The Generalized Steiner Cable-Trench Problem with Application to Error Correction in Vascular Image Analysis

Eric Landquist

Department of Mathematics, Kutztown University of Pennsylvania

elandqui@kutztown.edu

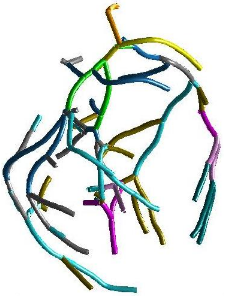
July 29, 2017

Joint work with: Fran Vasko, Greg Kresge, Adam Tal (Kutztown), Yifeng Jiang and Xenophon Papademetris (Yale University)

Outline

- Motivation: Analyzing CT Scans
- Background: The Cable-Trench Problem (CTP) and variants
- Other Applications
- Application to Vascular Image Analysis and Error Correction
- Modified Prim's Algorithms
- Results
- Conclusions
- Future Work and Open Questions

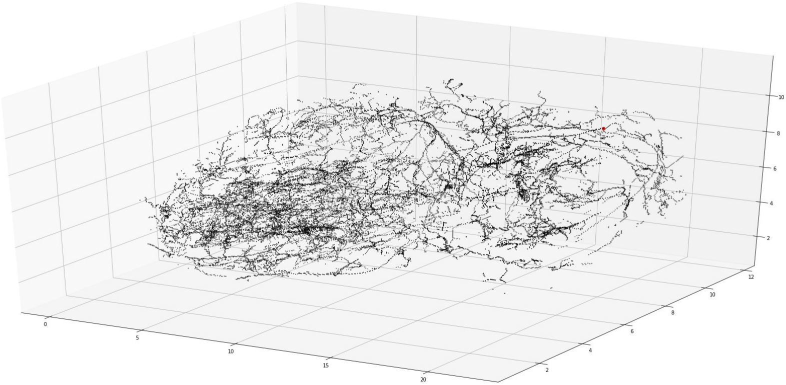
Motivation: Why do we care?



- Given: Data from a Micro-CT scan (combined microscope and CT scan).
- Digitally reconstruct a blood vessel network (vasculature) from a set of points and vessel radii and
- Automatically filter out errors,
- Accurately and efficiently.
- Study vasculature noninvasively for cancer detection, identifying blood clots, etc.

The Problem: Vascular Image Analysis from a CT Scan

- Example: This scan of a mouse's leg has $27,873$ vertices (3D).
- The red dot represents the main artery coming into the leg.
- Can we accurately and efficiently connect the dots to visualize the blood vessel network?

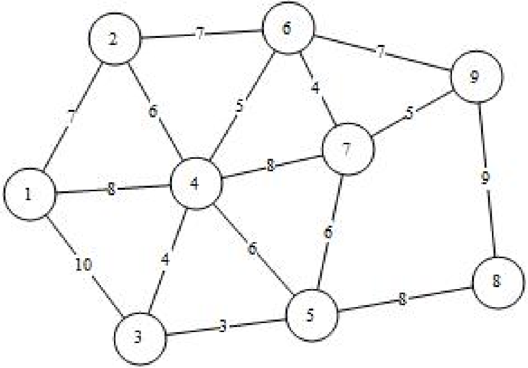


Background: The Cable-Trench Problem

- Seminal paper: Vasko et al. (2002)
- Suppose you needed to connect each building on a campus to a hub (IT building) with its own (underground) internet cable.
- How should you dig the trenches between buildings to minimize the total cost to dig the trenches and lay the cable?
- If the cables are free, then the solution is a Minimum Spanning Tree (MST).
- If the trenches are free, then the solution is a Shortest Path Tree (SPT).
- In the real world, both cables and trenches have a cost.
- The **Cable-Trench Problem** (CTP) establishes a continuum between the SPT and MST Problems on a weighted graph.

