The Traveling Salesman Problem

Definition: A **complete graph** K_N is a graph with N vertices and an edge between every two vertices.

Definition: A **weighted graph** is a graph in which each edge is assigned a *weight* (representing the time, distance, or cost of traversing that edge).

Definition: A **Hamilton circuit** is a circuit that uses every vertex of a graph once.

Definition: The **Traveling Salesman Problem (TSP)** is the problem of finding a **minimum-weight Hamilton circuit** in K_N.

The Traveling Salesman Problem

Example: Willy decides to visit every Australian city important enough to be listed on this Wikipedia page.

The Traveling Salesman Problem

Example: Willy decides to visit every Australian city important enough to be listed on this Wikipedia page.

To avoid rental-car fees, he must finish the tour in the same city he starts it in.

The Traveling Salesman Problem

Example: Willy decides to visit every Australian city important enough to be listed on this Wikipedia page.

To avoid rental-car fees, he must finish the tour in the same city he starts it in.

What route minimizes the total distance he has to travel?

The Traveling Salesman Problem

Example: Willy decides to visit every Australian city important enough to be listed on this Wikipedia page.

To avoid rental-car fees, he must finish the tour in the same city he starts it in.

What route minimizes the total distance he has to travel?

I.e., in this weighted K_{16}, which Hamilton circuit has the smallest total weight?

The Brute-Force Algorithm

Willy could solve the problem by **brute force**:

- Make a list of all possible Hamilton circuits.

The Brute-Force Algorithm

Willy could solve the problem by **brute force**:

- Make a list of all possible Hamilton circuits.
- Calculate the weight of each Hamilton circuit by adding up the weights of its edges.

The Brute-Force Algorithm

Willy could solve the problem by **brute force**:

- Make a list of all possible Hamilton circuits.
- Calculate the weight of each Hamilton circuit by adding up the weights of its edges.
- Pick the Hamilton circuit with the smallest total weight.

The Brute-Force Algorithm

Willy could solve the problem by **brute force**:

- Make a list of all possible Hamilton circuits.
- Calculate the weight of each Hamilton circuit by adding up the weights of its edges.
- Pick the Hamilton circuit with the smallest total weight.

BIG PROBLEM:

The Brute-Force Algorithm

Willy could solve the problem by **brute force**:

- Make a list of all possible Hamilton circuits.
- Calculate the weight of each Hamilton circuit by adding up the weights of its edges.
- Pick the Hamilton circuit with the smallest total weight.

BIG PROBLEM: There are 16 vertices, so there are

$$(16-1)! \;=\; 15! \;=\; 1,307,674,368,000$$

Hamilton circuits that each need to be checked.

Solving the TSP Without Brute Force

Idea: At each stage in your tour, choose the closest vertex that you have not visited yet.

This is called the **Nearest-Neighbor Algorithm (NNA).**

This spreadsheet shows what happens when Willy uses the NNA to construct a Hamilton circuit (with Sydney as the reference vertex).

The Nearest-Neighbor Algorithm

The result: The Nearest-Neighbor algorithm, using Sydney as the reference vertex, yields the Hamilton circuit

SY $\rightarrow$ CN $\rightarrow$ ML $\rightarrow$ HO $\rightarrow$ AD $\rightarrow$ AS $\rightarrow$ UL $\rightarrow$ BM $\rightarrow$ KU $\rightarrow$ DA $\rightarrow$ MI $\rightarrow$ CS $\rightarrow$ MK $\rightarrow$ BR $\rightarrow$ AL $\rightarrow$ PE $\rightarrow$ SY

whose total weight is **21,049 km**.

A randomly chosen Hamilton circuit would have averaged **40,680 km**, so this is pretty good.

The Nearest-Neighbor Algorithm

The result: The Nearest-Neighbor algorithm, using Sydney as the reference vertex, yields the Hamilton circuit

SY $\rightarrow$ CN $\rightarrow$ ML $\rightarrow$ HO $\rightarrow$ AD $\rightarrow$ AS $\rightarrow$ UL $\rightarrow$ BM $\rightarrow$ KU $\rightarrow$ DA $\rightarrow$ MI $\rightarrow$ CS $\rightarrow$ MK $\rightarrow$ BR $\rightarrow$ AL $\rightarrow$ PE $\rightarrow$ SY

whose total weight is **21,049 km**.

A randomly chosen Hamilton circuit would have averaged **40,680 km**, so this is pretty good.

But can Willy do better?

The Repetitive Nearest-Neighbor Algorithm

Observation: Willy can use **any city** as the reference vertex!

That is, Willy can execute the Nearest-Neighbor Algorithm sixteen times, using each city once as the reference vertex.

Then, he can pick the Hamilton circuit with the lowest total weight of these sixteen.

This is called the **Repetitive Nearest-Neighbor Algorithm (RNNA)**.

The Repetitive Nearest-Neighbor Algorithm

Ref. vertex	Hamilton circuit	Weight	
AD	AD,ML,HO,CN,SY,BR,MK,CS,MI,AS,UL,BM,KU,DA,PE,AL,AD	18543	
AL	AL,PE,BM,KU,DA,AS,UL,AD,ML,HO,CN,SY,BR,MK,CS,MI,AL	19795	
AS	AS,UL,BM,KU,DA,MI,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,AS	18459	
BR	BR,MK,CS,MI,AS,UL,BM,KU,DA,AD,ML,HO,CN,SY,AL,PE,BR	22113	
BM	BM,KU,DA,AS,UL,AD,ML,HO,CN,SY,BR,MK,CS,MI,PE,AL,BM	19148	
CS	CS,MK,BR,SY,CN,ML,HO,AD,AS,UL,BM,KU,DA,MI,PE,AL,CS	22936	
CN	CN,SY,ML,HO,AD,AS,UL,BM,KU,DA,MI,CS,MK,BR,AL,PE,CN	21149	
DA	DA,KU,BM,UL,AS,MI,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,DA	18543	
HO	HO,ML,CN,SY,BR,MK,CS,MI,AS,UL,BM,KU,DA,AD,AL,PE,HO	20141	
KU	KU,DA,AS,UL,BM,MI,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,KU	18785	
MK	MK,CS,MI,AS,UL,BM,KU,DA,AD,ML,HO,CN,SY,BR,AL,PE,MK	23255	
ML	ML,HO,CN,SY,BR,MK,CS,MI,AS,UL,BM,KU,DA,AD,AL,PE,ML	20141	
MI	MI,AS,UL,BM,KU,DA,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,MI	20877	
PE	PE,AL,BM,KU,DA,AS,UL,AD,ML,HO,CN,SY,BR,MK,CS,MI,PE	19148	
SY	SY,CN,ML,HO,AD,AS,UL,BM,KU,DA,MI,CS,MK,BR,AL,PE,SY	21049	(NNA)
UL	UL,AS,MI,CS,MK,BR,SY,CN,ML,HO,AD,BM,KU,DA,PE,AL,UL	20763	

