# 树上推广的 Multicut 问题的近似算法

张　鹏
（山东大学计算机科学与技术学院　济南　250101）
（algzhang@gmail.com）

# Approximation Algorithms for Generalized Multicut in Trees

Zhang Peng
(*School of Computer Science and Technology*, *Shandong University*, *Jinan* 250101)

**Abstract**　Given a tree $T$ with costs on edges and a collection of terminal sets $X=\{S_1, S_2, \cdots, S_l\}$, the generalized multicut in trees problem asks to find a set of edges on $T$ whose removal cuts every terminal set in $X$, such that the total cost of the picked edges is minimized. The problem has its own interest since it naturally generalizes the classical multicut problem and the multiway cut problem, respectively, and also is the dual of the generalized Steiner forest problem. It is shown that the full version of the problem can be approximated within a factor of 2 by reducing it to the classical multicut in trees problem. In the prize-collecting version of the problem, a terminal set must pay the penalty $\pi_i$ if it is not cut by the picked edges. A primal-dual 3-approximation algorithm is given for the prize-collecting generalized multicut in trees problem via primal-dual schema. The $k$-version of the problem has to cut at least $k$ terminal sets at the minimum cost, where $k$ is a given parameter. It is shown that the $k$-version of the problem can be approximated within a factor of $\min\{2(l-k+1), k\}$ via a non-uniform approach. Furthermore, it is shown that there is an interesting relation between the generalized $k$-multicut in trees problem and the dense $k$-subgraph problem, implying that approximating the $k$-version of the problem within $O(n^{1/6-\varepsilon})$ for some small $\varepsilon>0$ is difficult.

**Key words**　multicut; tree; linear programming; approximation algorithm; combinatorial optimization

**摘　要**　给定边上有费用的树 $T$，终端集合族 $X=\{S_1, S_2, \cdots, S_l\}$，推广的 Multicut 问题询问费用最小的边集，使得在树上删除边集中的边能够断开每一个终端集. 推广的 Multicut 问题有其独立的研究意义，因为该问题分别是经典的 Multicut 问题和 Multiway Cut 问题的自然推广，同时也是推广的 Steiner Forest问题的对偶问题. 树上推广的 Multicut 问题的完全版本可以归约到树上经典的 Multicut 问题近似求解. 对于该问题的 Prize-collecting 版本，给出了原始-对偶的 3 倍近似算法. 对于该问题的 $k$ 版本，通过非一致的途径给出了近似比为 $\min\{2(l-k+1), k\}$的近似算法. 以及找到了该问题的 $k$ 版本与 $k$-稠密子图问题之间的一个有趣的联系，从而证明了将 $k$ 版本的树上推广的 Multicut 问题近似到 $O(n^{1/6-\varepsilon})$以内是困难的（对某个小的常数 $\varepsilon>0$）.

**关键词**　Multicut；树；线性规划；近似算法；组合优化

**中图法分类号**　TP301.6

割集问题是组合优化领域中经典并且活跃的主题. 割集问题在总体上可分为两大类：一类是最小割集问题，该问题询问图上断开源顶点和目标顶点的费用最小的割集；另一类是最大割集问题，该问题

**收稿日期**：2007-01-11；**修回日期**：2008-03-20
**基金项目**：国家杰出青年基金项目（60325206）；国家自然科学基金项目（60310213）

询问容量最大的割集，将图划分成两个(或几个)不连通的子图. 本文提出并研究树上推广的 Multicut 问题，属于最小割集问题. 给定边上有费用的无向图 $G=(V,E)$，终端集合族 $X=\{S_1,S_2,\cdots,S_l\}$，推广的 Multicut 问题(generalized multicut，GMC)询问边集 $E$ 的一个子集，使得在图上删除该子集中的边能够断开每一个终端集，且所选择边的总费用最小. 其中，终端集 $S_i\in X$ 是顶点的集合，遵照惯例，$S_i$ 中的顶点称为终端. 使用符号 GMC(G)表示一般(无向)图上推广的 Multicut 问题，使用符号 GMC(T)表示树上推广的 Multicut 问题. 推广的 Multicut 问题有其独立的研究意义，因为该问题分别是经典的 Multicut 问题和 Multiway Cut 问题的自然推广，同时又是推广的 Steiner Forest 问题的对偶问题.

GMC 问题是经典的 Multicut 问题的自然推广. 若在 GMC 实例中每个终端集的大小限制为 2，则 GMC 问题恰好是 Multicut 问题. GMC 问题也推广了传统的 Multiway Cut 问题，因为若参数 $l$ 限制为 1，则 GMC 问题即归约到 Multiway Cut 问题. 另一方面，GMC 问题又是推广的 Steiner Forest 问题(Generalized Steiner forest，GSF)的对偶问题(目标相反). GSF 问题的实例与 GMC 问题相同，但其目标是使用最小费用的边集连通 $X$ 中的每一个终端集. 显然，GSF 问题的最优解是一个森林，而 GMC 问题的最优解是边的集合，该边集破坏了连通终端集的任意子图.