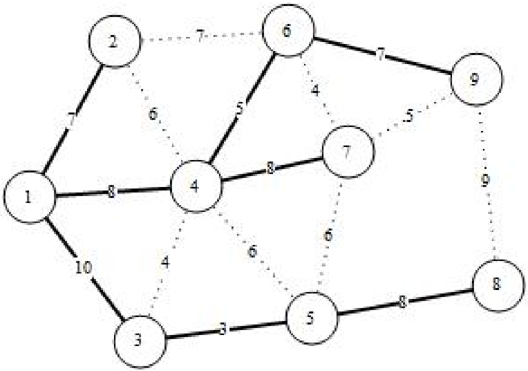
An Example: SPT vs. MST

Circles (nodes, vertices) represent the buildings; Node 1 is the hub.
Edges represent allowable routes for digging trenches.
Numbers (weights) on the edges represent their lengths.



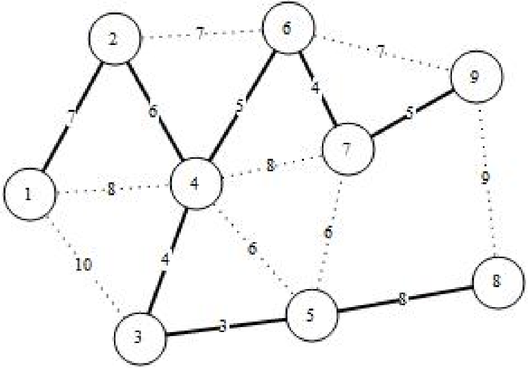
Shortest Path Tree (SPT)

If trenches are free, this minimizes the total cable weight (cost).
This is found using **Dijkstra's Algorithm**.



Minimum Spanning Tree (MST)

If cables are free, this minimizes the total trench weight (cost). This is found using **Prim's Algorithm**.



Background: The Cable-Trench Problem

- Let $G = (V, E)$ be a weighted graph, $V = \{v_1, \ldots, v_n\}$, with root node v_1, and t_{ij} the weight of $(v_i, v_j) \in E$.
- $w_c(T)$ is the total weight of the paths, $\mathcal{P}(v_1, v_j)$, from v_1, to each $v_j \in V$ of a spanning tree T of G: **total cable weight**.
- $w_t(T)$ is the total weight of T: **total trench weight**.
- Let $\gamma \geq 0$ be the **per-unit cable cost** and
- $\tau \geq 0$ the **per-unit trench cost**.

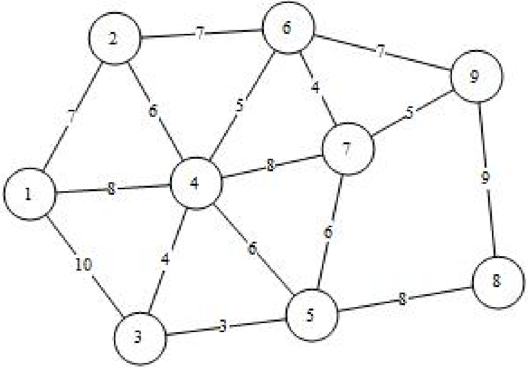
Definition

The **CTP** is the problem of finding a spanning tree $T = (V, E_T)$ of G that minimizes the total cable and trench weight (cost):

$$\gamma w_c(T) + \tau w_t(T) = \gamma \sum_{v_k \in V} \sum_{(v_i, v_j) \in \mathcal{P}(v_1, v_k)} t_{ij} + \tau \sum_{(v_i, v_j) \in E_T} t_{ij} \ .$$

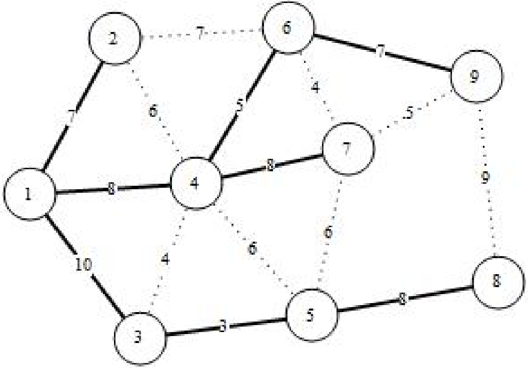
A CTP Example: Revisited

What are the optimal CTP solutions?



