O Problema Online Steiner Tree Star

Autor: Mário César San Felice
Orientador: Orlando Lee

22 de novembro de 2012, IC-Unicamp

O GSTS foi proposto por Khuller e Zhu [3] e é definido num:

O GSTS foi proposto por Khuller e Zhu [3] e é definido num:

- Grafo completo $G = (V, E)$,

O Problema General Steiner Tree Star

O GSTS foi proposto por Khuller e Zhu [3] e é definido num:

- Grafo completo $G = (V, E)$,
- Pesos nas arestas $w_{ij} : E \rightarrow R^{+}$,

O Problema General Steiner Tree Star

O GSTS foi proposto por Khuller e Zhu [3] e é definido num:

- Grafo completo $G = (V, E)$,
- Pesos nas arestas $w_{ij} : E \rightarrow R^{+}$,
- Pesos nos vértices $W_i : V \rightarrow R^{+}$,

O Problema General Steiner Tree Star

O GSTS foi proposto por Khuller e Zhu [3] e é definido num:

- Grafo completo $G = (V, E)$,
- Pesos nas arestas $w_{ij} : E \to R^+$,
- Pesos nos vértices $W_i : V \to R^+$,
- Vértices terminais $X \subseteq V$,

O Problema General Steiner Tree Star

O GSTS foi proposto por Khuller e Zhu [3] e é definido num:

- Grafo completo $G = (V, E)$,
- Pesos nas arestas $w_{ij} : E \rightarrow R^+$,
- Pesos nos vértices $W_i : V \rightarrow R^+$,
- Vértices terminais $X \subseteq V$,
- Vértices de Steiner $Y \subseteq V$.

O GSTS foi proposto por Khuller e Zhu [3] e é definido num:

- Grafo completo $G = (V, E)$,
- Pesos nas arestas $w_{ij} : E \to R^{+}$,
- Pesos nos vértices $W_i : V \to R^{+}$,
- Vértices terminais $X \subseteq V$,
- Vértices de Steiner $Y \subseteq V$.

O objetivo é encontrar uma árvore T que conecta todos os terminais e minimiza o custo:

$$\sum_{e=(i,j)\in T} w_{ij} + \sum_{j\in V(T):\delta(j)>1} W_j$$

O Problema Steiner Tree Star

O problema Steiner Tree Star (STS) é definido como o GSTS com a restrição adicional $X \cap Y = \emptyset$.

O Problema Steiner Tree Star

O problema Steiner Tree Star (STS) é definido como o GSTS com a restrição adicional $X \cap Y = \emptyset$.

Com uma simples mudança no grafo G é possível mostrar que GSTS e STS são equivalentes.

O problema OSTS é a versão online do STS. Nela os terminais chegam um por vez e cada um que chega deve ser atendido antes que o próximo seja revelado.

O problema OSTS é a versão online do STS. Nela os terminais chegam um por vez e cada um que chega deve ser atendido antes que o próximo seja revelado.

Além disso, as decisões tomadas são irrevogáveis. Neste problema isso significa que arestas e vértices colocadas na árvore não podem ser removidos

Análise Competitiva

Abordagem que usamos para analisar a qualidade dos algorimos online.

Análise Competitiva

Abordagem que usamos para analisar a qualidade dos algorimos online.

Dizemos que ALG é c-competitivo se, para toda entrada I e algum α constante que não depende de I, ele apresenta razão de competitividade c que satisfaz:

Análise Competitiva

Abordagem que usamos para analisar a qualidade dos algorimos online.

Dizemos que ALG é c-competitivo se, para toda entrada I e algum α constante que não depende de I, ele apresenta razão de competitividade c que satisfaz:

$$ALG(I) \leq cOPT(I) + \alpha.$$

Mapeando OSTS para OFL

Podemos mapear uma entrada do OSTS para o Online Facility Location (OFL) com as seguintes equivalências:

Mapeando OSTS para OFL

Podemos mapear uma entrada do OSTS para o Online Facility Location (OFL) com as seguintes equivalências:

- $G \leftrightarrow \mathcal{M}$,

Mapeando OSTS para OFL

Podemos mapear uma entrada do OSTS para o Online Facility Location (OFL) com as seguintes equivalências:

- $G \leftrightarrow \mathcal{M}$,
- $w_{ij} \leftrightarrow d(i,j)$,

Mapeando OSTS para OFL

Podemos mapear uma entrada do OSTS para o Online Facility Location (OFL) com as seguintes equivalências:

- $G \leftrightarrow \mathcal{M}$,
- $w_{ij} \leftrightarrow d(i,j)$,
- $W_i \leftrightarrow f(i)$,

Mapeando OSTS para OFL

Podemos mapear uma entrada do OSTS para o Online Facility Location (OFL) com as seguintes equivalências:

- $G \leftrightarrow \mathcal{M}$,
- $w_{ij} \leftrightarrow d(i,j)$,
- $W_i \leftrightarrow f(i)$,
- $X \leftrightarrow D$,

Mapeando OSTS para OFL

Podemos mapear uma entrada do OSTS para o Online Facility Location (OFL) com as seguintes equivalências:

- $G \leftrightarrow \mathcal{M}$,
- $w_{ij} \leftrightarrow d(i,j)$,
- $W_i \leftrightarrow f(i)$,
- $X \leftrightarrow D$,
- $Y \leftrightarrow F$.

