

Tema 2 ASC - Implementarea operatiei dsymv din BLAS

Ciocan Mihai
334CA

Implementarea “ de mana “ (main.c)

Pentru implementarea de mana a operatiei dsymv din BLAS am utilizat programul scris in fortran drept referinta si am scris codul in C doar pentru cazul $INCX = 1$, $INCY = 1$ si $UPLO = u$ pentru simplitate. Timpul e calculat folosind functia clock() inainte si dupa apelarea functiei dsymv ($(end - start)/CLOCKS_PER_SEC$) si e afisat intr-un fisier alaturi de ordinul de marime al matricii ce va fi utilizat pentru afisarea graficului. Alpha si Beta au valori arbitrare iar X si Y sunt initializate cu formule in functie de pozitia in vector (ex: $X[i] = (i * 3.0) / 7.0$).

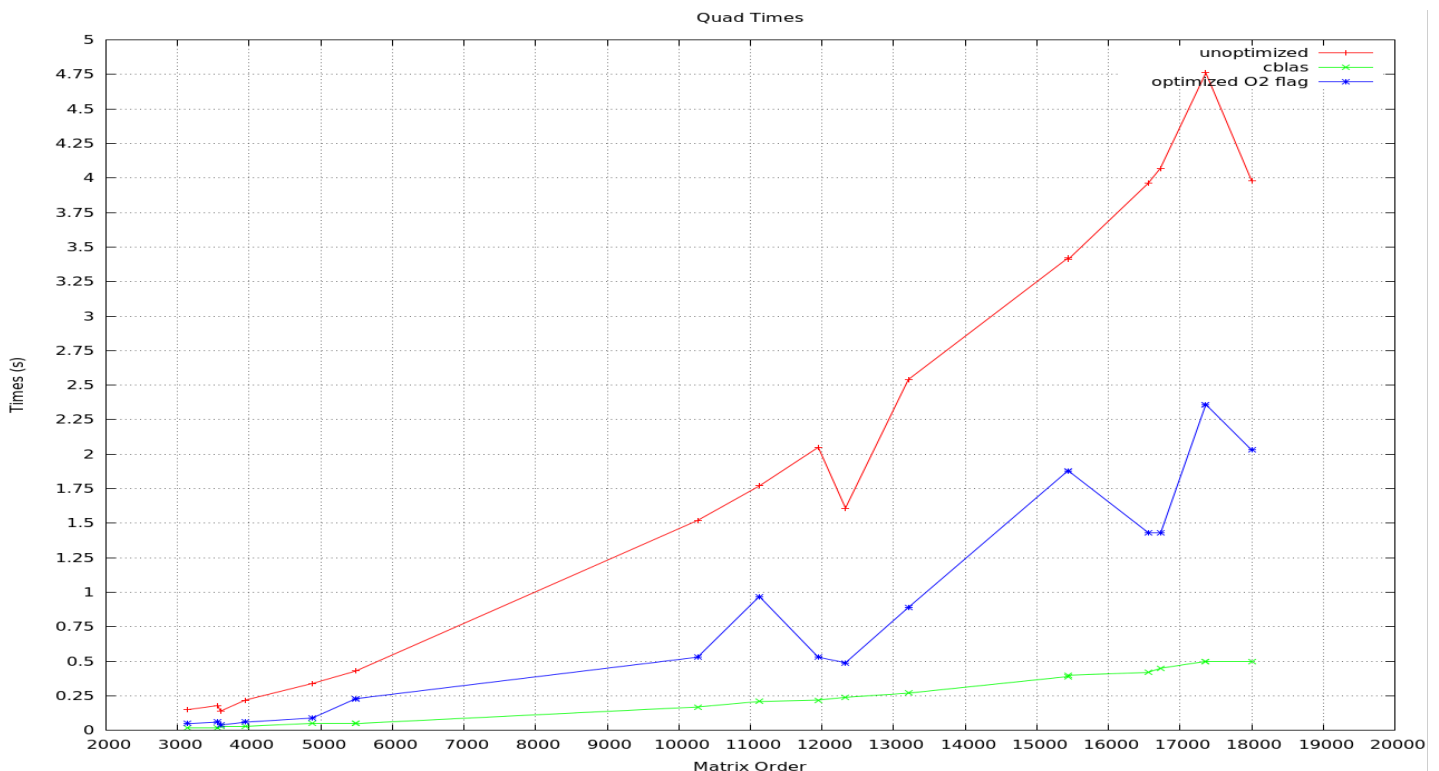
Utilizarea rutinei din BLAS (ATLAS) (blas.c)

Programul este similar cu cel descris mai sus diferenta fiind practic apelarea functiei dsymv din biblioteca cblas. Timpul e calculat la fel ca mai sus si e de asemenea scris intr-un fisier separat pentru afisarea graficului. Alpha, Beta, X si Y au acelesi valori ca in implementarea “de mana” pentru a putea confrunta rezultatele intre cele 2 implementari. La rulare am verificat daca rezultatele sunt identice (cu diff).

