Hard Problems in Computing

Jidong Yuan
Beijing Jiaotong University
yuanjd@bjtu.edu.cn

SCC 120: Fundamentals of Computer Science

Intractable Problems

Extremely simple but notoriously unyielding to efficiency!

- Traveling salesman
- Hamiltonian cycle
- Propositional satisfiability
- Propositional entailment
- Subset sum

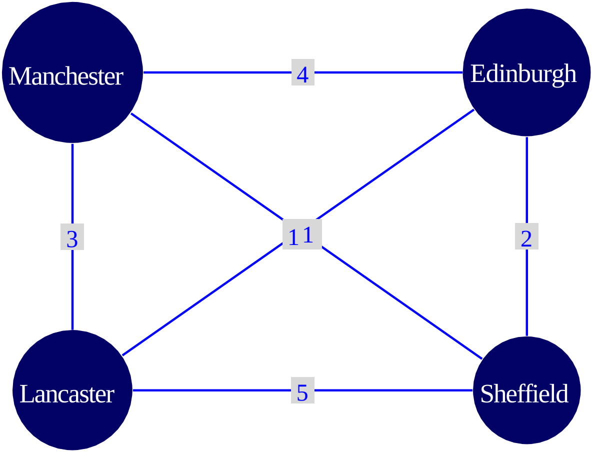
Traveling salesman

Find shortest *tour* of V cities

- Given: a set of cities V and distances among each pair of cities
- Find: The shortest *tour*
  - That visits every city exactly once, and
  - Ends with the starting city
- Problem of great importance in logistics!

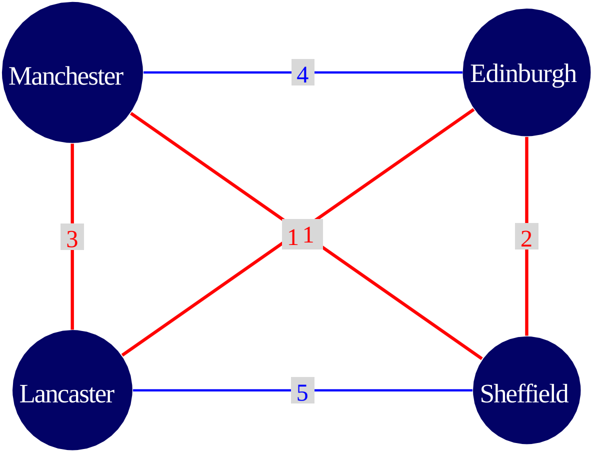
Traveling Salesman: Example

Which is the shortest tour?



Traveling Salesman: Example

Which is the shortest tour?



Traveling Salesman Worst Case Complexity

Imagine a brute force approach

Hint: Draw a tree of alternatives

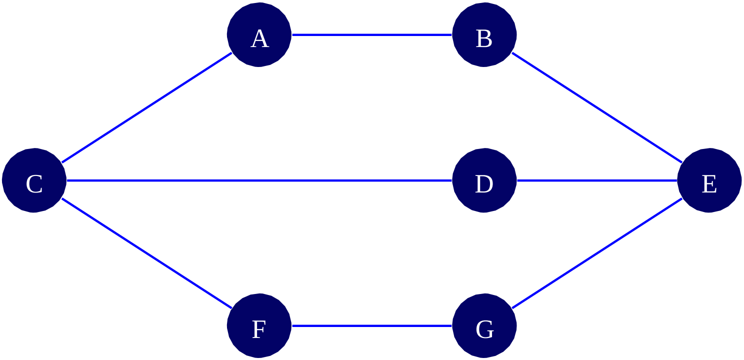
Traveling Salesman Brute-Force Complexity

Answer: $\Omega(|V|!)$, where $|V|$ is the *number* of nodes in set V

- $|V|!$ alternative tours. To find the shortest among them, we need to compute the total distance of each tour.
  - Therefore, we get $\Omega(|V|!)$
  - Enough to tell us that algorithm is intractable!
- If we wanted to be more precise, we could factor in the cost of adding
  - $|V|$ addition operations in each tour
  - Therefore, we get $\Omega(|V|!|V|)$
  - Note, however, that if an algorithm is $\Omega(|V|!|V|)$, then it is also $\Omega(|V|!)$

Hamiltonian Cycle

- Given a graph of set of nodes V and set of edges E, is there a cycle (tour) that
  - Visits each node exactly once
  - Ends in the starting node



Hamiltonian Cycle

Approach and worst case complexity?