1) 相关工作

对于树上的 Multicut 问题(MC(T))，Garg，Vazirani 和 Yannakakis[1] 给出了近似比为 2 的原始-对偶近似算法(在本文中称为 GVY 算法). 这也是 MC(T)问题当前已知最好的近似比. 最近，Levin 和 Segev[2] 考虑了 Prize-collecting 版本的树上的 Multicut 问题(pc-MC(T))和 $k$ 版本的树上的 Multicut 问题($k$-MC(T)). 他们将 pc-MC(T)问题归约到 MC(T)问题解决，证明了 GVY 算法对 pc-MC(T)问题具有拉格朗日乘数保持(Lagrangian multiplier preserving)性质[3]，从而通过使用拉格朗日松弛技术将 $k$-MC(T)归约到 pc-MC(T)的方法，对 $k$-MC(T)问题给出了近似比为 $8/3+\varepsilon$ 的近似算法，其中 $\varepsilon>0$ 是任意小的常数. 对 $k$-MC(T)同样的结果也由 Golovin，Nagarajan 和 Singh[4] 独立地得到. 事实上，由于 Jain 和 Vazirani[5] 对 Facility Location 问题和 $k$-Median 问题的开创性工作，在其中拉格朗日松弛技术首次被引入到近似算法的分析与设计中，关于许多优化问题的 Prize-collecting 版本和 $k$ 版本的研究已经引起了人们越来越广泛的关注(例如，参考文献[2,5-6]).

对于一般无向图上的 Multicut 问题(MC(G))，Garg，Vazirani 和 Yannakakis[7] 通过基于区域增长的线性规划取整方法给出了近似比为 $O(\log l)$ 的近似算法. 后来，Avidor 和 Langberg[8] 将区域增长技术推广到 GMC(G)问题，对 GMC(G)问题获得相同的近似比 $O(\log l)$.

若在 GSF 问题中每一个终端集大小限制为 2，则该问题即是经典的 Steiner Forest 问题. 根据 Agrawal，Klein 和 Ravi[9] 以及 Goemans 和 Williamson[10] 的结果，一般无向图上的 Steiner Forest 问题和 GSF 问题都可以近似到 2 以内. 最近，Hajiaghayi 和 Jain[11] 提出了一般无向图上 $k$ 版本的 Steiner Forest 问题($k$-SF(G))和树上 $k$ 版本的 Steiner Forest 问题($k$-SF(T)). 通过拉格朗日松弛技术，Segev 等人[12] 对 $k$-SF(G)问题给出了近似比为 $O(\min\{n^{2/3},\sqrt{l}\}\log l)$ 的近似算法. 后来这一近似比被 Zhang[13] 通过一个简单的贪心策略改进到 $O(\min\{n^{2/3},\sqrt{l}\}\log k)$. 另一方面，近似 $k$-SF(T)问题仍是未解决的.

对于最大割集问题，在近似算法方面一项著名的工作是 Goemans 和 Williamson[14] 采用半正定规划技术对 Max Cut 问题设计的近似比为 0.87856 的近似算法. Goemans 和 Williamson 的工作的重要意义在于这是首次将半正定规划应用于近似算法的分析与设计中，这一技术后来被人们迅速推广到与 Max Cut 问题相关的优化问题的近似算法设计中(参见文献[15-17]).

2) 我们的结果

我们考虑树上推广的 Multicut 问题和该问题的 Prize-collecting 版本(pc-GMC(T))以及 $k$ 版本($k$-GMC(T)). 注意到由 Garg，Vazirani 和 Yannakakis[1] 的结果，树上的 Multicut 问题可以近似到 2. Garg 等人[1] 还证明了即使在高度为 1 且边上均为单位费用的树上，MC(T)问题是 NP 难的和 MAX SNP 难的. 这意味着 GMC(T)问题具有同样的复杂度. GMC(T)可以容易地归约到 MC(T)问题：给定 GMC(T)问题的实例，对每个终端集 $S_i\in X$，选取所有 $\binom{|S_i|}{2}$ 个可能的源目标对加入到归约到的实例中. 显然这个归约是多项式时间可完成的.

于是，由文献[1]中的结果，GMC(T)问题具有近似比为 2 的近似算法.

**命题 1.** 树上推广的 Multicut 问题是 NP 难和 MAX SNP 难的.

**命题 2.** 树上推广的 Multicut 问题存在近似比为 2 的多项式时间近似算法.

在 pc-GMC(T)问题中，每一个终端集 $S_i$ 或者被所选择的边集断开，或者没有断开但不得不赔付一个惩罚 $\pi_i$. 注意到仅当终端集 $S_i$ 中的终端被相互断开时，我们说终端集 $S_i$ 被断开. pc-GMC(T)问题的目标是最小化所选取边的费用和所赔付的惩罚的总和. 本文扩展了 GVY 算法来处理 pc-GMC(T)问题，得到 pc-GMC(T)问题的一个原始-对偶近似算法. 对该算法的解证明了一个结构引理，该引理可看作是文献[1]中引理 5.1 的推广. 由结构引理和放松的互补松弛条件，最终证明算法的近似比为 3. 本文的算法与 GVY 算法的一个主要区别是本文的算法允许同时提升若干对偶变量，而 GVY 算法每次只提升一个对偶变量.

$k$-GMC(T)问题寻找最小费用的一个解，断开 $X$ 中至少 $k$ 个终端集. 基于拉格朗日松弛技术，Könemann，Parekh 和 Segev[18] 给出了一个统一的框架近似类 Set Cover 的 $k$ 版本优化问题. 尽管 $k$-MC(T)问题可以看作一个类 Set Cover 的问题(所有的终端对构成基本集 $U$，树上的每一条边对应集合族 $\mathcal{S}$ 中的一个集合，因为每一条边在树上可断开若干终端对)并因此可纳入文献[18]中的框架解决，我们注意到 $k$-GMC(T)问题并不能使用文献[18]中的框架解决，因为若把 $k$-GMC(T)转换成 Set Cover 问题，则所得到的集合族是指数规模的. 通过非一致的途径，本文对 $k$-GMC(T)问题给出了一个近似比为 $\min\{2(l-k+1),k\}$ 的多项式时间近似算法. 当 $k$ 接近于 1 或者 $l$ 时，近似比 $\min\{2(l-k+1),k\}$ 有良好的表现. 本文还找到了 $k$-GMC(T)问题和 $k$-稠密子图问题之间的一个有趣的联系，意味着对某个小的常数 $\varepsilon>0$，将 $k$-GMC(T)问题近似到 $O(n^{1/6-\varepsilon})$ 是困难的.

