arXiv:2305.03106v2 [math.CO] 8 May 2023

通过添加叶构建果园网络

Leo van Iersel
代尔夫特应用数学研究所，代尔夫特理工大学，荷兰
MarkJones !
代尔夫特应用数学研究所，代尔夫特理工大学，荷兰
EstherJulien !
代尔夫特应用数学研究所，代尔夫特理工大学，荷兰
YukihiroMurakami [1] !
代尔夫特应用数　　学研究所，代尔夫特理工大学，荷　　兰

摘要

系统发育网络用于表示物种的进化历史。最近，人们提出了果园网络这一新类别，后来证明它们可以解释为带有额外水平弧的树。这使得该网络类别非常适合描述涉及水平基因转移的进化历史。在此，我们研究使一个网络成为果园网络所需的最少额外叶节点数。我们证明，计算给定网络的这一接近度量是 NP困难的，并给出了一个紧上界。我们还提出了一种基于顶点标记的等价度量，以构建混合整数线性规划表述。我们的实验结果涵盖真实世界数据和合成数据，展示了该实现的有效性。

2012 ACM 学科分类 计算数学 → 树；计算数学 →图算法；应用计算 → 生物网络；应用计算 → 生物信息学

关键词和短语 系统发育学、网络、果园网络、接近度量、NP-困难性

数字对象标识符 10.4230/LIPIcs...

补充材料 源代码： `https://github.com/estherjulien/OrchardProximity`

资助*Leo van Iersel*：研究部分由荷兰科学研究组织（NWO）的资助项目 OCENW.KLEIN.125 和 OCENW.GROOT.2019.015 提供经费支持。
Mark Jones：研究经费由荷兰科学研究组织（NWO）项目 OCENW.KLEIN.125 提供。

Esther Julien：研究经费由荷兰科学研究组织（NWO）资助，资助项目编号为 OCENW.GROOT.2019.015。

1 引言

系统发育树用于表示物种的进化历史。虽然它们能够有效展示通过垂直传递发生的物种形成事件，但不足以表示更为复杂的进化过程。杂交和水平基因转移（水平基因转移）等网状事件可能产生无法在单棵树上表示的信号 [9, 21]。鉴于此，系统发育网络因其能够阐明网状进化过程而日益受到关注。

系统发育网络通常根据其拓扑特征被划分为不同的网络类。这些分类通常出于计算方面的考虑，但有些网络类也是根据其生物学相关性来定义的 [16]。网络类的经典示例包括树子

[1] 通讯作者

©Leo van Iersel、Mark Jones、Esther Julien 和 Yukihiro Murakami；根据知识共享署名 4.0 国际许可协议授权。Leibniz 国际信息学会议论文集，Schloss Dagstuhl – Leibniz 信息学中心，德国 Dagstuhl 出版社

XX:2 通过添加叶来构建果园网络

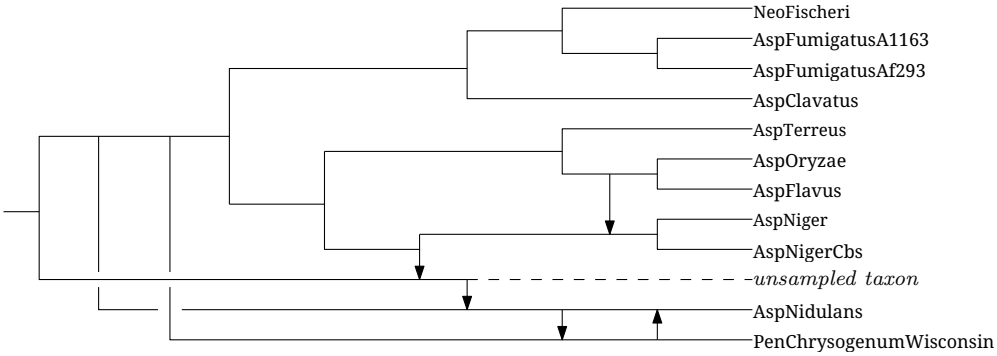


图1 一个包含11个不同真菌分类单元（不包括未采样分类群）的网络，其中包括5个网状顶点；该网络是 [18]所示更大网络的一部分。图中的有向弧是连接弧，表示基因转移通道。为了使所有连接弧都保持水平，我们需要添加一个额外叶节点（未采样分类群）来表示进化历史。

