The Euclidean Steiner Cable Trench Problem

Dylan Gaspar and Eric Landquist

Kutztown University

dgasp528@live.kutztown.edu, elandqui@kutztown.edu

August 18, 2016

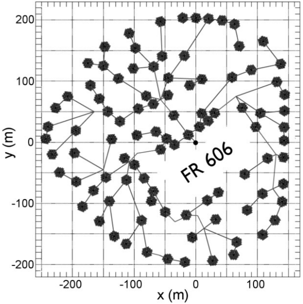
Outline

- Motivation
- Background
- The Euclidean Steiner Cable-Trench Problem
- Solution Approach: IMPS
- Results
- Future Work and Open Questions

Motivation (Why do we care?)

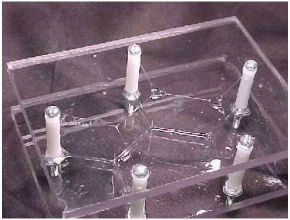
- Connect campus buildings to a central server with underground cables. Each building has a dedicated cable.
- Connect an array of radio telescopes to a data center.

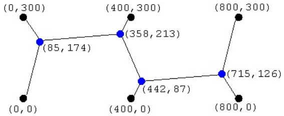
- Connect CT scan images of a vascular network.
- Build logging roads and sawmills.
- Crop irrigation
- Deploy wired and wireless networks.
- Other infrastructure development?



Background: The Euclidean Steiner Tree Problem

- Given a set of n **terminal** vertices $V \subseteq \mathbb{R}^d$,
- Determine a set of **Steiner** points $S \subseteq \mathbb{R}^d$ and
- Edges E so that the tree T with vertex set $V \cup S$ and edge set E has minimum total (Euclidean) length.
- Fact: Generally, there are $n - 2$ Steiner vertices, each of degree 3.
- Fact: Generally, each terminal vertex has degree 1.

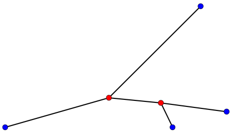


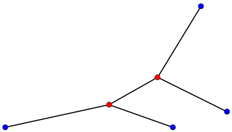


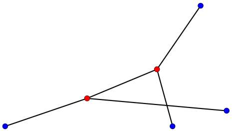
Background: The Euclidean Steiner Tree Problem

- NP-hard problem: $\frac{(2n-4)!}{2^{n-2}(n-2)!} = \Omega(n!)$ possible tree topologies – configurations and connections of the Steiner and terminal vertices.
- When $d = 2$, angles at each Steiner vertex are 120°.
- GeoSteiner (currently version 5.0.1) works in $d = 2$ and has found optimal solutions to instances of 10,000 vertices using characteristics unique to the plane.
- For $d \geq 3$, Smith (2008) proposed an iterative branch-and-bound method.
- Variants of Smith's algorithm find optimal solutions to ESTPs with 17 vertices when $d \geq 3$.

The three Steiner tree topologies for $n = 4$ terminal vertices:







Take Egon's advice. Crossing the streams would be bad. This restricts the number of topologies that we would realistically consider.

Enumerating the Topologies

There are $\frac{(2n-4)!}{2^{n-2}(n-2)!} = \Omega(n!)$ possible tree topologies.

There are $C_n = \frac{(2n)!}{(n!)(n+1)!}$ triangulations of a convex $(n+2)$-gon.

n	$\frac{(2n-4)!}{2^{n-2}(n-2)!}$	$\frac{(2n-4)!}{((n-2)!)(n-1)!}$
3	1	1
4	3	2
5	15	5
10	$2,027,025$	1430
15	$7,905,853,580,625$	$742,900$
20	$2.2 \cdot 10^{20}$	$477,638,700$
25	$2.5 \cdot 10^{28}$	$3.4 \cdot 10^{11}$
30	$8.7 \cdot 10^{36}$	$2.6 \cdot 10^{14}$

Except for very small problems, there are too many topologies for us to consider each one, even only "reasonable" ones.

