Aula 10 – Algoritmo de Fleury e Conectividade em Dígrafos

Notas de Aula de Teoria dos Grafos

Prof[a.]: Patrícia D. L. Machado

UFCG – Unidade Acadêmica de Sistemas e Computação

Sumário

Nesta aula, estudaremos o algoritmo de Fleury e alguns conceitos e propriedades sobre conectividade em grafos direcionados.

Algoritmo de Fleury

Como vimos anteriormente, uma trilha fechada (circuito) que passa por todas as arestas de um grafo exatamente uma vez é chamada de Circuito de Euler. Um grafo que possui um circuito de Euler é chamado de Euleriano. Sabemos que um grafo é Euleriano se e somente si, este grafo é par.

O **Algoritmo de Fleury**, abaixo listado, encontra um circuito de Euler em um grafo Euleriano. O algoritmo recebe como entrada um grafo par conectado *G* e um vértice específico *u* e produz como saída um circuito *W* em *G* que começa e termina em *u*.

1: **Faça** W := (), x:=u, F := G
2: **Enquanto** $\partial_F(\{x\}) \neq \emptyset$ **faça**
3: **Escolha uma aresta** e := xy $\in \partial_F(\{x\})$, onde e não é uma aresta de corte, a menos que não exista outra alternativa.
4: **Faça** W = W + xy, x := y, F := F\e
5: **fim do Enquanto**
6. **Retorne** W

Para tal, inicialmente, W = () – passeio vazio, x marca o vértice atual do passeio e uma cópia de G é feita em F. A estratégia geral é remover de F cada aresta escolhida para integrar o passeio até que fique vazio quando todas as arestas terão sido percorridas.

Da linha 2 a 5, o algoritmo realiza passos repetidos enquanto o corte associado ao conjunto formado pelo vértice x for diferente de vazio. Isto significa que ainda há arestas a serem percorridas a partir de x.

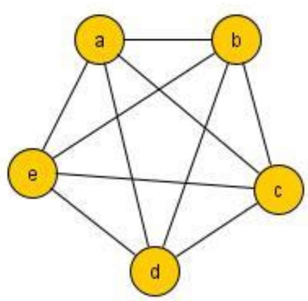
No passo 3, uma aresta qualquer do corte é escolhida para ser adicionada ao passeio, desde que a aresta não seja de corte, a menos que não haja outra alternativa. Como veremos no exemplo que apresentado a seguir, escolher uma aresta de corte pode implicar no encerramento do passeio antes de todas as arestas serem percorridas, porque ela é o único caminho entre seus terminais e o vértice destino pode ter grau 1.

No passo 4, o passeio W é substituído pelo novo passeio com a adição da aresta escolhida xy, a variável x passa apontar para o vértice referenciado por y, destino da aresta e e removemos a aresta e do grafo F.

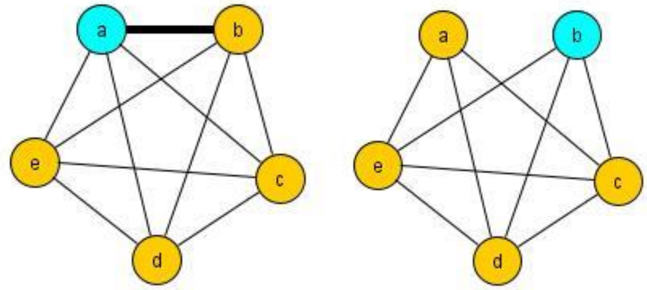
No passo 6, o passeio resultante é retornado.

Teorema. Se G é um grafo par conectado, o passeio W produzido pelo algoritmo de Fleury é um circuito de Euler de G.

