Approximation Algorithms by bounding the OPT

Instructor

Neelima Gupta

ngupta@cs.du.ac.in

Approximation Algorithms

- Approximate algorithms give approximate solution to NP-Complete problems which is close to optimum solution.

<u>Maximization problems</u>

E.g. Clique

Optimum clique > Approx. clique >= ½ *(Optimum clique)

<u>Minimization problems</u>

E.g. Vertex Cover

Minimum vc < Approx. vc < = 2*(Minimum vc)

Factor-2 algorithm

- Find a maximal matching in the graph and output the matched vertices.

Let S be this set of vertices.

Claim 1: S forms a vertex cover.

Proof: Suppose not. Then there exists an edge

e = (u,v) such that neither u nor v is in S. This implies that the matching could have been extended by this edge e and hence was not maximal --- a contradiction.

Claim 2: |S| <= 2 OPT

Proof: We'll prove this by giving some lower bound say LB for OPT and showing that |S| <= 2 LB - Standard Technique

Lower bounding the OPT

Claim: OPT >= size of any (maximal) matching

Proof: Let M be a (maximal) matching. For every e = (u,v) in M, any vertex cover must pick at least one of u and v. Hence size of any vertex cover >= |M|. Hence, in particular, OPT > = |M|

Clearly |S| = 2* |maximal matching|

Hence Claim2 follows.

Can the approximation guarantee be improved?

- Following Qs need to be addresses
  - Can the approximation guarantee be improved by a better analysis?
  - Can an approximation algorithm with a better guarantee be designed using the lower bounding scheme of maximal matching?
  - Is there some other lower bounding technique that can give an improved guarantee for vertex cover?

Tight Example

- What is the meaning of Q1?
- Can we get a solution S using the above algorithm such that |S| < 2* OPT (for every instance of the problem)? Say |S| = 3/2 * OPT?

- Answer to the Q is No. Here is an example of an instance on which the above algorithm will always give a solution whose cost = 2*OPT.

Complete Bipartite Graph: $K_{n,n}$: OPT = n, |S| = 2n.

Q2

- i.e. Can we design an algorithm that gives a vertex cover solution S such that |S| < 2* |maximal matching| (for every instance of the problem)? Say |S| <= 3/2* |maximal matching|?
- Ans: No. Here is an example of an instance where the size of any vertex cover is at least 2 * |maximal matching|.
- Example: K_n : Complete graph of size n, n odd.

|Size of maximal matching| is (n-1)/2 and OPT = n-1. Thus the size of any vertex cover >= OPT

= n - 1

= 2 * |maximal matching|.

Q3

- Still an Open Problem!!!

Metric Travelling Salesman Problem

Problem Statement

Given A complete graph G with non-negative edge costs that satisfy triangle inequality

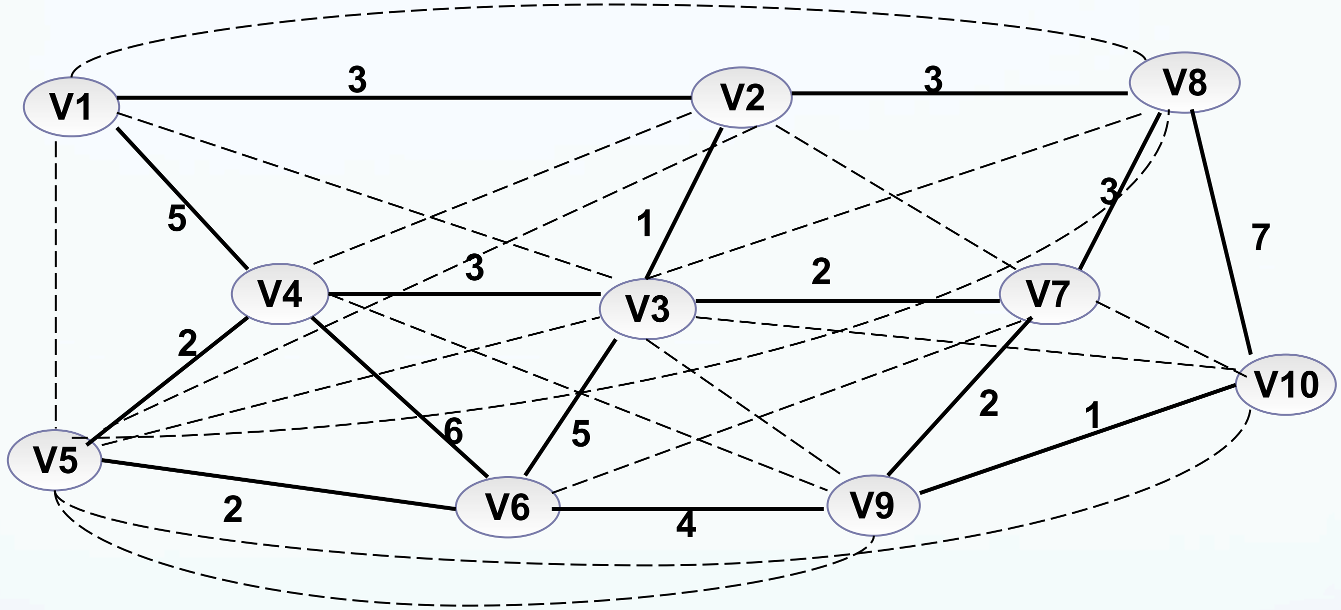
To Find A minimum cost cycle visiting every vertex exactly once.

