CS 163 & CS 265: Graph Algorithms

Week 5: More paths

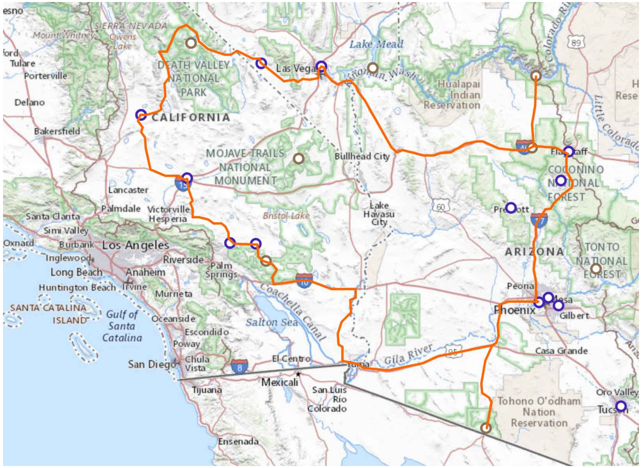
Lecture 5a: Approximate traveling salesperson tours

David Eppstein
University of California, Irvine

Winter Quarter, 2025

Some problems of arranging nearby things together

Plan a road trip through multiple parks



https://www.us-parks.com/road-trip.html

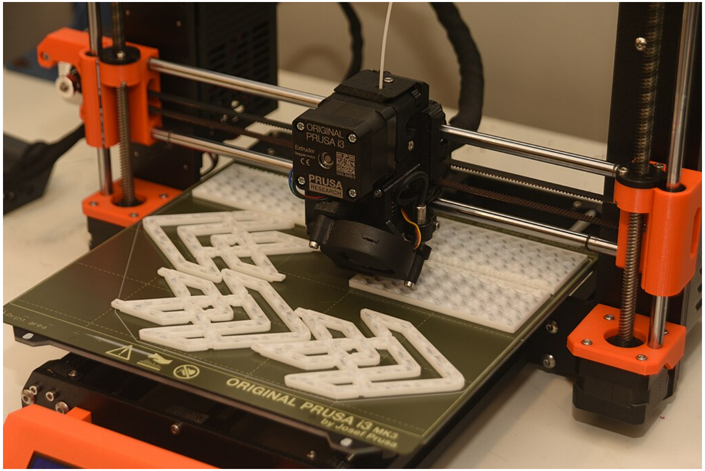
Some problems of arranging nearby things together

Schedule package deliveries

https://commons.wikimedia.org/wiki/File:Amazon_Prime_Delivery_Trucks_(44155424640).jpg

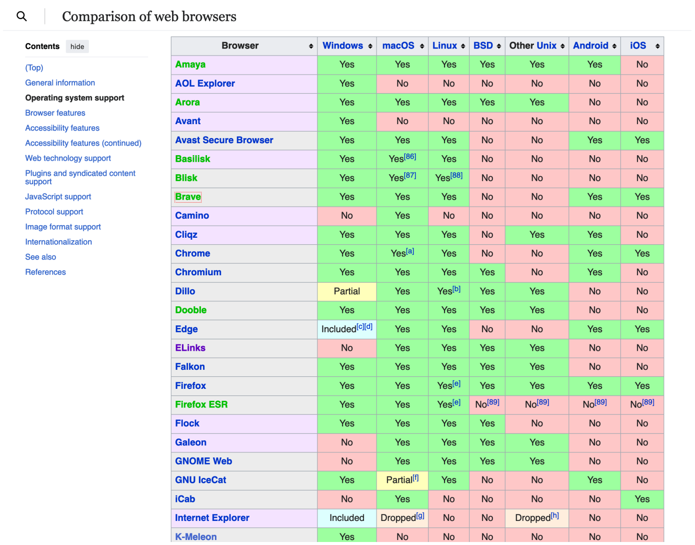
Some problems of arranging nearby things together