- The **Cable-Trench Problem** (CTP) establishes a continuum between the shortest path (SPT) and minimum spanning tree (MST) problems.
- Suppose you needed to connect each building on a campus to the server building with its own (underground) internet cable.
- How should you dig the trenches to minimize the total cost of digging the trenches and laying the cable?
- If the cables are free, then the solution is a MST (easy).
- If the trenches are free, then the solution is a SPT (easy).
- In the real world, both cables and trenches have a cost.
- The CTP is NP-hard.

Background: The CTP – Mathematical Definition

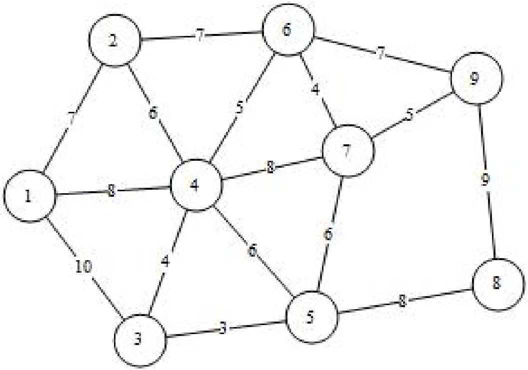
- $G = (V, E)$ is a weighted graph.
- $V = \{v_1, \ldots, v_n\}$, with root v_1.
- $|e|$ is the weight of $e \in E$.
- $T = (V, E_T)$ is a spanning tree of G.
- $\mathcal{P}(v_1, v_j)$ is the path from v_1 to v_j in T.
- $|T|_t$ is the total **trench weight** of T.
- $|T|_c$ is the total **cable weight** of T.
- $\gamma \geq 0$ is the **per-unit cable cost**.
- $\tau \geq 0$ is the **per-unit trench cost**.

Definition

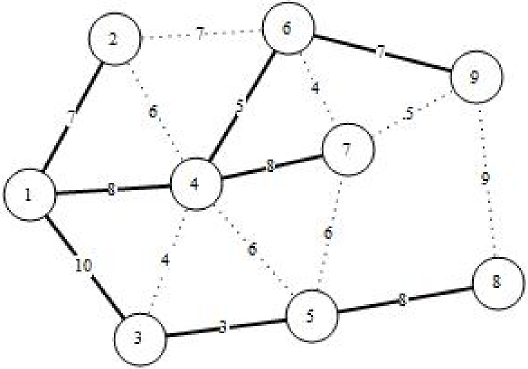
The **Cable-Trench Problem** (CTP) is the problem of finding a spanning tree T of a weighted graph G that minimizes

$$\gamma|T|_c + \tau|T|_t = \gamma \sum_{j=2}^{n} \sum_{e \in \mathcal{P}(v_1, v_j)} |e| + \tau \sum_{k=1}^{n} |e| \ .$$

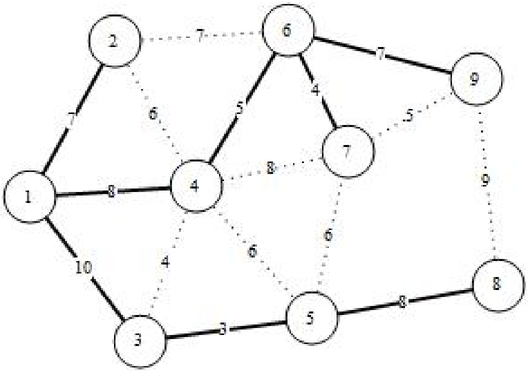
Example "4" from (Vasko et al., 2002)



