INDIAN INSTITUTE OF TECHNOLOGY ROORKEE

EEN-300 INDUSTRY ORIENTED PROBLEM

PROPOSAL

Case Study of different algorithms for various Matrix operations, and their C++ implementation as a template library

Guided by:

Prof. Dheeraj K Khatod, Deptartment of Electrical Engineering

Prepared by:

Aayush A Chaturvedi 16115002

Adarsh Kumar 16115009

Aviroop Pal 16115033

Samarth Joshi 161151099

OVERVIEW

The evaluation of the product of two matrices is computationally expensive. The most commonly used algorithm to compute multiplcation is $\tilde{O}(n^3)$. If the multiplication of two matrices can be done in $\tilde{O}(n^a)$, then the least upper bound for a is called the exponent of matrix multiplication and is denoted by w. Though the asymptotically best method for computing product of two matrices is $\tilde{O}(n^{2.373})$, they are rarely used in software packages due to heavy implementation.

The importance of matrix multiplication lies on the fact that many algorithms is heavily dependent on it. For example, in graph theory, the number of paths of length k between two nodes in a graph can be calculated by exponentiating the adjacency matrix of the graph to the power k. The Transformation matrix is a very practical matrix in both robotics and image processing. It also finds application in machine learning and artificial intelligence.

Our aim will be to study about some algorithms of matrix multiplication (Strassen's algorithm, Coppersmith and Winograd's algorithm to list a few) and implement these algorithms in *C*++ which can be used as a template library. The library will also include algorithms for matrix inversion and finding roots of a matrix. We'll determine benchmarks for each algorithms implemented and find bounds where each algorithm will be most efficient.

This library will be helpful to those who are working in image processing, robotics, machine learning, artificial intelligence etc. Since matrix multiplication is at a foundation level of matrix arithmetic and thus linear algebra, many operations will speed up. So they'll be able to deal with larger matrices and their efficiently will improve.

GOALS

- Study the various algorithms for matrix multiplication and analyse their time complexity.
- Determine benchmarks for each method and determine in which situation, which method will be most efficient.
- Use the different methods of multiplication in computing the inverse and computing the integral and fractional powers of a matrix.
- Implement these methods in C++ to be used as a template library.