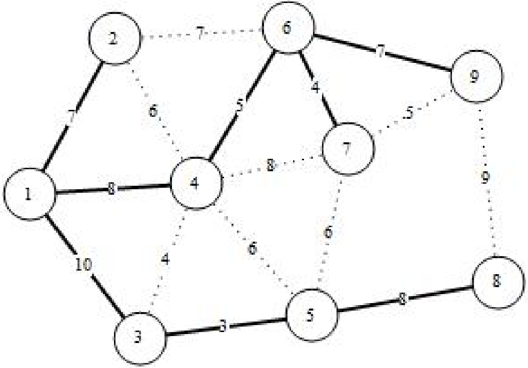
T_1: for $0 \leq \tau/\gamma \leq 1/4$; Cost: $56\tau + 108\gamma$ (SPT)

$\gamma, \tau \geq 0$ are the per-unit cable and trench costs, respectively.



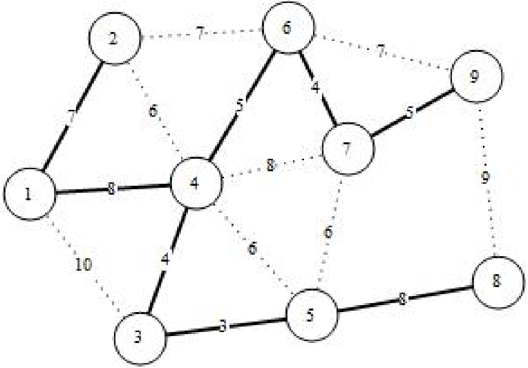
T_2: for $1/4 \le \tau/\gamma \le 1$; Cost: $52\tau + 109\gamma$

$\gamma, \tau \ge 0$ are the per-unit cable and trench costs, respectively.



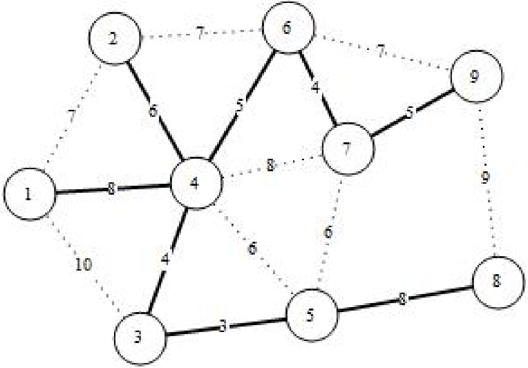
T_3: for $1 \leq \tau/\gamma \leq 7$; Cost: $44\tau + 117\gamma$

$\gamma, \tau \geq 0$ are the per-unit cable and trench costs, respectively.



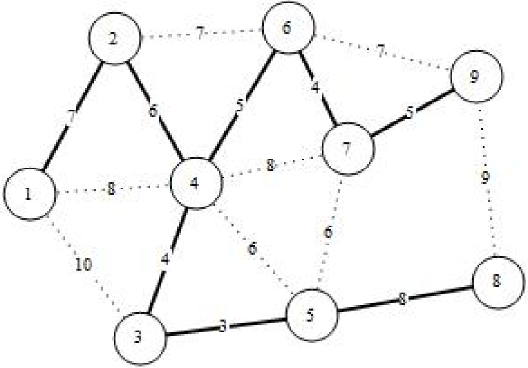
T_4: for $7 \le \tau/\gamma \le 28$; Cost: $43\tau + 124\gamma$

$\gamma, \tau \ge 0$ are the per-unit cable and trench costs, respectively.



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

$\gamma, \tau \geq 0$ are the per-unit cable and trench costs, respectively.



A Generalized Cable-Trench Problem

- What if the edge weights don't measure both the trench and cable weights, due to rocks, obstacles, overhead costs, etc.?
- For $(v_i, v_j) \in E$, let s_{ij} be the **cable weight** of (v_i, v_j) and
- t_{ij} the **trench weight** of (v_i, v_j).
- $w_c(T)$ and $w_t(T)$ are the total cable weight and total trench weight of the tree T.
- $\gamma, \tau \geq 0$ are the per-unit cable and trench costs, respectively.

Definition

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

$$\gamma w_c(T) + \tau w_t(T) = \gamma \sum_{v_k \in V} \sum_{(v_i,v_j)\in\mathcal{P}(v_1,v_k)} s_{ij} + \tau \sum_{(v_i,v_j)\in E_T} t_{ij} \ .$$

A Generalized Steiner Cable-Trench Problem

- What if we only require the solution tree T to contain a given subset, $F \subseteq V$?
- F is called the set of **terminal nodes**,
- and $N = V \setminus (F \cup \{v_1\})$ the set of **nonterminal nodes**.
- $w_c(T)$ and $w_t(T)$ are the total cable weight and total trench weight of the tree T.
- $\gamma, \tau \geq 0$ are the per-unit cable and trench costs.

