TDMA-2D solvers
Report

Guilherme Bertoldo

July 24, 2011

This report presents results of pallelization of a line-by-line TDMA solver for a linear system $Ax = b$, where A is a 5-diagonal matrix or a 9-diagonal matrix. Parallelization was implemented using OpenMP.

TDMA-2D 5-diagonals

Solver tdma2d5 was developed to solve a bidimentional problem with n_x volumes in the x-direction and n_y volumes in the y-direction totalizing $N = n_x n_y$ unknowns. This solver was tested for a fabricated linear system whose solution was known. Once the solver was verified, it was measured the time t_1 required by a single processor to solve a linear system of $n_x = n_y = 2^n$ unknowns ($N = 2^{2n}$), where $n = 2, 3, ..., 11$. Then the procedure was repeated using two processors instead of one, producing a different time t_2. Times t_1 and t_2 were measured three times (fig. 1) using an intrinsic timer of OpenMP (OMP_GET_WTIME()). One should expect that with two processors time t_2 would be a half of t_1, however fig. 1 shows that t_2 is about 70% of t_1 for $N = n_x n_y$ higher than 10^3. For smaller values of N, multiprocessing may be worse than single processing. This occurs because there is a time consumption spent to organize the threads that will share the work. For small values of N this organizing time is comparable to the working time. Another important observation is that time measurement may have a big fluctuation. This may be caused by the system use of one of the processors during the calculation. So this numerical experiment should be repeated in a computer with more than two cores.

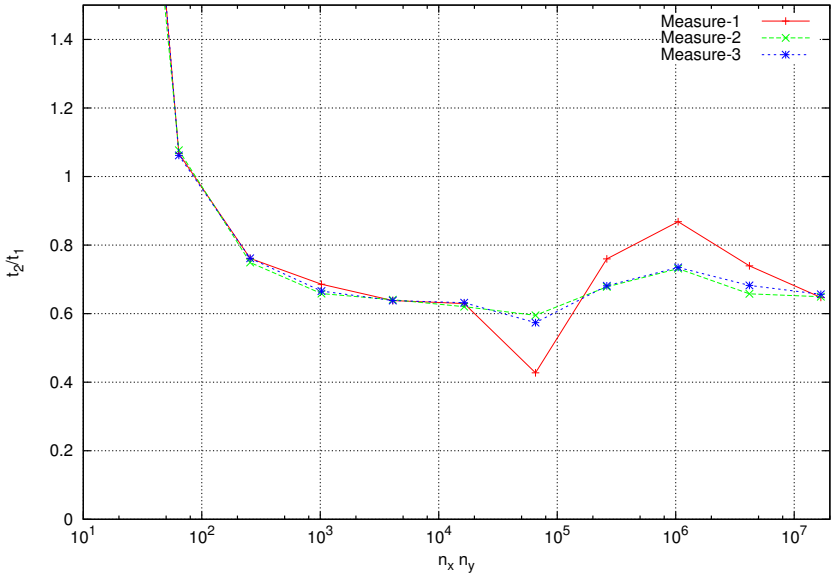

Figure 1: Time ratio *versus* linear system size for tdma2d5

TDMA-2D 9-diagonals

The same comments made to solver tdma2d5 are valid to tdma2d9, but now, A is a 9-diagonal matrix. The measurements are presented in fig. 2.

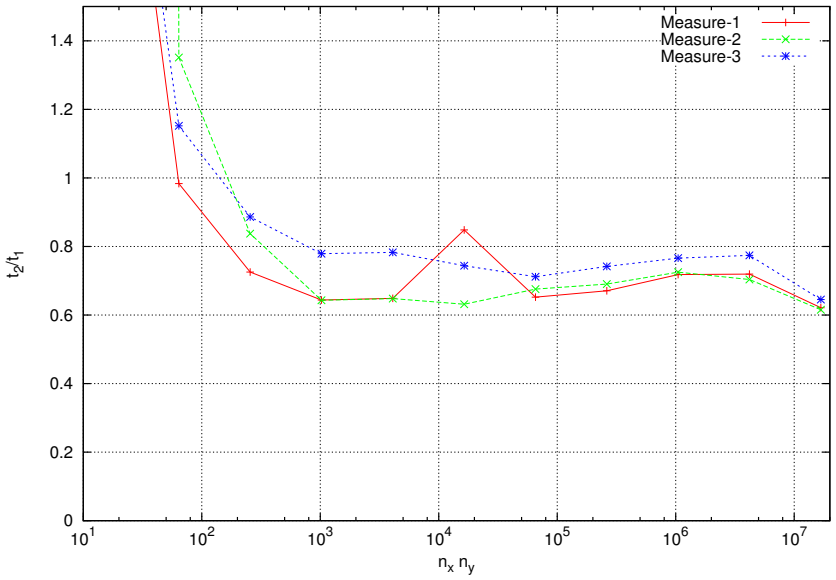

Figure 2: Time ratio *versus* linear system size for tdma2d9