Algorithms on Graphs

Umang Bhaskar & Juhi Chaudhary

Day 2, session 2: TSP

टामूअसं
tifr

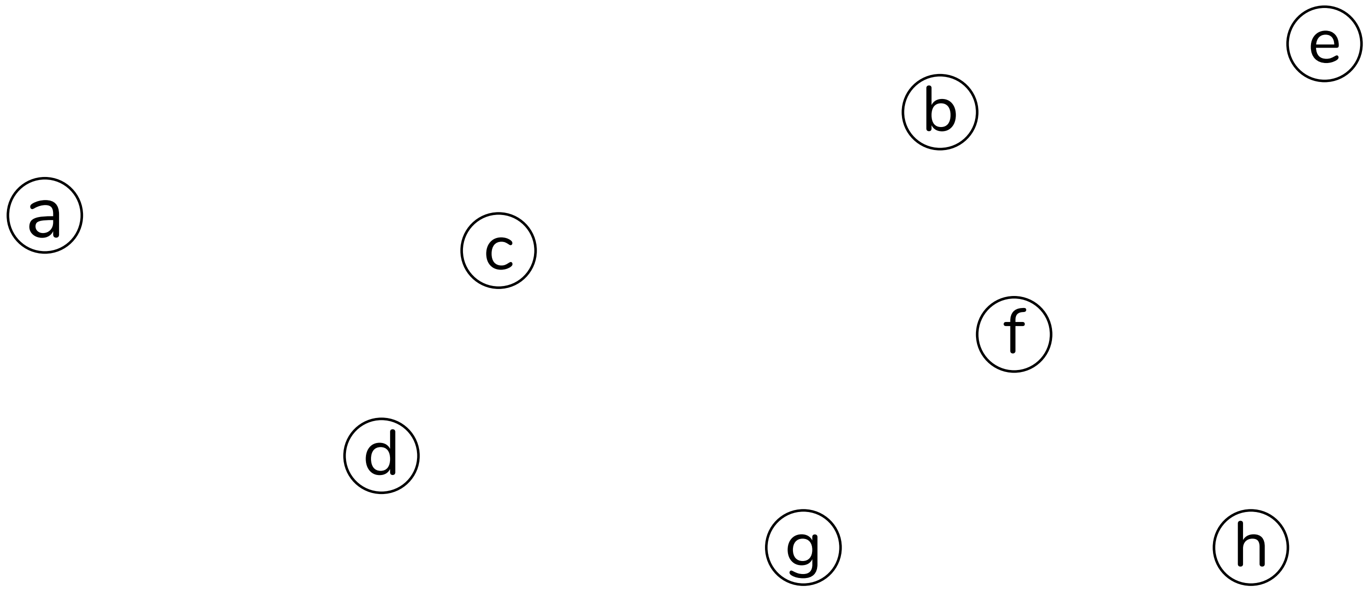
STCS Vigyan Vidushi 2024

TIFR
Vigyan Vidushi

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

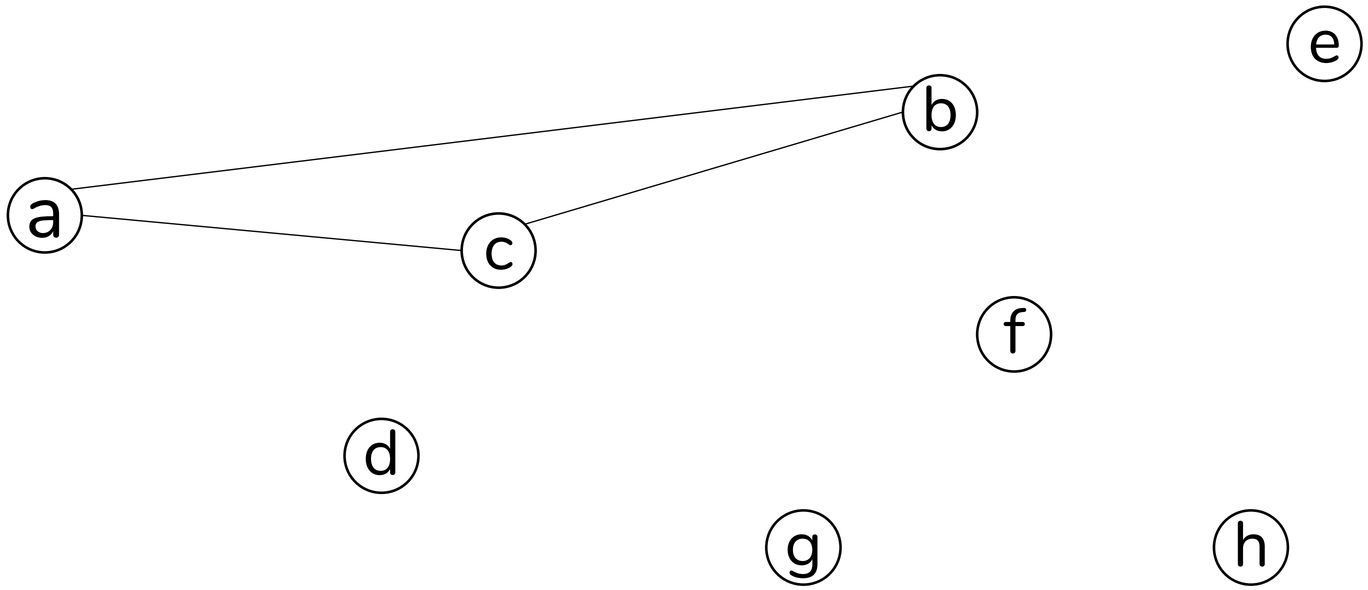


The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

by triangle inequality, $d(a,b) \leq d(a,c) + d(c,b)$

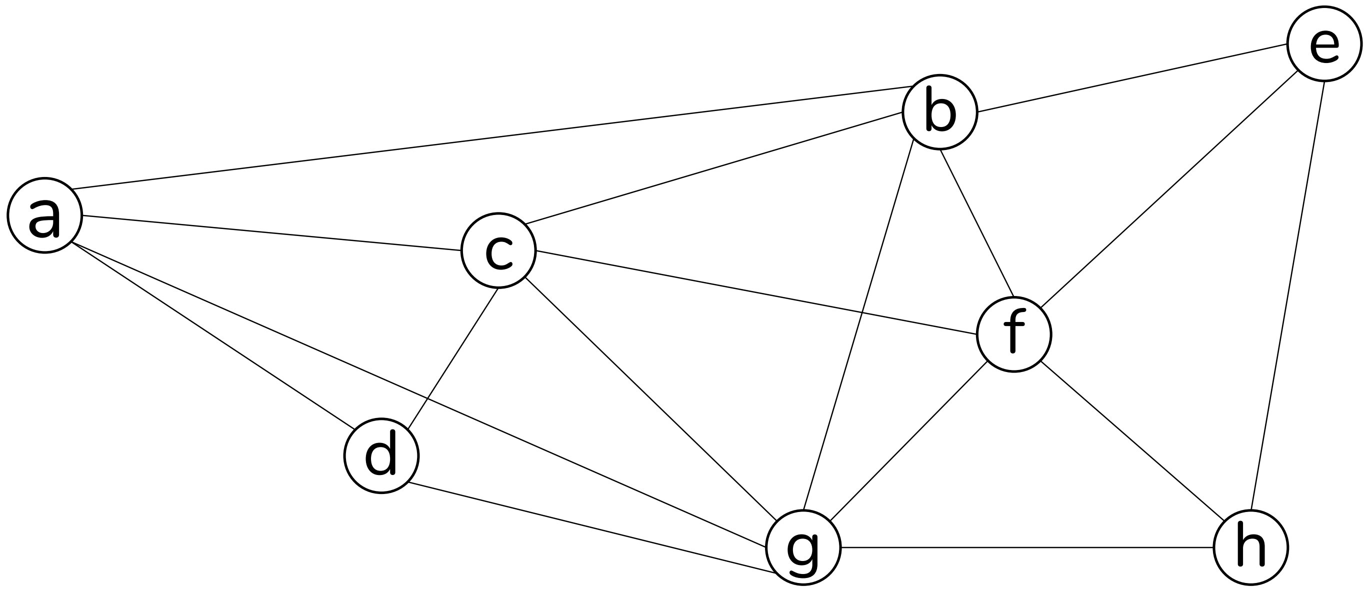


The Traveling Salesperson Problem

<u>Given</u>: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

by triangle inequality, $d(a,b) \leq d(a,c) + d(c,b)$



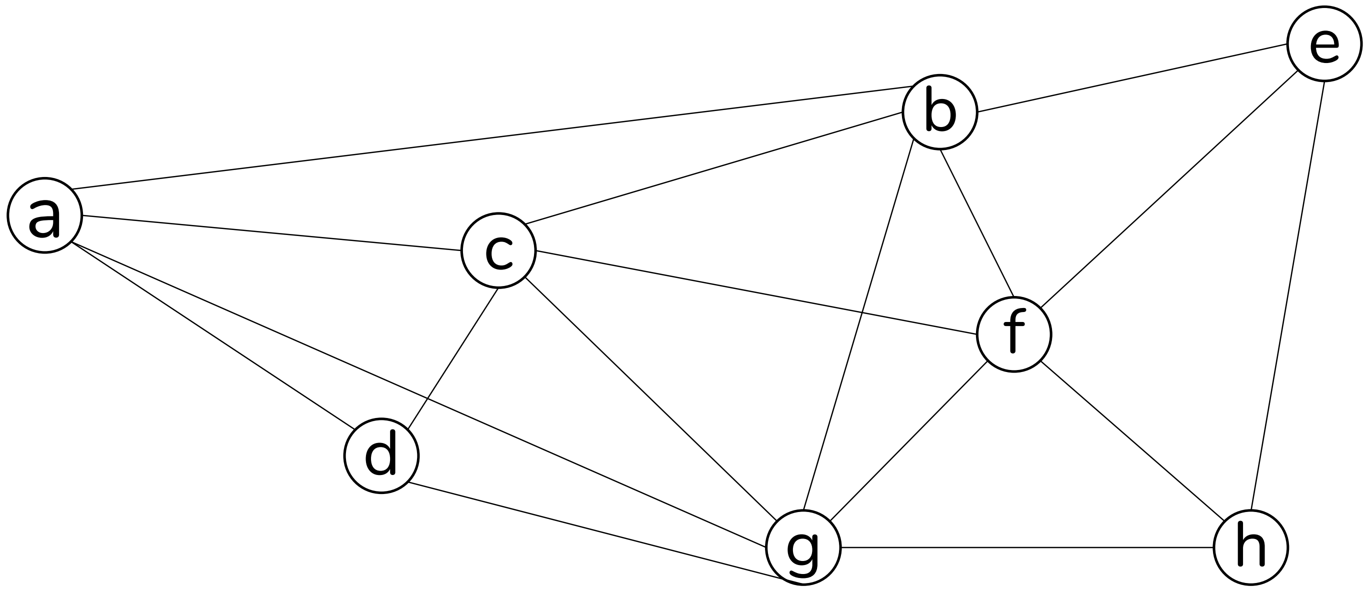
