

The Local Connectivity ATSP Problem

Fix some function $\text{lb} : 2^V \rightarrow \mathbb{R}^+$ that obeys the following properties:

1. $\text{lb}(S) = \sum_{v \in S} \text{lb}(\{v\})$ for all $S \subseteq V$
2. $\text{lb}(V) = \text{value}(I)$

One can think of lb as being defined by distributing the LP value among the vertices in some way.

Problem Statement.

Input: I (the instance) and V_1, \dots, V_k (a partitioning of V) such that V_i is strongly connected.

Output: F , an eulerian set of edges such that for every i , F exits (and hence enters) V_i at least once.

Definition 0.1. An algorithm for local connectivity ATSP is called **α -light** if it outputs F such that for each strongly connected component of F , say \tilde{G} , $w(\tilde{G}) \leq \alpha \cdot \text{lb}(\tilde{G})$.

Theorem 0.1 (Svensson '15). *An α -light algorithm for local connectivity ATSP implies a $O(\alpha)$ -approximation algorithm for ATSP.*

The paper describes an $O(1)$ -light algorithm for local connectivity on vertebrate pairs, which gives us an $O(1)$ -approximation algorithm for ATSP on vertebrate pairs, and hence an $O(1)$ -approximation algorithm for general ATSP.

We will also assume that $w(B) \leq O(1) \cdot \text{value}(I)$, since in the previous reduction, this inequality is always true.

Definition 0.2. For this algorithm, we will define lb as

$$\text{lb}(v) := \begin{cases} \frac{\text{value}(I)}{|V(B)|} & \text{if } v \in B \\ 2 \cdot y_v & \text{otherwise} \end{cases}$$

Note that this definition does not guarantee that $\text{lb}(V) = \text{value}(I)$, but $\text{lb}(V)$ is always a constant factor away from $\text{value}(I)$, so we can work with these lb values and suffer a constant factor loss in the approximation ratio.

To make corner cases easier, we add V to $L_{\geq 2}$ for the rest of this write up.

The following is the main technical lemma of the paper.

Lemma 0.1 (Lemma 7.3 in the paper). *Given U_1, \dots, U_l disjoint subsets of $V \setminus V(B)$ that are strongly connected in G , such that for each $S \in L_{\geq 2}$, either $U_i \in S$ or U_i is disjoint from S , we can find eulerian $F \subseteq E$ such that:*

- (1) $w(F) \leq 3 \cdot \text{value}(I)$
- (2) $|\delta_F^-(U_i)| \geq 1$ for all i
- (3) $|\delta_F^-(v)| \leq 4$ for all v such that $x(\delta^-(v)) = 1$
- (4) Any subtour in F that crosses a set in $L_{\geq 2}$ visits a vertex of the backbone

We will use the lemma to prove the following theorem, and prove the lemma later on.

Theorem 0.2. *There is an $O(1)$ -light algorithm for local connectivity ATSP on vertebrate pairs.*

Proof. We receive (I, B) and V_1, \dots, V_k as input. We output $F^* = B \cup P \cup F$, where P and F are defined as:

P : If B exists entirely within some V_i , then we pick $u \in B$ and $v \notin V_i$, and set P to be the shortest cycle containing u and v . Otherwise we keep P empty.

F : WLOG, let V_1, \dots, V_l be the partitions that are disjoint from B . Now let V'_i be the intersection of V_i with a minimal set in $L_{\geq 2}$ that V_i intersects. Let U_i be the first strongly connected component of V'_i (in the topological order). Set F to be the eulerian set guaranteed by running the algorithm of Lemma 0.1 on U_1, \dots, U_l .

Now we need to verify that $B \cup P \cup F$ is feasible, and that it is $O(1)$ -light.

For each V_i , there are three possibilities:

- (a) V_i completely contains the backbone. The set P guarantees that F^* enters V_i at least once.
- (b) V_i partially contains the backbone. In this case, B trivially crosses V_i , so F^* must enter V_i .
- (c) V_i is disjoint from the backbone. In this case, (2) tells us that some edge (u, v) must enter U_i . $u \notin V'_i$ since U_i is the source component of V'_i . So either $u \in S \setminus V_i$ or $u \notin S$. If the former is true, then (u, v) enters V_i from outside. If the latter is true, then (u, v) crosses S , and guarantee (4) of Lemma 0.1 tells us that the subtour containing (u, v) visits the backbone, which is outside V_i .

In any case, F^* crosses each V_i , so it is indeed feasible.

Now consider any strongly connected component of F^* , say \tilde{G} . We would like to bound $w(\tilde{G})$ by $O(\text{lb}(\tilde{G}))$. There are two possibilities for \tilde{G} :

- (a) \tilde{G} contains the backbone. So

$$w(\tilde{G}) \leq w(B) + w(P) + w(F) \leq O(\text{value}(I)) + \text{value}(I) + 3 \cdot \text{value}(I) = O(\text{value}(I))$$

$$\text{lb}(\tilde{G}) \geq \text{lb}(V(B)) = \text{value}(I)$$

And it follows that $w(\tilde{G}) \leq O(\text{lb}(\tilde{G}))$.

- (b) \tilde{G} is disjoint from the backbone. In this case, (4) guarantees that \tilde{G} did not cross any set in $L_{\geq 2}$, so the only sets it might have crossed are singletons. Additionally, (3) guarantees that it crosses each singleton at most four times.

$$w(\tilde{G}) \leq \sum_{v \in V(\tilde{G})} 4 \cdot 2 \cdot y_v = 4 \cdot \text{lb}(\tilde{G})$$

□

Proof of Lemma 0.1: The final ingredient

Order the laminar sets in $L_{\geq 2}$ by size so that $|S_1| \leq |S_2| \leq \dots \leq |S_l| = |V|$. For each $v \in V$, define $\text{level}(v)$ to be the index of the smallest set it is contained in.

