

Heurística da arborescência geradora mínima

Para guiar a explicação da heurística toma-se, como exemplo, o digrafo G , a seguir:

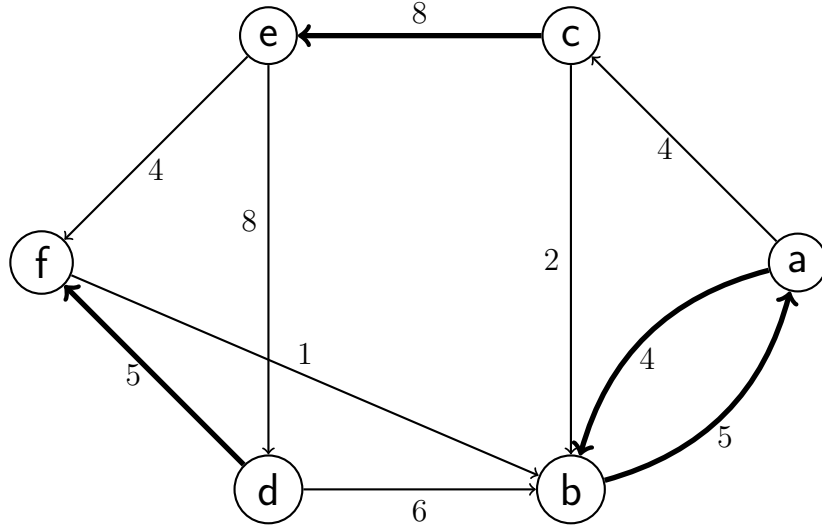


Figura 1: Digrafo G , o conjunto R corresponde aos arcos em negrito

A extensão de G em um digrafo euleriano $G' = (V', A')$ se dá de modo similar àquela do caso com grafos não direcionados:

O conjunto de vértices possui a mesma definição, $V' = \{u \in V : (u, v) \in R \text{ para algum } v \in V\}$.

Como no grafo da figura ?? todos vértices possuem ao menos um arco pertencente à R , define-se $V' = V$.

De modo semelhante, o conjunto de arcos A' será inicialmente acrescido dos arcos (v_i, v_j) para todo par $v_i, v_j \in V'$, de custo igual ao custo do menor caminho de v_i a v_j . Posteriormente remove-se de A' todo arco que pertence também a $A \setminus R$ e cujo custo c_{uv} é igual a $c_{uw} + c_{wv}$ para algum vértice w , removem-se também os arcos paralelos de mesmo valor que pertencem a $A \setminus R$.

Realizando tal extensão no exemplo sugerido, teremos o seguinte digrafo G' :

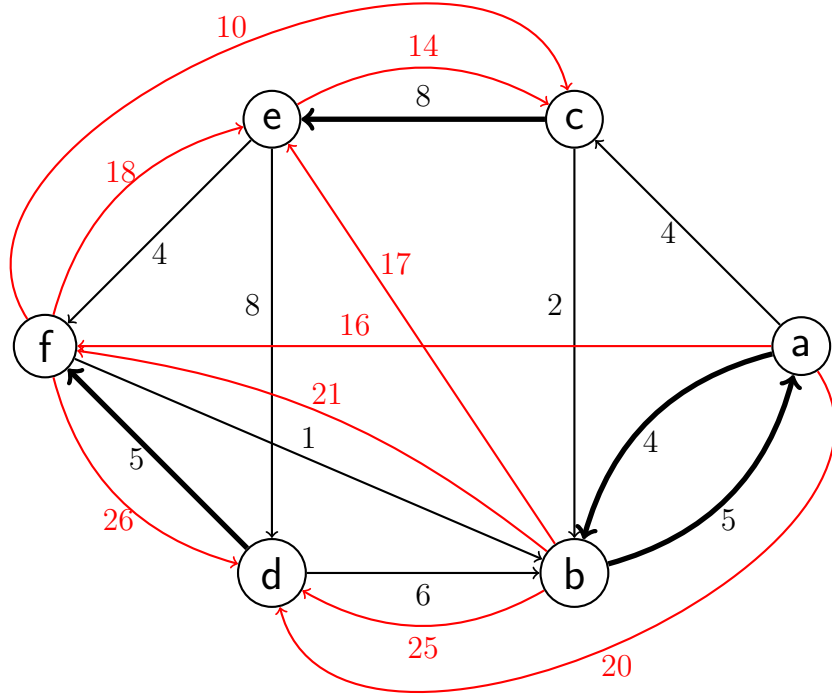


Figura 2: Digrafo G' , extensão de G , as arestas criadas na extensão são mostradas em vermelho.

