arXiv:2304.01729v1 [math.CO] 4 Apr 2023

使用整数规划构造极值无三角形图

Ali Erdem Banak, Tınaz Ekim,Z. Caner
Ta¸skınBo˘gazi¸ci大学，工业工程系

2023年4月5日

摘要

已确定匹配数为 m 且最大度为d 的图的最大边数 [1] 和 [2], ，并给出了一些极值图。随后，一个新问题随之出现：禁止这些极值图中出现的某些子图，会如何影响最大边数？在 [3], 中，针对 $d \geq m$，以及 $d < m$ 且 $Z(d) \leq m < 2d$ 或 $d \leq 6$的情形，在无三角形图中求解了该问题，其中 $Z(d)$ 约为$5d/4$。作者推导出了无三角形极值图的结构性质，这使我们能够专注于构造小型极值分量，再将它们组合成极值图。基于这些发现，本文提出了一个用于构造极值图的整数规划模型。由于该模型具有高度对称性，我们采用自行实现的轨道分支来消除对称性。我们还对整数规划模型进行了实现，以迭代方式限制可行域。结合这两种方法，我们将解的规模扩展至 $d \leq 10$ ，而非 $d \leq 6$ ，针对 $m > d$。我们的结果支持了 [3]中提出的关于所有极值无三角形图边数的公式。

关键词： 极值图；因子临界；整数规划；轨道分支。

1 引言

极值图论研究在某些局部约束下，图参数（通常是边数或顶点数）能够达到的最大值或最小值。此类最大化或最小化问题的最优解称为极值图。图兰图是极值图论中最著名的结果之一 [4]。在限制最大匹配的大小和最大度的条件下，使图的边数最大化这一极值问题，最初由 Erd¨os 和 Rado [5] 在更一般的背景下提出。Chv´atal 和 Hanson [1] 最初运用一些优化技术和 Berge匹配公式回答了这个问题。后来，Balachandran 和 Khare [2] 为同一问题给出了构造性证明。作者描述了具有三种类型分量的一些极值图，即星图、完全图和近完全图（其中包含 C_4，也就是长度为 4 的诱导圈）。这一构造为该问题的新变体铺平了道路：如果排除 [2]中所述的一种分量，最大边数会受到怎样的影响？据此，可以分别考虑对图施加无爪、无三角形或 C_4-free 限制时的同一极值问题：无爪图排除了最小的星图；无三角形图排除了最小的完全图；而 C_4-free 则排除了近完全图。

Dibek、Ekim 和 Heggernes 研究了无爪图，并通过指出最大边数达到一般图的最大边数以及严格低于一般情形的情况，结束了相关讨论 [6]。Blair、Heggernes、Lima 和 Lokshtanov 研究了弦图；弦图不包含 C_4，因此也是近完全图。他们表明，弦图的最大边数与一般图相同 [7]。类似地，Maland [8] 回答了关于二分图、分裂图、分裂图的不交并以及单位区间图的问题。

Ahanjideh、Ekim 和 Yıldız 在 [3] 中也从相同的角度研究了无三角形图。在他们的研究中，作者对极值问题给出了一些部分答案，因此该问题

尚未完全解决。作者为所有 $d \geq m$ 构造了极值图。对于满足 $d < m$ 的情形，他们解决了 $d \leq 6$ 或 $Z(d) \leq m < 2d$ 的情况，其中 $Z(d)$ 约为$5d/4$。因此，在 [3] 中，满足 $d \geq 7$ 且 $m \geq 2d$ 或 $d < m < Z(d)$ 的情形仍未解决。尽管如此，作者还是推导出了一些有助于理解这些未解决情形的结构性结果：对于所有自然数 $d \geq 2$ 和 m，都存在一个极值图，其分量要么是星图，要么是最大度为 d 且匹配数介于d 和 $Z(d)$ 之间的因子临界无三角形极值图。该结果表明，要为 $2d \leq m$ 构造极值图，只需找到极值分量，即匹配数介于 d 和 $Z(d)$ 之间的极值图，并确定每种分量各取多少个以及应取多少个星图。随后，这个问题被表述为一个背包公式，其中总容量为 m，各项容量为每个极值分量的匹配数，效用则为每个极值分量的最大边数。

基于上述观察，本文聚焦于计算极值分量的最大边数，并提出用于求解这一问题的整数规划方法。据我们所知，利用整数规划公式构造极值图还是较新的做法。少数例外包括近期一篇通过改进已知下界研究拉姆齐数的论文 [9] ，以及构造具有给定顶点数和/或度、并能优化某些不变量的（小规模）极值化学图 [10], [11]。第2节介绍相关符号，并概述一般图和无三角形图的已知结果。我们还给出 [3] 中的背包公式，用于计算 $d \geq 7$ 和 $m \geq 2d$的最大边数。我们提出猜想2.1 和猜想2.2，分别给出极值分量和所有无三角形极值图的最大边数公式。具体而言，我们展示 [3], 中的结果，据此可将研究范围限定为构造满足 $d > 6$和 $d < m < Z(d)$的极值因子临界无三角形分量。第3节阐述我们通过四种不同的精确方法构造极值分量的方法论。我们的基本公式利用 [3] 提供的结构信息，在不将匹配数显式纳入模型的情况下对其加以约束。鉴于基本公式具有高度对称性，我们提出多种消除对称性的方法。具体而言，我们采用轨道分支 [12], ，该方法识别等价变量，并在分支中为多个变量赋值，以减少分支割平面算法中的对称性。接下来，迭代方法在每次迭代中仅考虑可行域的一部分，从而求解基本公式。在此方法中，我们将所有顶点的度设为 d ，并检查问题是否可行。若不可行，则逐步检查是否存在边数达到下一个可能上界的图。最后，我们结合这两种方法，在迭代方法中应用轨道分支。第4节介绍我们的实现，其中使用 CPLEX 20.1.0 求解整数规划，并使用 nauty 计算轨道。

在第5节中，我们给出计算结果。我们表明，在使用默认分支策略时，轨道分支和迭代方法都优于基本公式；此外，将二者结合能够得到最佳结果。我们的方法使我们能够构造出 $d = 7$、8、9、10 和 $d < m < Z(d)$ 的所有极值分量。随后，我们将这些极值分量的最大边数作为 [3] 中提出的背包公式的参数。即使对于 $m = 2000$，我们也能在 0.2 秒内求解 $d = 7$、8、9、10 的背包公式。因此，对于所有可能的 m 值，我们都找到了 $d = 7$、8、9、10 的边极值图的边数。借助 $d < m < Z(d)$，我们还能够构造出 $d = 11$、12 和 13 的一些（但不是全部）极值分量。我们的研究结果支持 Ahanjideh、Ekim 和 Yıldız 在 [3] 中提出并在第2节中解释的猜想。特别是，现在我们有更有力的证据表明， [3] 中给出的公式（见定理 2.2 和猜想 2.2）对于所有自然数 $d \geq 2$ 和 m，给出了最大度至多为 d 且匹配数至多为 m 的无三角形图的最大边数。然而，形式证明仍有待未来研究。

2 符号与预备知识

在本文中，无向图表示为 $G = (V(G), E(G))$，其中 $V(G)$ 是图 G 的顶点集， $E(G)$ 是边集。顶点 v 的度 记作 $d(v)$，表示与 v 相邻的顶点数。图 G 的最大度 是 G 中顶点的最大度，记作 $\Delta(G)$。图 G 的匹配 定义为一组彼此没有公共顶点的边。图 G 的最大匹配的大小记作 $\nu(G)$。

