Computation of Allelic Frequencies Using the "*Cerberus*" Beowulf Cluster

Giuseppe Sena[1], Grant Fitzgerald[2], Richard Hayes[1], Justin Jiang[1], Julian Kuk[1], Patrick Ryan[1], Chester Moses[1], Lindsay Schulman[2], and Bruce Jackson[2]
Massachusetts Bay Community College
Wellesley Hills, MA, USA

Keywords—allele; Beowulf cluster; distributed systems; DNA; MPI; Short Tandem Repeats; STR

ABSTRACT

Short Tandem Repeats (STR) are specific repeats of a sequence of four or more nucleotides with no breaks in it (e.g. ATAGATAGATAGATAG where 'ATAG' is the repeating sequence). In criminal investigations involving biological evidence, the FBI recognizes 13 core loci (specific areas on chromosomes where STRs are most commonly found) where usually one or two alleles exist for every individual [1]. These 13 loci are organized in a table form, called an Allele Panel, showing the locus analyzed and the number of paternal and maternal STRs that are present in each. However, in some criminal and anthropological cases, more than two alleles may present themselves in any allele of a panel. This can be problematic, since one particular individual should only have one or two STRs per allele (one inherited from their father and one inherited from their mother). If 3 or more STRs are present, for example, minimally 3 or more subjects' DNA is being analyzed. This sort of mixture can occur by contamination from the scientist(s) testing the sample or by the presence of DNAs from multiple individuals.

Many times it is necessary to generate all possible allele combinations for specific genes. Forensic scientists can use that information to determine the probability that a specific DNA sample is from a particular individual. If the number of samples to be analyzed is very large, it is necessary to perform all computation (combinations) using a high-performance computer.

"***Cerberus***" (see Fig. 1) is an 8-node dual-core Linux Beowulf cluster [2] designed and implemented during the fall of 2012 at the MassBay Community College[1]. Funding for ***Cerberus*** came from a professional development grant [3] with the idea of creating labs and micro-labs to introduce parallelism and distributed systems concepts into the CS curricula.

In this work we used ***Cerberus*** to generate all possible allele combinations for a group of genes on large DNA samples. We developed a distributed application on a cluster using the **MPI**[3] library. We analyzed the performance and visualized the output using a simple GUI developed in **Python**. We have achieved picosecond allele analysis of mixed DNA samples using ***Cerberus***, and the application is now being used for the Forensic DNA Science Program at MassBay Community College. This technology now allows the reanalysis of questionable DNA findings in mixed sample cases and the platform for much improved STR analysis.

Fig. 1: "Cerberus" Beowulf Cluster

For one specific gene, if you have **n** alleles and **r** the number of choices (here 2 as each individual has 2 alleles), then the number of combinations is:

$$\binom{n}{r} = \frac{n!}{r!\,(n-r)!}$$

$$for\ r = 2, \qquad \binom{n}{2} = \frac{n!}{2!\,(n-2)!} = \frac{n(n-1)}{2}$$

For testing purposes, we generated random samples using a basic probability distribution to show the behavior of the system when the number of samples and number of processors are changed. Allele frequencies data for this project was obtained from the Promega site [4]. The application follows a Master-Slave paradigm, and it is written in **C** and **OpenMPI** [5]. The master node generates random samples and sends

[3] Message Passing Interface

each sample to each slave node. Each slave node computes all allele combinations for the loci in the sample, and sends the result back to the master node. This process continues until there are no more DNA samples to analyze. In addition, a **Python** program is also running on the master node showing the progress of the computation using a GUI. Communication between the GUI (**Python**) and the master node (**C**) uses **UDP Sockets**.

The Fig. 2 shows the parallel execution time on the cluster when processing 1,000,000 samples, and the number of MPI processes is increased. Note that the ***Cerberus*** cluster architecture is able to run a maximum 32 tasks in parallel. That is why if we increase the number of MPI processes more than 32, the performance deteriorate (execution time increases). Communication time take over computation time.

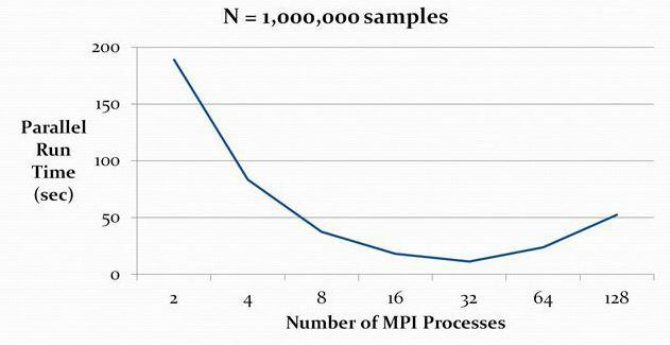

Fig. 2: Running time for N=1,000,000 samples

This work was a combined effort between the Computer Science and the Biotechnology departments at MassBay Community College. We are currently working on applications in the areas of networking, computer security, encryption, bioinformatics, DNA analysis, digital image processing, and computational science.

REFERENCES

[1] Federal Bureau of Investigation (FBI), "FAQs on the CODIS Program and the National DNA Index System," [Online]. Available: http://www.fbi.gov/about-us/lab/biometric-analysis/codis/codis-and-ndis-fact-sheet. [Accessed September 2013].

[2] R. W. Lucke, Building Clustered Linux Systems, Prentice Hall PTR, 2005.

[3] G. Sena, "Developing Labs and Microlabs for Adding Parallelism and Distributed Computing into Computer Science Curricula," Summer Professional Development Grant, unpublished, 2012.

[4] Promega, "Allele Frequencies," [Online]. Available: http://www.promega.com/products/pm/genetic-identity/population-statistics/allele-frequencies/. [Accessed 4 January 2014].

[5] R. L. Graham, G. M. Shipman, B. W. Barrett, R. H. Castain, G. Bosilca and A. Lumsdaine, "Open MPI: A High-Performance, Heterogeneous MPI".

[1] Computer Science Department
[2] Biotechnology Department