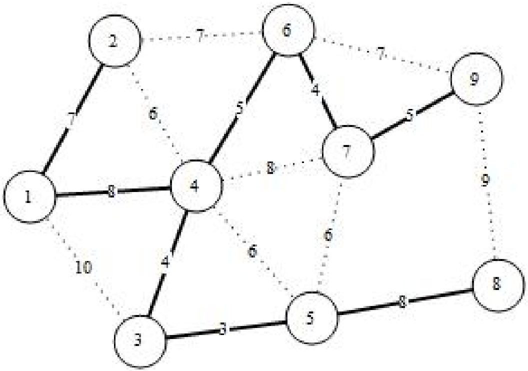
T_1: for $0 \leq \tau/\gamma \leq 1/4$; Cost: $56\tau + 108\gamma$ (SPT)



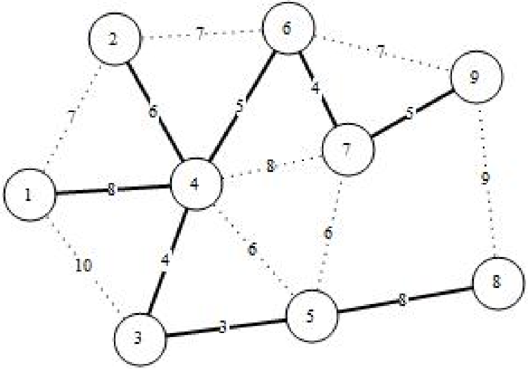
T_2: for $1/4 \leq \tau/\gamma \leq 1$; Cost: $52\tau + 109\gamma$



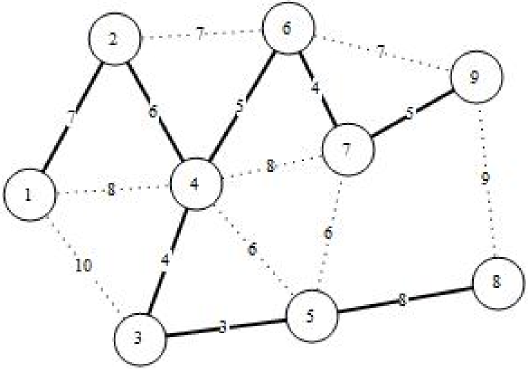
T_3: for $1 \leq \tau/\gamma \leq 7$; Cost: $44\tau + 117\gamma$



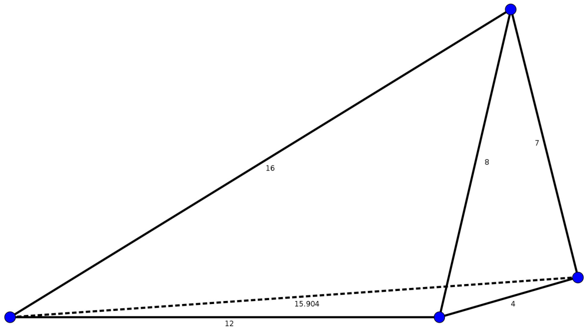
T_4: for $7 \leq \tau/\gamma \leq 28$; Cost: $43\tau + 124\gamma$



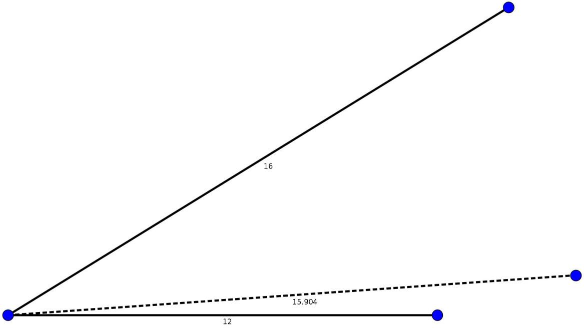
T_5: for $28 \leq \tau/\gamma$; Cost: $42\tau + 152\gamma$ (MST)



