高技术通讯 2010年 第20卷 第3期:264~269

doi:10.3772/j.issn.1002-0470.2010.03.008

# 带权最大割问题的一种基于划分技术的固定参数可解算法①

刘运龙②* **　王建新*

(*中南大学信息科学与工程学院　长沙 410083)

(**湖南师范大学继续教育学院　长沙 410013)

**摘　要**　运用参数计算复杂性理论和技术对带权最大割问题进行了研究。首先对该问题及其相关概念进行了参数化定义,然后对参数化带权最大割问题提出了一种基于随机划分技术的随机算法。该随机算法依次将实例图的顶点进行$\lceil \ln(1/\varepsilon) \rceil \times 2^k (0 < \varepsilon < 1)$次随机划分,并选择其中权值最大的 $k$-划分作为输出解,因而能在时间 $O^*(\ln(1/\varepsilon)2^k)$ 内以至少 $1-\varepsilon$ 的概率找到目标解。接着在此基础上着重运用最新改进的 $(n,k)$-全集划分技术对参数化带权最大割问题提出了一个时间复杂度为 $O^*(2^{2k+12\log^2(2k)})$ 的确定性算法,表明了带权最大割问题是固定参数可解的。

**关键词**　带权最大割问题,固定参数可解,随机划分,$(n,k)$-全集

## 0　引　言

带权的最大割问题是计算复杂性理论中一个经典的NP难解问题,在电路设计和统计物理学中有着广泛的实际应用[1]。自从1972年被证明是NP难的以来,该问题一直是近似计算领域中的一个主要研究对象[2-4]。在参数计算复杂性领域近年来人们也开始关注了该问题,但当前的研究主要集中于该问题的特殊情况,即研究参数化的最大割问题[5-7]。本文的主要工作是对参数化带权的最大割问题进行研究,并对该问题设计有效的固定参数可解算法,即对参数化带权的最大割问题给出一个基于随机划分技术的随机算法,对参数化带权的最大割问题提出一个基于$(n,k)$-全集划分技术的确定性算法。

## 1　基本概念

**定义1** (最大割问题[8])　对于一个无向图 $G=(V,E)$,如果 $V$ 的两个非空子集 $S$ 和 $T$ 满足 $S \cup T = V$ 和 $S \cap T = \varnothing$,则称 $S$ 和 $T$ 构成图 $G$ 的一个割。割的规模是一端在 $S$ 中,另一端在 $T$ 中的边的数目。最大割问题就是对于给定的一个无向图 $G$,求图 $G$ 的一个规模最大的割。

如果 $S$ 和 $T$ 构成图 $G$ 的一个割,本文用$(S,T)$表示该个割。对于图 $G$ 的任意一个割$(S,T)$和任意一条边 $d$,如果边 $d$ 的一个端点属于 $S$,另一个端点属于 $T$,本文简称割$(S,T)$含有边 $d$ 或者边 $d$ 属于割$(S,T)$。

**定义2** (参数化最大割问题[5])　给定一个无向图 $G=(V,E)$ 和一个正整数 $k$,目标是判定图 $G$ 是否存在一个规模至少为 $k$ 的割。

同样,本文称图 $G$ 的一个规模至少为 $k$ 的割为图 $G$ 的一个 $k$-割(或 $k$-划分),并用符号 $k$-Cut 表示。

**定义3** (带权的最大割问题[1])　对于一个无向图 $G=(V,E)$,其中每一条边$(i,j) \in E$ 都带有一个非负权值 $w_{ij}$,图 $G$ 上一个割的权值等于该个割中所含有边的权值之和。带权的最大割问题就是对于给定的一个无向带权图 $G$,求图 $G$ 的一个权值最大的割。

根据参数计算复杂性理论,本文对带权的最大割问题给出如下的参数化定义。

**定义4** (参数化带权的最大割问题)　对于一个边带权的无向图 $G=(V,E)$ 和一个正整数 $k$,图 $G$ 上一个 $k$-Cut 的权等于该个 $k$-Cut 中权值最大的 $k$ 条边的权值之和。参数化带权的最大割问题就是在图 $G$ 中找出一个权值最大的 $k$-Cut。如果图 $G$ 中不存在一个 $k$-Cut,则报告不存在。

