CHAPTER VII

CONCLUSIONS AND FUTURE WORK

Conclusions

This work is a synthesis of several new results. In Chapter II we presented a new heuristic for the SMT problem as well as a new measure of accuracy for heuristics. The heuristic was based upon Simulated Annealing and was an outgrowth of the 1-Steiner problem. The measure of accuracy expresses how much of the distance from the MST to the SMT a heuristic covers on a particular problem. This measure showed that our heuristic has achieved a score of over 95% on every randomly generated 100 point problem we tested, while using less than $\frac{1}{10}$th the cpu time currently used to solve the problem exactly.

In Chapter III we presented a new parallel algorithm for calculating the Steiner Hull and thereby decomposing the problem. This algorithm while not having a great theoretical speedup, in practice, computes the Steiner Hull for randomly generated 100 point problems rapidly.

The foundation of this thesis, presented in Chapters IV and VI, is a parallel algorithm for the generation of T_list and its implementation. When Winter developed his algorithm, the time for extraction dominated the overall computation time. After Cockayne and Hewgill's work, the time to generate T_list dominated the overall computation time. The parallel algorithms we presented in Chapters III and IV were implemented in a program called PARSTEINER94, and the results presented in Chapter IV show that the time to generate T_list has now been cut so much that the extraction time once again dominates the overall computation time.

In Chapter IV we also presented a characterization of SMT's for certain sizes of grids. The characterization of the SMT for a 2 x m grid has been studied in depth, and is well defined. The only other characterization for grids previously known were for square grids. In this work we present the characterizations of SMT's for 2 x m grids, as well as conjectured characterizations of SMT's for 3 x m, and 4 x m grids.

Further Parallelization

There remains a great deal of work that can be done on the Steiner Minimal Tree problem in the parallel arena. The first thing to consider is whether there are other ways to approach the parallel generation of T_list that would be more efficient. Improvement in this area would push the computation pendulum even further away from T_list generation and towards SMT extraction.

The next thing to consider is the entire extraction process. The initial generation of the *incompatibility matrix* has the appearance of easy parallelization. The forest management technique introduced by Cockayne and Hewgill could also be put into a parallel framework, thereby speeding up the preparation for extraction quite a bit.

With this initialization out of the way, decomposition could then be considered. The best possible enhancement here might be the addition of thresholds. As with most parallel algorithms, for any problem smaller than a particular size it is usually faster to solve it sequentially. These thresholds could come into play in determining whether to call a further decomposition, such as the cycle decomposition introduced by Cockayne and Hewgill that was discussed in Chapter V.

The final option for parallelization is one that may yield the best results, and that is in the extraction itself. Extraction is basically a branch and bound process, using the *incompatibility matrix*. This branch and bound is primed with the length of the MST as the initial bound, and continues until all possible combinations have been considered. The easiest implementation here would probably be the idea presented in the paper by Quinn and Deo [37] that served as the basis for the parallel algorithm in Chapter IV.

Additional Problems

1-Reliable Steiner Tree Problem

If we would like to be able to sustain a single failure of any vertex, without interrupting communication among remaining vertices, the minimum length network problem takes on a decidedly different structure. For example, in any FST all of the original vertices are of degree 1, and hence any one can be disconnected from the network by a single failure of the adjacent Steiner Point.

We would clearly like a minimum length 2-connected network. The answer can be the minimum length Hamiltonian cycle (consider the vertices of the unit square), but it need not be, as shown in the Θ graph given in Figure 38.

Here we can add Steiner points near the vertices of degree 3, and reduce the network length without sacrificing 2-connectivity. This is not just a single graph, but is a member of a family of graphs that look like ladders, where the Θ graph has only one internal rung. We hope to extend earlier work providing constructions on 2-connected graphs [41] to allow effective application of an Annealing Algorithm that could walk through graphs within the 2-connected class.

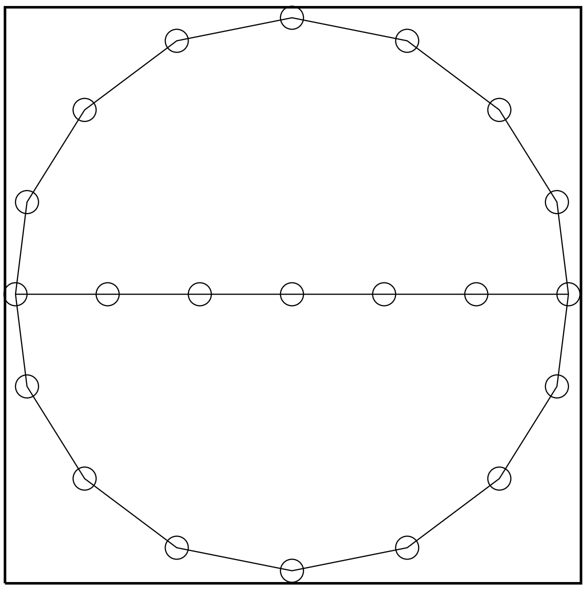

Figure 38. Theta graph.

Addition to Existing Networks

Cockayne has presented the following problem: Given a connected graph G and a set of vertices, V, which are not on edges of G, find a connected graph H which has G as a subgraph and contains the vertices in V.

This problem is very interesting when you consider it in the context of a growing road network. In this case you have a small town with an existing road network and a few new houses and businesses that have been built outside of the town. Your job is to construct a set of roads with minimum length that connects these new houses and businesses to the town's road network.

The Steiner Minimal Tree problem we have discussed here can be seen as a special case of this new problem. For the SMT, the existing graph is empty, and all new vertices must be connected with minimal length. Therefore it is quite simple to deduce that this is also an NP-Complete problem.