04 – Grafos: caminhos e coloração

SCC0503 – Algoritmos e Estruturas de Dados II

Prof. Moacir Ponti Jr.

www.icmc.usp.br/~moacir

Instituto de Ciências Matemáticas e de Computação – USP

2011/1

Sumário

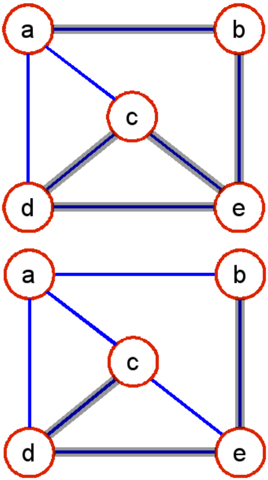
Sumário

1. Caminhos
   - Recordando
   - Problema das Pontes de Königsberg e Caminho Euleriano
   - Problema do Caixeiro Viajante e Caminho Hamiltoniano
2. Coloração
   - k-Coloração e Número Cromático
   - Algoritmos e aplicações

- **Caminho**: sequência de vértices $v_1, v_2, \cdots, v_k$ tal que os vértices consecutivos v_i e v_{i+1} são adjacentes

Ao lado temos os caminhos:

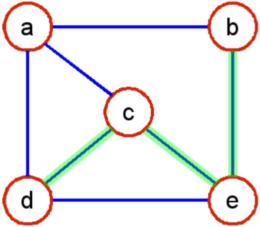
- a, b, e, d, c, e
- b, e, d, c

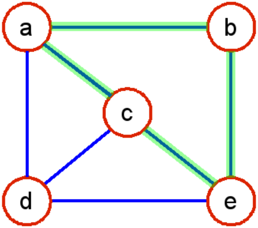


Caminho (II)

- **Caminho simples**: caminho para o qual não há vértices repetidos

- **Ciclo simples**: caminho simples $v_1, v_2, \cdots, v_k$, onde $v_k = v_1$.





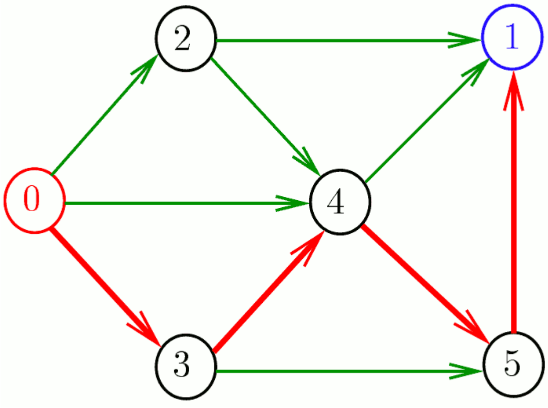
Caminho simples b, e, c, d

Ciclo simples a, b, e, c, a

Procurando um caminho

Problema: dados um digrafo G e dois vértices s e t, decidir se existe um caminho de s a t.

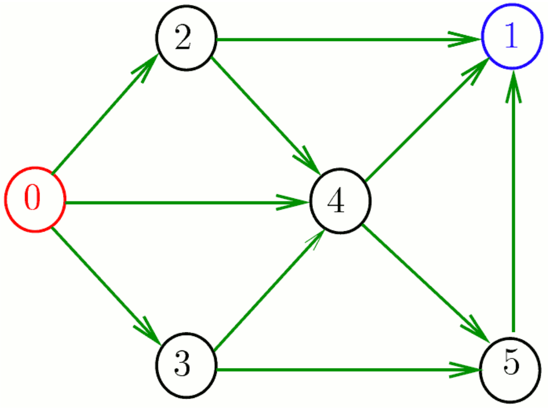
Exemplo: para s=0 e t=1, sim!



adaptado do material do Prof. José Coelho Pina (IME/USP)

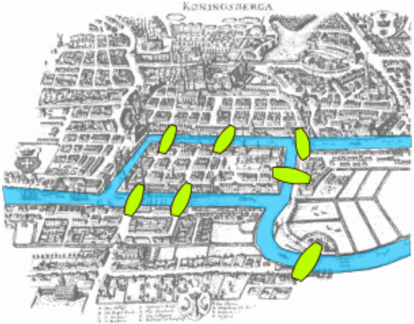
Problema: dados um digrafo G e dois vértices s e t, decidir se existe um caminho de s a t.