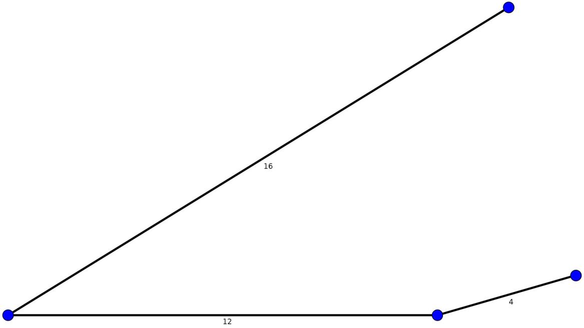
"Euclideanized" Example "1" from (Vasko et al., 2002)



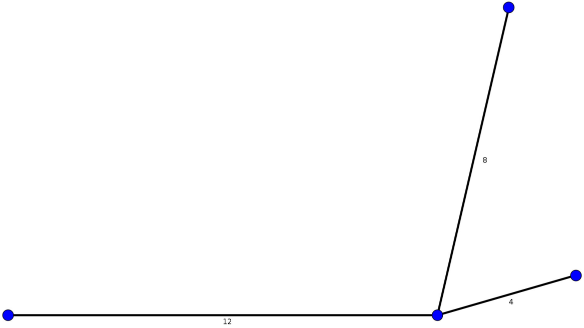
T_0: for $0 \leq \tau/\gamma \leq 0.008$; Cost: $43.904(\tau + \gamma)$ (SPT)



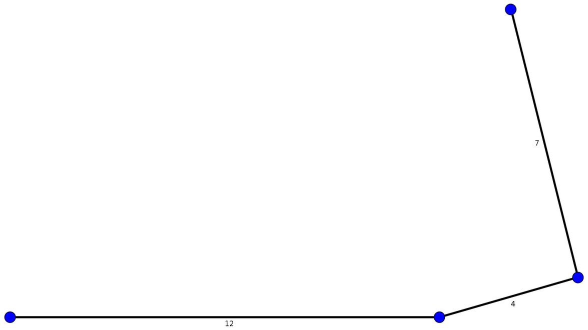
T_1: for $0 \leq \tau/\gamma \leq 1/2$; Cost: $32\tau + 44\gamma$ (SPT)



T_2: for $1/2 \leq \tau/\gamma \leq 3$; Cost: $24\tau + 48\gamma$



T_3: for $3 \leq \tau/\gamma$; Cost: $23\tau + 51\gamma$ (MST)



The Euclidean Steiner Cable-Trench Problem (ESCTP)

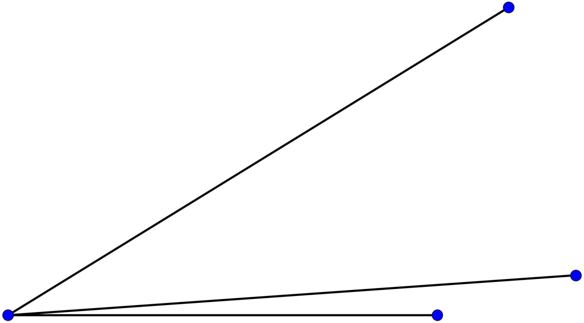
- Combines elements of the ESTP and CTP.
- Given a set of n vertices $V \subseteq \mathbb{R}^d$, $d \geq 2$, with root v_1,
- Determine a set of $(n-2)$ **Steiner** points $S \subseteq \mathbb{R}^d$ and
- Edges E so that the total cable and trench cost of the tree T with vertex set $V \cup S$ and edge set E is minimized.

Definition

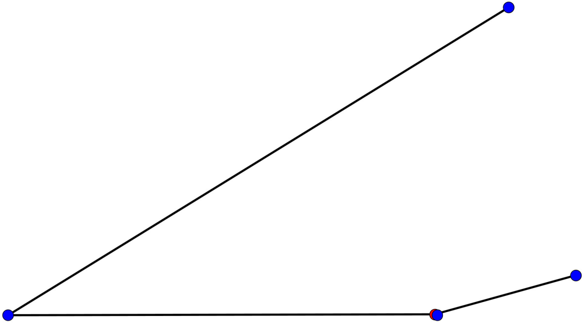
The ESCTP is the problem of finding a spanning tree T of a weighted graph G that minimizes

$$\gamma|T|_c + \tau|T|_t = \gamma \sum_{v_k \in V} \sum_{e \in \mathcal{P}(v_1, v_k)} |e| + \tau \sum_{e \in E} |e| \ .$$