## 1　Prize-collecting 版本的树上推广的 Multicut 问题

在 pc-GMC(T)问题中，给定边上有费用 $c_e\in\mathbb{Q}^+$ 的树 $T=(V,E)$，终端集族 $X=\{S_1,S_2,\cdots,S_l\}$，以及对应于每个终端集 $S_i\in X$ 的惩罚 $\pi_i\in\mathbb{Q}^+$. 问题询问一组边，在树上删除这些边能够断开某些终端集，使得所选择边集的费用和没有断开的终端集的惩罚之和最小.

### 1.1　问题的线性规划表示

首先给出 GMC(T)问题的分数线性规划，如($LP_{\mathrm{f}}$)所示(下标 f 表示问题的完全版本，即断开所有的终端集).

$$(LP_{\mathrm{f}})\min\sum_{e\in E}c_e x_e,\tag{1}$$

subject to

$$\sum_{e\in p}x_e\geqslant 1,\ \forall i\in[l],\forall p\in P_i,\tag{2}$$
$$x_e\geqslant 0,\ \forall e\in E.$$

为了更好地理解线性规划($LP_{\mathrm{f}}$)，考虑它的整数版本. 对每一条边 $e\in E$ 定义了一个变量 $x_e$，用于指示边 $e$ 是否被选择. 为简明，使用符号$[l]$来表示集合$\{1,2,\cdots,l\}$. 因为 GMC(T)问题定义在树上，每一个终端对的两个终端之间有唯一一条路径. 使用符号$[t,t']$表示终端对$(t,t')$之间的唯一路径. 对每一个终端集 $S_i$，共有$\binom{|S_i|}{2}$个可能的终端对. 约束式(2)中的符号 $P_i$ 表示 $S_i$ 中所有终端对的唯一路径集合. 于是约束式(2)表明对每一个终端集 $S_i$，对 $P_i$ 中的每一条路径 $p$，$p$ 上至少要有一条边满足 $x_e=1$(因此 $S_i$ 被边集$\{e\in E:x_e=1\}$断开). 因为目标函数(式(1))是最小化所选取边的总费用，($LP_{\mathrm{f}}$)的一个整数最优解恰好对应于 GMC(T)问题的一个最优解，反之亦然.

pc-GMC(T)问题的分数线性规划可写为($LP_{\mathrm{p}}$)(下标 p 表示 GMC(T)问题的 Prize-collecting 版本).

$$(LP_{\mathrm{p}})\min\quad\sum_{e\in E}c_e x_e+\sum_{i\in[l]}\pi_i z_i,\tag{3}$$

subject to

$$\sum_{e\in p}x_e+z_i\geqslant 1,\ \forall i\in[l],\forall p\in P_i,\tag{4}$$
$$x_e\geqslant 0,\ \forall e\in E,$$
$$z_i\geqslant 0,\ \forall i\in[l].$$

在线性规划($LP_{\mathrm{p}}$)中，对每个终端集 $S_i$ 定义了一个变量 $z_i$，用于指示终端集 $S_i$ 是否被边集$\{e\in E:x_e=1\}$断开. 若终端集 $S_i$ 没有被断开，则 $z_i=1$，并因此 $S_i$ 不得不赔付惩罚 $\pi_i$. 约束式(4)表示对每个终端集 $S_i$，或者 $P_i$ 中的每一条路径上至少有一条边 $e$ 满足 $x_e=1$，或者 $z_i=1$. ($LP_{\mathrm{p}}$)的目标函数式(3)表示最小化所选取的边的费用和所赔付的惩罚之和.

$(LP_{\mathrm{p}})$的对偶规划如$(DP_{\mathrm{p}})$所示：

$$(DP_{\mathrm{p}})\max \quad \sum_{i\in[l]}\sum_{p\in P_i} f_{\mathrm{p}}$$

subject to

$$\sum_{i\in[l]}\sum_{\substack{p:e\in p\\ p\in P_i}} f_{\mathrm{p}} \leqslant c_e,\ \forall e\in E, \tag{5}$$

$$\sum_{p\in P_i} f_{\mathrm{p}} \leqslant \pi_i,\ \forall i\in[l], \tag{6}$$

$$f_{\mathrm{p}}\geqslant 0,\ \forall i\in[l], \forall p\in P_i.$$

对偶规划$(DP_{\mathrm{p}})$为每一个终端集的每一条路径定义了变量$f_{\mathrm{p}}$. $f_{\mathrm{p}}$可解释为路径$p$上的流值. 事实上对偶规划$(DP_{\mathrm{p}})$描述了与pc-GMC(T)问题对偶的流问题. 注意到可能存在这样的终端对，比如$(t,t')$，出现在多于一个的终端集中. 因此，在对偶规划$(DP_{\mathrm{p}})$中路径$[t,t']$会有多个代理路径，每一条代理路径$p$对应于$[t,t']$在一个终端集中的一次出现. 变量$f_{\mathrm{p}}$实际上是定义在每一条代理路径上.