Exemplo: para s=1 e t=0, não!



adaptado do material do Prof. José Coelho Pina (IME/USP)

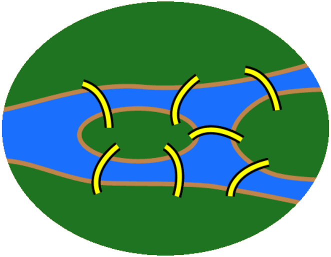
Problema das Pontes de Königsberg

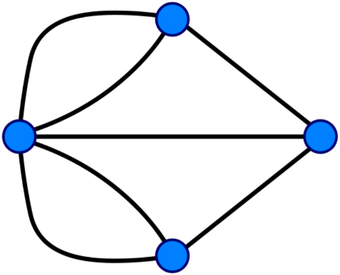


- Problema baseado na cidade de Königsberg *(Prússia até 1945, atual Kaliningrado, Rússia)* que é cortada pelo Rio Pregolia.
- Há duas grandes ilhas que na época contavam com sete pontes.

- **Problema**: encontrar um caminho que passe por cada ponte uma vez, e apenas uma vez.
  - as ilhas não podem ser alcançadas por outra rota que não as pontes
  - cada ponte deve ser sempre cruzada completamente.

- Leonard Euler, em 1735, resolveu o problema, escrevendo um teorema provando que o caminho não era possível. Acredita-se ser o primeiro uso de "grafos" da história.

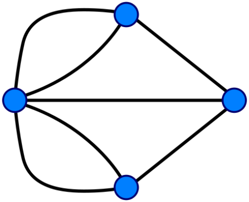




Caminho e Grafo Euleriano

- O modelo requer arestas múltiplas no grafo.
- O **teorema** de Euler é considerado o resultado mais antigo da **teoria dos grafos**.

- Euler estabeleceu que um caminho que passe por todos as **arestas** uma única vez — chamado **caminho euleriano** —, depende do grau dos vértices do grafo.
  - se o caminho iniciar e terminar no mesmo vértice, é chamado de **circuito euleriano**
  - grafos que possuem esse circuito são chamados **grafos eulerianos**.



- Para existir um **circuito euleriano** em um grafo, todos os vértices devem possuir grau par.
- Para haver um **caminho euleriano**, devem haver exatamente 2 vértices com grau ímpar. Grafos que possuem um caminho euleriano (mas não um circuito euleriano) são chamados atravessáveis ou semi-eulerianos.

Problema do Carteiro Chinês ou da Inspeção de Rotas

Problema e requisitos

- Imagine um carteiro que deve percorrer uma rota diariamente, e quer identificar um circuito que o permita minimizar a distância total percorrida.
- Esse problema pode ser modelado por um grafo onde as ruas são as arestas e os vértices são os cruzamentos, sendo preciso encontrar um caminho mais curto que visite cada **aresta** do grafo.

Soluções

- Quando o grafo tem um circuito Euleriano, esse representa a solução ótima. E há algoritmos para encontrar esse circuito, por meio da decomposição do grafo em vários ciclos.
- Quando o grafo não possui o circuito Euleriano é preciso que tenha vértices de grau par (e número de vértices ímpar).
    - transformar o grafo em um grafo euleriano duplicando as arestas que formam o caminho mais curto entre dois vértices de grau ímpar.

Classe de complexidade

- O problema pode ser resolvido em tempo polinomial para grafos e digrafos.
- No entanto, para grafos mistos $G = (V, E, A)$, a solução para esse problema e algumas de suas variantes é NP-Completo.
    - Assim, não existe um algoritmo eficiente que encontra a solução ótima, sendo necessárias aproximações e heurísticas.