Algoritmo Online STS

A seguir descrevemos um algoritmo para o OSTS que usa como subrotina o algoritmo *compFL*, proposto por Fotakis [4], e simula o algoritmo *compST*, proposto por Imase e Waxman [1].

Algoritmo Online STS

A seguir descrevemos um algoritmo para o OSTS que usa como subrotina o algoritmo *compFL*, proposto por Fotakis [4], e simula o algoritmo *compST*, proposto por Imase e Waxman [1].

Online STS Algorithm:

Map the STS instance to a FL instance;
$D \leftarrow \emptyset$; $F' \leftarrow \emptyset$; $T \leftarrow \emptyset$;
while a new client j arrives **do**
  send j to *compFL*;
    if a new facility i was opened **then**
      send i to *compST*;
      $F' \leftarrow F' \cup \{i\}$;
      $T \leftarrow T \cup path(i, V(T))$;
  let i be the closest facility to j;
  $D \leftarrow D \cup \{j\}$; $a(j) \leftarrow i$;
return (F', T, a).

Análise do Algoritmo Online STS

Na análise dividimos o custo do algoritmo em:

$$ALG_{OSTS}(D) = O + C + S,$$

Análise do Algoritmo Online STS

Na análise dividimos o custo do algoritmo em:

$$ALG_{OSTS}(D) = O + C + S,$$

sendo O o custo dos vértices internos, C o custo das arestas conectando os terminais aos vértices internos e S o custo da árvore conectando os vértices internos entre si.

Análise do Algoritmo Online STS

Na análise dividimos o custo do algoritmo em:

$$ALG_{OSTS}(D) = O + C + S,$$

sendo O o custo dos vértices internos, C o custo das arestas conectando os terminais aos vértices internos e S o custo da árvore conectando os vértices internos entre si.

Nós comparamos este algoritmo com o algoritmo ótimo para o STS, cujo custo denotamos:

$$OPT_{STS}(D) = O^* + C^* + S^*.$$

Análise do Algoritmo Online STS (cont.)

Lemma

A árvore T custa $S \in O(\log n(S^* + C^* + C))$.

Análise do Algoritmo Online STS (cont.)

Lemma

A árvore T custa $S \in O(\log n(S^* + C^* + C))$.

Lemma

Os vértices internos e a conexão dos terminais a estes custa $(O + C) \in O(\log n(O^* + C^*))$.

Análise do Algoritmo Online STS (cont.)

Lemma

A árvore T custa $S \in O(\log n(S^* + C^* + C))$.

Lemma

Os vértices internos e a conexão dos terminais a estes custa $(O + C) \in O(\log n(O^* + C^*))$.

Theorem

As soluções do algoritmo Online STS custam $ALG_{OSTS}(D) \in O(\log^2 n OPT_{STS}(D))$.

Limitante Inferior para o OSTS

É conhecido um limitante inferior de $\Omega(\log n)$ para a competitividade do Online Steiner Tree (OST).

Limitante Inferior para o OSTS

É conhecido um limitante inferior de $\Omega(\log n)$ para a competitividade do Online Steiner Tree (OST).

Este limitante se aplica ao OSTS, pois qualquer instância do OST é uma instância do OSTS cujos vértices tem custo zero.

Referências

M. Imase and M.B. Waxman.
Dynamic Steiner Tree Problem.
SIAM Journal on Discrete Mathematics, 369–384, 1991.

A. Meyerson.
Online Facility Location.
FOCS, 426, IEEE, 2001.

S. Khuller and A. Zhu.
The General Steiner Tree-Star Problem.
Inf. Process. Lett., 215–220, Elsevier, 2002.

D. Fotakis.
A Primal-Dual Algorithm for Online Non-Uniform Facility Location. JDA, 141–148, Elsevier, 2007.

C. Nagarajan and D.P. Williamson.
Offline and Online Facility Leasing.
IPCO, 303–315, Springer-Verlag, 2008.

Agradecimentos

Obrigado!

Agradecimentos

Obrigado!

Dúvidas?