An Approximation Algorithm for the Weighted Hamiltonian Path Completion Problem on a Tree

Author: Aaron Solomon

Dr. C. L. Lu

Prof. R. C. T. Lee

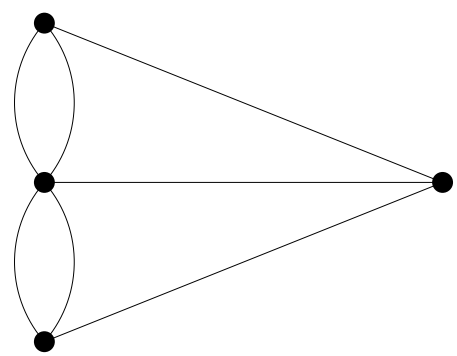
National Center for High-Performance Computing

Hsinchu, Taiwan

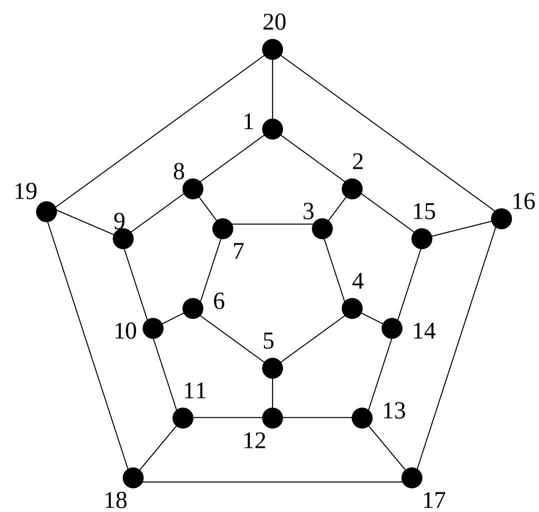
December, 2000

Graph Traversing

Euler 1736



Hamilton 1859



Hamiltonian Path Completion Problem

On unweighted graphs:

Given $G = (V, E_0)$, find an augmenting edge set E' with minimum cardinality such that $G' = (V, E_0 \cup E')$ has a Hamiltonian path.

1. NP-complete
2. P if G is a tree, a forest, an interval graph, a circular-arc graph, a bipartite permutation graph, etc.

Weighted Hamiltonian Path Completion Problem

Given a complete graph $G = (V, E)$, $w : E \rightarrow R^+$, $E_0 \subseteq E$, find an augment $E' \subseteq E$ such that $G' = (V, E_0 \cup E')$ has a Hamiltonian path and $\sum_{e \in E'} w(e)$ is minimized.

We shall restrict our discussion on the cases that E_0 constitutes a tree.

Approximation

- It is believed that there exists no polynomial time algorithm that is able to find the optimum solution for any NP-complete problem.
- If we relax our goal to find an ”nearly optimal” solution instead of an optimal one, polynomial time algorithms may exist for some NP-complete problems.

Performance Ratio

Definition: The performance ratio of an approximation algorithm of problem A is α, if for all instance I of problem A,

$$\frac{APX_A(I)}{OPT_A(I)} \leq \alpha$$

- For minimum node cover problem, there exists an approximation algorithm with performance ratio 2.
- For TSP, no approximation algorithm with constant performance ratio has been found.

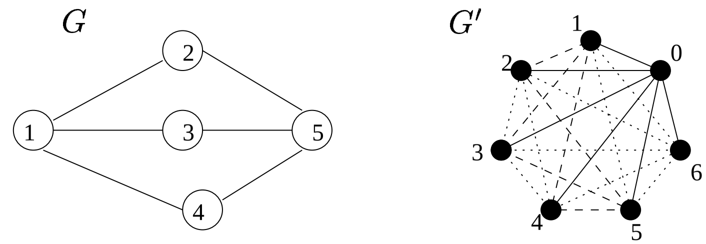
Main Result

Weighted Hamiltonian path completion problem

- Cannot be approximated within any constant ratio.
- NP-hard when the given edge set constitutes a tree and edge weights are restricted to be either 1 or 2.
- A 2-approximate algorithm for the above problem.
- No FPTAS for this problem.
- A 1.5-approximate algorithm on 1-stars.
- A 1.5-approximate algorithm on k-stars.