### 1.2 原始-对偶近似算法

给定pc-GMC(T)问题的原始线性规划和对偶线性规划，基于原始-对偶方案，设计了pc-GMC(T)问题的近似算法，如算法1所示. 算法$\mathcal{A}$是GVY算法的扩展，其每一次迭代中新加入了对惩罚的处理. 算法$\mathcal{A}$与GVY算法的一个显著的区别是算法$\mathcal{A}$允许同时提升多个对偶变量$f_{\mathrm{p}}$，相比之下在GVY算法中每个对偶变量是单独提升的.

**算法$\mathcal{A}$.**

输入：树$T$，终端集族$\{S_i\}$及其惩罚$\{\pi_i\}$.

输出：边集$D$和指标集$N$，在树上删除$D$中的边断开$\{S_i: i\notin N\}$中的所有终端集.

1) let $D\leftarrow\varnothing, f_{\mathrm{p}}\leftarrow 0, N\leftarrow\varnothing$.
2) 固定任一顶点为树的根.
3) while 树上存在未处理的顶点时，对最深的未处理的顶点$v$ do
4) for each $S_i$中的终端对$(t,t')$，满足$i\notin N$且$lca(t,t')=v$ do
5) 假设包含终端对$(t,t')$并且没有被完全赔付的终端集为$S_{i_1}, S_{i_2},\cdots,S_{i_r}$（即，对每一个$j\in[r]$，$\{t,t'\}\subseteq S_{i_j}$且$i_j\notin N$）. 对每一个$j\in[r]$，以$p_j$表示路径$[t,t']$在$P_{i_j}$中的代理.
6) 对所有的$1\leqslant j\leqslant r$，同时提升$f_{p_j}$；并且在此过程中：
7) if 对某个$j\in[r]$，惩罚$\pi_{i_j}$被完全赔付 then 停止提升$f_{p_j}$，let $N\leftarrow N\cup\{i_j\}$.
8) if 某条边$e\in[t,t']$饱和 then 停止提升所有$f_{p_j}$；let $D\leftarrow D\cup\{e\}$.
9) end
10) end
11) 设$D$中的边按加入的先后次序为$e_1,e_2,\cdots,e_t$，设$A$为$X$中被$D$断开的终端集下标的集合.
12) for $j=t$ downto 1 do
13) if $D-\{e_j\}$仍能够断开$A$中的所有终端集 then 从$D$中移去边$e_j$.
14) end
15) output $D$和$N$.

算法$\mathcal{A}$中，$D$表示算法选取的边，$N$表示算法已完全赔付其惩罚的终端集的下标的集合. 即，边$e\in D$当且仅当$x_e=1$，下标$i\in N$当且仅当$z_i=1$. 算法$\mathcal{A}$第4)步中的$lca(t,t')$表示终端$t$和$t'$在树上的最小公共祖先. 算法从原始不可行解($\boldsymbol{x}=\boldsymbol{0},\boldsymbol{z}=\boldsymbol{0}$)和一个平凡的对偶可行解$\boldsymbol{f}=\boldsymbol{0}$开始. 在每次迭代中，算法$\mathcal{A}$不断改进原始解$(\boldsymbol{x},\boldsymbol{z})$的可行性和对偶解$f$的最优性. 最终，算法$\mathcal{A}$找到$(LP_{\mathrm{p}})$的一个整数可行解，和$(DP_{\mathrm{p}})$的一个尽可能最优的解，同时两个解之间满足放松的互余松弛条件.

在步骤3)到10)中，算法$\mathcal{A}$以自底向上的方式处理树上的每一个顶点. 对要处理的顶点$v$，以及对每一个终端对$(t,t')$，满足其最小公共祖先是$v$以及存在一个终端集$S_i$，$S_i$包含$\{t,t'\}$且$i\notin N$，算法贪心地在路径$[t,t']$上路由流$f_{p_1},f_{p_2},\cdots,f_{p_r}$，在此$p_j$是路径$[t,t']$在$P_{i_j}$中的代理，$i_j$是包含$\{t,t'\}$且不在$N$中的终端集$S_{i_j}$的下标. 在这个过程中，其惩罚被完全赔付的终端集的下标（即对此终端集约束式(6)取等式）被加入$N$，路径$[t,t']$上饱和的边（即对此边约束式(5)取等式）被加入$D$. 注意到路径$[t,t']$的代理$p_j$仅对惩罚$\pi_{i_j}$的赔付作出贡献，而路径$[t,t']$的所有代理均对$[t,t']$上边的饱和作出贡献. 算法在步骤12)到14)执行一个反向删除过程来删除$D$中冗余的边. 反向删除过程对以下引理2的成立起到关键作用.

首先证明算法$\mathcal{A}$的输出是pc-GMC(T)问题的一个可行解.

**引理1.** 算法$\mathcal{A}$的输出$(D,N)$是pc-GMC(T)问题的一个可行解.

证明. 仅需要证明在反向删除过程之前$D$是$A$中所有终端集的一个Multicut，因为反向删除过程仅去掉了冗余的边. 固定任一个下标$i\in A$. 反证. 假设$S_i$中有一个终端对$(t,t')$没有被$D$断开. 以$p$表

示路径$[t,t']$在 $P_i$ 中的代理. 由反证假设，路径$[t,t']$上既没有饱和的边，惩罚 $\pi_i$ 也没有被完全赔付. 因此至少可以在路径$[t,t']$上增加流 $f_p$，直到或者$[t,t']$上的某条边饱和，或者惩罚 $\pi_i$ 被完全赔付. 但这与算法的步骤 6)～8)矛盾. 证毕.

