Properties and algorithms for Minimum Steiner Tree problem

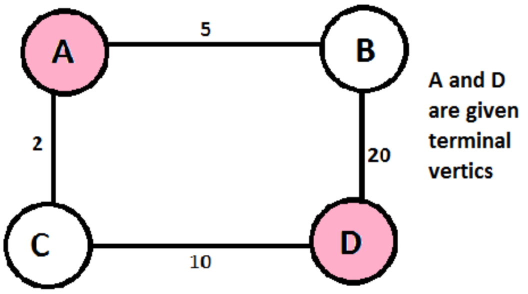
By:-
Aman Garg
U101116FCS281
S3
2016-2020

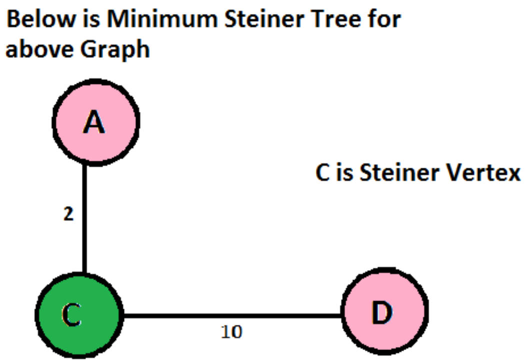
What is Steiner Tree?

Given a graph and a **subset** of vertices in the graph, a steiner tree spans though the given subset. The Steiner Tree may contain some vertices which are not in given subset but are used to connect the vertices of subset.

The given set of vertices is called ***Terminal Vertices*** and other vertices that are used to construct Steiner tree are called ***Steiner vertices***.

The Steiner Tree Problem is to find the minimum cost Steiner Tree. See below for an example.





The Steiner Minimal Tree (SMT) Problem: Given a set P of n points, determine a set S of

Steiner points such that the minimum spanning tree (MST) cost over P ∪ S is minimized.

An optimal solution to this problem is referred to as a Steiner minimal tree (or simply "Steiner tree") over P , denoted SMT (P). An edge in a tree T has cost equal to the distance between its endpoints, and the cost of T itself is the sum of its edge costs, denoted cost(T). The wiring cost between a pair of pins (x1, y1) and (x2, y2) in a VLSI layout is typically modeled by the Manhattan, or rectilinear distance1:

$\mathrm{dist}((x1, y1), (x2, y2)) = (\Delta x) + (\Delta y) = |x1 - x2| + |y1 - y2|$

We will focus on the rectilinear Steiner minimal tree problem, where every edge is embedded in the plane using a path of one or more alternating horizontal and vertical segments between its endpoints. Figure 1 depicts an MST and an SMT for the same pointset in the Manhattan plane. The bounding box of a pointset P denotes the smallest rectangle2 which contains all points of P and whose sides are oriented parallel to the coordinate axes. If an edge between two points is embedded with minimum possible wirelength, its routing segments will remain within the bounding box induced by its endpoints.

The Iterated 1-Steiner (I1S) Approach

This section outlines the Iterated 1-Steiner heuristic [55, 57], which repeatedly finds optimum single Steiner points for inclusion into the pointset. Given two pointsets A and B, we define the MST savings of B with respect to A as:

ΔMST (A, B) = cost(MST (A)) − cost(MST (A ∪ B)).

Let H(P) denote the Steiner candidate set, i.e., the intersection points of all horizontal and vertical lines passing through points of P (as defined by Hanan's theorem [38] - see Figure 2).

For any pointset P , a 1-Steiner point with respect to P is a point x ∈ H(P) that maximizes ΔMST (P, {x}) > 0. Starting with a pointset P and a set S = ∅ of Steiner points, the Iterated

1-Steiner (I1S) method repeatedly finds a 1-Steiner point x for P ∪ S and sets S ← S ∪ {x}. The cost of MST (P ∪ S) will decrease with each added point, and the construction terminates when there no longer exists any point x with ΔMST (P ∪ S, {x}) > 0.

An optimal Steiner tree over n points has at most n−2 Steiner points of degree at least 3 (this follows from simple degree arguments [35]). However, the I1S method can (or rare occasions) add

more than n − 2 Steiner points. Therefore, at each iteration we eliminate any extraneous Steiner points which have degree ≤ 2 in the MST over P ∪ S (since such points can not contribute to the tree cost savings). Figure 3 formally describes the algorithm, and Figure 4 illustrates a sample

execution.

