

FINDING NONOVERLAPPING SUBSTRUCTURES OF A SPARSE MATRIX*

ALI PINAR[†] AND VIRGINIA VASSILEVSKA[‡]

Dedicated to Alan George on the occasion of his 60th birthday

Abstract. Many applications of scientific computing rely on sparse matrix computations, thus efficient implementations of sparse matrix kernels are crucial for the overall efficiency of these applications. Due to the low compute-to-memory ratio and irregular memory access patterns, the performance of sparse matrix kernels is often far away from the peak performance on modern processors. Alternative matrix representations have been proposed, where the matrix A is split into A_d and A_s , so that A_d contains all dense blocks of a specified form in the matrix, and A_s contains the remaining entries. This facilitates using dense matrix kernels on the entries of A_d , producing better memory performance. We study the problem of finding a maximum number of nonoverlapping rectangular dense blocks in a sparse matrix. We show that the maximum nonoverlapping dense blocks problem is NP-complete by a reduction from the maximum independent set problem on cubic planar graphs. We also propose a $2/3$ -approximation algorithm for 2×2 blocks that runs in linear time in the number of nonzeros in the matrix. We discuss alternatives to rectangular blocks such as diagonal blocks and cross blocks and present complexity analysis and approximation algorithms.

Key words. Memory performance, memory-efficient data structures, high-performance computing, sparse matrices, independent sets, NP-completeness, approximation algorithms

AMS subject classifications. 65F50, 65Y20, 05C50, 05C69, 68Q17, 68W25

*Received August 10, 2004. Accepted for publication November 8, 2005. Recommended by S. Toledo. This work was supported by the Director, Office of Science, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy under contract DE-AC03-76SF00098.

[†]Corresponding author. Computational Research Division, Lawrence Berkeley National Laboratory (apinar@lbl.gov).

[‡]Computer Science Department, Carnegie Mellon University (virgi@cs.cmu.edu).