CTC – 20
Estruturas Discretas para Computação

Prof. Armando Gouveia

Algoritmo de Kruskal

Árvores geradoras de custo mínimo

Problema:

Dado um grafo *G* conexo com custos nas arestas
encontrar *T* uma árvore geradora de *G*
de custo mínimo.
Onde custo (*T*) = soma dos custos das arestas de *T*.

Aplicações?

- Circuitos eletrônicos
- Internet?
- Conexões via fibra ótica
- É o modo "mais barato" de tornar conexo.

Observações

Minimum Spanning Tree

Árvore espalhada mínima

Árvore geradora de custo mínimo

Como resolver?

Estratégia “greedy”

= ganancioso = guloso ☺

Algoritmo de Kruskal

Entrada:

$G = (VG, AG)$ conexo

$f : AG \rightarrow IR$ /* função custo */

Saída:

$T = (VT, AT)$ /* árvore T geradora de G com $f(T)$ mínimo */

Algoritmo de Kruskal

```
Início:
ordene as arestas de G segundo os custos.
   Supor que sejam a_1, a_2, ..., a_m
   tais que f(a_1) ≤ f(a_2) ≤ ... ≤ f(a_m).
```

Início:

ordene as arestas de G segundo os custos.

  Supor que sejam $a_1, a_2, \ldots, a_m$

  tais que $f(a_1) \le f(a_2) \le \ldots \le f(a_m)$.

$VT \leftarrow VG;$

$AT \leftarrow \varnothing;$

para i = 1,...,m

  se $G[AT \cup \{a_i\}]$ é acíclico

    então $AT \leftarrow AT \cup \{a_i\}$

return $T = (VT, AT);$

Fim.

Teorema

O subgrafo criado pelo Algoritmo de Kruskal
é uma árvore geradora de custo mínimo.

Demonstração

(i) T é acíclica.
Óbvio.

(ii) T é conexa.
Se não fosse, poderíamos acrescentar alguma aresta
sem formar circuito.

(iii) $\left.\begin{array}{l}\text{Temos } VT = VG \\ \text{e, por (i) e (ii), } T \text{ é árvore}\end{array}\right\}$ portanto T é árvore geradora de G.

(iv) Falta demonstrar que T possui custo mínimo.

Seja $n = |VG|$.

Seja T_{min} uma árvore geradora de G de custo mínimo

tal que $s = |\, AT \cap AT_{min} \,|$ seja máximo.

Se $s = n - 1$,

então $T = T_{min}$ e, portanto, T é mínima.

Suponhamos, então que $s < n - 1$.

Seja $a \in AT - AT_{min}$ de menor custo.

Seja C o único circuito de $T_{min} + a$.

Por que existe?

Por que é único?

Como T é acíclico, então $AC - AT \neq \varnothing$

Seja $a_{min} \in AC - AT$.

$(\longrightarrow)$

Seja $a_{min} \in AC - AT$.

Afirmação: $f(a) \geq f(a_{min})$.

Prova: Suponha que $f(a) < f(a_{min})$.

Neste caso, considere $T_{min} + a - a_{min}$.

Observe que esse subgrafo é uma árvore geradora de G

cujo custo é estritamente menor que o custo de T_{min}. [Contradição]

Afirmação: $f(a) \leq f(a_{min})$.

Prova: $a_{min} \notin AT$

Isso significa que, em certo instante no Algoritmo de Kruskal,

quando tentamos adicionar a aresta a_{min} ao conjunto AT,

notamos que isso levaria à formação de circuito.

Mas o algoritmo considera as arestas em ordem crescente de custo;

então as arestas de tal circuito custavam no máximo o custo de a_{min}.

Logo $f(a) \leq f(a_{min})$.

Das afirmações anteriores, concluímos que $f(a) = f(a_{min})$.

Agora consideremos $T' = T_{min} + a - a_{min}$.

Então T' é árvore geradora

(Prove ☺)

Temos custo (T') = custo (T_{min}).