Theorem 1 For any $\alpha > 1$, if there exists an α-approximate algorithm for HPCT, then NP = P.

Proof. Reduction from the Hamiltonian path problem.



$$V' = V \cup \{v_0, v_{n+1}\}$$

$$E' = \{(v_i, v_j) \mid 0 \le i \le n+1, 0 \le j \le n+1 \text{ and } i \ne j\}$$

$$E_0 = \{(v_0, v_i) \mid 1 \le i \le n+1\}$$

For each $e \in E'$, $w(e) = \begin{cases} 1 & \text{if } e \in E, \\ \alpha|E|(n-1) & \text{otherwise.} \end{cases}$

If G has a Hamiltonian path, then G' has an augment with $n-1$ edges, and each of these edges has weight 1. (For $K_{1,n+1}$, $n-1$ edges are optimal.)

Suppose that we have an α-approximate algorithm $\mathcal{A}$. If the optimal augment for G' has cost $n-1$, then $\mathcal{A}$ cannot generate a solution containing any edge with weight $\alpha|E|(n-1)$. Otherwise,

$$\frac{\alpha|E|(n-1)}{n-1} = \alpha|E| > \alpha$$

Therefore, the solution generated by $\mathcal{A}$ will only contain edges with weight 1, and thus have total weight less than or equal to $|E|$.

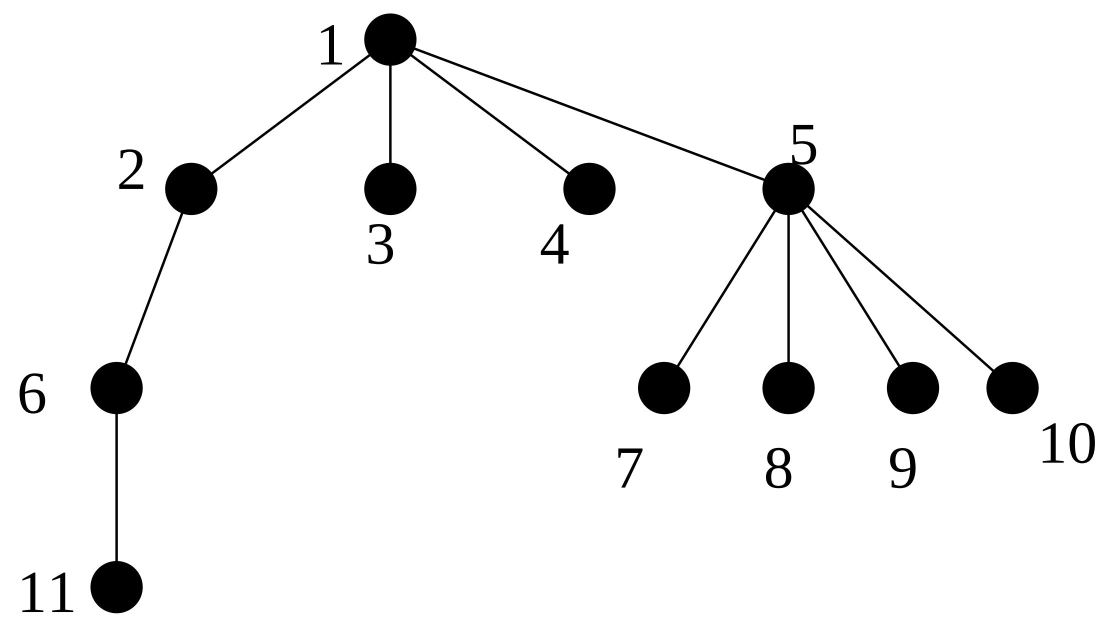
If $\mathcal{A}$ generates an augment with cost less than or equal to $|E|$, then obviously it does not contain any edge with cost $\alpha|E|(n-1)$. In other words, all these edges are contained in E. Since such a subset in E constitutes a Hamiltonian path, G has a Hamiltonian path.