(complete graph with edge lengths)

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.



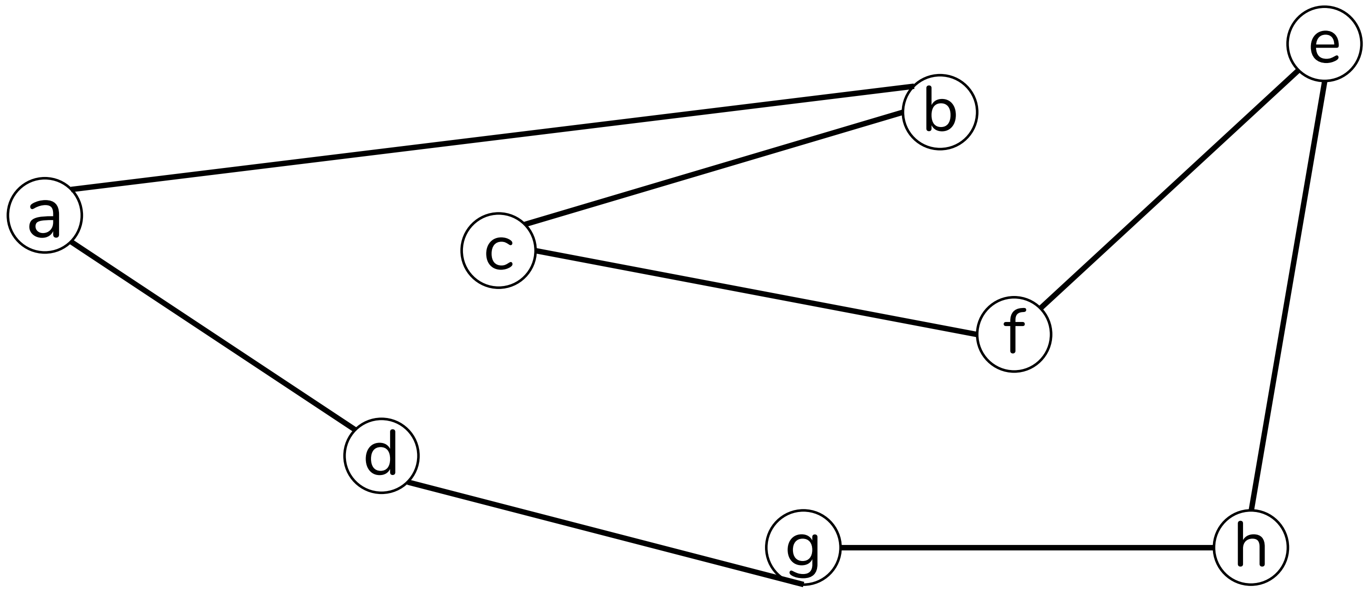
(complete graph with edge lengths)

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible <u>cycle</u> that visits all points.



(complete graph
with edge lengths)

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.

Trivial algorithm:

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible <u>cycle</u> that visits all points.

Trivial algorithm: Each cycle corresponds to a permutation of the vertices

Try all $n! \sim 2^{n \log n}$ permutations

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.