Metric TSP - factor 2 approx. algorithm

1. Find an Minimum Spanning Tree (MST) T of G.
2. Double every edge of the MST to obtain an Eulerian graph.
3. Find a Eulerian tour, T', on G.
4. Output the tour that visits vertices of G in the order of their first appearance in T'. Call this tour C. (Short Cutting)

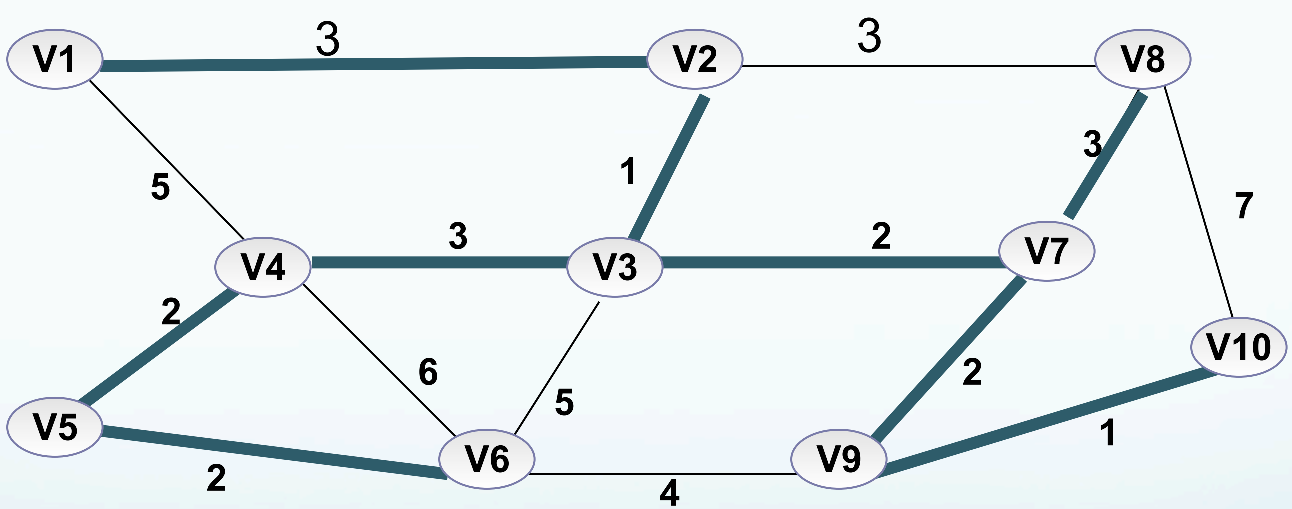
Example

Given a Complete Graph

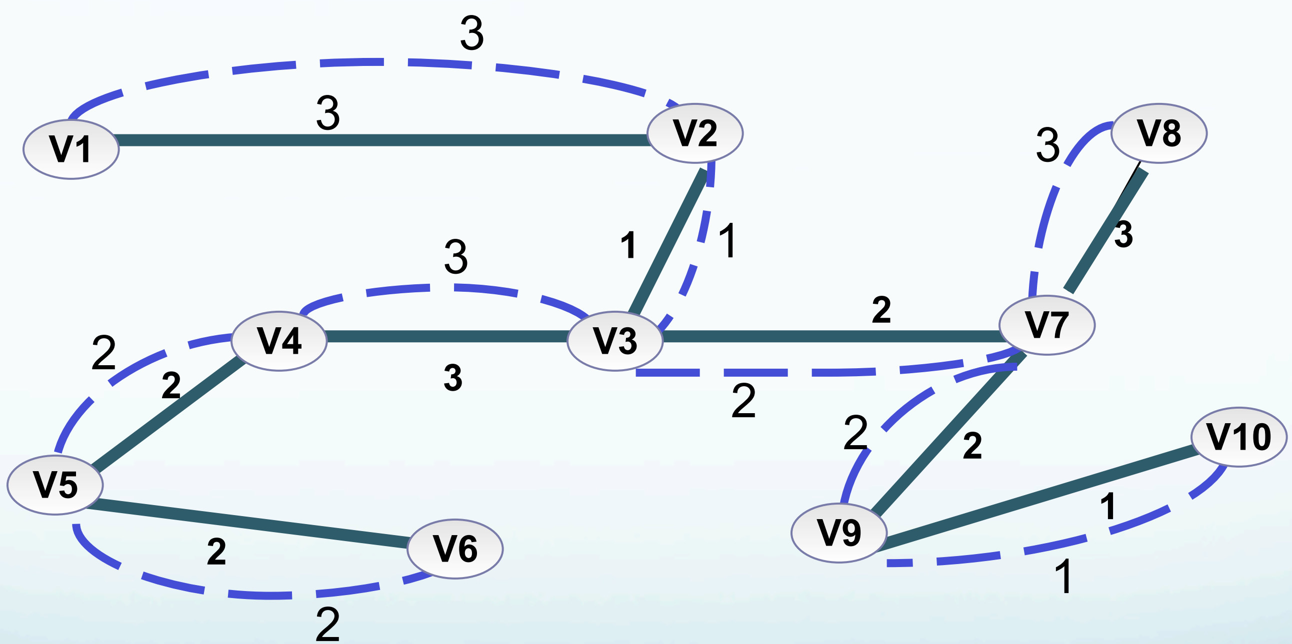


Edges shown dotted do not carry weight and are assumed to be shortest path between the pair of vertices(due to triangular inequality).