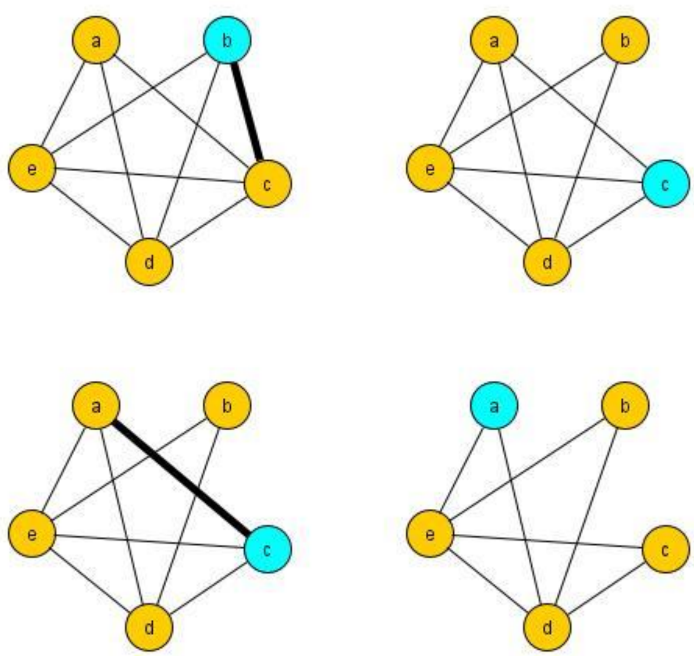
Vejamos agora um exemplo. Considere o grafo G abaixo.



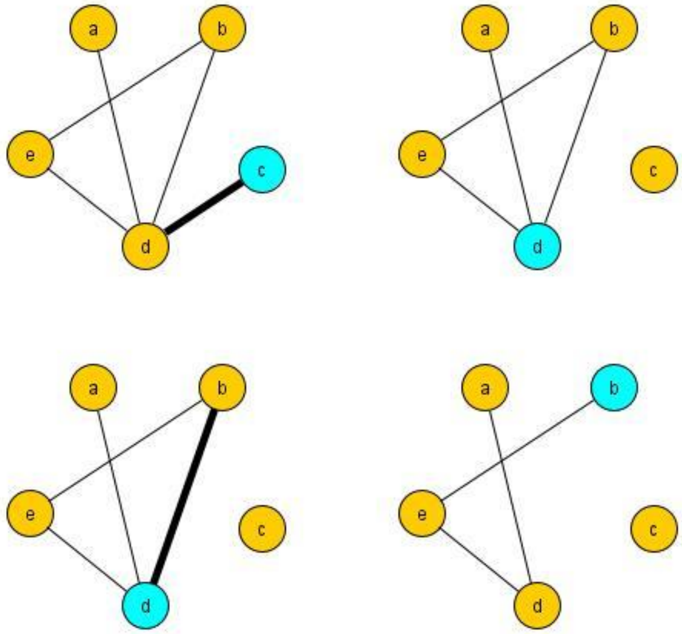
Primeiramente para poder aplicar o algoritmo, precisamos garantir que ele é par e conectado. Em seguida, escolhemos um vértice inicial, neste caso o vértice a, e iniciamos a execução do algoritmo, criando a cópia F apresentada abaixo (grafo à esquerda) e marcando o vértice a como inicial. Na primeira execução do loop, $\partial(\{a\}) = \{ab, ac, ad, ae\}$. Neste corte, não temos aresta de corte. Assim escolhemos a aresta *ab,* esta aresta é removida de *F,* é adicionada ao passeio W e o próximo vértice a ser considerado será b (grafo à direita).



Na próxima execução do loop, $\partial(\{\mathrm{b}\}) = \{\mathrm{be}, \mathrm{bd}, \mathrm{bc}\}$. Neste corte, não temos aresta de corte. Assim, como podemos escolher qualquer aresta deste conjunto, escolhemos a aresta *bc* e o vértice atual do passeio passa a ser o vértice *c,* $W = (ab, bc)$. Na próxima execução do loop, $\partial(\{\mathrm{c}\}) = \{\mathrm{ca}, \mathrm{ce}, \mathrm{cd}\}$. Neste corte, não temos aresta de corte. Escolhemos a aresta *ca* e o vértice atual do passeio passa a ser *a,* $W = (ab, bc, ca)$. Estas duas passagens pelo loop estão ilustradas na figura abaixo.



