## Parallelism of Sparse Matrix Operations

Written by Clayton Herbst (22245091)

Focus Questions:

* What operating system did you use in your submission?
* What sparse matrix operations did you use?
* For each matrix operation implemented
  + Description of the parallelism implemented
  + Some informal reasoning about expected run-time and scalability
  + Testing results
  + Comment on the performance observed

##### Background

Matrices are the foundational structure used to represent the data around us. The computer’s ability to naturally store and manipulate this information in contiguous memory locations provides exceptional computation benefits to otherwise complicated calculations. Matrix-like data structures closely represent the relationship between various datasets, often called ‘vectors’ or matrices. The terms ‘vector’ and matrix have slight difference however are interchangeable for the most part. Both of these structures help physicians & mathematicians represent the complex systems in our world through systems of linear equations and the operations on these linear equations. Matrix computation on computer systems are often used to give good approximations of complicated/computationally intensive calculations (MIT News). In this way matrices can be used to store and represent nearly all sets of data. Other examples of applications include complex data structures such as binary trees, which can use matrix representation effectively to represent the relationship between various nodes within a tree (MIT News). Graphics rely heavily on matrices to represent pixels within a file and their respective RGB colours (MIT News). Graphical effects are the result of manipulating these pixel values using operations such as scalar matrix multiplication and matrix vector multiplication to create distortion. An understanding of the many implementations of matrices allows one to gain a greater appreciation for their importance in the modern data driven world. Any improvements gained on the computational efficiency of matrix operations are thus highly valuable to society and the focus of this report.

Through the implementation of parallel computing using the OpenMP library I have endeavoured to make use of modern computer architecture to gain efficiencies in the calculations of common matrix operations. Parallel computing … . The practical processing times of the parallelised matrix operations will be compared to the practical computation times of a traditional synchronous, single threaded program. These results will be gathered using the macOS Mojave (v10.14.6) operating system, operating on a single Intel Core i5 processor with 2.3GHz processing speed and four physical cores. The hardware installed provides an L2 cache size of 256KB, L3 cache size of 6MB and 8GB of random-access memory (RAM). All results gathered and inferences made will be highly specific to the architecture used due to the nature of parallel computing and computing phenomena such as cache thrashing.

##### Sparse Matrix Data Structures

##### Parallel Sparse Matrix Scalar Multiplication

##### Parallel Sparse Matrix Trace Calculation

##### Parallel Sparse Matrix Addition

##### Parallel Sparse Matrix Transposition

##### Parallel Sparse Matrix Vector Multiplication

##### References:

Hardesty L., 2013. *Explained: Matrices,* MIT News. http://news.mit.edu/2013/explained-

matrices-1206. [17 Sep 2019].