① 973计划(2008CB317107),国家自然科学基金(60433020,60773111),新世纪优秀人才计划(NCET-05-0683),教育部创新团队计划(IRT0661),湖南省杰出青年基金(06JJ10009)和湖南省自然科学基金(09JJ3116)资助项目。

② 男,1971年生,博士;研究方向:参数计算、计算机算法;联系人,E-mail: hnsdlyl@163.com

(收稿日期:2009-03-02)

对于参数化带权最大割问题的一个实例($G$, $k$),如果存在一个参数算法,当参数 $k$ 为 1,2,3,…,$k_0$ 时都能正确地返回一个权值最大的 $k$-Cut,而当 $k = k_0 + 1$ 时,返回不存在 $k$-Cut,则图 $G$ 上某一个权值最大的 $k$-Cut ($1 \leqslant k \leqslant k_0$)就一定正好是图 $G$ 上一个权值最大的割,这样就通过参数算法有效地求解了带权的最大割问题。参数算法在什么情况下是有效的呢? 为此引述固定参数可解的定义。

**定义 5**(固定参数可解[9]) 对于一个参数化带权的最大值问题 **P**,如果存在一个算法 **A**,对给定的一个实例 $x$,当问题的可行解集 $S_P(x) \neq \varnothing$ 时,算法 **A** 能在时间 $O(f(k)n^{O(1)})$内返回 $m^*(x) = \max\{m(x,y) \mid y \in S_P(x)\}$,其中 $n$ 为实例 $x$ 的大小,$f$ 是独立于 $n$ 的一个函数,当问题的可行解集 $S_P(x) = \varnothing$ 时算法 **A** 给出适当提示,就说问题 **P** 是固定参数可解的(fixed parameter tractable)。相应地,算法 **A** 就是问题 **P** 的一个固定参数可解算法。

从定义 5 可以看出,固定参数可解算法时间复杂度的形式为 $O(f(k)n^{O(1)})$,通常简写为 $O^*(f(k))$。显然,当 $k$ 比较小时,固定参数可解算法是高效的。因此对于大量的 NP 难解问题,人们一直在寻求其固定参数可解算法。

## 2 一种基于随机划分技术的随机算法

随机划分技术是设计参数算法的一种新技术,其基本原理是:对于一个已知集合 $U$ 和其中的一个未知子集 $H$,如果将集合 $U$ 随机地进行一次划分,则子集 $H$ 的任何一种预先给定的划分都必定以一定的概率出现。根据这一原理,要使子集 $H$ 的某一种预定的划分以至少某个概率出现,只要对集合 $U$ 进行足够多次重复随机划分。随机划分技术应用到参数化问题最初是由 Cai 等人在文献[10]中提出来的,最近 Chen 等人在文献[11-13]中做了进一步的研究,并被应用到 Path, Packing 和 Matching 等问题的参数算法研究中。

对于参数化带权最大割问题的一个给定的实例($G = (V,E)$, $k$),假设 $G$ 的一个权值最大的 $k$-Cut 为 $(S,T)$。如果将顶点集 $V$ 随机地划分为两个子集 $V_L$ 和 $V_R$,则$(S,T)$中权值最大的 $k$ 条边必定以一定的概率属于割 $(V_L,V_R)$。因此,如果随机试探划分足够多次,便可以一个很大的概率找到 $G$ 的一个权值最大的 $k$-Cut。

基于上述思想,可以对参数化带权的最大割问题提出一个随机算法。图 1 详细描述了该算法的具体步骤。其中 $Q_0$ 用来存放 $k$-Cut 中权值最大的 $k$ 条边,其初始值为空集。$S_0$ 用来存放 $k$-Cut ($V_L$, $V_R$)中的点集 $V_L$,其初始值也为空集。算法的第 2 步是对顶点集 $V$ 进行$\lceil \ln(1/\varepsilon) \rceil \times 2^k$ 次随机划分,第 2.2 步至 2.4 步是判定每一次划分是否构成图 $G$ 的一个权值最大的 $k$-Cut。

算法 **MaxCut1**($\boldsymbol{G}, \boldsymbol{k}, \boldsymbol{\varepsilon}$)

