Faculté des Sciences de Base
Master in mathematics
Master in computational science and engineering
Scientific computing applied to linear algebra
May 3, 2011

Solution of Serie 10

Exercise 1 *Since we have two vectors, the main loop for the Jacobi iterations and also the one for computing the residual do not have any dependency (one can check it by using the compiler option for the automatic parallelization). Hence both loops can be parallelized with the directive* `#pragma omp parallel for private(i,j)`*. The speedup is always quite good, but it can deteriorate for* $p > 8$*, depending on the run.*

Exercise 2 *The loop for computing the residual is the same as the one of Exercice 1. The parallelization of the main loop for the asynchronous Gauss-Seidel method requires the definition of an auxiliary private variable* `s` *for storing the component* `x1[i]` *on which we are working; this loop is parallelized with the directive* `#pragma omp parallel for private(i,j, s)`*. The code executed on only one processor realizes the standard Gauss-Seidel method, which converges in* 12 *iterations, while the Jacobi method converges in* 21 *iterations. For* $p > 1$*, the asynchronous method converge sometimes faster than the true Jacobi method, and the CPU time is always better. The remark about the speedup of Exercise 1 is also true here.*