最大度不超过 d 且匹配数不超过 m 的一般图的最大边数记作 $f_{GEN}(d,m)$。可以看出，如果我们不限制 $\Delta(G)$ 或 $\nu(G)$ 中的任何一个，图的边数就可能无限多；例如，一个中心顶点可以有无穷多个邻居，而无穷多条独立边也可以构成这样的例子。因此，我们关注 $\nu(G)$ 不超过 m 且 $\Delta(G)$ 不超过 d 的图中的边数。

无三角形图定义为不含三顶点环的图。换言之，非空无三角形图中的最大完全子图的规模为二。令 Δ 表示所有无三角形图组成的类。于是，最大度不超过 d 且最大匹配大小不超过 m 的无三角形图的最大边数记为 $f_{\Delta}(d,\ m)$。

ℓ-星图 $K_{1,\ \ell}$ 是一种二分图，其中一侧有一个（中心）顶点，该顶点与另一侧的 ℓ个顶点相邻。如果一个图的所有顶点度数都相同，均为 d，则称其为 d-正则图。如果只有一个顶点的度为 $d-1$，其余顶点的度均为 d，则称其为近似 d-正则图。若从图 G 中移除任意一个顶点后，剩余图都存在完美匹配，即存在一个覆盖所有顶点的匹配，则称该图为因子临界 图。

以下定理 [2] 给出了 $f_{GEN}(d,m)$ 的公式，适用于所有 d和 m，并给出了极值图的构造。

定理 2.1。 *[2]* 对于满足 $\Delta(G)\ \le d$ 和 $\nu(G)\ \le m$的一般图，极值图的最大边数为

$$f_{GEN}(d,m)=dm+\lfloor d/2\rfloor\lfloor m/\lceil d/2\rceil\rfloor.$$

此外，具有 $f_{GEN}(d,m)$ 条边的极值图可以通过取 r 个 d-星图与 q 个

$$\begin{cases}K_{d+1} & if\ d{+}1\ is\ odd\\ K_{d+1}^{'} & if\ d{+}1\ is\ even,\end{cases}$$

其中， q是满足 $m=q\lceil d/2\rceil+r$和 $r\ge 0$的最大整数； $K_{d+1}^{'}$是这样得到的图：从具有 $d+1$ 个顶点的完全图 K_{d+1} 中删除一个完美匹配，添加一个新顶点v，并使 v 与其余顶点中的 d 个相邻。

Ahan-jideh、Ekim 和 Yıldız 在 [3] 中研究了限制为无三角形图的同一极值问题。作者解决了 $d\ge m$的所有情形，以及满足$d\le 6$ 或 $Z(d)\ \le m<2d$ 的 $d<m$ 情形，其中 $Z(d)$ 定义如下。

定义 2.1。 *[3]* 对于任意 $d\ge 2$，令 $Z(d)$ 为满足以下条件的最小自然数：存在一个d-正则（若 d 为偶数）或近似 d-正则（若 d 为奇数）的无三角形因子临界图 G，且$\nu(G)\ =n$。

由于 $Z(d)$ 在描述无三角形极值图时起着关键作用，作者首先介绍一种图构造，以证明 $Z(d)$ 的存在性，然后进一步研究$Z(d)$ 的取值。

引理 2.1。 *[3]* 我们有 $Z(d)\ =d$，其中 $d\in\{2,3\}$；并且 $Z(d)\ =d+1$，其中 $d\in\{4,5\}$。

引理2.2。 *[3]* 对于 $d\ \ge\ 2$，若 d 为偶数，则有 $Z(d)\ \ =\ \lfloor 5d/4\rfloor$；若 d 为奇数，则有$\lfloor 5(d-1)/4\rfloor\le Z(d)\ \le\lceil 5(d+1)/4\rceil$。

[3] 中的主要结果给出了所有已求解情形下 $f_{\Delta}(d,m)$ 的以下公式，并描述了极值图。为简洁起见，这里省略该描述。

定理 2.2. *[3]* 设 d 和 m 为满足 $d\ \ge$ 2的自然数，并设 k 和 r 为满足以下条件的非负整数： $m=kZ(d)+r$，且 $0\le r<Z(d)$。那么，对于所有满足 $d\ge m$的情形，以及满足$d<m$ 且 $d\le 6$ 或 $Z(d)\ \le m<2d$的情形，我们有

$$f_{\Delta}(d,m)=\begin{cases}dm+k\lfloor d/2\rfloor, & if\ r<d\\ dm+k\lfloor d/2\rfloor+r-d+1, & if\ r\ge d.\end{cases}$$

由定理 2.2 可知，剩余的未解决情形为 $d \geq 7$，其中 $m \geq 2d$ 或$d < m < Z(d)$。除了关于 $f_\Delta(d,\ m$的公式之外，[3] 中的一项重要贡献是以下结果，它表明存在一个具有特殊结构的无三角形极值图。这种结构性质由 。2 和引理 4。4）中的若干结果（即推论 2 [3],所表达，对构建我们的整数规划模型至关重要。

引理 2.3. *[3]* 设 d 和 m 为满足 $d \geq 2$的自然数，且 G 是一个边极值图，其中同构于 d-星图的连通分支数量最多。那么，对于 G的每个连通分支 H，以下条件之一成立：

(i) H 是一个 d 星图。*(ii)* H 是因子临界 r-*critical with* $|E(H)| = f_\Delta(d, \nu(H))$ *and* $|V(H)| = 2\nu(H)+1$ *where* $d \leq \nu(H) \leq Z(d)$.

现在，考虑关于 $m \geq 2d$ 和 $d \geq 7$的未解决情形。引理 2.3 指出，存在一个无三角形边极值图，其非 d-星图分量是边极值、因子临界且无三角形的图，并且其匹配数介于 d 和 Z之间（含端点）（d）。本文将后者称为极值分量。在 [3] 中，我们利用这一信息，将关于 $m \geq 2d$ 和$d \geq 7$ 的未解决情形表述为一个背包问题，其中 $d \leq i \leq Z$（d）的效用参数为 f_Δ（d, i），这些参数尚待计算。设 x_i 为 G 中匹配数为 i 的极值分量数量。那么，d 和 m 对应的极值图可按如下方式构造。对于 $d \leq i \leq Z$（d），取 x_i 个匹配数为 i的极值分量；它们对边总数 f_Δ（d, m）的贡献是方程（1）左端的第二个求和项，并使图的匹配数增加 $\sum_{i=d}^{Z(d)} ix_i$。再使用 d-星图将匹配数补足至 m；每个星图使 f_Δ（d, m）增加 d。这一部分由方程（1）左端的第一项表示。于是得到以下结论：

$$f_\Delta(d,m) = d(m - \sum_{i=d}^{Z(d)} ix_i) + \sum_{i=d}^{Z(d)} f_\Delta(d,i)x_i = dm + \sum_{i=d}^{Z(d)} (f_\Delta(d,i) - di)x_i. \tag{1}$$

基于这些观察，Ahanjideh、Ekim 和 Yıldız 提出了以下有界背包公式，其中物品 i 的效用为（$f_\Delta(d,$ $i) - di$），体积为 i [3]。

$$\text{(Knapsack Formulation)} \quad \max dm + \sum_{i=d}^{Z(d)} (f_\Delta(d,i) - di)x_i \tag{2}$$

$$\text{s.t.} \quad \sum_{i=d}^{Z(d)} ix_i \leq m \tag{3}$$

$$x_i \geq 0, \quad x_i \in \mathbb{Z} \tag{4}$$

采用这一公式，如果已知固定 d下极值分量的边数，就可以找到所有 m 的边极值图。因此，本文着重于寻找极值分量，即对于具有 $d > 6$ 且$d < \nu(H) < Z(d)$ 的 $d < m$，寻找边极值图 H。注意，匹配数为 d 且 $Z(d)$ 的极值分量可由定理 2.2 得出；我们有 $f_\Delta(d, d) = d^2 + 1$ 以及 $f_\Delta(d, Z(d))$ $= dZ(d) + \lfloor \frac{d}{2} \rfloor$。然而，当 d 为奇数时，$Z(d)$ 尚不明确。