Plan the motion of a 3d printer



https://commons.wikimedia.org/wiki/File:Project_Arc_(7308324).jpeg

Some problems of arranging nearby things together



Comparison of web browsers

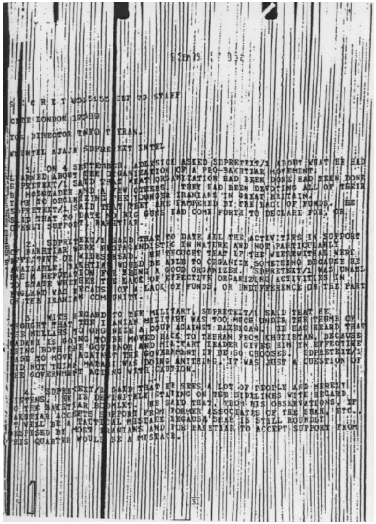
Contents hide

Browser	Windows	macOS	Linux	BSD	Other Unix	Android	iOS
Amaya	Yes	Yes	Yes	Yes	Yes	Yes	No
AOL Explorer	Yes	No	No	No	No	No	No
Arora	Yes	Yes	Yes	Yes	Yes	No	No
Avant	Yes	No	No	No	No	No	No
Avast Secure Browser	Yes	Yes	Yes	No	No	Yes	Yes
Basilisk	Yes	Yes[86]	Yes	No	No	No	No
Blisk	Yes	Yes[87]	Yes[88]	No	No	No	No
Brave	Yes	Yes	Yes	No	No	Yes	Yes
Camino	No	Yes	No	No	No	No	No
Cliqz	Yes	Yes	Yes	No	Yes	Yes	No
Chrome	Yes	Yes[a]	Yes	No	No	Yes	Yes
Chromium	Yes	Yes	Yes	Yes	No	Yes	No
Dillo	Partial	Yes	Yes[b]	Yes	Yes	No	No
Dooble	Yes	Yes	Yes	Yes	Yes	No	No
Edge	Included[c][d]	Yes	Yes	No	No	Yes	Yes
ELinks	No	Yes	Yes	Yes	Yes	No	No
Falkon	Yes	Yes	Yes	Yes	Yes	No	No
Firefox	Yes	Yes	Yes[e]	Yes	Yes	Yes	Yes
Firefox ESR	Yes	Yes	Yes[e]	No[89]	No[89]	No[89]	No[89]
Flock	Yes	Yes	Yes	Yes	No	No	No
Galeon	No	Yes	Yes	Yes	Yes	No	No
GNOME Web	No	Yes	Yes	Yes	Yes	No	No
GNU IceCat	Yes	Partial[f]	Yes	No	No	Yes	No
iCab	No	Yes	No	No	No	No	Yes
Internet Explorer	Included	Dropped[g]	No	No	Dropped[h]	No	No
K-Meleon	Yes	No	No	No	No	No	No

Product comparison

Reorder rows and columns so similar ones are adjacent

Some problems of arranging nearby things together



8 SEP 79 17 03 2

S E C R E T 051515Z SEP 79 STAFF

CITE LONDON 17789

TO: DIRECTOR INFO TEHRAN.

WNINTEL AJAJA SDPRETEXT INTEL

1. ON 4 SEPTEMBER, ADDESSICK ASKED SDPRETEXT/1 ABOUT WHAT HE HAD LEARNED ABOUT THE ORGANIZATION OF A PRO-BAKHTIAR MOVEMENT. SDPRETEXT/1 SAID THAT WHAT ORGANIZATION HAD BEEN DONE HAD BEEN DONE BY HOMAYUNZADEH AND A FEW OTHERS. THEY HAD BEEN DEVOTING ALL OF THEIR TIME TO ORGANIZING THE YOUNGER IRANIANS IN GREAT BRITAIN. SDPRETEXT/1 SAID THAT THEY ARE HAMPERED BY THE LACK OF FUNDS. HE SAID THAT TO DATE NO BIG GUNS HAD COME FORTH TO DECLARE FOR, OR OPENLY SUPPORT, BAKHTIAR.

