International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 9 (2015) pp. 21877-21884

http://www.ripublication.com

Cancer Classification on Expression Data Using Phylogenetic Methods With Parallel Reconstruction of Neighbor Joining Tress Using CUDA

A. Natarajan [1], Dr.T.Ravi[2]
[1]Assistant Professor, Department of Information Technlogy
Jayaraj Annapackiam CSI College of Engineering, Nazareth, Tamilnadu, India.
[2]Principal, Srinivasa Institute of Engineering and Technology, Chennai, India.
[1]ayyanarnatarajan@gmail.com

Abstract

Phylogenetics has played an important role in the field of data mining and Bioinformatics. Computational cancer phylogenetics is very important in cancer classification. From the sample tissues gene expression levels of healthy and cancerous cells are evaluated using microarray technology. A many researchers have examined the problems of cancer classification. Cancer classification is the problem of finding a number of classes of cancers using the data from a DNA chip. The current work has applied phylogenetic methods for classifying the cancer. In this cancer classification problem, we require a metric on a set of cancer as a role of their expression levels of genes, and require a tree construction, using tree fitting methods are used from the field of phylogenetics. We proposed a Parallel Network Joining method which was implemented on GPU for constructing a phylogenetic tree. This work is mainly focused on parallelizing the tree fitting to achieve good speedups for our sample input data using CUDA.

Keywords: Cancer classification, Microarray gene expression data, Phylogenetics, Parallel Neighbor joining, CUDA, Breast cancer Data Set.

Introduction

The Phylogenetics field [1] is one of the important areas of data mining and Medical bioinformatics which deals with computational methods to infer evolutionary heritage of organisms and genes. This new field is used for cancer classification on gene expression data. Cancer classification is major medical data mining problems in the field of microarray expression analysis of cancer datasets. The microarray gene expression data are collected with a classifier variable which gives every samples to one of a number of classes to diagnose the cancer. Cancer classification is the process

of identify classes based only on the gene expression data in the nonappearance of classifier variables.

A number of methods have been developed to solve the class discovery problem. The method of hierarchical clustering [2], which require a set of genes in hierarchical structure. The classification is done by a distance function on the objects. Another method for clustering is the k-means approach [3], which ask for an data around K centroids on optimal clustering .The other approach is self organizing maps (SOMs),explained by [4],[5]. Class discovery can be done with construction of classification trees whose leaves communicate to the objects being classified. Up to the classification of we constructed the classification tree to classify unknown samples. There are so many phylogenetic methods like distance matrix, maximum parsimony, maximum likelihood, Bayesian methods and probabilistic inference have possible applications in cancer research.

In this paper, we proposed Parallel Neighbor joining method programs in PHYLIP [6] under a multi-core GPU environment which was achieved by GTX 780 NVIDIA Graphics[7] card. Neighbor joining method implemented in parallel can achieve speedups up to 24 X. The result shows that GPU environments are powerful and cost-effective parallel environments, which can significantly improve the performance.

Breast Cancer Classification using Phylogenetic Tree with sequential implementation

Selection of Genes and Collection of Data:

In our assumption the gene expression data is as follows: for each of p tissue samples in the set S={s1,s2,…,sm},we are given the gene expression level used for every of q genes in the set Gn={gn1,gn2,gn3…,gnq}.This yields a p * q matrix A= ((a_{uv})), where a_{ij} is the expression level in the sample si of the gene gn_i. In our cases y_i be a discrete classifier variable for 1<u<p,assigning each sample s_i to one of a small number of sets or. Clusters traditionally, expression data studies have focused on the following class discovery problem

Linear Transformation For Normalizing The Input Matrix:

In our approach tree fitting metric is applied to the expression data. Since the input matrix A consist of gene expression levels of wide variety of genes and different genes make active at various expression levels, it is better not to comparing the actual values of a_{uv},hence we have to do tree fitting on the unprocessed data, the tree topology is determined by those variables that contain the maximum values. To overcome this problem, we have normalized the input matrix. This problem is avoided by normalizing the input matrix with linear transformation ultimately making each gene weighing equally.

