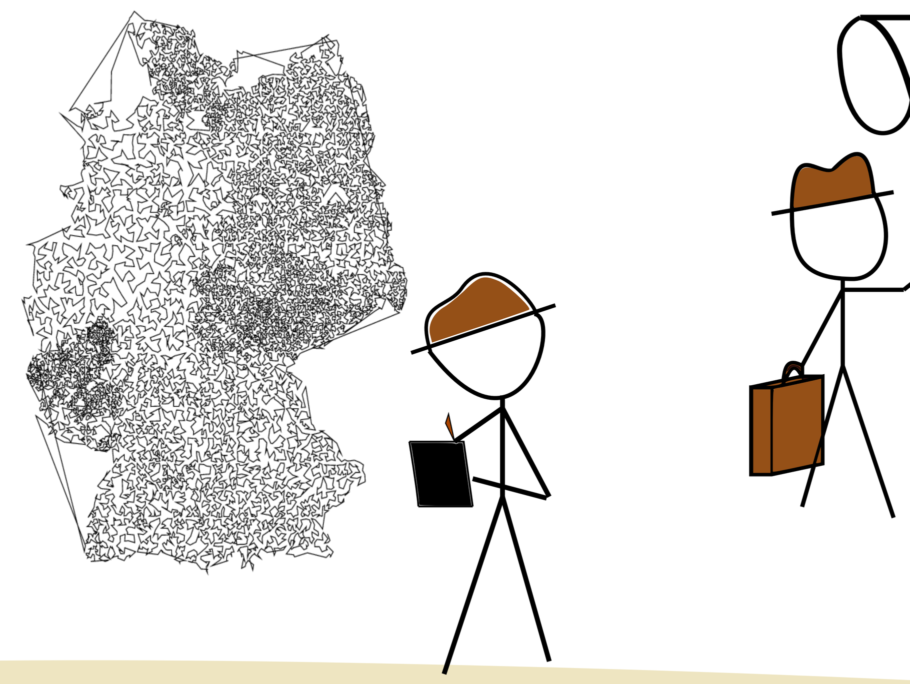
the traveling salesman problem

towards a 4/3 approximation for the metric case

once upon a time

The traveling salesman problem was first talked about in the 1800s by the Irish mathematician W. R. Hamilton and the British mathematician Thomas Kirkman

Given a collection of cities and the cost of travel between them, the TSP finds the shortest tour that visits each city exactly once and returns to the starting point



Applications
- Manufacture of microchips
- Scan Chain Optimization
- Visiting a layout of cities, payphones, customers
- Airport tours using Concorde
- Design of sonet rings at Telcordia
- Genome sequencing

Variants

Metric TSP: 3/2 approximation given by Christofides, 1976

Assymetric TSP: log(n)/loglog(n) by Goemans, Madry, Gharan Saberi, 2010

1,2 - TSP: 8/7 approximation by Berman and Karpinski, 2006.

Euclidean TSP - PTAS by S. Arora, 1998

Algorithms

3/2 - 5/389 approximation algorithm for cubic 3-edge-connected graphs - Gamarnik, Lewenstein, Sviridenko (OR Letters 33, 2005).

2005

(3/2 - e, e>0) randomised approximation algorithm for graphic TSP - Gharan, Saberi, Singh (FOCS, 2011).