尽管 [3] 中的作者将进一步开发解决这一背包公式的方法留待以后，但他们仍对其解提供了一些见解。具体而言，他们猜想其未知参数，即 $f_\Delta(d,\ i)$ 对于 $7 \leq d < i < Z(d)$ 遵循定理 2.2给出的公式，在此情形下可简化如下：

猜想 2.1. *[3]* 对于 $7 \leq d < i < Z(d)$，有 $f_\Delta(d,\ i) = di + i - d + 1$。

作者还证明，如果猜想 2.1 成立，那么背包公式存在一个特殊的最优解，其中匹配数为 $Z(d)$ 的极值分量数量尽可能多，且至多有一个匹配数较小的极值分量。

命题 2.1。 *[3]* 如果猜想 *2.1* 成立，那么对于 $7 \le d < m < Z(d)$，背包公式存在一个最优解，使得 $\sum_{i=d}^{Z(d)-1} x_i \le 1$。换言之，$x_{Z(d)}$ 达到最大值，且至多有另一个 x_i 取值为 *1*（其余均为零）。

鉴于命题 2.1中对极值图的描述与定理2.2 （我们在论文中略去了该定理）中的描述相似，作者还猜想定理 2.2中的公式普遍成立（包括未解决情形）。

猜想 2.2。 *[3]* 对于所有自然数 $d \ge 2$ 和 m，令 $m = kZ(d) + r$。则有

$$f_\Delta(d,m) = \begin{cases} dm + k\lfloor d/2 \rfloor, & if\ r < d \\ dm + k\lfloor d/2 \rfloor + r - d + 1, & if\ r \ge d. \end{cases}$$

接下来，我们构建整数规划模型，以构造极值分量。我们的方法能够为 $d = 7,8,9,10$ 构造所有极值分量，从而解决这些 d 值（以及任意 m）对应的背包公式。我们的所有发现都支持猜想 2.1、命题 2.1和猜想 2.2，从而进一步强化了它们。此外，我们还得到了一些关于 $Z(d)$ 的新取值，即 $Z(7) = 9, Z(9) = 13, Z(11) = 15$，以及 $Z(13) = 17$。

3 方法论

我们将构造匹配数至多为 m 且度数至多为 d 的边数极值无三角形图表述为一个整数规划问题。令 V 为顶点集，并令 x_{ij}为定义在 $i \ne j$ 上的二元变量：若 i、$j \in V$之间存在一条边，则取值为 1，否则取值为 0。我们研究无向图，因此只使用 x_{ij} 和 x_{ji} 变量中的一个。为简化模型中的记号，应将 x_{ij} 中的 ij 视为无序集合，而边仅对 $i > j$定义。

$$\text{(Basic Formulation)} \quad \max \sum_{i,j \in V} x_{ij} \tag{5}$$

$$\text{s.t.} \quad x_{ij} + x_{jk} + x_{ik} \le 2 \quad \forall i,j,k \in V \tag{6}$$

$$\sum_{j \in V} x_{ij} \le d \quad \forall i \in V \tag{7}$$

$$x_{ij} \in \{0,1\} \quad \forall i,j \in V \tag{8}$$

目标函数（5）使边数最大化。约束（6）确保所得图是无三角形图，约束（7）则限制最大度。我们不显式约束匹配数，而是利用引理2.3，该引理保证存在一个因子临界的边数极值无三角形图，并据此相应地固定 $|V| = 2m + 1$。显然，这将匹配数限制为 m。由于对每个 d 和 m，都存在一个具有 $2m + 1$ 个顶点且度数受 d 限制的无三角形图，因此基本公式的最优解描述了一个具有 $f_\Delta(d, m)$ 条边的极值无三角形图。基本公式得出的图可能是因子临界的，也可能不是；然而，引理 2.3 保证它有 $f_\Delta(d, m)$ 条边，因为至少存在一个因子临界的极值分量。

由于所有 x_{ij} 都可以互换，基本公式具有很强的对称性。为了解决这一问题，我们提出两种方法，并将它们结合成第三种方法。第一种方法是轨道分支，即通过分支割平面算法，在求解基本公式的过程中修改分支决策。轨道分支有助于我们在分支割平面树中消除对称解。第二种方法是对问题进行少量修改后迭代求解，我们称之为迭代方法。在第一次迭代中，所有顶点度都设为 d。如果不存在可行解，则将除一个以外的所有度都设为 d，并约束剩下的一个小于 d，从而降低解的上界。只要不存在可行解，每次迭代就再约束一个顶点的度小于 d。这种方法实质上限制了问题的可行域。

3.1 轨道分支

我们在求解基本公式时使用轨道分支来修改分支决策。其思路是在分支割平面树的每个节点上，同时对同一对称群中的多个变量进行分支，而不是每次只对单个变量进行分支。

在用形式化术语解释对称群和变量的等价性之前，我们先来考察一下基本公式的结构。在该公式中，所有 x_{ij} 都可以互换，因为没有任何因素将一条边与其他边区分开来。假设我们为每个变量 $x_{12} = 1$、 $x_{13} = 1$ 等分别创建一个分支，同时创建一个所有变量均为 0 的分支，以覆盖可行域。即使各分支之间存在交集，这仍然是一种合理的分支策略，因为整个可行域都得到了覆盖。然而，我们可以观察到，每个满足$x_{1j} = 1, j \in V \setminus \{1\}$ 的子节点都会产生相同的子问题。因此，我们不必在 $|V|$ 个节点上进行分支，而只需选择一个代表节点（例如 $x_{12} = 1$），再创建另一个带有 $x_{12} = x_{13} = ... = 0$ 的分支，即可仅在两个节点上进行分支。这就是轨道分支背后的主要思想。在根节点处，很容易看出哪些变量是等价的；但在第一次分支之后，情况会变得更加复杂。现在，x_{13} 与 x_{34} 并不等价，因为由于此时变量 $x_{12} = 1$ 已经被固定，它们会产生不同的子问题。因此，我们需要对对称群进行形式化定义，以识别出这些变量。为此，我们考虑以下问题：

$$\max_{x \in \{0,1\}^n} \{c^T x | Ax \leq b\}. \tag{9}$$

矩阵 A 的对称群定义为使 A 保持不变的置换集合 [13]。这意味着存在一种对变量（列）和约束（行）的置换，依次应用于 A 后会得到相同的矩阵 A [12]。如果列的置换也会生成相同的 c，且行的置换也会生成相同的 b，那么新问题就与原问题等价。整数规划的对称群可以通过图自同构来确定。整数规划会转换为二分图 B（V， M， E），其中 V表示变量， M 表示约束。当且仅当 $A_{ij} \neq 0$时，v_i与 m_j之间存在一条边；也就是说，变量 i 以非零系数出现在约束 j 中。为体现 c、 b和 A 值之间的差异，使用颜色编码标示 B [14]中可互换的顶点。一般而言，每种不同的成本系数 c_i 值都会使用不同的颜色来标记顶点$v_i \in V$。同样，顶点 $m_j \in M$ 根据 b_j 和约束的不等式类型着色。在这种设置下，改变列的顺序相当于改变 V中顶点的标签，而改变行的顺序相当于改变 M中顶点的标签。请注意，应避免将 V中的顶点与 M中的顶点互换，因此它们也被赋予不同的颜色。于是，整数规划的对称群与图 B（V， M， E）的自同构群相同。如果存在一个自同构将 x_i 映射到 $x_{i'}$，则它们被视为等价，并属于同一个轨道。轨道由等价变量构成，并用于轨道分支中的分支决策。