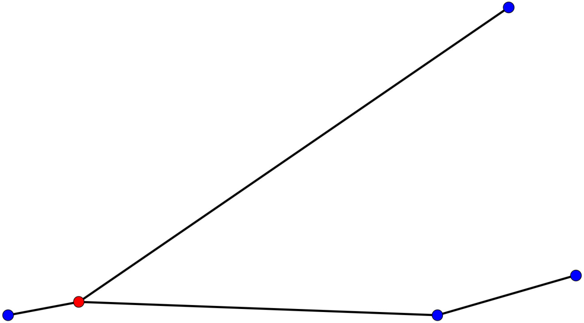
Example: $\tau = 0$, $\gamma = 1$ (SPT)



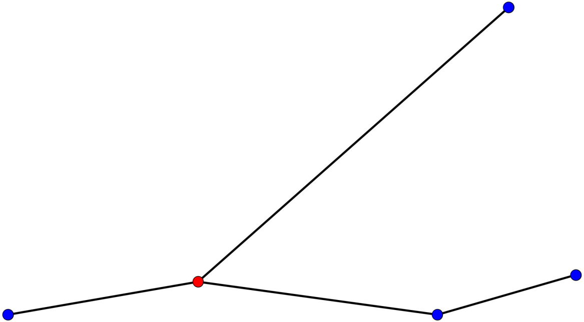
Example: $\tau = 1/16$, $\gamma = 1$



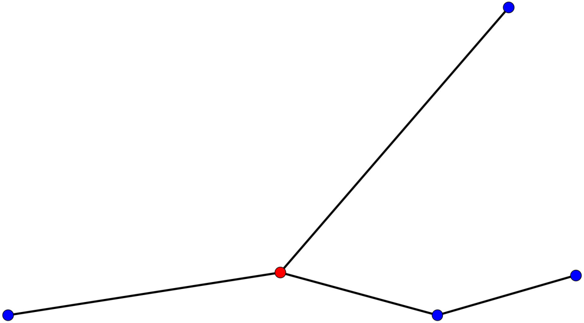
Example: $\tau = 1/8$, $\gamma = 1$



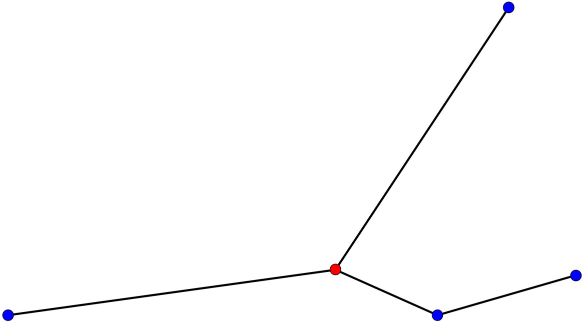
Example: $\tau = 1/4$, $\gamma = 1$



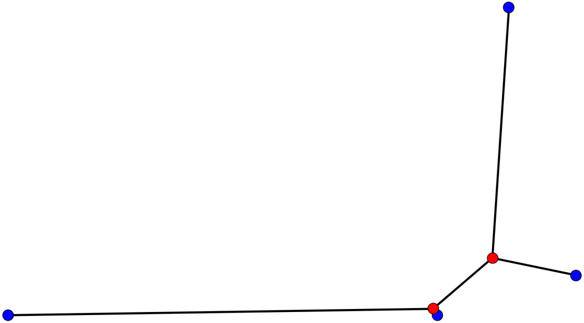
Example: $\tau = 1/2$, $\gamma = 1$



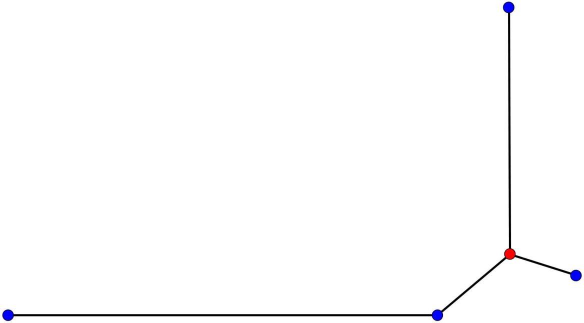
Example: $\tau = 1$, $\gamma = 1$



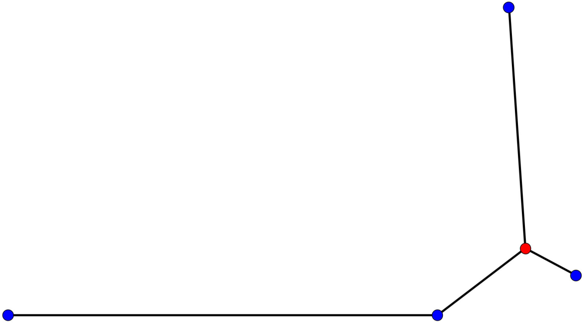
Example: $\tau = 2$, $\gamma = 1$



Example: $\tau = 4$, $\gamma = 1$



Example: $\tau = 1$, $\gamma = 0$ (SMT)



Linear Programming Formulation

- Let $v_1, v_2, ..., v_n \in V \subset \mathbb{R}^d$ be terminal vertices with root v_1.
- Let $v_{n+1}, v_{n+2}, ..., v_{2n-2} \in S \subset \mathbb{R}^d$ be Steiner vertices.
- Let $(i, j) \in E$ if $1 \leq i < j \leq 2n - 2$.
- Let $E = E_S \cup E_V$, where
- $E_S = \{(i, j) | n < i < j\}$ are edges between Steiner vertices and