Após a execução de mais passos do loop, considere que chegamos no grafo *F* abaixo (primeiro a esquerda), sendo *x* = *c* e $W = (ab, bc, ca, ae, ec)$. Note que $\partial(\{\mathrm{c}\}) = \{\mathrm{cd}\}$ e *cd* é uma aresta de corte. Como não temos outra opção, *cd* deve ser escolhida. Assim, *x* = *d* e $W = (ab, bc, ca, ae, ec, cd)$. Agora, $\partial(\{\mathrm{d}\}) = \{\mathrm{de}, \mathrm{db}, \mathrm{da}\}$. Neste caso, note que *da* é uma aresta de corte e não deve ser escolhida, visto que o corte também possui as arestas *de* e *db* que não são de corte e, portanto, uma delas deve ser escolhida. Como resultado, escolhendo *db*, temos que *F* passa a ser o grafo do canto inferior direito, *x* = *b* e $W = (ab, bc, ca, ae, ec, cd, db)$.



Quando todas as arestas tiverem sido removidas de F e adicionadas a W, o algoritmo termina retornando $W = (ab, bc, ca, ae, ec, cd, db, be, ed, da)$. As escolhas dentre arestas que não são de corte em cada passo do loop determinaram o valor final de W. Observe, portanto, que podemos encontrar diferentes circuitos neste grafo usando o algoritmo.

Conectividade em Grafos Direcionados

Passeios, trilhas e circuitos também podem ser definidos para grafos direcionados ou dígrafos. Porém, neste caso, como os arcos possuem direção, não teremos uma relação de equivalência usando o conceito de conectividade, tal como definimos em aulas anteriores para grafos não-direcionados. O fato de que existe um passeio entre um vértice x e um vértice y não indica que existe um passeio também de y para x, quebrando a propriedade de simetria.

Passeio Dirigido

Formalmente, um **passeio dirigido** em um dígrafo *D* é uma sequência alternada de vértices e arcos representada por $W := (a_1,...,a_l)$ tal que chamamos de v_{i-1} e v_i a cauda e a cabeça do arco a_i respectivamente onde *i* varia entre *1* e *l*, sendo *l* é o tamanho do passeio. Em outras palavras, nesta sequência de arcos, o vértice que está na cabeça do arco a_i, é a cauda do arco a_{i+1} e assim por diante. O passeio inicia no vértice a_1 e termina no vértice a_l.

Se *x* é o vértice inicial e *y* é o vértice final no passeio dirigido *W,* então dizemos que *W* é um **(xy)-passeio dirigido.**

Como exemplo, considere o grafo abaixo. W_1 = ((3,2),(2,1),(1,9)) é um passeio dirigido, mas W_2 = ((3,2),(2,1),(7,1)) não é um passeio dirigido, já que a cabeça do segundo arco é diferente da cauda do terceiro.

Corte de Arcos

O conceito de **corte** também precisa ser definido especificamente para dígrafos a fim de considerar o direcionamento dos arcos.

Considerando da mesma forma, que *X* é um conjunto de vértices e *Y* = V/X é o seu complemento, definimos o corte de saída de *X*, representado por $\partial^+(X)$, como o conjunto de arcos que possuem cauda em *X* e cabeça em *Y*. Definimos o corte de entrada, representado por $\partial^-(X)$, como o conjunto de arcos que possuem cauda em *Y* e cabeça em *X*.

Observe que como *X* é o complemento de *Y*, $\partial^+(X) = \partial^-(Y)$.

Se considerarmos o grafo base (ignorando as orientações) $\partial(X) = \partial^+(X) \cup \partial^-(X)$.

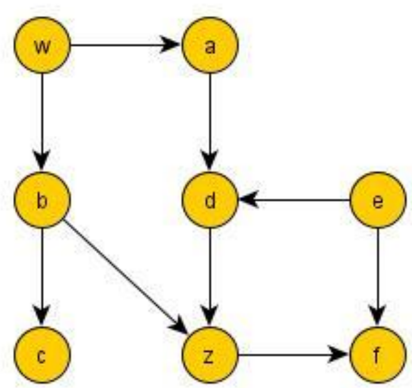
Como exemplo, vamos considerar o grafo abaixo. Observe que:

$\partial^+(\{d,z\}) =\{(z,f)\}$

$\partial^-(\{d,z\}) = \{(a,d),(e,d),(b,z)\}$

$\partial^+(\{d,z\}) = \partial^-(\{w,b,c,a,e,f\})$

$\partial(\{d,z\}) = \partial^+(\{d,z\}) \cup \partial^-(\{d,z\})$



Alcançabilidade

Teorema. Sejam *s* e *t* dois vértices de um dígrafo *D*. Dizemos que *t* é alcançável a partir de *s* se e somente se $\partial^+(X) \neq \emptyset$ para todo subconjunto *X* de *V* que contém *s* mas não *t*.

