Bhavesh Rajesh Talreja (A20516822)
CS 553 ACM Lecture

The ACM A.M. Turing Award video lecture starts with the host Dr. Cherri Pancake giving a spectacular introduction and the achievements of **Dr. Jack Dongarra** and congratulating him for winning the **Turing award** which is considered as the **Nobel prize in computing**.

Dr. Dongarra speaks about his 40 years of experience with computing systems. The first task that he and his colleagues had was to do an implementation of Algol in Fortran. Algol had its basis in matrices as row-wise and Fortran reads matrices column-wise fashion. In 1973, he got his first publication published about loop unrolling in Fortran. In today's era, it is assumed that the compiler will take care of that. Going ahead, they used vector computers which had vectors registers in their architecture. Following which he and other people of that era witnessed the defacto standard of vector operations which led to LINPACK project. Until now, my understanding was exactly like LINPACK is a benchmarking tool, but, as he mentions that it is a collection of software for solving linear equations. **It was interesting to watch the first benchmark report done in 1977 and that too on 24 super machines of that era. This benchmark report list grew, and it still rolls out twice a year as Top500 project.**

In the 1990s, shared memory and cache-based systems were introduced. To maximize the use of these caches, L2 (Matrix-Vector operations) and L3 BLAS (Matrix-Matrix operations) were implemented to enhance the performance and thus LAPACK was introduced. **LAPACK implementation was designed for shared memory leveraged both EISPACK and LINPACK. An interesting thing is that every organization had their own way of implementation, and they felt a need of standard which led to the development of MPI. ScaLAPACK is an extension of LINPACK to cater to distributed memory systems.** He makes a very imminent point "A software design been instigated by the needs of the hardware changes taken place in recent times". Then came the era of multi-cores, and since the no. of available cores was more and accelerators were present to boost the performance, they developed PBLAS to enhance the floating-point performance of algorithms. They realized that there was a need for batched operations so that the operations done on different data can be implemented in parallel, this implementation was PaRSEC and Batched BLAS. He talks about recent times where we have a hybrid architecture, example: ORNL's summit computer where 97.7% performance is GPU based. Currently, they are working on implementing SLATE which will be used for computing on exascale processing.

**He made an interesting argument that in case of deciding two algorithms on a machine, we often choose the algorithm with less FLOPS, but here the key is communication and hence, we need to look at what kind of communication goes on with these algorithms. He talks about current supercomputers are at exaflops. He gives the analogy that even if every person on this earth does one operation, it would take 4 years to do that, and this computer can do it in one second. This is quite an interesting point about these incredible super computers.** I found one of the Q&A question relevant and relatable to projects I have worked on "Since, software and algorithms are being implemented in the software architectures talked about in the lecture, how provenance can be maintained and guarantee of reproduceable results", Dr. Dongarra replies that it is an important issue and there is a need of benchmark indicator mechanism and it is crucial that first we need a idea about how the solution looks like rather than blindly following the machine and moving on. Another strategy to guarantee reproduceable results is having a mechanism that describes the algorithms in detail so that if the algorithm is taken on a different machine, it should produce the same results.

**An interesting point is that in 1980s they thought that they would at most get to petaflops, and now there is already existence of couple of exaflops (billion billions transactions) and this is all due to GPUs in the architecture and his projection is that we will reach the first magnitude of zetaflops in the next 8 years.**