Consider the gene $g_{u\,,}$ whose values are represented in the v th column of A. Let p_v denote the mean expression levels for $g_{u\,,}$ and let σ_v indicate the equivalent standard variation. Here we classify the matrix with normalized entry ((b_{uv})) by

$$F_{u,v} = \frac{a_u - \mu_v}{\sigma_v}$$

$$T_{distance}(R)=\sum_{u=1}^{n}\sum_{j=i+1}^{n}\left[(2*T_{Rw}(l_{u,}\, l_{v,})+T_{pq}(lp_{u,}\, lp_{v}))\right]+o(\ q^2) \quad (1)$$

$$T_{tree}(R)=\sum_{u=1}^{q-3}\sum_{u=1}^{q-u+1}\sum_{v=u+1}^{q-y+1} C_{tree}+o(q^2) \quad (2)$$

By Normalizing the matrix we had the advanatage of comparing the expression levels across columns that weighs each column equally. This has created meaningful information. It has also reduced the effects of outlier data. The input datas are trimmed by more than four standard deviations from the mean. Through this is the following constraint was impared $| b_{ij} | \leq 4$ for all ii and j. The above constraint was implemented to reduce the effect outliers on distance based tree reconstruction algorithms [9]. This process has played a major role in resampling process.

Classification Method:

In this graph A is a pair $G_1=(V_1,E_1)$ where V_1 is a finite set of objects and E_1 is a set of pair of objects from V. The sequence of cycle in graph $c=(v_0,e_1,v_1,e_2,\ldots e_k,v_k)$,where $v_i \neq v_j$ for $1\leq I \leq j \leq k$,but $v_0=v_k$ and $e_i=(v_i\text{-}1,v_i) \in E$ for all I in which case we say v_0 and v_k are connected by p.A graph is connected if for all pairs $x,y \in V$,there is a path p_{xy} in G connecting x and y .A connected graph containing no cycles is called a tree.

Let L be a set of objects and R the set of real numbers. without loss of generality L={1,2,..n}.A metric on L is a function $D:L\times L\rightarrow R$ satisfying three properties.

1. $D1(u,v)\geq 0$ for all $u,v \in L$ with $D1(u,v)=0$ if and only if u=v.
2. $D1(u,v)=D(v,u)$ for all $x,y \in L$.
3. For all $u,v,y \in L$,$D1(u,y)\leq D1(u,v)+D(u,y)$

We can express D1 as distance matrix D1,with entries ((d1uv)) where duv=D(u,v).

A tree metric is a specific kind of metric .Let be a tree ,T=(V,E) with L(V,L the set of leaves of T.Let l be a function $l:E\rightarrow R^+$.For any pairs of leaves $i,j \in L$,define p_{ij} to be the unique path in T from x to y.Let DT be defined by

$$D^T(i,j)=\sum l(e)\cdot$$

Suppose D1 is a metric on L. The tree fitting problem has given D, find tree T such that DT is a good approximation for D. Tree fitting is one of a variety of methods that taxonomists use to build phylogenetic trees to calculate evolutionary history. Leading tree fitting algorithms include the Neighbor Joining algorithm [10] and , the least-squares approach of [11].The advantages of using tree fitting include the ability to use well-refined, pre-existing software packages, whether commercial (PAUP) or in the public domain (PHYLIP),and a supporting body of literature which can guide us to which problems are computationally feasible and which are not. Traditionally, tree fitting has been used in a setting where there is some reason to suppose that the input metric D can be well approximated by a tree metric.

PHYLIP

This package is being used from 1980 onwards. There are thousands of registered users for this software. PHYLIP package can be used with microarray data along with Neighbor joining methods for searching phylogenetic tress. It has also included routines that allowed variety of resampling techniques that include neighbor joining tree fitting method. It has the capability to reconstruct consensus tree from the resample data analyses by using strict agreement or numerous deviation of majority rule agreement. In PHYLIP bootstrapping is a separate process which has resulted in using same bootstrapped dataset as input to different phylogenetic method. This has resulted in association of trees formed by different methods from a general bootstrapping framework.

Proposed Method:

Parallel Neighbor Joining Algorithm using CUDA:

The algorithm was parallelized using a master-slave model with a hybrid message passing model. Before going into an in-depth review of the solution developed, certain aspects of the Neighbor-Joining algorithm should be analyzed, since these might explain why an efficient parallel solution is difficult to obtain. First, it should be noted that for each iteration of the main loop, a new node is added to the distance matrix, but those from the two originating nodes are removed. This means that, in a parallel solution, the work carried out by each task in each iteration decreases as the iterations of the main loop progress. Also, in distributed environments, the distances of the new nodes must be distributed among all processes forming the solution. The search for the pair of nodes whose distance is minimal represents the most expensive part of the main loop from a computational standpoint. Taking into account that the distance matrix is triangular, distributing the work required for the search process by assigning the same number of rows to each task could result into idle time, since not all rows have the same number of cells. Since idle time negatively affects the performance of an algorithm, it must be removed or, if this is not possible, minimized. For this reason, the workload distribution strategy must be chosen trying to make it as equitable as possible for all tasks.

Pseudocode of the Parallel Neighbor Joining Algorithm.