- Imagine a brute force algorithm
  - Again, draw a tree of alternatives
- Worst case complexity of brute force algorithm
  - $\Omega(|V|!)$, by reasoning analogous to traveling salesman

Finding Shortest Path Between Two Nodes

Dijkstra's algorithm

- Worst case complexity?
- Dijkstra's original algorithm: $O(|V|^2)$

Propositional (Boolean) Satisfiability

Problem of fundamental importance

- Given: A formula in propositional logic
  - $(p \wedge \neg p) \vee q$
  - $(p \vee q) \wedge (\neg r \wedge \neg p)$
  - $(p \rightarrow q) \wedge (p \wedge \neg q)$
- Problem: Is there an assignment of either 1 (true) or 0 (false) to each propositional variable ($p, q, \ldots$) in the formula such that the formula is *satisfied*, that is, evaluates to 1?

Answers

- $(p \wedge \neg p) \vee q$
  - $p = 0, q = 1$
  - $p = 1, q = 1$
- $(p \vee q) \wedge (\neg r \wedge \neg p)$
  - $p = 0, q = 1, r = 0$
- $(p \rightarrow q) \wedge (p \wedge \neg q)$
  - Unsatisfiable

Propositional Satisfiability

Approach and complexity?

Propositional Satisfiability

Approach and worst case complexity

- Truth tables!
- For formula with N variables, complexity is $\Omega(2^N)$

Propositional Entailment

- Given: Two propositional formulas, say X and Y
- Problem: Does X entail Y? Equivalently, one may ask
  - Does Y follow from X?
  - Does X derive Y? (Can Y be derived from X?)

- Does $(p \wedge \neg p) \vee r$ entail r?
- Does $(p \wedge \neg p) \wedge (r \vee s)$ entail r?
- Does p entail $p \vee r$?

Propositional Entailment

Approach and worst case complexity?

Propositional Entailment

Approach and worst case complexity?

- Approach
  - Construct truth table for $X \rightarrow Y$
  - For each row, if X is 1, and the corresponding $X \rightarrow Y$ is also 1, then X entails Y; otherwise, no. (Note: we do not care about those models that evaluate X to 0. The result of evaluating Y for these models is irrelevant.)
- Worst case complexity is $\Omega(2^N)$, where N is the number of variables
  - The same as propositional satisfiability (PS)
    - Why? We must construct the entire truth table for both satisfiability and entailment.

Subset Sum

- Given: A set of integers
- Problem: Is there a subset such that the sum of its elements is k?
  - Special case: $k = 0$
- Is there a subset of $\{-6, 4, 44, 23, -1, 11, 10, 3\}$ such that the sum of its elements is 0?

Subset Sum

Approach and worst case complexity?

Subset Sum

Approach and worst case complexity

- Calculate all possible subsets and sum up each
- $\Omega(2^N)$, where N is the cardinality of the set

Summary

"Intractable", that is, hard problems

- Considered naïve (brute-force) algorithms
  - Traveling salesman: factorial complexity
  - Hamiltonian cycle: factorial complexity
  - Propositional satisfiability: exponential complexity
  - Propositional entailment: exponential complexity
  - Subset sum: exponential complexity
- Clever algorithms devised that perform better than naïve algorithms we studied, but yet no known polynomial-time algorithms
- In following lectures, we will understand "intractable" more formally, when we consider notions such as *NP-completeness*