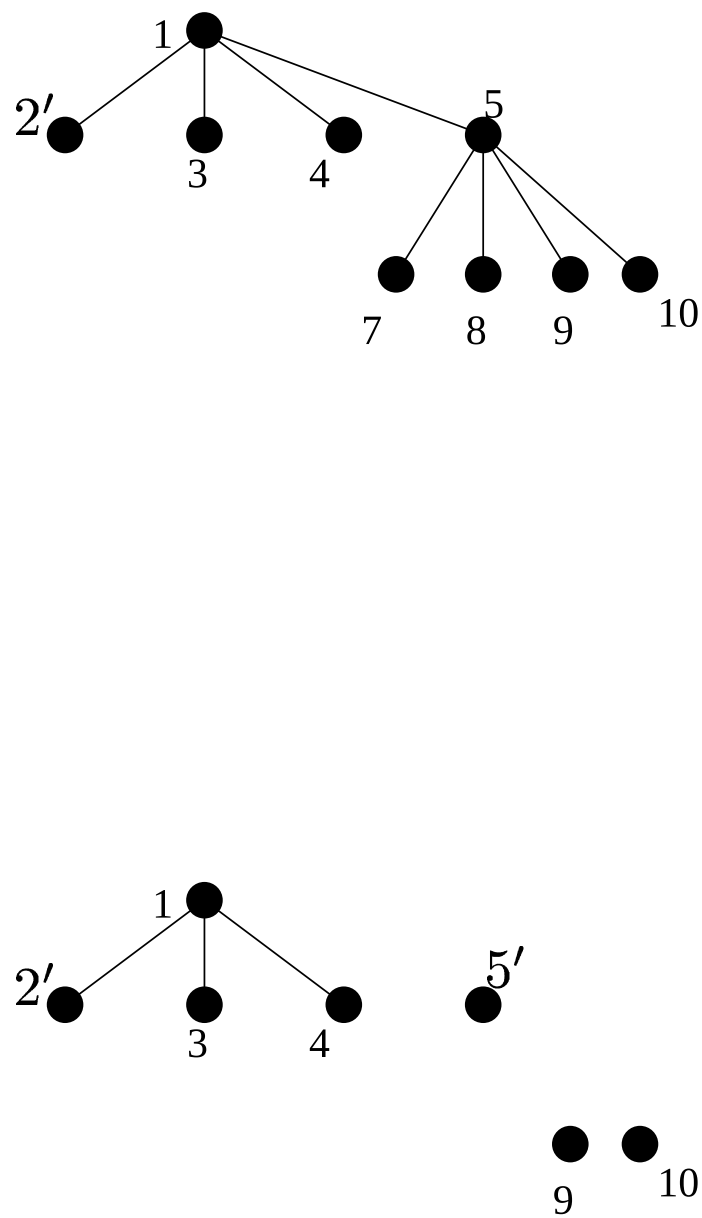
G has a Hamiltonian path if and only if $\mathcal{A}$ generates an augment with cost less than or equal to $|E|$ for G'.

Thus, if HPCT has a polynomial-time α-approximate algorithm, then NP=P.

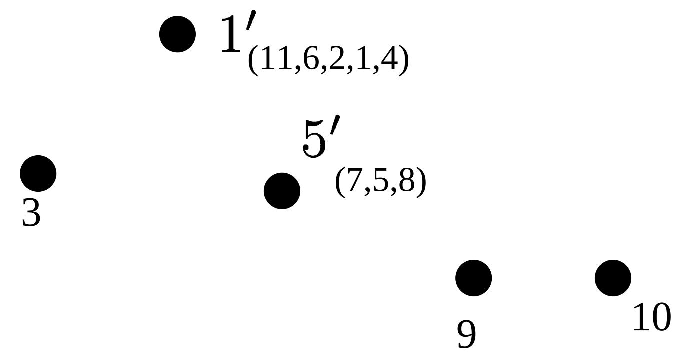
When the edge weights are restricted to be either 1 or 2, this problem remains to be NP-hard (Section 3).