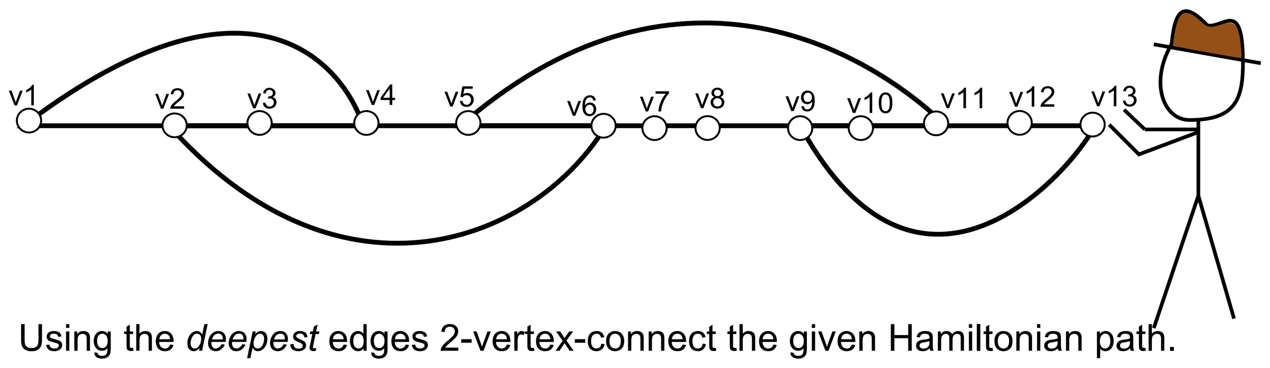
2011

4/3 approximation for cubic graphs - Boyd, Sitters, Ster, Stougie (IPCO, 2011)

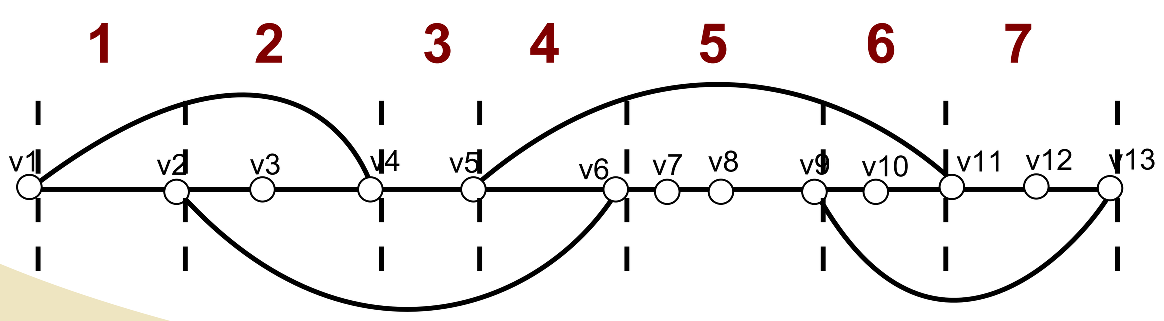
..