Garg 等人在文献[1]中证明了在 MC(T)问题中 GVY 算法输出的解的一个结构特征. 对每一个断开的终端对$(s_i,t_i)$，若在路径$[s_i,t_i]$上流值非零，则在每条路径$[s_i,v]$和$[v,t_i]$上最多只有一条边被 GVY 算法选取，在此 $v$ 是 $s_i$ 和 $t_i$ 的最小公共祖先. 对算法 $\mathcal{A}$ 尽管可能存在许多路径其流值非零，而算法并没有选取这些路径上的边(这样的路径没有被断开，其上的流值赔付了没有断开的终端集的惩罚)，我们证明对每一个断开的终端对结构特征仍然是保持的，从而把结构特征推广到 pc-GMC(T)问题中.

**引理 2**[1]. 对每一个断开的终端对$(t,t')$，若路径$[t,t']$上的流值非零，则在每条路径$[t,v]$和$[v,t']$上最多只有一条边被算法 $\mathcal{A}$ 选取，其中 $v=lca(t,t')$是 $t$ 和 $t'$的最小公共祖先.

证明. 反证. 假设经过反向删除后路径$[t,v]$上仍有两条边 $e$ 和 $e'$留在 $D$ 中. 不失一般性，设在树上 $e$ 比 $e'$低. 因为在步骤 13)中边 $e$ 被测试时 $e$ 没有被删除，必有一个终端对$(s,s')$，$(s,s')$在终端集 $S_j$ 中且 $j\notin N$，使得 $e$ 是分开 $s$ 和 $s'$的唯一的边.

因为 $e$ 比 $e'$低，$u=lca(s,s')$比 $v$ 低. 因此，在算法 $\mathcal{A}$ 中 $u$ 在 $v$ 之前被处理. 当 $u$ 被处理时，$e$ 必定不在 $D$ 中. 否则，路径$[t,t']$上就不会有非零的流. 因此，当 $u$ 被处理时，$(s,s')$是被另一条边 $\tilde{e}$ 断开的，这意味着在 $D$ 中 $\tilde{e}$ 的次序领先于 $e$. 但这与当 $e$ 被测试时 $e$ 是 $D$ 中分开 $s$ 和 $s'$的唯一的边相矛盾. 证毕.

由引理 1 和引理 2，可以证明本节的主要定理，即定理 1.

**定理 1.** 算法 $\mathcal{A}$ 是 pc-GMC(T)问题的多项式时间 3 倍近似算法.

证明. 首先，由引理 1，算法 $\mathcal{A}$ 的输出是 pc-GMC(T)问题的可行解. 由算法 $\mathcal{A}$ 的原始-对偶方案可知，算法找到的原始解$(\boldsymbol{x},\boldsymbol{z})$和对偶解 $\boldsymbol{f}$ 满足：

$$x_e=1\Leftrightarrow\sum_{i\in[l]}\sum_{\substack{p:e\in p\\ p\in P_i}}f_p=c_e,$$

$$z_i=1\Leftrightarrow\sum_{p:p\in P_i}f_p=\pi_i.$$

因此，对边集 $D$ 有

$$\sum_{e\in D}c_e=\sum_{e\in D}\sum_{i\in[l]}\sum_{\substack{p:e\in p\\ p\in P_i}}f_p=$$

$$\sum_{i\in[l]}\sum_{p\in P_i}f_p\,|\,D\cap p\,|\leqslant 2\sum_{i\in[l]}\sum_{p\in P_i}f_p. \tag{7}$$

在不等式(7)中，第 2 个等号成立是因为交换了求和的次序. 最后一个不等号从如下两个方面进行了放松：首先，由引理 2 对每一个切断的路径 $p$，$f_p>0$ 意味着$|D\cap p|\leqslant 2$. 其次，仍然存在某些路径 $p$ 满足 $f_p>0$ 且$|D\cap p|=0$. 由算法可知，这些路径上的流赔付了未断开的终端集的惩罚.

对于未断开的终端集的下标集合 $N$，有

$$\sum_{i\in N}\pi_i=\sum_{i\in N}\sum_{p\in P_i}f_p\leqslant\sum_{i\in[l]}\sum_{p\in P_i}f_p. \tag{8}$$

由不等式(7)(8)，可知算法的解值 $A(I)$，满足

$$A(I)=\sum_{e\in D}c_e+\sum_{i\in N}\pi_i\leqslant 3\sum_{i\in[l]}\sum_{p\in P_i}f_p\leqslant$$

$$3\,OPT_f(LP_p)\leqslant 3OPT, \tag{9}$$

在此$OPT_f(LP_p)$表示线性规划$(LP_p)$的分数最优解值，$OPT$ 表示 pc-GMC(T)问题实例的最优解值. 由线性规划理论的原始-对偶定理，不等式(9)中的第 2 个不等号成立.

因为算法 $\mathcal{A}$ 不需要解线性规划，算法 $\mathcal{A}$ 是一个纯组合算法. 并且，通过确定步骤 6)～8)中实际上是离散的各个事件的发生时刻，算法 $\mathcal{A}$ 可以在多项式时间内实现. 证毕.

## 2　*k* 版本的树上推广的 Multicut 问题

给定边上具有费用 $c_e\in\mathbb{Q}^+$的树 $T=(V,E)$，终端集族 $X=\{S_1,S_2,\cdots,S_l\}$以及整数 $k\in\mathbb{Z}^+$，$k$-GMC(T)问题询问边集 $E$ 的费用最小的一个子集，使得从树上删除该子集中的边断开 $X$ 中至少 $k$ 个终端集.

$k$-GMC(T)问题是 GMC(T)问题的自然的 $k$ 版本扩展. 本节给出了 $k$-GMC(T)问题和 $k$-稠密子图问题之间的一个有趣的联系，这意味着简单地增加一个参数 $k$ 极大地增加了树上推广的 Multicut 问题的难度. 然后通过非一致的方法给出了 $k$-GMC(T)问题的一个多项式时间近似算法，近似比为 $\min\{2(l-k+1),k\}$.