The Repetitive Nearest-Neighbor Algorithm

Ref. vertex	Hamilton circuit	Weight	
AD	AD,ML,HO,CN,SY,BR,MK,CS,MI,AS,UL,BM,KU,DA,PE,AL,AD	18543	
AL	AL,PE,BM,KU,DA,AS,UL,AD,ML,HO,CN,SY,BR,MK,CS,MI,AL	19795	
AS	AS,UL,BM,KU,DA,MI,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,AS	18459	**(best)**
BR	BR,MK,CS,MI,AS,UL,BM,KU,DA,AD,ML,HO,CN,SY,AL,PE,BR	22113	
BM	BM,KU,DA,AS,UL,AD,ML,HO,CN,SY,BR,MK,CS,MI,PE,AL,BM	19148	
CS	CS,MK,BR,SY,CN,ML,HO,AD,AS,UL,BM,KU,DA,MI,PE,AL,CS	22936	
CN	CN,SY,ML,HO,AD,AS,UL,BM,KU,DA,MI,CS,MK,BR,AL,PE,CN	21149	
DA	DA,KU,BM,UL,AS,MI,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,DA	18543	
HO	HO,ML,CN,SY,BR,MK,CS,MI,AS,UL,BM,KU,DA,AD,AL,PE,HO	20141	
KU	KU,DA,AS,UL,BM,MI,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,KU	18785	
MK	MK,CS,MI,AS,UL,BM,KU,DA,AD,ML,HO,CN,SY,BR,AL,PE,MK	23255	
ML	ML,HO,CN,SY,BR,MK,CS,MI,AS,UL,BM,KU,DA,AD,AL,PE,ML	20141	
MI	MI,AS,UL,BM,KU,DA,CS,MK,BR,SY,CN,ML,HO,AD,AL,PE,MI	20877	
PE	PE,AL,BM,KU,DA,AS,UL,AD,ML,HO,CN,SY,BR,MK,CS,MI,PE	19148	
SY	SY,CN,ML,HO,AD,AS,UL,BM,KU,DA,MI,CS,MK,BR,AL,PE,SY	21049	(NNA)
UL	UL,AS,MI,CS,MK,BR,SY,CN,ML,HO,AD,BM,KU,DA,PE,AL,UL	20763	

The Repetitive Nearest-Neighbor Algorithm

Apparently, using Alice Springs (AS) as the reference vertex yields the best Hamilton circuit so far, namely

AS $\rightarrow$ UL $\rightarrow$ BM $\rightarrow$ KU $\rightarrow$ DA $\rightarrow$ MI $\rightarrow$ CS $\rightarrow$ MK $\rightarrow$ BR $\rightarrow$ SY $\rightarrow$ CN $\rightarrow$ ML $\rightarrow$ HO $\rightarrow$ AD $\rightarrow$ AL $\rightarrow$ PE $\rightarrow$ AS

The Repetitive Nearest-Neighbor Algorithm

Apparently, using Alice Springs (AS) as the reference vertex yields the best Hamilton circuit so far, namely

AS $\rightarrow$ UL $\rightarrow$ BM $\rightarrow$ KU $\rightarrow$ DA $\rightarrow$ MI $\rightarrow$ CS $\rightarrow$ MK $\rightarrow$ BR
$\rightarrow$ SY $\rightarrow$ CN $\rightarrow$ ML $\rightarrow$ HO $\rightarrow$ AD $\rightarrow$ AL $\rightarrow$ PE $\rightarrow$ AS

Remember: Willy can still start anywhere he wants!
For instance,

SY $\rightarrow$ CN $\rightarrow$ ML $\rightarrow$ HO $\rightarrow$ AD $\rightarrow$ AL $\rightarrow$ PE $\rightarrow$ AS
$\rightarrow$ UL $\rightarrow$ BM $\rightarrow$ KU $\rightarrow$ DA $\rightarrow$ MI $\rightarrow$ CS $\rightarrow$ MK $\rightarrow$ BR $\rightarrow$ SY

represents the same Hamilton circuit.

The Repetitive Nearest-Neighbor Algorithm

Randomly chosen Hamilton circuit:	40,680 km
Hamilton circuit using NNA/Sydney:	21,049 km
Hamilton circuit using RNNA:	**18,459 km**

The Repetitive Nearest-Neighbor Algorithm

Randomly chosen Hamilton circuit:	40,680 km
Hamilton circuit using NNA/Sydney:	21,049 km
Hamilton circuit using RNNA:	**18,459 km**

- In general, there's no way of knowing in advance which reference vertex will yield the best result.

The Repetitive Nearest-Neighbor Algorithm

Randomly chosen Hamilton circuit:	40,680 km
Hamilton circuit using NNA/Sydney:	21,049 km
Hamilton circuit using RNNA:	**18,459 km**

- In general, there's no way of knowing in advance which reference vertex will yield the best result.
- This algorithm is still **efficient**, but ...

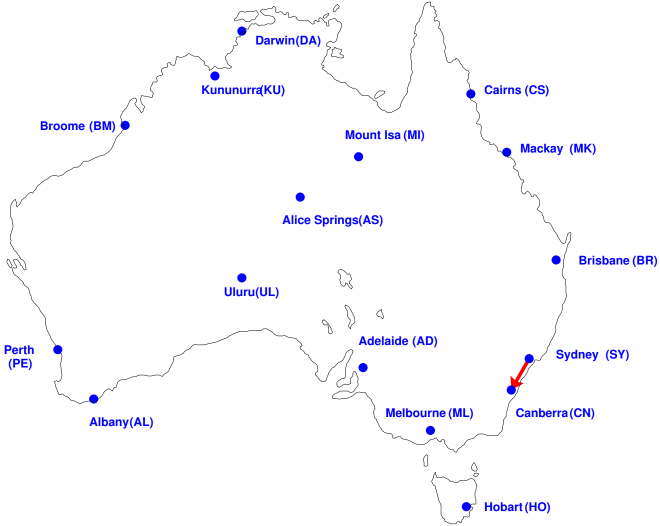
The Repetitive Nearest-Neighbor Algorithm

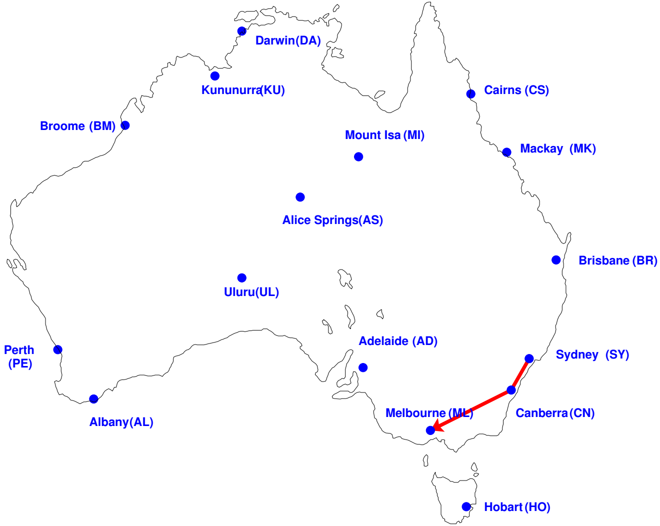
Randomly chosen Hamilton circuit:	40,680 km
Hamilton circuit using NNA/Sydney:	21,049 km
Hamilton circuit using RNNA:	**18,459 km**

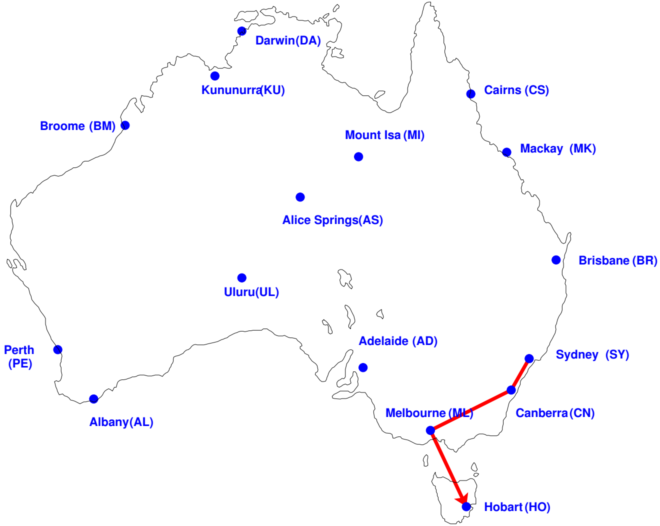
- In general, there's no way of knowing in advance which reference vertex will yield the best result.
- This algorithm is still **efficient**, but . . .
- Is it **optimal?** That is, Can Willy do even better?