Sumário

1. Caminhos
   - Recordando
   - Problema das Pontes de Königsberg e Caminho Euleriano
   - Problema do Caixeiro Viajante e Caminho Hamiltoniano
2. Coloração
   - k-Coloração e Número Cromático
   - Algoritmos e aplicações

Problema do Caixeiro Viajante (PCV)

Definição

- Geralmente chamado de TSP *(Travelling Salesman Problem)*, é um problema de otimização que pode ser modelado por grafos.
- Consiste na procura por um **circuito** (começa e termina no mesmo vértice) que possua a menor distância, que, partindo de uma cidade, visita cada cidade precisamente umav ez e regresse à cidade inicial.

Formulação combinatória

Dado um conjunto de cidades $C = \{c_1, \cdots, c_n\}$ e uma matriz de distâncias $P_{i,j}$ de tamanho $n \times n$, encontrar a permutação entre os camínhos possíveis no qual a soma dos pesos seja a mínima.

Problema do Caixeiro Viajante (PCV)

Complexidade

- O tamanho do espaço de procura aumenta exponencialmente com relação ao número de cidades, pois os circuitos possíveis são:

$$\frac{(n-1)!}{2} \approx \frac{1}{2}\sqrt{2\pi(n-2)}\left(\frac{n-1}{e}\right)^{(n-1)}$$

- Pertence à categoria NP-Completo, com complexidade exponencial.
- Não existe um algoritmo eficiente que encontra a solução ótima, sendo necessárias aproximações e heurísticas, como os métodos de construção de circuitos.
  1. Existem algoritmos rápidos que obtem solução próxima à ótima [11]
  2. Um algoritmo baseado no comportamento de abelhas [10].
  3. Um computador bacterial também foi usado para reduzir a complexidade do problema [9].

Caminho hamiltoniano

Definição

- Um caminho que permite passar por todos os vértices de um grafo G, visitando cada vértice apenas uma vez.
- Caso esse caminho descreva um ciclo, esse é denominado circuito ou ciclo hamiltoniano.
- Um grafo que possua um caminho hamiltoniano é chamado grafo rastreável.
- Um grafo que possua um circuito hamiltoniano é chamado grafo hamiltoniano.

Classe de complexidade

- Relacionado ao TSP, pertence à categoria NP-Completo, com complexidade exponencial.

Sumário

1 Caminhos
- Recordando
- Problema das Pontes de Königsberg e Caminho Euleriano
- Problema do Caixeiro Viajante e Caminho Hamiltoniano

2 Coloração
- k-Coloração e Número Cromático
- Algoritmos e aplicações

Coloração

Definição

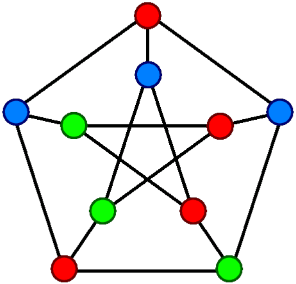
- Seja $G(V, A)$ um grafo, e C um conjunto de cores.
- Uma coloração de G é uma atribuição de alguma cor de C para cada vértice de V, de tal modo que a dois vértices adjacentes sejam atribuídas cores diferentes.
- Uma coloração de G é uma função $f : V \rightarrow C$ tal que para cada par de vértices $(v, w) \in A \rightarrow f(v) \neq f(w)$.
- Uma k-coloração de G é uma coloração que utiliza um total de k cores.

Número cromático

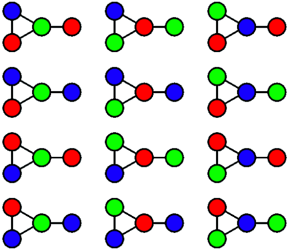
- Denomina-se número cromático $X(G)$ ao menor número de cores k, para o qual existe uma k-coloração de G. Essa é a coloração ótima.

Coloração

- A coloração é um caso especial de rotulação de grafos.
- A coloração pode ser feita em vértices ou também em:
  - **arestas**: arestas adjacentes não possuem a mesma cor.
  - **faces** (grafos planares): faces adjacentes não possuem a mesma cor.



