The Traveling Salesman Problem

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

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

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

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).

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

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

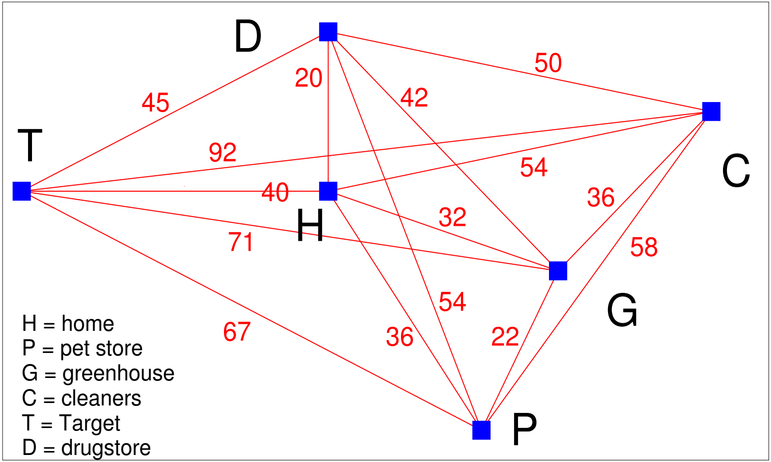
Example: The Traveling Saleswitch Problem

Example: Sabrina has the following list of errands:

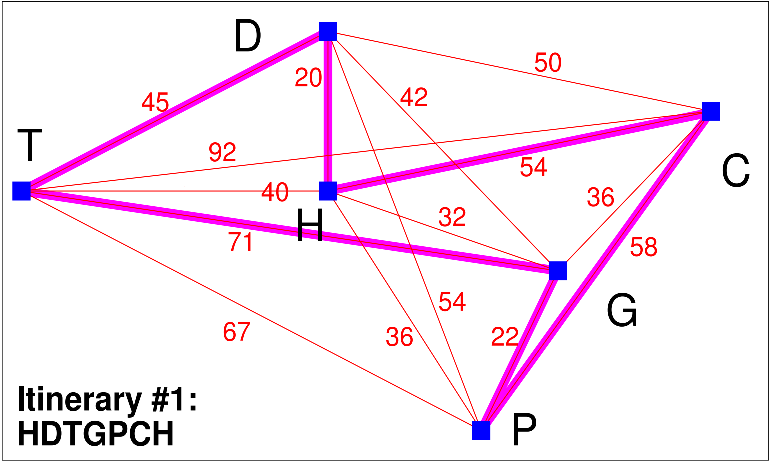
- **P**et store (the black cat needs a new litterbox)
- **G**reenhouse (replenish supply of deadly nightshade)
- **C**leaners (pick up black hat)
- **D**rugstore (eye of newt, wing of bat, toothpaste)
- **T**arget (weekly special on cauldrons)

In ~~witch~~ which order should she do these errands in order to minimize the time spent on her broom? (Of course, she wants to start from home and end at home.)