A 2-Approximate Algorithm for (1,2)-HPCT





1
2′
3
4
5
7
8
9
10
1
2′
3
4
5′
9
10



- A: (11,6,2,1,4)
- B: (3)
- C: (7,5,8)
- D: (9)
- E: (10)

Add edges to concatenate these paths.

e.g. $E_2 = \{(4,3),(3,7),(8,9),(9,10)\}$

Algorithm 1

1. Choose any internal node to be the root of the tree.
2. If there is a leaf v, whose parent node u has only one child, then apply the Type 1 merging to merge u and v. Repeat this step until there is no such node.
3. Choose a deepest leaf v. If its parent node u has k children where $k \geq 2$, then apply the Type 2 merging and disconnect all edges incident to u.
4. If there is any edge left, then goto 2.
5. To obtain a Hamiltonian path, add edges to concatenate the paths corresponding to the remaining isolated vertices.

Let ζ denote the minimum number of edges to be inserted to make a given tree Hamiltonian. If G is unweighted then $|E_2| = \zeta$ [Goodman 74].

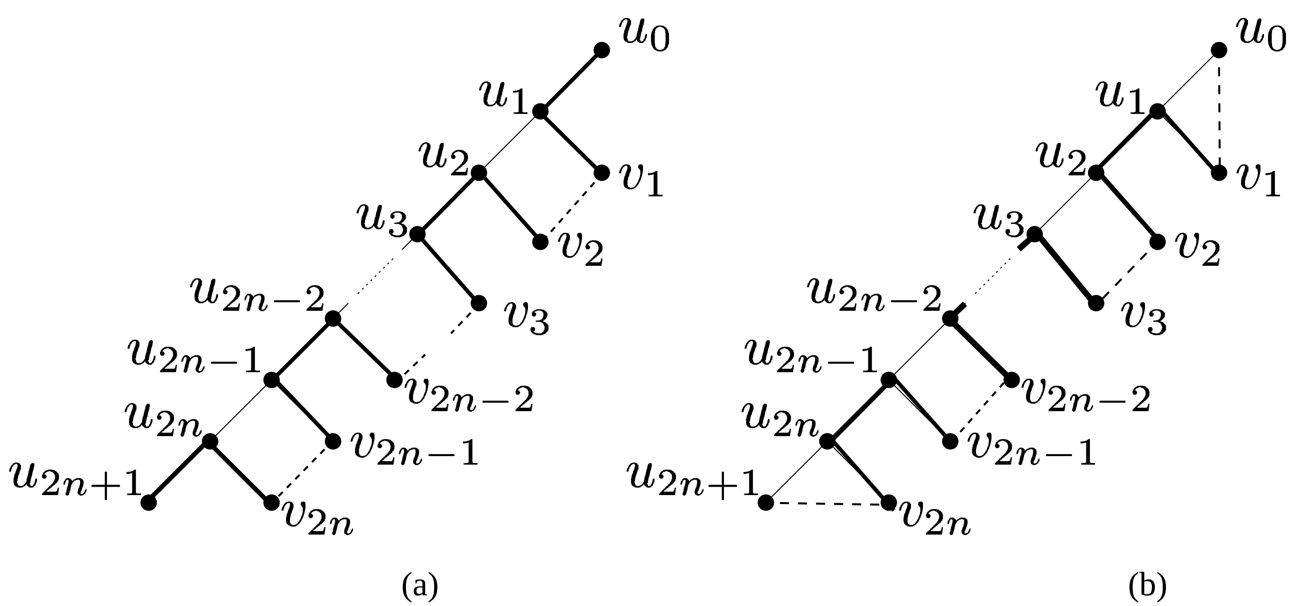
$$|E_2| = \zeta \leq |E_2^*|$$

$$w(E_2) \leq 2\zeta \leq 2 \times |E_2^*|$$

$$w(E_2^*) \geq 1 \times |E_2^*|$$

$$\Rightarrow \quad \frac{w_{apx}}{w_{opt}} = \frac{w(E_2)}{w(E_2^*)} \leq \frac{2|E_2^*|}{|E_2^*|} = 2$$

The ratio 2 is tight.