输入:一个无向带权图 $G = (V,E)$,一个正整数 $k$ 和一个正实数 $\varepsilon (0 < \varepsilon < 1)$;

输出:图 $G$ 的一个权值最大的 $k$-Cut 或者报告不存在 $k$-Cut;

1 $Q_0 = \varnothing$; $S_0 = \varnothing$; $N = \lceil \ln(1/\varepsilon) \rceil$;

2 **for** $i = 1$ **to** $N \times 2^k$ **do**

2.1 将顶点集 $V$ 随机地划分为子集 $V_L$ 和 $V_R$,使得 $V$ 中任意点 $v$ 以 1/2 的概率放入 $V_L$,以 1/2 的概率放入 $V_R$;

2.2 设 $Q$ 为 $V_L$ 和 $V_R$ 之间边的集合;

2.3 **if** $Q$ 中含有至少 $k$ 条边 **then**

{保留 $Q$ 中权值最大 $k$ 条边而删除其余边;

2.4 **if** $Q$ 中边的权值和大于 $Q_0$ 中边的权值和

**then** { $Q_0 = Q$; $S_0 = V_L$ } }

3 **if** $Q_0 \neq \varnothing$ **then** 返回($S_0$, $V$-$S_0$)

**else** 报告不存在 $k$-Cut。

**图 1 参数化带权最大割问题的一种随机算法**

**定理 1** 对于参数化带权最大割问题的一个实例($G$, $k$, $\varepsilon$),如果存在 $k$-Cut,则算法 **MaxCut1**($\boldsymbol{G}$, $\boldsymbol{k}$, $\boldsymbol{\varepsilon}$)能够在时间 $O^*(\ln(1/\varepsilon)2^k)$ 内以至少 $1 - \varepsilon$ 的概率返回图 $G$ 的一个权值最大的 $k$-Cut。

证明:首先证明二分连通图的一个重要性质:设 $G_c = (V_c, E_c)$为一个二分连通图,则 $|V_c| \leqslant |E_c| + 1$。

由于图 $G_c$ 是一个连通的二分图,因此在图 $G_c$ 上一定可以找到一些没有公共点的子图(这些子图或者为极大路径或者为孤立点),使得 $V_c$ 中的任意一点属于且仅属于其中的一个子图。不失一般性,设这些子图为 $P_1, P_2, \cdots, P_l$, $1 \leqslant l \leqslant |V_c|/2$。显然对于任意 $i (1 \leqslant i \leqslant l)$,子图 $P_i$ 上的边数 $m_i$ 与顶点数 $n_i$ 有关系式 $m_i = n_i - 1$。同时由于图 $G_c$ 是一个连通图,则图 $G_c$ 上必定至少存在 $l - 1$ 条边将子图 $P_1, P_2, \cdots, P_l$ 连接起来,因此图 $G_c$ 的边数 $m$ 和顶点数 $n$ 满足关系式 $m \geqslant \sum_{i=1}^{l} m_i + (l - 1) = \sum_{i=1}^{l}(n_i - 1) + (l - 1) = n - l + (l - 1) = n -$

1。故关系式 $|V_c| \leqslant |E_c| + 1$ 成立。

设 $E_{max} = \{d_1, d_2, \cdots, d_k\}$ 为图 $G$ 上一个权值最大的 $k$-Cut($S, T$)中权值最大的 $k$ 条边,对于顶点 $V$ 的任意一次随机划分 $(V_L, V_R)$,接下来证明,$E_{max}$ 正好属于割($V_L, V_R$)的概率至少为 $1/2^k$。