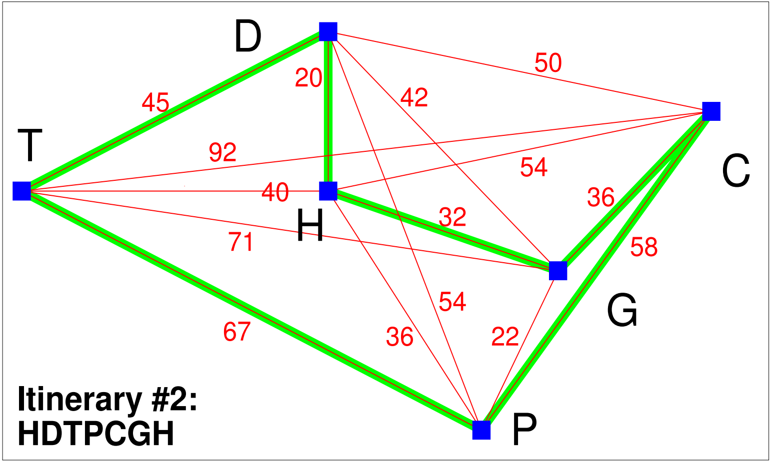
The Traveling Saleswitch Problem



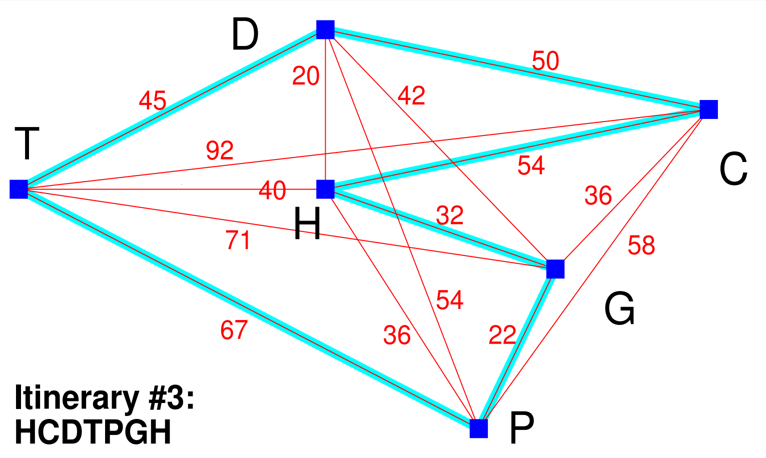
The Traveling Saleswitch Problem



The Traveling Saleswitch Problem



The Traveling Saleswitch Problem



The Traveling Saleswitch Problem

Times between each pair of locations (minutes):

	H	P	G	C	D	T
Home (H)	0	36	32	54	20	40
Pet store (P)	36	0	22	58	54	67
Greenhouse (G)	32	22	0	36	42	71
Cleaners (C)	54	58	36	0	50	92
Drugstore (D)	20	54	42	50	0	45
Target (T)	40	67	71	92	45	0

Possible Hamilton Circuits

	H	P	G	C	D	T
Home (H)	0	36	32	54	**20**	40
Pet store (P)	36	0	22	**58**	54	67
Greenhouse (G)	32	**22**	0	36	42	71
Cleaners (C)	**54**	58	36	0	50	92
Drugstore (D)	20	54	42	50	0	**45**
Target (T)	40	67	**71**	92	45	0

$$\text{Weight(HDTGPCH)} = 20 + 45 + 71 + 22 + 58 + 54 = \mathbf{270}$$

Possible Hamilton Circuits

	H	P	G	C	D	T
Home (H)	0	36	32	54	**20**	40
Pet store (P)	36	0	22	**58**	54	67
Greenhouse (G)	**32**	22	0	36	42	71
Cleaners (C)	54	58	**36**	0	50	92
Drugstore (D)	20	54	42	50	0	**45**
Target (T)	40	**67**	71	92	45	0

$$\text{Weight(HDTGPCH)} = 20 + 45 + 71 + 22 + 58 + 54 = \mathbf{270}$$
$$\text{Weight(HDTPCGH)} = 20 + 45 + 67 + 58 + 36 + 32 = \mathbf{258}$$

Possible Hamilton Circuits

	H	P	G	C	D	T
Home (H)	0	36	32	**54**	20	40
Pet store (P)	36	0	**22**	58	54	67
Greenhouse (G)	**32**	22	0	36	42	71
Cleaners (C)	54	58	36	0	**50**	92
Drugstore (D)	20	54	42	50	0	**45**
Target (T)	40	**67**	71	92	45	0

$$\text{Weight(HDTGPCH)} = 20 + 45 + 71 + 22 + 58 + 54 = \mathbf{270}$$
$$\text{Weight(HDTPCGH)} = 20 + 45 + 67 + 58 + 36 + 32 = \mathbf{258}$$
$$\text{Weight(HCDTPGH)} = 54 + 50 + 45 + 67 + 22 + 32 = \mathbf{270}$$

Possible Hamilton Circuits

So far, we know that Itinerary #2 (HDTPCGH) is more efficient than either Itinerary #1 (HDTGPCH) or Itinerary #3 (HCDTPGH).