网络 [3] 和树基网络 [8]。粗略地说，树子网络是指每个顶点都通过垂直传递将基因传递给了某个现存物种的网络，而树基网络则是指可以通过在树弧之间添加所谓的连接弧从一棵树得到的网络。近期的发展最终促成了果园网络的引入；就包含关系而言，它们介于上述两个网络类之间 [14, 4]。事实证明，该类网络在算法上具有吸引力，在生物学上也具有相关性；它们被定义为可以通过一系列所谓的摘樱桃操作归约为单个叶节点的网络，并且已证明它们可以通过向树添加水平弧而得到（其中树的根位于顶部，弧指向下方） [19]。此类水平弧可用于模拟水平基因转移事件，因此，果园网络特别适合表示每个网状事件都是水平转移的进化情景。果园网络还基于所谓的樱桃覆盖进行了静态刻画 [20]。

在考虑非果园网络时，一个自然的问题是：需要多少个额外叶节点，才能将该网络转变为果园网络？从生物学角度来看，这个问题可以理解为：需要将多少个灭绝物种或未采样类群引入网络，才能得到这样一种情形，即每个网状顶点都代表一次水平基因转移事件。鉴于水平基因转移是细菌网状演化的主要驱动力 [10]，这是一项至关重要的研究。我们在图1中给出了一个包含少数真菌物种的网络；要使其成为果园网络，需要添加一个额外叶节点。形式上，计算这一叶节点添加度量的问题如下。

$L_{\mathcal{OR}}$-距离（判定）**输入：**一个网络 N、一个分类单元集合 X 以及一个自然数 k。**判定：**能否通过至多 k 次叶添加将 N 构造为果园网络？

在相关研究中，叶添加度量已针对其他网络类展开研究。研究表明，可以在多项式时间内找出使网络成为基于树的网络所需的最少叶添加次数 [7]。同一篇论文还证明，叶添加度量等价于另外两种接近度量，即基于生成树和不相交路径划分的度量。对于无根变体（网络中的弧为无向），研究者也提出了同样的问题，而该问题被证明是 NP 完全的 [5]。这篇后续论文共引入了八种接近度量，包括基于边添加和重排移动的度量。有些论文不考虑叶添加，而是将叶删除（通常为顶点删除）

作为所谓的基于边的网络 [6]类的接近度量。最后，针对果园网络，近期一篇学士学位论文比较了叶添加邻近度量与另一种基于弧删除的接近度量在一般情况下有何不同 [17]。

本文证明，对于树子网络这一类网络，叶添加邻近度量可以在多项式时间内计算；并给出一种更高效的算法，用于计算树基网络的该度量。我们通过从三度顶点覆盖进行多项式时间归约，证明 $L_{\mathcal{OR}}$-Distance 是 NP 完全的。为了将该问题建模为一个 混合整数线性规划 (混合整数线性规划)，我们根据顶点标记重新表述叶添加度量。已知果园网络是带有水平弧的树；粗略来说，这意味着我们可以为果园网络的顶点标记，使每个入度为 2 的顶点恰有一条入弧，其端点具有相同的标记。重新表述后的度量称为垂直弧 邻近度量，它对所有可能的顶点标记（在第6.1节中正式定义）进行考察，并统计其中仅有非水平弧的入度为 2 的顶点的最小数量。我们的实验结果颇具前景，因为实际案例都能在不到一秒的时间内求解。此外，该模型在合成数据上的表现也很好。

本文结构如下。第2节中，我们给出果园网络和树基网络所需的全部定义和刻画。第3节中，我们正式引入树子网络、果园网络和树基网络类别的叶添加度量。第4节中，我们证明 $L_{\mathcal{OR}}$-Distance 是 NP 完全的（定理17）。第5节中，我们给出叶添加邻近度量的紧上界。第6节中，我们重新表述叶添加度量，以描述用于求解 $L_{\mathcal{OR}}$- Distance 的混合整数线性规划；第6.3节中，我们展示该混合整数线性规划应用于真实和模拟网络时的实验结果。第7节中，我们简要讨论研究结果，并探讨未来可能的研究方向。附录 A 中收录了部分结果的证明。