Trivial algorithm: Each cycle corresponds to a permutation of the vertices

Try all $n! \sim 2^{n \log n}$ permutations

Better algorithms? Can do $O(2^n)$ by a dynamic program (Held-Karp algorithm)

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible <u>cycle</u> that visits all points.

$\boxed{\text{TSP is NP-hard}}$

(what does this mean?)

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.

TSP is NP-hard

(what does this mean?)

There probably isn't a polynomial-time algorithm for this problem.

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.

TSP is NP-hard

(what does this mean?)

There probably isn't a polynomial-time algorithm for this problem.

Instead of an optimal solution, we look for a near-optimal solution.

The Traveling Salesperson Problem

Given: n points in R^2

point $a = \left(a_x, a_y\right)$, distance $d(a, b) = \sqrt{(a_x - b_x)^2 + \left(a_y - b_y\right)^2}$

TSP: find shortest possible cycle that visits all points.

For any instance I of n points in R^2,
let $OPT(I)$ denote the length of the shortest possible cycle.

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.

For any instance I of n points in R^2,
let $OPT(I)$ denote the length of the shortest possible cycle.

Definition: An algorithm is an α–approximation
if the cycle it obtains for any instance I
has length at most $\alpha \times OPT(I)$.

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.

For any instance I of n points in R^2,
let $OPT(I)$ denote the length of the shortest possible cycle.

Definition: An algorithm is an α–approximation
if the cycle it obtains for any instance I
has length at most $\alpha \times OPT(I)$.

A 2-approximate algorithm gives a cycle of length at most $2 \times OPT(I)$.

The Traveling Salesperson Problem

Given: n points in R^2

point $a = (a_x, a_y)$, distance $d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

TSP: find shortest possible cycle that visits all points.

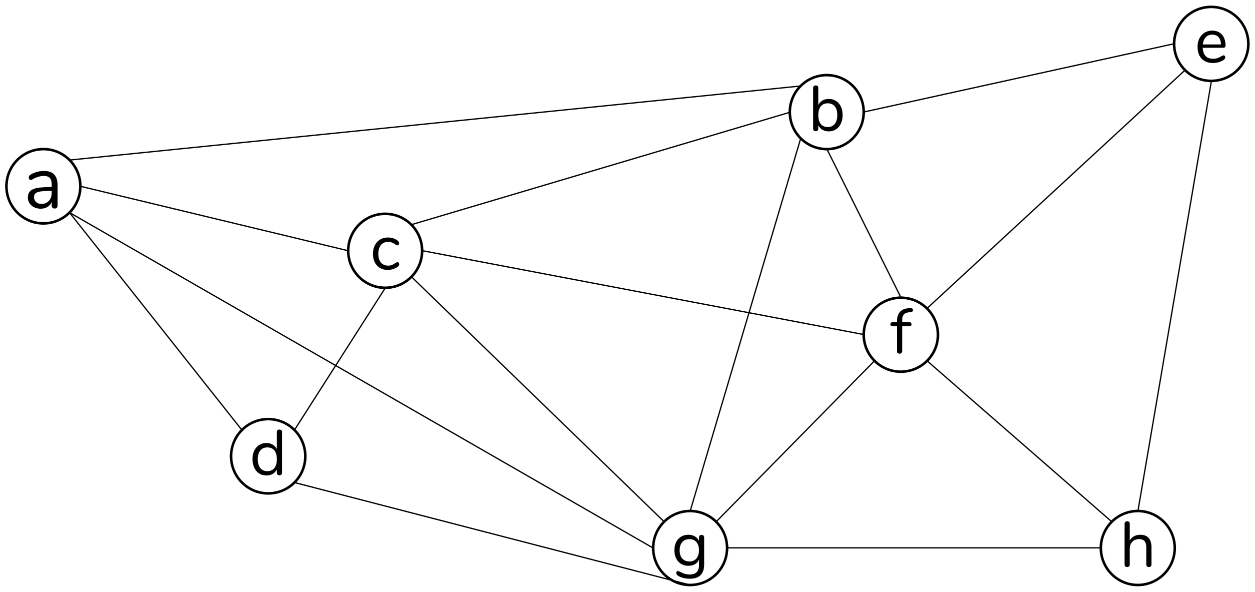
For any instance I of n points in R^2,
let $OPT(I)$ denote the length of the shortest possible cycle.

Definition: An algorithm is an α–approximation
if the cycle it obtains for any instance I
has length at most $\alpha \times OPT(I)$.

We will give a 1.5-approximate algorithm for TSP.

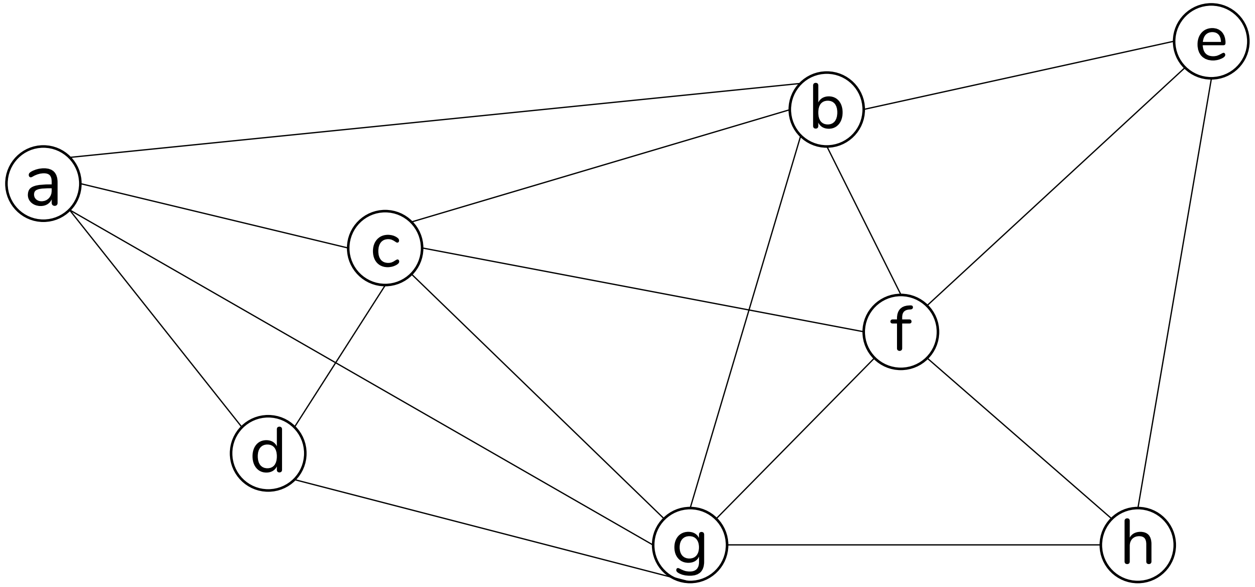
The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible <u>cycle</u> that visits all points.



