arXiv:2303.12560v1 [math.CO] 22 Mar 2023

有界树宽下的度序列优化

Shmuel Onn *†

摘要

我们研究这样一个问题：在给定图中寻找一个子图，使各顶点处给定函数在其子图度数上的取值之和最小。尽管当所有函数相同时，该问题已是 NP难问题，但我们证明，对于树宽有界的图，任意函数下该问题均可在多项式时间内求解。该问题的复杂度仍远未明了，尤其是在完全图和完全二部图上的复杂度。

关键词：图、组合优化、度序列、因子、匹配

1 引言

简单图 $G=(V,E)$ 在 $V=[n]:=\{1,\dots,n\}$ 上的度序列是向量$d(G)=(d_1(G)\dots,d_n(G))$，其中 $d_i(G):=|\{e\in E:\ i\in e\}|$ 是顶点 i 的度，对所有 i均成立。

本文关注以下离散优化问题。

度序列优化。给定一个图 H 在 $[n]$上，以及对 $i\in[n]$ 的函数$f_i:\{0,1,\dots,n-1\}\to\mathbb{Z}$，求子图 $G\subseteq H$在 $[n]$上的 $\sum_{i=1}^n f_i(d_i(G))$ 的最小值。

对于图 G，我们分别用 $V(G)$ 和 $E(G)$ 表示其顶点集和边集。下文中，对于图 G,H，我们用 $G\subseteq H$ 表示 $V(G)=V(H)$，以及 $E(G)\subseteq E(H)$。

我们问题的一种特殊情形是一般因子问题 [2], ，其目标是判定：给定H以及对所有 i的子集 $B_i\subseteq\{0,1,\dots,n-1\}$，是否存在一个 $G\subseteq H$，使得对所有 i都有 $d_i(G)\in B_i$。事实上，对于每个 i，定义 $f_i(x):=0$ 当 $x\in B_i$ 时成立，并定义 $f_i(x):=1$ 当 $x\notin B_i$时成立。那么，当且仅当存在一个因子时，我们问题的最优值为零。一个更特殊的情形是 Lov´asz [7], 引入的（l,u）-因子问题，其中每个 $B_i=\{l_i,\dots,u_i\}$ 都是一个区间。即使采用凸函数，这个问题也可归约为我们的问题，其中 $f_i(x):=l_i-x$ 当 $0\le x\le l_i$时成立，$f_i(x):=0$ 当$l_i\le x\le u_i$时成立，并且 $f_i(x):=x-u_i$ 当 $u_i\le x\le n-1$时成立。 b-匹配问题是对所有 i 而言 $B_i=\{b_i\}$ 均为单元素集合的情形，而完美匹配问题则是对所有 i而言 $b_i=1$ 均成立的情形。

当所有 f_i 均为凸函数[1, 5]时；当所有 f_i 相同且 $H=K_n$ 为完全 [4]时；以及当 $H=K_{m,n}$ 为完全二部图 [6]时，对于单调子图上的任意函数，度序列优化均可在多项式时间内完成。对于一般图，即使除一个函数外的所有函数都相同，该问题仍为 NP难，因为 H 含有非空三次子图（判定这一点是 NP完全的），当且仅当存在某个 i，使得以 $f_i(x):=(x-3)^2$ 和 $f_j(x):=x(x-3)^2$（其中 $j\neq i$）进行度序列优化的最优值等于零，见 [5]。

与此相反，我们在此证明如下结论。

*以色列理工学院。电子邮箱：onn@technion.ac.il

†我的手稿没有相关数据。

定理 1.1 对任意固定的 k，对于任意 n、任意树宽由 k 界定的图 H（其顶点集为 $[n]$），以及任意函数 f_1，……，f_n，度序列优化问题均可在多项式时间内求解。

我们猜想，对于所有函数 f_i，在完全图 $H = K_n$ 和完全二部图 $H = K_{m,n}$ 上进行度序列优化都可以在多项式时间内完成。

2 证明

我们需要一些记号。设 $H = ([n], E)$ 是一个以 $[n]$ 为顶点的图。H 的优美树分解是一个有根树 T，其中每个节点 $v \in V(T)$ 都有一个集合 $I_v \subseteq [n]$，称为 v 的包，满足：每个 $i \in [n]$ 都属于某个包；每条边 $\{i, j\} \in E$ 都包含在某个包中；如果 $u, v, w \in V(T)$，其中 v 位于 T 中 u 与 w 之间的路径上，且 $i \in I_u \cap I_w$，那么 $i \in I_v$；T 的根 r 和每个叶节点 l 都满足 $I_r = I_l = \emptyset$；每个非叶节点 $v \in V(T)$ 都属于以下类型之一：

