

Collaborators: Sahitya Mannepalli (u1418271)
: Kavita (u1418910)

OpenMP:

For atb case,
we have tried loop unrolling, and loop permutations

In case tiling we tried for various tiling values.
And also various un-rolling factors.

Of all the attempts, parallelizing "i" loop &
unrolling for a degree of 2 gave better results for some test
cases, and unrolling for a degree of 4 gave better performance for
other test cases.

When N_k is very high when compared to N_i, N_j unrolling

by a factor of 2 is better, in all other cases, unrolling
by a factor of 4 is better.

For atbt,
we tried tiling, unrolling, and loop permutation.

In the end, tiling (i,j) and the loop permutation ikj
gave better performance.

The performance reports for these has been attached.

In case AMD CPUs overall multi-threaded performance is good where as
for Intel CPUs single thread

GPU:

For atbt:

We tried using multiple block sizes, k-loop unrolling and
shared memory.

k-loop unrolling is performing better when we do un-rolling
else shared-memory version is giving better results.

A block size of 8 is giving better results.

For atb:

Shared Memory implementation is giving better results.

We tried multiple test block sizes and a block size
of 8 is giving better performance.

Overall, The performance of the code in case of GPU (Guda) is
better when compared to OpenMP.

The individual performance results for all the trials has been attached
for all the test cases in the form of text files.

The code has also been attached.

Note: In case of GPU codes, please use the main functions submitted
by me, as we need to define block sizes in the main function.