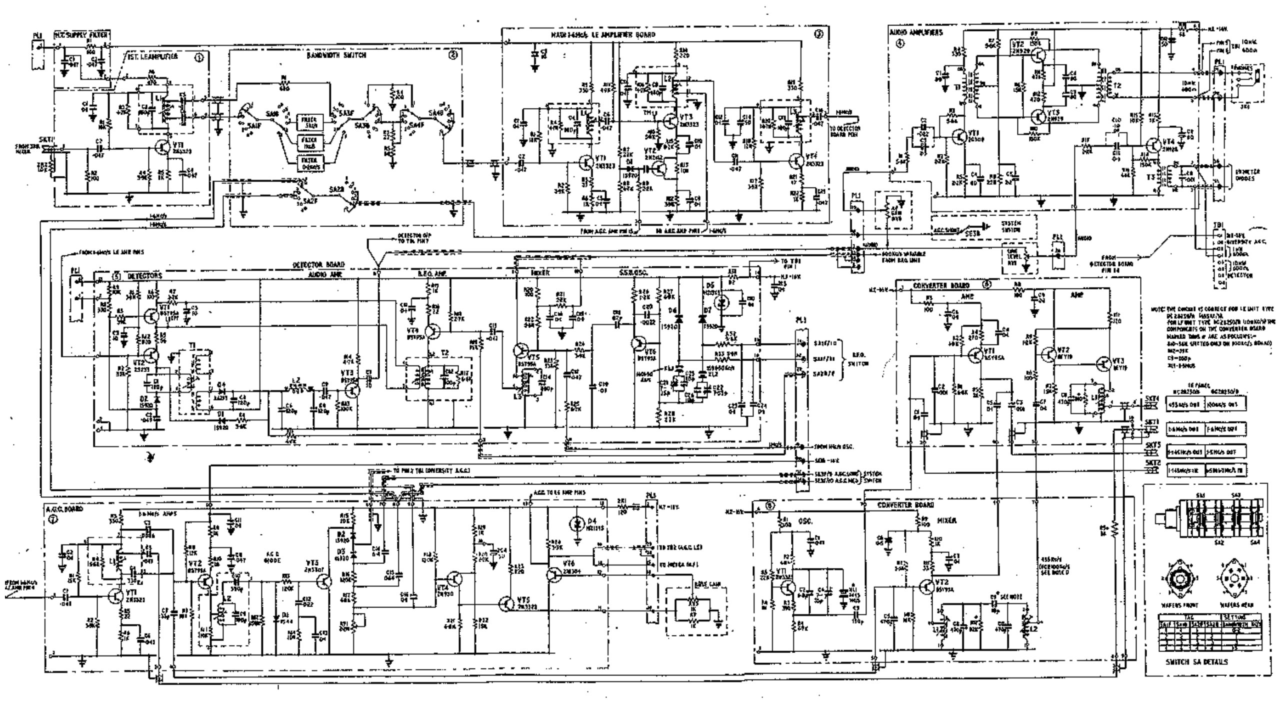
RELEVANCE IN THE INDUSTRY

Matrices are a prominant tools for storing relational data. They are very easy to create, use and operate. As a result they have found an extensive use in many computional sciences and statistics. However, matrices are used much more in daily life than people would have thought. In fact it is in front of us every day when going to work, at the university and even at home.

Matrix Multiplication is the kernel of many scientific applications. It is a binary operation that takes a pair of matrices, and produces another matrix. If A is an *n-by-m* matrix and B is an *m-by-p* matrix, the result AB of their multiplication is an *n-by-p* matrix defined only if the number of columns m of the left matrix A is the equal to the number of rows of the right matrix B. The procedure for finding an element of the resultant matrix is to multiply the first element of a given row from the first matrix times the first element of a given column from the second matrix, then add to that the product of the second element of the same row from the first matrix and the second element of the same column from the second matrix, then add the product of the third elements and so on, until the last element of that row from the first matrix is multiplied by the last element of that column from the second matrix and added to the sum of the other products.

Matrices and the Electrical Industry

Perhaps one of the most apparent uses of linear algebra is that which is used in Electrical Engineering. Matrices are used in study of electrical and magnetic circuits. The resistor, voltage source and capacitor take the stage as well as their accompanying language consisting of Kirchoff and Ohm. From solving a simple circuit to solving a complex circuit with multiple loops and branches, matrices play a pivotal role in shaping the solutions of the complicated problems. Besides circuits, Matrices are extensively used in control theory, in making state-space models, and designing and analysing linear as well as nonlinear control systems. Matrix mutiplication is the heart of the operations used here.



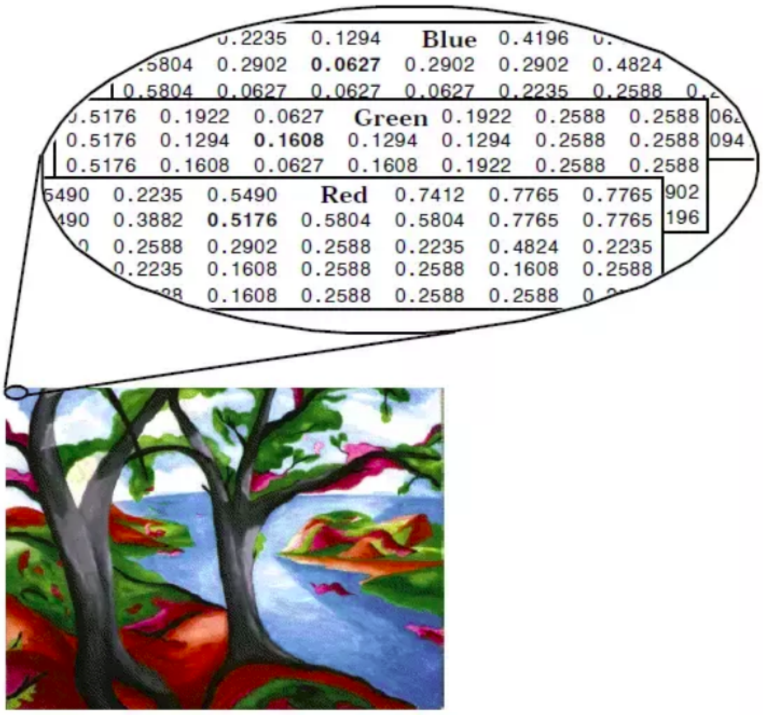
Circuit: I.F. Unit

The RA329 Receiver, Tube Radio Australia, IF Unit. Analysis of such complex circuits is not possible without matrices.

Matrices in the Graphic Industry

Graphic software such as Adobe Photoshop on your personal computer uses matrices to process linear transformations to render images. A square matrix can represent a linear transformation of a geometric object. In a video game, for example, a matrix would render the upside-down mirror image of an assassin reflected in a pond of blood.

If the video game has curved reflecting surfaces, such as a shiny metal shield, the matrix would be more complicated, to stretch or shrink the reflection.



An example image showing how matrices store the vital color codes' imformation for image storage and retreival.