- 引入节点，表示它有一个子节点 u，且对于某个 $i \in [n]$，$I_v = I_u \uplus \{i\}$；
- **fo**根节点意味着它只有一个子节点 u 和 $I_u = I_v \uplus \{i\}$，其中某个 $i \in [n]$ ；
- 合并节点意味着它有两个子节点 u, w 和 $I_u = I_v = I_w$。

树的宽度是包的最大基数减一。H的树宽记为 $tw(H)$，是其优美树分解的最小宽度。已知，对于任意固定的 k，如果 H 的树宽至多为 k，则可以在多项式时间内计算出 H 的一个宽度至多为 k的优美树分解 T，且 $|V(T)| = O(n)$。更多详情请参见 [3]。因此，从这里开始，我们可以假设已知 H的一个优美树分解 T。

我们令 $H[I]$ 为由 $I \subseteq [n]$诱导的 H 的子图。对于 $v \in V(T)$，令 T_v 为以 v 为根的 T 的子树，并且 $I(T_v) := \cup\{I_u : u \in V(T_v)\}$ 中所有节点的包的并集。

对于每个 $v \in V(T)$、函数 $c_v : I_v \to \{0, 1, \ldots, n-1\}$ 和子集 $F_v \subseteq E(H[I_v])$，令

$$g(v, c_v, F_v) := \min \left\{ \sum_{i \in I(T_v)} f_i(d_i(G)) \ : \ G \subseteq H[I(T_v)],\ E(G[I_v]) = F_v,\ d_i(G) = c_v(i),\ i \in I_v \right\},$$

若集合为空，则取为 ∞。注意，在树根 r 处，我们有 $I(T_r) = [n]$，因此 $H[I(T_r)] = H$，且 $I_r = \emptyset$ 因此 F_r 必须是空集，而 $c_r = c_\emptyset$ 是定义在 $\emptyset$ 上的无值函数，因此 $g(r, c_\emptyset, \emptyset) = \min\{\sum_{i=1}^{n} f_i(d_i(G)) : G \subseteq H\}$ 是我们问题的最优解。

函数 $g(v, c_v, F_v)$ 是度序列优化问题的一个子问题的最优值。该子问题限定在图 $H[I(T_v)]$ 的子图 G 上，这些子图由顶点集 $I(T_v)$ 诱导得到；该顶点集是以 v 为根的子树 T_v 中所有节点的包的并集。此外，还对 G 施加了进一步限制，要求其由 v 的包 I_v 诱导出的边子集 $E(G[I_v])$ 等于F_v，且每个顶点 $i \in I_v$ 的度 $d_i(G)$ 等于 $c_v(i)$。利用这些函数，我们可以从树叶向根遍历这棵树，并通过递归计算，求出树中任意节点 v 的所有值 $g(v, c_v, F_v)$；计算时使用此前在 v 的所有后代节点u 处求得的值 $g(u, c_u, F_u)$。证明通过说明如何针对三种非叶节点分别进行这些计算来展开。特别地，在引入节点 v 或合并节点 v处（此处的计算比遗忘节点处更复杂），我们将定义一个辅助函数 $h(v, c_v, F_v)$，利用此前在 v 的后代节点 u 处求得的值 $g(u, c_u, F_u)$ 给出该函数的显式公式，然后论证 $g(v, c_v, F_v) = h(v, c_v, F_v)$。

接下来，我们说明如何从树叶向根计算所有 $g(v, c_v, F_v)$，从而求解度优化问题。在任意树叶 l 处，我们有 $I(T_l) = I_l = \emptyset$ 和 $F_l = \emptyset, c_l = c_\emptyset$，且该集合中唯一的图为 $G_\emptyset := (\emptyset, \emptyset)$，并且 $g(l, c_\emptyset, \emptyset) = \sum_{i \in \emptyset} f_i(d_i(G_\emptyset)) = 0$。
引入节点 v，其子节点为 u 和 $I_v = I_u \uplus \{i\}$。对于 I_v 上的任意 c_v 以及 $F_v \subseteq E(H[I_v])$，令$F_u := F_v \cap E(H[I_u])$，并在 I_u 上定义 c_u 如下：

$$c_u(j) := \begin{cases} c_v(j), & \text{if } \{i,j\} \notin F_v; \\ c_v(j) - 1, & \text{if } \{i,j\} \in F_v, \end{cases}$$