Compararea timpilor programului propriu cu implementarea BLAS

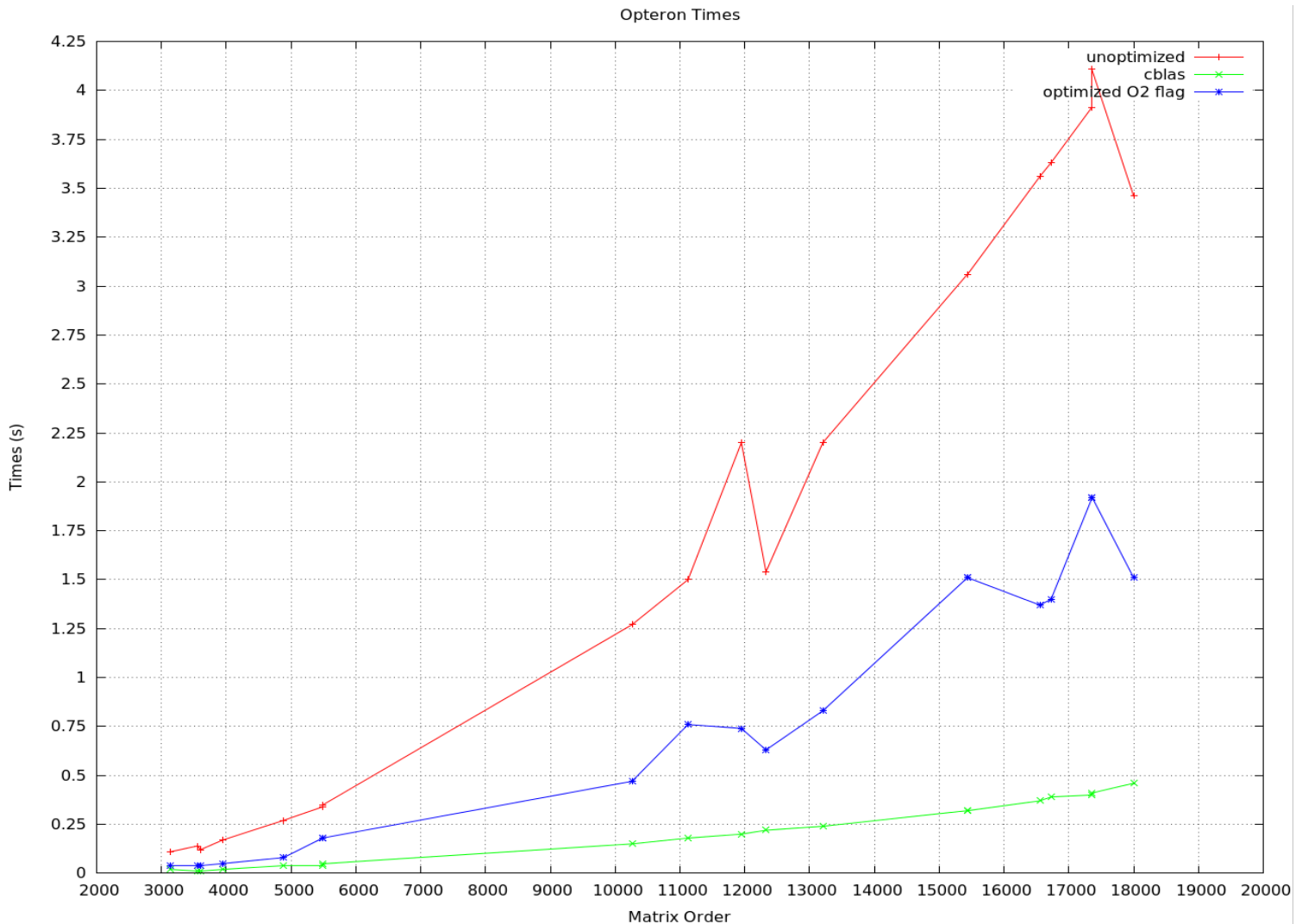
Pentru strangerea datelor am utilizat exemple de matrici de pe MatrixMarket cu ordinul de marime intre 3000 si 18000, simetrice si de tipul real.