设 $(l_1, r_1), (l_2, r_2), \cdots, (l_k, r_k)$ 为边集 $E_{max} = \{d_1, d_2, \cdots, d_k\}$ 中各对应边的端点,且对于满足 $1 \leqslant i \leqslant k$ 的 $i$, $l_i \in S$, $r_i \in T$。并设 $L = \{l_1, l_2, \cdots, l_k\}$, $R = \{r_1, r_2, \cdots, r_k\}$, $V_b = L \cup R$, $E_b = \{(l_i, r_i) | 1 \leqslant i \leqslant k\}$,显然 $G_b = (V_b, E_b)$ 必定构成图 $G$ 上的一个二分子图。由于 $\{d_1, d_2, \cdots, d_k\}$ 中的一些边可能存在公共端点,因此集合 $L$ 和 $R$ 中都可能不含有 $k$ 个不同的点。假设子图 $G_b$ 含有 $t$ 个连通的部件 $C_1, C_2, \cdots, C_t$,并且对于其中任意一个部件 $C_i (1 \leqslant i \leqslant t)$,设 $C_i = (V_i, E_i)$, $n_i = |V_i|$, $m_i = |E_i|$。在图 $G_b$ 上显然 $\sum_{i=1}^{t} m_i = k$,同时前面已经证明,对于任意 $1 \leqslant i \leqslant t$, $n_i \leqslant m_i + 1$。如果将图 $G$ 的顶点集 $V$ 随机地划分为两个子集 $V_L$ 和 $V_R$,使得 $V$ 中的任意一点放入 $V_L$ 的概率为 1/2,放入 $V_R$ 的概率也为 1/2,则对于二分子图 $G_b$ 的每一个连通部件 $C_i (1 \leqslant i \leqslant t)$ 来说,$C_i$ 中的顶点 $V_i$ 被恰当划分(即或者 $L \cap V_i \subseteq V_L$, $R \cap V_i \subseteq V_R$ 或者 $L \cap V_i \subseteq V_R$, $R \cap V_i \subseteq V_L$)的概率为 $2/2^{n_i}$。因此子图 $G_b$ 上 $t$ 个连通部件中的顶点同时被恰当划分的概率为 $2/2^{n_1} \times 2/2^{n_2} \times \cdots \times 2/2^{n_t} \geqslant 2/2^{m_1+1} \times 2/2^{m_2+1} \times \cdots \times 2/2^{m_t+1} = 2^t/2^{\sum_{i=1}^{t} m_i + t} = 1/2^k$。该概率也就是 $E_{max}$ 正好属于划分($V_L$, $V_R$)的概率。

有了前述基础,下面证明算法 **MaxCut1**($\boldsymbol{G}, \boldsymbol{k}, \boldsymbol{\varepsilon}$)的正确性。

设对图 $G$ 的顶点 $V$ 进行一次随机划分为一次随机试验 $Y$,找到图 $G$ 的一个权值最大的 $k$-Cut 为事件 $B$,从前述证明可以看出,在一次随机试验 $Y$ 中,事件 $B$ 发生的概率至少为 $1/2^k$。如果进行足够多次随机试验 $Y$,事件 $B$ 必定以很大的概率发生。算法 **MaxCut1**($\boldsymbol{G}, \boldsymbol{k}, \boldsymbol{\varepsilon}$)正好实现了这种思想。下面重点分析算法 **MaxCut1**($\boldsymbol{G}, \boldsymbol{k}, \boldsymbol{\varepsilon}$)执行一次结束时,事件 $B$ 发生的概率。

从图 1 可以看出,算法 **MaxCut1**($\boldsymbol{G}, \boldsymbol{k}, \boldsymbol{\varepsilon}$)的一次执行过程实际上是做一次($N \times 2^k$)重贝努利试验。根据概率论的有关知识,在一次($N \times 2^k$)重贝努利试验中,事件 $B$ 不发生的概率为 $(1 - 1/2^k)^{N \times 2^k}$。根据公式 $\lim_{n \to \infty}(1 + 1/n)^n = e$,容易推导 $(1 - 1/2^k)^{N \times 2^k} < e^{-N}$,又 $N = \lceil \ln(1/\varepsilon) \rceil$,可得 $(1 - 1/2^k)^{N \times 2^k} < e^{-N} < \varepsilon$。因此事件"找到一个权值最大的 $k$-Cut"发生的概率至少为 $1 - \varepsilon$。

接下来分析算法 **MaxCut1**($\boldsymbol{G}, \boldsymbol{k}, \boldsymbol{\varepsilon}$)的时间复杂度。