并令

$$h(v, c_v, F_v) := \begin{cases} \infty, & \text{if } c_v(i) \neq |\{j \in I_u : \{i,j\} \in F_v\}| \text{ or } c_u(j) < 0 \text{ for some } j \in I_u; \\ g(u, c_u, F_u) + \sum_{j \in I_v} f_j(c_v(j)) - \sum_{j \in I_u} f_j(c_u(j)), & \text{otherwise.} \end{cases}$$

现在证明 $g(v, c_v, F_v) = h(v, c_v, F_v)$。首先证明 $g(v, c_v, F_v) \geq h(v, c_v, F_v)$。如果$g(v, c_v, F_v) = \infty$，则显然成立。否则，选取满足 $E(G_v[I]) = F_v$且对所有 $j \in I_v$都有 $d_j(G_v) = c_v(j)$ 的 $G_v \subseteq H[I(T_v)]$，使其取得 $\sum_{j \in I(T_v)} f_j(d_j(G_v)) = g(v, c_v, F_v)$。由于 T是 H的树分解，因此在 $G_v \subseteq H[I(T_v)]$中不存在具有 $j \notin I_u$的边 $\{i, j\}$，于是$c_v(i) = |\{j \in I_u : \{i,j\} \in F_v\}|$。令 $G_u = G_v[I(T_u)]$。那么 $E(G[I_u]) = F_u$，并且对于每个 $j \in I_u$,

$$c_u(j) = \begin{cases} c_v(j) = d_j(G_v) = d_j(G_u), & \text{if } \{i,j\} \notin F_v; \\ c_v(j) - 1 = d_j(G_v) - 1 = d_j(G_u), & \text{if } \{i,j\} \in F, \end{cases}$$

因此，对所有 $j \in I_u$，$c_u(j) = d_j(G_u)$，特别地 $c_u(j) \geq 0$。因此

$$\begin{aligned} g(v, c_v, F_v) &= \sum_{j \in I(T_v)} f_j(d_j(G_v)) \\ &= \sum_{j \in I(T_u)} f_j(d_j(G_u)) + \sum_{j \in I_v} f_j(c_v(j)) - \sum_{j \in I_u} f_j(c_u(j)) \\ &\geq g(u, c_u, F_u) + \sum_{j \in I_v} f_j(c_v(j)) - \sum_{j \in I_u} f_j(c_u(j)) \\ &= h(v, c_v, F_v) . \end{aligned}$$

接下来我们证明 $g(v, c_v, F_v) \leq h(v, c_v, F_v)$。若 $h(v, c_v, F_v) = \infty$，则结论成立。否则

$$h(v, c_v, F_v) = g(u, c_u, F_u) + \sum_{j \in I_v} f_j(c_v(j)) - \sum_{j \in I_u} f_j(c_u(j))$$

且 $g(u, c_u, F_u)$ 是有限的。选取满足 $E(G_u[I]) = F_u$ 且 $d_j(G_u) = c_u(j)$ 的 $G_u \subseteq H[I(T_u)]$，使其达到 $\sum_{j \in I(T_u)} f_j(d_j(G_u)) = g(u, c_u, F_u)$。令 $G_v \subseteq H[I(T_v)]$ 满足 $G_v[I(T_u)] = G_u$ 且 $E(G_v) \cap E(H[I_v]) = F_v$。那么，对于 $j \in I_v$，$d_j(G_v) = c_v(j)$，并且

$$\begin{aligned} g(v, c_v, F_v) &\leq \sum_{j \in I(T_v)} f_j(d_j(G_v)) \\ &= \sum_{j \in I(T_u)} f_j(d_j(G_u)) + \sum_{j \in I_v} f_j(c_v(j)) - \sum_{j \in I_u} f_j(c_u(j)) \\ &= g(u, c_u, F_u) + \sum_{j \in I_v} f_j(c_v(j)) - \sum_{j \in I_u} f_j(c_u(j)) = h(v, c_v, F_v) . \end{aligned}$$