2 预备知识

一个二叉有向系统发育网络定义在非空集合 X 上，是一个有向无环图，具有一个根，其入度为 0、出度为 1；树节点的入度为 1、出度为 2；网状顶点的入度为 2、出度为 1；叶的入度为 1、出度为 0，并且由 X中的元素进行双射标记。

为简洁起见，我们将二叉有向系统发育网络简称为网络。在本文中，除非另有说明，假设 N 是定义在某个非空集合 X上的网络，其中 $|X| = n$。不含网状结构的网络称为树。树节点和网状顶点有时统称为内部顶点。

如果 u 是网络的根，则网络的弧 uv 称为根弧 。如果 v 是网状顶点，则网络的弧 uv 称为网状弧 ，否则称为树弧 。如果 uv 是网络中的一条弧，我们称顶点 u 为另一个顶点 v 的父节点 ；在这种情况下，我们称 v 为 u的子节点 。此外，我们分别称 u 和 v 为弧 uv 的尾 和头 。换言之，我们可以将弧写作 $uv = \text{tail}(uv)\ \text{head}(uv)$。 v 的邻居指以 v为父节点或子节点的顶点集合。如果顶点 u和 v 具有相同的父节点，我们也称它们为兄弟节点 。

接下来，我们将定义基于顶点和弧删除的图操作。为确保生成的图仍是网络，每次图操作后我们都会进行清理。形式上，我们通过反复执行以下操作清理网络，直到没有任何一项操作适用为止。

XX:4 通过添加叶来构建一个 NetworkOrchard

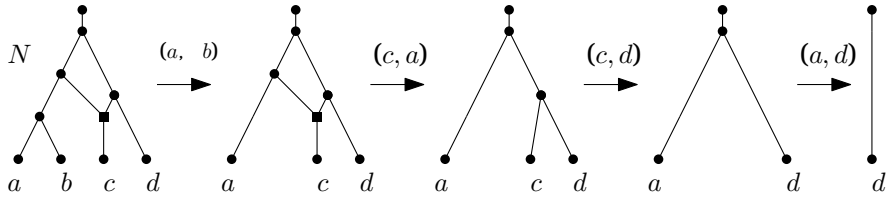


图 2 一个果园网络示例 N，通过序列 $(a,b)(c,a)(c,d)(a,d)$ 进行归约。该网络 N 包含一个樱桃 (a,b) 和一个网状樱桃 (c,d)。后续网络均由前一个网络经一次樱桃采摘归约得到。例如，第二个网络 $N(a,b)$ 是从 N 中移除叶节点 b 并进行清理后得到的。

- 抑制一个入度为 1、出度为 1 的顶点（例如，如果 uv 和 vw 是弧，且 v 是一个入度为 1、出度为 1 的顶点，则删除顶点 v 并添加一条弧 uw，以此抑制 v ）。将平行弧替换为一条弧（例如，如果 uv 是网络中的一条重复弧，则删除其中一条弧 uv）。

我们注意到，删除一条非网状弧并进行清理后，会得到一个包含两个入度为 0 的顶点的图。另一方面，删除一条网状弧并进行清理后，会得到一个网络。因此，我们将“弧删除”用于指代网状弧删除。

2.1 树子网络

如果每个非叶节点都有一个子节点是树顶点或叶节点，则称该网络为树子网络。如果网络中的一个内部顶点的所有子节点都是网状顶点，则称该内部顶点为全能节点 [15]。根据定义，网络是树子网络，当且仅当它不包含全能节点。

2.2 果园网络

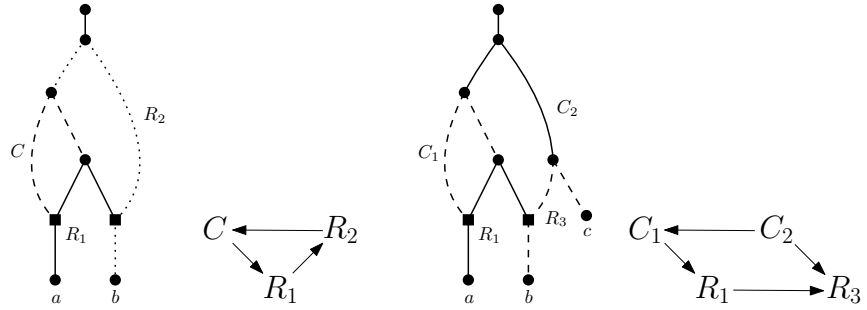
要定义果园网络，我们必须先定义樱桃和网状樱桃，以及将它们约简的操作。下文定义的示意图见图 2。设 N 是一个网络。若 N 的两个叶节点 x 和 y 互为兄弟节点，则它们构成一个樱桃。在这种情况下，我们称 N 包含樱桃（x，y）或樱桃（y，x）。若 x 的父节点 p_x 是网状顶点，且 y 的父节点也是 p_x的父节点，则 N的两个叶节点 x 和 y 构成一个网状樱桃。在这种情况下，我们称 N 包含网状樱桃（x，y）。从 N 中约简樱桃（x，y），是指删除叶节点 x 并进行清理的过程。从 N 中约简网状樱桃（x，y），是指删除从 y 的父节点到 x 的父节点的弧并进行清理的过程。在这两种情况下，我们用 N（x，y）表示所得的网络。