O grafo G' será composto por k conjuntos conexos G_1, G_2, \dots, G_k , induzidos por R . Isto é, cada conjunto G_i será composto apenas por arcos pertencentes a R e será conexo, mas não necessariamente fortemente conexo.

No exemplo apresentado, G possui três componentes, G_1 composta pelos vértices a, b , G_2 composta por c, e e G_3 composta por d, f .

Passo 1. Encontrar uma arborescência geradora T de custo mínimo que conecte os subgrafos G_1, \dots, G_k e tem raiz em um vértice qualquer.

Para encontrar tal arborescência, pode-se utilizar o algoritmo proposto independentemente pela dupla Yoeng-Jin Chu e Tseng-Hong Liu

e por Edmonds[13], sendo, portanto, chamado de algoritmo de Chu-Liu/Edmonds.

Condensando o digrafo G' em suas componentes G_1, G_2, G_3 temos o seguinte grafo:

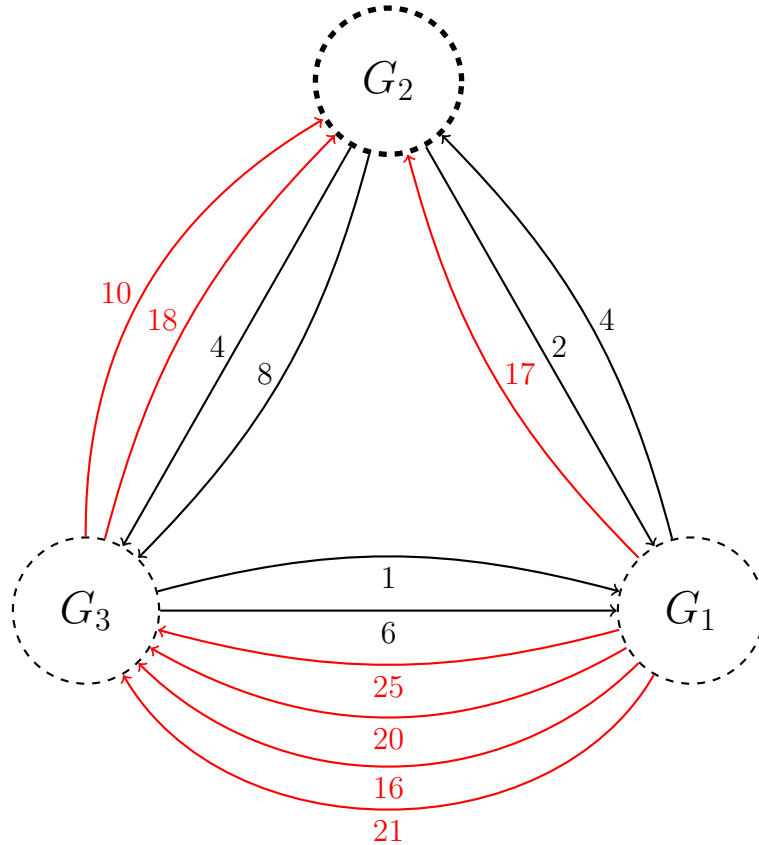


Figura 3: Digrafo G' condensado em suas componentes induzidas por R

O primeiro passo do algoritmo de Chu-Liu/Edmonds consiste em definir um vértice raiz r para a arborescência. Como a heurística de Christofides não requer que um vértice específico seja tal raiz, podemos escolher o vértice condensado G_1 para cumprir tal função ($r = G_1$).