The Traveling Salesperson Problem

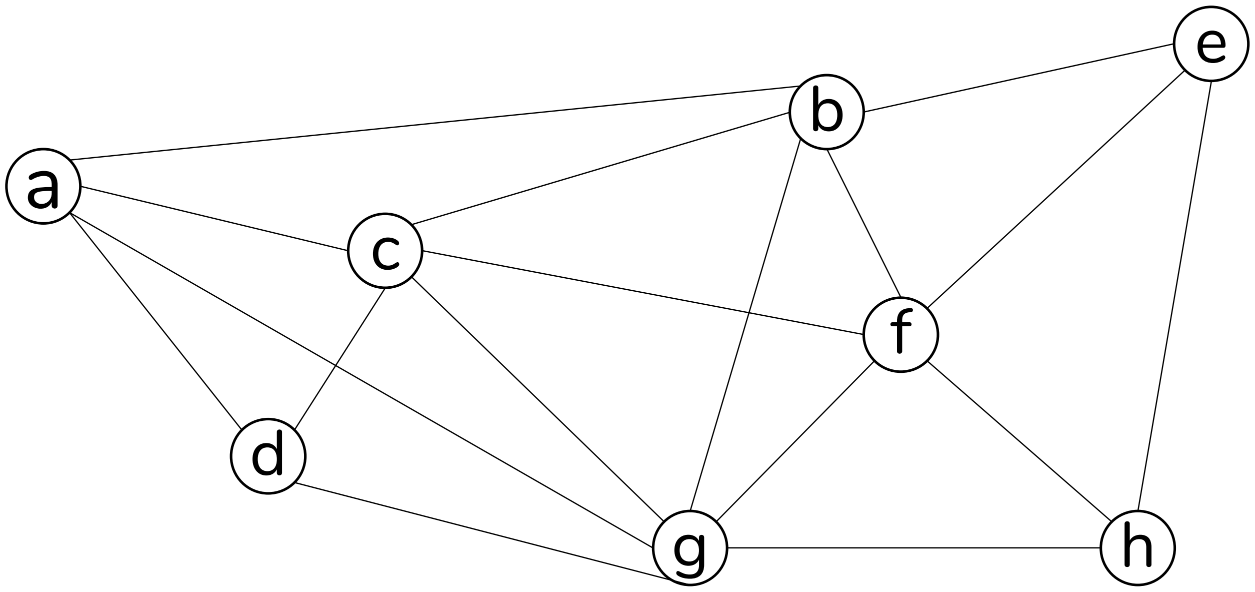
<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph



The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

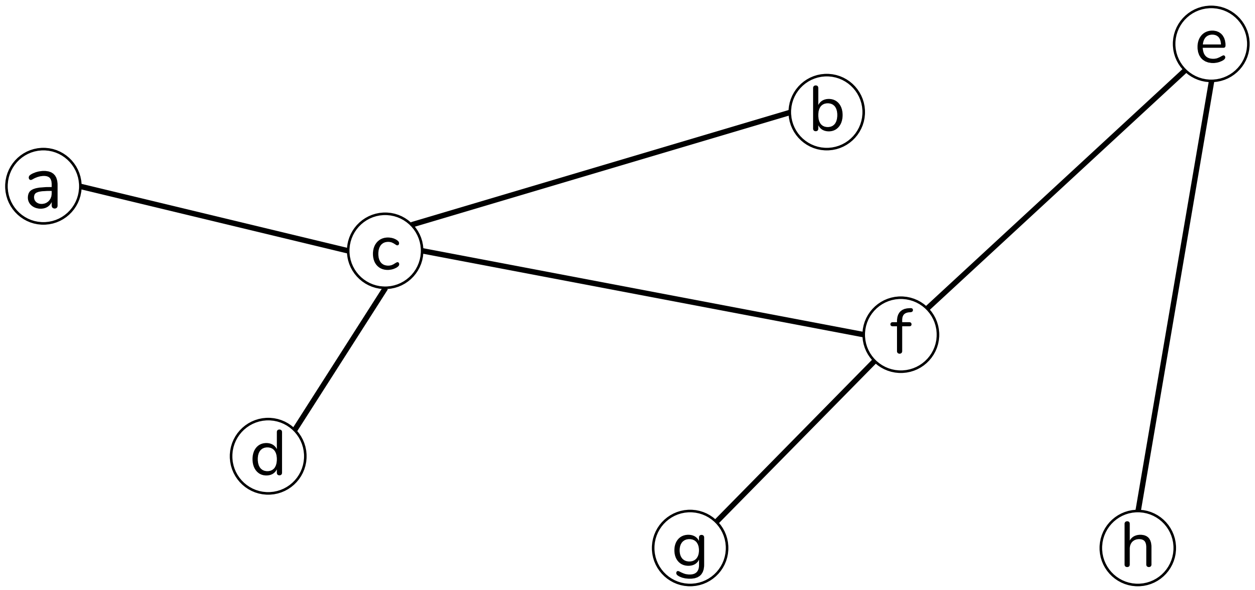
This is a minimum spanning tree!



The Traveling Salesperson Problem

Given: n points in R^2 find shortest possible ~~cycle~~ graph that visits all points.

This is a minimum spanning tree!

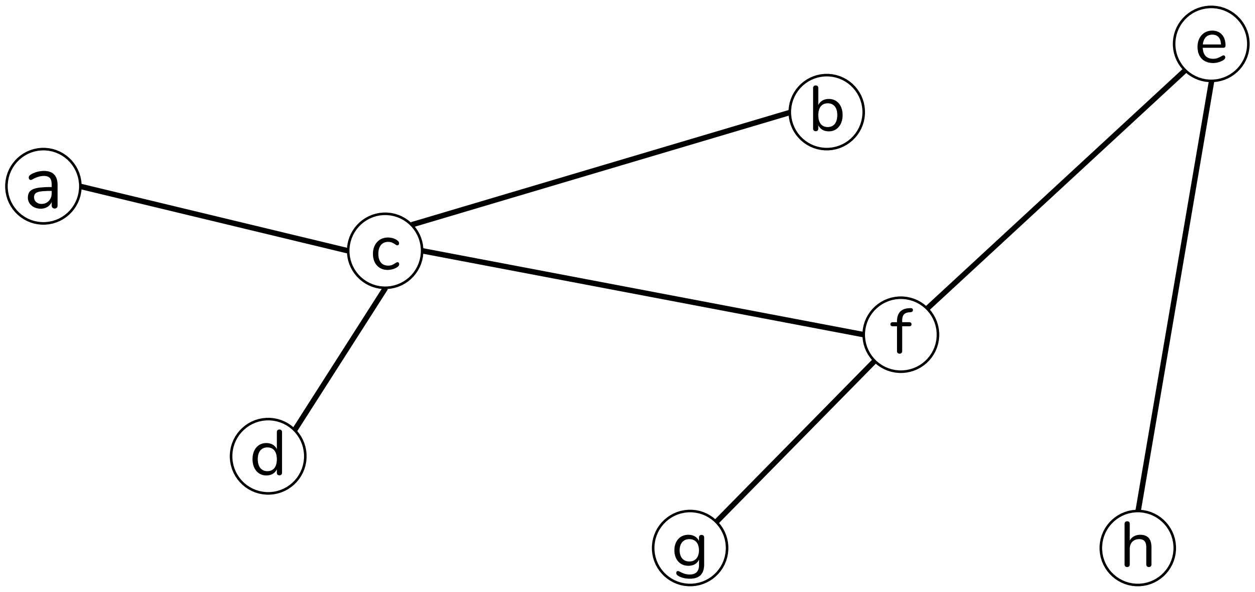


Claim: length of MST $\leq$ length of shortest cycle that visits all pts.