计算自同构群会将顶点划分为轨道，即由等价顶点组成的集合。轨道分为两类：变量轨道和约束轨道。在我们的轨道分支方法中，我们使用变量轨道，并优先选择元素数量最多的轨道进行分支。

在图 2中给出了下面模型的轨道分支示例。

$$\begin{aligned}
\max \quad & x + y + z + t && (10)\\
\text{s.t.} \quad & x + y \geq 1 && (11)\\
& x + y \geq 1 && (12)\\
& x + y \geq 1 && (13)\\
& x, y, z, t \in \{0, 1\} && (14)
\end{aligned}$$

在图 1 中模型 (10)-(14) 的图表示里，顶点 x, y, z, t 表示变量，顶点 $c1, c2, c3$ 表示约束。将图 1 左侧的图称为原始二分图。由此可见，交换顶点标签 y, z 以及 $c1, c2$ 会生成相同的关联矩阵。这等价于说，右侧的图是通过以不同方式重新标记左侧图的顶点得到的，因此图 1 中的两幅图同构。此过程表示为 $(y, z)(c1, c2)$，它是该二分图的一个自同构群。

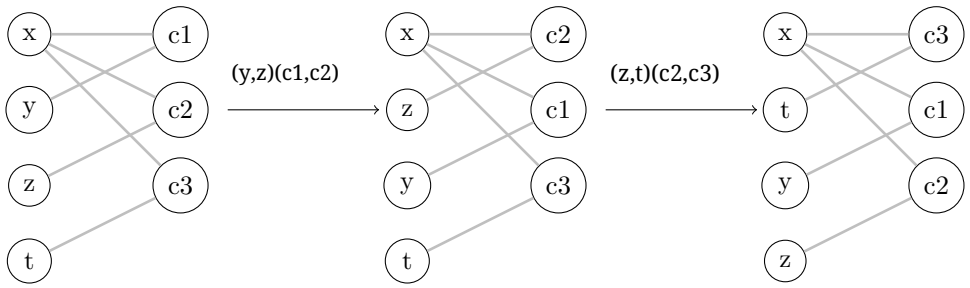

图 1：与模型 (10)-(14)对应的二分图的一个自同构

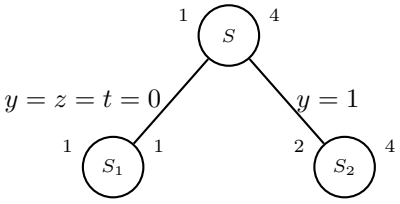

图2：使用轨道分支的分支割平面树

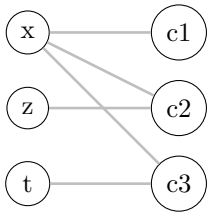

图3： S_2 的二分图

另一个自同构群是（z， t）（$c2$， $c3$）。连续应用自同构群也会得到一个自同构群。因此，我们可以看到 y、 z、 t 可以互换，并且属于同一个轨道。因此，轨道分支建议采用图 2所示的分支方式，其中每个节点的上界和下界分别写在该节点的右侧和左侧。在分支割平面树深度为 1 的每个节点中，都会找到一个最优解，并根据新图重新计算轨道。作为示例，表示子问题 S_2 的二分图如图 3所示。令 F_0^a 和 F_1^a分别表示在分支割平面的节点 a 中设为 0 和 1 的变量。那么，轨道分支算法可概括如下，见算法1：

算法1轨道分支

步骤 1： 给定一个分支割节点 $a = (F_0^a,\ F_1^a)$，计算节点 a处的轨道集合 O_a

步骤 2： 任意选择一个轨道 $O \in O_a$

步骤 3： 任意选择一个变量 $v_i \in O$，返回新节点 $l = (F_0^a \cup \{O\},\ F_1^a)$、 $r = (F_0^a,\ F_1^a \cup \{v_i\})$

注意到在基本公式中，所有 c 值均为 1；因此我们不需要对变量顶点着色。

3.2 迭代方法

由于顶点的最大度和顶点数均有限，基本公式目标函数值的预计算上界 为 $\lfloor (2m+1) * d/2 \rfloor$。只有当所有顶点的度都达到最大度 d时，才能达到此上界。如果其中恰有一个顶点的度为 $d-1$，而其余所有顶点的度均为 d，那么向下取整前的表达式值会减少 $1/2$，因此上界可能降低。我们可以利用这一信息，迭代降低基本公式的上界。例如，令 $d = 8$ 和 $m = 9$；因此共有 $2 * 9 + 1 = 19$个顶点。我们得到 $(8 * 19)/2 = 76$ 作为上界。可以将所有度设为 8，并检查是否存在可行解。如果存在可行解，它就是基本公式的最优解。如果该问题不可行，则可知该问题的上界为 75，且至少有一个顶点的度不超过 $d-1$。

我们的迭代方法首先将所有度固定为度数界 d，求解由此得到的整数规划，然后逐一降低上界。每次迭代都在原始可行域的一部分上进行，并迭代降低上界。其思路是通过在受限可行域中求解公式，获得基本公式的最优解。我们可以

考虑这一方法的不同公式：针对每个目标值枚举所有可能的度组合，或者简单地通过将部分顶点的度设为小于 d 、其余顶点的度设为 d 来降低上界。我们采用后一种公式，因为枚举所有可能的度会导致指数级迭代次数。

我们区分两个顶点集合；令 V_d 为度等于 d 的顶点集合， V_{d-1} 为度至多为 $d-1$的顶点集合。该公式可表示为：

$$\text{(Iterative Formulation)} \quad \max \sum_{i,j \in V} x_{ij} \tag{15}$$

$$\text{s.t.} \quad x_{ij} + x_{jk} + x_{ik} \leq 2 \quad \forall i,j,k \in V \tag{16}$$

$$\sum_{j \in V} x_{ij} = d \quad \forall i \in V_d, \tag{17}$$

$$\sum_{j \in V} x_{ij} \leq d-1 \quad \forall i \in V_{d-1}, \tag{18}$$

$$x_{ij} \in \{0,1\} \quad \forall i,j \in V \tag{19}$$

需要注意的是，含有 $|V_{d-1}| = 1$ 的迭代公式的最优解，不一定是基本公式的最优解。事实上，所有顶点的度均为 d 、但有一个顶点的度为 $d-4$ 的图，其边数少于具有 $|V(G)|-2$ 个度为d的顶点，以及两个度为 $d-1$的顶点的图。因此，含有 $|V_{d-1}| = 1$ 的迭代公式的最优解给出了一个下界，之后可以通过在V_{d-1}中添加更多顶点来进一步改进该下界。在每次迭代中，都会更新集合 V_d 和 V_{d-1} ，并求解迭代公式。算法 2 中总结的迭代方法通过不断更新上界，依次求解迭代公式，直至上界等于由最佳可行解得到的下界。下文令 $N = 2m+1$ 表示顶点数。