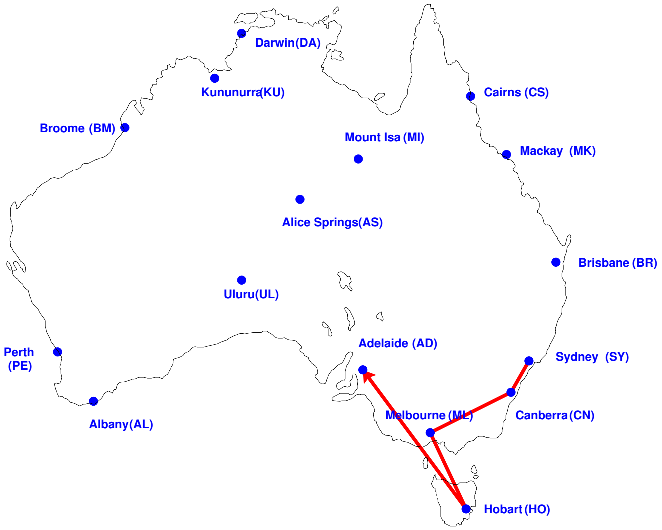
The Repetitive Nearest-Neighbor Algorithm

If we look at the map (warning: not quite to scale!) it becomes clear that the RNNA has not produced an optimal Hamilton circuit.

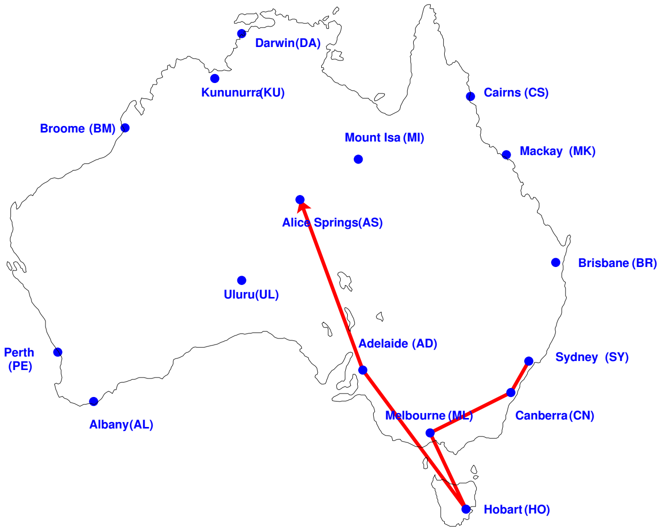

Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth (PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

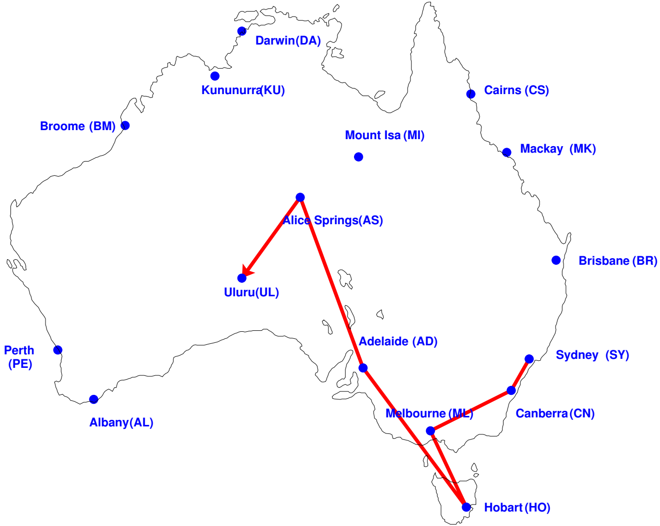

Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth (PE)
Sydney (SY)
Canberra (CN)
Melbourne (ML)
Albany(AL)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)

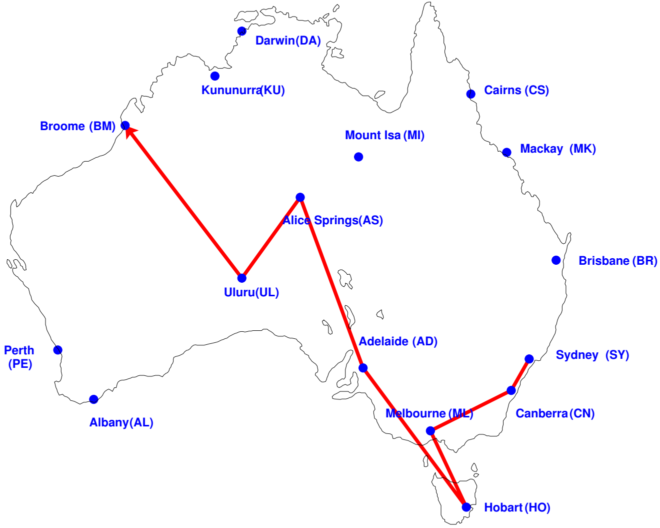


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth (PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Sydney (SY)
Perth
(PE)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

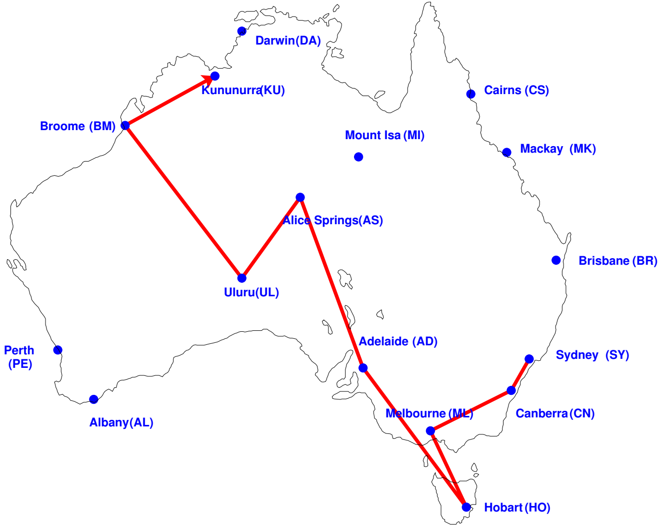


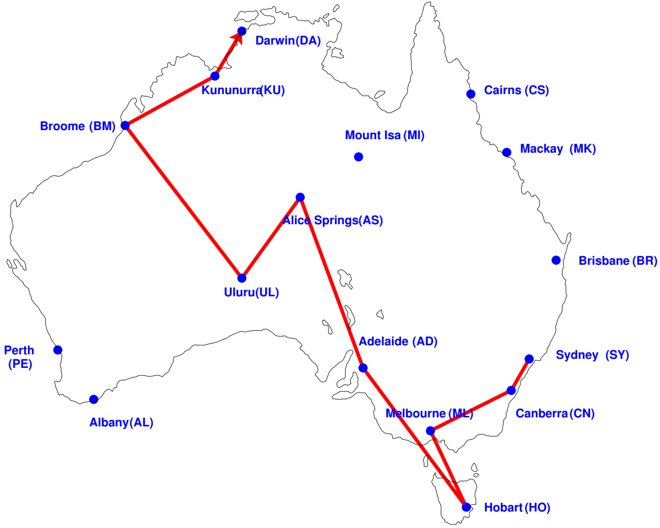
Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