Iterated 1-Steiner (I1S) Heuristic [36, 55, 57]
Input: set P of n points **Output:** rectilinear Steiner tree spanning P

$S = \emptyset$
While Candidate_Set $= \{x \in H(P \cup S) \mid \Delta MST(P \cup S, \{x\}) > 0\} \neq \emptyset$ **Do**
    Find $x \in$ Candidate_Set which maximizes $\Delta MST(P \cup S, \{x\})$
    $S = S \cup \{x\}$
    Remove points in S which have degree ≤ 2 in $MST(P \cup S)$
Output MST$(P \cup S)$

To find a 1-Steiner point in the Manhattan plane, it suffices to construct an MST over |P ∪S|+1 points for each of the O(n2) members of the Steiner candidate set (i.e., Hanan grid points), and then pick a candidate which minimizes the overall MST cost. Each MST computation can be performed in O(n log n) time [72], yielding an O(n3 log n) time method to find a single 1-Steiner

point. A more efficient algorithm based on [33] can find a new 1-Steiner point within O(n2) time

[57]. A linear number of Steiner points can therefore be found in O(n3) time, and trees with a bounded number of k Steiner points require O(kn2) time. Since the MSTs between trying one

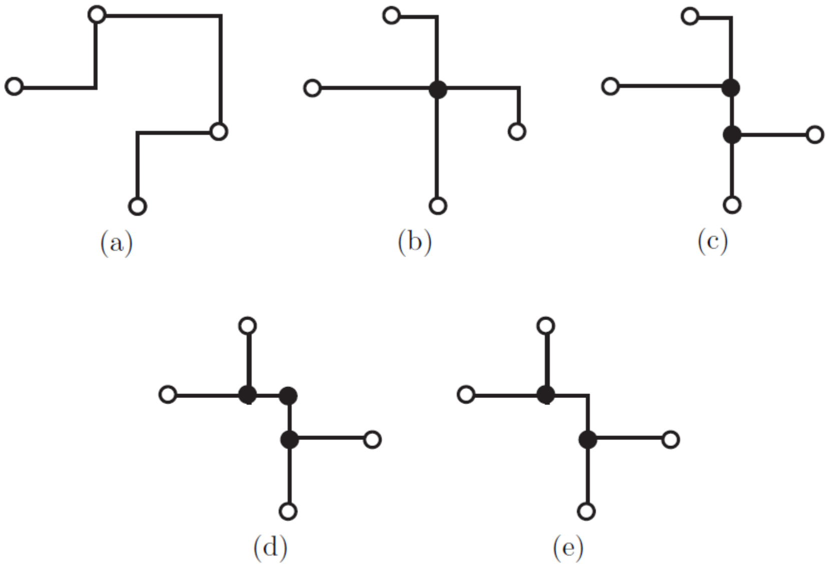


Figure 4: Execution of Iterated 1-Steiner (I1S) on a 4-pin net. Note that in step (d) a superfluous degree-2 Steiner point forms, and is then eliminated from the topology in step (e).

candidate Steiner point and the next change very little (by only a constant number of tree edges), incremental/dynamic MST updating schemes can be employed, resulting in further asymptotic time complexity improvements [36, 57].

In practice, the number of iterations performed by I1S averages less than n 2 for uniformly distributed random pointsets [57]. Furthermore, the I1S heuristic is provably optimal for 4 or less points [57]; this is not a trivial observation, since many earlier heuristics were not optimal even for 4 points. On the other hand, the worst-case performance ratio of I1S over small pointsets is at least 7 6 and 13 11 for 5 and 9 points, respectively [55, 57], and is at least 1.3 in general [8]. The next subsection discusses a batched variant of the I1S approach, which offers runtime improvements in practice.

