Lecture Notes - Week VI

NP Problems and Graphs II

Fernando Dias, Philine Schiewe and Piyalee Pattanaik

January 8, 2024

CHAPTER 1
Cliques

In another definition, a **clique** is a subset of nodes in a graph where every two distinct nodes is adjacent (e.g. there is an edge connecting every pair of nodes in the subset).

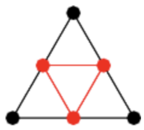

Figure 1.1: Example of a clique consisting of three nodes.

This problem dates back to the 1940s with **Turan's theorem** regarding clique in dense graphs. However, it gained popularity from the work of **Luce** & **Perry** in 1949 in social network applications.

Most efforts in developing approximations and relaxations are present in the field of **communication networks** (starting from Prihar in 1959 and in following decades) and in **Bioinformatics** (by Ben-Dor, Shamir & Yakhini, 1999).

Most of its applications are in Communication:

- Design of efficient circuits;
- Automatic test pattern generation;
- Hierarchical partition.

and in Bioinformatics:

- Gene expression;
- Ecological niches;
- Metagenomics and evolutionary tree;

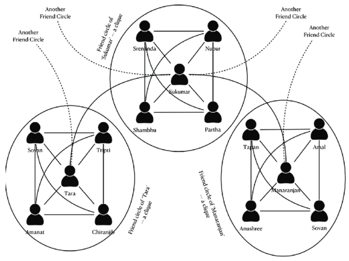

Figure 1.2: Clique application in telecommunication

1.1 MODELLING

In its decision version, as well as several other versions (such as maximum clique, maximum clique with weights), is a **NP-hard problem**. For the maximum clique problem (the largest possible clique in a graph), there is an ILP formulation.

For each node $v \in V$, the decision variable x_v is assigned.

$$x_v = \left\{ \begin{array}{ll} 1, & \text{if node } v \text{ is selected as part the maximum clique} \\ 0, & \text{otherwise} \end{array} \right\}$$

For objective function, the goal is to minimize the number of nodes selected as part of the maximum clique:

$$\min \sum_{v \in V}^{n} x_v$$

Remark: It can be further changed into **maximum weighted clique** by multiply the weight w_v to each selected node.
For constraint, a new concept is required to be introduced. Let $\overline{E}$ be the **complementary edges** from G, such that:

- $(u, v) \notin \overline{E}$ if (u, v) in G;
- $(u, v) \in \overline{E}$, if (u, v) is not in G.

Hence, the constraint for maximum clique requires that for each edge $(u, v) \in \overline{E}$, at **least one** of the nodes **has** to be selected:

$$x_u + x_v \leq 1 \qquad\qquad \forall (u, v) \in \overline{E}$$

The resulting model is as follows:

$$\text{Minimize} \quad \sum_{v \in V} x_v$$

Subject to:

$$x_u + x_v \leq 1 \qquad \forall (u, v) \in \overline{E}$$
$$x_v \in \{0, 1\} \qquad \forall v \in V$$

CHAPTER 2
Independent Set

An **independent set** is a subset of nodes in a graph where **there is no adjacency** between all pairs of nodes.

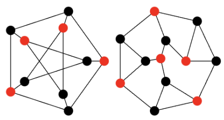

Figure 2.1: Example of independent sets

It is frequently called the anti-clique problem.

Most of the effort put into this problem is related to studies on **clique problems** and **vertex cover**. However, some special versions were studied by Perrin in the late 1890s and by Padovan in the 1990s. Its applications are also related to clique problems, emphasizing stable genetic regions.

2.1 MODELLING

Known as the **anti-clique**, the same clique models can be used. However, it requires the use of complementary edges.

Definition 1 *A set of nodes in a graph $G = (V, E)$ creates a clique if the same set form as an independent set in the complimentary graph $\overline{G} = (V, \overline{E}$.*

Alternatively, it also has an ILP formulation. For each node $v \in V$, the decision variable x_v is assigned.

$$x_v = \left\{ \begin{array}{ll} 1, & \text{if node } v \text{ is selected as part an independent set} \\ 0, & \text{otherwise} \end{array} \right\}$$

For objective function, the goal is to minimize the number of nodes selected as part of an independent set:

$$\min \sum_{v \in V}^{n} x_v$$

Now for the constraint: if a node is part of an independent set, it cannot be adjacent to any other node in the independent set. Hence,

$$x_u + x_v \leq 1 \qquad \forall (u, v) \in \overline{E}$$

The full model is summarized below:

$$
\begin{aligned}
&\text{Minimize} \quad \sum_{v \in V} x_v \\
&\text{Subject to:} \\
&x_u + x_v \leq 1 && \forall (u, v) \in E \\
&x_v \in \{0, 1\} && \forall v \in V
\end{aligned}
$$

CHAPTER 3
Dominant Set

As per definition, a **dominant set** is a subset of nodes from a graph $G = (V, E)$ such that any node $v \in V$ is either part of the subset or adjacent.

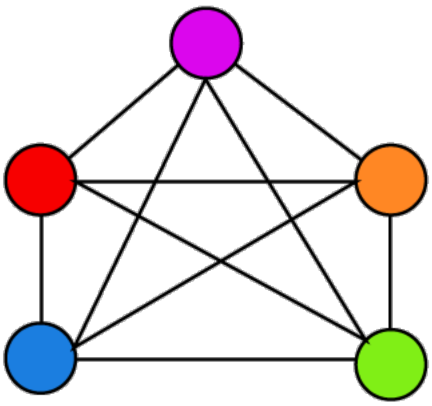

Figure 3.1: Example of a graph with four nodes dominant set

The **domination problem** (in various forms and extensions) has been present in the literature since the 1950s, with increased efforts in the 1970s due to its application in **networking** and **electrical grids**.