- $E_V = \{(i, j) | 1 \leq i \leq n < j\}$ are edges between Steiner vertices and terminals.
- $y_{ij} = 0$ if $(i, j) \notin E$ and $y_{ij} = 1$ if $(i, j) \in E$.
- $c_{ij} \in \mathbb{N}_0$ is the number of cables in trench (i, j).

Linear Programming Formulation

Minimize:

$$\gamma \left(\sum_{i=1}^{2n-1} \sum_{j=i+1}^{2n-2} ||v_i - v_j|| \right) c_{ij} + \tau \left(\sum_{i=1}^{2n-1} \sum_{j=i+1}^{2n-2} ||v_i - v_j|| \right) y_{ij}$$

subject to:

$$\sum_{j \in S} y_{ij} = 1 \text{ for all } i \in V \qquad (1)$$

$$\sum_{i \in V} y_{ij} + \sum_{k<j, k \in S} y_{kj} + \sum_{k>j} y_{jk} = 3 \text{ for all } j \in S \qquad (2)$$

$$\sum_{k<j, k \in S} y_{kj} = 1 \text{ for all } j \in S \setminus \{n+1\} \qquad (3)$$

Linear Programming Formulation, Continued

$$\sum_{j=n+1}^{2n-2} c_{1j} = n - 1 \tag{4}$$

$$\sum_{j=n+1}^{2n-2} c_{ij} = 1 \text{ for all } 2 \leq i \leq n \tag{5}$$

$$0 \leq \sum_{i=1}^{n} c_{ij} - \sum_{k<j, k \in S} c_{kj} - \sum_{k>j} c_{jk} \leq 2 \text{ for all } j \in S \tag{6}$$