Step0: Normalization of expression data
Step1: Mast Procs div dist_matrix into P procs and distribute p-1 with slaves At each procs→approx ((N)x(N-1)/2P) from dist_matrix.p
Step2: Each procs calculate nodes.
Step3: for H in 1 to N-2 do
  3.1 Each procs calculates→ local min Di,j
  3.2 The mast_procs→ all local min and calculates→ global minDi,j
  3.3 The mast_procs→new node(k)
Step4: node(k) calculate Dik and dij.
Step5: node(k) calculate distance from remaining nodes.

Step6: node(k)→ broadcasts to other procs.
Step7: End.

In the above code uses a master-slave model of *P* processes as parallelization strategy. Each process generates *T* threads when computation begins. Then, the iterations belonging to different process loops are distributed among the threads that have been generated. The distances of each newly created node are distributed among all processes following a circular order.

At the initialization stage, the distance matrix has loaded into GPU memory from host memory. After the each pair of node is selected, the values of data of the associated cells in the distance matrix has to be updated for the subsequent iteration. If the total matrix is completely reloaded, the time operating cost would be quite high due to the comparatively narrow memory bandwidth between the GPU and the host. In order to extensively data transferred amount is decreased, The suitable cells are updated for only changed cells.. The values of data in single column and single row cells in the distance matrix have to to be transferred from host to GPU in every iteration. This formulates the data transfer in negligible. The Shared memory is used for storing the temporary results of each block. Every thread in one thread block evaluates and chooses the node pair whose arrangement into a latest node provides the very small branch length between the node pair allocated to it. It stores the preferred node pair and its value $S1_{min}$ into the storage space which is allocated to it in the shared memory.

CUDA Programming Model

NVIDIA's CUDA [13] upgrading environment facilitates developers to use the marvelous computational capability and high in bandwidth of GPUs. The scalable array of multithreaded streaming multiprocessors are calculated by CUDA enabled GPU.The every multiprocessor is consisting of eight cores of processor. The CUDA program is running on the host CPU call a kernel grid which could run on a GPU with accessible execution ability, the thread blocks of the grid is distributed to the all processors. The threads block are set into distorts, which is executed parallel on one of the multi core processors. The thread blocks have contained up to 256 threads. The threads in a thread block can be executed in the same kernel and are scheduled independently.

Experiments and Results:

We have implemented the parallel Network joining algorithm using CUDA and evaluated this algorithm on an NVIDIA GeForce GTX 780 graphics card with 2880CUDA Cores. The topology of the phylogenetic tree is depending on the gene expression data. For the parallel neighbor joining method the runtime of the neighbor-joining tree fitting method mostly depends on the number of genes. Because of this actual genes have less impact on the runtime. All the values of data in the cells are positive distance matrix are positive. For simplicity this was produced from a random number generator. During these process the concept of majority was not lost. In this

implementations, each thread block has 256 threads and every thread processes are having a 16 different cells in the equivalent cell block.

We considered the data set from [12] , consisting of 1500 genes measured from 22 breast cancer cancers. The cancers are divided into three groups 1. Six cancers among BRCA1 mutations, eight cancers from seven patients cancers with BRCA2 mutations, and seven sporadic cancers, one of which contained a hyper ethylated BRCA1 promoter region. Considered only two binary classification questions, namely, whether each cancer carried a BRCA1 or BRCA2 mutation, respectively. Our attention focused on 200 genes, selected by[12], using a method based on an F-test to be those genes whose variation in expression best differentiated among the three types of cancers.

1. All the seven of the BRCA1 cancers are clustered together in one subtree.
2. All the eight of the BRCA2 cancers are clustered together in another subtree.
3. The seven sporadic cancers lay between the BRCA1 and BRCA2 subtrees, mostly as leaves off a main path between the two main clusters.
4. The sporadic cancer with a methylated BRCA1 promoter region was the sporadic cancer closest to the BRCA1 cluster.

The trees resulting from the correlation metric were of particular interest, as the edges connecting the sporadic cancers to the tree were much longer, on average, than the edges pendant to the BRCA1 or BRCA2 cancers. In fact, the six longest edges in the tree are pendant edges connecting six of the seven sporadic cancers to the backbone of the tree.