Definida a raiz r , deve-se retirar do digrafo analisado todos arcos que tem como destino r . Além disso, pode-se também substituir qualquer conjunto de arcos paralelos por um único arco com custo igual ao menor custo dos arcos paralelos removidos. Pode-se visualizar o efeito de tais modificações em G' na figura ??.

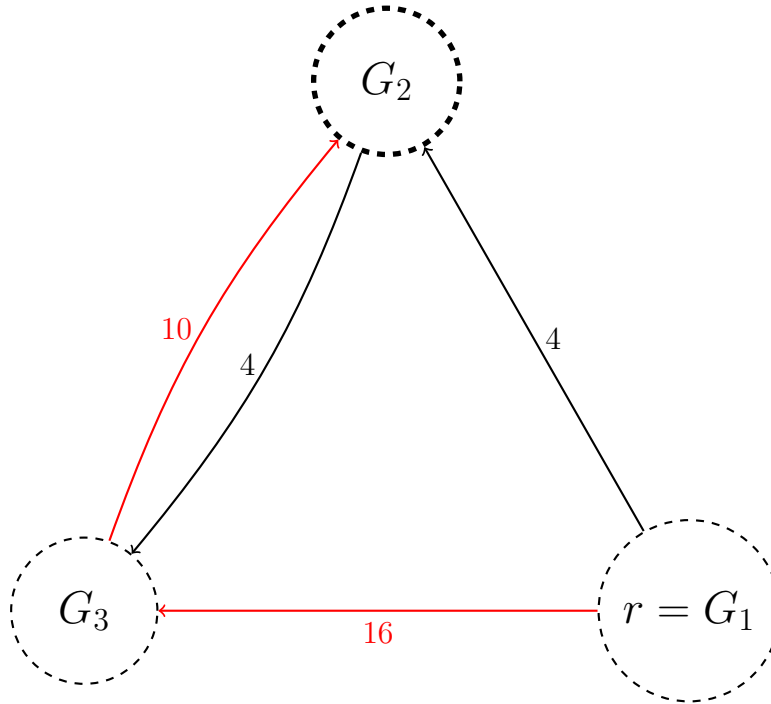


Figura 4: Digrafo G' após a remoção de arcos sugeridas pelo algoritmo de Chu-Liu/Edmonds

Para todo vértice v do grafo condensado diferente da raiz ($v \neq r$) encontra-se o arco de menor custo que chega em v . Define-se como $\pi(v)$ o vértice origem de tal arco.

Se o conjunto de arcos $T = \{(\pi(v), v) | v \neq r\}$ não contem circuitos, então T é uma arborescência de custo mínimo enraizada em r . Do contrário, realiza-se uma contração dos circuitos existentes em T , atualizam-se os custos dos arcos e repete-se recursivamente o mesmo procedimento de criação de T .

No exemplo analisado, o conjunto T consiste nos arcos de G_1 a G_2 e G_2 a G_3 , ambos de custo 4.