我们需要针对 c_v 的 $n^{|I_v|} = O(n^{k+1})$ 个取值计算 $h(v, c_v, F_v)$，以及计算 $O(2^{\binom{|I_v|}{2}}) = 2^{O(k^2)}$ 个可能的集合 F_v；每次计算都涉及常数次运算，因此整体为多项式时间。

遗忘节点 de v 有子节点 u 和 $I_u = I_v \uplus \{i\}$。那么，对于任意定义在 I_v 上的 c_v 和 $F_v \subseteq E(H[I_v])$，

$$g(v, c_v, F_v) \;=\; \min\{g(u, c_u, F_u) : F_u \subseteq E(H[I_u]),\ F_u \cap E(H[I_v]) = F_v,\ c_u(j) = c_v(j),\ j \in I_v\}\ .$$

我们需要针对 c_v 和 $O(2^{\binom{|I_v|}{2}})$ 的 $n^{|I_v|} = O(n^k)$ 个取值计算此结果，每次计算都要在大小为 $O(n^2)$ 的集合上取最小值，因此总时间为多项式级。$= 2^{O(k^2)}$ 个可能的 F_v，

合并节点 v 有子节点 u、w 和 $I_u = I_v = I_w$。对于定义在 I_v 上的任意 c_v 和 $F_v \subseteq E(H[I_v])$，令

$$\begin{aligned} h(v, c_v, F_v) \;:=\; & \min\{g(u, c_u, F_u) + g(w, c_w, F_w) \\ & +\; \sum_{i \in I_v} (f_i(c_v(i)) - f_i(c_u(i)) - f_i(c_w(i)))\ :\ F_v = F_u \uplus F_w,\ c_v = c_u + c_w\}\ , \end{aligned}$$

该集合遍历所有划分 $F_v = F_u \uplus F_w$ 和函数分解 $c_v = c_u + c_w$。

现在我们证明 $g(v, c_v, F_v) = h(v, c_v, F_v)$。令 $I := I_u = I_v = I_w$。令 $I_u^c := I(T_u) \setminus I$和 $I_w^c := I(T_w) \setminus I$。首先证明 $g(v, c_v, F_v) \geq h(v, c_v, F_v)$。若 $g(v, c_v, F_v) = \infty$，则结论显然成立。否则，选取 $G_v \subseteq H[I(T_v)]$，使得 $E(G_v[I]) = F_v$ 且对所有$i \in I$都有 $d_i(G_v) = c_v(i)$，并且该选择达到 $\sum_{i \in I(T_v)} f_i(d_i(G_v)) = g(v, c_v, F_v)$。考虑任意划分 $F_v = F_u \uplus F_w$。令 $G_u \subseteq G_v[I(T_u)]$ 为满足 $E(G_u) = E(G_v[I(T_u)]) \setminus F_u$的子图，并令 $G_w \subseteq G_v[I(T_w)]$为满足 $E(G_w) = E(G_v[I(T_w)]) \setminus F_u$的子图。在 I 上定义函数 c_u 和 c_w，使得对所有 $i \in I$，$c_u(i) := d_i(G_u)$ 且 $c_w(i) := d_i(G_w)$。由于 T 是 H的树分解，我们有 $I_u^c \cap I_w^c = \emptyset$。因此，$E(G_v) = E(G_u) \uplus E(G_w)$，并且对于 $i \in I_u^c$，$d_i(G_v) = d_i(G_u)$；对于 $i \in I_w^c$，$d_i(G_v) = d_i(G_w)$；对于 $i \in I$，$d_i(G_v) = d_i(G_u) + d_i(G_w)$，故 $c_v = c_u + c_w$。因此