算法的第 2.1 步通过调用一个标准的随机函数可以在时间 $O(1)$ 内实现,算法的第 2.2 步找出一个割中的所有边可以在时间 $O(m)$($m$ 为图 $G$ 中边的条数)内实现,算法的 2.3 步选择权值最大的 $k$ 条边可在时间 $O(km)$ 内完成。因此整个算法的时间复杂度为 $O(\ln(1/\varepsilon)2^k(m + km)) = O^*(\ln(1/\varepsilon)2^k)$。

## 3 一种基于($n, k$)-全集划分技术的确定性算法

前一部分对参数化带权的最大割问题提出了一个随机算法,在此基础上接下来研究它的一种确定性算法。

由文献[5]可知,给定一个含有 $n$ 个顶点、$m$ 条边的图 $G$,可以在时间 $O(m + n)$ 内找到图 $G$ 的一个大小至少为 $\lceil m/2 \rceil$ 的割。因此对于参数化带权最大割问题一个给定的实例($G = (V, E), k$),要确定性地找到一个权值最大的 $k$-Cut,可以分两种不同情况讨论。

(1) 当 $k > \lceil m/2 \rceil$ 时,即 $m < 2k$,从 $m$ 条边中任选 $k$ 条边的组合数 $\binom{m}{k} \leqslant 2^m < 2^{2k}$。因此如果图 $G$ 存在一个 $k$-Cut,可以从边的 $\binom{m}{k}$ 种不同组合中选择出一种权值和最大的组合,并进一步构造出图 $G$ 的一个权值最大的 $k$-Cut。

(2) 当 $k \leqslant \lceil m/2 \rceil$ 时,显然图 $G$ 一定存在 $k$-Cut。但关系式 $\binom{m}{k} < 2^{2k}$ 不成立,枚举出图 $G$ 的所有 $k$-Cut 的计算复杂性目前也还是未知的。此时可运用($n, k$)-全集技术来寻找图 $G$ 的一个权值最大的 $k$-Cut。

下面简要地给出($n, k$)-全集技术的相关定义和定理。

**定义 6**[14] ($n, k$)-全集是一个向量集合 $T \subset \{0,1\}^n$,使得对于任意索引子集 $W \subseteq \{1,2,\cdots,n\}$ 且 $|W| = k$,$T$ 在 $W$ 上的映射包含了 $k$ 位二进制数的所有可能的 $2^k$ 种状态值。

人们在运用($n, k$)-全集技术设计参数化算法时,寻求的目标是希望($n, k$)-全集中含有向量的个数具有 $f(k)n^{O(1)}$ 的表示形式并且尽可能少。

($n, k$)-全集技术最初是由 Naor 等人在文献[14]

中提出来的,并证明了对于任意两个整数 $n,k(k\geqslant 1,n\geqslant k)$ 都可以在线性时间内构造一个大小为 $2^{k}k^{O(\log k)}\log n$ 的 $(n,k)$-全集,最近 Chen 等人在文献[11]中明显改进了该技术,并得到了下述结论。

**引理 1**[11] 对于任意两个整数 $n,k(k\geqslant 1,n\geqslant k)$,存在一个大小为 $O(n2^{k+12\log^2 k-4\log k})$ 的 $(n,k)$-全集,并且它可以在时间 $O(n2^{k+12\log^2 k-4\log k})$ 内构造。

进一步地,对于给定的一个图 $G=(V,E)$ 和 $E$ 中任意 $k$ 条边至多 $2k$ 个端点的集合 $V'$,根据引理 1,总存在 $V$ 的一个划分函数族 $F(V,2k)$(即一个 $(n,2k)$-全集),使得 $V'$ 的任意划分都可由 $F(V,2k)$ 中的一个划分函数映射到 $V'$ 而得到。

综合上述分析,可以得到求解参数化带权最大割问题的一个确定性算法。图 2 详细描述了该算法的具体步骤。其中算法的 2.1 步至 2.5 步是对情况 $k>\lceil |E|/2\rceil$ 的处理,算法的 3.1 步至 3.5 步是对情况 $k\leqslant\lceil |E|/2\rceil$ 的处理。

算法 **MaxCut2**$(G,k)$

输入:一个无向带权图 $G=(V,E)$,一个正整数 $k$;

输出:图 $G$ 的一个权值最大的 $k$-Cut 或者报告不存在 $k$-Cut;