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

This is a minimum spanning tree!

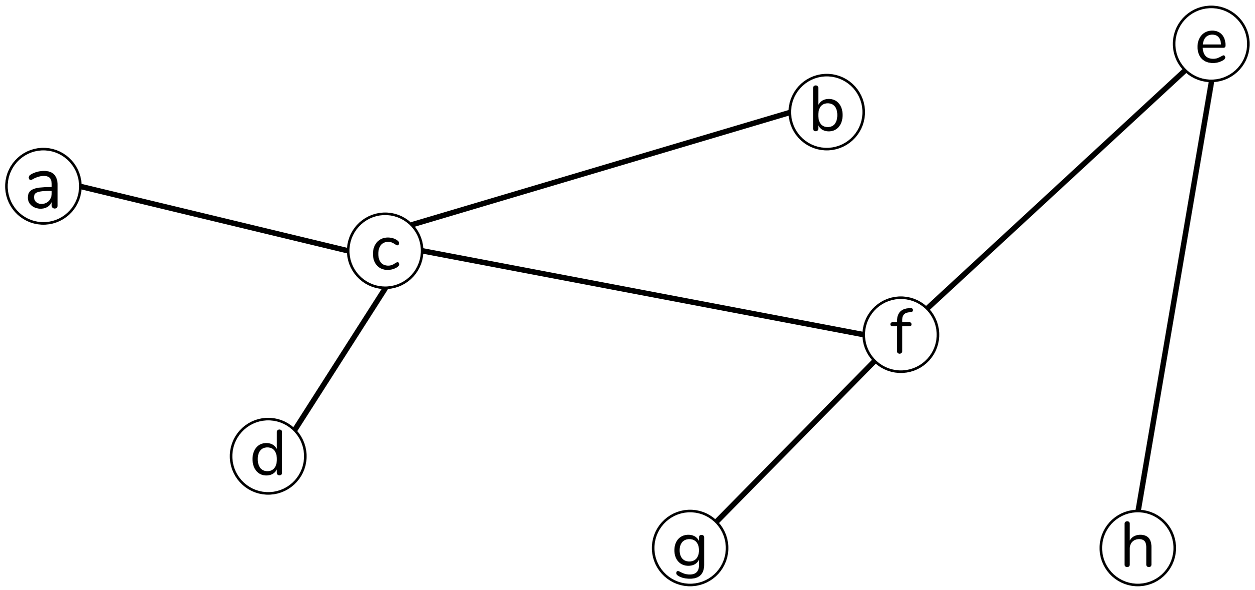


<u>Claim</u>: length of MST $\leq$ length of OPT

The Traveling Salesperson Problem

Given: n points in R^2 find shortest possible ~~cycle~~ graph that visits all points.

This is a minimum spanning tree!



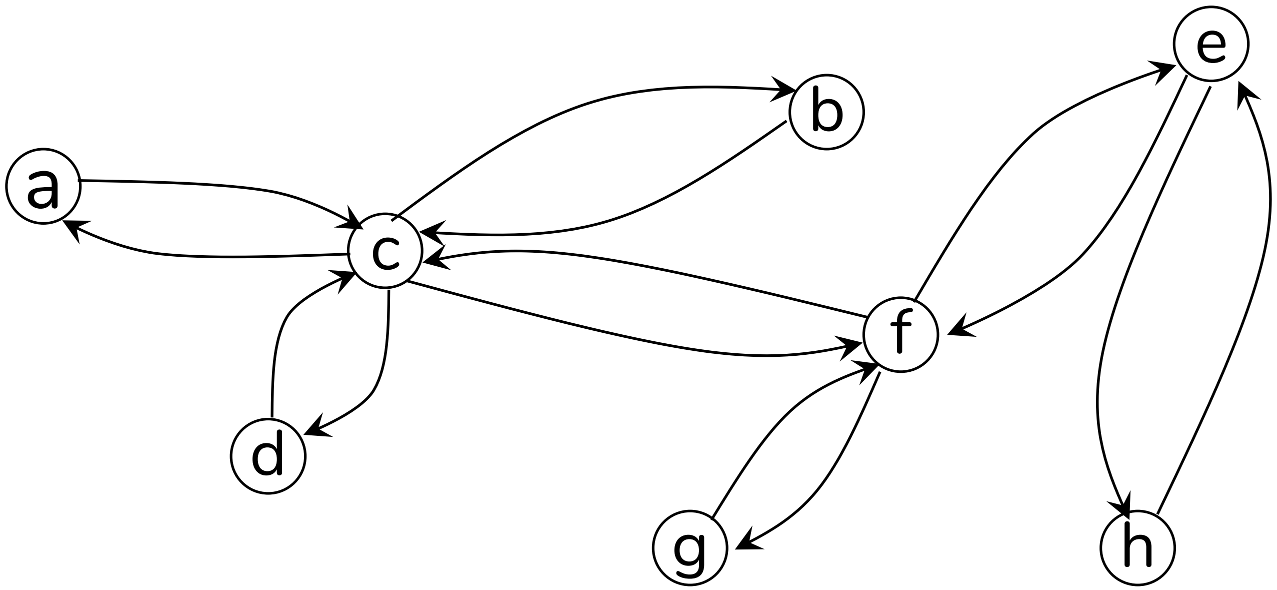
Claim: length of MST $\leq$ length of OPT

Sure, but we want a cycle…

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

This is a minimum spanning tree!



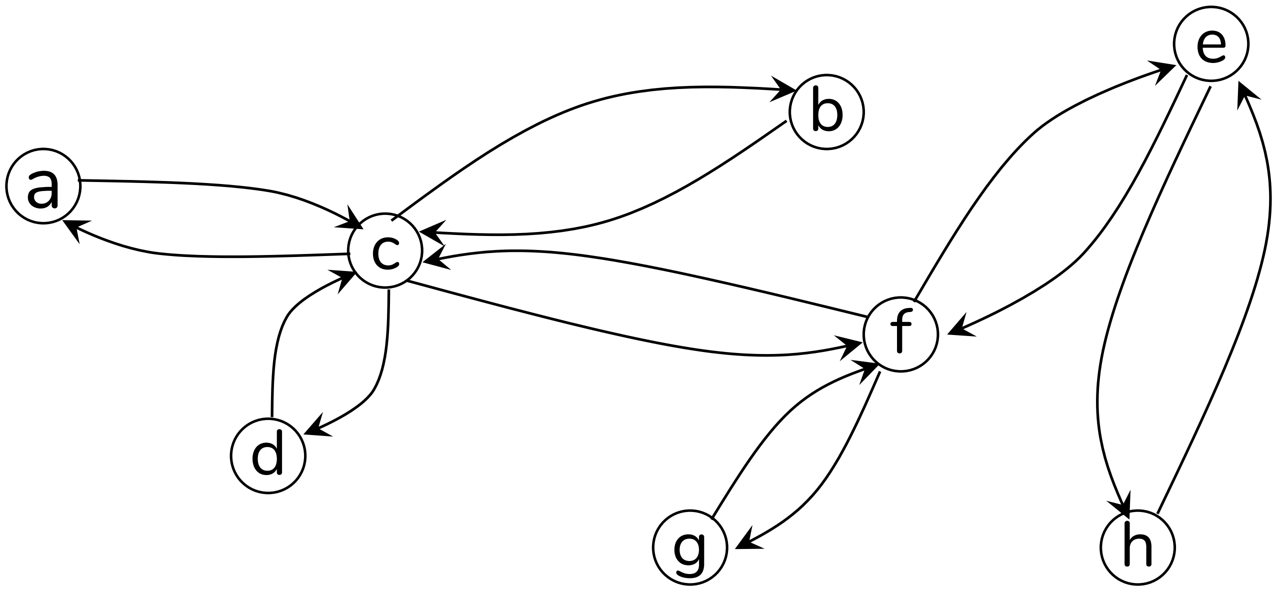
<u>Claim</u>: length of MST $\leq$ length of OPT

