CS F364
Design & Analysis of Algorithms

Algorithm Design - Approximation

Approximation Algorithms

- Non-approximability of TSP
- Relative Approximation Algorithm for Metric TSP

Relative Approximation

- On Relative Approximation:
  - Is it possible to find r-approximation algorithms for all NP-complete problems?
    - *Note that **r** must be a constant.*
  - For a given NP-complete problem, what is the best approximation ration obtainable?

Non-Approximability of TSP

- Theorem:
  - There exists no polynomial time r-approximation algorithm for TSP for any constant integer r >1, unless $\mathbb{P} = \mathbb{NP}$
- Proof (By contradiction):
  - Reduce HAM (i.e the Hamiltonian Cycle problem) to TSP such that
    - If there is a polynomial time r-approximation algorithm A for TSP there exists a polynomial time algorithm for HAM.
    - *See next slides* for the reduction and Lemma 3.
  - But HAM is an $\mathbb{NP}$-complete problem
    - i.e. there is no polynomial time algorithm for HAM unless $\mathbb{P} = \mathbb{NP}$.

NON-APPROXIMABILITY OF TSP [2]

- Reduction from HAM to (approximation of) TSP:
  - Given an instance of HAM, say a graph G = (V,E)
    - Construct a new graph H = (V, V x V, w) such that
      - w(e) = 1 if e is in E
      - w(e) = 1 + r*|V| otherwise, where r is a +ve integer > 1
- Lemma 1:
  - G has a Hamiltonian cycle iff H has a tour of weight |V|
  - Proof: (*Trivial*)
- Lemma 2:
  - Any sub-optimal tour T for H will have at least one edge with weight **1+r*|V|** i.e. m(H, T) > r * m*(H)

Non-Approximability of TSP [3]

- Lemma 3:
  - If there is a polynomial time r-approximation algorithm A for TSP then there exists a polynomial time algorithm for HAM.
- Proof:
  1. Assume A is a polynomial time r-approximation algorithm for TSP: then m(H, A(H,r)) <= r * |V| if m*(H)=|V|
  2. But by the reduction (*see previous slide*) and Lemma 2,

     the only solution A can return is the optimal solution!
  3. Now define A'(G) :
     - construct H from G as defined in the reduction;
     - Let T=A(H, r); if (m(H,T) <= r * |V|) return 1; else return 0;
  - A' solves HAM in polynomial time.
    - By Lemma 1 (prev. slide) , (2) above, and (1) above.

Problem – Metric TSP

- Definition: Triangle Inequality on Graphs:
  - A weighted graph G = (V,E,w) is said to satisfy if the triangle inequality if for any three vertices v1, v2, and v3 in V
    - w((v1,v2)) + w((v2,v3)) >= w((v1,v3))
- Problem Definition: Metric TSP:
  - Given a weighted graph G that satisfies the triangle inequality, find a minimum weight tour for G.
- Metric TSP is $\mathbb{NP}$-complete:
  - TSP is $\mathbb{NP}$-complete even in the special case where all weights are the same.

Relative Approximation – Metric TSP

- Algorithm A2_MTSP(G)
  1. Construct a minimum spanning tree M of G
  2. Construct an Euler Walk T of M

     i.e. a walk where each edge is visited exactly once in each direction.
  3. Construct a short-circuited tour S from T such that

     if (v1, v2) and (v2, v3) are in T where v2 has already been visited replace these two edges by (v1, v3).
  4. return S
- Claim 1:
  - A2_MTSP(G) is a TSP tour of G
  - Proof:
    - Step 2 ensures that all vertices are visited.
    - Step 3 ensures that each vertex is visited exactly once.

Relative Approximation – Metric TSP

- Algorithm A2_MTSP(G)
  1. Construct a minimum spanning tree M of G
  2. Construct an Euler Walk T of M

     i.e. a walk where each edge is visited exactly once in each direction.
  3. Construct a short-circuited tour S from T such that

     if (v1, v2) and (v2, v3) are in T where v2 has already been visited replace these two edges by (v1, v3).
  4. return S after adding an edge from the last vertex to the first vertex.
- Claim 2:
  - A2_MTSP(G) runs in polynomial time
  - Proof:
    - Each of the steps 1, 2, and 3 run in polynomial time

Relative Approximation – Example – Metric TSP [3]

- Claim 3:
  - The cost of a minimum spanning tree M is a lower bound on the cost of a TSP tour M for a given graph.
  - Proof: *The (optimal) tour T' with one edge removed is a spanning tree. Therefore w(T') >= w(M)*
- Claim 3:
  - m(G,A2_MTSP(G)) <= 2 * m*(G)
  - Proof:
    1. For any H, let **w(H) = $\Sigma_{e \text{ in } H}$ w(e)**
    2. From step 2 of A2_MTSP(G) **w(T) = 2 * w(M)**
    3. From step 3 of A2_MTSP(G) **w(S) <= w(T)**
  - From steps 1 to 3 of the proof and Claim 3:
    - **w(S) <= 2 * w(T')** QED

Relative Approximation – Example – Metric TSP [4]

- Theorem:
  - A2_MTSP(G) is a polynomial time 2-approximation algorithm for Metric TSP.
  - Proof:
    - By claims 1,2, and 3 (see previous slides).
- Improvement:
  - The approximation factor can be improved:
    - observe that we are doubling all the edges in the spanning tree to ensure that all vertices are of even degree (so that an Euler Tour can be constructed).
    - but one can double only those edges incident on vertices with odd degree i.e. cost of the Euler tour will be lower than twice the cost of the spanning tree:
  - Derive the bound for this modified algorithm!