1 $Q_0=\varnothing$; $S_0=\varnothing$;

2 **if** $k>\lceil |E|/2\rceil$ **then**

2.1 列举 $E$ 中所有含有 $k$ 条边的子集 $H_1,H_2,\cdots,H_r$;

2.2 **for** $i=1$ **to** $r$ **do**

2.3 **if** $(V,H_i)$构成一个二分图 **then**

2.4 { **if** $H_i$ 中边的权值和大于 $Q_0$ 中边的权值和

2.5 **then** { 构造 $G$ 的一个割$(V_L,V_R)$; $Q_0=H_i$; $S_0=V_L$ } }

3 **else**

3.1 构造 $V$ 的一个$(n,2k)$-全集 $F(V,2k)$;

3.2 **for** $F(V,2k)$中的每一个划分$(V_L,V_R)$ **do**

3.3 设 $Q$ 为 $V_L$ 和 $V_R$ 之间的边的集合;

3.4 **if** $Q$ 中含有至少 $k$ 条边 **then** {保留 $Q$ 中 $k$ 条权值最大边而删除其余边;

3.5 **if** $Q$ 中边的权值和大于 $Q_0$ 中边的权值和 **then** { $Q_0=Q$; $S_0=V_L$ } }

4 **if** $Q_0\neq\varnothing$ **then** 返回 $(S_0,V-S_0)$ **else** 报告不存在 $k$-Cut。

**图 2 参数化带权最大割问题的一种确定性算法**

**定理 2** 算法 **MaxCut2**$(G,k)$ 能在时间 $O^*(2^{2k+12\log^2(2k)})$ 内确定性地求解参数化带权的最大割问题。

证明:首先根据参数 $k$ 的大小分两种情况分别证明算法 **MaxCut2**$(G,k)$的正确性。

当 $k>\lceil m/2\rceil$ 时,即 $m<2k$,可以从边集 $E$ 中找出所有正好含有 $k$ 条边的边子集 $H_1,H_2,\cdots,H_r$,其中 $r=\binom{m}{k}$,然后依次判定子图$(V,H_i)(1\leqslant i\leqslant r)$是否构成一个二分子图,同时从构成二分子图的所有边子集中选择一个权值和最大的边子集,便可进一步构造出图 $G$ 的一个权值最大的 $k$-Cut。算法的 2.1 步至 2.5 步正好实现了该思想。

下面详细给出算法 2.5 步构造图 $G$ 一个 $k$-Cut 的具体过程及其相关证明。算法执行到 2.4 步时,表明 $(V,H_i)$ 构成一个二分图。根据二分图的定义,边集 $H_i$ 的端点可分为两个互不相交的子集 $V_1$, $V_2$,使得 $H_i$ 中任意一条边的两个端点分别属于 $V_1$ 和 $V_2$。于是设 $V_L=V_1$, $V_R=(V-V_1)$,则 $(V_L,V_R)$ 一定构成图 $G$ 的一个 $k$-Cut。这是因为,对于图 $G=(V,E)$来说,顶点集 $V$ 事实上分为了 $V_1$, $V_2$ 和 $V_3=V-V_1-V_2$ 三部份,并且 $V_R=V_2\cup V_3$。边集 $E$ 相应地分为 $E_1=\{d_{12}\mid d_{12}$ 为 $V_1$ 与 $V_2$ 之间的边$\}$、$E_2=\{d_{13}\mid d_{13}$ 为 $V_1$ 与 $V_3$ 之间的边$\}$和 $E_3=\{d_{23}\mid d_{23}$ 为 $V_2$ 与 $V_3$ 之间的边$\}$。显然 $E_1=H_i$, $E_2$ 属于割$(V_L,V_R)$中,$E_3$ 中各边的端点都属于 $V_R$。由定义 1 和定义 2 可知,$(V_L,V_R)$ 构成图 $G$ 的一个 $k$-Cut。

当实例不存在一个 $k$-Cut 时,任意$(V,H_i)$必定不构成 $G$ 的一个二分子图。算法的 2.3 步进行了判定和控制,算法的 2.4 步将不执行,$Q_0$ 的值在算法的执行过程中始终为空集 $\varnothing$。因此算法结束时,正确返回不存在。