Definition

The **Generalized Steiner Cable-Trench Problem** (GSCTP) is the problem of finding a subtree $T = (V_T, E_T)$ of G such that $\{v_1\} \cup F \subseteq V_T$ that minimizes

$$\gamma w_c(T) + \tau w_t(T) = \gamma \sum_{v_k \in F} \sum_{(v_i, v_j) \in \mathcal{P}(v_1, v_k)} s_{ij} + \tau \sum_{(v_i, v_j) \in E_T} t_{ij} \ .$$

Some Applications of the CTP and Generalizations

- Let multiple utilities share a common trench network. (Schwarze and Lalla-Ruiz)
- Connect an array of radio telescopes. (Girard, Zyma et al.)
- Construction of logging roads and sawmills. (Marianov et al.)
- Construction of irrigation canals and wells. (Marianov et al.)
- Layout of wireless and wired access networks. (Nielsen et al., Calik et al., Lalla-Ruiz et al.)
- Carpooling (Combine individuals' time and the cost of gas.)
- Other infrastructure development or vehicle routing?

Vascular Image Analysis and Murray's Principle (1926)

- Good solution (previous state-of-the-art): Find the MST of the graph from a Micro-CT scan via Prim's Algorithm, using vessel segment volume as the edge weight.
- Better idea: Incorporate Murray's Minimum Work Principle.
- Work in a circulatory system is primarily due to two factors:
  1. Overcome blood flow resistance (friction; inversely proportional to vessel radius), and
  2. Metabolic support for the blood volume (proportional to vessel segment volume).
- The body minimizes the total work due to these two factors.
- Applied by Jiang, et al. to improve on the MST model. (2011)

The GCTP Model for Vascular Image Analysis

- "Cables:" Work to overcome friction
- "Trenches:" Work due to metabolic support
- Notation:
  - $r(e_k) =$ Radius of vessel segment $e_k = (v_i, v_j) \in E$
  - $\ell(e_k) =$ Length of vessel segment $e_k \in E$
- Total Cable Length:

$$w_c(T) = \sum_{j=2}^{n} \sum_{e_k \in \mathcal{P}(v_1, v_j)} \frac{\ell(e_k)}{r(e_k)} \ , \ \text{so} \quad s_{ij} = \frac{\ell(e_k)}{r(e_k)}$$

- Total Trench Length:

$$w_t(T) = \sum_{k=1}^{n-1} \ell(e_k) \cdot r^2(e_k) \ , \ \text{so} \quad t_{ij} = \ell(e_k) \cdot r^2(e_k)$$

- Find edges $e_1, \ldots, e_{n-1} \in E$ to minimize the total work:

$$\gamma w_c(T) + \tau w_t(T) = \gamma \sum_{j=2}^{n} \sum_{e_k \in \mathcal{P}(v_1, v_j)} \frac{\ell(e_k)}{r(e_k)} + \tau \sum_{k=1}^{n-1} \ell(e_k) \cdot r^2(e_k)$$

- $\gamma = 1$ and $35000 \leq \tau \leq 175000$ for this application.

Error-correction in Vascular Imaging

- Problem: There are errors in the imaging process.
- Current methods require tedious manual correction.
- Can we automate error-correction using a CTP-based model?
- Information: The actual leaf nodes can be determined.
- Solution: Let F be the set of leaf nodes of T.
- The other (**Steiner**) nodes are optional.
- Vascular imaging error-correction is a GSCTP!

- Next problem: Find good solutions to large GSCTP instances using efficient heuristics.
- The CTP and its generalizations are NP-hard.
- The obvious exceptions are $\tau = 0$ (SPT) and $\gamma = 0$ (MST).
- Determining optimal solutions of large instances of the CTP and GSCTP is computationally infeasible.

Must go faster. Must go faster.

