EECE5640 HW3

Quinn Arbolante

March 2025

Question 5

I looked at the minimal quadtree format[1] for this question. The minimal quadtree format stores a matrix via a quadtree, which is a tree whose nodes always have exactly four children. The matrix is partitioned into 2-by-2 blocks, and a node contains information about the four quadrants of its submatrix (for example, the root contains information about the quadrants of the entire matrix, and a child of the root contains information about the quadrants of the submatrix that is a quadrant of the entire matrix).

The quadtree structure is stored as an array of bits. Each bit is a node which says whether the subquadtree is empty or not. By doing so, the number of nodes is mimimized, as if a subquadtree is empty, its node has no children. Nonzero elements of the matrix are stored in a separate array, similar to CSR and CSC.

The minimal quadtree format has a smaller space complexity than CSR or CSC, which is why it shines for I/O operations. CSR, for example, requires $O(N * S(n) + n * S(N))$ space, where N is the number of nonzero elements, n is the order of the matrix, and $S(n)$ is the minimum number of bits required to store an array indexed from o to n, which is dependent on hardware and the operating system. This value for the space complexity comes from the two arrays aside from the array of elements that needs to be stored.

The minimal quadtree format, on the other hand, has a lower bound space complexity of $O(4 * (\frac{N}{3} + \log_4(\frac{n^2}{N})))$ and an upper bound space complexity of $O(4 * N(\frac{1}{3} + \log_4(\frac{n^2}{N}))$.

References

[1] I. Simecek, D. Langr, and P. Tvrdík. “Minimal Quadtree Format for Compression of Sparse Matrices Storage”. In: *2012 14th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*. 2012, pp. 359–364. DOI: 10.1109/SYNASC.2012.30.