Lam Dang

Matrix-Matrix Multiplication Result Sheet

I start with charliep matrix-matrix multiplication serial solution, which gives me 78-80 seconds of execution. The reason is that the matrix have to produce a product in every position of the matrix, computing rows by rows to complete the result matrix. However, I discover that the calculations of every element in 1 row is completely independent for another row (as in the result of an element in one row would not change that of another row)

My solution: Distribute evenly chunks of rows for different threads so that each thread is responsible for the calculations of different rows. Since the calculation of one chunk of rows is the same work as the other, I use static scheduling, mapping total_number_row/number_ thread amount of rows to each thread.

My result:

- Strong Scaling: By increasing the number of thread while keeping the number of items in dimension fixed, I got a speedup of almost 0.5 (Reference the strong scale pdf generated by the automated scripts) => High chance this is the case of Strong Scalability
- Weak Scaling: By increasing the problem size proportionally with the number of thread, I get a near linear increase in execution time => not Weak Scaling (Reference the weak scale pdf generated by the automated scripts)