The Batched 1-Steiner Variant

Although a single 1-Steiner point may be found in O(n2) time, the required computational geometry techniques are complicated and not easy to implement. To address these issues, a batched variant of I1S was developed [55, 57], which amortizes the computational expense of finding 1-Steiner points by adding as many “independent” 1-Steiner points as possible in every round. The Batched 1-Steiner (B1S) variant computes _MST(P, {x}) for each candidate Steiner point x ∈ H(P) (i.e., the Hanan grid candidate points). Two candidate Steiner points x and y are independent if:

_MST(P, {x}) + _MST(P, {y}) ≤ _MST(P, {x, y}),

introducing each of the two 1-Steiner points does not reduce the potential gain in MST cost relative of the other 1-Steiner point. Given pointset P and a set of Steiner points S, each round of B1S greedily adds into S a maximal set of independent 1-Steiner points. Termination occurs when a round fails to add any new Steiner points (Figure 5). The total time required for each round is O(n2 log n).

Batched 1-Steiner (B1S) Heuristic [55, 57]
Input: set P of n points **Output:** rectilinear Steiner tree spanning P

While $T = \{x \in H(P) \mid \Delta MST(P, \{x\}) > 0\} \neq \emptyset$ **Do**
$S = \emptyset$
For $x \in \{T$ in order of non-increasing $\Delta MST\}$ **Do**
If $\Delta MST(P \cup S, \{x\}) \geq \Delta MST(P, \{x\})$ **Then** $S = S \cup \{x\}$
$P = P \cup S$
Remove from P Steiner points with degree ≤ 2 in MST(P)
Output MST(P)

In three dimensions, I1S exploits a generalization of Hanan's theorem to higher dimensions [88], namely that there always exists an optimal Steiner tree whose Steiner points are selected from the O(n3) intersections of all axis-orthogonal planes passing through points of P. The three dimensional analog of Hwang's result suggests that the Steiner ratio, i.e. the maximum cost(MST) cost(SMT) ratio for three dimensions is at most 5 3 ; however, this is only a conjecture and generalizing Hwang's theorem to dimensions three and higher is still an open problem. An example consisting of six points located in the middle of the faces of a rectilinear cube establishes that 5 3 is a lower bound for the Steiner ratio in three dimensions.

The I1S and B1S algorithms are highly parallelizable since each processor can independently compute the MST savings of different candidate Steiner points. The Iterated Steiner approach is therefore very amenable to parallel implementation on grid computers [36, 57]. As with I1S, the time complexity and practical runtime of B1S can be further improved using incremental / dynamic MST update techniques [16]. Moreover, by exploiting tighter bounds on the maximum MST degree in the rectilinear metric4, further runtime improvements can be obtained [36, 57, 78].

Empirical Performance of Iterated 1-Steiner

In benchmark tests, I1S and B1S compare very favorably with optimal Steiner tree algorithms, such as those of Salowe and Warme [82, 92] on random uniformly distributed pointsets (i.e., the standard testbed for Steiner tree heuristics [48]). Both I1S and B1S exhibit very similar average performance in terms of solution quality, approaching 11% average improvement over MST cost, which is on average less than half a percent from optimal. Moreover, I1S and B1S produce optimal solutions on 90% of all random 8-point instances (and on more than half of all random 15-point instances). For n = 30 points, I1S and B1S are on average only about 0.3% away from optimal, and yield optimal solutions in about one quarter of the cases [36, 57]. I1S and B1S also perform similarly well in three dimensions and in other Lk norms [36, 57]. Empirical experiments also indicate that the number of rounds required by B1S grows very slowly (i.e., apparently logarithmically) with the number of points [36, 57]. For example, on sets of 300 points the average number of B1S rounds is only 2.5, and was never observed to be more than 5 on any instance. As expected, over 95% of the total tree cost improvement occurs in the first B1S round, and over 99% of the total improvement occurs in the first two rounds [36, 57].

The average number of Steiner points generated by B1S grows linearly with the number of points (and is typically less than half the number of input points) [36, 57]. An example of the output of B1S on a random set of 300 points is shown in Figure 6.

Experimental data also indicates that only a small fraction of the Hanan candidates yield positive MST savings in a given B1S round, and that such positive-gain candidates are more likely to produce positive MST savings in subsequent rounds [55, 57]. Therefore, rather than examine the MST savings of all Hanan candidates in a given round, subsequent rounds may consider only the candidates that produced positive savings in the previous round. In practice, this strategy significantly contributes to reduction in the time spent during each round, without affecting the solution quality.