(a) An approximate augment with n edges, whose cost is $2n$. (b) An optimal augment with cost $n+1$.

Theorem 3 If (1,2)-HPCT has an FPTAS, then NP=P.

Proof. Suppose that (1,2)-HPCT has an FPTAS, i.e., $\forall \epsilon > 0$, there exists a $(1+\epsilon)$-approximation algorithm for the (1,2)-HPCT problem such that its time complexity is polynomial in the size of the input and $\frac{1}{\epsilon}$.

Then for $G = (V, E)$, choose $\epsilon = \frac{1}{2|E|}$.

$$\frac{w_{apx}}{w_{opt}} \leq 1 + \frac{1}{2|E|} \Rightarrow w_{apx} \leq w_{opt} + \frac{w_{opt}}{2|E|} \Rightarrow w_{apx} - w_{opt} \leq \frac{w_{opt}}{2|E|}.$$

Notice that the optimal augment E^* will never contain any edge in E_0, i.e., $|E^*| < |E|$. Since the weight on each edge is either 1 or 2,

$$\frac{w_{opt}}{2|E|} = \frac{w(E^*)}{2|E|} \leq \frac{2|E^*|}{2|E|} < 1.$$

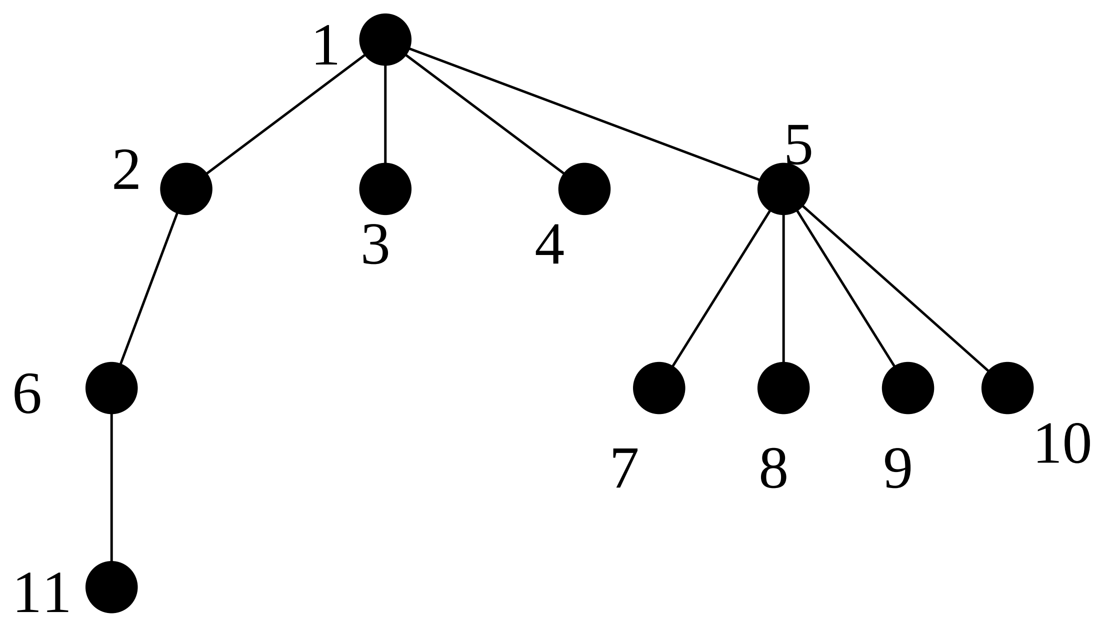
Therefore, $w_{apx} - w_{opt} < 1$

$$w_{apx} - w_{opt} = 0$$

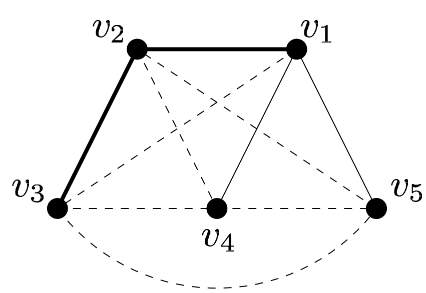
The approximation algorithm always generates an optimal solution in polynomial time. This contradicts with that (1,2)-HPCT is NP-hard as we proved in Section 3.

A 1.5-Approximate Algorithm for (1,2)-Hamiltonian Path Completion Problem on k-Stars

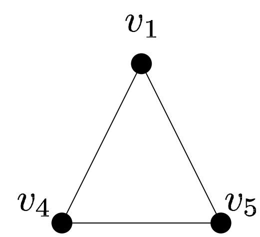
A 4-Star



Shrink



(a)

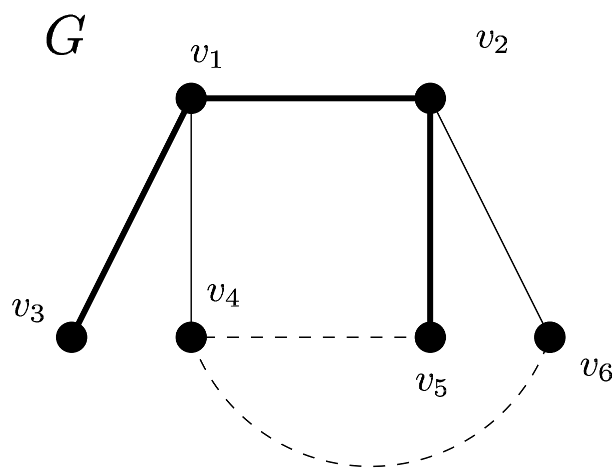


(b)

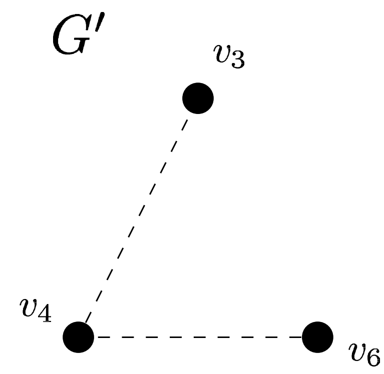
The path (v_1, v_2, v_3) is shrunk to a vertex v_1.

	v_1	v_2	v_3	v_4	v_5
v_1	∞	1	2	1	2
v_2		∞	1	1	1
v_3			∞	2	1
v_4				∞	2
v_5					∞

	v_1	v_4	v_5
v_1	∞	1	1
v_4		∞	2
v_5			∞

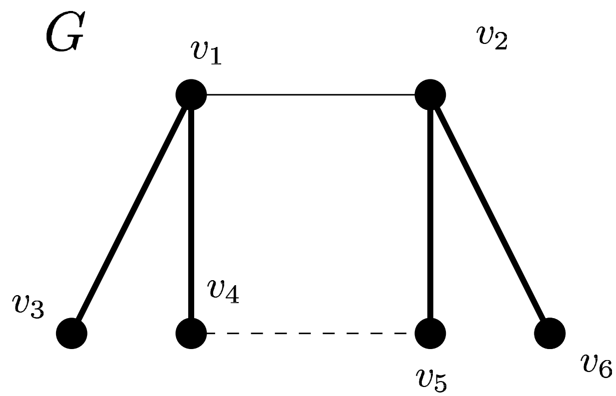


(a)

