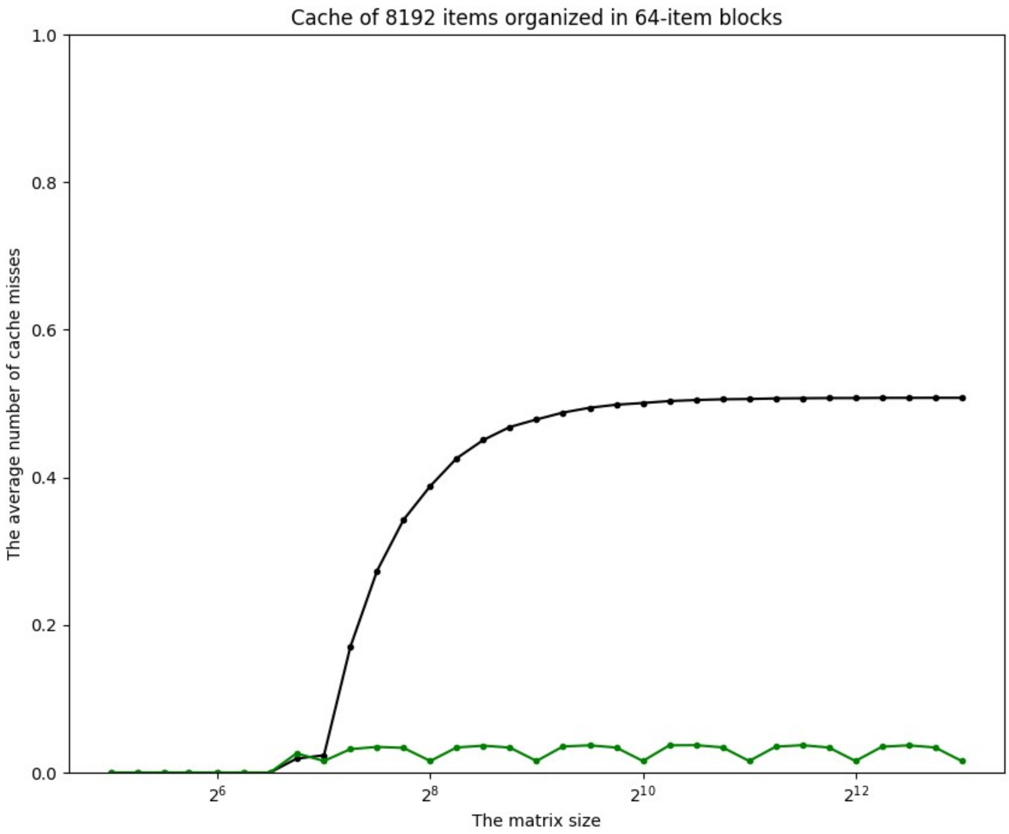
The number of misses is 0 on the matrix size n, if $n^2 < M$ (the whole matrix can be stored in cache).

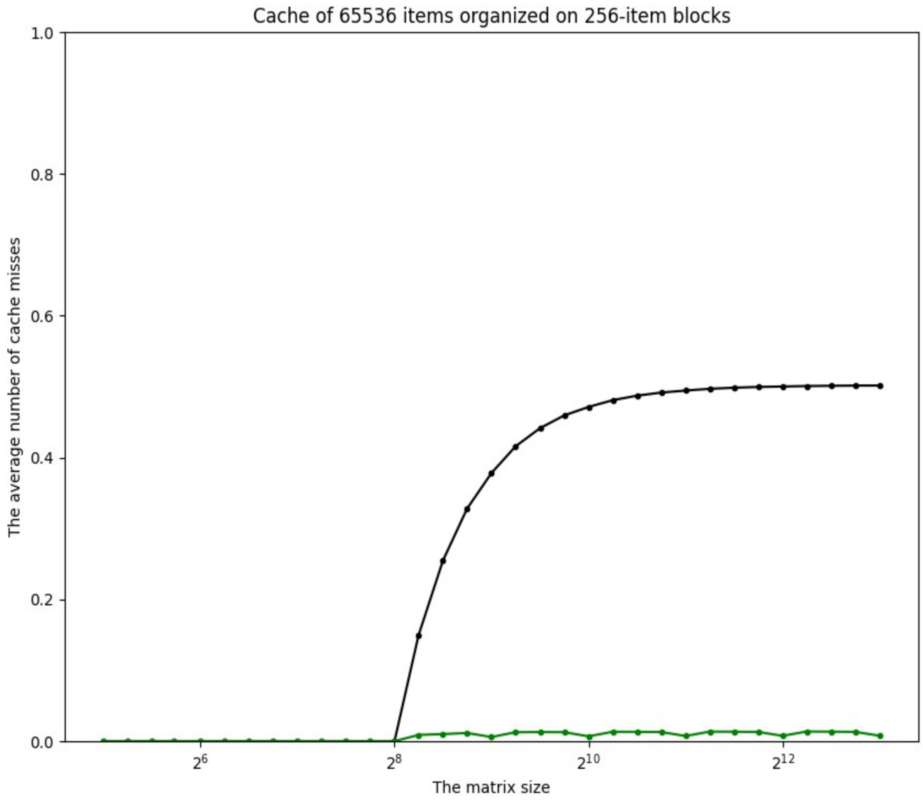
If the whole matrix can not be stored in cache, the average number of misses is at least ½:

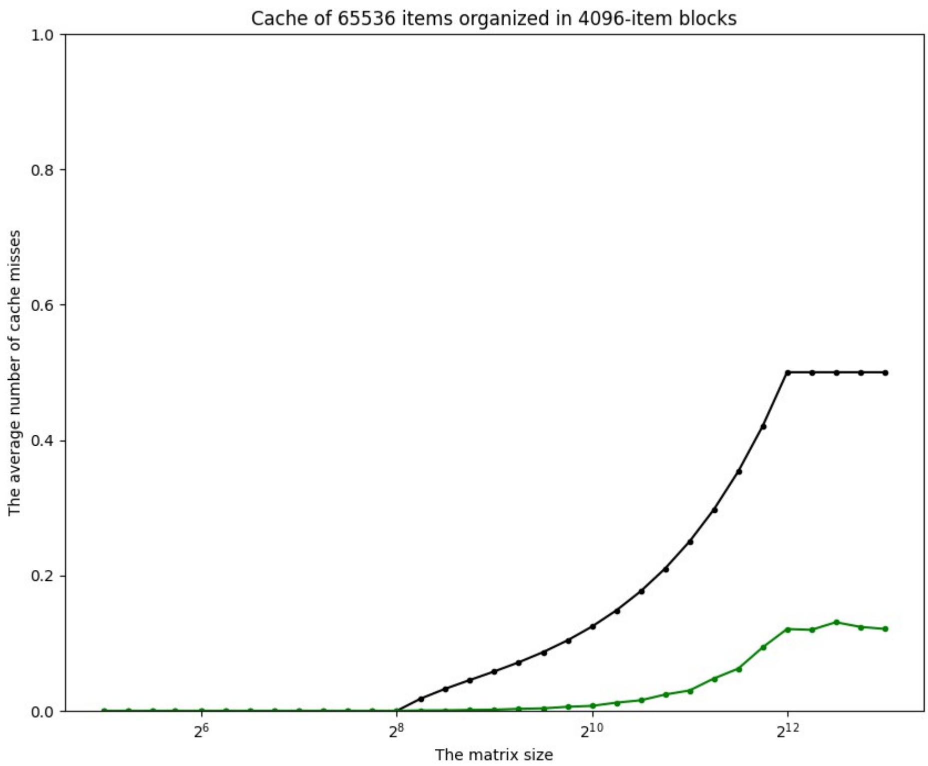
- We need one memory transfer for each transposition of pair of matrix elements a(i, j) and a(j, i), i < j.

    - Consider the block that contains a(j, i-1) and a(j, i):
        - the block was transferred when a(i-1, j) and a(j, i-1) were transposed.
        - than (n – 1) other blocks were transferred:
          one block for each of elements a(j +1, i-1), ..., a(n, i-1), a(1, i), …, a(j-1, i).
        - when x is transposed, the block is no longer cached and we need to transfer it again.


Cache of 1024 items organized in 16-item blocks
The average number of cache misses
0.0
0.2
0.4
0.6
0.8
1.0
2^6
2^8
2^{10}
2^{12}
The matrix size


Cache of 8192 items organized in 64-item blocks
The average number of cache misses
0.0
0.2
0.4
0.6
0.8
1.0
2^6
2^8
2^{10}
2^{12}
The matrix size


Cache of 65536 items organized on 256-item blocks
The average number of cache misses
The matrix size
1.0
0.8
0.6
0.4
0.2
0.0
2^6
2^8
2^{10}
2^{12}


Cache of 65536 items organized in 4096-item blocks
The average number of cache misses
The matrix size
1.0
0.8
0.6
0.4
0.2
0.0
2^6
2^8
2^{10}
2^{12}