



DIAGONALIZACION ESCALABLE EN CUDA PARA MATRICES SIMETRICAS

Amilcar Meneses Viveros¹, Erika Hernández Rubio², Alberto Estrella Cruz¹, Juan Cipriano Hernández Cortes¹

¹Departamento de Computación; Cinvestav-IPN; México

²Instituto Politécnico Nacional; SEPI-ESCOM; México

e-mail: ameneses@cs.cinvestav.mx

La diagonalización de matrices es uno de los procedimientos computacionales de uso frecuente en química, física e ingeniería. Las matrices que se operan son simétricas. El procedimiento tradicional para diagonalizar estas matrices es aplicar una bifactorización de Householder para obtener una matriz tridiagonal y después aplicar un procedimiento de diagonalización por división o por iteración de factorización QR (Press et al., 1996). Sin embargo, estos procesos son computacionalmente costosos. En los últimos años se han popularizado el uso de GPUs para acelerar los cálculos (Maia et al, 2012; Du et al, 2012) y se han desarrollado varios programas para aprovechar las ventajas que ofrecen este tipo de aceleradores. Sin embargo, muchos de estos programas no escalan cuando se tiene servidores con varias tarjetas o clusters con nodos GPUs (Maia et al, 2012; Du et al, 2012; Haidar et al, 2015). En este trabajo se presenta la implementación paralela combinando MPI, OpenMP y CUDA de los algoritmos de bifactorización Householder y de Cuppen para matrices simétricas. El uso de MPI permite hacer el escalamiento a través de los nodos de un cluster, OpenMP permite el control de diversas tarjetas de GPUs por nodo y CUDA para mandar trabajos a las GPUs. Se realizan comparaciones con MLK y MAGMA y se muestra que la implementación escalable tiene mejor desempeño que el de estas bibliotecas numéricas.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. *Numerical recipes in C*. Cambridge: Cambridge university press. **1996**. Vol. 2.

Maia, J. D. C., Urquiza Carvalho, G. A., Manguiera Jr, C. P., Santana, S. R., Cabral, L. A. F., & Rocha, G. B. GPU linear algebra libraries and GPGPU programming for accelerating MOPAC semiempirical quantum chemistry calculations. *Journal of chemical theory and computation*. **2012**, Vol. 8; No. 9, 3072-3081.

Du, P., Weber, R., Luszczek, P., Tomov, S., Peterson, G., & Dongarra, J. From CUDA to OpenCL: Towards a performance-portable solution for multi-platform GPU programming. *Parallel Computing*; **2012**, vol 38, no 8, pp. 391-407.

Haidar, A., Tomov, S., Luszczek, P., & Dongarra, J. (2015). MAGMA embedded: Towards a dense linear algebra library for energy efficient extreme computing. In *High Performance Extreme Computing Conference (HPEC)*, **2015** IEEE (pp. 1-6). IEEE.