Step1: Compute Minimum Spanning Tree

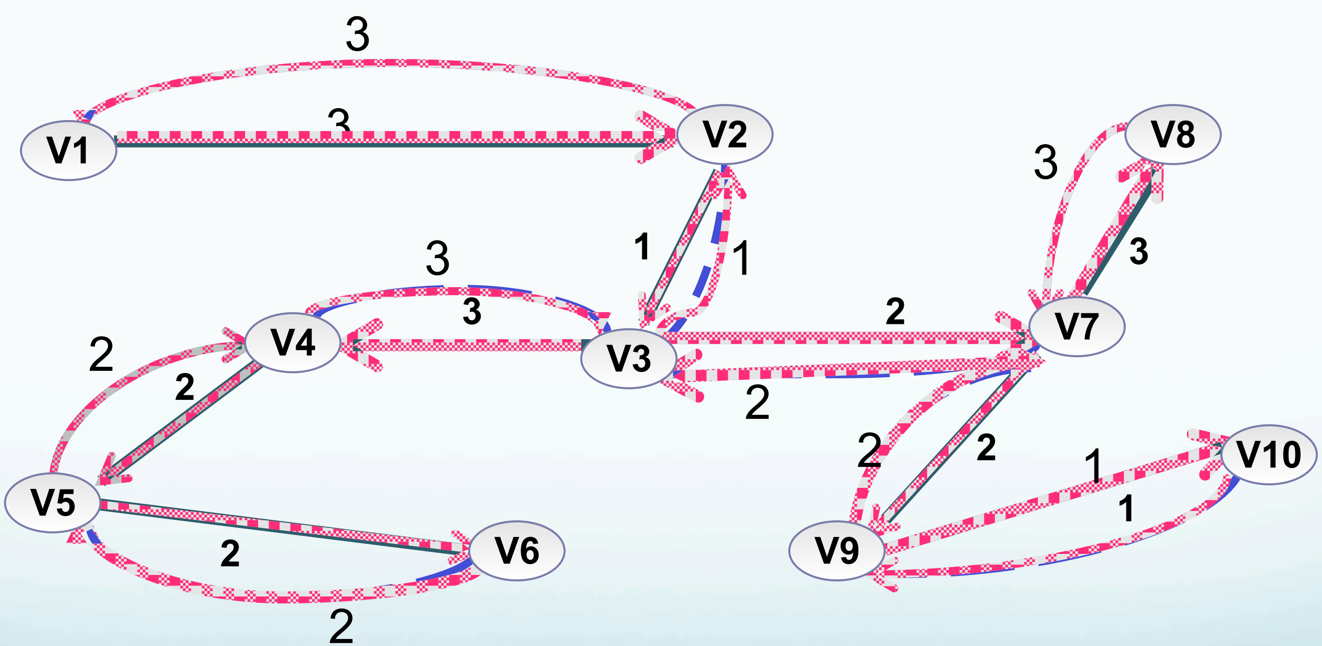


Step 2: Double each edge of MST



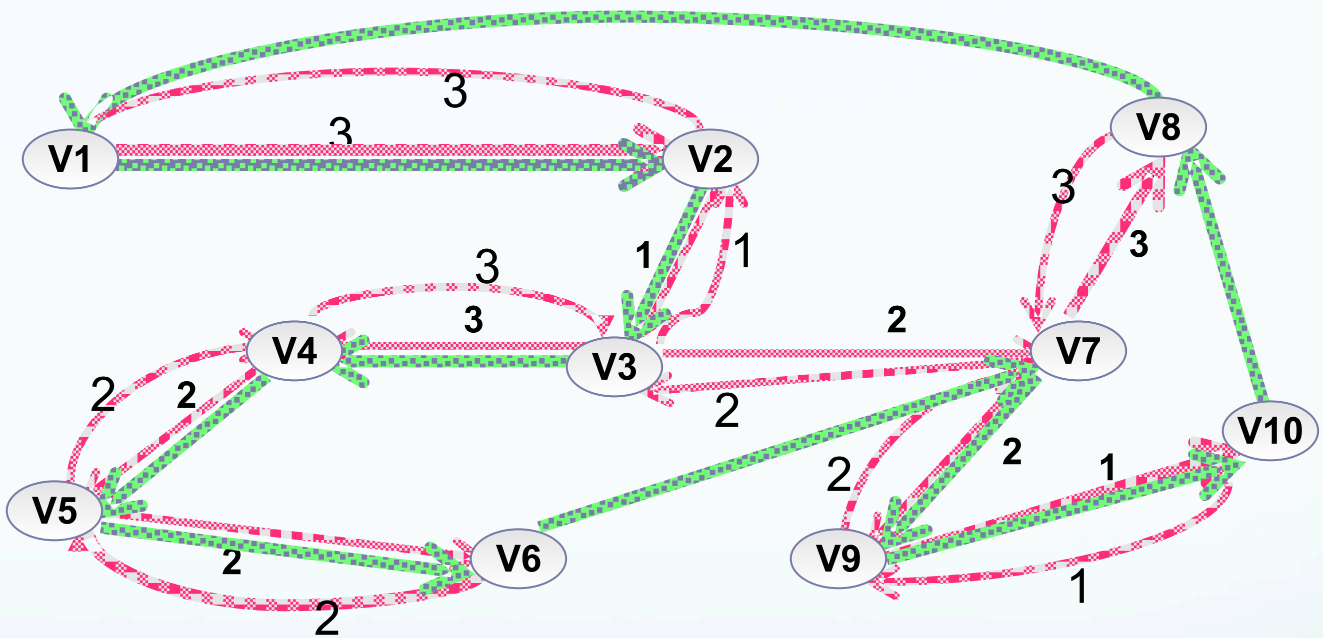
Step 3: Computing Eulerian Cycle

A cycle is one in which each edge visited exactly once



v1 → v2 → v3 → v4 →v5 → v6 → v5 → v4 → v3 →v7 →v9 →v10 →v9 →v7→v8→v7→v3→v2→v1

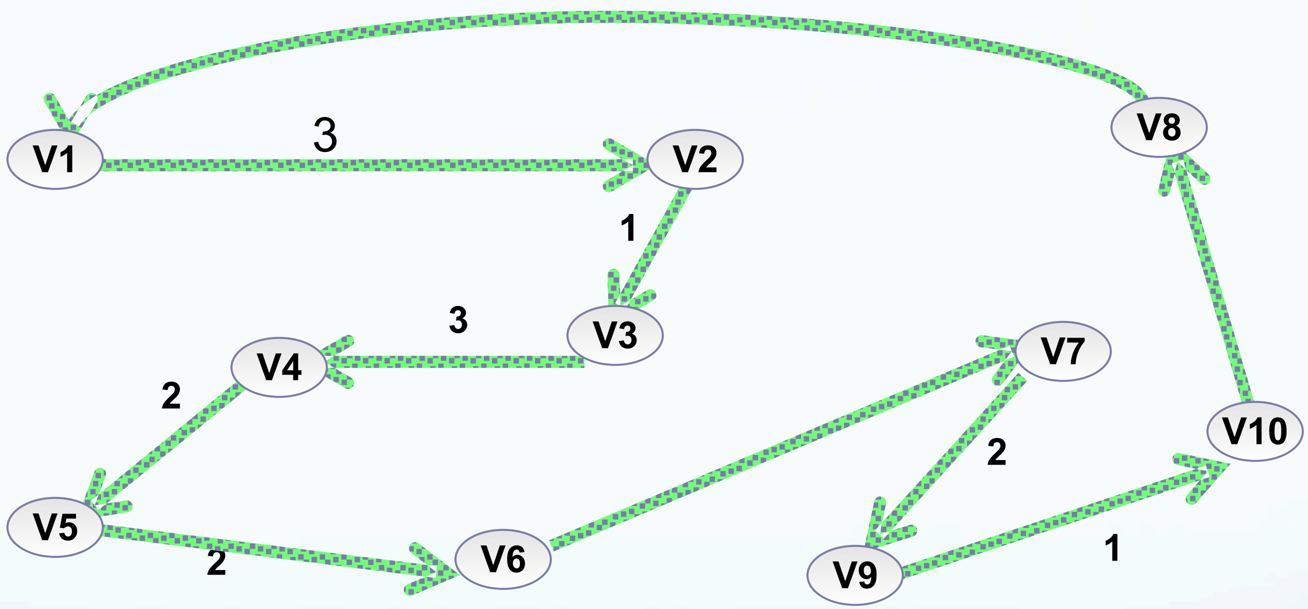
Step 4: Computing solution for TSP



v1 → v2 → v3 → v4 → v5 → v6 → v5 → v4 → v3 → v7 → v9 → v10 → v9

→ v7 → v8 → v7 → v3 → v2 → v1

Approximate solution for TSP



v1 → v2 → v3 → v4 →v5 → v6 →v7 →v9 →v10 →v8→v1