算法2迭代方法

输入： $V_{d-1} = \emptyset$， $V_d = V$
$LB = 0$, $UB = \lfloor ((Nd)/2 \rfloor$
当 $UB > LB$ **时执行**
　求解迭代公式 D ，使用 V_d 和 V_{d-1}。
　如果 D 可行 **则**
　　$LB = \max(z^*, LB)$，其中 z^* 是 D 的最优目标函数值
　结束 if
　选取 $u \in V_d$
　$V_d = V_d \setminus \{u\}$
　$V_{d-1} = V_{d-1} \cup \{u\}$
　$UB = \max(\lfloor ((Nd)/2 - 0.5|V_{d-1}| \rfloor, LB)$
结束 while
返回 UB 以及迭代公式的最优解，该解给出了最佳的 LB。

我们的迭代方法并不主要着眼于减少对称性，而是着眼于缩小可行域。在第一次迭代中，$V_{d-1} = \emptyset$ 中的所有变量都等价。不过，每次迭代中，某个顶点的一个度约束都会发生变化，从而减少对称性。为了进一步减少对称性，我们可以按字典序排列 V_{d-1} 中的顶点，使其度数非递增。

3.3 带轨道分支的迭代方法

轨道分支利用对称性进行分支决策，而迭代方法在保留大部分对称性的同时缩小可行域。此外，轨道分支会干扰求解器的默认分支割决策，而迭代方法则会修改问题公式。因此，可以联合使用这两种方法：执行与迭代方法中相同的迭代，但使用轨道分支进行分支决策。迭代公式的对称性弱于基本公式，因为公式（17）是等式。

4 计算实验

计算是在配备 32 GB 内存的 Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz 上，使用 CPLEX 20.1.0 和 10 个线程完成的。我们使用 CPLEX 20.1.0 和 C++ 实现这些方法。每次运行的时间限制为 1800 秒。在以下展示计算结果的表格中，“PreUB”表示预计算上界，“LB”表示 CPLEX 找到的下界，“UB”表示 CPLEX 找到的上界，“Gap”表示相对最优性间隙，“Time”表示执行时间，而“Node”表示分支割平面树中的节点数。在表 1、2 和 3 中，与最优性对应的行以深色背景标记。我们测试了具有 $7 \leq d \leq 13$ 和 $d < m \leq Z(d)$ 的实例。当 d 为奇数时，$Z(d)$ 事先未知。然而，根据 $Z(d)$ 的定义，匹配数为 $m = Z(d)$ 的图几乎是 d-正则图。因此，我们根据所得边极值图中的度数确定 $Z(d)$。更准确地说，$Z(d)$ 是匹配数 m 的值，此时 LB 等于预计算 UB，且所得极值图为因子临界图；我们通过计算对其进行检查。当 d 为偶数时，根据引理2.2，有 $Z(d) = \lfloor 5d/4 \rfloor$ 。这意味着 $Z(8) = 10$、$Z(10) = 12$ 和 $Z(12) = 15$。随后，由定理 2.2 可得 $f_{\Delta}(8, 10) = 84$、$f_{\Delta}(10, 12) = 125$ 和 $f_{\Delta}(12, 15) = 186$。通过计算，我们得到了新的 d 值 7、9、11 和 13 所对应的 $Z(d)$ 值；即 $Z(7) = 9$、$Z(9) = 12$、$Z(11) = 15$、$Z(13) = 17$。总体而言，如果检查已求解的实例，边数与 Ahanjideh、Ekim 和 Yıldız [3]中给出的猜想 2.2 一致。

为求解基本公式（5）-（8），我们先使用默认参数运行 CPLEX，再使用将对称性破缺参数设为最激进设置的 CPLEX（$IloC\text{-}plex::Param::Preprocessing::Symmetry = 5$）。结果见表1。两种方法都只将 7 个实例求解至最优。总体而言，启用对称性破缺的 CPLEX 表现略好，因为其平均时间更短。两种方法的间隙相同。请注意，上界与通过最大度约束可达到的预计算上界相同。例如，当 d 为 7 且 m 为 8 时，共有 17 个顶点，因为我们可以假设极值图是因子临界的。由于 d 为奇数，最大边数可由一个近似 d正则图达到（所有度数之和应为偶数）：16 个顶点的度为 7，另一个顶点的度为 6，因此总边数为 59。这解释了为什么只有 $m = Z(d)$ 对应的实例已求解。当 $m = Z(d)$ 时，所得图为 d正则图或近似 d正则图，其边数与我们讨论的上界相同。通过检查结果可以得出结论：我们的初始方法不足以降低 CPLEX 得出的上界，但有望找到可行解。这可能是由对称性导致的。在下一项实验中，我们测试使用基本公式进行轨道分支（第3.1节）的效果。

我们使用 CPLEX 的回调机制实现了轨道分支。为了寻找轨道，我们使用了 nauty 2.7r3，这是 McKay 和 Piperno 开发的用于计算图的自同构群的软件库 [15]。在每个分支割节点上，我们从 CPLEX 获取取值为 0 和 1 的变量，并相应地更新约束。然后，我们利用绑定约束构建更新后的二分图。随后将生成的图（及其着色）传递给 nauty。为了计算轨道，nauty 利用图同构问题计算自同构群，而图同构问题是一个 $GI-complete$ 问题 [16]。这意味着，计算轨道的问题既未被证明是 NP 完全问题，也未被证明是多项式时间可解问题。对于大型实例，运行时间最高可达 150 秒；而对于小型实例，运行时间约为 0。1 秒。这一点可见图 4，其中实例 7 8 表示顶点数为 $d = 7$ 且 $m = 8$的实例。由于分支割树中可能有数万个节点，即使计算轨道只需一秒，也会导致 CPU 时间过长。因此，我们分析分支割树中各深度的平均轨道大小。随着分支割树深度增加，轨道往往会变小，如图 5所示。例如，对于 $d = 7$ 且 $m = 8$的实例，在深度 20 之后使用轨道分支只会增加开销。因此，我们将轨道分支运行到某个特定深度。开始时，该深度被初始化为一个较大的数值。当我们找到一个轨道大小为 1 的节点时，该节点的深度便成为上限；对于更深的节点，则不再计算轨道。超过该深度后，将使用 CPLEX 的默认分支策略。分支时选择顶点数最多的轨道，因为这样可以为子节点中的更多变量设定取值。如果最大的轨道只有一个变量，则使用 CPLEX 的默认分支策略。

参数			基本公式					基本公式 +对称性破缺				
d	m	预上界	LB	UB	Gap	Time	Node	LB	UB	Gap	Time	Node
7	8	59	58	59	1.72%	1800	8412700	58	59	1.72%	1800	4657959
7	9	66	66	66	0.00%	0	0	66	66	0.00%	0	0
8	9	76	74	76	2.70%	1800	4605513	74	76	2.70%	1800	3012473
8	10	84	84	84	0.00%	0	0	84	84	0.00%	0	0
9	10	94	92	94	2.17%	1800	2952258	92	94	2.17%	1800	1836200
9	11	103	102	103	0.98%	1800	1356502	102	103	0.98%	1800	1122091
9	12	112	112	112	0.00%	2	428	112	112	0.00%	3	465
10	11	115	112	115	2.68%	1800	2049077	112	115	2.68%	1800	1274473
10	12	125	125	125	0.00%	0	0	125	125	0.00%	0	0
11	12	137	134	137	2.24%	1800	530890	134	137	2.24%	1800	809227
11	13	148	146	148	1.37%	1800	293767	146	148	1.37%	1800	311019
11	14	159	158	159	0.63%	1800	257540	158	159	0.63%	1800	257540
11	15	170	170	170	0.00%	0	0	170	170	0.00%	0	0
12	13	162	158	162	2.53%	1800	471193	158	162	2.53%	1800	392793
12	14	174	171	174	1.75%	1800	271165	171	174	1.75%	1800	218547
12	15	186	186	186	0.00%	53	6077	186	186	0.00%	707	82348
13	14	188	184	188	2.17%	1800	326937	184	188	2.17%	1800	292989
13	15	201	198	201	1.52%	1800	265385	198	201	1.52%	1800	173732
13	16	214	212	214	0.94%	1800	148400	212	214	0.94%	1800	144746
13	17	227	227	227	0.00%	1200	45158	227	227	0.00%	21	311
Avg			138.45	140	1.12%	1233	1099650	138.45	140	1.12%	1207	729346

