

Lecture V - NP Problems and Graphs

¹ Department of Mathematics and Systems Analysis,
Systems Analysis Laboratory, Aalto University, Finland

February 4, 2024



Aalto University

Previously on..

Matching Problems:

- Weighted Matching;
- Maximum Matching.

PREVIOUSLY ON...

So far, all problems either have **algorithm** or a **mixed integer** formulation.

But what happens when **an algorithm** is not achievable nor **efficient**? A **mixed integer formulation** **might still** be possible, but would it be **enough**?

Complexity class P

Decision Problems

Decision problem is a **yes-or-no** problem.

They differ from **optimization problem**, because the former requires an answer that have an **optimal configuration**.

For instance: "*What is the shortest path between two nodes?*" **vs** "*Is a particular path P the shortest path between these two nodes?*".

Remark: An optimization problem has a **corresponding** decision problem.

Examples

- P is the **class** of all decision problems (X, Y) for which there is a **polynomial** time algorithm.
 - Given $x \in A^*$: compute $f(x) \in \{0, 1\}$ with $\text{time}(x) \leq p(\text{size}(x))$.
- linear inequalities;
 - shortest path;
 - maximum matching;
 - minimum cost flow.
 - ...

Complexity class NP

Examples

- decision problem (X, Y) belongs to class NP if for each $y \in Y$ a certificate c can be verified in **polynomial time**
 - usually c is a **feasible solution** to the problem
 - name $NP =$ **nondeterministic** polynomial: “guessing” certificates long enough would work
 - (X, Y) can be solved by **nondeterministic in polynomial time**
- integer linear inequalities $\rightarrow c$ is feasible vector x
 - knapsack $\rightarrow c$ is a feasible set of items to take

P vs. NP

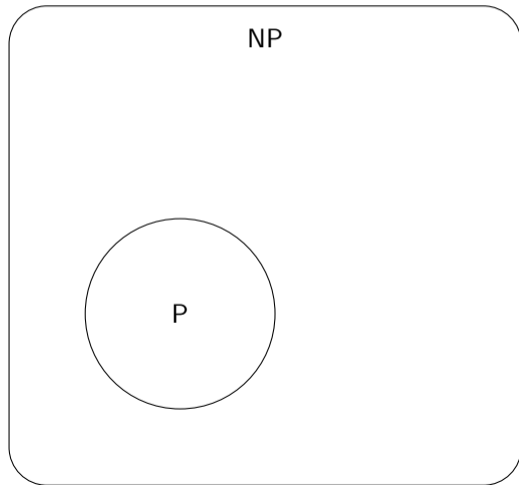
Theorem

$$P \subseteq NP.$$

Proof.

$(X, Y) \in P$ can be decided in **polynomial time**. $\Rightarrow x$ can be used as **certificate**. □

P vs. NP



Combinatorial
Optimization

Previously on..

Complexity
class P

Complexity
class NP

TSP -
Travelling
Salesman
Problem

TSP - Travelling Salesman Problem

Definition

TSP: Travelling Salesman (or Salesperson) Problem

Imagine a scenario, where a set of cities are expecting a visit from a **travelling** merchant.

As part of their visit, this **salesman** has to start and finish their travel in the same city, cannot visit the same city more than once and every city has to be visited in a single trip.

Example



Figure: A lone traveller about to make important decisions

→ What is the shortest possible route?

History

Such problem is called **Travelling Salesman Problem**, very important to the fields of theoretical computer science and operations research.

It was first described by Irish mathematician **W.R. Hamilton** and British mathematician **Thomas Kirkman** in the 1800s through the description of a game where the solution involved a cycle without overlapping nodes.



Initial Approach

At first glance, the first solution is to **try all possibilities** and choose the best solution.

→ enumeration process

Initial challenge: for an instance with n cities, there are 2^n possible combinations.

→ **impractical**.

Is there any **better alternative**?

First, let us assume that the set of cities can be modelled as graph $G = (V, E, f)$, where:

- V is the set of individual cities;
- E represent the paths between a pair of cities and;
- f_{ij} is the cost to travel from city i to city j , for all $(i, j) \in E$.