Matrices in Cryptography and encoding

Matrices and inverse matrices are used for coding and encrypting messages. A message is made as a sequence of numbers in a binary format for communication and it follows code theory for solving. Encryption and decryption require the use of some secret information, usually referred to as a key. Depending on the encryption mechanism used, the same key might be used for both encryption and decryption, while for other mechanisms, the keys used for encryption and decryption might be different. Today governments use sophisticated methods of coding and decoding messages. One type of code, which is extremely difficult to break, makes use of a large matrix to encode a message. The receiver of the message decodes it using the inverse of the matrix. This first matrix is called the encoding matrix and its inverse is called the decoding matrix.

Matrices in Geology

In geology, matrices are used for making seismic surveys. They are used for plotting graphs, statistics and also to do scientific studies and research in almost different fields. Matrices are also used in representing the real world data's like the population of people, infant mortality rate, etc. They are best representation methods for plotting surveys. In economics very large matrices are used for optimization of problems, for example in making the best use of assets, whether labour or capital, in the manufacturing of a product and managing very large supply chains.

DEVELOPMENT OF DIFFERENT METHODS

Matrices have long been the subject of study. However, the rise of computers in the late 20th century has led to new problems. They are required to do many Matrix operations at a time. Hence it is desirable to implement algorithms, to reduce the number of steps required to perform computation.

Multiplication and the asymptotics

We consider that the number of operations required to multiply two n-*by-n* matrix is of the order $\tilde{O}(n^{\omega})$, we look at ways to in which ω can be reduced.

The most trivial technique of matrix multiplication was known until 1968. This is as follows:

If A and B are two n-by-n matrix and matrix $C = A*B$, then

$$C_{ij} = \sum_{k=1}^{n} A_{ik} * B_{jk}$$

The above algorithm does the computation at an upper limit of $\omega = 3$. As there must be a total of n^2 elements in the output, so at least n^2 operations are required to do the process, hence

$$2 \le \omega \le 3$$

Strassen showed that $\omega \le log_2 7$ can be achieved via a recursive procedure. This optimisation arises because it was found that two $2*2$ matrices can be multiplied in 7 multiplications as opposed to 8 using the earlier known technique. Thus, now we have

$$\omega = 2.8073$$

Coppersmith and Winograd introduced the idea of multiple disjoint matrix multiplication to find quicker algorithms capable of multiplying larger matrices. This method gives

$$\omega \le 2.49$$

In 2005, Cohn and Umans, used the group theoretic context to provide new conjecture. This conjecture which if proved can show that $\omega = 2$ is feasible.

Inversion

The matrix is considered to be invertible. The adjoint of a matrix can be used to find its inverse.

$$A^{-1} = \frac{adj(A)}{det(A)}$$

This is a very slow process as it involves repeated computations. One another method to find inverse is by Cayley - Hamilton theorem. The basis of this theorem is that every square matrix satisfies its own characteristics equation given by

$$|A - LI| = 0$$

This method however gives rise to an extra 'n' factor which makes its extremely slow for computational purposes.

A variant of gaussian elimination called Gauss - Jordan elimination can be used to find the inverse of matrix by augmenting it with an identity matrix and then converting the matrix to its row echelon form. An alternative is the LU decomposition which generates upper and lower triangular matrices which are easier to invert and hence are employed in high end softwares like MATLAB. These lower and upper triangular matrices after inversion are to be multiplied to get the result. Here, the upper bound of computation is determined by multiplication rather than inversion.

PROPOSED TIMELINE

22 February	- Complete the study of various algorithms, and report.
Early March	- Explore more possibilies in the existing algorithms, and find benchmarks.
Mid March	- Start the implementation in C++.
Mid April	- Complete the final library implementation in C++.

CONCLUSION

The study of various matrix operation algorithms is of prime importance in many fields of Engineering and Sciences. Also, the current implementation of the few fastest algorithms is generally not included in software packages due to their bulky nature and thus is not easily available.

We, through this project, aim to create a powerful implementation of the fastest algorithms in industry standards and make them available open source through GitHub.

REFERENCES

Khaled Thabet, Sumaia AL-Ghuribi, "*Matrix Multiplication Algorithms"* IJCSNS International Journal of Computer Science and Network Security, VOL.12 No.2, February 2012.

Kågström, B, Ling, P and Loan, C. (1995). "*GEMMBased Level 3 BLAS: High Performance Model Implementations and Performance Evaluation Benchmark*". Technical Report UMINF-95.18, Umeå University, Oct 1995.

Rönsch W. and Strauß, H. (1989). "*The Level 3BLAS Forms of Parallel Factorization Methods*". In D. Evans, G. Joubert, and F. Peters, editors, Parallel Computing 89, pages 85– 92. Elsevier Science Publisher B.V., 1989.

Stothers A. J. , "*On the Complexity of Matrix Multiplication*", Doctor of Philosophy, University of Edinburgh, 2010.