Definition 0.3. An edge e is called a **forward edge**, **backward edge**, or **neutral edge** if $e \in E_f$, $e \in E_b$, or $e \in E_n$ respectively, where E_f, E_b , and E_n are defined as:

- $E_f := \{(u, v) \in E \mid \text{level}(u) > \text{level}(v)\}$
- $E_b := \{(u, v) \in E \mid \text{level}(u) < \text{level}(v)\}$
- $E_n := E \setminus (E_f \cup E_b)$

Definition 0.4. Call G_{sp} the **split graph**, where we define it as follows: for each $v \in V$, create v^0 and v^1 in $V(G_{sp})$. Create the following edges along with weights w_{sp} :

- (v^0, v^1) for all $v \in V$ with weight 0. Call these edges 0 – 1 edges.
- (v^1, v^0) for all $v \in V(B)$ with weight 0. Call these edges 1 – 0 edges.
- (u^1, v^1) for all $(u, v) \in E_f \cup E_n$ with weight $w(u, v)$. Call these edges 1 – 1 edges.

– (u^0, v^0) for all $(u, v) \in E_b \cup E_n$ with weight $w(u, v)$. Call these edges 0 – 0 edges.

Consider the lift of some subtour in G_{sp} that crosses at least one set in $L_{\geq 2}$. Let S be the smallest set in $L_{\geq 2}$ that the subtour crosses. Then the edge of the subtour going into S must be a forward edge (i.e. a 1 – 1 edge) by definition, and the edge going out of S must be a backward edge (i.e. a 0 – 0 edge). This means at some point, the subtour must use a 1 – 0 edge, which it can only do if it crosses the backbone.

So if we restrict ourself to lifts of subtours of G_{sp} (instead of all subtours of G), guarantee (4) will always be true.

Lemma 0.2. *We can find x_{sp} , an eulerian vector on G_{sp} that the image of x_{sp} in G is x .*

If we want the image of x_{sp} to be x , we must necessarily have the following conditions:

- (a) $x_{sp}(u^0, v^0) = 0$, for all $(u, v) \in E_f$, since there is no (u^0, v^0) edge in G_{sp} .
- (b) $x_{sp}(u^1, v^1) = 0$, for all $(u, v) \in E_b$, since there is no (u^1, v^1) edge in G_{sp} .
- (c) $x_{sp}(u^0, v^0) + x_{sp}(u^1, v^1) = x(u, v)$ for all $(u, v) \in E$, since the image of x_{sp} in G must be x .

After defining some x_{sp} values on the 0 – 0 and 1 – 1 edges, there will be some “potential” on each vertex (i.e. the net outgoing flow), and since x was eulerian, the potential of the 0-copy of a vertex, say v , is exactly the negative potential of the 1-copy. If the potential of the 1-copy is positive, v^0 and v^1 can be brought down to zero potential by routing that amount of flow through (v^0, v^1) . But if it is negative, this can only be equalized using a (v^1, v^0) edge, which is impossible unless $v \in V(B)$. In other words, if we want x_{sp} to be eulerian, it must also fulfil the following condition:

- (d) $x_{sp}(\delta^-(v_1) \setminus (v^0, v^1)) \geq x_{sp}(\delta^+(v_1) \setminus (v^0, v^1))$ for all $v \notin V(B)$.

It can easily be seen that (a), (b), (c), and (d) are also sufficient for some nonnegative x_{sp} to fulfil the guarantees of Lemma 0.2. This gives rise to the following LP and its dual where f can be interpreted as x_{sp} restricted to 1 – 1 edges:

$$\begin{array}{ll|ll}
 \max & \sum_{e \in E_f} f(e) & \min & \sum_{e \in E_f \cup E_n} x(e)z(e) \\
 s.t. & f(\delta^+(v)) \geq f(\delta^-(v)) & & \pi_v - \pi_u + z(u, v) \geq 1 \quad (u, v) \in E_f \\
 & f(e) = 0 & & \pi_v - \pi_u + z(u, v) \geq 0 \quad (u, v) \in E_n \\
 & 0 \leq f(e) \leq x(e) & & \pi_v = 0 \quad v \in V(B) \\
 & & & \pi, z \geq 0
 \end{array}$$

Once we have some primal feasible f , we can extend it by defining x_{sp} as f on 1 – 1 edges, $x - f$ on 0 – 0 edges, and eliminating potential differences between 0 and 1 copies using 0 – 1 and 1 – 0 edges. The first inequality of the LP guarantees (d) by definition. The second inequality guarantees (b) by definition. And the way we extend f to 0 – 0 edges guarantees (c). Now if we know that the primal optimal value is equal to $\sum_{e \in E_f} x_e$, then it guarantees (a) as well.

Claim 0.1. *The primal optimal value is equal to $\sum_{e \in E_f} x_e$.*

Proof. Note that if there were a dual optimal solution where $\pi = 0$, then the dual optimal (and hence primal optimal) value would be precisely $\sum_{e \in E_f} 1 \cdot x_e$. So consider any dual optimal solution, say (z, π) . Our goal is to convert it into a dual optimal solution whose π has strictly smaller support. Running this procedure repeatedly proves that there is a dual optimal where $\pi = 0$.

Let T be the set of vertices in $\text{support}(\pi)$ with the smallest level, and let S be the set corresponding to that level. Also, let $F = \delta(V \setminus S, T)$, and $F' = \delta(T, S \setminus T)$. \square