But how do we know whether it is the most efficient of all possible itineraries?

Possible Hamilton Circuits

So far, we know that Itinerary #2 (HDTPCGH) is more efficient than either Itinerary #1 (HDTGPCH) or Itinerary #3 (HCDTPGH).

But how do we know whether it is the most efficient of all possible itineraries?

- We could list all possible itineraries starting and ending at H.
- For each itinerary, sum up the weights on its edges.
- Then, choose the itinerary of smallest total weight.
- The number of Hamilton circuits in K_6 is $5! = 120$.

Possible Hamilton Circuits (Page 1)

Hamilton circuit	Weight	Hamilton circuit	Weight
H,C,D,G,P,T,H	275	H,C,P,D,G,T,H	319
H,C,D,G,T,P,H	320	H,C,P,D,T,G,H	314
H,C,D,P,G,T,H	291	H,C,P,G,D,T,H	261
H,C,D,P,T,G,H	328	H,C,P,G,T,D,H	270
H,C,D,T,G,P,H	278	H,C,P,T,D,G,H	298
H,C,D,T,P,G,H	270	H,C,P,T,G,D,H	312
H,C,G,D,P,T,H	293	H,C,T,D,G,P,H	291
H,C,G,D,T,P,H	280	H,C,T,D,P,G,H	299
H,C,G,P,D,T,H	251	H,C,T,G,D,P,H	349
H,C,G,P,T,D,H	244	H,C,T,G,P,D,H	313
H,C,G,T,D,P,H	296	H,C,T,P,D,G,H	341
H,C,G,T,P,D,H	302	H,C,T,P,G,D,H	297

Possible Hamilton Circuits (Page 2)

Hamilton circuit	Weight	Hamilton circuit	Weight
H,D,C,G,P,T,H	235	H,D,P,C,G,T,H	279
H,D,C,G,T,P,H	280	H,D,P,C,T,G,H	327
H,D,C,P,G,T,H	261	H,D,P,G,C,T,H	264
H,D,C,P,T,G,H	298	H,D,P,G,T,C,H	313
H,D,C,T,G,P,H	291	H,D,P,T,C,G,H	301
H,D,C,T,P,G,H	283	H,D,P,T,G,C,H	302
H,D,G,C,P,T,H	263	H,D,T,C,G,P,H	251
H,D,G,C,T,P,H	293	H,D,T,C,P,G,H	269
H,D,G,P,C,T,H	274	H,D,T,G,C,P,H	266
H,D,G,P,T,C,H	297	H,D,T,G,P,C,H	270
H,D,G,T,C,P,H	319	H,D,T,P,C,G,H	258
H,D,G,T,P,C,H	312	H,D,T,P,G,C,H	244

Possible Hamilton Circuits (Page 3)

Hamilton circuit	Weight	Hamilton circuit	Weight
H,G,C,D,P,T,H	279	H,G,P,C,D,T,H	247
H,G,C,D,T,P,H	266	H,G,P,C,T,D,H	269
H,G,C,P,D,T,H	265	H,G,P,D,C,T,H	290
H,G,C,P,T,D,H	258	H,G,P,D,T,C,H	299
H,G,C,T,D,P,H	295	H,G,P,T,C,D,H	283
H,G,C,T,P,D,H	301	H,G,P,T,D,C,H	270
H,G,D,C,P,T,H	289	H,G,T,C,D,P,H	335
H,G,D,C,T,P,H	319	H,G,T,C,P,D,H	327
H,G,D,P,C,T,H	318	H,G,T,D,C,P,H	292
H,G,D,P,T,C,H	341	H,G,T,D,P,C,H	314
H,G,D,T,C,P,H	305	H,G,T,P,C,D,H	298
H,G,D,T,P,C,H	298	H,G,T,P,D,C,H	328