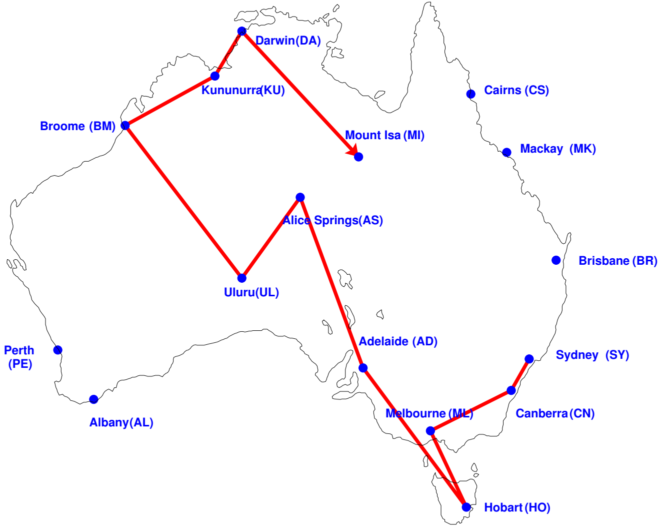


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)

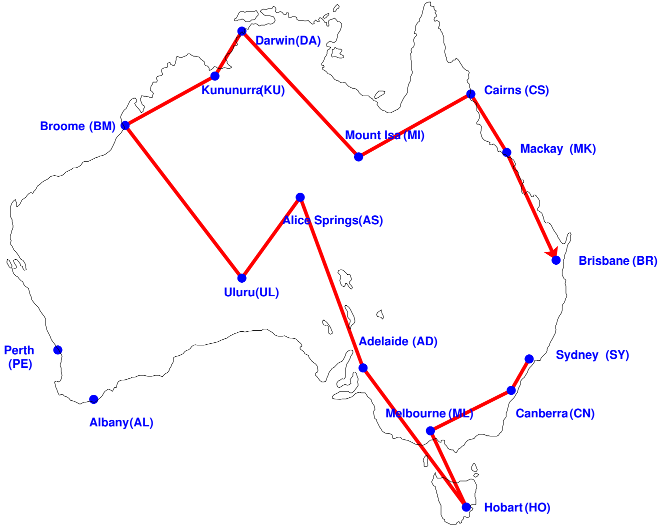
Oops.


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra(CN)
Hobart(HO)

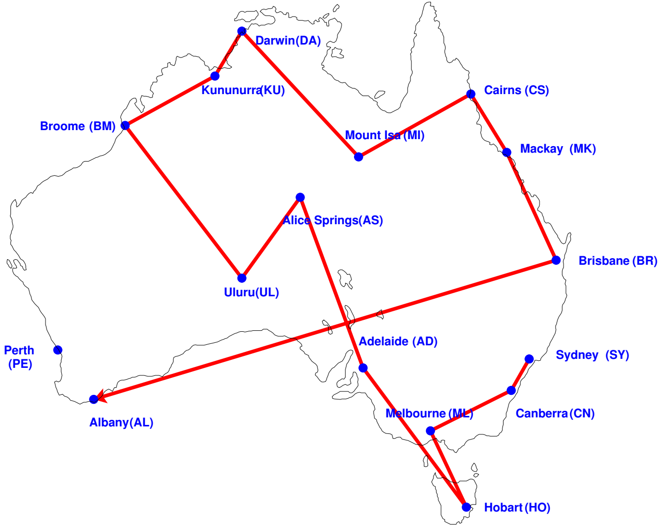


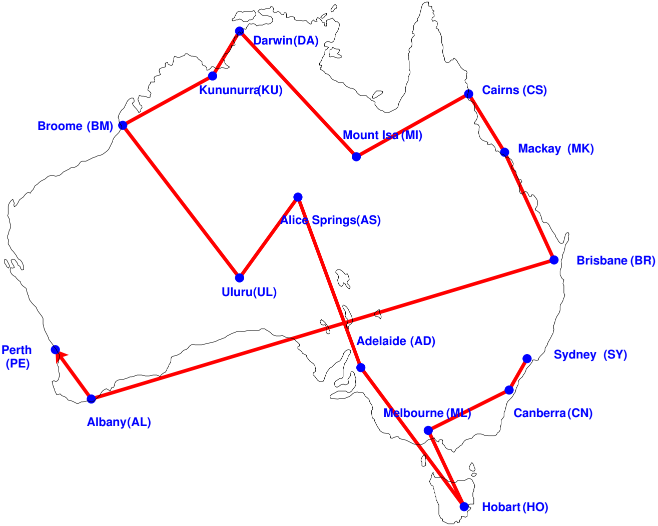
Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

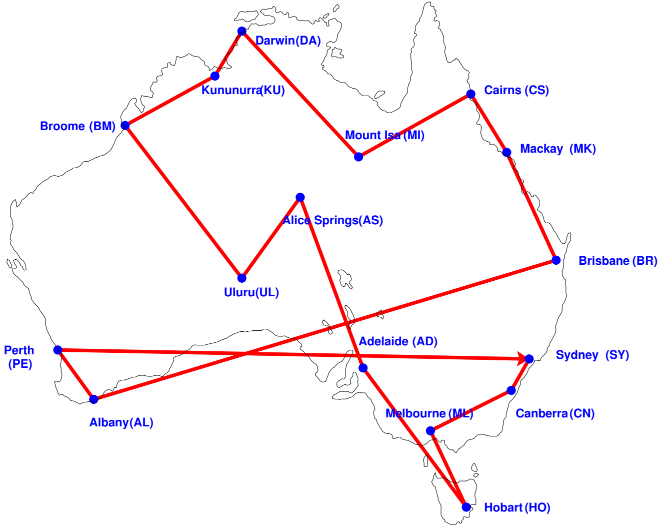


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

Now the algorithm is stuck — some very expensive edges are required to complete the Hamilton circuit.