path algorithm

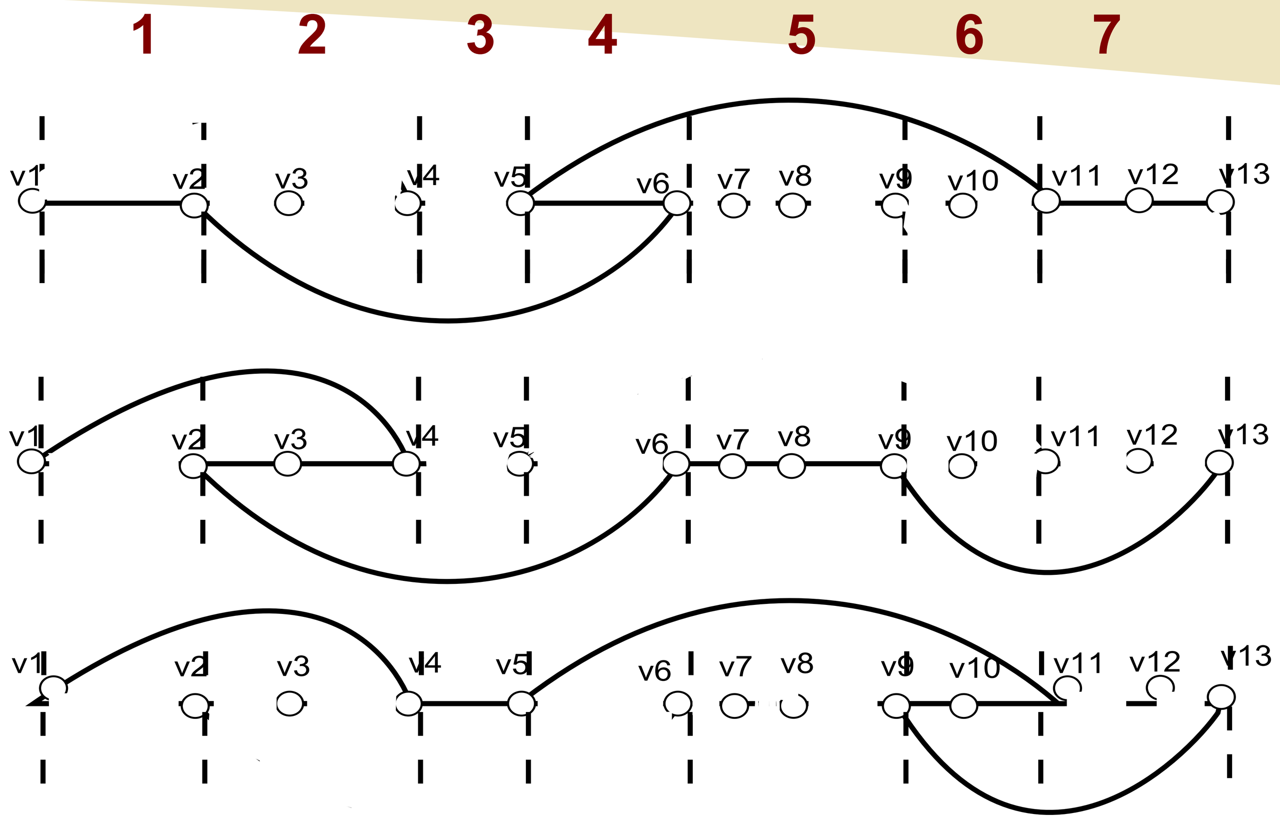
Given a Hamiltonian path in a 2-vertex connected graph on n vertices, find a walk that visits every vertex using at most 4n/3 edges.



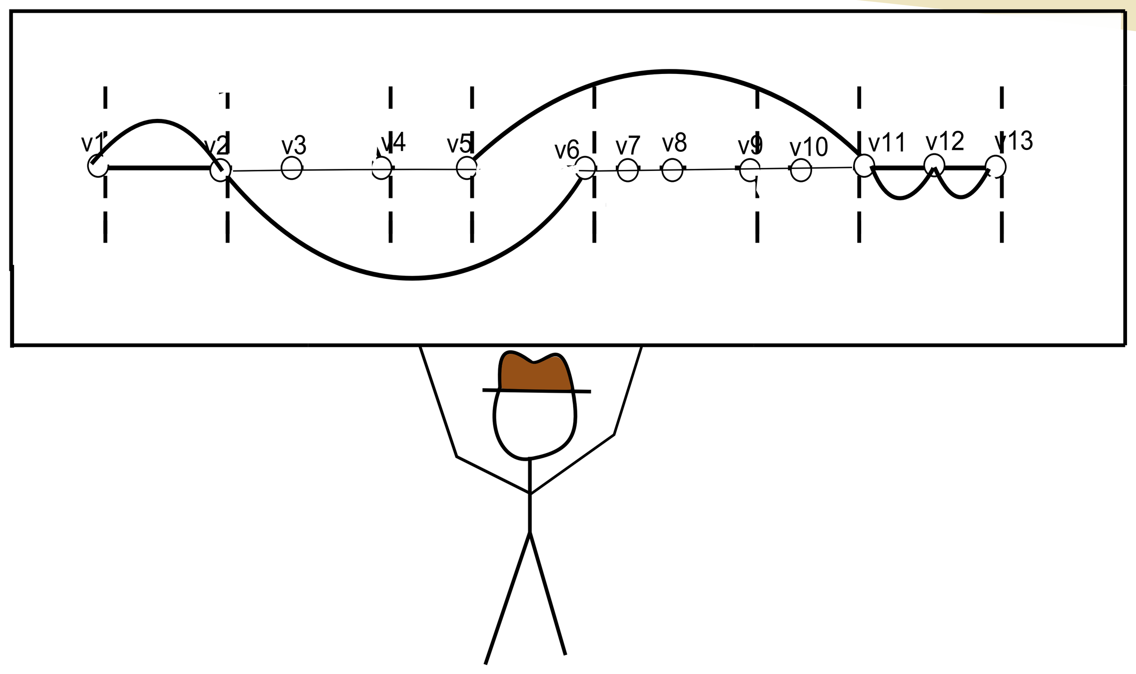
Using the *deepest* edges 2-vertex-connect the given Hamiltonian path.