3-coloração do grafo de Petersen com $X(G) = 3$



esse grafo tem 12 possíveis 3-colorações

Coloração

- Atribuir cores distintas a vértices distintos sempre gera uma coloração correta, então podemos dizer que $1 \leq X(G) \leq n$, onde n é o número de vértices.
- Os únicos grafos que podem possuir 1-coloração são aqueles que não possuem arestas.
- ... e os grafos quer requerem n cores são os grafos completos K_n.
- A coloração de vértices usando um algoritmo guloso mostra o número cromático de um grafo relacionado ao vértice de maior grau, mais especificamente: $X(G) \leq \max(\deg(G)) + 1$

Algoritmos

- Determinar se um grafo pode ser colorido com 2 cores é equivalente a determinar se o grafo é bipartido, e portanto computável em tempo linear usando busca em largura.
- Um algoritmo de **força-bruta** para uma k-coloração considera cada uma das k^n atribuições de k cores a n vértices, e verifica se é válida.
- Usando **programação dinâmica** uma k-coloração pode ser decidida em tempo $O(2.455^n)$.
- Um **algoritmo guloso** considera os vértices em uma ordem específica $v_1, \cdots, v_n$ e atribui a v_i a menor cor disponível não usada pelos vértices adjacentes, adicionando uma nova cor se necessário.

Coloração

Aplicações

- **Escalonamento ou agendamento**: um vértice para cada tarefa e uma aresta para cada par de tarefas conflitantes, sendo o número cromático o tempo ótimo para finalizar todas as tarefas sem conflitos.
- **Alocação de registros**: os valores devem ser atribuídos a registros de forma que todos os valores sendo utilizados num dado instante possam estar armazenados em registros, aqui cada vértice é um registro e uma aresta conecta dois vértices (registros) se eles são requeridos em um mesmo instante. Se esse grafo possui uma k-coloração, então as variáveis podem ser armazenadas em k registros.
- **Jogos**: sudoku pode ser visto como completar uma 9-coloração, dado um grafo com 81 vértices

Bibliografia I

- SEDGEWICK, R.
  Algorithms in C: part 5, 3.ed., Addison-Wesley, 2002.
  Graph ADT—Adjacency-Lists Representation (Seções 17.2, 17.3 e 17.4)
- ZIVIANI, N.
  Projeto de Algoritmos, 3.ed. Cengage, 2004.
  (Capítulo 7)
- TENEMBAUM, A.M. et al.
  Estruturas de Dados Usando C. Pearson Makron, 1995.
  Grafos e suas aplicações (Capítulo 8).
- CORMEN, T. H. et al.
  Algoritmos: teoria e prática. Campus-Elsevier, 2002.
  Algoritmos de Grafos (parte IV).

Bibliografia II

KNUTH, D.
The Art of Computer Programming: fundamental algorithms, v.1.
Addison-Wesley, 1969.
Basic Mathematical Properties of Trees (Seção 2.3.4)

FEOFILOFF, P.
Estruturas de Dados.
http://www.ime.usp.br/~pf/algoritmos_para_grafos/aulas/digraphdatastructs.html

FEOFILOFF, P.
Listas de adjacência.
http://www.ime.usp.br/~pf/algoritmos_para_grafos/aulas/adjlists.html

Bibliografia III

FEOFILOFF, P.
Matrizes de adjacência.
http://www.ime.usp.br/~pf/algoritmos_para_grafos/aulas/adjmatrix.html

BAUMGARDNER, J. et al.
Solving a Hamiltonian Path Problem with a bacterial computer.
Journal of Biological Engineering, v.3, n.11, 2009.
http://www.jbioleng.org/content/3/1/11

LIHOREAU, M.; CHITTKA, L; RAINE, N.E.
Travel Optimization by Foraging Bumblebees through Readjustments of Traplines after Discovery of New Feeding Locations. The American Naturalist, v.176, n.6, 2010.
http://www.jstor.org/stable/10.1086/657042

Bibliografia IV

APPLEGATE, D.L. et al
The Traveling Salesman Problem: a computational study.
Princeton, 2006.
http://www.tsp.gatech.edu