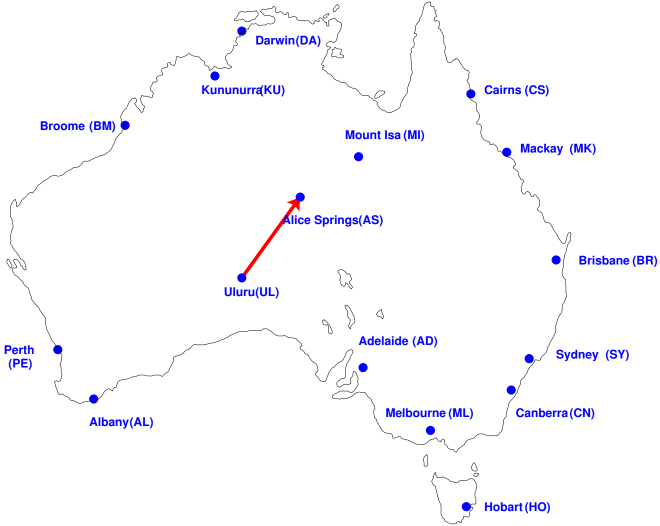

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

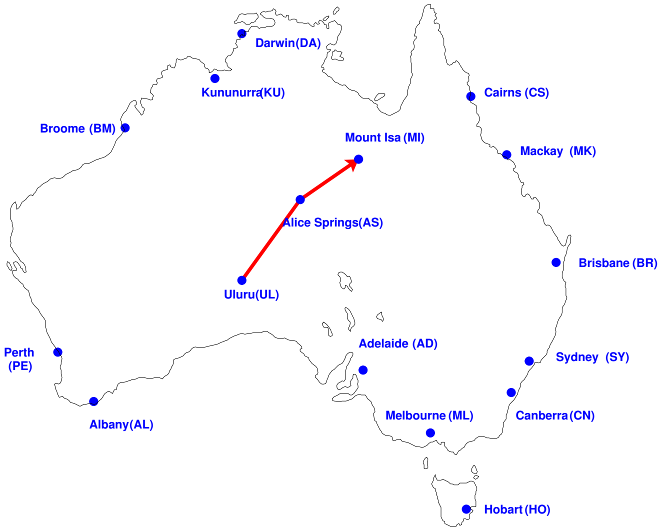

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

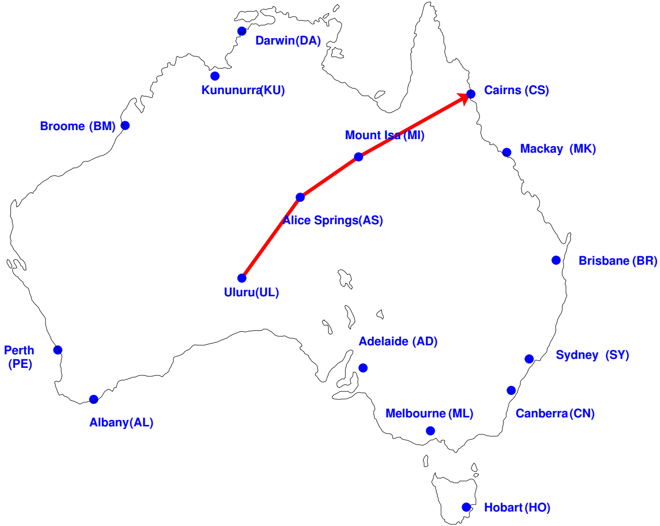

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

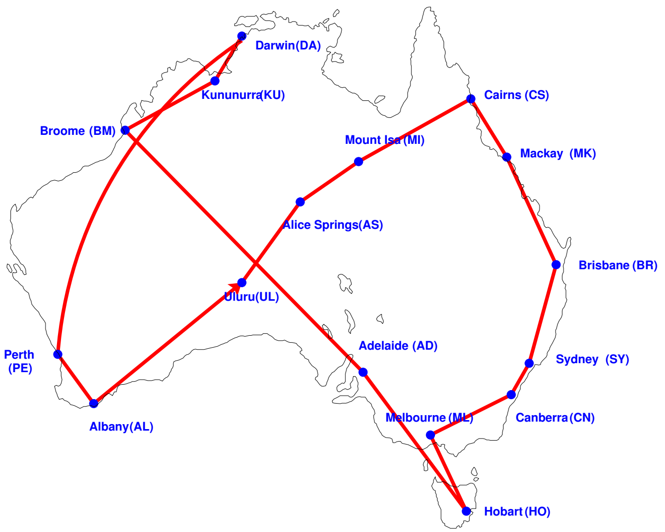
The Repetitive Nearest-Neighbor Algorithm

Starting from Uluru would have created a different problem.


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

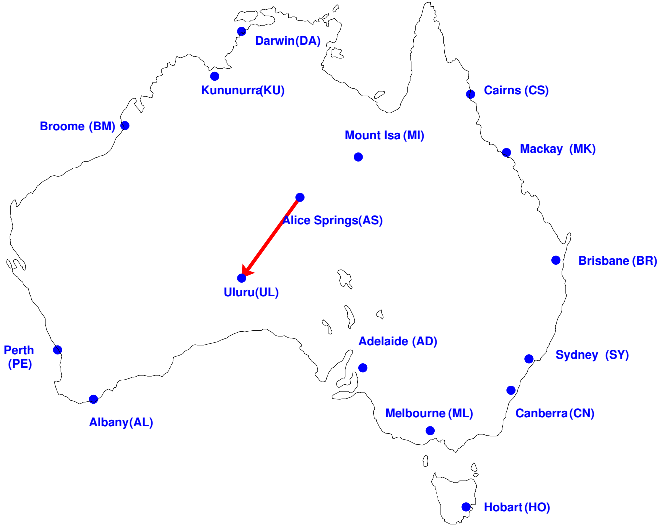

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

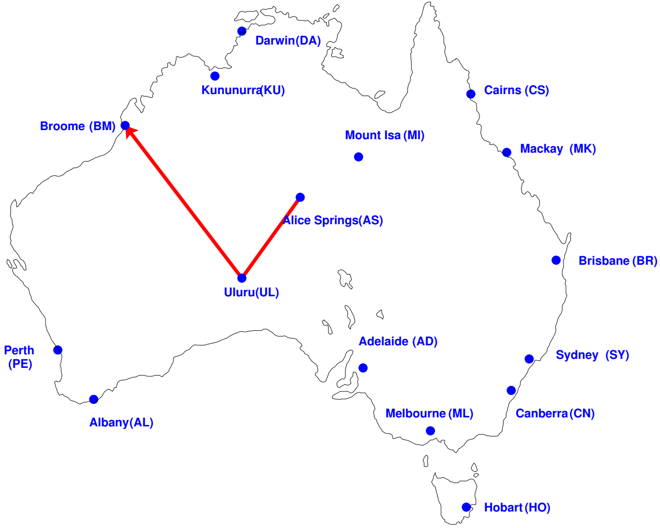

Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

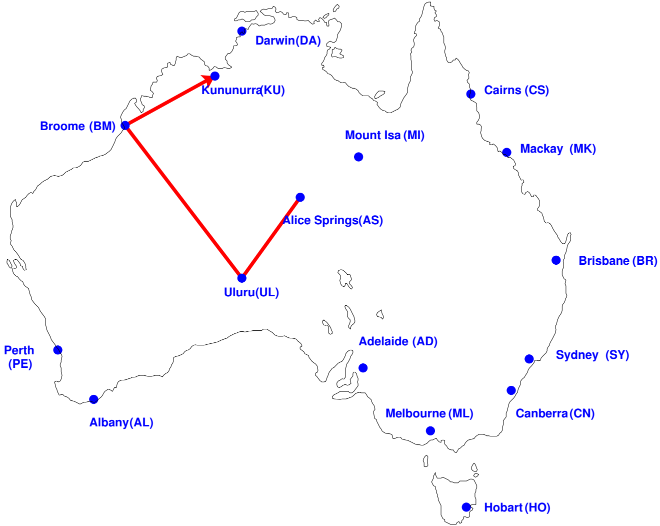

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

The Repetitive Nearest-Neighbor Algorithm

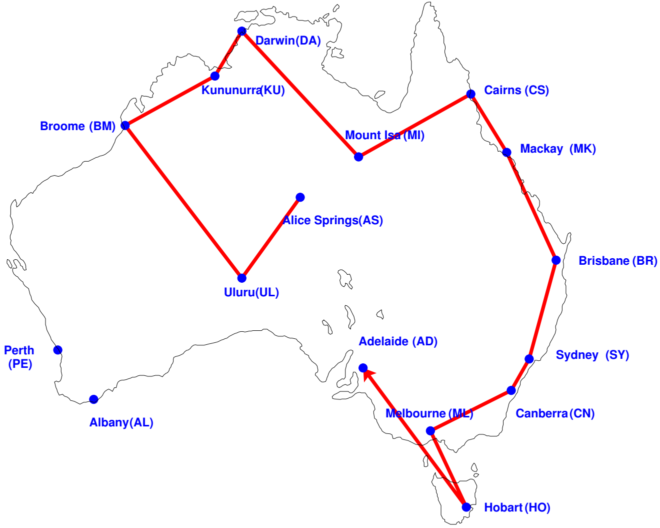
Starting from Alice Springs would have created a different problem (but a less harmful one).

