O problema do carteiro chinês

1 As sete pontes de Königsberg

O problema das sete pontes de Königsberg foi descrito e solucionado pelo matemático Leonhard Euler em 1736, o problema consistia em decidir se seria possível traçar no mapa de Königsberg um trajeto que percorresse cada uma de suas 7 pontes uma única vez, sem repetições.

A importância deste problema é enorme para a área da computação, já que o artigo de Euler publicado sobre ele é considerado o primeiro artigo sobre teoria dos grafos.

Vamos definir como **passeio** em um grafo uma sequência finita não vazia $P = \{v_0, a_1, v_2, a_2, \ldots, a_k, v_k\}$, cujos termos são alternadamente vértices v_i e arestas a_j e tal que, para todo i, $1 \leq i \leq k$, os extremos de a_i são v_{i-1} e v_i. Os vértices v_0 e v_k são a origem e o término de P, respectivamente; e os vértices $v_1, v_2, \ldots, v_{k-1}$ são chamaos vértices internos de P.

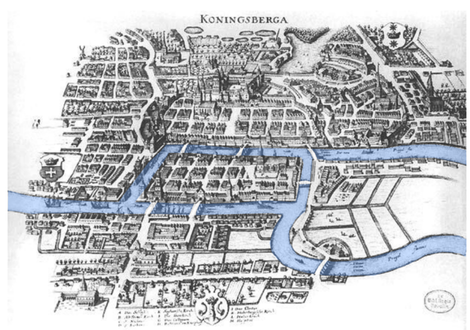

Figura 1: Representação das sete pontes de Königsberg

Uma **trilha** é um passeio sem arestas repetidas. Um **caminho** é um passeio sem vértices repetidos. Por comprimento de um passeio, denotado por $||P||$, é o número de arestas de P.

Um passeio é considerado **fechado** se sua origem e término são iguais.

Uma trilha fechada cuja origem e vértices internos são todos distintos é um **circuito**.

Devido às contribuições de Euler ao problema descrito, definiu-se **trilha euleriana** (ou caminho euleriano) como uma trilha que passa por todas arestas de um grafo e **grafo euleriano** como um grafo que possui uma trilha euleriana fechada.

Euler modelou essa área da cidade como um grafo, representado na figura 2, tratando as pontes como arestas e as massas de terra como vértices. A partir de tal modelagem e das definições feitas, o problema de Königsberg consiste, em definir se o grafo que representa a cidade possui ou não uma trilha euleriana.

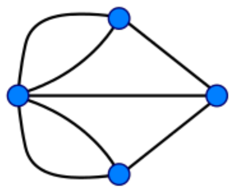

Figura 2: Modelagem de Königsberg como um grafo

O matemático provou que uma condição necessária para que um grafo seja euleriano é que todos seus vértices possuam grau par; ele também afirmou, porém sem prova, que qualquer grafo conexo cujos vértices possuem grau par é euleriano.

A prova para essa afirmação só foi publicada mais de cem anos depois, em 1873, por Carl Hierholzer [2].

Teorema 1.1 (Teorema de Euler ou de Euler-Hierholzer)**.** *Um grafo é euleriano se, e somente se, é conexo e todos seus vértices possuem grau par.*

Antes de trabalhar na prova de tal teorema, primeiramente trabalharemos em um lema:

Lema 1. *Se G é um grafo tal que todos seus vértices possuem grau maior ou igual a 2, então G possui um ciclo.*

Demonstração. Vamos assumir que um grafo qualquer $G = \{V, A\}$ não possui um ciclo, neste caso G deverá possuir no máximo $|V|-1$ arestas, havendo assim a possibilidade de G ser uma árvore.

Como cada aresta contribui em duas unidades para a soma dos graus de um grafo, se vale que $|A| \leq |V| - 1$ então vale também que $\sum_{v \in V} g(v) \leq 2|V| - 2$, sendo $g(u)$ o grau de um vértice u.

Porém, faz parte da hipótese do lema que todos vértices possuem grau maior ou igual a 2, o que implica na seguinte desigualdade: $\sum_{v \in V} g(v) \geq 2|V|$, o que é inconsistente com o que havíamos descoberto.

Por contradição, provamos que G deverá possuir ciclos dadas as condições do lema. □

Provado tal lema, podemos agora provar o teorema 1.1:

Demonstração. Seja $G = \{V, E\}$ o grafo em questão.

Começamos provando que se um grafo é euleriano e conexo então todos seus vértices possuem grau par.

Agora, provaremos por indução no número de arestas de G que se G for conexo e se todos seus nós possuem grau par, então ele é euleriano.

□

Corolário 1. *Um grafo possui um caminho euleriano se, e somente se, é conexo e possui apenas zero ou dois vértices de grau ímpar.*

Demonstração. Seja G um grafo conexo qualquer. Realizaremos a demonstração para os seguintes casos:

1. G não possui vértices de grau ímpar. Neste caso, G possui, segundo o teorema 1.1, um circuito euleriano, que é um tipo de caminho euleriano.

2. G possui apenas um vértice de grau ímpar. Este caso é impossível de se acontecer, já que a soma do grau de todos vértices deve ser par.