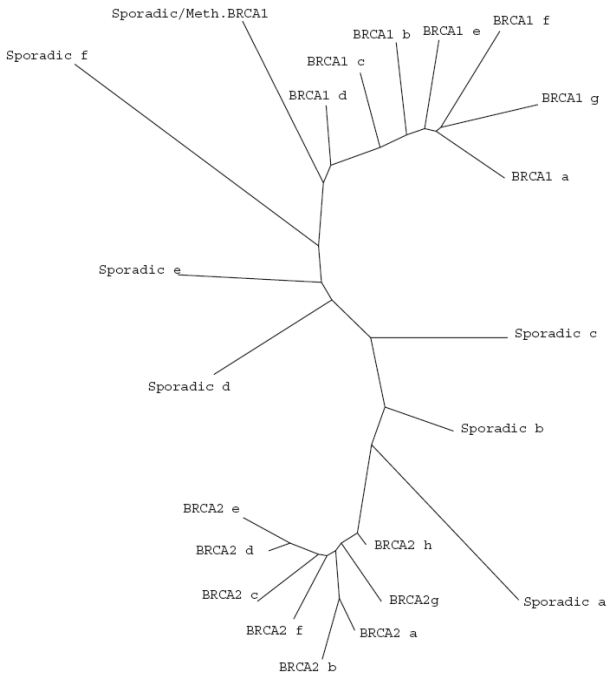

Figure 2: Breast Cancer Cancers Classified Using Neighbor Joining Method

Table 1: Parallel Neighbor joining using MPI

MPI Program	**Genes**	**Classification of Breast cancer Dataset**	**Elapsed time in hours**	
			Single Processor	GeForce GTX 780 GPU
MPI Prot dist	1500	BRCA1, BRCA2, BCAR3,BCAR2, BRMS1.	72	30

Conclusion and Future Work

To conclude tree fitting method is best for classifying cancer based on expression data. In this work we have demonstrated parallel neighbor joining method in multi core environment and achieved speed up of up to 24X when comparing with sequential implementations .The classification accuracy is also maintained when comparing with hierarchical clustering, k-means and self organizing maps. In this work gene feature selection were not consisidered. A combination of gene feature selection and parallel implementation of neighbor joining tree fitting algorithm may give still better accuracy in lesser time.

References

[1] Sanderson MJ, Driskell AC (2003) The challenge of constructing large phylogenetic trees.Trends Plant Sci 8: 373–379.

[2] Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D., 1998. Cluster analysis and display of genome-wide expression patterns. Proc.Natl Acad. Sci. USA 95, 14863–14868.

[3] Herwig, R., Poustka, A.J., M. uller, C., Bull, C., Lehrach, H., O'Brien,J., 1999. Large-scale clustering of cDNA-fingerprinting data.Genome Res. 9, 1093–1105.

[4] Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S.,Dmitrovsky, E., Lander, E.S., Golub, T.R., 1999. Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. Proc. Natl. Acad. Sci.USA 96, 2907–2912.

[5] Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M.,Mesirov, J.P, Coller, H., Loh, M.L., Downing, J.R., Caligiuri,M.A., Bloomfield, C.D., Lander, E.S., 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286, 531–537.

[6] Alexander J. Ropelewski, Hugh B. Nicholas Jr., Ricardo R. Gonzalez Mendez,2010. MPI-PHYLIP: Parallelizing Computationally Intensive Phylogenetic Analysis Routines for the Analysis of Large Protein Families. PLoS ONE, November 2010 | Volume 5 | Issue 11 | e13999

[7] Nvidia Corporation. NVIDIA CUDA Compute Unified Device Architecture Programming Guide version 2.0, 2008
[8] Felsenstein J (1989) PHYLIP - Phylogeny Inference Package (Version 3.2). Cladistics 5: 164–166.
[9] Huson, D.H., Smith, K.A., Warnow, T.J., 1999. Estimating large distances in phylogenetic reconstruction. In: Vitter, J.S., Zaroliagis,C.D. (Eds.), Algorithm Engineering, Proceedings of the Third International Workshop, WAE '99. Lecture Notes in Computer Science, 1668. Springer, Berlin, pp. 271–285.
[10] Saitou, N., Nei, M., 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Mol. Biol. Evol. 4, 406–424.
[11] Fitch, W.M.,Margoliash, E., 1967. Construction of phylogenetic trees.Science 155, 279–284.
[12] Hedenfalk, I., Duggan, D., Chen, Y., Radmacher, M., Bittner, M.,Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Kallioniemi, O.-P., Wilfond, B., Borg, A(., Trent, J., 2001. Gene-expression profiles in hereditary breast cancer. N. Engl. J. Med. 344 (8), 539–548.
[13] Muyan-Ozcelik P., Owens J.D., Xia J., Samant S.S.: Fast Deformable Registration on the GPU: A CUDA Implementation of Demons. International Conference on Computational Sciences and Its Applications. pp. 223-233, 2008
[14] Luebke D.: CUDA: Scalable parallel programming for high-performance scientific computing. ISBI 2008