若存在一个序列 $S=(x_1,\ y_1)(x_2,\ y_2)\ldots\ldots(x_k,\ y_k)$，使得 NS是一个仅含单个叶节点的网络 y_k，则网络 N是果园网络。已证明，依次规约（网状）樱桃的顺序无关紧要 [4, 14]。除了这种递归定义外，果园网络还可通过樱桃覆盖（弧分解）[20] 和顶点标记 [19]来刻画。我们在此介绍这两种刻画。

樱桃覆盖（参见 [20] 了解更多详情）： A樱桃形是由三个互不相同的顶点 x、y、p及弧 px和 py构成的子图。樱桃形的内部顶点是 p，端点是 x 和 y。网状樱桃形是由四个互不相同的顶点 x、y、p_x、p_y及弧 p_xx、p_yp_x、p_yy构成的子图，其中 p_x是网络中的一个网状顶点。网状樱桃形的内部顶点是 p_x 和 p_y，端点是 x 和 y。网状樱桃形的中间弧是 p_yp_x。我们常用弧来指代樱桃形

和网状樱桃形（例如，我们会将上述樱桃形记作$\{p_x x、\ p_y y\}$，将网状樱桃形记作$\{p_x x、\ p_y p_x、\ p_y y\}$）。若弧 uv属于樱桃形或网状樱桃形 C，且 $uv \in C$，则称弧 uv被该樱桃形或网状樱桃形覆盖。二元网络的樱桃覆盖是由樱桃形和网状樱桃形组成的集合 P，其中除根弧外的每条弧都恰好被 P覆盖一次。一般来说，一个网络可以有不止一种樱桃覆盖。

我们将樱桃覆盖的樱桃覆盖辅助图 $G = (V, A)$定义如下。对于所有形状 $B \in P$，都有 $v_B \in V$。如果形状 $B \in P$的一个内部顶点是形状 B的端点，则形状 $B \in P$直接位于另一个形状 $C \in P$的上方。于是，$v_B v_C \in A$。（改编自 [20, 定义 2.13]）。如果樱桃覆盖的辅助图中存在一个环，我们就称它是有环的。否则称它无环。图 3 展示了有环和无环樱桃覆盖的示例。



(a) 网络 N_1 **(b)** 樱桃覆盖 aux. graph of N_1 **(c)** 网络 N_2 **(d)** 樱桃覆盖辅助。graph of N_2

图 3 樱桃覆盖示例。(a) 一个网络 N_1，其顶点集为 $\{a, b\}$，具有樱桃覆盖 $\{C, R_1, R_2\}$。(b) 基于 (a) 中樱桃覆盖的 N_1 的（有环）辅助图。(c) 通过向 N_1 添加一个叶节点 c 得到的网络 N_2，具有樱桃覆盖 $\{C_1, C_2, R_1, R_3\}$ (d) 基于 (d) 中樱桃覆盖的 N_2 的（无环）辅助图。

▶ **定理 1 (** [20])。网络 N 是果园网络，当且仅当它具有无环樱桃覆盖。

非时间标记：设 N 是一个具有顶点集 V（N）的网络。N 的一个 非时间标记[2] 是一个标记 t： V（N）$\to \mathbb{R}$，满足

- 对于所有弧 uv, $t(u) \leq t(v)$，仅当 v 是网状顶点时才允许相等；对于每个内部顶点 u，
- 存在一个 u 的子节点 v，使得 $t(u) < t(v)$；对于每个具有父节点 u 和 v的网状顶点
- r， $t(u) = t(r)$ 和 $t(v) = t(r)$ 至多有一个成立。

