# Heuristics for Minimum Euclidean Skeletons

Yossi Frenkel, Supervised by Dr Charl Ras, The University of Melbourne

## Introduction

Imagine you're a robot navigating your way through an obstacle field, a captain steering around a coast, or a lithographer packing transistors into a microchip. You can model obstacles as polygons in the plane and construct a '*skeleton*' to help avoid collisions.

The scope of this project is to find an *approximation* algorithm to the Minimum Euclidean Skeleton problem. This project lies at the intersection of Graph theory, Euclidean geometry, and combinatorial optimisation. We embed a graph (representing a polygon) into a Euclidean metric space, where we care about the *geometric* relationships between the vertices and edges.

## Polygonal Euclidean Skeletons

### Polygons and Edges

- Let $\Omega$ be a *simple* polygon in the plane (no holes or self-intersections)
- Let $V$ be the set of $n$ vertices of $\Omega$, where no three vertices of $V$ are collinear; a *convex* vertex of $\Omega$ has interior angle less than $\pi$ rad
- An *edge* of a skeleton is a *Euclidean* line segment; a *maximal length* edge has its end-nodes on the boundary of $\Omega$
- A set of edges $E$ is *connected* if there exists a polygonal chain (piecewise linear curve) between any two edges in $E$ that lies entirely along edges in $E$ (i.e. you can trace a path between any two points in $E$ that uses only edges in $E$)

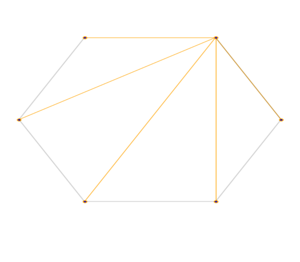

Fig. 1: A skeleton ($n = 6$)

Fig. 2: A skeleton ($n = 10$)

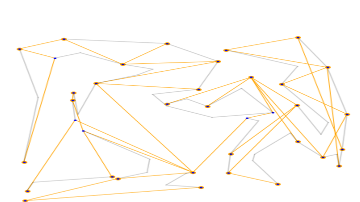

Fig. 3: A skeleton ($n = 60$)

### Skeletons

- A *skeleton* $S$ of $\Omega$ is a set of edges such that the line segment $P_1 - P_2$ between any two points $P_1$, $P_2$ that lie in the *exterior* to $\Omega$ intersects $\Omega$ iff $P_1 - P_2$ intersects (at least one) edge in $S$
- A *minimum* skeleton is the skeleton $S'$ of $\Omega$ with smallest cardinality

## An Equivalent Problem

In [1], it is shown, with sufficiency and necessity, that any set $S$ of maximal length edges that:

1. Intersects every **convex vertex** of $\Omega$
2. Is **connected**

is a skeleton of $\Omega$.

There exists a (computationally expensive) **exact algorithm** [1] to generate a minimum Euclidean skeleton, and this project seeks an **approximation** algorithm to minimise the skeleton size.

**Heuristic:** The problem of approximating a minimum skeleton of $\Omega$ reduces to finding some way to *cover* all *convex* vertices with as small a set of *connected* edges as possible. To do this we can employ a combination of the *Steiner tree on graphs* and the notion of a *visibility graph* of a polygon...

## Visibility Graphs and Steiner Trees in Graphs

**Visibility Graph** The *visibility graph* of a polygon $\Omega$ is the graph $G$ where each node in $G$ represents a vertex of $\Omega$, and nodes $u, v$ are adjacent iff the line segment $u - v$ lies entirely within the interior or on the boundary of $\Omega$ (i.e. there is a 'line-of-sight' between vertex $u$ and vertex $v$).

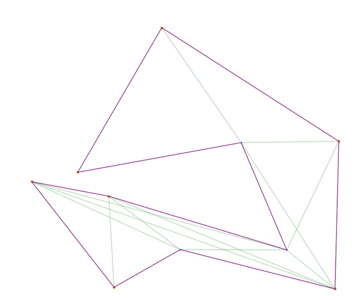

Fig. 4: Visibility graph ($n = 10$)

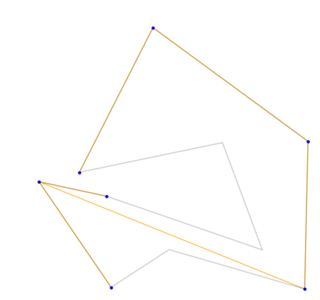

Fig. 5: (Approximate) Steiner tree ($n = 10$)

**Steiner Trees in Graphs:** Let $G = (V, E)$ be an undirected graph with non-negative edge weights $c$ and let $S \subseteq V$ be a subset of vertices, called *terminals*. The *Steiner tree problem* asks for the minimum-weight tree of $G$ that covers all terminals.

