Problem 6

b09902004 資工二 郭懷元

> Refs:
>
> https://en.wikipedia.org/wiki/Christofides_algorithm

1

A graph $G(V, E)$ contains an Eulerian cycle if and only if $deg(v)$ is even $\forall\ v \in V$.

2.

Consider an undirected graph G with 0 edge at start. Because all vertices has degree 0 (even), $|V'| = 0$.

For each edge added, only degrees of the two vertices on this edge change, and there are three cases:

- Case 1: Both vertices were with even degree.

  Since they both change from even to odd, $|V'|$ is increased by 2.

- Case 2: Both vertices were with odd degree.

  Since they both change from odd to even, $|V'|$ is decreased by 2.

- Case 3: One vertex has odd degree and the other has even.

  Since one will change from odd to even, and the other will change from even to odd, $|V'|$ remains the same.

In all three cases, the parity of $|V'|$ always remains the same, which is even. Since a tree is an undirected graph, $|V'|$ is also even.

3.

Suppose $cost(M) > OPT/2$.

Let C be a cycle that yields OPT, and C contains edges $c_1, c_2, \cdots, c_{|V'|}$.

Both $C_{\text{odd}} = \{c_{2k-1} | 0 < k \le \frac{|V'|}{2}\}$ and $C_{\text{even}} = \{c_{2k} | 0 < k \le \frac{|V'|}{2}\}$ are perfect matchings. And because $cost(C_{\text{odd}}) + cost(C_{\text{even}}) = OPT$, $\min\{C_{\text{odd}}, C_{\text{even}}\} \le OPT/2 < cost(M)$.

By choosing the one with smaller cost, we have a cost lower than $cost(M)$. M doesn't have the minimum cost, which contradicts with the fact that M is optimal.

Therefore $cost(M)$ must be less than or equal to $OPT/2$.

4.

Algorithm

1. Find a minimum spanning tree T of G.
2. Let O be the set of vertices with odd degree in T. By subproblem 2, $|O|$ is even.

   Run $\text{Oracle}(O, E)$ to find a minimum perfect matching M.
3. Construct graph $H = T \cup M$.
4. Find a Eulerian cycle in H.
   1. Start from an arbitrary vertex v, and just follow any unvisited edges and keep going until we select an edge back to v. Because every vertices have even degree, every time we visit a vertex there is an edge out.
   2. If we select an edge back to v, choose other available edges if possible. If there are no available edges left, go back to v and complete the cycle.
5. Use this cycle to construct tour P by skipping repeated vertices.

$\frac{3}{2}$ - approximation

Let the optimal tour be P^*. $OPT = cost(P^*) \ge cost(T)$.

By triangle inequality, $cost(P) \le cost(H)$ $$ \begin{aligned} cost(H) &= cost(T \cup M) \ &\le cost(T) + cost(M) \ &\le OPT + \frac{OPT}{2} = \frac{3}{2} OPT \ \end{aligned} $$

Polynomial time complexity

1. Prim's algorithm can find a MST in polynomial time.
2. Orcale runs in polynomial time.

3. Constructing H can be done naively in $O(|V| + O|E|)$ time.
4. Finding a Eulerian cycle takes $O(|E|)$ time.
5. Constructing P takes $O(|V|)$ time.
6. Algorithm in total takes polynomial time.