Possible Hamilton Circuits (Page 4)

Hamilton circuit	Weight	Hamilton circuit	Weight
H,P,C,D,G,T,H	297	H,P,G,C,D,T,H	229
H,P,C,D,T,G,H	292	H,P,G,C,T,D,H	251
H,P,C,G,D,T,H	257	H,P,G,D,C,T,H	282
H,P,C,G,T,D,H	266	H,P,G,D,T,C,H	291
H,P,C,T,D,G,H	305	H,P,G,T,C,D,H	291
H,P,C,T,G,D,H	319	H,P,G,T,D,C,H	278
H,P,D,C,G,T,H	287	H,P,T,C,D,G,H	319
H,P,D,C,T,G,H	335	H,P,T,C,G,D,H	293
H,P,D,G,C,T,H	300	H,P,T,D,C,G,H	266
H,P,D,G,T,C,H	349	H,P,T,D,G,C,H	280
H,P,D,T,C,G,H	295	H,P,T,G,C,D,H	280
H,P,D,T,G,C,H	296	H,P,T,G,D,C,H	320

Possible Hamilton Circuits (Page 5)

Hamilton circuit	Weight	Hamilton circuit	Weight
H,T,C,D,G,P,H	282	H,T,G,C,D,P,H	287
H,T,C,D,P,G,H	290	H,T,G,C,P,D,H	279
H,T,C,G,D,P,H	300	H,T,G,D,C,P,H	297
H,T,C,G,P,D,H	264	H,T,G,D,P,C,H	319
H,T,C,P,D,G,H	318	H,T,G,P,C,D,H	261
H,T,C,P,G,D,H	274	H,T,G,P,D,C,H	291
H,T,D,C,G,P,H	229	H,T,P,C,D,G,H	289
H,T,D,C,P,G,H	247	H,T,P,C,G,D,H	263
H,T,D,G,C,P,H	257	H,T,P,D,C,G,H	279
H,T,D,G,P,C,H	261	H,T,P,D,G,C,H	293
H,T,D,P,C,G,H	265	H,T,P,G,C,D,H	235
H,T,D,P,G,C,H	251	H,T,P,G,D,C,H	275

Solving the TSP by Brute Force

What we have just done is the **Brute-Force Algorithm**:

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

Solving the TSP by Brute Force

What we have just done is the **Brute-Force Algorithm**:

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

- The Brute-Force Algorithm is **optimal**: it is guaranteed to find a solution.
- OTOH, the algorithm is **inefficient**: it has to look at all $(N-1)!$ Hamilton circuits, and this can take a long time.

Solving the TSP by Brute Force

If your computer can compute **one million** Hamilton circuits per second...

- $N = 6, 7, 8, 9$: instantaneous
- $N = 10$: about $1/3$ second
- $N = 11$: about 4 seconds
- $N = 12$: about 40 seconds
- $N = 13$: about 8 minutes
- $N = 14$: nearly 2 hours
- $N = 15$: a little over a day
- $N = 20$: over a million years

Solving the TSP Without Brute Force

Is there a better way to tackle the TSP?

That is, is there an optimal algorithm that is also efficient?

Solving the TSP Without Brute Force

Is there a better way to tackle the TSP?

That is, is there an optimal algorithm that is also efficient?

Surprisingly, the answer to this question is **unknown.**

There are optimal algorithms and there are efficient algorithms, but no one has yet discovered an efficient optimal algorithm!

Solving the TSP Without Brute Force

In the next part of the course, we are going to look at some algorithms that find what promises to be a **good** Hamilton circuit in a complete graph, without any guarantee that it is the **best** one.

These algorithms are "natural" (they are sensible ways you might go about trying to construct a Hamilton circuit as cheaply as possible) and they are efficient (they don't require listing all $(N-1)!$ possible Hamilton circuits).