2. SDPRETEXT/1 SAID THAT TO DATE ALL THE ACTIVITIES IN SUPPORT OF BAKHTIAR WERE PROPAGANDISTIC IN NATURE AND NOT PARTICULARLY EFFECTIVE OR WIDESPREAD. HE THOUGHT THAT IF THE WHEREWITHAL WERE AVAILABLE, BAKHTIAR WOULD BE ABLE TO ORGANIZE SOMETHING BECAUSE HE HAS A REPUTATION FOR BEING A GOOD ORGANIZER. SDPRETEXT/1 WAS UNABLE TO STATE WHETHER THE LACK OF EFFECTIVE ORGANIZING ACTIVITIES IN ENGLAND WAS BECAUSE OF A LACK OF FUNDS, OR INDIFFERENCE ON THE PART OF THE IRANIAN COMMUNITY.

3. WITH REGARD TO THE MILITARY, SDPRETEXT/1 SAID THAT HE THOUGHT THAT THE IRANIAN MILITARY WAS TOO MUCH UNDER THE THUMB OF THE MULLAHS TO ORGANIZE A COUP AGAINST BAZARGAN. HE HAS HEARD THAT MADANI IS GOING TO BE MOVED BACK TO TEHRAN FROM KHUZISTAN, BECAUSE BEING BOTH THE GOVERNOR AND MILITARY LEADER GIVES HIM AN EFFECTIVE BASE TO MOVE AGAINST THE GOVERNMENT IF HE SO CHOOSES. SDPRETEXT/1 DID NOT THINK MADANI WAS DOING ANYTHING. IT WAS JUST A QUESTION OF THE GOVERNMENT ACTING WITH CAUTION.

4. SDPRETEXT/1 SAID THAT HE SEES A LOT OF PEOPLE AND MERELY LISTENS. HE IS DEFINITELY STANDING ON THE SIDELINES WITH REGARD TO THE BAKHTIAR BOOMLET. HE SAID THAT, FROM HIS OBSERVATIONS, IF BAKHTIAR ACCEPTS SUPPORT FROM FORMER ASSOCIATES OF THE SHAH, ETC. IT WILL BE A TACTICAL MISTAKE BECAUSE SHAH IS STILL ROUNDLY DESPISED BY MOST IRANIANS AND FOR BAKHTIAR TO ACCEPT SUPPORT FROM THIS QUARTER WOULD BE A MISTAKE.

31

Reassembling small (possibly overlapping) pieces of a document into the original unshredded document

This is how DNA sequencing works!

The traveling salesperson problem

For distances:

Input is a matrix of distances between n points, satisfying

- Symmetric: $D[i,j] = D[j,i]$
- Positive: $D[i,j] \geq 0$ $(0 \Rightarrow i = j)$
- Triangle inequality: $D[i,j] + D[j,k] \geq D[i,k]$

Goal: find cyclic order using each point exactly once, minimizing sum of distances between consecutive points

For graphs:

Input: connected undirected graph with edge lengths > 0

Goal: find a walk that visits each vertex at least once, returning to start vertex, minimizing sum of edge lengths

Converting between matrices and graphs

To convert a distance matrix D into an equivalent graph problem:

- Use a complete graph: an undirected graph with an edge between each pair of vertices
- Weight each edge by the distance between endpoints
- If you get a solution with repeated vertices, omit them
- Triangle inequality implies that omitting repetitions cannot increase tour length

To convert a graph into an equivalent distance matrix:

- Compute matrix of all-pairs shortest path distances
- Turn any tour of the resulting matrix into a walk in the graph by concatenating together shortest paths between consecutive vertices in the tour

Hardness

TSP is NP-hard. Even for graphs with all weights $= 1$, finding a walk with total length $\leq n$ is NP-complete.

- NP: yes-no problems that can be solved by a brute force search over solutions of polynomial size. For weight-1 TSP, try all length-n walks, checking whether any walk visits all vertices.
- NP-complete: In NP and all other NP problems can be translated into it. E.g., we could find an input to a Boolean circuit that makes the circuit output true by translating into an equivalent TSP problem and then using a TSP solver.
- TSP itself is not in NP because it's not a yes-no problem. That's why it's called NP-hard rather than NP-complete.

Implications of hardness

All NP-complete problems including TSP can be solved by exponential-time brute force search algorithms

Some of them have faster (still exponential time!) algorithms based on other methods; we will see one for TSP next time

None have known polynomial-time algorithms; if you found one, it would give you a general method for automatically translating all brute force searches into much more efficient algorithms.