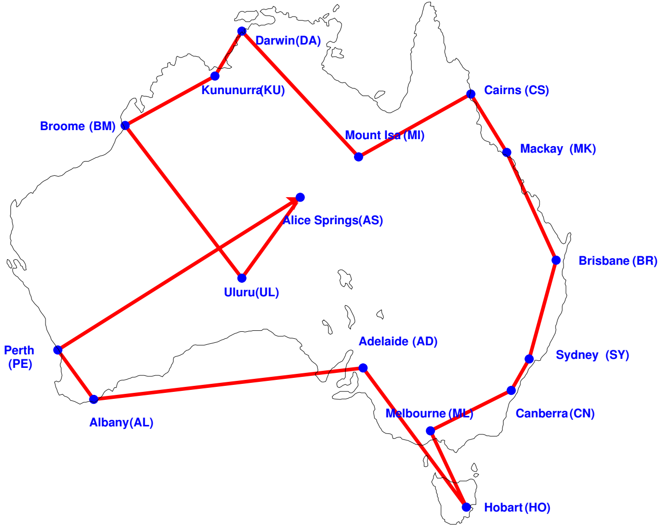

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany (AL)
Hobart (HO)

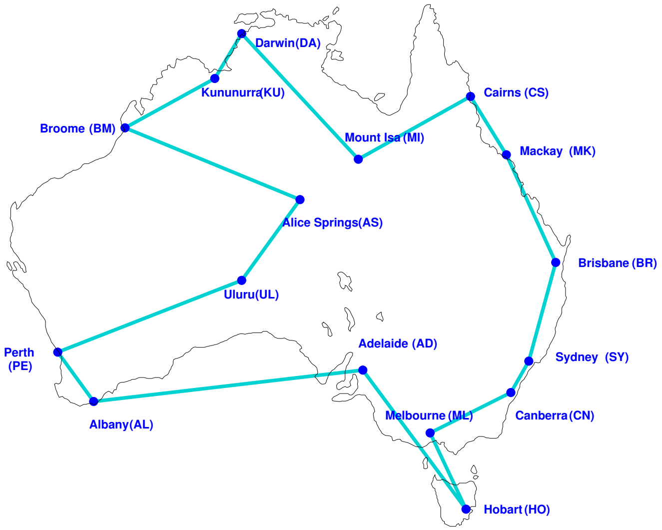


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

It is easy for a human to look at this Hamilton circuit and find a way to improve it.


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

It is easy for a human to look at this Hamilton circuit and find a way to improve it.

But how do you make such improvements **part of the algorithm?**

Does starting How about

The Cheapest-Link Algorithm

Idea: **Start in the middle.**

The Cheapest-Link Algorithm

Idea: **Start in the middle.**

- Find the single edge that would be cheapest to add.

The Cheapest-Link Algorithm

Idea: **Start in the middle.**

- Find the single edge that would be cheapest to add.
- Keep doing this until you have a Hamilton circuit.

The Cheapest-Link Algorithm

Idea: **Start in the middle.**

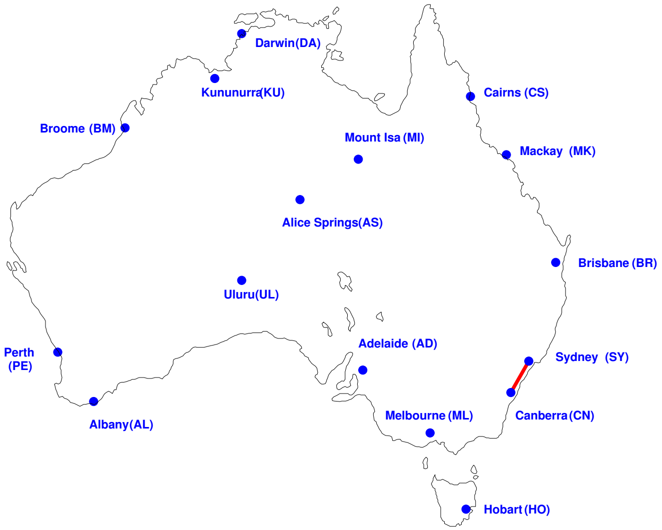
- Find the single edge that would be cheapest to add.
- Keep doing this until you have a Hamilton circuit.
- Make sure you add exactly two edges at each vertex.

The Cheapest-Link Algorithm

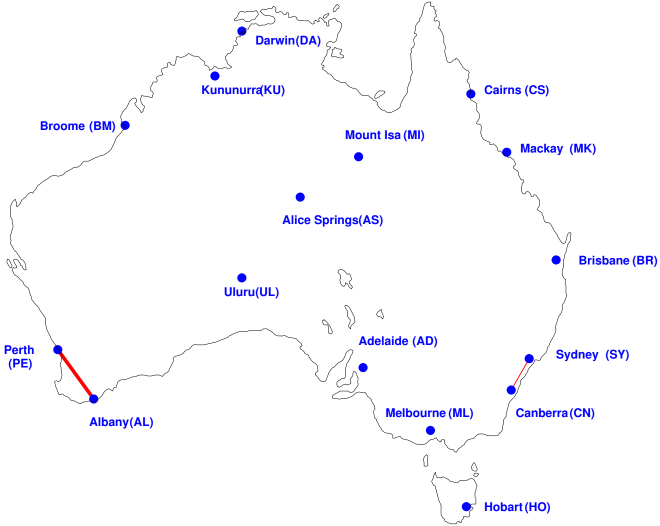
Idea: **Start in the middle.**

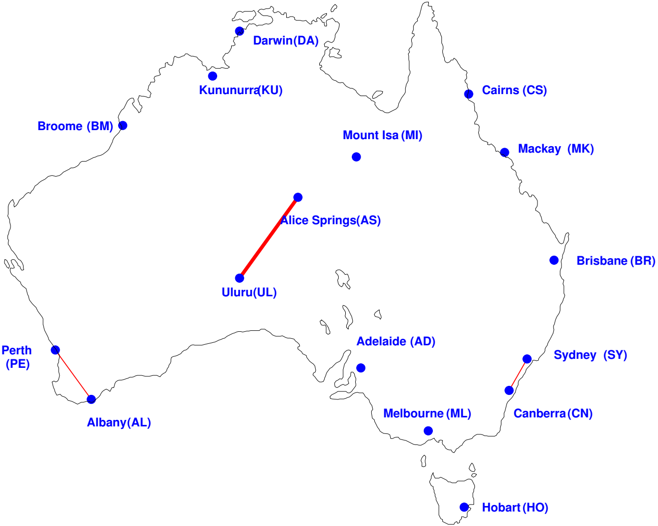
- Find the single edge that would be cheapest to add.
- Keep doing this until you have a Hamilton circuit.
- Make sure you add exactly two edges at each vertex.

This is called the **Cheapest-Link Algorithm, or CLA**.
Here is an example.

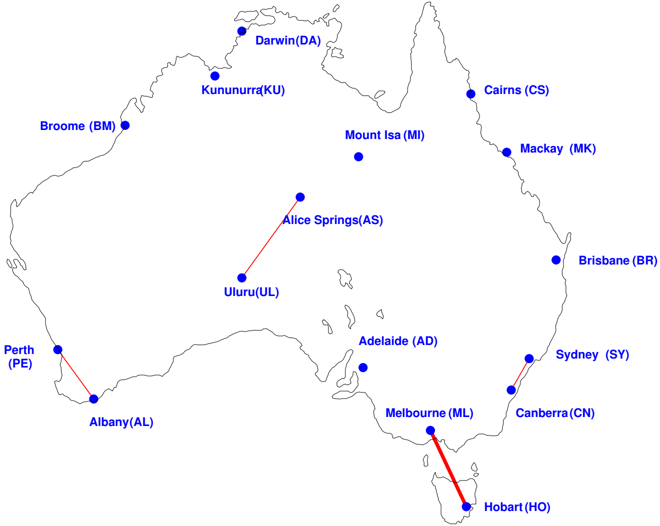


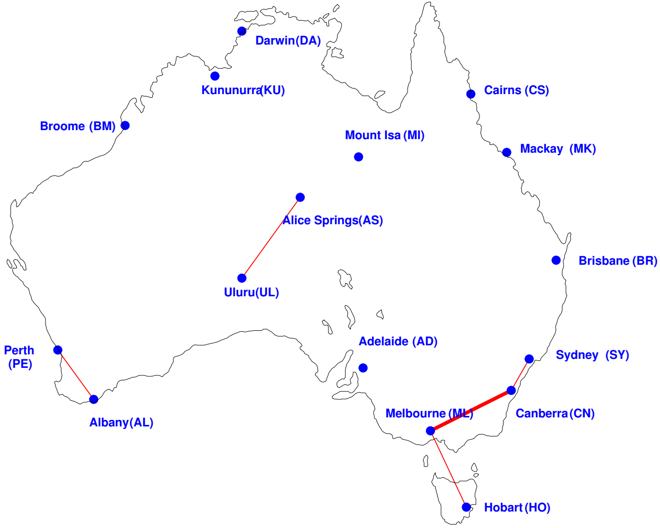
Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)