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.**

The Traveling Saleswitch Problem

	H	P	G	C	D	T
Home (H)	0	36	32	54	20	40
Pet store (P)	36	0	22	58	54	67
Greenhouse (G)	32	22	0	36	42	71
Cleaners (C)	54	58	36	0	50	92
Drugstore (D)	20	54	42	50	0	45
Target (T)	40	67	71	92	45	0

The Traveling Saleswitch Problem

	H	P	G	C	D	T
Home (H)	0	36	32	54	**20**	40
Pet store (P)	36	0	22	58	54	67
Greenhouse (G)	32	22	0	36	42	71
Cleaners (C)	54	58	36	0	50	92
Drugstore (D)	20	54	42	50	0	45
Target (T)	40	67	71	92	45	0

- Starting at home, the closest destination is the drugstore.

The Traveling Saleswitch Problem

	H	P	G	C	D	T
Home (H)	0	36	32	54	**20**	40
Pet store (P)	36	0	22	58	54	67
Greenhouse (G)	32	22	0	36	42	71
Cleaners (C)	54	58	36	0	50	92
Drugstore (D)	20	54	42	50	0	45
Target (T)	40	67	71	92	45	0

- Starting at home, the closest destination is the drugstore.
- Therefore, start an itinerary with HD.

The Traveling Saleswitch Problem

	H	P	G	C	D	T
Home (H)	0	36	32	54	**20**	40
Pet store (P)	36	0	22	58	54	67
Greenhouse (G)	32	22	0	36	42	71
Cleaners (C)	54	58	36	0	50	92
Drugstore (D)	20	54	42	50	0	45
Target (T)	40	67	71	92	45	0

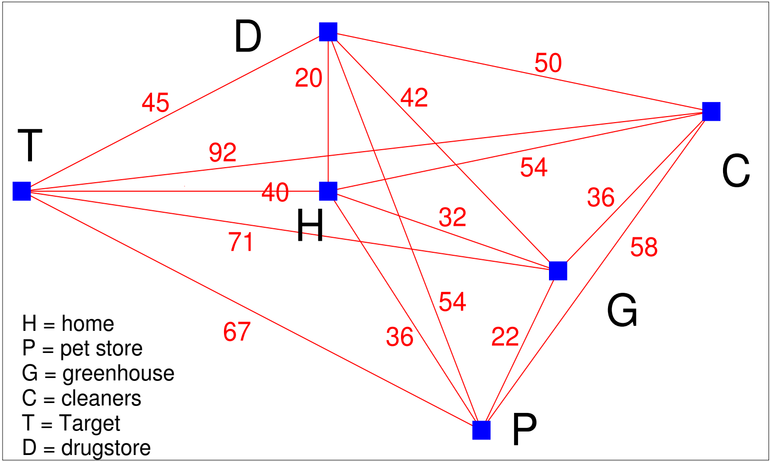
- Starting at home, the closest destination is the drugstore.
- Therefore, start an itinerary with HD.
- Then go from D to the nearest destination not already visited...

The Traveling Saleswitch Problem

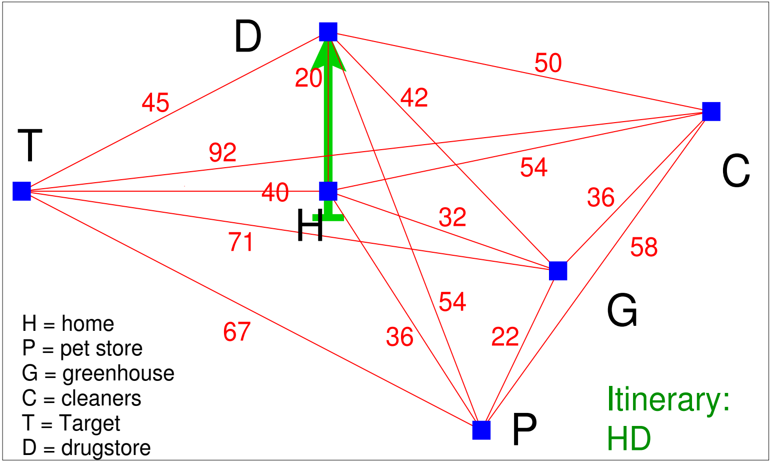
	H	P	G	C	D	T
Home (H)	0	36	32	54	**20**	40
Pet store (P)	36	0	22	58	54	67
Greenhouse (G)	32	22	0	36	42	71
Cleaners (C)	54	58	36	0	50	92
Drugstore (D)	20	54	42	50	0	45
Target (T)	40	67	71	92	45	0