$$\sum_{i=1}^{2n-1} \sum_{j=i+1}^{2n-2} y_{ij} = 2n - 1 \tag{7}$$

$$(n-1)y_{ij} - c_{ij} \geq 0 \text{ for all } i < j \tag{8}$$

Linear Programming Formulation

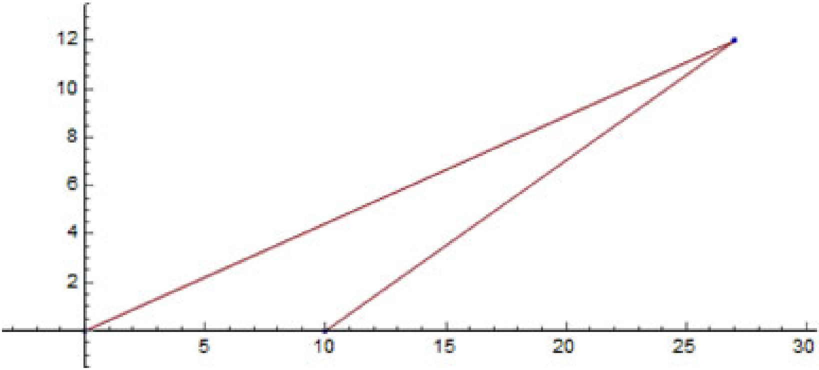
- This formulation is not likely to be very effective:
- The Steiner vertices are not fixed.
- A longer formulation involving a multi-commodity flow formulation and
- the introduction of a new parameter bounding edge lengths may give some useful bounds for relatively small examples.
- Geometric and graph-theoretic approaches are more intuitive and likely more efficient and effective.

Solution Approach: IMPS

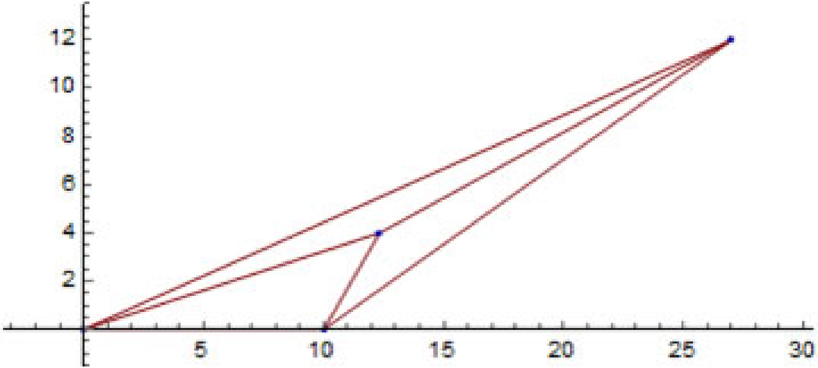
- IMPS: Iterated Midpoint Pattern Search
- Select a topology:
  - Consider all possible (reasonable) topologies (small n).
  - Consider only some topologies (larger n).
  - Base the topology on a Modified Prim's algorithm approach,
  - Consider a triangulation of the graph,
  - Start with the vertices closest to the root; gradually add more.
- Find centroids of each triangle of the implied triangulation; make connections.
- Find the three centroids of the smaller triangles; make connections.
- Determine the best of the four centroids. Iterate.

IMPS on a 3-vertex Example

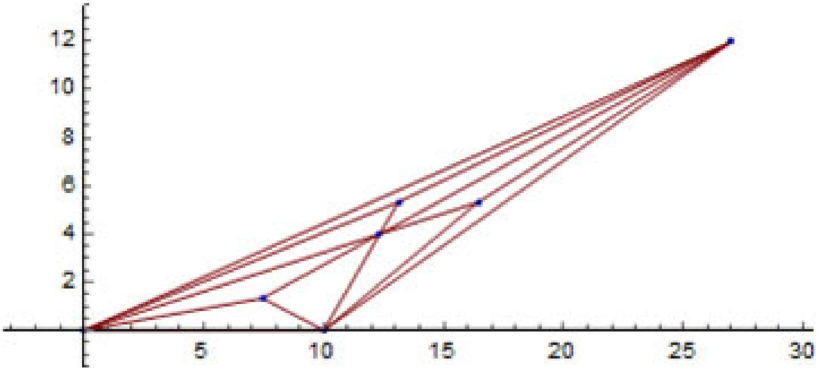
Consider the following 3-vertex graph.