Intuitivamente isto quer dizer que, se existe um conjunto de vértices *X* onde *s* pertence a *X* e *t* não pertence a *X*, tal que o corte de saída é vazio, não é possível definir um passeio dirigido dos vértices de *X* para o vértice *t*. Para isto, precisaríamos de pelo menos um arco de saída.

Vejamos em nosso exemplo. Considere o grafo acima.

O vértice *z* é alcançável a partir de *w*? Sim, pois para todo *X*, onde $w \in X$ e $z \notin X$, $\partial^+(X) \neq \emptyset$. Um exemplo é o conjunto X formado pelos vértices w e a. Neste caso, $\partial^+(\{w,a\})=\{(w,b),(a,d)\}$.

Vamos agora observar no sentido oposto. O vértice *w* é alcançável a partir de *z*? Não. Existe um *X*, onde $z \in X$ e $w \notin X$, $\partial^+(X) = \emptyset$. Por exemplo, $X = \{z\}$.

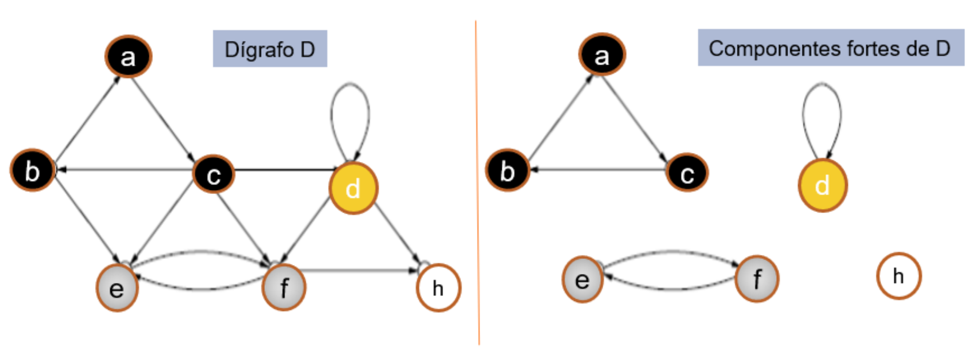
Componentes Fortes

Como comentamos anteriormente, em um dígrafo, o fato de existir um passeio de *x* para *y* não garante que exista um passeio de *y* para *x*, visto que os arcos estabelecem o relacionamento de vértices em apenas uma direção. Por tanto, conectividade em dígrafos é definida de forma a considerar este direcionamento através da observação da existência de passeios dirigidos em ambos os sentidos.

Em um dígrafo *D*, dois vértices *x* e *y* são **fortemente conectados** se existe um *(x,y)*-passeio dirigido e também um *(y,x)*-passeio dirigido.

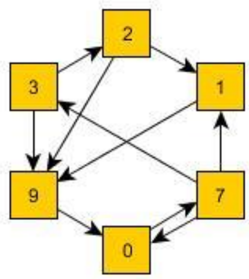
Fortemente conectado é uma relação de equivalência sobre o conjunto de vértices de um digrafo. Os sub-dígrafos são chamados de **componentes fortes** de *D.*

Em nosso exemplo ilustrado abaixo, observamos que os vértices *a*, *b* e *c* são fortemente conectados porque existe um passeio dirigido entre eles em ambos os sentidos. Assim formam um componente forte do dígrafo *D*. Observe também que nenhum outro vértice do grafo é fortemente conectado a *a*,*b* e *c*. Portanto, fazem parte de outros componentes fortes. Os vértices *e* e *f* formam um componente forte e os vértices *d* e *h* que não são fortemente conectados a nenhum vértice formam dois outros componentes fortes de D, isoladamente.