表1：基本公式性能摘要。

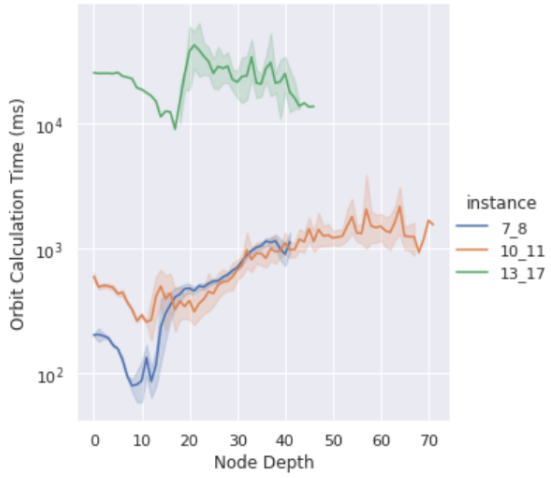

图4：nauty 随深度变化的运行时间。

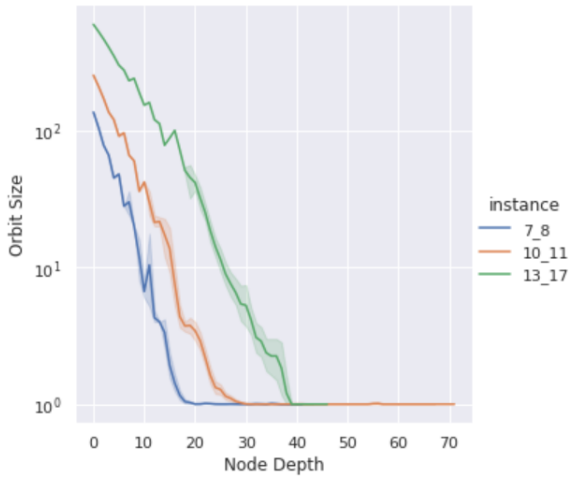

图5：不同深度下的轨道大小。

参数			轨道分支				
d	m	预上界	LB	UB	Gap	Time	Node
7	8	59	58	58	0.00%	14	57945
7	9	66	66	66	0.00%	0	0
8	9	76	74	74	0.00%	61	236040
8	10	84	84	84	0.00%	0	0
9	10	94	92	92	0.00%	558	1781858
9	11	103	102	103	0.98%	1800	2651099
9	12	112	112	112	0.00%	21	193
10	11	115	112	114	1.79%	1800	1838556
10	12	125	125	125	0.00%	0	0
11	12	137	134	137	2.24%	1800	1223806
11	13	148	146	148	1.37%	1800	1142867
11	14	159	158	159	0.63%	1800	428121
11	15	170	170	170	0.00%	0	0
12	13	162	158	162	2.53%	1800	567720
12	14	174	171	174	1.75%	1800	444420
12	15	186	186	186	0.00%	361	4769
13	14	188	184	188	2.17%	1800	278258
13	15	201	198	201	1.52%	1800	85467
13	16	214	212	214	0.94%	1800	33132
13	17	227	225	227	0.89%	1800	2756
Avg			138.4	139.7	0.98%	1041	538850

表 2：轨道分支性能总结。

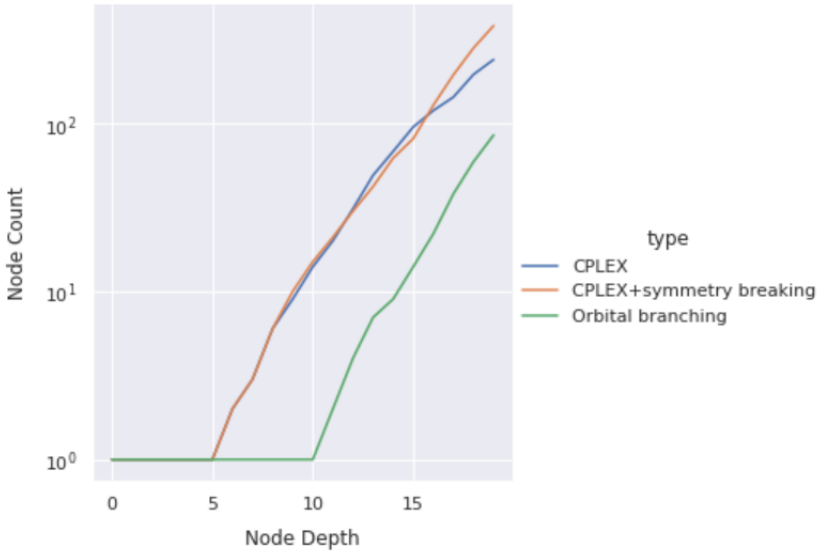

图 6：分支切割节点数与分支切割节点深度

轨道分支的结果列于表 2 中。轨道分支求解了 9 个实例。比较表 1 和2，我们发现，基本公式无法为 $m = Z(d)$ 以外的实例找到解，而轨道分支则求解了所有 $d = 7$ 和 $d = 8$ 的实例。这一差异可归因于轨道分支对初始节点的剪枝方式，如图 6 所示。对于满足 $d = 7$和 $m = 8$ 的实例，轨道分支会将除一个节点外的所有节点剪除，直至深度达到 10。这使它能够在规模小得多的可行域中运行。然而，随着问题规模增大，其性能会下降。对于 $d = 13$ 和 $m = 17$，它无法找到可证明为最优解的解，而采用激进对称性破缺的 CPLEX 则用更少的节点找到了这样的解。部分差异源于

参数			迭代式					迭代式 + 轨道分支				
d	m	预上界	LB	UB	Gap	Time	Node	LB	UB	Gap	Time	Node
7	8	59	58	58	0.00%	0	145	58	58	0.00%	0	57
7	9	66	66	66	0.00%	0	0	66	66	0.00%	0	0
8	9	76	74	74	0.00%	3	12880	74	74	0.00%	4	2794
8	10	84	84	84	0.00%	0	0	84	84	0.00%	0	0
9	10	94	92	92	0.00%	2	5149	92	92	0.00%	11	1990
9	11	103	102	102	0.00%	5	12459	102	102	0.00%	36	4170
9	12	112	112	112	0.00%	1	33	112	112	0.00%	368	229
10	11	115	112	112	0.00%	100	208001	112	112	0.00%	79	10114
10	12	125	125	125	0.00%	1	0	125	125	0.00%	2	0
11	12	137	134	134	0.00%	176	253162	134	134	0.00%	146	15106
11	13	148	146	146	0.00%	206	362834	146	146	0.00%	355	371215
11	14	159	158	159	0.63%	1800	1087900	158	159	0.63%	1800	706832
11	15	170	170	170	0.00%	2	38	170	170	0.00%	1	0
12	13	162	158	159	0.63%	1800	2264281	158	158	0.00%	1623	2020924
12	14	174	171	172	0.58%	1800	1579790	171	172	0.58%	1800	1101510
12	15	186	186	186	0.00%	0	0	186	186	0.00%	0	0
13	14	188	184	185	0.54%	1800	1495811	184	185	0.54%	1800	2294463
13	15	201	198	200	1.01%	1800	1313790	198	200	1.01%	1800	935875
13	16	214	212	213	0.47%	1800	690328	212	213	0.47%	1800	577438
13	17	227	227	227	0.00%	56	5696	227	227	0.00%	72	543
Avg			138.45	138.8	0.25%	568	464615	138.45	138.75	0.22%	585	402163