### 2.1　*k*-GMC(T)问题的近似难度

本节证明 $k$-GMC(T)问题的近似难度可归约到 $k$-稠密子图问题的近似难度. 首先证明 $k$-严格顶点覆盖问题可保持近似比地归约到 $k$-GMC(T)问题. $k$-严格顶点覆盖问题($k$-strict vertex cover，$k$-SVC)的实例包括无向图 $G$ 和正整数 $k$，问题的目标是寻找图 $G$ 上最小数目的顶点，使得其导出子图上

包含至少 $k$ 条边.

**引理 3.** $k$-SVC 问题可保持近似比地归约到 $k$-GMC(T)问题.

证明. 归约到 $k$-GMC(T)问题的实例可按如下方式构造. 实例中树的根是 $r$. 对 $k$-SVC 实例的每一个顶点 $v_i$，在归约到的实例上有一个顶点 $v_i$，其双亲是 $r$. 对 $k$-SVC 实例的每一条边 $e=(v_i, v_j)$，在归约到的实例上有一个终端集 $S=\{r, v_i, v_j\}$. 于是，若在 $k$-SVC 实例上有一个顶点集 $V'$ 严格覆盖至少 $k$ 条边，则在 $k$-GMC(T)实例上有一个边集(每一条边对应 $V'$ 中的一个顶点)，移去该边集中的边断开至少 $k$ 个终端集(每一个被断开的终端集对应 $k$-SVC 中被严格覆盖的一条边)，反之亦然. 证毕.

给定无向图 $G$ 和正整数 $k$，$k$-稠密子图问题询问大小为 $k$ 的顶点集，使得其导出子图上包含的边的数目最多. Hajiaghayi 和 Jain[11] 证明了若 $k$-SVC 问题可被近似到 $\alpha$，则 $k$-稠密子图问题可被近似到 $2\alpha^2$. 对 $k$-稠密子图问题当前已知最好的近似比是 $O(n^{1/3-\varepsilon})$(对某个小的常数 $\varepsilon>0$)，并且对该近似比的改进被认为是困难的[19-20]. 由上述事实以及引理 3，可得定理 2.

**定理 2.** 若 $k$-GMC(T)问题能够在多项式时间内近似到 $O(n^{1/6-\varepsilon})$(对某个小的常数 $\varepsilon>0$)，则 $k$-稠密子图问题存在近似比为 $O(n^{1/3-2\varepsilon})$ 的多项式时间近似算法.

### 2.2 $k$-GMC(T)问题的近似算法

本节通过非一致的方法给出 $k$-GMC(T)问题近似比为 $\min\{2(l-k+1), k\}$ 的多项式时间近似算法. 事实上对 $k$-GMC(T)问题设计了两个近似算法，其中一个算法基于贪心方法，用于处理 $k$ 较小的情形，另一个算法基于 LP-rounding 技术，用于处理 $l-k$ 较小的情形.

若在 $k$-GMC(T)问题中参数 $l$ 被限制为 1，则该问题归约到树上的 Multiway Cut 问题. 我们证明树上的 Multiway Cut 问题是多项式时间可解的.

**引理 4.** 树上的 Multiway Cut 问题可在多项式时间内最优求解.

证明. 给出构造性的证明. 固定任一个顶点为树的根，初始化边集 $F$ 为空集. 首先在树上执行一个剪枝过程. 若树上存在不是终端的叶子，则移去该叶子及其邻接的边. 重复这个过程直到每一个叶子均是终端.

然后寻找树上最深的非叶子顶点 $u$. 若 $u$ 是终端，则在 $u$ 的每一个分支上选取一条费用最小的边加入 $F$，然后在树上删除所有这些分支. 否则 $u$ 不是一个终端. 假设 $u$ 的分支数为 $f$. 在 $u$ 的每条分支的费用最小的边中选取 $f-1$ 条最轻的边加入 $F$，然后在树上删除这 $f-1$ 条断开的分支. 重复这个过程直到树上仅剩余一个终端.

上述过程将原问题在多项式时间内最优地分解到了若干子问题，每个子问题又是多项式时间可最优求解的. 证毕.

树上 Mutliway Cut 问题的多项式时间可解性自然地给出了 $k$-GMC(T)问题的一个贪心近似算法.

**引理 5.** $k$-GMC(T)问题可在多项式时间内被近似到 $k$ 以内.

证明. 对 $X$ 中每一个终端集 $S_i$ 求解 Multiway Cut 实例. 假设对应于 $S_i$ 的实例的最优解为 $F_i$. 则最轻的 $k$ 个子集 $F_i$ 的并是 $k$-GMC(T)的 $k$ 倍近似解，因为 $k$-GMC(T)的最优解必断开了至少 $k$ 个终端集，因此所选取的 $k$ 个子集中的每个子集的费用都不超过 $k$-GMC(T)的最优解值. 证毕.

然后给出 $k$-GMC(T)问题的另一个以参数 $l$ 和 $k$ 表示的近似比. 所采用的技术是 LP-rounding 技术和放缩(scaling)技术. 首先给出 $k$-GMC(T)问题的分数线性规划，如($LP_k$)所示(在此下标 $k$ 表示 GMC(T)问题的 $k$ 版本)：

$$(LP_k)\min \quad \sum_{e\in E} c_e x_e,$$

subject to

$$\sum_{e\in p} x_e + z_i \geqslant 1,\ \forall i\in[l], \forall p\in P_i, \quad (10)$$

$$\sum_{i\in[l]} z_i \leqslant l-k, \quad (11)$$

$$x_e \geqslant 0,\ \forall e\in E,\ z_i\geqslant 0,\ \forall i\in[l].$$

考虑线性规划($LP_k$)的整数版本. 变量 $z_i$ 指示终端集 $S_i$ 是否被边集 $\{e\in E: x_e=1\}$ 切断. 若 $S_i$ 没有被断开，则 $z_i=1$. 约束(式(11))表示被断开的终端集的数目必须至少为 $k$ 个. ($LP_k$)的目标函数是最小化断开至少 $k$ 个终端集的边的总费用.