Um dígrafo D é **fortemente conectado** (*strongly connected*) se cada par de vértices de D é fortemente conectado. Abaixo temos um exemplo de um dígrafo fortemente conectado.

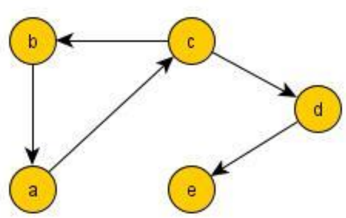
Um dígrafo D é **fracamente conectado** (*weakly connected*) se seu grafo base é conectado.



Na **JGraphT**, encontramos a classe **KosarajuStrongConnectivityInspector** que implementa métodos para determinar se um dígrafo é fortemente conectado, ou seja se possui um único componente forte e também para determinar os componentes fortes de um dígrafo.

A classe **ConnectivityInspector** da JGraphT (que estudamos anteriormente para explorar a conectividade em grafos não-direcionados) pode ser utilizada para determinar se um dígrafo é fracamente conectado, determinando a existência de caminhos entre dois vértices no grafo base.

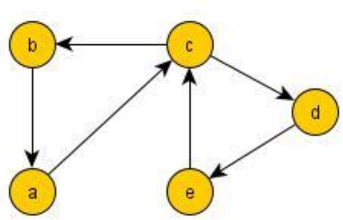
Exercício: Para os dígrafos abaixo, encontre os componentes fortes.



Neste dígrafo, temos os seguintes componentes:

C1 = {a,b,c}, C2={d}, C3={e}

Note que *a,b* e *c* estão conectados entre si; há um passeio dirigido em ambos os sentidos para cada par de vértices. Por outro lado, *d* e *e* não são fortemente conectados a qualquer outro vértice. Portanto, formam novos componentes distintos.



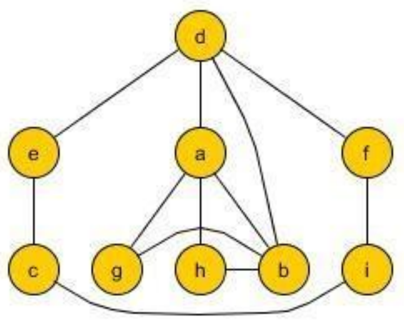
Este dígrafo possui um único componente forte: C = {a,b,c,d,e}. Ou seja, é fortemente conectado.

Exercícios Propostos

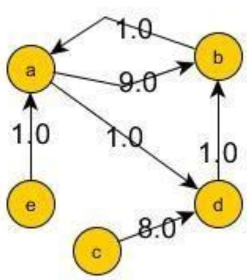
1. Mostre que um dígrafo é forte se e somente se possui exatamente um componente forte.

2. Considerando o Algoritmo de Fleury visto em sala, explique porque no passo 3 existe a recomendação de que uma aresta de corte não deve ser escolhida.

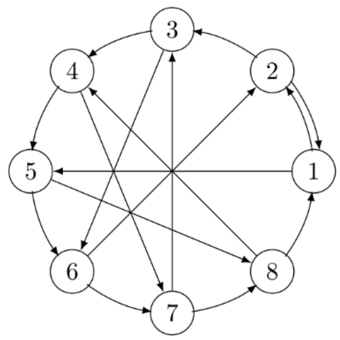
3. Considerando o Algoritmo de Fleury visto em sala, encontre um circuito de Euler neste grafo, tendo o vértice "a" como inicial.



4. Encontre os componentes fortes deste grafo.



5. Considere uma adaptação do algoritmo de Fleury para dígrafos, onde o próximo arco a ser escolhido deve ser de saída para o vértice sendo visitado a fim de possibilitar a definição de um passeio dirigido. Encontre um circuito dirigido de Euler para este grafo, tendo como vértice inicial o vértice "1".
Grafo obtido de: https://math.stackexchange.com/a/460641



Referências

J. A. Bondy and U. S. R. Murty. Graph Theory. Springer, 2008,2010.

3.1, 3.3, 3.4

http://www.pathuku.com (Jogo - Circuito de Euler)