NSF Graduate Research Fellowship Previous Research

Bill March

1 Research Overview and Interests

I have participated in research during my last two years as an undergraduate. I have studied fast algorithms for several geometry and graph problems and explored applications of these algorithms. I participated in the NSF Research Experience for Undergraduates at Georgia Tech in Summer 2005, where I explored fast algorithms for online graph coloring. Since the Summer of 2006, I have been working with Alex Gray on fast algorithms for machine learning and scientific applications. I have also been investigating applications of these algorithms in science, particularly computational biology and chemistry.

2 Euclidean Minimum Spanning Trees and Hierarchical Clustering

Context. Euclidean Minimum Spanning Trees are fundamental structures in computational geometry and are widely applied in network design, optimization, and computer vision. Additionally, the EMST is equivalent to a form of hierarchical clustering of the underlying points. This method of clustering is commonly applied throughout science and is particularly popular for clustering gene microarray data and cosmological surveys.

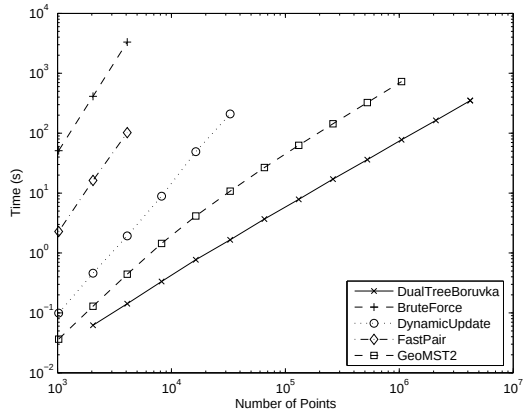
Since it is such a fundamental and widely applied problem, finding the EMST is well-studied. Many of these are quite effective in limited cases, such as the planar case. However, none of the existing algorithms are truly scalable to massive data sets and high dimensions. New algorithms are necessary to handle large-scale data gathering efforts, such as the Sloan Digital Sky Survey and the Human Genome Project.

Alex suggested it might be possible to apply some of his techniques to the EMST problem. I surveyed the literature for existing algorithms. I worked with some of Alex's other graduate students to better understand the N-body framework. I developed the algorithm on my own, with occasional discussions with Alex and other students. I implemented the algorithm and designed the experiments. I was also the primary author of the conference paper describing the new algorithm.

Details. I applied an existing framework, known as *generalized N-body problems* [1], to finding EMST's. This framework has been previously used to create fast algorithms for many problems in statistics and machine learning including the N-point correlation, all-nearest-neighbors, and kernel density estimation. Using these ideas, I was able to design and implement the fastest existing algorithm for EMST's in any dimension.

The algorithm, DualTreeBoruvka, applies multi-tree recursion to Borůvka's algorithm for finding MST's. Borůvka's algorithm is similar to Kruskal's, in that it maintains a spanning forest and iteratively connects components to build the tree. While Kruskal's algorithm connects the closest two components in each step, Borůvka's connects each component with its nearest neighbor. I used the space-partitioning trees and multi-tree recursion ideas to quickly compute the nearest neighbor of each component.

The algorithm uses a *kd-tree* to organize the points. Each node of the tree consists of a hyper-rectangle or bounding box containing a subset of the data. The root node contains the entire data set. Children are created as follows: choose the longest dimension of the current node's bounding box, partition the data along the midpoint in this dimension, and form two



(a) Runtimes on synthetic clustered data.

dim	N	GeoMST2	DTB	Speedup
3	389354	78.0	16.9	4.6
12	320000	702	62.3	11.3

(b) Runtimes for SDSS data and protein folding trajectories.

new, smaller bounding boxes to cover the subsets. The tree forms leaves when a node contains fewer than some specified number of points.

Dual-tree Recursion. The simplest way to compute the nearest neighbor of a component Q is to compute the distances from all points $q \in Q$ to all points $r \in \overline{Q}$ and keep the smallest. For each q, we store the distance to the nearest neighbor found so far, $d^u(q)$, as an upper bound on the true distance. If, for any point $r, d(q,r) > d^u(q)$, then we can be confident r is not the nearest neighbor of q.

Using the *kd*-tree, we can improve on this method. Instead of iterating over the points $r \in \overline{Q}$, we compare a point q with an entire node R. Using the bounding box of R, we can compute a lower bound $d^l(q,R) < d(q,r)$ for all $r \in R$. Then, if $d^l(q,R) > d^u(q)$ for some R, we can *prune* any further consideration of points in R. Otherwise, we recursively consider the two children of R.

This idea can be extended even further. Since we want the nearest neighbor of each component Q, we should exploit the fact that these points are also grouped in the *kd*-tree. We can maintain a bound $d^u(Q)$ on the upper bound for a nearest neighbor of all points in a node Q. We compare nodes Q and R, compute the minimum distance between their bounding boxes $d^l(Q,R)$, and prune if $d^l(Q,R) > d^u(Q)$. Otherwise, we recursively compare the children of Q with the children of R. By accounting for points in the same component, this method can quickly and efficiently solve the problem of finding each component's nearest neighbor.

Results This method proved to be the fastest known algorithm for finding EMST's in general metric spaces. I compared it to several well known MST algorithms, including GeoMST2, the previous fastest algorithm. Since hierarchical clustering is important in many domains, including cosmology and bioinformatics, I also tested my algorithm on two large data sets from these areas. One is three-dimensional spectral data taken from the SDSS, and the other is a compressed (12-dimensional) representation of 16,000 steps of folding simulations for 20 proteins.

What I learned. I gained experience with designing faster algorithms for a long standing and well-studied problem. I had to master a diverse literature covering the topic from many different fields. I gained an appreciation for the potential of truly scalable algorithms in the

sciences.

3 Publications

EMST paper (submitted).

References

[1] A. Gray and A. W. Moore. N-body problems in statistical learning. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13 (December 2000)*. MIT Press, 2001.