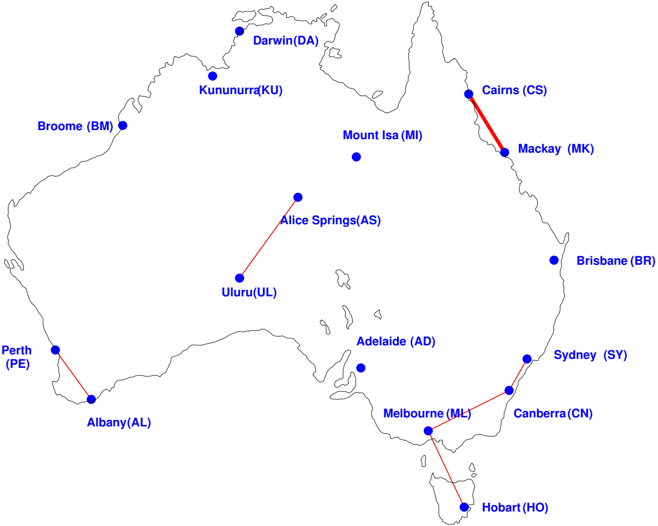

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)



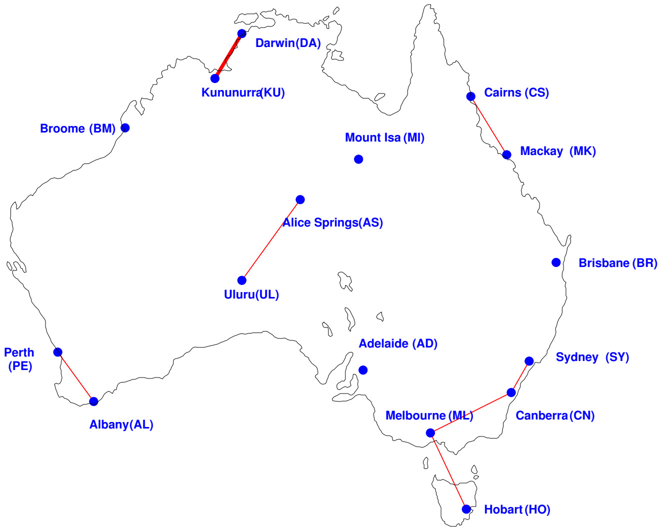
Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

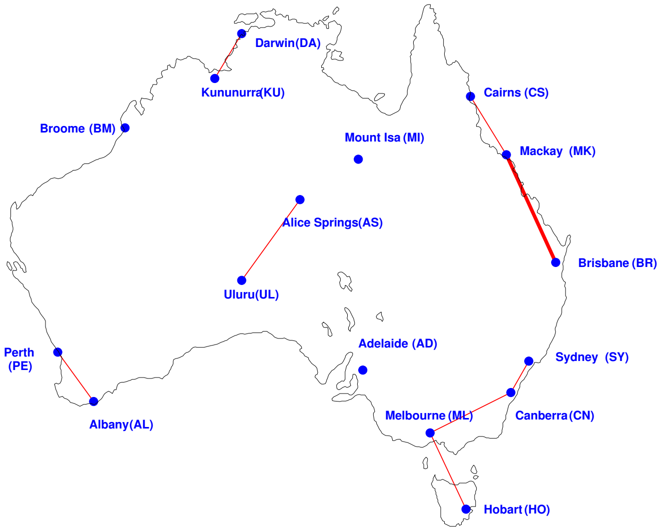

Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth (PE)
Sydney (SY)
Canberra (CN)
Melbourne (ML)
Albany(AL)
Hobart (HO)

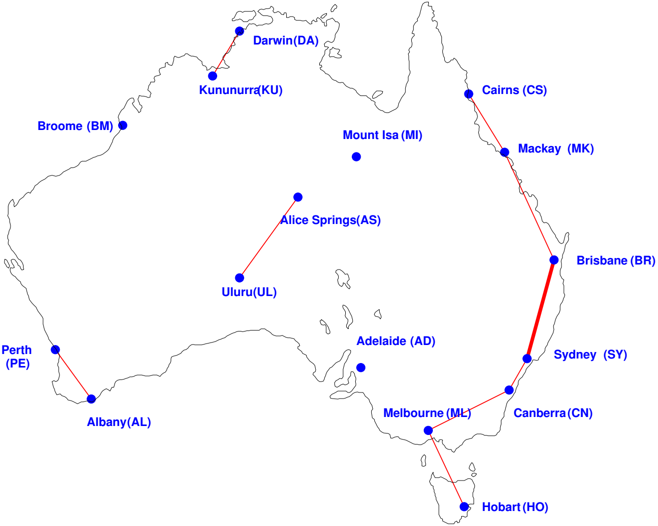

Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)



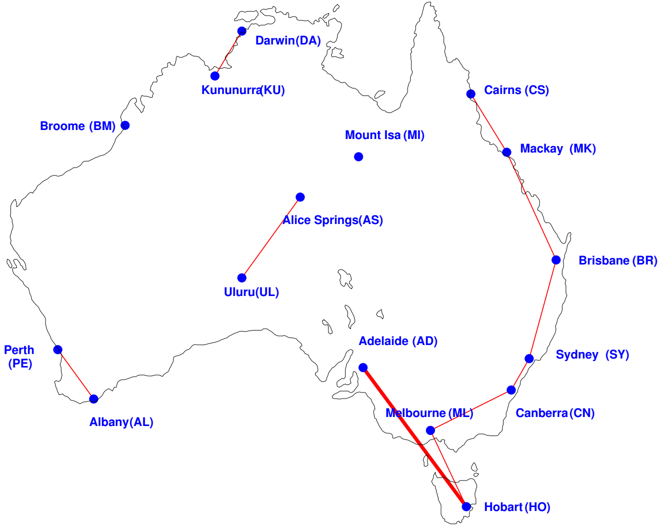
Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth (PE)
Adelaide (AD)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)

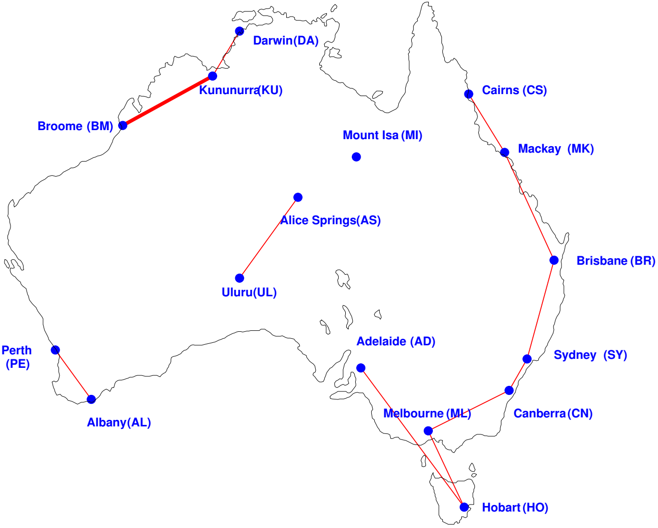

Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Broome (BM)
Mount Isa (MI)
Cairns (CS)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