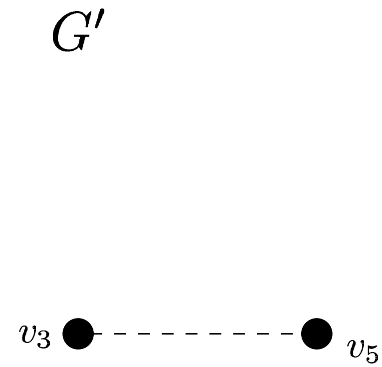


(b)

Suppose the optimal augment is $\{(v_4, v_5), (v_4, v_6)\}$ which yields a Hamiltonian path $(v_3, v_1, v_2, v_5, v_4, v_6)$. Then by shrinking the path (v_3, v_1, v_2, v_5) we obtain G', in which a Hamiltonian path has 2 edges.



(a)

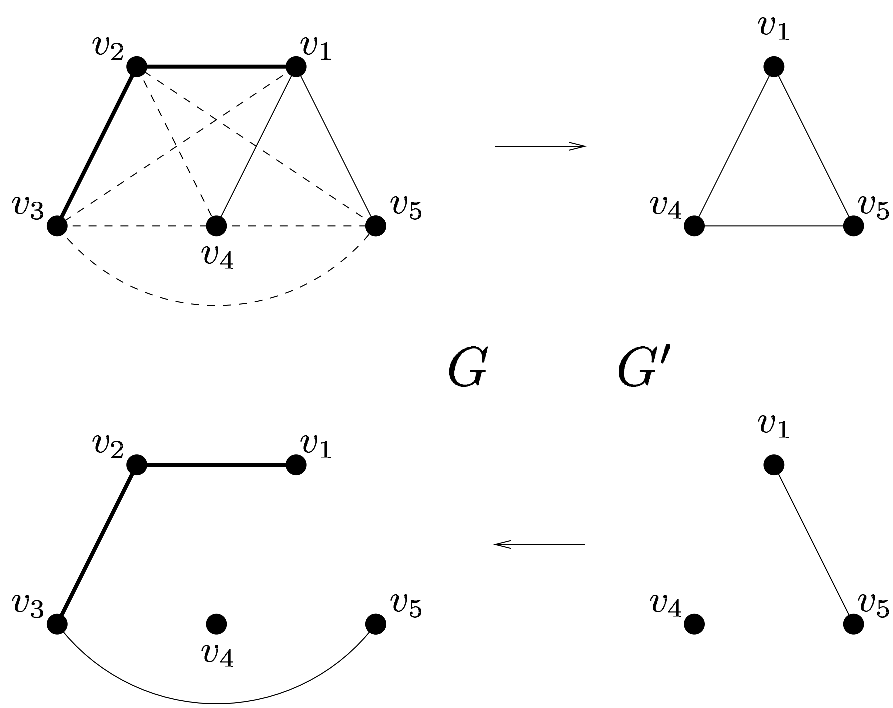


(b)

Suppose the optimal augment is $\{(v_4, v_5)\}$, which yields a Hamiltonian path $(v_3, v_1, v_4, v_5, v_2, v_6)$. Then by shrinking the two paths (v_3, v_1, v_4) and (v_5, v_2, v_6), we obtain G', in which a Hamiltonian path has 1 edge.

The number of edges in G' is the same as the cardinality of the augment in G.

1. On the shrunk graph G', we apply the minimum-weight maximal-matching algorithm [Lawler 76].
2. Map back to G.
3. Add edges to catenate these paths serially, say, (v_5, v_4).



Suppose the optimal augment contains n_1 edges with weight 1, n_2 edges with weight 2.

Then the minimum-weight maximal-matching in G' will contain at least $\frac{n_1}{2}$ edges with weight 1 (Lemma 2).