Metric TSP - factor 2 approx. algorithm

We now show that the proposed algorithm is indeed a factor 2 approximation algorithm for metric TSP

Observe that:

- Removing any edge from an optimal solution to TSP would give a spanning tree of the graph.
- So the cost of an MST in the graph can be used as lower bound for obtaining factor 2 for this algorithm

Metric TSP - factor 2 approx. algorithm

- Therefore, cost(T) <= OPT
- T' contains each edge of T twice, so cost(T') = 2*cost(T)
- Also, cost(C) <= cost(T') because of triangle inequality
- Hence cost(C) <= 2*OPT

FACTOR 3/2 APPROXIMATION ALGORITHM FOR TSP

Metric TSP – improving the factor to 3/2

Observations:

Consider why did we have to double the MST – to obtain an Euler tour.

Can we have an Euler tour with lower cost?

YES !

A graph has an Euler tour if and only if all its vertices have even degrees. We therefore need to be bothered about the vertices of odd degree only.

Metric TSP - improving the factor to 3/2

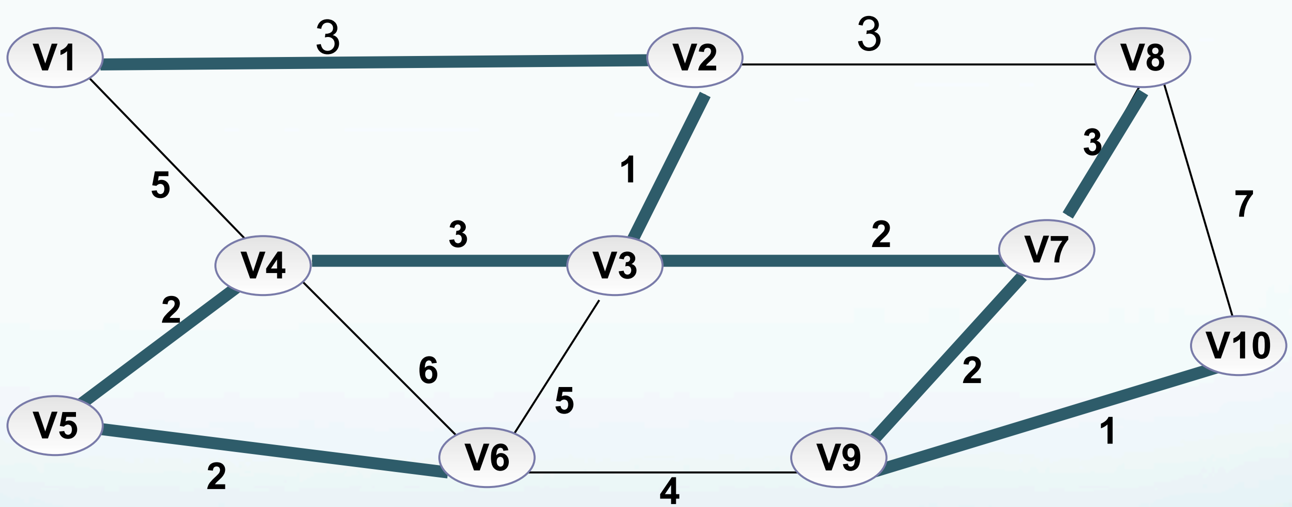
- Let V' be the set of vertices of odd degree
- Cardinality of V' must be even. WHY?