Sure, but we want a cycle…

The Traveling Salesperson Problem

Given: n points in R^2 find shortest possible ~~cycle~~ graph that visits all points.

This is a minimum spanning tree!



Claim: length of MST $\leq$ length of OPT

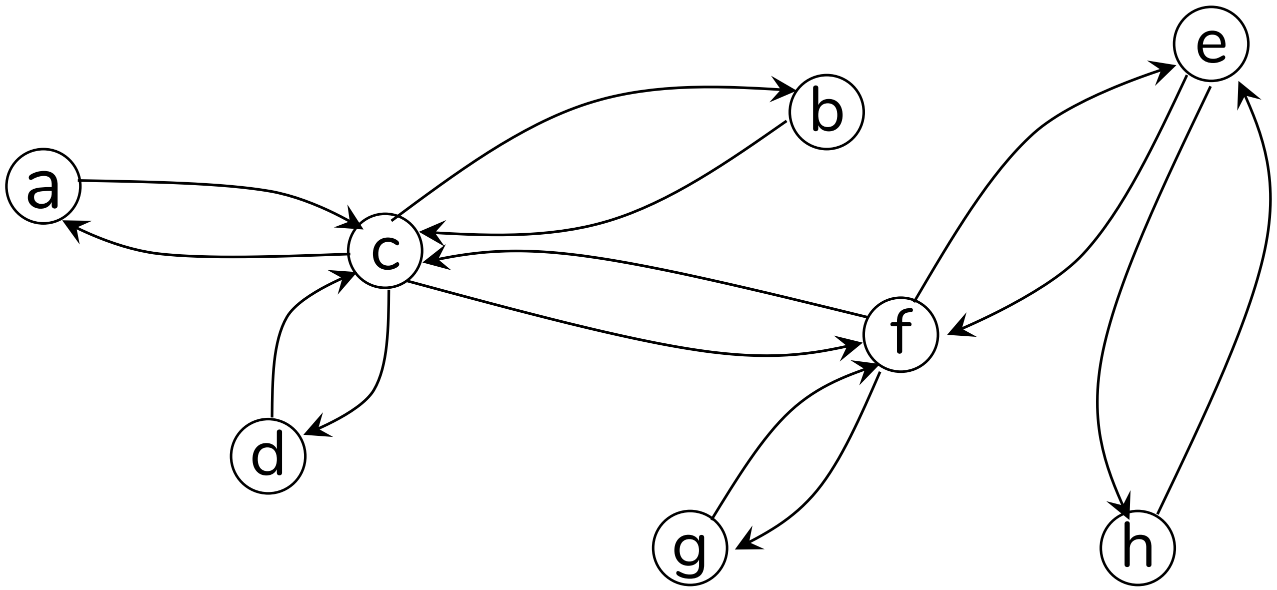
Sure, but we want a cycle…

By doubling edges of MST, can get a walk of length 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

Given: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

This is a minimum spanning tree!



Claim: length of MST $\leq$ length of OPT

Sure, but we want a cycle...

By doubling edges of MST, can get a walk of length 2 x MST $\leq$ 2 x OPT

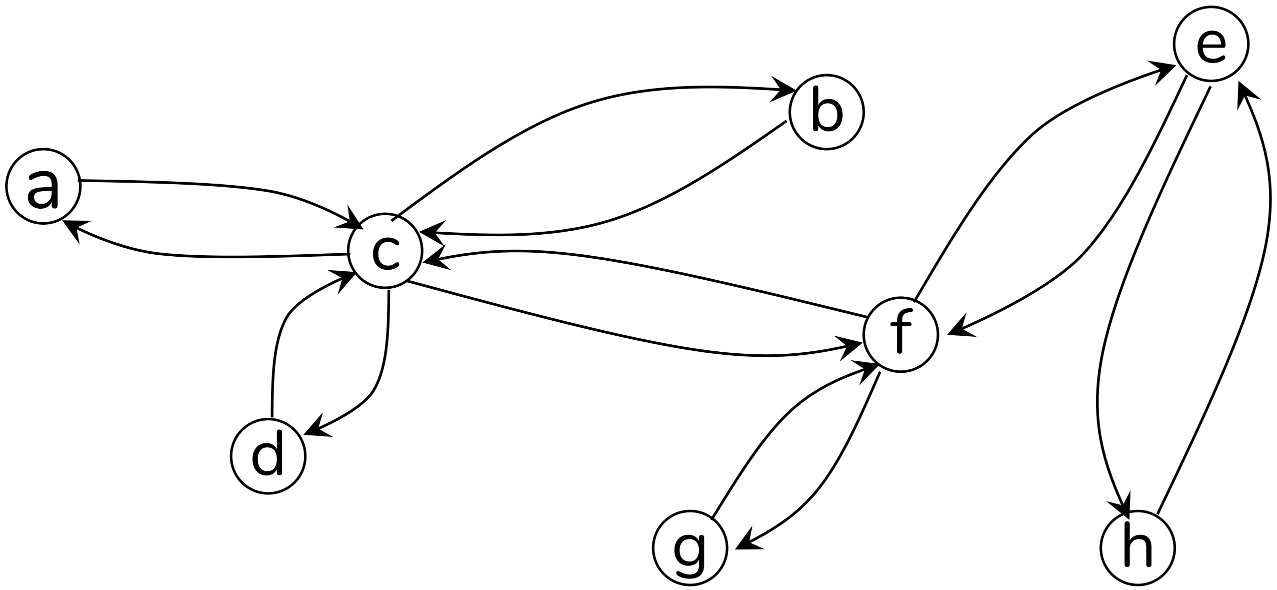
Can short-circuit walk to get a cycle of length $\leq$ 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ graph that visits all points.