3. G possui dois vértices de grau ímpar. Sejam u e v os únicos vértices de G que possuem grau ímpar. Adiciona-se uma aresta fictícia ao grafo G, a aresta uv, fazendo com que tanto u quanto v possuam graus pares. Chamaremos o grafo G acrescido da aresta uv de G' Como ambos eram os únicos vértices de grau ímpar de G, isso implica que todos vértices de G' possuirão grau par. Como, da premissa, G é conexo, G' também deverá ser conexo, valendo assim o teorema 1.1, provando a existência de um circuito euleriano em G', que chamaremos de p. Como p é um circuito euleriano, isso implica que $uv \in p$.

   A ideia então é retirar de p a aresta uv,

4. G possui três ou mais vértices de grau ímpar.

□

Como os quatro vértices do grafo da figura 2 possuem grau ímpar, é impossível, pela condição estabelecida, que exista um caminho euleriano.

Segue agora uma implementação de um algoritmo que encontra o circuito euleriano de um grafo dado que o mesmo possui um circuito euleriano:

```
// maximum node and edge constraints
const int MAXN = 212345;
const int MAXE = 2123456;

int s[MAXN];

// adjacency list
vector<pii> adj[MAXN];
int cnt;

int mrk[MAXE];

int temp_path[MAXE], final_path[MAXE], tsz, fsz;

void add_edge(int a, int b){
        // undirected edge
        adj[a].pb(pii(b, cnt));
        adj[b].pb(pii(a, cnt++));
}

// Only use if graph is guaranteed to have an euler cycle
void euler_dfs(int u){
        temp_path[tsz++] = u;
        for(pii ar: adj[u]){
                int id = ar.snd;
                if(mrk[id]) continue;
                mrk[id] = 1;
```

```
            int v = ar.fst; euler_dfs(v);
        }
        final_path[fsz++] = temp_path[--tsz];
}
// Cycle stored in final_path[0...fsz-1]
```

2 Undirected CPP

O grafo deverá ser conexo.

Solução:

1. Criar um grafo completo ligando todos vértices u, v que possuirem grau ímpar com arestas de custo igual ao menor caminho entre u e v.

2. Rodar algoritmo de emparelhamento perfeito de custo mínimo (Húngaro com otimização do Edmonds e Karp, $\mathcal{O}(n^3)$).

Observações:

- Pelo "Lema do aperto de mão"é garantido que haverá um número par de vértices de grau ímpar.

- Essa solução não se aproveita do grafo ser esparso, há outra formulação do Edmonds e Johnson que leva isso em consideração.

- Um problema similar é o de cobrir todas arestas com ciclos simples, de modo que o comprimento total dos ciclos é minimizado. Para grafos planares esses problemas são equivalentes.

3 Directed CPP

O grafo deverá ser fortemente conexo.

Solução:

1. Criar um grafo P, Q-bipartido completo. P deve conter todos os vértices do grafo original com excesso de grau de entrada, e Q deve conter todos vértices com excesso de grau de saída. que possuem valores diferentes de grau de entrada e saída. O custo das arestas entre P e Q deverá ser igual ao custo do menor caminho entre os dois vértices que a mesma liga.

2. Modelar uma rede de fluxo: vértices de P recebem um excesso igual a diferença do grau de entrada pelo grau de saída, vértices de Q recebem uma demanda igual a diferença do grau de saída pelo grau de entrada.

3. Rodar um algoritmo de fluxo de custo mínimo.

Observações:

- Modelagem de fluxo com arestas de capacidade unitária, aumentando a velocidade do algoritmo.

4 Mixed CPP

NP-hard, mesmo se G for planar e se todos c_{ij} forem iguais.

5 Windy Postman Problem

NP-hard

Se todo ciclo do grafo tem o mesmo custo em ambos sentidos, transforma a aresta entre i e j de custos c_{ij} e c_{ji} em uma aresta não direcionada com custo $\frac{c_{ij}+c_{ji}}{2}$. Essa transformação reduz o problema de a um CPP não direcionado, que tem solução polinomial, porém para isso é necessário checar se todo ciclo tem o mesmo custo nas duas direções.

6 Hierarchical Postman Problem

NP-hard

Seja $P = \{A_1, A_2, \ldots, A_k\}$ uma partição do conjunto de arestas A. Determinar um caminho em G de custo mínimo que sai de um vértice s e atinge um vértice t, respeitando uma hierarquia das arestas. O caminho só poderá possuir uma aresta pertencente a A_j se o caminho passar anteriormente por todas arestas pertencentes a A_i, se $i < j$.

7 Anotações

- Todo mixed CPP pode ser transformado em um WPP.

Euler, Leonhard (1736). "Solutio problematis ad geometriam situs pertinentis". .

Referências

[1] Euler, Leonhard *Solution problematis ad geometriam situs pertinentis.* Comment. Acad. Sci. U. Petrop 8, 128–40, 1736.

[2] Hierholzer, Carl *"Ueber die Möglichkeit, einen Linienzug ohne Wiederholung und ohne Unterbrechung zu umfahren"*, Mathematische Annalen, 6 (1): 30–32, doi:10.1007/BF01442866, 1873.