  Because the sum of degrees of all vertices in MST has to be even.
- Add to the MST, a minimum cost perfect matching on V' so that every vertex has an even degree.
- We also know that a polynomial time algorithm exists for finding the minimum cost perfect matching.

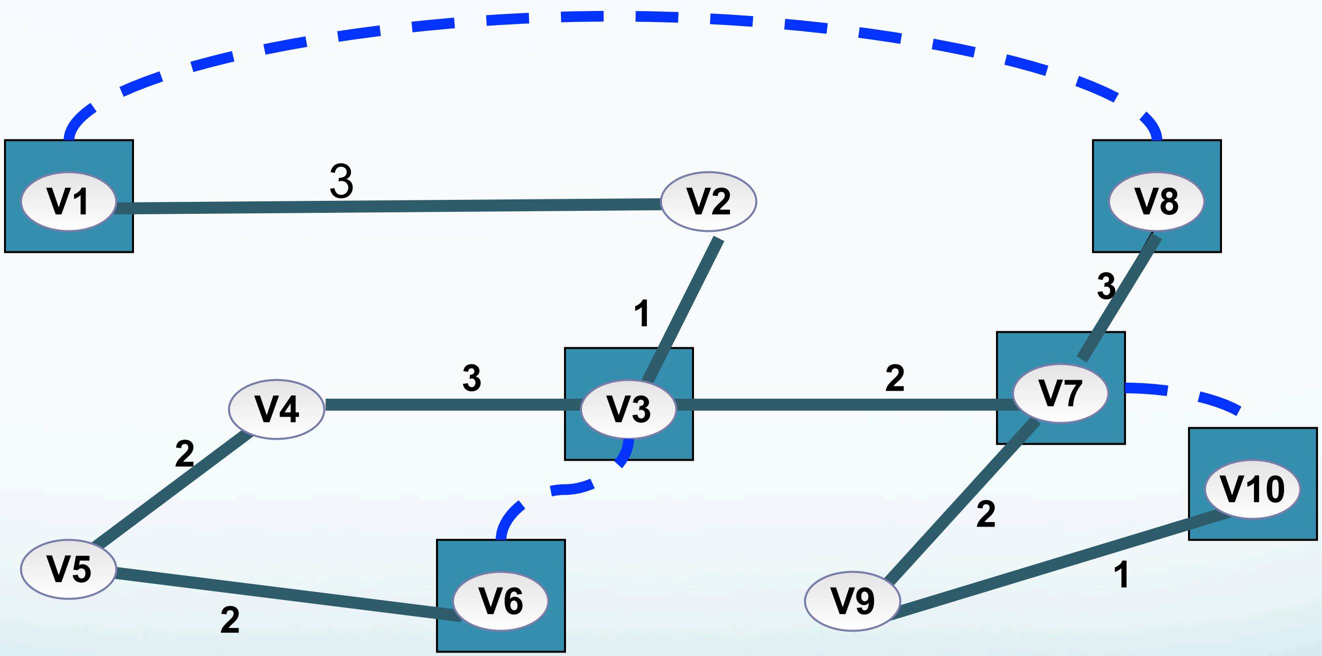
Metric TSP - factor 3/2 approx. algorithm

- Step 1: Find an MST, T, of G.
- Step 2: Compute a minimum cost perfect matching, M, on the odd degree vertices of T. Add M to T and obtain an Eulerian graph.
- Step 3: Find an Euler tour, T', of this graph.
- Step 4: Output the tour that visits vertices of G in order of their first appearance in T'. Call this tour C.