Divide the vertices into intervals



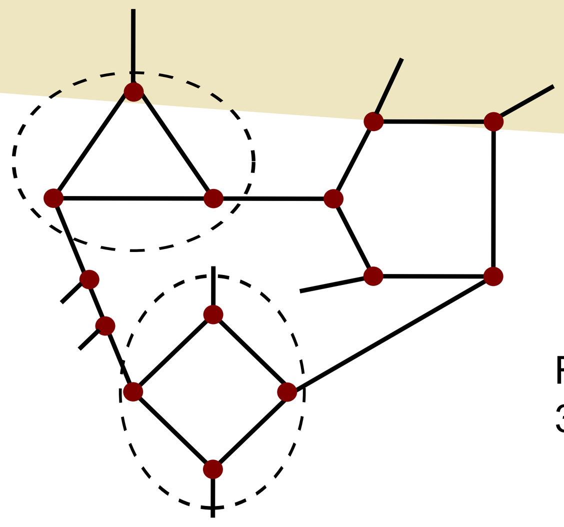
Cost of a pattern = 2*number of deep-edges + vertices in selected intervals
Total cost of the 3 patterns = number of vertices
Select the one with least cost (first one here)



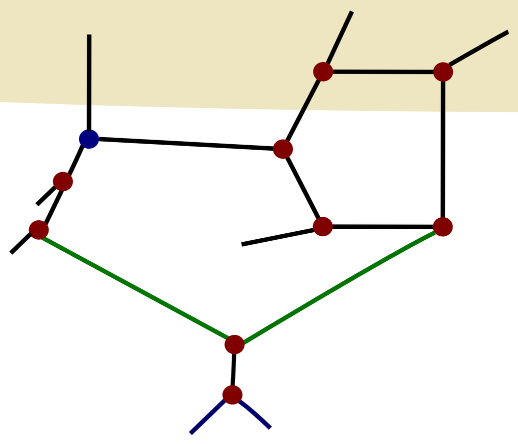
Use it to construct the final walk

cubic algorithm

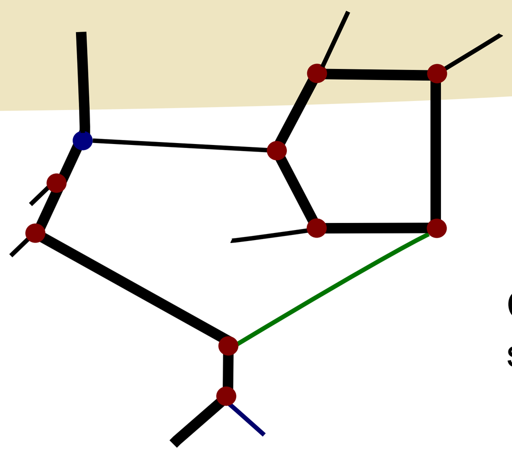
A 4/3 approximation algorithm for cubic 3-edge-connected graphs.