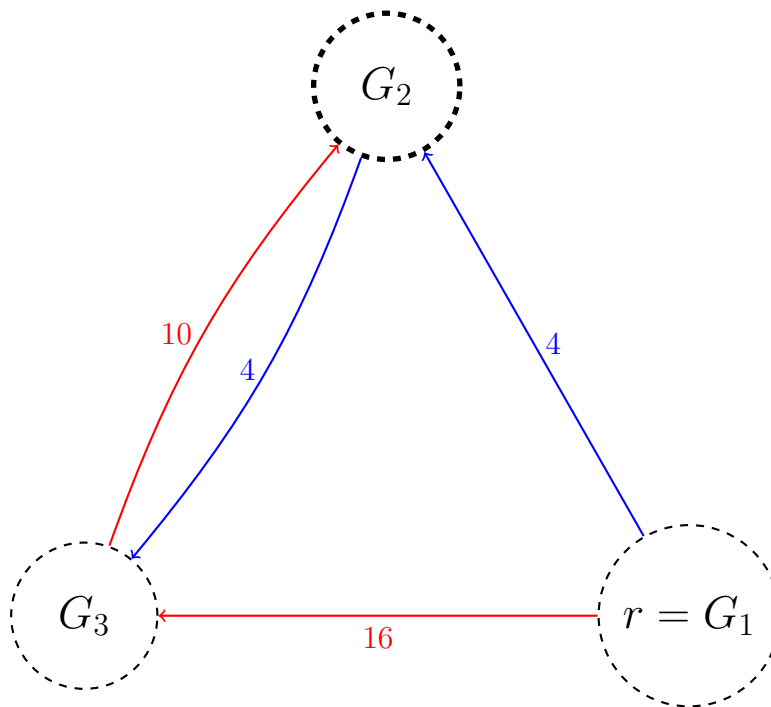


Figura 5: Conjunto de arcos T definidos pelo algoritmo de Chu-Liu/Edmonds em azul.

Como T não possui circuitos, não é necessário realizar uma nova iteração do algoritmo. T é o conjunto de arcos que induz a arborescência de custo mínimo enraizada em G_1 .

Define-se como $R \cup T$ o digrafo induzido pelos arcos de R e da arborescência T .

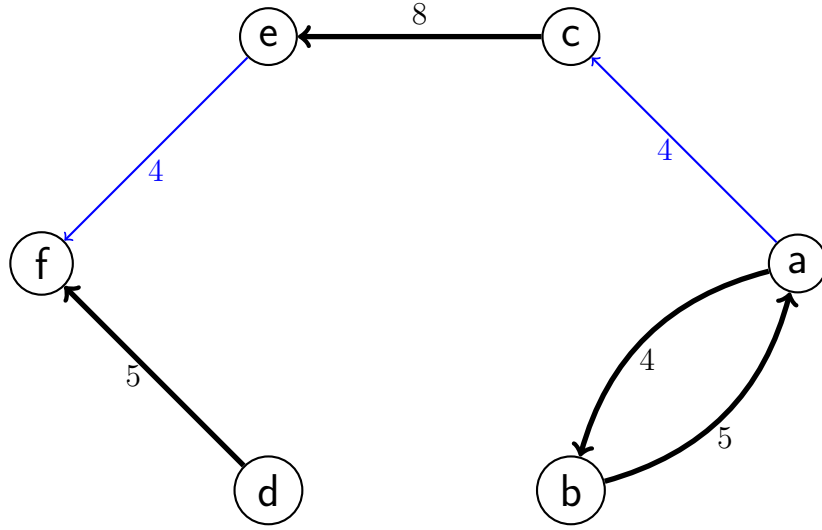


Figura 6: Digrafo $R \cup T$, os arcos de T são representados em azul

Passo 2. Encontrar um multiconjunto M de menor custo composto por arcos de A' que torna o digrafo induzido por $R \cup T$ euleriano, ou seja iguala os graus de entrada e saída de todos vértices.

Pode-se determinar M a partir da resolução de um problema de transporte: O problema de transporte será definido tendo como base o grafo G' , porém as funções de oferta e demanda são definidas a partir dos graus dos vértices no grafo induzido por $R \cup T$.

Um vértice $v \in V(G')$ cujo grau de entrada é maior que seu grau de saída (em relação ao subgrafo $R \cup T$) possuirá uma oferta igual ao valor absoluto da diferença de seus graus, do contrário, o valor absoluto da diferença representará a demanda do vértice v .

No exemplo abordado, figura ??, o vértice f possui uma oferta de valor 2, os vértices a e d possuem uma demanda de valor 1 e os vértices restantes já se encontram em igualdade de grau de entrada e saída.

A resolução do problema de transporte modelado gera um conjunto de caminhos, representando a distribuição ótima da oferta. O multiconjunto dos arcos pertencentes a união dos caminhos que solucionam o problema é o multiconjunto M desejado.

Sendo assim, como apenas uma origem existe no problema apresentado como exemplo, a solução do mesmo consiste na utilização dos caminhos de menor custo de f a a e de f a d .