In **wireless network**, approximation algorithms have been used to find near-optimal routes within ad-hoc networks.

Further applications are related to **document summarization** and **safety** in electrical grids.

3.1 MODELLING

For each node $v \in V$, the decision variable x_v is assigned.

$$x_v = \left\{ \begin{array}{ll} 1, & \text{if node } v \text{ is selected as part an independent set} \\ 0, & \text{otherwise} \end{array} \right\}$$

For objective function, the goal is to minimize the number of nodes selected as part of a dominant set:

$$\min \sum_{v \in V}^{n} x_v$$

In a graph $G = (V, E)$, every node has two roles to play:

- It is part of the dominant set;
- It is adjacent to a node in the dominant set.

Those two roles are **mutually exclusive**: only one can be satisfied for each node v in V. This can be modelled by the boolean operator OR.

For the constraint, it can be broken down into two roles:

- It is part of the dominant;

$$x_v = 1 \qquad \forall v \in V$$

- It is adjacent to a node in the dominant set.

$$\sum_{(u,v) \in E} x_u = 1 \qquad \forall v \in V$$

Applying the boolean operator results in the following constraint:

$$x_v + \sum_{(u,v) \in E} x_u \geq 1 \qquad \forall v \in V$$

The full model is as follows:

$$\begin{aligned}
&\text{Minimize} \quad \sum_{v \in V} x_v \\
&\text{Subject to:} \\
&x_v + \sum_{(u,v) \in E} x_u \geq 1 && \forall v \in V \\
&x_v \in \{0, 1\} && \forall v \in V
\end{aligned}$$

CHAPTER 4
Hamiltonian Path

In a final definition, **Hamiltonian Path**/**Cycle** (or traceable path/circuit) is a path (or cycle) in a graph where **every node is visited only once**.

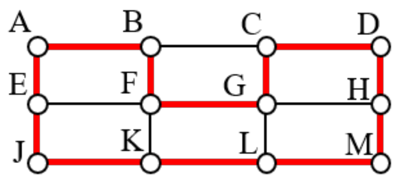

Figure 4.1: Example of a Hamiltonian path

Its name is associated with **William Rowan Hamilton**, inventor of the icosian game (finding a cycle in the edge graph of a dodecahedron - also known as Hamilton's puzzle). However, Thomas Kirkman has studied this problem earlier, and similar problems have been addressed by **Indian** and **Islamic mathematician** since the 9th century.

One of the most **famous NP-hard problem**, it has been widely researched since 1972, when concrete solutions were proposed by Bondy-Chvatal based on previous work by Dirac and Ore in 1952.

Many areas with interest in TSP also research Hamiltonian paths, with emphasis in:

- Routing problems, such as vehicle routing problems;
- Electronic circuit design;
- Computer graphics;
- Genome mapping.

4.1 MODELLING

In both Hamiltonian path or cycle formulation, the decision variable x_{uv} is assigned to each edge (u, v).

$$x_{uv} = \left\{ \begin{array}{ll} 1, & \text{if edge } (u, v) \text{ is part of a Hamiltonian path or cycle} \\ 0, & \text{otherwise} \end{array} \right\}$$

Our objective function is to minimize the cardinality of such path or cycle:

$$\min \sum_{(uv)\in E} x_{uv}$$

For **both cycle and path**, it is required that each node is visited by two edges. Then, the following constraint:

$$\sum_{v\in V} x_{uv} = 2 \qquad\qquad \forall u \in V$$

For Hamiltonian cycles, for any subset of nodes that is not V or $\emptyset$, there should be at least two edges being used, resulting in the constraint:

$$\sum_{(uv):u\in S, v\notin S} x_{uv} \geq 2 \qquad\qquad \forall S \subset V, S \neq 0$$

For the Hamiltonian path, both **source** $s \in V$ and **sink** $t \in V$ should be selected (assuming a single source and sink). For any intermediary node, the same amount of incoming edges should be equal to outgoing edges.

This results in the following constraints:

$$\sum_{(uv)\in\delta^+(u)} x_{uv} - \sum_{(vu)\in\delta^-(u)} x_{uv} = b_u \qquad\qquad \forall u \in V$$

where $\delta^+(u)$ and $\delta^-(u)$ represents the **out-** and **in-edges** from u. b_v assumes different values if v is a **source** $b_v = 1$, a **sink** $b_v = -1$ or an **intermediary** node $b_v = 0$.

For Hamiltonian path, two extra constraint are required:

- Only one out-edge is used:

$$\sum_{(uv)\in\delta^+(u)} x_{uv} \leq 1 \qquad\qquad \forall u \in V$$

- To impose that no subtour is present, similar to subtour elimination from TSP.

CHAPTER 5
Conclusion

So far, all problems in the last two lectures have two elements in common:

- Hard/impossible to solve in polynomial time;
- Easy to verify a solution in polynomial time.

Also, all these problems can be easily "transformed" into each other, which means that all of these problems are **NP problems**.

The remaining question is: how do you **prove** a novel problem is NP?

Is there any **correlation** between different NP problems?

Any form of **transformation** as well? Moreover, how to do it effectively?