- Starting at home, the closest destination is the drugstore.
- Therefore, start an itinerary with HD.
- Then go from D to the nearest destination not already visited...
- and repeat.

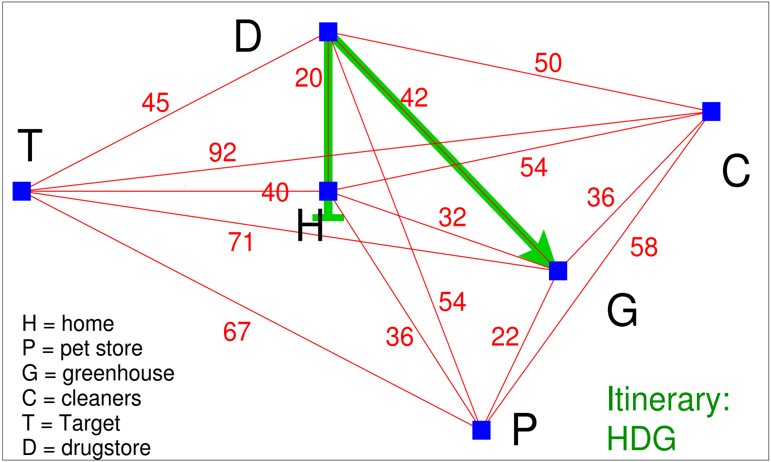
The Traveling Saleswitch Problem



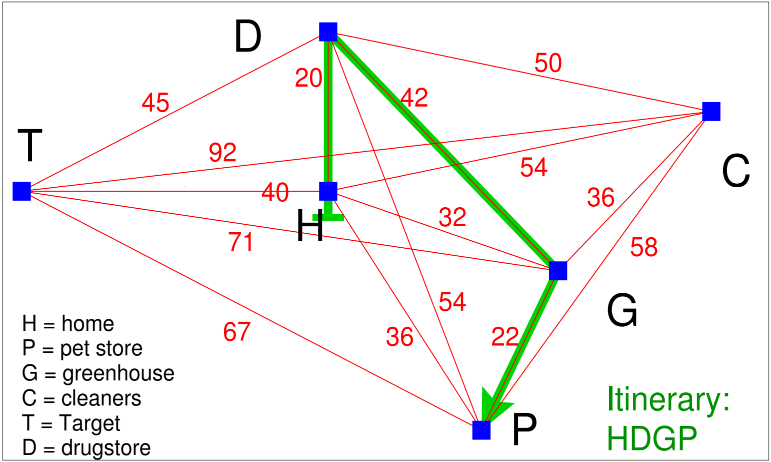
The Traveling Saleswitch Problem



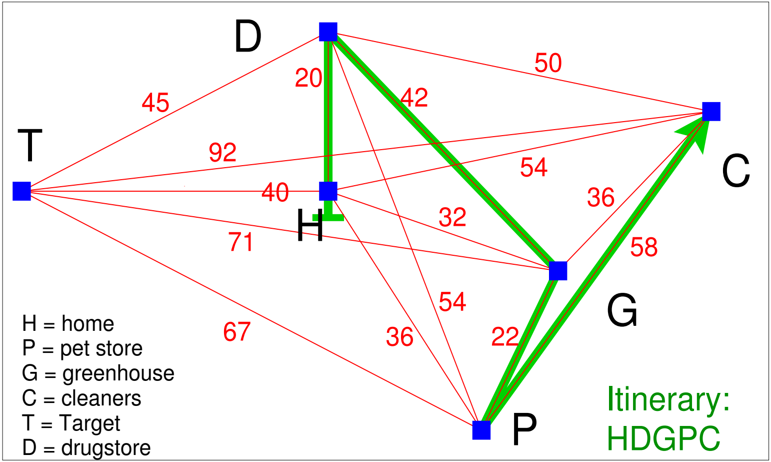
The Traveling Saleswitch Problem



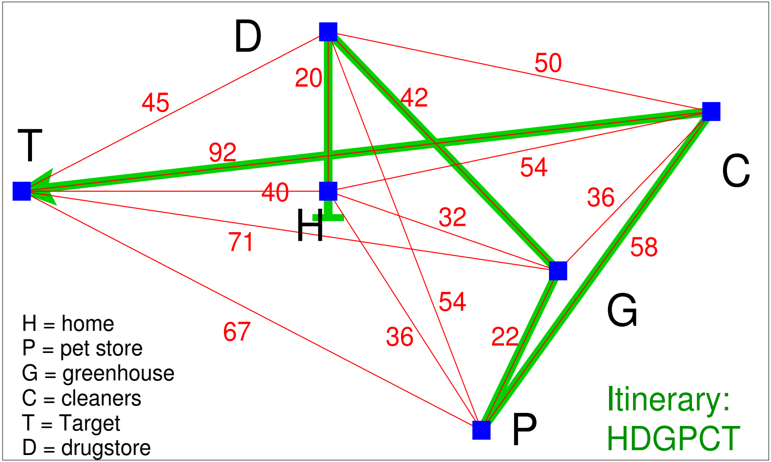
The Traveling Saleswitch Problem



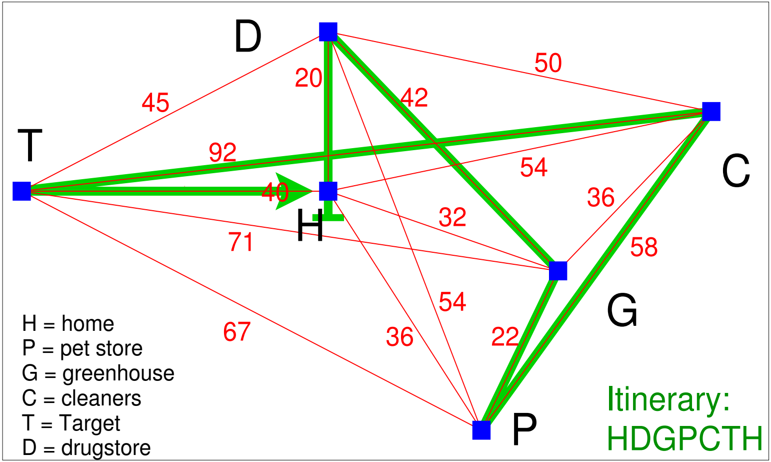
The Traveling Saleswitch Problem



The Traveling Saleswitch Problem



The Traveling Saleswitch Problem



The Traveling Saleswitch Problem

Eventually, we end up with the Hamilton circuit

H, D, G, P, C, T, H.

- Weight of this circuit: 274
- Weight of an optimal circuit: 229
- Average weight of a circuit: 287.6

- So the algorithm has produced a **good solution** (better than average), but **not the best solution**.

Comparing Brute-Force and Nearest-Neighbor

The Brute-Force Algorithm is **optimal** but **inefficient**.

- It is guaranteed to find a solution, but it may take an unreasonably long time to do so.

The Nearest-Neighbor Algorithm is **efficient** but **nonoptimal**.

- It is quick and easy, but does not always find the lowest-weight Hamilton circuit.