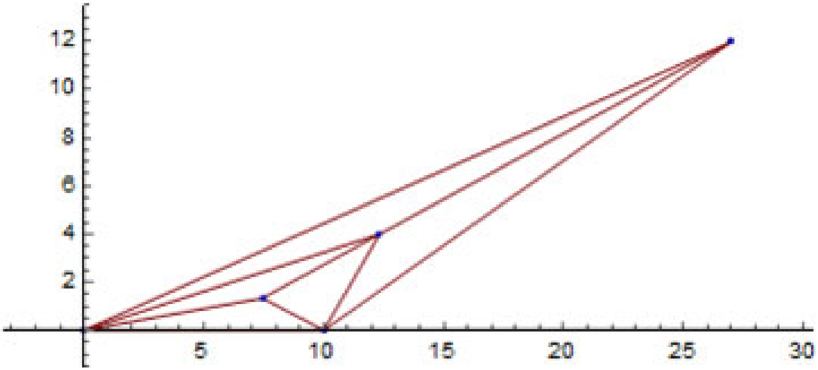
Find the centroid (aka "midpoint") of the triangle and connect it to each vertex.



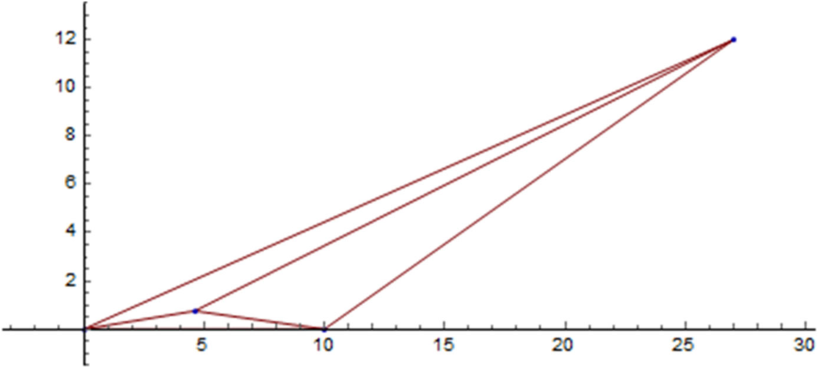
Find the centroids of the three smaller triangles.



Compare the costs associated with the four centroids.



Pick the centroid yielding the least cost. Repeat this idea until the desired precision is found.



4-vertex example: CTP vs. ESCTP solutions.

τ/γ	Opt.	CTP	CTP: K_4	%Impr./CTP	%Impr./K_4
0	43.904	44	43.904	0.22	0
1/16	46.000	46	46	0.00	0.00
1/8	47.966	48	48	0.07	0.07
1/4	51.518	52	52	0.93	0.93
1/2	58.055	60	60	3.24	3.24
1	70.482	72	72	2.11	2.11
2	94.479	96	96	1.58	1.58
4	139.918	143	143	2.16	2.16
8	230.534	235	235	1.90	1.90
∞	22.630	23	23	1.61	1.61

Future Work and Open Questions

- Improve algorithms further:
  - What is the best way to scale up IMPS?
  - Genetic algorithms?
  - Teaching-learning-based optimization?
  - Branch-and-bound techniques?
  - Are there other fundamental approaches?
- Theoretical considerations:
  - What is the biggest problem that we can solve optimally or within certain tolerances?
  - How do we find good lower bounds of optimal solutions?
  - What kind of geometric properties do optimal solutions have?
  - Is there a connection between optimal CTP solutions and the topology of optimal ESCTP solutions?

Thank you!


GHOSTBUSTERS
DON'T CROSS THE STREAMS
Good advice is timeless
ICANHASCHEEZBURGER.COM BY