Step1: Compute Minimum Spanning Tree



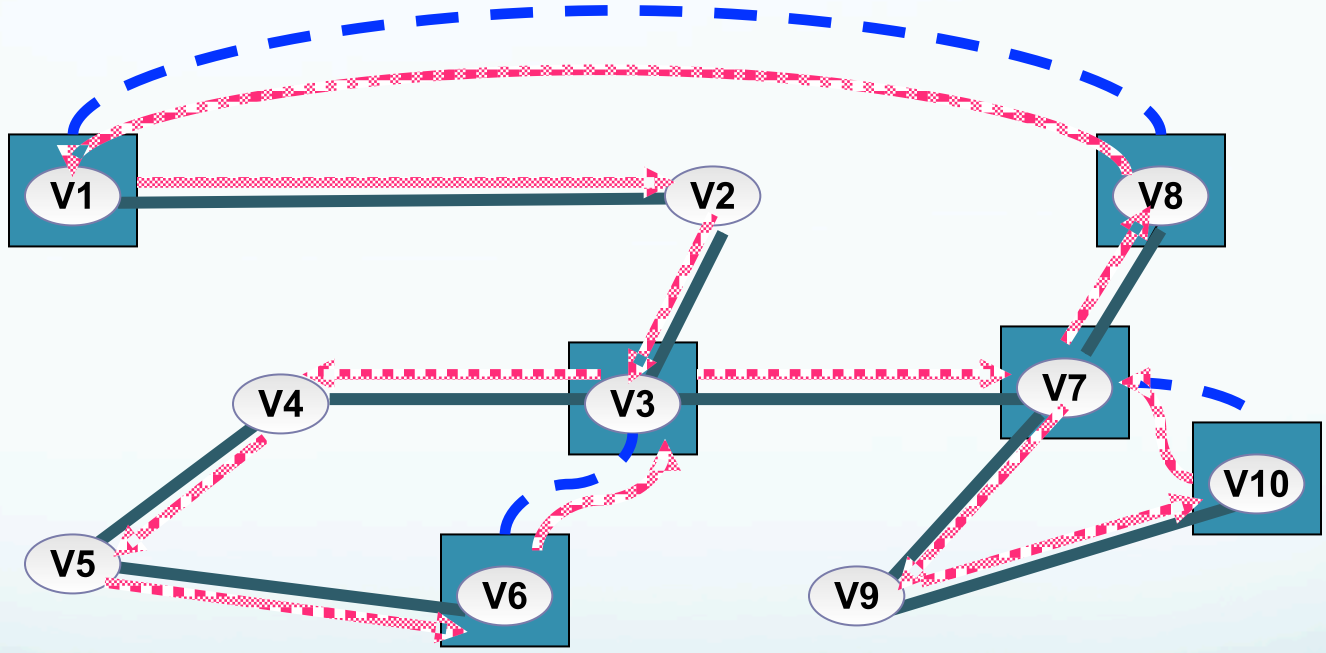
Step2: Compute Minimum Cost Perfect Matching



V1, V3, V6, V7, V8, V10 are odd degree vertices

Step 3: Computing Eulerian Cycle

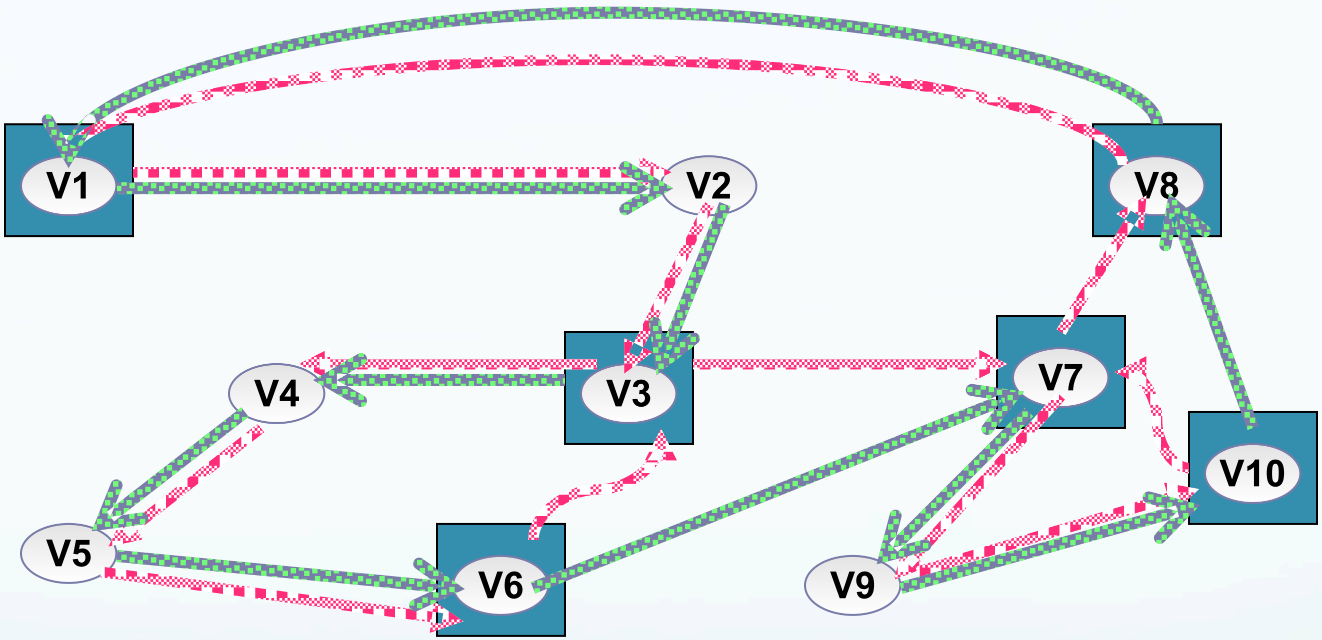
A cycle is one in which each edge visited exactly once



