NTIN066 - solutions 6

David Mareček

April 9, 2024

1. What happens in the tiled transposition for a cache-aware model with a weaker tall-cache assumption, i.e. when we do not have the cache size $M \geq 4B^2$, but only $M \geq B^2$?
Would it change the I/O complexity?

No. We can use four times smaller tiles.

2. Prove that the "divide and conquer" algorithm for the cache-oblivious matrix transposition with naive recursion into 4 transpose and one swap operation lead to worse time complexity.

Number of swaps in each level of recursion would be $N^2/4$, so it is $O(N^2 \log N)$ in total.

3. Analyze what happens in a recursive transposition algorithm if the size of the matrix is not a power of two.
Prove that sub-problems still work with almost square matrices.

All the submatrices are always either in $k \times k$ or $k \times (k+1)$ format.

4. Think about cache-oblivious strategy for matrix multiplication.

See https://en.algorithmica.org/hpc/external-memory/oblivious/