This is a minimum spanning tree!

a – c – b – c – d – c – f – e – h – e – f – g – f – c – a



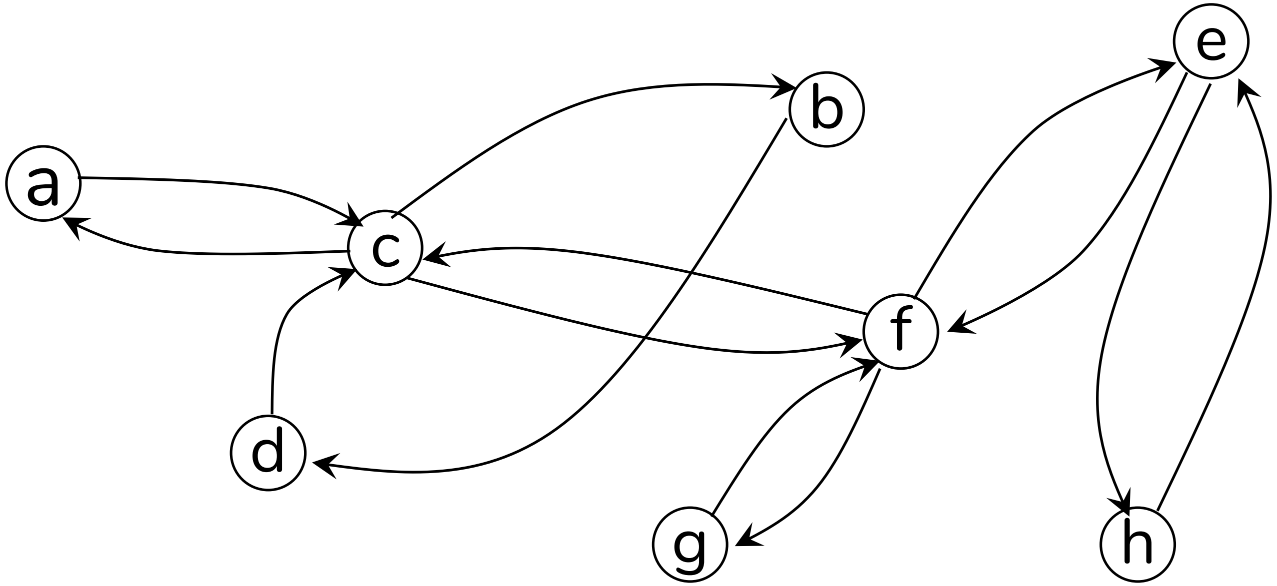
Can <u>short-circuit walk</u> to get a cycle of length $\leq$ 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ graph that visits all points.

This is a minimum spanning tree!

a – c – b – ~~c –~~ d – c – f – e – h – e – f – g – f – c – a



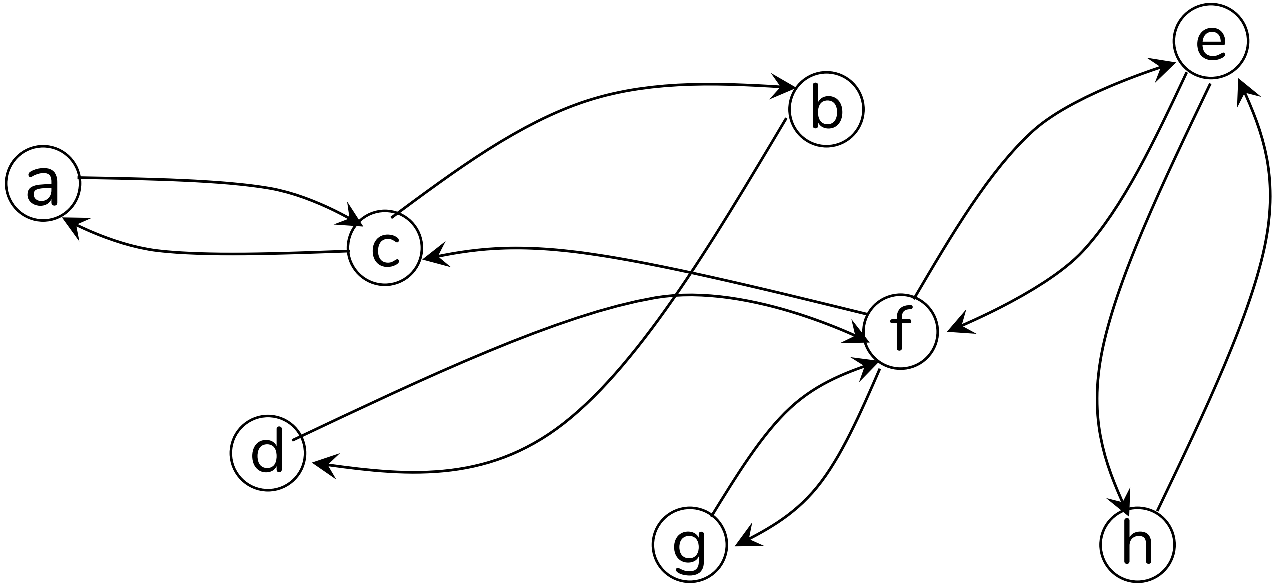
Can <u>short-circuit walk</u> to get a cycle of length $\leq$ 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

This is a minimum spanning tree!

a – c – b – ~~c –~~ d – ~~c –~~ f – e – h – e – f – g – f – c – a



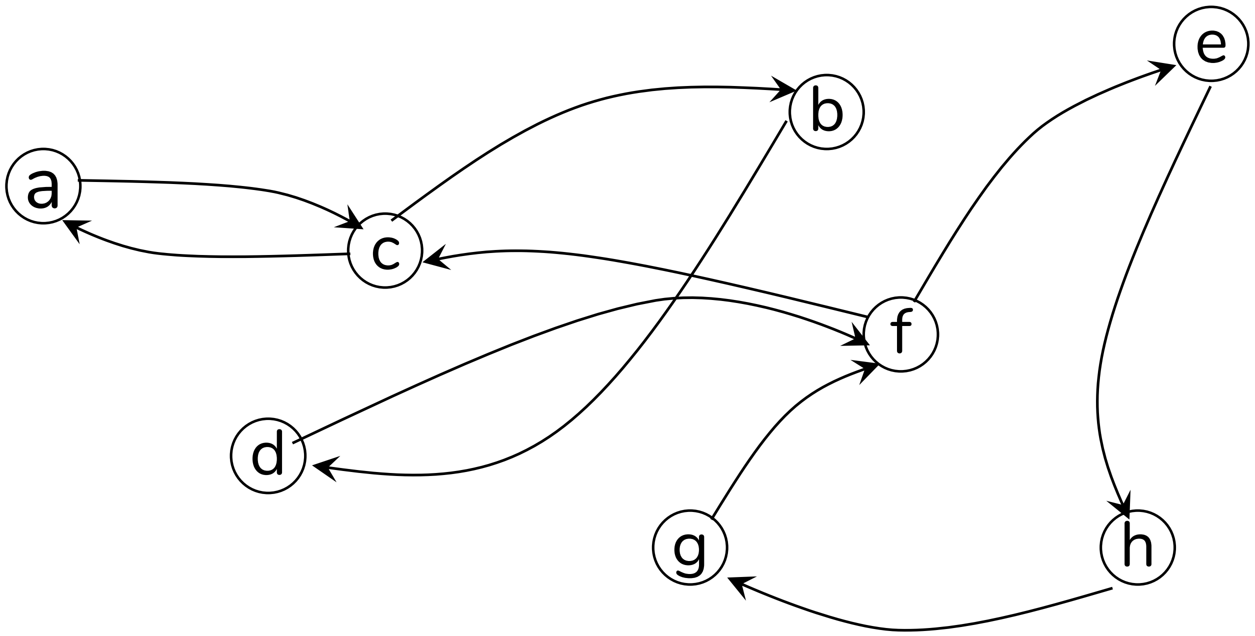
Can <u>short-circuit walk</u> to get a cycle of length $\leq$ 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