请注意，每个网络（无论是否为果园网络）都可以进行非时间标记：将每个顶点标记为其到根的最长距离（假设每条弧的权重均为 1）。

在非时间标记下，如果一条弧的两个端点具有相同的标记，我们称其为水平 弧；否则称其为垂直 弧。根据定义，只有网状弧才可能是水平弧。若每个网状顶点恰好关联一条入水平弧，我们就称该非时间标记为水平基因转移一致标记 。我们回顾以下关键结果。

▶ **定理2**（ [19]）。一个网络是果园网络，当且仅当它存在一个与水平基因转移（*HGT*）一致的标记。

[2] 这个名称是为了与 [1] 的时间表示相区别。在时间表示中，要求每条网状弧的两个端点都具有相同的标签。

XX：6 通过添加叶来构造 NetworkOrchard

2.3 基于树的网络

如果一个网络 N 可以通过以下步骤从 T 获得，则称其为具有 基于树的 基树 T。

1. 将 T 中的一些弧替换为路径，这些路径的内部顶点称为 附着点；每个附着点的入度和出度均为 1。2. 在附着点之间添加弧，称为 连接弧，使图中不存在总度大于 3的顶点，并保持无环。3. 清理。

树子网络、果园网络和树基网络这几个类别之间的关系可以表述如下。

▶ **引理 3**（[14]以及 [19]）。 如果一个网络是树子网络，那么它就是果园网络。如果一个网络是果园网络，那么它就是树基网络。

这里给出一种基于弧划分的树基网络静态刻画，称为极大之字形迹 [11, 23]。设 N 为一个网络。长度为 k 的之字形迹 是弧序列 $(a_1, a_2, \ldots\ldots, a_k)$，其中 $k \geq 1$，且对于 $i \in [k-1] = \{1, 2, \ldots\ldots, k-1\}$，满足 tail$(a_i)$ $=$tail(a_{i+1}) 或 head(a_i) $=$head(a_{i+1})。如果不存在包含 Z 作为子序列的之字形迹，我们就称之字形迹 Z 极大的 。根据 tail(a_1) 和 tail(a_k) 的性质，极大之字形迹有四种可能。

- 冠： $k \geq 4$ 为偶数，且 tail(a_1) $=$ tail(a_k) 或 head(a_1) $=$ head(a_k)。*M*-栅栏：
- $k \geq 2$ 为偶数，不是冠，且对于每个 $i \in [k]$，tail(a_i) 都是树顶点。*N*-围栏： $k \geq 1$
- 为奇数，且 tail(a_1) 或 tail(a_k)（但不能两者同时）是网状顶点。通过重新排列弧，以下不妨假设 tail(a_1) 是网状顶点，而 tail(a_k) 是树顶点。

- *W*-栅栏： $k \geq 2$ 为偶数，且 tail(a_1) 和 tail(a_k) 均为网状顶点。我们称 S 个极大的之字形迹的集合为 N 的 之字形分解，如果 S 中的元素恰好划分了 N 中除根弧之外的所有弧。

▶ **引理 4 (** [11]的推论 4.6**)**。 设 N 为一个网络。当且仅当 N 没有 *W*围栏时，它才是基于树的。

▶ **定理 5**（定理 4.2，来自 [11]）。 任意网络 N 都有唯一的之字形分解。

▶ **定理 6**（定理 3.3，来自 [20]）。 设 N 为一个网络。当且仅当 N 具有樱桃覆盖时， N 是基于树的。

3 叶节点添加邻近性度量

设 N 为 X上的一个网络。添加叶节点 $x \notin X$ 到弧 e，该弧属于网络 N，是指添加一个带标签的顶点 x，用顶点 w 细分弧 e（如果 $e = uv$，则删除弧 uv，添加顶点 w，并添加弧 uw 和 wv），以及添加一条弧 wx。我们将所得网络记为 $N + (e, x)$。当上述弧 e 无关紧要或含义明确时，我们简称这一过程为添加叶节点 x 到 N，并将所得网络记为 $N + x$。