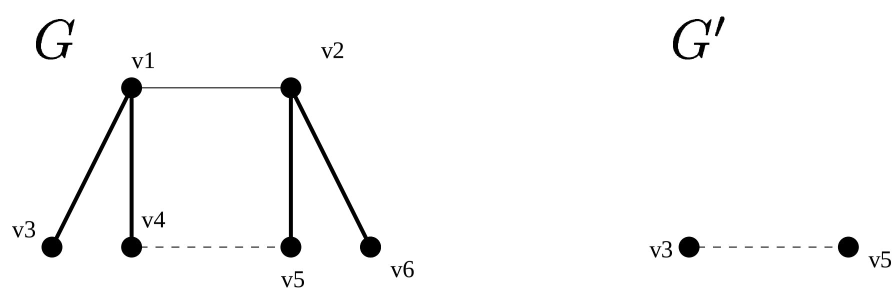
Even in the worst case that the remaining $(n_1 + n_2 - \frac{n_1}{2})$ edges chosen by our algorithm are all with weight 2, the total cost will be $\frac{3}{2}n_1 + 2n_2$.

$$\frac{\frac{3}{2}n_1 + 2n_2}{n_1 + 2n_2} \leq \frac{3}{2}$$

- This is the result on G', which is obtained by shrinking the path(s) contained in the optimal Hamiltonian path in G.
- But, given G, how do we know which paths are to be shrunk?
- Trying all possibilities and choose the minimum one among them will certainly work, but this lead to an exponential algorithm.

Lemma 3 If H is a Hamiltonian path in G, and E_0 constitutes a k-star, then H contains at most $2k$ edges in E_0.

$$C_0^n + C_1^n + C_2^n + \cdots + C_{2k}^n = O(n^{2k})$$



(a) (b)

Algorithm 2

Input: A weighted complete graph $G = (V, E)$, where the weight on each edge is either 1 or 2, and an edge set E_0 that constitutes a spanning tree on G, where the spanning tree has k internal nodes.

Output: An augment $E_2 \subseteq E$ such that $G' = (V, E_0 \cup E_2)$ has a Hamiltonian path.

Goal: Minimize the cost of E_2, i.e., $\sum_{e \in E_2} w(e)$.

Steps:

1. $W \leftarrow \infty, AUG \leftarrow \emptyset$.

2. **For** all subsets of E_0 with no more than $2k$ edges **do**

    If the subset has 3 or more edges incident to the same vertex **Then**

    /* do nothing */

    Else

    Suppose the subset consists of vertex-disjoint paths $P_1, P_2, \cdots, P_i$.

    shrink paths $P_1, P_2, \cdots, P_i$ to obtain $G_{\{P_1, P_2, \cdots, P_i\}}$.

    Find a minimum-weight maximal-matching MM on $G_{\{P_1, P_2, \cdots, P_i\}}$.

    Map these matching edges to paths in G.

Let MM be mapped to MM'.

Add edges E' to concatenate these paths serially to form a Hamiltonian path.

If the cost of $E' \cup MM'$ is smaller than W **Then**

$W \leftarrow w(E' \cup MM')$

$AUG \leftarrow E' \cup MM'$

End If

End If

Next

3. Report AUG as the solution and stop.

Time complexity: $O(n^{2k+3})$.

We obtain a polynomial-time 1.5-approximate algorithm for the Hamiltonian path completion problem on k-stars.

Conclusion

Weighted Hamiltonian path completion problem

- Cannot be approximated within any constant ratio.
- NP-hard when the given edge set constitutes a tree and edge weights are restricted to be either 1 or 2.
- A 2-approximate algorithm for the above problem.
- No FPTAS for this problem.
- A 1.5-approximate algorithm on 1-stars.
- A 1.5-approximate algorithm on k-stars.

Future Research

(1,2)-Hamiltonian Path Completion Problem

- For k-stars, 1.5-approximate - optimal or lower ratio?
- For trees, 2-approximate - optimal or lower ratio?
- For general graphs with weights 1 or 2
- For (a, b)-Hamiltonian path completion problem, what ratio can we obtain?