基于 LP-rounding 和放缩技术的算法简述如下. 求解($LP_k$)得分数最优解($\boldsymbol{x}^*$，$\boldsymbol{z}^*$)，算法将前 $l-k$ 个最大的 $z_i^*$ 取整到 1，余下的 $z_i^*$ 取整到 0. 于是定义在取整到 0 的终端集上的实例归约到 GMC(T)问题. 算法求解归约到实例，并将其解作为原问题 $k$-GMC(T)的解. 注意到经过适当放缩的 $\boldsymbol{x}^*$ 是归约到实例的可行解. 这个事实给出了算法的近似比.

**引理 6.** $k$-GMC(T)问题可在多项式时间内近

似到 $2(l-k+1)$以内.

证明. 不失一般性，通过重排下标的次序，假设最优解$(\boldsymbol{x}^*,\boldsymbol{z}^*)$中变量 $z_i^*$ 满足 $z_1^*\geqslant z_2^*\geqslant\cdots\geqslant z_l^*$. 将变量 $z_1^*,\cdots,z_{l-k}^*$取整到 1，余下的变量 $z_{l-k+1}^*,\cdots,z_l^*$ 取整到 0. 即，算法将断开终端集 $S_{l-k+1},\cdots,S_l$，而不理会 $S_1,\cdots,S_{l-k}$. 记取整后的 $\boldsymbol{z}^*$ 为 $\hat{\boldsymbol{z}}$，记下标集$\{i:\hat{z}_i=0\}$为 $Q$. 对每一个 $i\in Q$ 必有 $z_i^*\leqslant(l-k)/(l-k+1)$，否则 $z_1^*+z_{l-k+1}^*>l-k$，与约束(式(11))矛盾. 因此，由约束(式(10))，可知 $\forall i\in Q,\forall p\in P_i$，

$$\sum_{e\in p}x_e\geqslant 1-z_i^*\geqslant 1-\frac{l-k}{l-k+1}.$$

记 $a=(l-k)/(l-k+1)$. 对每条边 $e$，定义

$$x'_e=\begin{cases}\dfrac{1}{1-a}x_e^*, & x_e^*\leqslant 1-a,\\ 1, & \text{otherwise.}\end{cases}$$

于是，$x'$是 $Q$ 上线性规划$(LP_{\mathrm{f}})$的一个可行解.

$Q$ 上$(LP_{\mathrm{f}})$恰好是终端集族为 $X=\{S_i:i\in Q\}$的 GMC(T)问题的线性规划. 由 GVY 算法，可在多项式时间内求得 $Q$ 上$(LP_{\mathrm{f}})$的 2 倍整数近似解 $\hat{\boldsymbol{x}}$. 现在，$(\hat{\boldsymbol{x}},\hat{\boldsymbol{z}})$是$(LP_k)$的一个整数可行解，其解值满足

$$\sum_{e\in E}c_e\hat{x}_e\leqslant 2\,OPT_{\mathrm{f}}(LP_{\mathrm{f}})\leqslant 2\sum_{e\in E}x'_e\leqslant \frac{2}{1-a}\sum_{e\in E}x_e^*\leqslant\frac{2}{1-a}OPT, \tag{12}$$

在此$OPT_{\mathrm{f}}(LP_{\mathrm{f}})$表示$(LP_{\mathrm{f}})$的分数最优解值，$OPT$ 表示 $k$-GMC(T)的最优解值. 在不等式(12)中，第 2 个不等号成立是因为 $x'$是 $Q$ 上$(LP_{\mathrm{f}})$的可行解，第 3 个不等号成立是因为由定义对每条边 $e$ 有 $x'_e\leqslant\dfrac{1}{1-a}x_e^*$. 由于 $2/(1-a)=2(l-k+1)$，引理得证.

证毕.

引理 5 和引理 6 给出了本节的主要定理.

**定理 3.** $k$-GMC(T)问题存在近似比为 $\min\{2(l-k+1),k\}$的多项式时间近似算法.

注意到定理 3 中的近似比 $\min\{2(l-k+1),k\}$在 $k$ 接近于 1 或 $l$ 时都有良好的表现. 并且，因为 $k$ 在$\{1,\cdots,l\}$上取值，总是有 $\min\{2(l-k+1),k\}\leqslant(2/3)l+2/3$.

## 3　结束语

本文提出并研究树上推广的 Multicut 问题. 通过将 GMC(T)问题归约到树上的 Multicut 问题，证明了(完全版本的)GMC(T)问题有 2 倍近似算法. 对 Prize-collecting 版本的 GMC(T)问题，给出了近似比为 3 的原始-对偶近似算法. 对 $k$ 版本的 GMC(T)问题，给出了近似比为 $\min\{2(l-k+1),k\}$的近似算法.

本文还证明了将 $k$-GMC(T)问题近似到 $O(n^{1/6-\varepsilon})$以内是困难的. 对这些近似结果和近似难度结果的改进自然成为有趣的研究问题. 尤其是，尽管本文对 $k$-GMC(T)问题给出了非平凡的近似比 $\min\{2(l-k+1),k\}$，该近似比看起来仍有较大的改进空间.