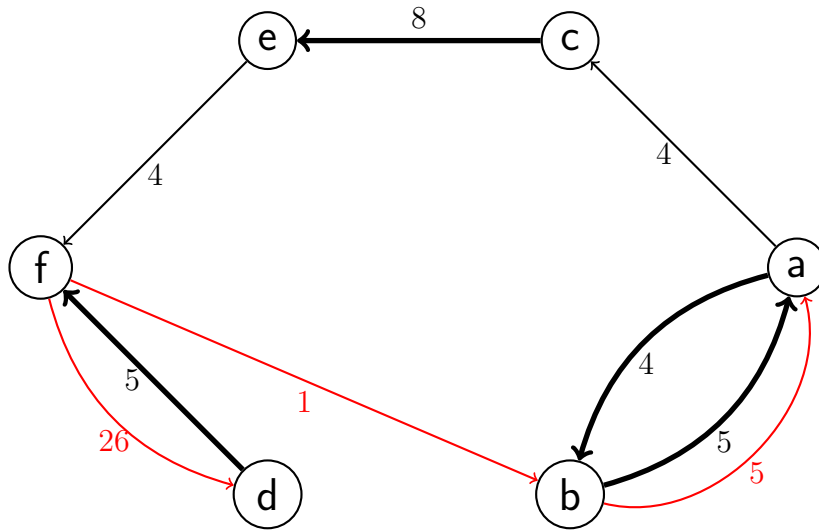


Figura 7: Digrafo eurleriano $R \cup T \cup M$ com arcos de M representados em vermelho.

Na figura ?? pode-se visualizar o digrafo induzido pelos arcos $R \cup T \cup M$. Note que o arco de b a a de custo 5 é representado duas vezes, isso pois ele pertence tanto a R quanto a M .

Passo 3. O digrafo induzido pelo multiconjunto de arcos $R \cup T \cup M$ é, pela definição de M , euleriano.

Portanto, pode-se derivar um circuito euleriano de tal grafo. Por sua vez, a partir deste circuito, é possível derivar uma solução do PCR para o grafo original G .

Um possível circuito euleriano para o exemplo apresentado é representado na figura ?? a seguir.

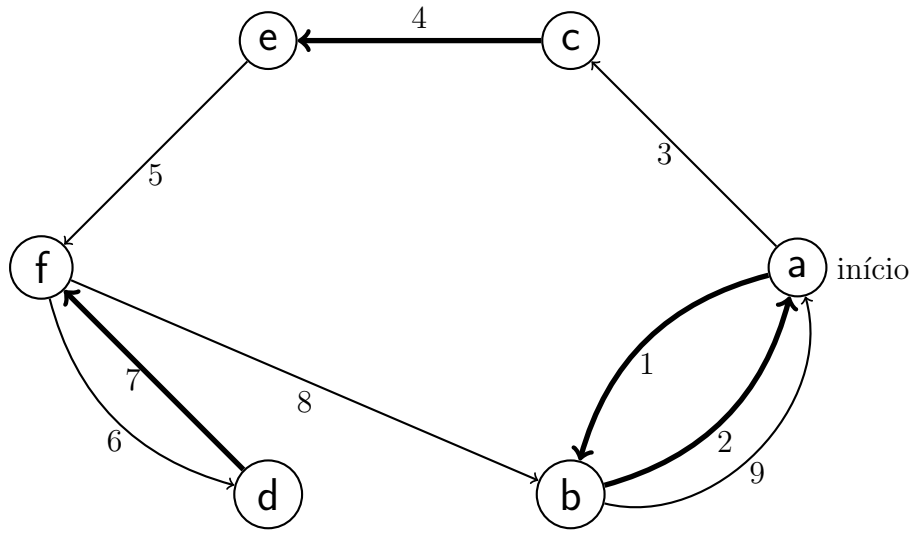


Figura 8: Circuito euleriano de $R \cup T \cup M$.

A partir de tal circuito pode-se encontrar uma solução para o PCR do digrafo original G expandindo os arcos contraídos na construção de G' e removendo as duplicatas de arcos criados no procedimento da heurística. Por exemplo, o arco do nó f ao d de custo 26 pertencente a M consiste na condensação do caminho mínimo de f a d : $\{f, b, a, c, e, d\}$.

