Error corrrection in high-throughput short-read data on GPU

Aman Mangal, Chirag Jain

October 7, 2014

1 Abstract

Personalised medicine and interpreting genomic data quickly is one of the most promising advances to come out of the intersection of HPC and life sciences. Error correction of short reads is an important component of the genome remapping software programs, given huge length of human DNA (of the order of billion). The latest sequencing technologies like Illumina are known to genereate more than million short reads of the genome with errors of about 0.5% to 2.5%. Fortunately, the renundancy in the data can be leveraged to bring this error down. The presence of big data and large compute requirement here gives us the scope of utilising our HPC expertise gained in the CSE 6230 class and improve the state of the art in this sphere. Our goal in the project will be to improve the runtime of GPU based solution to this problem.

2 Targets

Please see table 1.

3 Work Distribution

We each plan to read different set of papers and then discuss the ideas of the papers with the other person. We will divide the programming part equally as we proceed in the project.

4 Resource Requirement

We should be able to run our experiments on *jinx* nodes with GPU cards.

Table 1: Steps and timeline of the project

Date	**Target**
Oct 17	• Literature survey • Browse existing CUDA implementations upon which we can build further • Collect the publicly available datasets for experimentation
Oct 27	• First base implementation working • Run, profile and identify bottlenecks in the base implementation
Nov 07	• Reproduce the publication results • Have concrete plan on how to improve the timings plus the algorithm to implement which can potentially give us improvements • Also work on the details of algorithm proposed in the Kishore's report[2]
Nov 27	• Complete the new implementation • Profile it for further revision/optimisation of the expensive components
Dec 05	• Complete the optimisations planned • Finalise the implementation including code comments and README file • Evaluate the performance and compute the speedups gained. • Wrap up the project report.

5 Risk Assessment & Management

We will begin with setting up existing codebase [1] on jinx cluster. The code was last updated in 2009. Hence, there is a slight chance that things may not work out for us. In such a scenario, we plan to implement the algorithm ourself using exisiting code sgments as far as possible.

6 Acknowledgement

Our project idea seeded out of past CSE 6140 project of Nagakishore Jammula (PhD student in Prof Aluru's group). Contribution of this project report[2] are:

1. Defining the error correction problem and types of algorithmic solutions.
2. Proposing a potential bottleneck in the GPU implementation in [4]
3. Suggested a way to map exising distributed solution [3] on a GPU hardware.

References

[1] A fast parallel error correction tool for short reads. CUDA code available here `http://cuda-ec.sourceforge.net/`.

[2] Nagakishore Jammula. CSE 6140 (Fall 2013) project report: An implementation of the spectrum-based short read error correction framework on GPU. pdf available here `http://tinyurl.com/o93u3un`.

[3] Ankit Shah, Sriram P. Chockalingam, and Srinivas Aluru. A parallel algorithm for spectrum-based short read error correction. In *IPDPS*, pages 60–70. IEEE Computer Society, 2012.

[4] Haixiang Shi, Bertil Schmidt, Weiguo Liu, and Wolfgang Müller-Wittig. A parallel algorithm for error correction in high-throughput short-read data on cuda-enabled graphics hardware. *Journal of Computational Biology*, 17(4):603–615, 2010.