表3：迭代公式性能总结。

方法	已求解	Gap	Time	Node
基本公式	7	1.12%	1233	1099650
基本公式 + 对称性破缺	7	1.12%	1207	729346
轨道分支	9	0,98%	1041	538850
迭代方法	14	0,25%	568	464615
迭代方法 +轨道分支	15	0.22%	585	402163

表4：所有方法的性能总结。

nauty 的迭代耗时超过 50 秒。然而，对于所有方法而言，大多数实例仍未求解，降低上界的问题依然存在。

在下一项实验中，我们探究迭代方法的性能（第3.2 节和3.3节）。在迭代方法中，我们使用 CPLEX 的全局回调机制。我们不必等待一次迭代结束，而是利用先前迭代的下界信息，在必要时停止当前迭代。表3展示了迭代方法以及结合轨道分支的迭代方法的结果。迭代方法将 14 个实例求解至最优，而结合轨道分支的迭代方法则多将一个实例求解至最优。这些数字远高于使用基本公式的方法。迭代方法的运行时间略短，而组合方法使用显著更少的节点求解问题。

整体性能总结见表 4。可以看到，随着采用的方法日益复杂，已求解的实例数量也随之增加。基本公式已求解的实例数量为 7 个，但这些实例的 $m = Z(d)$ 值均为如此；因此，新方法显著提升了性能。尤其是，两种迭代方法的表现都明显优于其他方法。

值得注意的是，迭代方法不仅解决了一些支持猜想 2.1 的新实例，还为我们提供了呈现某种规律的图。这一趋势使我们能够提出一种一般构造，用于构造猜想 2.1 中对所有最大度d 和匹配数 i（其中 $7 \leq d < i < Z(d)$）所声称的极值图。对于 $0 \leq t < Z(d) - d$，我们描述如下的图$B_{d,d+t}$，其中 $i = d + t$（见图4）：取一个完全二部图 $K_{d+t,d+t}$，每侧各有 $d + t$ 个顶点。选取每侧的 $d - 1$ 个顶点（图4中标记为 F 的集合），并移除它们之间的t 个互不相交的完美匹配。注意，这是可行的，因为由引理2.2可知，$t < Z(d) - d$ 且$Z(d) \leq \lceil 5(d+1)/4 \rceil$，因此有 $t < \lceil (d+1)/4 \rceil + 1 < d$。然后，考虑每侧剩余的$t + 1$ 个顶点（图4中标记为 H 的集合），移除它们之间的所有边，并引入一个

额外顶点 v，并使其与每侧的所有 $t+1$ 顶点相邻。通过这种构造，除 v 之外的所有顶点的度均为 d。值得一提的是，对于 $t=0$，图 $B_{d,d+t}$ 对应于图 A_d 中给出的、在 且 $(,d) = d^2+1$ 条边的情况下的极值图。图 A_d，或者等价地说，$B_{d,d}$，可以看作一个 5-圈，其中两个相邻顶点被替换为大小各为 $d-1$ 的独立集，而这些集合与其他顶点（或集合）之间的边则替换为完全连接。从这个角度看，我们对图 $B_{d,d+t}$ 的构造可以视为 A_d 的推广，适用于介于 $d+1$ 和 $Z(d)-1$ 之间的匹配数。

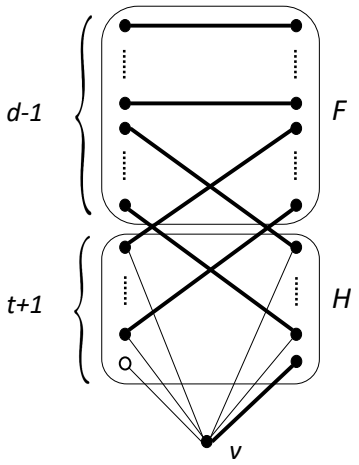

图7：图 $B_{d,\ d+t}$ 其中集合 H 诱导出一个独立集，集合 F 诱导出一个完全二部图，并从中移除了 t 个不同的完美匹配。删除顶点 v 后，得到一个二分图，该图在集合 F 和H的相对部之间具有完全连接。粗体边表示大小为 $d+t$的最大匹配。

命题4.1。 对于所有 $i=d+t$ 满足 $d<i<Z(d)$ 的情形，图 $B_{d,\ d+t}$ 具有 $di+i-d+1$ 条边，$\Delta(B_{d,\ d+t}) = d$ 以及 $\nu(B_{d,\ d+t}) = d+t=i$；因此，如果猜想 *2.1* 成立，它就是一个极值图。

证明。首先，我们注意到 $t<Z(d)-d$，因此图 $B_{d,d+t}$ 有一个度为 $2t+2$的顶点（附加顶点）以及 $2d+2t$ 个度均为 d的顶点。由于 $t<\lceil (d+1)/4\rceil+1$，我们有$2t+2\le d$ 对于 $d\ge 7$，从而可得 $\Delta(B_{d,d+t})\le d$。此外，由于该图有 $2d+2t+1$ 个顶点，我们有 $\nu(B_{d,d+t})\le d+t=i$。更具体地说，如图4中用粗体边所示，可以得到一个大小为 $d+t=i$ 的匹配；在 F中取一个大小为 $d-t-2$ 的匹配，将 v与 H中的一个顶点匹配，使 H 另一侧的一个顶点保持未匹配。现在，由于 $t+1\le d/2<d-1$， H 中除这两个顶点外的所有顶点都可以与 F 中剩余的顶点匹配（记住， F 的每一侧都与 H的相对侧完全连接）。现在，根据度数计算边数，得到 $d^2+dt+t+1$条边；对于 $i=d+t$，该数等于 $di+i-d+1$，正如猜想2.1所预示的那样。

□

最后，我们利用上述结果和背包公式，为所有 d 值 7、8、9、10 构造极值图。对于这些 d 值，背包公式中至多有 4 个分量；因此，即使对于 $d=10$ 和 $m=10000$，求得最优解也只需 0.2 秒。最优解中各极值分量的数量以及由此得到的边数见表5。表5列出了 $2d<m\le 3d$ 的边数。列“# star ”表示 d-星图的数量，“#Z(d)”表示匹配数为 $Z(d)$ 的图的数量，“#other ”表示其他分量的数量。表 6 列出了 $d=8$ 更详尽的结果，其中“comp i”表示极值图中匹配数为 i 的分量数量。当 m 增加 10 时，唯一的变化是多出一个匹配数为 $Z(8)=10$ 的分量。表 5 和表 6 表明，对于 $d=7$、 8 、 9 和 10，命题 2.1（以猜想 2.1 为条件）成立。命题 2.1 描述了边极值图的一种构造模式；它指出，存在一种极值图，其中匹配数为 $Z(d)$的极值分量数量尽可能多，并辅以至多一个匹配数介于 d 和 $Z(d)$ 之间的极值分量，以及 d-星图；我们的所有结果都与这一预期相符，并为其提供支持

d	m	边数	#d-星图	$\#Z(d)$	#other
7	15	108	6	1	0
7	16	116	0	1	1
7	17	124	0	1	1
7	18	132	0	2	0
7	19	139	1	2	0
7	20	146	2	2	0
7	21	153	3	2	0
8	17	140	7	1	0
8	18	148	8	1	0
8	19	158	0	1	1
8	20	168	0	2	0
8	21	176	1	2	0
8	22	184	2	2	0
8	23	192	3	2	0
8	24	200	4	2	0
9	19	175	7	1	0
9	20	184	8	1	0
9	21	194	0	1	1
9	22	204	0	1	1
9	23	214	0	1	1
9	24	224	0	2	0
9	25	233	1	2	0
9	26	242	2	2	0
9	27	251	3	2	0
10	21	215	9	1	0
10	22	226	0	1	1
10	23	237	0	1	1
10	24	250	0	2	0
10	25	260	1	2	0
10	26	270	2	2	0
10	27	280	3	2	0
10	28	290	4	2	0
10	29	300	5	2	0
10	30	310	6	2	0