**致谢**　作者感谢李昂生研究员对本研究给予的指导和热情鼓励！

## 参　考　文　献


[1] Garg N, Vazirani V, Yannakakis M. Primal-dual approximation algorithm for integral flow and multicut in trees[J]. Algorithmica, 1997, 18(1): 3-20

[2] Levin A, Segev D. Partial multicuts in trees[C] //Proc of the 3rd Workshop on Approximation and Online Algorithms (WAOA). Berlin: Springer, 2005: 320-333

[3] Jain K, Mahdian M, Markaksi E, *et al*. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP[J]. Journal of the ACM, 2003, 50(6): 795-824

[4] Golovin D, Nagarajan V, Singh M. Approximating the $k$-multicut problem [C] //Proc of the 17th ACM-SIAM Symposium on Discrete Algorithms (SODA). New York: ACM Press, 2006: 621-630

[5] Jain K, Vazirani V. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and Lagrangian relaxation[J]. Journal of the ACM, 2001, 48(2): 274-296

[6] Chudak F, Roughgarden T, Williamson D. Approximate $k$-MSTs and $k$-Steiner trees via the primal-dual method and Lagrangean relaxation [J]. Mathematical Programming: Series A, 2004, 100(2): 411-421

[7] Garg N, Vazirani V, Yannakakis M. Approximate max-flow min-(multi)cut theorems and their applications [J]. SIAM Journal on Computing, 1996, 25(2): 235-251

[8] Avidor A, Langberg M. The multi-multiway cut problem [C] //Proc of the 9th Scandinavian Workshop on Algorithm Theory (SWAT). Berlin: Springer, 2004: 273-284

[9] Agrawal A, Klein P, Ravi R. When trees collide: An approximation algorithm for the generalized Steiner problem on networks [J]. SIAM Journal of Computing, 1995, 24(3): 440-456

[10] Geomans M, Williamson D. A general approximation technique for constrained forest problems[J]. SIAM Journal on Computing, 1995, 24(2): 296-317

[11] Hajiaghayi M, Jain K. The prize-collecting generalized Steiner tree problem via a new approach of primal-dual schema [C] //Proc of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). New York: ACM, 2006: 631-640
[12] Segev D, Segev G. Approximate *k*-Steiner forests via the Lagrangian relaxation technique with internal preprocessing [C] //Proc of the 14th Annual European Symp on Algorithms (ESA). Berlin: Springer, 2006: 600-611
[13] Zhang P. An approximation algorithm to the *k*-Steiner forest problem [J]. Theoretical Computer Science (Accepted).
[14] Goemans M, Williamson D. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming [J]. Journal of the ACM, 1995, 42(6): 1115-1145
[15] Frieze A, Jerrum M. Improved approximation algorithms for max *k*-cut and max bisection [J]. Algorithmica, 1997, 18 (1): 67-81
[16] Ye Y. A.699-approximation algorithm for max-bisection [J]. Mathematical Programming: Series A, 2001, 90(1): 101-111
[17] Xu Dachuan, Han Jiye, Du Donglei. Improved approximation algorithm for graph partition problem [J]. Acta Mathematicae Applicatae Sinica, 2005, 28(4): 587-597 (in Chinese)
(徐大川，韩继业，杜东雷. 关于图划分问题的改进的近似算法[J]. 应用数学学报，2005，28(4)：587-597)
[18] Könemann J, Parekh O, Segev D. A unified approach to approximating partial covering problems [C] //Proc of the 14th Annual European Symp on Algorithms (ESA6). Berlin: Springer, 2006: 468-479
[19] Feige U. Relations between average case complexity and approximation complexity [C] //Proc of the 34th Annual ACM Symp on Theory of Computing (STOC). New York: ACM, 2002: 534-543
[20] Khot S. Ruling out PTAS for graph min-bisection, denset subgraph and bipartite clique [C] //Proc of the 44th Annual IEEE Symposium on the Foundations of Computer Science (FOCS). Los Alamitos: IEEE Computer Society, 2004: 136-145

**Zhang Peng**, born in 1973. Received his bachelor degree and master degree from the School of Computer Science and Technology, Shandong University, China, in 1999 and 2004, respectively. Received his Ph. D. degree in computer science from the Institute of Software, the Chinese Academy of Sciences in 2007. His main research interests include approximation algorithms, combinatorial optimization and computational complexity.

**张　鹏**，1973年生，博士，主要研究方向为近似算法、组合优化和计算复杂性.

**Research Background**

Cut problem is a classical and active topic in combinatorial optimization and approximation algorithms for many years. The classical multicut problem has been studied by Garg, Vazirani and Yannakakis *et al*. Also, since the seminal work of Jain and Vazirani for the facility location and the *k*-median problems, the study of approximating the prize-collecting version and *k*-version of many optimization problems has attracted more attention. Very recently (in ESA'06), Segev *et al* considered the problem *k*-multicut in trees (MC(T)) and gave a $8/3+\varepsilon$-approximation for arbitrarily small $\varepsilon>0$. We propose and study the generalized multicut in trees (GMC(T)) problem, in which each terminal pair is generalized to a terminal set, and its prize-collecting version and *k*-version. The GMC(T) problem has its own interest since not only it generalizes both the MC(T) problem and the multiway cut problem, but also it is the dual of the generalized Steiner forest problem, which is studied by Agrawa, Klein and Ravi, Goemans and Williamson, and Hajiaghayi and Jain *et al*. This work is supported by the National Natural Science Foundation of China under grant No. 60325206 and No. 60310213.