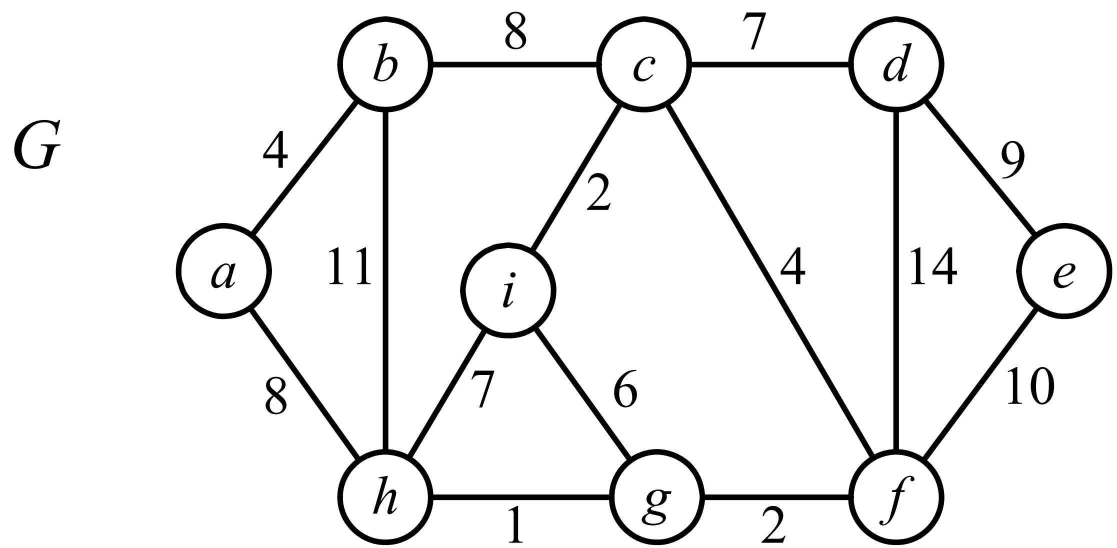
E reparemos que $| AT \cap AT_{min} | > s$.

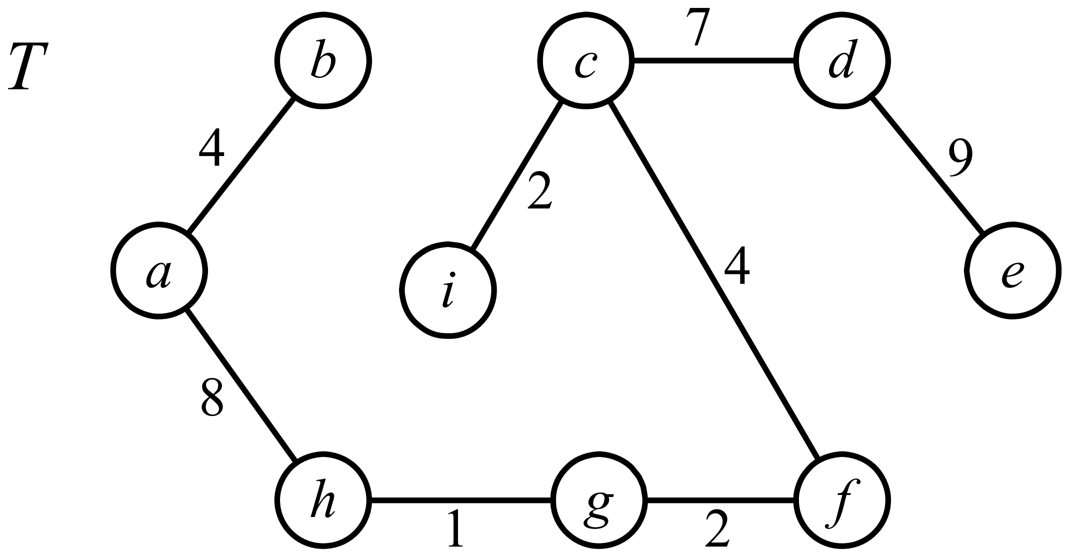
Contradição!

pois escolhemos T_{min} tal que s fosse máximo.

[CQD]

Simulação





custo $(T) = 37$