Realizando a remoção dos arcos artificiais adicionados a G chegamos no seguinte circuito, baseado no circuito euleriano mostrado na figura ??:

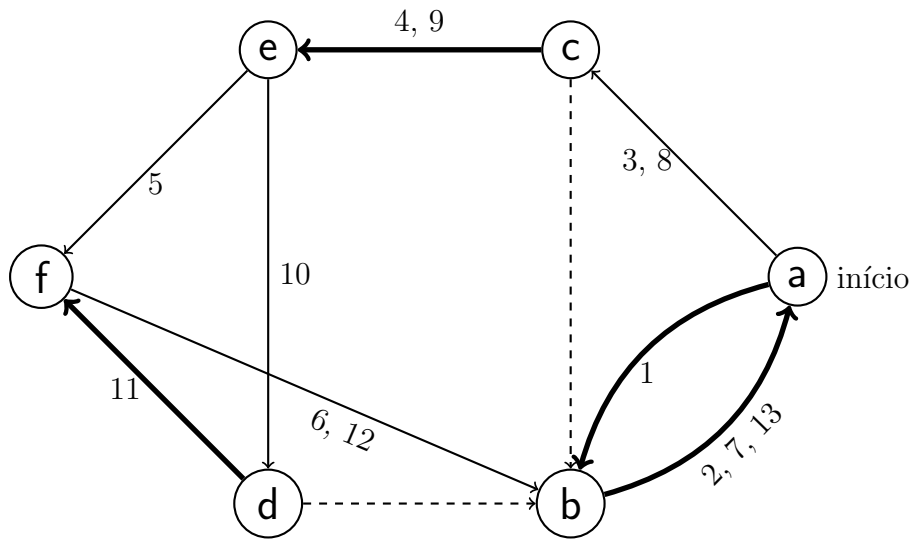


Figura 9: Construção de um circuito que resolve o PCR do grafo G . Representam-se em pontilhado os arcos de G não percorridos na solução desenvolvida.

Finaliza-se assim a execução da heurística da arborescência geradora mínima, que gera uma solução para o PCR de G com custo 62.

Referências

- [1] Euler, Leonhard *Solution problematis ad geometriam situs pertinentis*. Comment. Acad. Sci. U. Petrop 8, 128–40, 1736.
- [2] Hierholzer, Carl “Über die Möglichkeit, einen Linienzug ohne Wiederholung und ohne Unterbrechung zu umfahren”, Mathematische Annalen, 6 (1): 30–32, doi:10.1007/BF01442866, 1873.
- [3] Problema D do round #288 (Div. 2) retirado do Codeforces codeforces.com/contest/508/problem/D
- [4] Solução para o problema Tanya and Password, desenvolvida em C++ github.com/gafeol/competitive-programming/blob/master/ojs/cf/508/D.cpp
- [5] Problema C do round #215 (Div. 1) retirado do Codeforces codeforces.com/problemset/problem/367/C

- [6] Solução para o problema Sereja and the Arrangement of Numbers, desenvolvida em C++
github.com/gafeol/competitive-programming/blob/master/ojs/cf/367/C.cpp
- [7] Problema 10296 retirado do UVa
onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&page=show_problem&problem=1237
- [8] Solução para o problema Jogging Trails, desenvolvida em C++
github.com/gafeol/competitive-programming/blob/master/ojs/UVa/1237.cpp
- [9] Exemplo retirado do site do MIT, exercício 6.6.c
web.mit.edu/urban_or_book/www/book/chapter6/problems6/6.6.html
- [10] Página da wikipedia sobre o algoritmo de Christofides
wikipedia.org/wiki/Christofides_algorithm
- [11] Artigo “Arc Routing Problems Part II: The Rural Postman Problem” publicado por Michel Gendreau e Gilbert Laporte. pubsonline.informs.org/doi/10.1287/opre.43.3.399
- [12] Christofides, Nicos, et al. ”An algorithm for the rural postman problem on a directed graph.” Netflow at pisa. Springer, Berlin, Heidelberg, 1986. 155-166.
link.springer.com/chapter/10.1007/BFb0121091
- [13] nvlpubs.nist.gov/nistpubs/jres/71B/jresv71Bn4p233_A1b.pdf