Xeon Quad



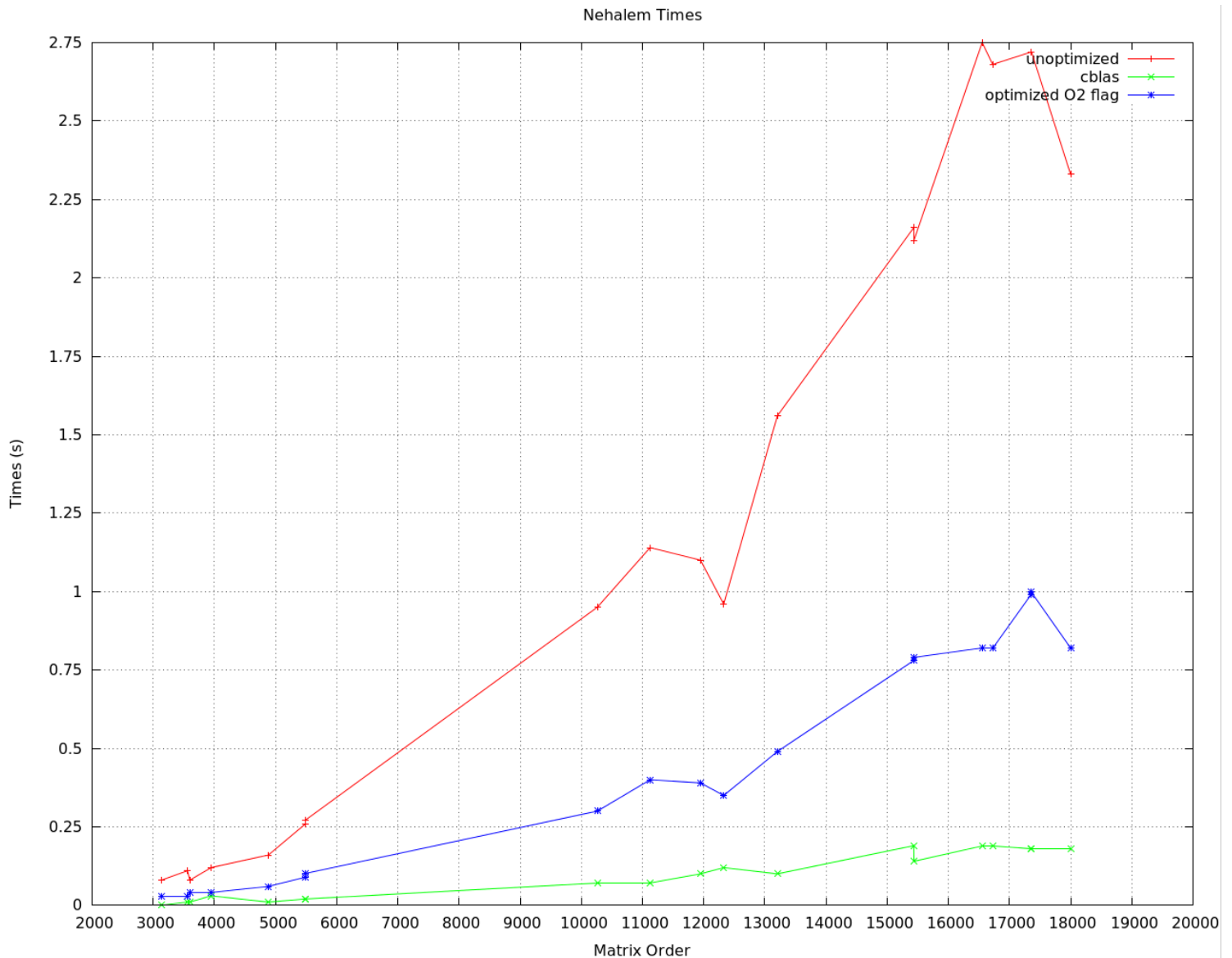
Pentru arhitectura Xeon Quad se poate observa ca pentru matrici de ordinul 17000 -18000 timpul de procesare al programului propriu este de aproape 5 secunde, iar acelasi program optimizat ruleaza in timp de 2 ori mai mic. Implementarea BLAS ruleaza de 10 ori mai repede - in doar 0.5 secunde. Se observa ca implementarea BLAS are o crestere liniara spre deosebire de celelalte 2 abordari.

AMD Opteron



Pentru AMD Opteron timpul pentru acelasi ordin de marime specificat mai sus este mai mic fata de arhitectura Quad pentru versiunea proprie neoptimizata. Varianta optimizata ruleaza cu timpi cu 0.5 secunde mai mici fata de cea pentru Quad. Implementarea BLAS are insa timpi comparabili cu cei de pe arhitectura Quad in jur de 0.5 sec.

Xeon Nehalem



Programele rulate pe arhitectura Nehalem performante de aproape 2 ori mai bune decat Quad, si 1.5 ori mai bune decat Opteron pentru matrici cu ordinul de 17000- 18000. Varianta optimizata ruleaza de 2 ori mai rapid decat aceleasi variante pentru celelalte arhitecturi, la fel si implementarea BLAS.

Se pare ca matricea de ordin ~12000 pe care am utilizat-o in strangerea de date prezinta o particularitate ce modifica liniaritatea graficului pe toate cele 3 arhitecturi pentru implementarea realizata de mine.