当 $k\leqslant\lceil m/2\rceil$ 时,显然实例一定存在一个 $k$-Cut,现设$(S,T)$为 $G$ 的一个权值最大的 $k$-Cut, $V'$ 为$(S,T)$中权值最大 $k$ 条边的 $2k$ 个端点。如果对图 $G$ 的顶点 $V$ 构造一个$(n,2k)$-全集,即构造 $V$ 的一个划分函数族 $F(V,2k)$,根据引理 1, $V'$ 的任意一种划分状态都可由划分函数族 $F(V,2k)$ 中的一个划分函数映射到 $V'$ 而得到,因此一个权值最大的 $k$-Cut 必定存在 $F(V,2k)$ 中。算法的 3.1 至 3.5 步具体地实现了该思想。其中 3.1 步调用文献[11]中的算法构造一个$(n,2k)$-全集 $F(V,2k)$, 3.2 至 3.5 步对 $F(V,2k)$中的所有划分依次判定和选择,找到的 $k$-Cut 必定是图 $G$ 的一个权值最大的 $k$-Cut。

接下来分析算法 **MaxCut2**$(G,k)$的时间复杂

度。

首先分析该算法第2步的运行时间。算法的第2.1步列举图 $G$ 的所有正好含有 $k$ 条边的边组合。由于 $k > \lceil m/2 \rceil$，即 $m < 2k$，可得 $\binom{m}{k} \leqslant 2^m < 2^{2k}$，因此第2.1步可在时间 $O(2^{2k})$ 内完成。算法的第2.3步判定一个图是否为二分图可以通过使用两种不同标号来标识相邻顶点的简单算法来实现，即可在时间 $O(m)$ 内完成。第2.4步构造一个 $k$-Cut 可以在时间 $O(1)$ 内完成，比较两个 $k$-Cut 权值的大小可以在时间 $O(k)$ 内完成。因此算法第2步的运行时间为 $O(2^{2k}) + O(2^{2k}k) = O(k2^{2k}) = O^*(2^{2k})$。

算法的第3.1步根据引理1可在时间 $O(n2^{2k+12\log^2(2k)-4\log(2k)})$ 内完成。算法的3.3至3.5步的分析类似算法 **MaxCut1**($\boldsymbol{G}, \boldsymbol{k}, \boldsymbol{\varepsilon}$)中2.2至2.4步的分析，可在时间 $O(m + km)$ 内完成。于是 **MaxCut2**($\boldsymbol{G}, \boldsymbol{k}$)的3.1步至3.5步的运行时间为

$$\begin{aligned} &O(n2^{2k+12\log^2(2k)-4\log(2k)}) + O(n2^{2k+12\log^2(2k)-4\log(2k)} \\ &\quad \times (m + km)) = O(km2^{2k+12\log^2(2k)})) \\ &\quad = O^*(2^{2k+12\log^2(2k)})。\end{aligned}$$

因此算法 **MaxCut2**($\boldsymbol{G}, \boldsymbol{k}$)的时间复杂度为 $O^*(2^{2k+12\log^2(2k)})$。

值得指出的是，算法 **MaxCut2**($\boldsymbol{G}, \boldsymbol{k}$)综合考虑了两个方面的因素：(1) 算法的时间复杂度。虽然该算法的理论时间为 $O^*(2^{2k+12\log^2(2k)})$，但当 $k > \lceil m/2 \rceil$ 时算法的实际时间为 $O*(4^k)$。(2) 算法的实际应用。运用该算法求解带权的最大割问题时，由于实例中权值最大割的规模可能小于 $\lceil m/2 \rceil$，因此算法对参数值 $k \leqslant \lceil m/2 \rceil$ 的处理是必需的。

## 4 结 论

本文首次运用参数计算理论和技术对带权的最大割问题进行了专门研究。首先根据参数计算理论对该问题给出了正式的参数化定义，接着运用随机划分技术提出了一个时间复杂度 $O^*(\ln(1/\varepsilon)2^k)$ 随机算法，它能以任意小的概率误差 $\varepsilon$ 找到目标解。然后在此基础上着重运用最新的($n, k$)-全集技术提出了一个时间复杂度为 $O^*(2^{2k+12\log^2(2k)})$的确定性算法，表明了参数化带权的最大割问题是固定参数可解的。