<u>Given</u>: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

This is a minimum spanning tree!

a – c – b – ~~c –~~ d – ~~c –~~ f – e – h – ~~e – f –~~ g – f – c – a



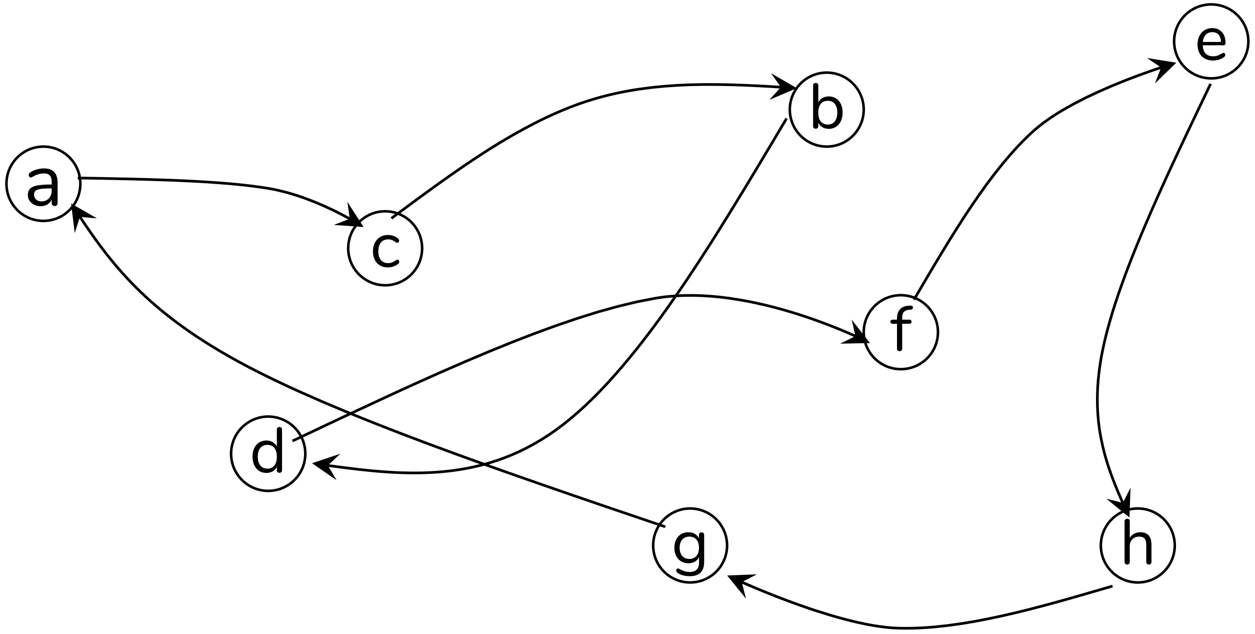
Can <u>short-circuit walk</u> to get a cycle of length $\leq$ 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

Given: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

This is a minimum spanning tree!

a – c – b – ~~c –~~ d – ~~c –~~ f – e – h – ~~e – f~~ – g – ~~f – c –~~ a



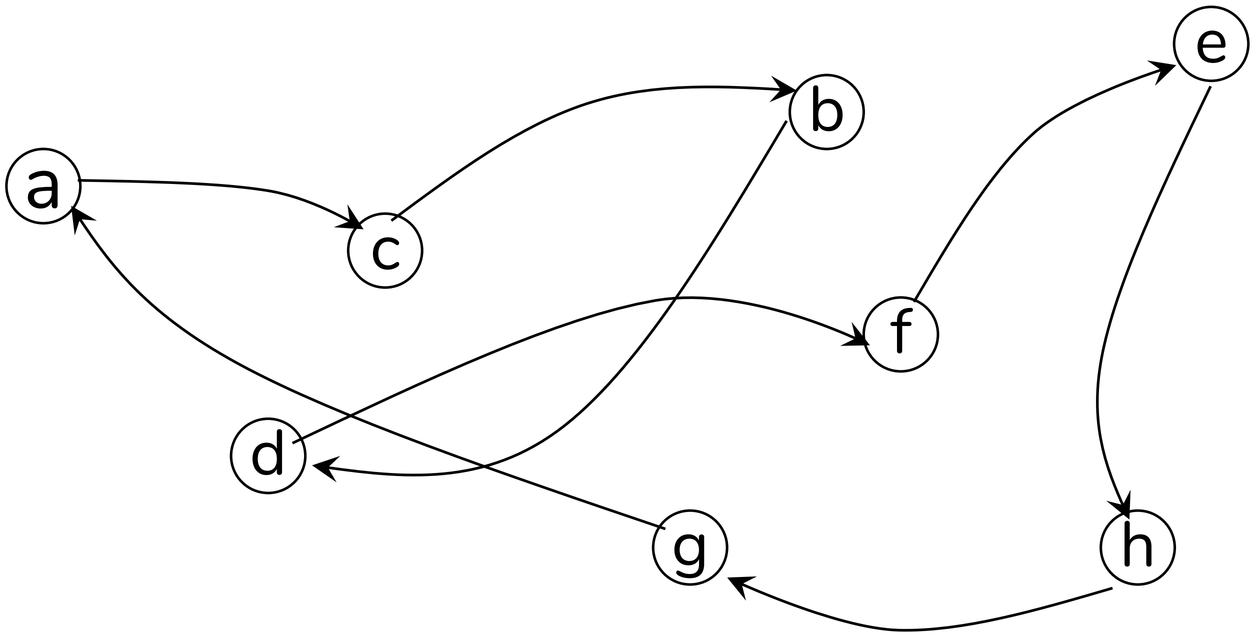
Can short-circuit walk to get a cycle of length $\leq$ 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

Given: n points in R^2 find shortest possible ~~cycle~~ that visits all points.
graph

This is a minimum spanning tree!

a – c – b – ~~c –~~ d – ~~c –~~ f – e – h – ~~e – f~~ – g – ~~f – c –~~ a



This gives us a 2-approximation algorithm for TSP.

Can short-circuit walk to get a cycle of length $\leq$ 2 x MST $\leq$ 2 x OPT

The Traveling Salesperson Problem

Given: n points in R^2 find shortest possible cycle that visits all points.

A 2-approximation algorithm:

1. Find MST (length ≤ OPT)
2. Double edges, find Eulerian walk (length ≤ 2 x OPT)
3. Short-circuit edges to get cycle of length ≤ 2 x OPT

(Remainder on board)