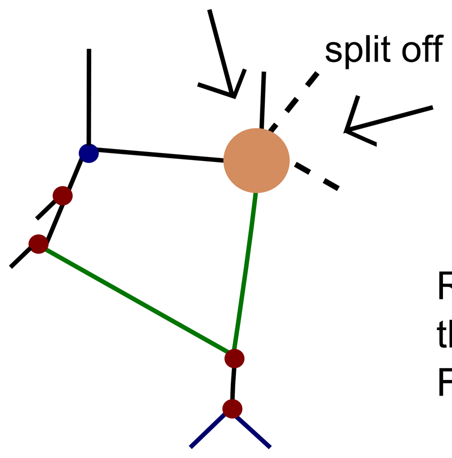
Remove *essential* 3-cuts and 4 cycles



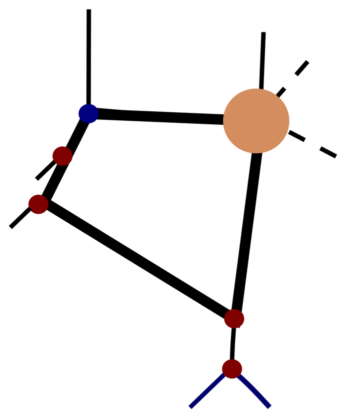
Find a 2-factor



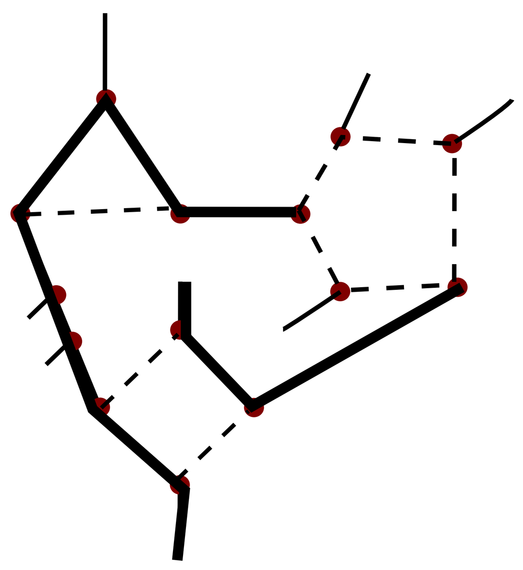
Compress any 5-cycles and split the new vertex formed



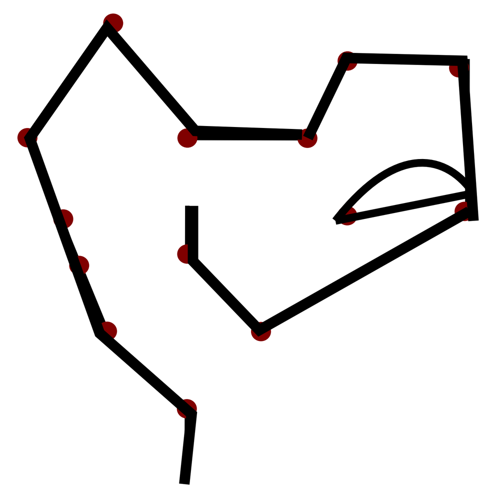
Repeat from A on the new graph.
Find a 2-factor again.



Expand the 5,4 and 3-cycles
Expand the 3-cycles



Convert to an Eulerian subgraph!



The algorithm:

1. Jackson, Yoshimoto (2009) : Given G is a 3-edge-connected graph on n vertices, G has a spanning even sub-graph in which each component has atleast min(n,5) vertices.
2. Using **1**, one can find a 2-factor with no 3 and 4 cycles in cubic 3-edge-connected graphs. We shrink each 5-cycle in the 2-factor into super vertices, split off the 5-degree super-vertex to obtain a cubic 3-edge-connected graph. This is repeated till a 2-factor without 5 cycles (that do not contain a super-vertex) is found. We then show how to convert each component of this 2-factor into an even sub-graph of 2G. To connect the different components, a spanning tree on the components can be calculated and its edges be doubled. The above illustration depicts some key ideas.

Joint work with Nishita Agarwal and Naveen Garg at IIT Delhi