如何在深入分析问题结构特点的基础上对该问题提出更加有效的固定参数可解算法是我们下一阶段的研究重点。

**参考文献**


[1] Karp R M. On reducibility among combinatorial problems. In: Miller RE, Thatcher JW (editors). Complexity of Computer Computations. New York: Plenum, 1972. 85-103

[2] Vitanyi P M. How well can a graph be n-colored? *Discrete Mathematics*, 1981, 34:69-80

[3] Hagin D J, Venkatesan S M. Approximation and intractability results for the maximum cut problem and its variants. *IEEE Transactions on Computers*, 1991, 40:110-113

[4] Goemans M X, Williamson D P. Improved approximation algorithms for Maximum Cut and Satisfiability problems using semidefinite programming. *Journal of the ACM*, 1995, 42(6): 1115-1145

[5] Mahajan M, Raman V. Parameterizing above guaranteed values: MaxSat and MaxCut. *Journal of Algorithms*, 1999, 31(2):335-354

[6] Prieto E. The method of extremal structure on the $k$-Maxmum Cut problem. In: Proceedings of Computing: the Australasian Theory Symposium, Australian Computer Society, Darlinghurst, Australia, 2005. 119-126

[7] Raman V, Saurabh S. Improved fixed parameter tractable algorithms for two edge problems: MAXCUT and MAXDAG. *Information Processing Letters*, 2007, 104:65-72

[8] Garey M R, Johnson D S. Computers and Intractability: A Guide to the Theory of NP-Completeness. New York, USA: W. H. Freeman & Co, 1979

[9] Fernau H. Parameterized Maximization: [technical report]. Germany, Tübingen: Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, 2001

[10] Cai L, Chan S M, Chan S O. Random separation: a new method for solving fixed-cardinality optimization problems. In: Proceedings of the 2nd International Work shop on Parameterized and Exact Computation, Zürich, Switzerland, 2006. 239-250

[11] Chen J, Lu S. Improved algorighms for weighted and unweighted set splitting problems. In : Proceedings of the 13th Annual International Computing and Combinatorics Conference, Banff, Alberta, Canada, 2007. 537-547

[12] Chen J. Randomized disposal of unknowns and implicitly enforced bounds on parameters. In: Proceedings of the 3nd International Workshop on Parameterized and Exact Computation, Victoria, British Columbia, Canada, 2008. 1-8

[13] Chen J, Kneis J, Lu S, et al. Randomized divide-and-conquer: Improved path, matching, and packing algorithms. *SIAM Journal on Computing*, 2009, 38(6):2526-2547

[14] Naor M, Schulman J, Srinivasan A. Splitters and near-opti-

mal derandomization. In: Proceedings of the 36th Annual Symposium on Foundations of Computer Science, Milwaukee, Wisconsin, 1995.182-191

# Fixed-parameter tractable algorithms based on partition for the weighted maximum cut problem

Liu Yunlong[*][**], Wang Jianxin[*]

([*]School of Information Science and Engineering, Central South University, Changsha 410083)

([**]School of Continuing Education, Hunan Normal University, Changsha 410013)

**Abstract**

This paper studies the weighted maximum cut problem in terms of the parameterized computational complexity theory. After defining the parameterized version of the problem, the paper presents a randomized algorithm based on random separation for the parameterized problem. The algorithm randomly bipartitions the vertex set of a given instance for $\lceil \ln(1/\varepsilon) \rceil \times 2^k (0 < \varepsilon < 1)$ times, and returns a $k$-cut of maximum weight as the output, so that it can obtain the solution with the probability of at least $1 - \varepsilon$ in time $O^*(\ln(1/\varepsilon)2^k)$. On the basis of the study above, the paper also proposes a deterministic parameterized algorithm with the time complexity of $O^*(2^{2k+12\log^2(2k)})$ by mainly employing the recent improved $(n, k)$-universal set techniques, which shows that the maximum cut problem is fixed-parameter tractable.

**Key words**: weighted maximum cut, fixed-parameter tractable, random separation, $(n, k)$-universal set