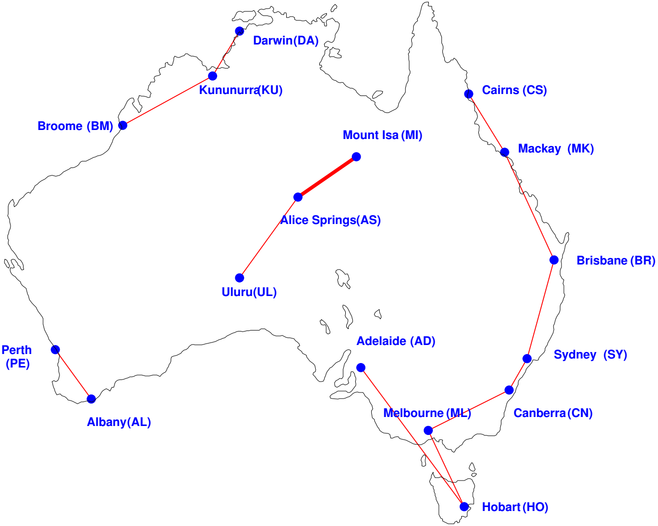


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)

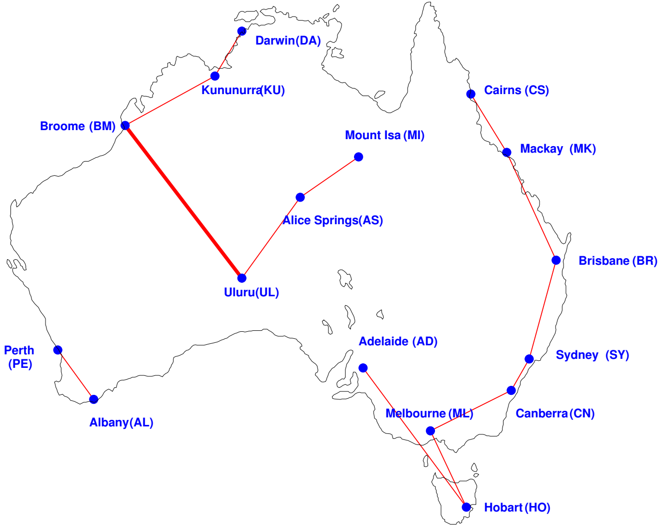

Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

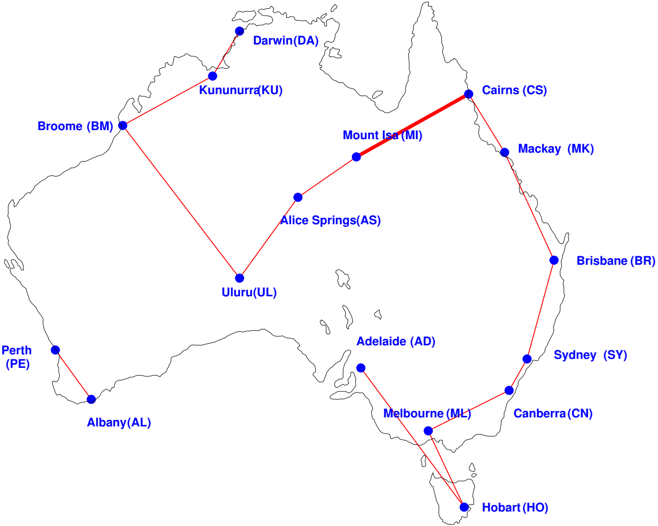


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth (PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

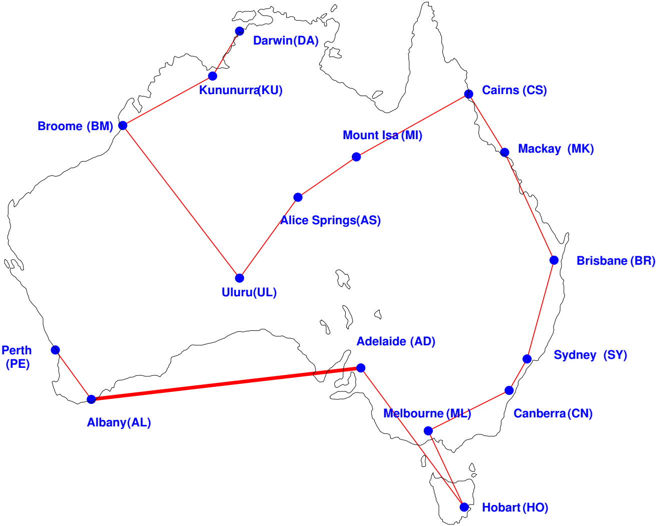


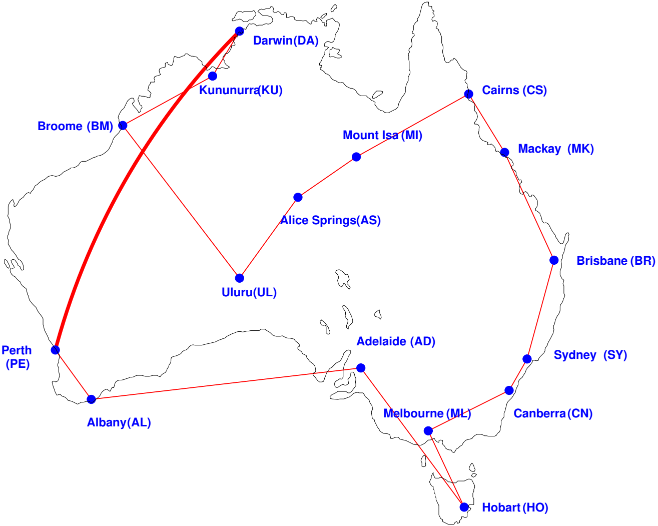
Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Broome (BM)
Mount Isa (MI)
Cairns (CS)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Perth
(PE)
Adelaide (AD)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)



Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Melbourne (ML)
Canberra (CN)
Albany(AL)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Cairns (CS)
Broome (BM)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)


Darwin(DA)
Kununurra(KU)
Broome (BM)
Cairns (CS)
Mount Isa (MI)
Mackay (MK)
Alice Springs(AS)
Brisbane (BR)
Uluru(UL)
Adelaide (AD)
Perth
(PE)
Sydney (SY)
Albany(AL)
Melbourne (ML)
Canberra (CN)
Hobart (HO)

Comparing Algorithms

Randomly chosen Hamilton circuit:	40,680 km
Hamilton circuit using NNA/Sydney:	21,049 km
Hamilton circuit using RNNA:	**18,459 km**
Hamilton circuit using CLA:	**18,543 km**

Comparing Algorithms

Randomly chosen Hamilton circuit:	40,680 km
Hamilton circuit using NNA/Sydney:	21,049 km
Hamilton circuit using RNNA:	**18,459 km**
Hamilton circuit using CLA:	**18,543 km**

This didn't help in this case. But it might help in a different example (next time).

Comparing Algorithms

Randomly chosen Hamilton circuit:	40,680 km
Hamilton circuit using NNA/Sydney:	21,049 km
Hamilton circuit using RNNA:	**18,459 km**
Hamilton circuit using CLA:	**18,543 km**

This didn't help in this case. But it might help in a different example (next time).

Can Willy do even better?

The Bad News

There is no known algorithm to solve the TSP that is both optimal and efficient.

The Bad News

There is no known algorithm to solve the TSP that is both optimal and efficient.

- Brute-force is optimal but not efficient.
- NNA, RNNA, and CLA are efficient but not optimal.