表5：背包公式结果。

因此，猜想 2.1 和 2.2均成立。由此可知，对于所有 m值以及 $d \in \{7, 8, 9, 10\}$，极值图的边数均等于猜想 2.2给出的公式。

d	m	边数 d-星图 comp 8		_	comp 9	comp 10
8	15	124	5	0	0	1
8	16	132	6	0	0	1
8	17	140	7	0	0	1
8	18	149	0	1	0	1
8	19	158	0	0	1	1
8	20	168	0	0	0	2
8	21	176	1	0	0	2
8	22	184	2	0	0	2
8	23	192	3	0	0	2
8	24	200	4	0	0	2
8	25	208	5	0	0	2
8	26	216	6	0	0	2
8	27	224	7	0	0	2
8	28	233	0	1	0	2
8	29	242	0	0	1	2
8	30	252	0	0	0	3
8	31	260	1	0	0	3
8	32	268	2	0	0	3
8	33	276	3	0	0	3
8	34	284	4	0	0	3
8	35	292	5	0	0	3
8	36	300	6	0	0	3
8	37	308	7	0	0	3
8	38	317	0	1	0	3
8	39	326	0	0	1	3
8	40	336	0	0	0	4
8	41	344	1	0	0	4
8	42	352	2	0	0	4
8	43	360	3	0	0	4
8	44	368	4	0	0	4
8	45	376	5	0	0	4
8	46	384	6	0	0	4
8	47	392	7	0	0	4

表 6：针对 $d = 8$ 的背包公式结果。

5 结论

本文研究了如下问题中的未解决情形：求最大度不超过 d 、匹配数不超过 m 的无三角形图的最大边数。据我们所知，我们提出的几种整数规划方法为解决这一极值问题提供了一种新思路。

由于我们的基本公式具有高度对称性，我们的大部分工作都集中在打破这种对称性上。我们提出了两种不同的方法：首先利用轨道分支进行分支决策以利用对称性，然后采用迭代方法，利用预计算上界与最优解值之间的接近程度。由于我们结合使用了迭代方法和轨道分支，因此最优求解的实例数量有所增加，整体求解时间也有所缩短。我们的方法将已知情形从 $d \leq 6$ 中的 [3] 扩展到 $d \leq 10$ (两者均适用于所有 $m > d$)。我们还找到了 $d = 11$、12 和 13 的一些极值分量。对于 $d = 11$ ，唯一尚未找到的极值分量对应于 $m = 14$；如果找到这个极值分量，背包公式就能计算出 $d = 11$ 以及任意 $m > d$的所有极值图。

需要特别指出的是，尽管整数规划方法在寻找更高 d 和 m 值的极值分量方面将受到限制，但它们为猜想 2.1、2.2以及命题 2.1提供了更多证据。事实上，我们的所有发现都支持以下主张：存在一个无三角形

图，其中匹配数为 $Z(d)$ 的因子临界极值分支数量尽可能多，匹配数小于 $Z(d)$ 的因子临界极值分支至多有一个，此外还有一些 d-星图。因此，我们的结果进一步促使人们寻求猜想2.1 和 2.2的形式化（结构性）证明，而这很可能需要更强大的工具。

最后，我们提出以下问题。需要指出的是，我们的整数规划模型利用极值分量为因子临界图这一事实，并在模型中采用了由此推得的顶点数。然而，它们并未明确强制构造出的图为因子临界图。显然，匹配数为 m 的因子临界图有 $2m+1$ 个顶点，但有 $2m+1$ 个顶点的图不一定是因子临界图。不过，事实证明，我们的整数规划模型得到的所有极值分量都是因子临界图。这一观察表明，研究以下问题或许很有意义：匹配数为 m、最大度为 d，且满足 $d < m < Z(d)$ 并有 $2m+1$ 个顶点的无三角形极值图 G 是否一定是因子临界图?

致谢

本研究得到土耳其科学技术研究理事会¨ ˙(TUBITAK)资助，项目编号为 122M452。

参考文献

[1] V. Chv´atal and D. Hanson，《度与匹配》，组合论杂志 B 辑，第 20 卷，第 2 期，第 128–138 页，1976年。[2] N. Balachandran and N. Khare，《具有受限度和匹配数的图》，离散数学，第 309 卷，第 12 期，第 4176–4180 页，2009年。[3] M. Ahanjideh, T. Ekim, and M. A. Yıldız，《最大度和匹配数受限的无三角形图的最大规模》，arXiv，预印本，2022年。arXiv：2207.02271。[4] P. Turan，《图论中的一个极值问题》，K¨oz´episk. Mat. ´es Fiz. Lapok，第 48 卷，第 436–452 页，1941年。[5] P. Erd¨os and R. Rado，《集合系的交集定理》，《伦敦数学学会杂志》，第 1 卷，第 1 期，第 85–90 页，1960年。[6] C. Dibek, T. Ekim, and P. Heggernes，《最大度和匹配数受限的无爪图的最大边数》，离散数学，第 340 卷，第 5 期，第 927–934 页，2017年。[7] J. Blair, P. Heggernes, P. Lima, and D. Lokshtanov，《度和匹配数受限的弦图的最大边数》，Algorithmica，第 84 卷，第 3587–3602 页，2022年。[8] E. K. M˚aland，《度和匹配约束下图类的最大边数》，硕士论文，卑尔根大学，2015年。[9] F. Furini, I. Ljubi´c, and P. S. Segundo，《一种用于计算拉姆齐数的双层优化新方法》，手稿，2021年。 [在线]。可访问：https: / / optimization - online . org/?p=17323。[10] P. Hansen, M. Aouchiche, G. Caporossi, A. Hertz, and C. Sellal，《混合整数规划与极值化学图》，化学与化学工程系统杂志，第 3 卷，第 22–30 页，2018年。[11] R. Ito, N. A. Azam, C. Wang, A. Shurbevski, H. Nagamochi, and T. Akutsu，《一种基于人工神经网络和整数规划的单环化合物逆向 QSAR/QSPR 新方法》。Cham：Springer，2021年，第 641–655 页。[12] J. Ostrowski, J. Linderoth, F. Rossi, and S. Smriglio，《轨道分支》，数学规划，第 126 卷，第 1 期，第 147–178 页，2011年。[13] F. Margot，《利用对称整数线性规划中的轨道》，数学规划，第 98 卷，第 1 期，第 3–21 页，2003年。

[14] M. E. Pfetsch and T. Rehn，《混合整数规划对称性处理方法的计算比较》，*Mathematical Programming Computation*，第 11 卷，第 1 期，第 37–93 页，2019年。[15] B. D. McKay and A. Piperno，《实用图同构，II》，符号计算杂志，第 60 卷，第 94–112 页，2014年。[16] D. S. Johnson，《NP 完全性专栏》，*ACM Transactions on Algorithms (TALG)*，第 1 卷，第 1 期，第 160–176 页，2005 年。