Modified Prim's Algorithm for the GCTP (`MPrim`)

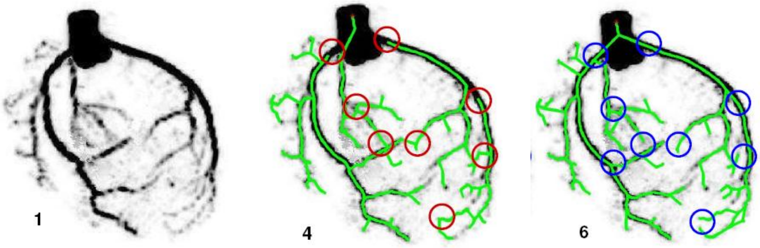
- Combines Dijkstra's (SPT) and Prim's (MST) Algorithms.
- Build out a solution tree from the root.
- **Theorem**: Running time and space is $O\left(n^2\right)$.

1. Initialize the solution tree T as the root vertex v_1.
2. `While` T does does contain V:
    - Find the edge in $G \setminus T$ adjacent to T that would contribute the lowest total cable plus trench cost.
    - Add that edge and vertex to T.
3. Return T.

The Steiner `MPrim` Algorithm for GSCTP: `SMPrim`

- We can further modify `MPrim` for the GSCTP.
- Two key changes:
  1. Stop when the solution tree, T, contains all terminal nodes, F.
  2. "Trim" any leaf nodes of T that are in N, the set of nonterminal nodes.
- **Theorem**: Running time and space is $O(n^2)$.
- `MPrim` and `SMPrim` give good, but not always optimal, solutions.

MST Model vs. GCTP Model



- The right-most image used `MPrim`.
- Blue circles indicate improvements.
- There is still room for improvement. Can we do better?

Moral of the story thus far

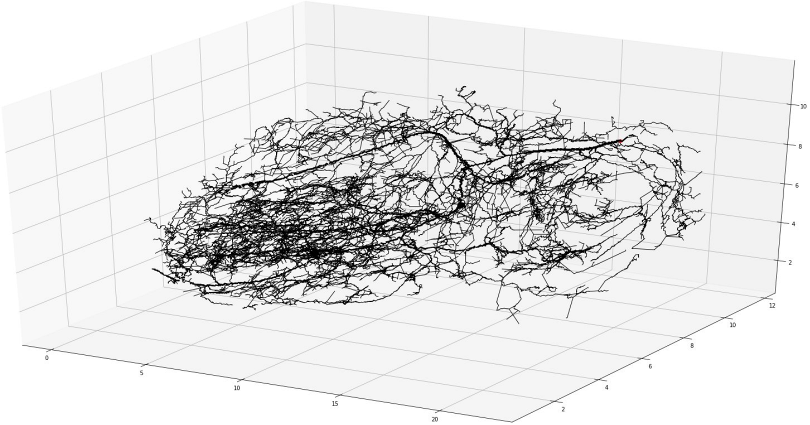
"I like fast algorithms. They're kind of like sports cars for nerds."
(Nate Wentzel)

Three Variations of MPrim and SMPrim

- All variations require $O(n^2)$ time and space.

1. Encourage the inclusion of terminal nodes and Steiner nodes adjacent to terminal nodes.
    - Steiner nodes are good branching-off points to reach terminal nodes.
    - This approach tended to work the best.
2. Semi-Greedy (deterministic) approach
3. Partially-Stochastic (randomize things at the start) approach.

Results: `MPrim` on the Mouse Leg, $\tau = 100,000$



Conclusions

- We accurately modeled a vascular imaging problem as a new graph-theoretic problem.
- We were able to accurately eliminate errors without manual correction.
- We found very good suboptimal solutions to GSCTPs using `SMPrim` and various enhancements.
- If more accurate results are desired to instances of the GSCTP, `SMPrim` and its variants yield excellent “seeds” to initialize other heuristics.

Future Work and Open Questions

- Use so-called "1-opt local search" techniques to tweak and improve solutions.
- Experiment with Teaching-Learning-Based Optimization, Jaya, and other population-based heuristics.
- Use Lagrangean relaxation to determine good lower bounds for solutions to the GSCTP.
- More rigorous statistical analysis.
- Are there other good heuristic approaches?
- Theoretical considerations:
  - How close to optimal is `SMPrim` guaranteed to be?
  - When are we guaranteed optimal solutions?

Thank you!

elandqui@kutztown.edu