$$\begin{aligned} g(v, c_v, F_v) \;=\; & \sum_{i \in I(T_v)} f_i(d_i(G_v)) \\ =\; & \sum_{i \in I_u^c} f_i(d_i(G_u)) + \sum_{i \in I} f_i(c_v(i)) + \sum_{i \in I_w^c} f_i(d_i(G_w)) \;=\; \sum_{i \in I} f_i(c_v(i)) \\ +\; & \left(\sum_{i \in I(T_u)} f_i(d_i(G_u)) - \sum_{i \in I} f_i(c_u(i))\right) + \left(\sum_{i \in I(T_w)} f_i(d_i(G_w)) - \sum_{i \in I} f_i(c_w(i))\right) \\ =\; & \sum_{i \in I(T_u)} f_i(d_i(G_u)) + \sum_{i \in I(T_w)} f_i(d_i(G_w)) + \sum_{i \in I} (f_i(c_v(i)) - f_i(c_u(i)) - f_i(c_w(i))) \\ \geq\; & g(u, c_u, F_u) + g(w, c_w, F_w) + \sum_{i \in I_v} (f_i(c_v(i)) - f_i(c_u(i)) - f_i(c_w(i))) \\ \geq\; & h(v, c_v, F_v)\ . \end{aligned}$$

反过来，考虑一个划分 $F_v = F_u \uplus F_w$ 和一个分解 $c_v = c_u + c_w$，它们达到

$$g(u, c_u, F_u) + g(w, c_w, F_w) + \sum_{i \in I_v} (f_i(c_v(i)) - f_i(c_u(i)) - f_i(c_w(i))) \;=\; h(v, c_v, F_v)\ .$$

若 $h(v, c_v, F_v) = \infty$则 $g(v, c_v, F_v) \leq h(v, c_v, F_v)$。否则，$g(u, c_u, F_u)$ 和 $g(w, c_w, F_w)$ 均为有限值。选取 $G_u \subseteq H[I(T_u)]$，使得 $E(G_u[I]) = F_u$ 且 $d_i(G_u) = c_u(i)$，以便 $i \in I$ 达到 $\sum_{i \in I(T_u)} f_i(d_i(G_u)) = g(u, c_u, F_u)$；再选取 $G_w \subseteq H[I(T_w)]$，使得 $E(G_w[I]) = F_u$ 且

$d_i(G_w) = c_w(i)$，以达到 $i \in I \sum_{i \in I(T_w)} f_i(d_i(G_w)) = g(w, c_w, F_w)$。令 $G_v \subseteq H[I(T_v)]$ 满足 $E(G_v) = E(G_u) \uplus E(G_w)$。则 $E(G_v[I]) = F_v$ 且 $d_i(G_v) = c_v(i)$，对 $i \in I$ 而言，并且

$$\begin{aligned} g(v, c_v, F_v) &\leq \sum_{i \in I(T_v)} f_i(d_i(G_v)) = \sum_{i \in I(T_u)} f_i(d_i(G_u)) + \sum_{i \in I(T_w)} f_i(d_i(G_w)) \\ &+ \sum_{i \in I} (f_i(c_v(i)) - f_i(c_u(i)) - f_i(c_w(i))) \\ &= g(u, c_u, F_u) + g(w, c_w, F_w) + \sum_{i \in I} (f_i(c_v(i)) - f_i(c_u(i)) - f_i(c_w(i))) \\ &= h(v, c_v, F_v) \ . \end{aligned}$$

我们需要针对 $n^{|I_v|} = O(n^{k+1})$ 的 c_v 值以及 $O(2(\binom{I_v}{2})) = 2^{O(k^2)}$ 个可能的 F_v，计算 $g(v, c_v, F_v) = h(v, c_v, F_v)$。每次计算都需要在所有 $2^{O(k^2)}$ 划分 $F_v = F_u \uplus F_w$ 和 $O(n^{k+1})$ 分解 $c_v = c_u + c_w$ 中取最小值。因此，所有计算都可以在多项式时间内完成。

致谢

S. Onn 获得了以色列科学基金会的资助以及德雷斯纳讲席的支持。

参考文献

[1] Apollonio, N., Seb˝o, A.: 图中指定大小的最小凸因子。SIAM 离散数学杂志 23:1297–1310 (2009)[2] Cornu´ejols, G.: 图的一般因子。组合理论杂志 B 辑 45:185–198 (1988)[3] Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: 参数化算法。施普林格 (2015)[4] Deza, A., Levin, A., Meesum, S.M., Onn, S.: 度序列上的优化。SIAM 离散数学杂志 32:2067–2079 (2018)[5] Deza, G., Onn, S.: 图的度序列上的优化。离散应用数学 296:2–8 (2021)[6] Kouteck´y, M., Onn, S.: 一致单调行和优化。离散应用数学 298:165–170 (2021)[7] Lov´asz, L.: 具有指定度数的子图。组合理论杂志 8:391– 416 (1970)