We think a polynomial-time algorithm does not exist
but we don't know how to prove that.

Finding a polynomial-time algorithm or proving its nonexistence would win a $1,000,000 prize

Approximation algorithms

If we can't solve it quickly and exactly, what can we do?

- Solve it slowly and exactly (next time)
- Solve it quickly and less exactly, but still with some guarantees on solution quality (this time)

Approximation algorithm:

- Algorithm for producing inexact solutions
- Should run in polynomial time
- Should produce solution of guaranteed quality

How to measure approximation quality?

Approximation ratio of an algorithm: The largest possible value

algorithm's solution quality/optimal solution quality

that any worst-case input to the algorithm can cause it to produce
(for minimization problems like TSP where bigger ratios are bad)

Goal: Get the approximation ratio is as close to 1 as possible

Overview of TSP approximation

- Approximation ratio 2 is easy using minimum spanning trees
- Approximation ratio $\frac{3}{2}$ known since 1976 using minimum spanning trees + matching [Christofides 1976; Serdyukov 1978]
- Recent breakthrough: $\frac{3}{2} - \frac{1}{10^{36}}$ [Karlin et al. 2021]

  Very rough sketch: random spanning trees + matching
- Several special cases have better approximations, e.g. for unweighted graphs (or all edge lengths 1) there is an algorithm with ratio $\frac{7}{5}$ [Sebő and Vygen 2014]
- For the general problem, NP-hard to approximate with ratio better than $\frac{123}{122}$ [Karpinski et al. 2015]

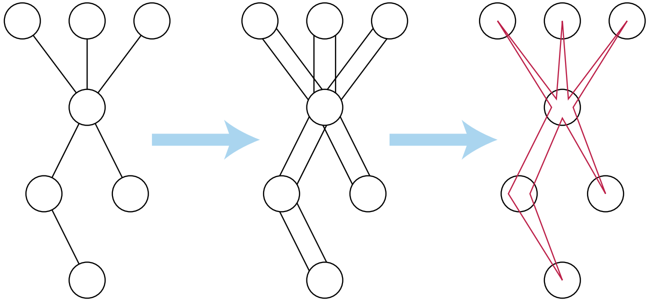
  Still a big gap between $\frac{123}{122}$ and $\frac{3}{2} - \frac{1}{10^{36}}$ open for research

2-approximation

Use graph version of TSP, so repeated vertices are allowed

Find a minimum spanning tree

Make two copies of each tree edge $\Rightarrow$ multigraph, all degrees even



Find an Euler tour and return it as the approximate TSP tour

Why is this a 2-approximation?

Approximate tour $=$ 2(minimum spanning tree)

Optimal tour $\geq$ path formed by removing any one of its edges

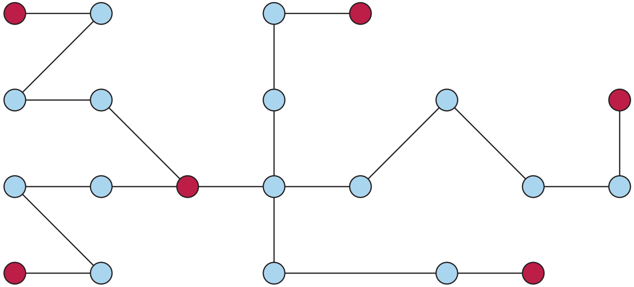
But this path is a spanning tree!

So length of minimum spanning tree $\leq$ length of optimal tour

Put it together: approximate tour $\leq$ 2 optimal tour

3/2-approximation, step 1

Construct a minimum spanning tree

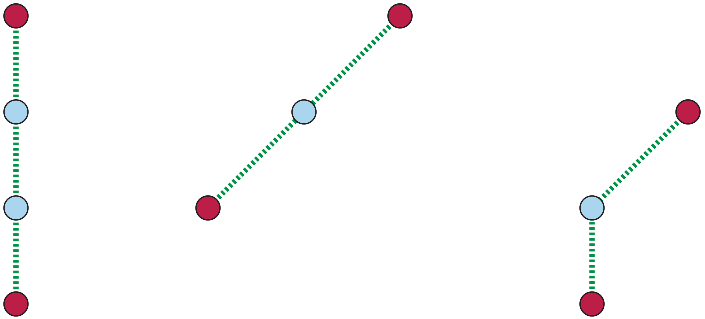


Identify its odd vertices
(odd by their degree in the spanning tree, not in the whole graph)