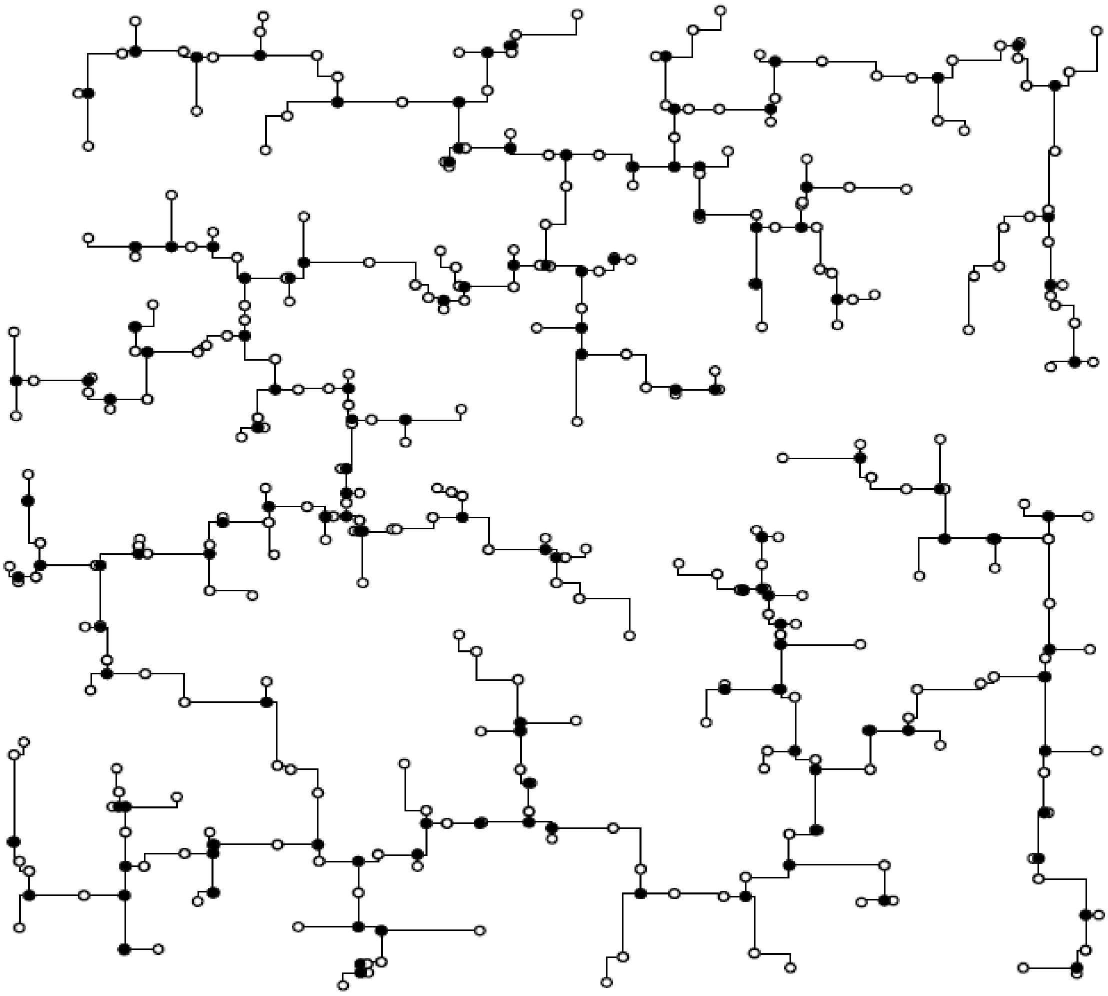


Figure 6: An example of the output of B1S on a random set of 300 points (hollow dots). The Steiner points produced by B1S are denoted by solid dots.

Steiner ratio

The Steiner ratio is the supremum of the ratio of the total length of the minimum spanning tree to the minimum Steiner tree for a set of points in the Euclidean plane. In the Euclidean Steiner tree problem, the Steiner ratio is conjectured to be 2/root(3) . Despite earlier claims of a proof, the conjecture is still open. In the rectilinear Steiner tree problem, the Steiner ratio is 3/2.

Source of Information:

1. GeeksforGeeks (GoG)
2. Wikipedia
3. Google