Northeastern University
College of Computer and Information Science

Retracting Graphs to Cycles

Sam Haney[2], Mehraneh Liaee[1], Bruce Maggs[2], Debmalyy Panigrahi[2], Rajmohan Rajaraman[1], Ravi Sundaram[1]
[1]Northeastern University, [2]Duke University

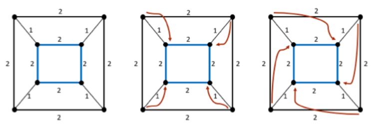
Problem Definition

Graph Retraction Problem:
Given an unweighted *guest* graph $G=(V,E)$ and a *host* subgraph $H=(A,E')$ of G, a mapping $f: V \rightarrow A$ is a retraction of G to H, if $f(v)=v$ for all $v \in A$.

Objective: The goal is to find a mapping function f such that $\max_{(u,v)\in E}\{d_H(u,v)\}$ is minimized, where d_H is the distance metric induced by H. Such mapping function f is called **minimum-stretch graph retraction**.

The **stretch** of a retraction is defined to be the maximum distance between the images of the endpoints of any edge, as measured in the subgraph.

Example:



a) Graph G and subgraph H b) Optimal embedding with stretch 1 c) An embedding with stretch 2

Applications

Our original motivation for studying minimum-stretch graph retraction comes from a distributed systems scenario where the aim is to map processes comprising a distributed computation to a network of servers where some processes are constrained to be mapped onto specific servers.
The objective is to minimize the maximum communication latency between two communicating processes in the embedding.

Such anchored embedding problems can be shown to be equivalent to graph retraction for general subgraphs, and arise in several other domains including:
- VLSI layout
- Multi-processor placement
- Graph drawing and visualization

Related Works

- **0-extension and metric labeling problem**
  In the 0-extension problem, one seeks to minimize the average stretch, which can be solved to an $O(\log k / \log\log k)$ approximation using a natural LP relaxation. In contrast, we give polynomial integrality gaps for the graph retraction problem.
- **Minimum bandwidth problem**
  In the minimum bandwidth problem, the objective is to minimize maximum stretch, but the constraint is that the map must be isomorphic rather than that the anchor vertices must be fixed. Minimum bandwidth problem can be solved to a poly logarithmic approximation.
- **Low-distortion embedding**
  - Low-distortion embedding, where it is considered embedding one specific n-point metric to another n-point metric, typically require non-contracting isomorphic maps, which distinguishes them significantly from the graph retraction problem.

Overview of Results

The graph retraction problem is easy if the subgraph H is acyclic; Though, the problem becomes NP-complete even when H is just a 4-cycle.

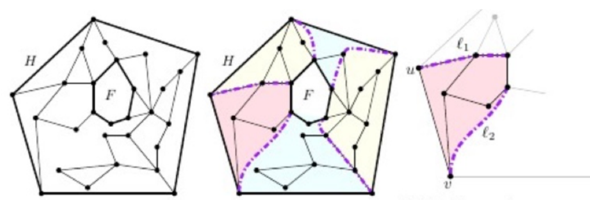
Given this intractability result, a natural goal is to obtain an algorithm for retracting graphs to cycles that ***approximately*** minimizes the stretch of the retraction. Here, we present our three main results:

- $O(min\{k,\sqrt{n}\})$-approximation algorithm for retracting an arbitrary graph to a cycle, where k and n represent the number of vertices in the cycle and the graph respectively.
- Polynomial-time optimal algorithm for retracting a planar graph to a cycle.
- Constant approximation algorithm for retracting a set of points in the Euclidean plane to a uniform cycle of points, building on our planar graph algorithm.

Retracting a Planar Graph to a Cycle

Algorithm 1: *Overview of the optimal algorithm*
.Input Planar Graph G, host cycle H of size k
.output Mapping function f

1. **Redraw G:** such that H is the outer face of G.
2. While $i < k/2$
   1. **Compute G_i:** Replace each edge of G (not in H) by a path of i edges.
   2. **Find a stretch-1 retraction** f**:** of G_i to H in the following way:
      1. For each inner face F in G_i do
         Compute maximum number of valid curves $p_1, p_2, \dots, p_l$ between F and H. (see figure 1)
         If $l=k$ then
         Compute stretch-1 retraction f from G_i to H using $p_1, p_2, \dots, p_k$.
   3. If stretch-1 retraction found return f.



(a) A graph G. The outer cycle H and the face F are shown in bold. **(b)** Non-intersecting curves partition the region contained in H but not in F. **(c)** Vertices on ℓ_1 are mapped to u, and vertices on ℓ_2 are mapped to v. All other vertices in the region are mapped arbitrarily to u or v.

Figure 1. Using non-intersecting curves to find an embedding from face F to H.

Retracting an arbitrary Graph to a Cycle

Algorithm 2: Overview of the algorithm with $O(min\{k,\sqrt{n}\})$-approximation factor
.Input Graph G, host cycle H
.Output Embedding function f

1. **Embedding in a grid:** Determine embedding g from G into a $k/4 * k/4$ grid M such that H is embedded one-to-one to the boundary of M and for every $u, v \in V$, d_∞(g(u),g(v)) $\le l(G,H)\, d_G(u,v)$.
2. **Find largest hole:** Find the largest square sub-grid D of M such that
   (a) its center c is at L_∞ distance at most $k/16$ from the center of M, and
   (b) there is no vertex u in G for which $g(u)$ is in the interior of D.
3. **Projection embedding:** For all v in G:
   1. $R(v) \leftarrow$ ray originating from the center D and passing through $g(v)$.
   2. $f(v) \leftarrow$ the anchor in H nearest in the clockwise direction to the intersection of $R(v)$ with the boundary of M.

Return f

Retracting Points on the 2D Plane to a Uniform Cycle

Algorithm 3: Overview of the algorithm with constant factor approximation
.Input Points set V, anchor set $A \subseteq V$ which forms a uniform cycle
.Output Retraction function ϕ from V to A

1. **Construct a planar spanner:** Set weighted graph $\hat{G}$ to be a planar Euclidean distance spanner for the points set V, with the weight $w(e)$ of an edge $e=(u,v)$ denoting the Euclidean distance between the two vertices u and v.
2. **Contract very small edges:** Set weighted graph $\bar{G}$ to be the graph obtained from $\hat{G}$ by following steps.
   1. Contract all edges of weight less than $2/kn$.
   2. Remove self loops, and among parallel edges, remove all but the shortest edge.
3. **Convert to unweighted graph:** Construct unweighted graph G' as follows.
   Replace each edge e by a new path consisting of $\lfloor kn\, w(e)/2 \rfloor$ edges.
4. **Construct cycle H:** Obtain simple cycle H by taking union of shortest path between any pair of consecutive anchors.
5. Set ϕ' to be the optimal retraction of G' to H, obtained by algorithm 1.
6. For each vertex $v \in V$, set $\phi(v)$ to be the closest anchor to ϕ'(v).

Return ϕ

Open Problems

- Obtaining a better approximation factor for retracting an arbitrary graph to cycle
- Studying graph retraction of a planar graph to a subgraph of it, not necessarily a cycle.

Contact

Mehraneh Liaee
College of Computer and Information Science, Northeastern University
Email: mehraneh@ccis.neu.edu