Before using an Euler tour, we need to make their degrees even, more cheaply than doubling every edge in the whole tree

3/2-approximation, step 2

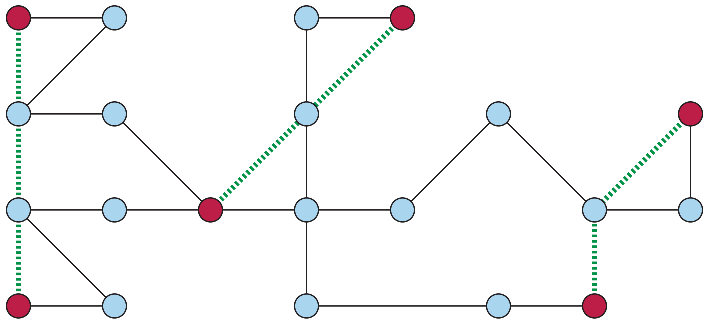
Handshaking lemma: The number of odd vertices we found is even



Connect them in pairs by shortest paths from the original graph choosing pairings in a way that minimizes the sum of path lengths (this is a problem of matching; we will discuss matching in week 9)

3/2-approximation, step 3

Put minimum spanning tree and paired paths together in one graph
(or multigraph if the paths and spanning tree have shared edges)

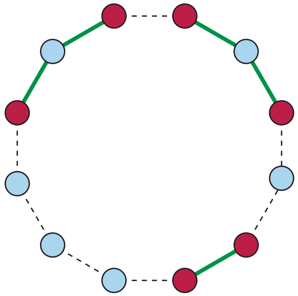


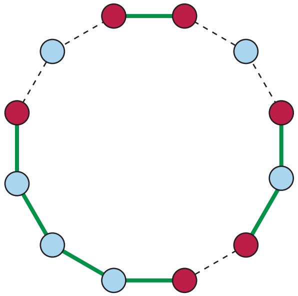
Find an Euler tour and return it as the approximate TSP tour

Why is it a 3/2-approximation?

Minimum spanning tree $\leq$ optimal tour as before

There are two ways to pair consecutive odd vertices in the optimal tour, by paths around the tour. Together they add to whole tour





The pairing by shortest paths that we find is at least as good as the best of these two ways $\Rightarrow \leq \frac{1}{2}$ optimal tour

Morals of the story

Equivalence between graph and distance versions of traveling salesperson

What it means for traveling salesperson to be NP-hard:

We don't know how to solve it in polynomial time

Although we think there is no polynomial-time solution, we also don't know how to prove it is impossible

The general concept of an approximation algorithm, and the 2- and 3/2-approximations for traveling salesperson

References

Nicos Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Graduate School of Industrial Administration, CMU, 1976. URL https://apps.dtic.mil/dtic/tr/fulltext/u2/a025602.pdf.

Anna R. Karlin, Nathan Klein, and Shayan Oveis Gharan. A (slightly) improved approximation algorithm for metric TSP. In *Proc. 53rd ACM Symp. on Theory of Computing (STOC '21)*, pages 32–45, 2021. doi:10.1145/3406325.3451009.

Marek Karpinski, Michael Lampis, and Richard Schmied. New inapproximability bounds for TSP. *Journal of Computer and System Sciences*, 81(8):1665–1677, 2015. doi:10.1016/j.jcss.2015.06.003.

András Sebő and Jens Vygen. Shorter tours by nicer ears: 7/5-approximation for the graph-TSP, 3/2 for the path version, and 4/3 for two-edge-connected subgraphs. *Combinatorica*, 34(5):597–629, 2014. doi:10.1007/s00493-014-2960-3.

Anatoliy I. Serdyukov. On some extremal walks in graphs. *Upravlyaemye Sistemy*, 17: 76–79, 1978. URL https://nas1.math.nsc.ru/aim/journals/us/us17/us17_007.pdf.