**Applying the Steiner Tree**:

1. Let the set of terminals be $S = \mathrm{Conv}(\Omega)$, the set of *convex* vertices of the polygon
2. Let the edge-weights $c = 1$ be constant
3. Let $G$ be the *visibility graph* of the polygon
4. Then the minimum-weight Steiner tree of $G$ will approximate the minimum skeleton of $\Omega$!

However, the Steiner tree problem is *NP-Hard*! So to utilise the Steiner tree, we need to approximate the Steiner tree problem itself! [2]

## Metric Closures and Minimum Spanning Trees

**Metric Closure:** The *metric closure* of a graph $G$ is the complete graph in which each edge is weighted by the *shortest path distance* between the nodes in $G$.

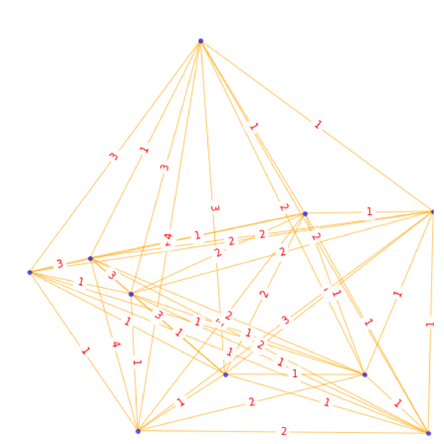

Fig. 6: Metric closure of above visibility graph

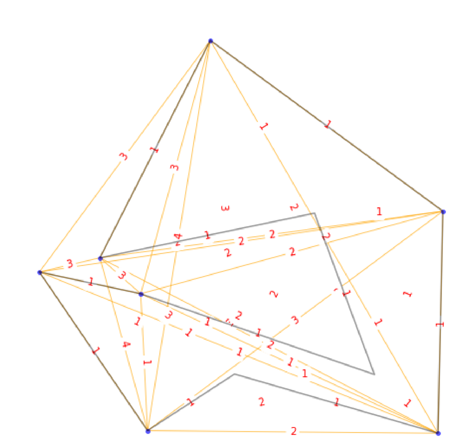

Fig. 7: Subgraph of metric closure induced by convex vertices (terminals)

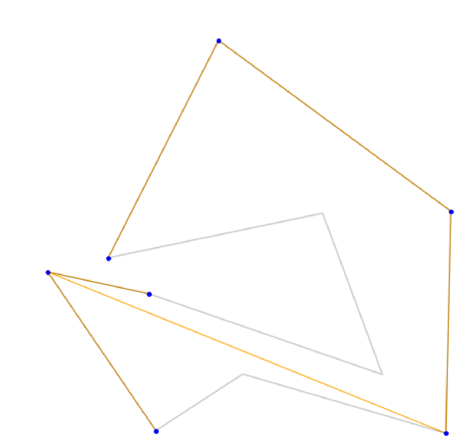

Fig. 8: MST of induced subgraph of metric closure

**Minimum Spanning Tree:** A *minimum spanning tree* (MST) is a subset of the edges of a connected, edge-weighted undirected graph that connects all the vertices together, without any cycles and with the *minimum possible total edge weight*.

**Complete Heuristic:** Compute the MST of the subgraph (induced by the convex vertices of $\Omega$) of the metric closure of the visibility graph of $\Omega$

## Results

**Bounds on Minimum Euclidean skeletons:** The number of vertices of $\Omega$ is an upper bound for $|S'|$, since the boundary of $\Omega$ is a feasible skeleton. Additionally, $|\mathrm{Conv}(\Omega)|/2$ is a lower bound for $|S'|$, since we need to intersect every convex vertex. **This heuristic** seemed to produce a skeleton with $\approx 0.6n$ edges, compared to the optimal solution using the exact algorithm [1] of $\approx 0.4n$ edges.