Modelling - Variable and Objective

The **choice** to travel from city i to city j using a path (edge in our modelling) connecting them is modelled by our **decision variable**.

$$x_{ij} = \left\{ \begin{array}{ll} 1, & \text{if path goes from city } i \text{ to city } j \\ 0, & \text{otherwise} \end{array} \right\}$$

Our **objective function** can be also derived from the problem description:

$$\min \sum_i^n \sum_{j, j \neq i}^n x_{ij} f_{ij}$$

where n is the number of cities ($|V| = n$).

Modelling - Constraints

Two constraints can also be derived directly from description:

Singular incoming degree:

$$\sum_{i=1, i \neq j}^n x_{ij} = 1 \quad \forall j \in \{1, \dots, n\}$$

Singular outgoing degree:

$$\sum_{j=1, j \neq i}^n x_{ij} = 1 \quad \forall i \in \{1, \dots, n\}$$

Those constraints are characterize **paths**.

Combinatorial
Optimization

Previously on..

Complexity
class P

Complexity
class NP

TSP -
Travelling
Salesman
Problem

Modelling - Split version

These constraints imposed that every city is visited only once.

However, they do not guarantee that there is a single trip will connecting all cities.

For instance:

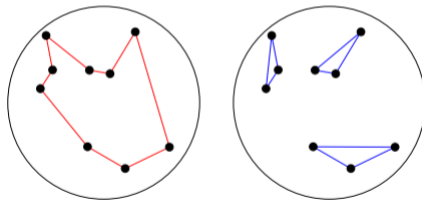


Figure: Two solutions that do not violate the previous constraints, but only one has a single trip.

Modelling Decisions - MTZ vs DFJ

There are two main strategies to prevent separate tour (**subtour**) from our potential solution: **Miller-Tucker-Zemlin** and **Dantzig-Fulkerson-Johnson**.

Both impose the presence of a single tour using **linear constraints**.

Miller-Tucker-Zemlin

Requires an **additional variable** to track which city has been visited starting from initial city $i = 1$.

→ By setting $u_j > u_i$, it determines the **order** of visiting each city (city j will be visited after city i).

This leads to following requirement:

$$u_j \geq u_i + 1 \text{ if } x_{ij} = 1$$

which can be encapsulated as the following constraints:

$$u_i - u_j + 1 \leq (n - 1)(1 - x_{ij})$$

$$2 \leq u_i \leq n$$

$$\text{for } 2 \leq i \neq j \leq n$$

$$\text{for } 2 \leq i \leq n$$

$$\text{Minimize } \sum_i^n \sum_{j, j \neq i}^n x_{ij} f_{ij}$$

Subject to:

$$\sum_{i=1, i \neq j}^n x_{ij} = 1$$

$$\forall j \in \{1, \dots, n\}$$

$$\sum_{j=1, j \neq i}^n x_{ij} = 1$$

$$\forall i \in \{1, \dots, n\}$$

$$u_i - u_j + 1 \leq (n - 1)(1 - x_{ij})$$

$$2 \leq i \neq j \leq n$$

$$2 \leq u_i \leq n$$

$$2 \leq i \leq n$$

$$x_{ij} \in \{0, 1\}$$

$$i, j \in \{1, \dots, n\}$$

Impose an extra requirement that **eliminates all subset of nodes to create a subtour**.

$$\sum_{i \in Q} \sum_{j \neq i, j \in Q} x_{ij} \leq |Q| - 1 \quad \forall Q \subset \{1, \dots, n\}, |Q| \geq 2$$

Dantzig-Fulkerson-Johnson

$$\text{Minimize } \sum_i^n \sum_{j, j \neq i}^n x_{ij} f_{ij}$$

Subject to:

$$\sum_{i=1, i \neq j}^n x_{ij} = 1 \quad \forall j \in \{1, \dots, n\}$$

$$\sum_{j=1, j \neq i}^n x_{ij} = 1 \quad \forall i \in \{1, \dots, n\}$$

$$\sum_{i \in Q} \sum_{j \neq i, j \in Q} x_{ij} \leq |Q| - 1 \quad \forall Q \subset \{1, \dots, n\}, |Q| \geq 2$$

$$x_{ij} \in \{0, 1\} \quad i, j \in \{1, \dots, n\}$$



Combinatorial
Optimization

Previously on..

Complexity
class P

Complexity
class NP

TSP -
Travelling
Salesman
Problem