Eulerian Cycle :

V1 → V2 → V3 → V4→ V5 → V6 → V3 → V7 → V9 → V10 → V7 → V8 → V1

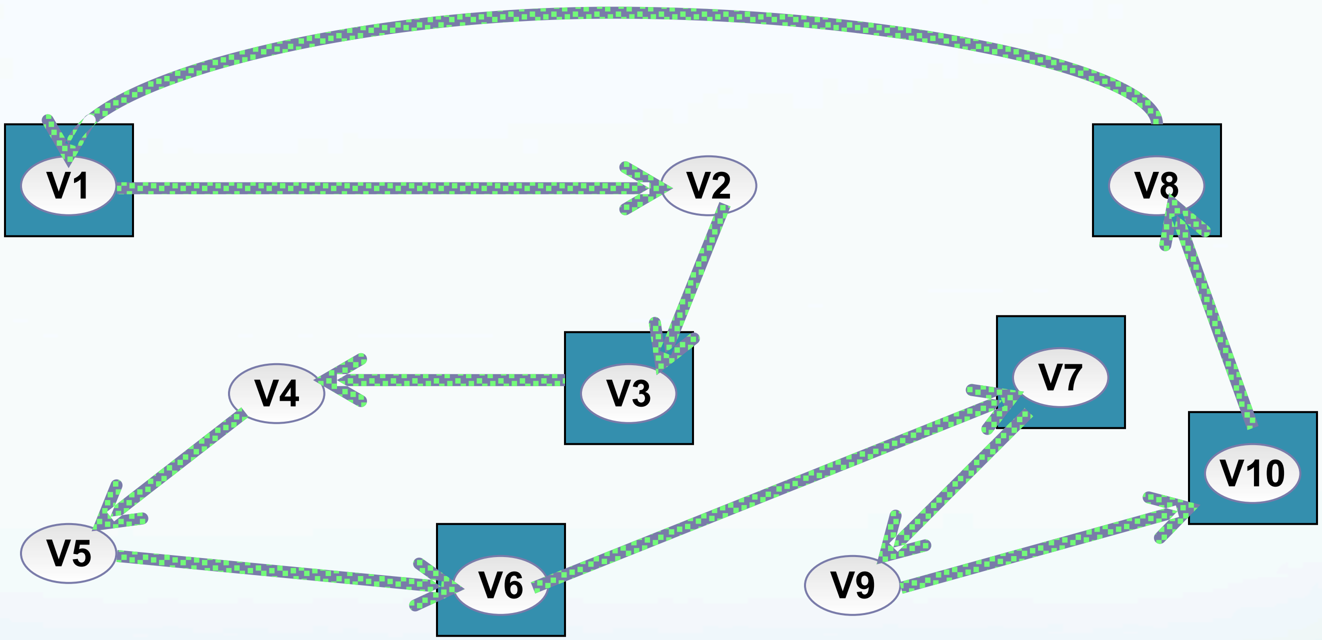
Step 4: Computing solution for TSP



Solution for TSP :

V1 → V2 → V3 → V4→ V5 → V6 → V7 → V9 → V10 → V8 → V1

Approximate solution for TSP



Solution for TSP :

V1 → V2 → V3 → V4→ V5 → V6 → V7 → V9 → V10 → V8 → V1

Metric TSP - factor 3/2 approx. algorithm

In order to show that the proposed algorithm is a factor 3/2 approximation algorithm for metric TSP, we first need to understand the following:

Given a subset V' of V with even number of elements, and a minimum cost perfect matching M on V', cost(M) <= OPT/2

Let us try to prove the above result !

Metric TSP - factor 3/2 approx. algorithm

- Consider an optimal TSP tour of G, say t.
- Let t' be the tour on V' obtained by shortcutting t.
- Clearly, cost(t')<=cost(t) because of triangle inequality.
- Now t' is the union of two perfect matchings on V' each consisting of alternate edges of t. Therefore, the cheaper of these matchings has cost <= cost(t')/2<=OPT/2.
- Hence the optimal matching also has cost at most OPT/2.

Metric TSP - factor 3/2 approx. algorithm

In view of this result, let us now see if the proposed algorithm ensures an approximation guarantee of 3/2 for metric TSP Problem

Cost of the Euler tour,

$$\cos t(T') \leq \cos t(T) + \cos t(M) \leq OPT + 1/2OPT = 3/2OPT$$

Using triangle inequality, cost(C)<=cost(T').

Hence Proved!

Acknowledgements

- Swati Singhal
- Varun Mendiratta
- Sumedha Upadhyaya
- Prachi Nagpal

Any Questions….

Thank You!