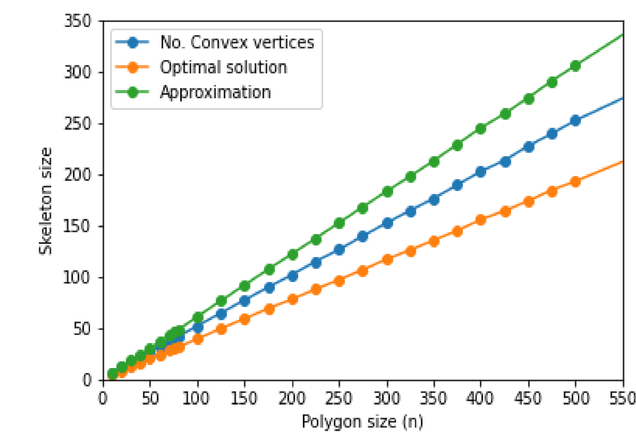


Fig. 9: Skeleton size

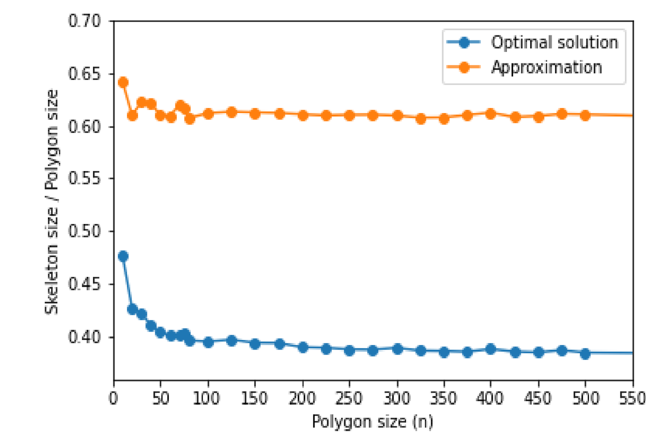


Fig. 10: Ratio of skeleton size to polygon size

## Next Steps

The (naive) visibility graph computation employed was the most computationally expensive part of the algorithm; but there exist alternate faster methods that could be implemented. The Steiner tree approximation was very fast ($\approx$ 2mins on an $n = 4000$ instance), and returns a solution with weight within a $2 - \frac{2}{|\mathrm{Conv}(\Omega)|}$ factor of the optimal Steiner tree [2]. In fact, from my data, it appears that the cardinality of the skeleton produced by the heuristic is within the same factor of the exact solution.

It would be interesting to explore alternate heuristics for the Minimum Euclidean Skeleton problem, and to adapt the existing heuristic, perhaps by using a non-constant cost function to weight the edges, or by generating a vertex-edge or edge-edge visibility or intersection graph to reduce the number of edges required.

## Acknowledgements

I'd like to thank Dr Charl Ras for his supervision and guidance, and Nicolau Andrés-Thió for contributing his time and assistance to this project. Thank you also to The School of Mathematics & Statistics for providing me with a wonderful mathematics research opportunity over the summer!

## References

[1] N. Andrés-Thió, M. Brazil, C. Ras, D. Thomas, and M. Volz. "An Exact Algorithm for Constructing Minimum Euclidean Skeletons of Polygons" ().

[2] L. Kou, G. Markowsky, and L. Berman. "A fast algorithm for Steiner trees". *Acta Informatica* (1981), pp. 141–145.

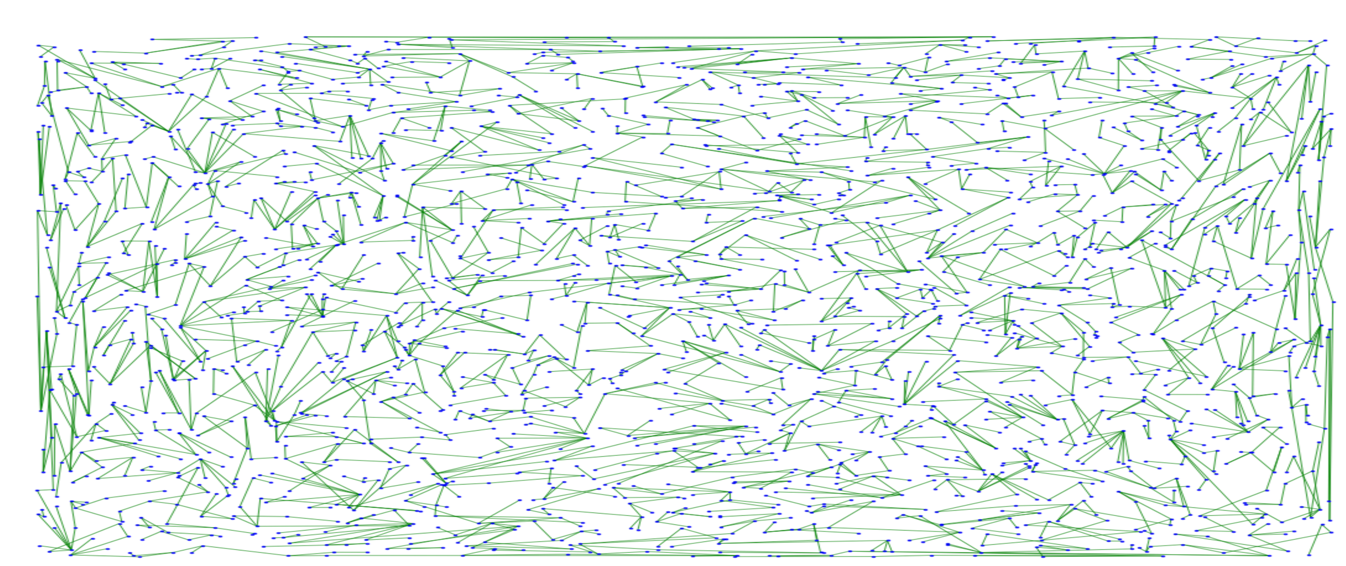

Fig. 11: An $n = 4000$ skeleton solution