在本节中，$\mathcal{C}$ 将用来表示一个网络类。特别地，我们将用$\mathcal{TC}$,$\mathcal{OR}$和$\mathcal{TB}$ 分别表示树子网络、果园网络和树基网络的类。令 $L_{\mathcal{C}}(N)$表示通过叶添加使网络 N成为$\mathcal{C}$的成员所需的最少次数。我们首先证明，计算$L_{\mathcal{TC}}(N)$和$L_{\mathcal{TB}}(N)$可以在多项式时间内完成。

▶ **引理 7**。 设 N 是一个网络。那么 $L_{\mathcal{TC}}(N)$ 等于全父节点的数量。此外，通过在每个全能节点的一条出弧上恰好添加一个叶节点，可以使 N 成为树子网络。

▶ **引理8**。 设 N 为一个网络。那么 $L_{\mathcal{TC}}(N)$ 可以在 $O(|N|)$ 时间内计算得出。

已经证明，$L_{\mathcal{TB}}(N)$ 可以在 $O(|N|^{3/2})$ 时间内计算，其中 $|N|$ 是 N [7] 中的顶点数。我们将证明，实际上可以在 $O(|N|)$ 时间内完成计算。

▶ **引理9**。 设 N 为一个网络。那么 $L_{\mathcal{TB}}(N)$ 等于 W-围栏的数量。此外，在 N 中每个 W-围栏的任意弧上添加一个叶节点后，N 便成为基于树的网络。

▶ **引理10**。 设 N 为一个网络。那么 $L_{\mathcal{TB}}(N)$ 可在 $O(|N|)$ 时间内计算。

有趣的是，计算 $L_{\mathcal{OR}}(N)$ 被证明是一个困难的问题，尽管对于其相邻网络类而言，叶添加邻近度量很容易计算。我们将在第4节中证明以下结论。

▷ **定理17**。 设 N 为一个网络。计算 $L_{\mathcal{OR}}(N)$ 是 NP困难的。

我们还给出以下定理：该定理指出，在考虑果园网络的叶添加邻近度量时，只需考虑向网状弧添加叶节点。因此，以下我们假设所有叶添加都是在网状弧上进行的。

▶**定理 11** （ [17]定理4.1）。 网络 N 是果园网络，当且仅当向 N 的一条树弧添加一个叶节点后得到的网络是果园网络。

本文接下来将聚焦于计算 $L_{\mathcal{OR}}(N)$ 的问题。

4 困难性证明

本节中，我们通过从三度顶点覆盖问题归约，证明计算 $L_{\mathcal{OR}}(N)$ 是 NP困难的。

三度顶点覆盖（判定）**输入：** 一个 3-正则图 $G=(V,\ E)$ 和一个自然数 k。**判定：** G 是否有一个大小至多为 k 个顶点的顶点覆盖?

$L_{\mathcal{OR}}$-距离（判定）**输入：** 一个定义在分类单元集合 X 上的网络 N 和一个自然数 k。**判定：** 能否通过至多 k 次叶添加，使 N 成为果园网络?

现在我们描述从三度顶点覆盖到 $L_{\mathcal{OR}}$-距离的归约。对于一个图 G，令 $V(G)$ 和 $E(G)$ 分别为其顶点集和边集。给定一个三度顶点覆盖实例 $(G,\ k)$，按如下方式构造一个 $L_{\mathcal{OR}}$-距离实例 $(N_G,\ k)$（见图4）：

1. 对于每个顶点 v 在 $V(G)$，构造一个构件 $\mathrm{Gad}(v)$，如下所述。下文中，形如 ℓ_i^v 的顶点是叶节点，顶点 r_i^v 是网状顶点，顶点 w_i^v、m_i^v 和 ρ^v 是树节点。

Gad(v)中的关键结构是一个包含15条弧的N-栅栏，始于弧 $\boldsymbol{r_0^v r_1^v}$，接着是对每个 $i \in [6]$ 而言形如 $\boldsymbol{w_i^v r_i^v}, \boldsymbol{w_i^v r_{i+1}^v}$ 的弧，最后是弧 $\boldsymbol{w_7^v r_7^v}, \boldsymbol{w_7^v \ell_5^v}$。这组以粗体显示的弧称为Gad($v$)的主体部分 。此外，网状顶点 $r_1^v, r_2^v, r_3^v, r_4^v$ 分别有叶子节点 $\ell_1^v, \ell_2^v, \ell_3^v, \ell_4^v$。

XX：8 通过添加叶构建 NetworkOrchard

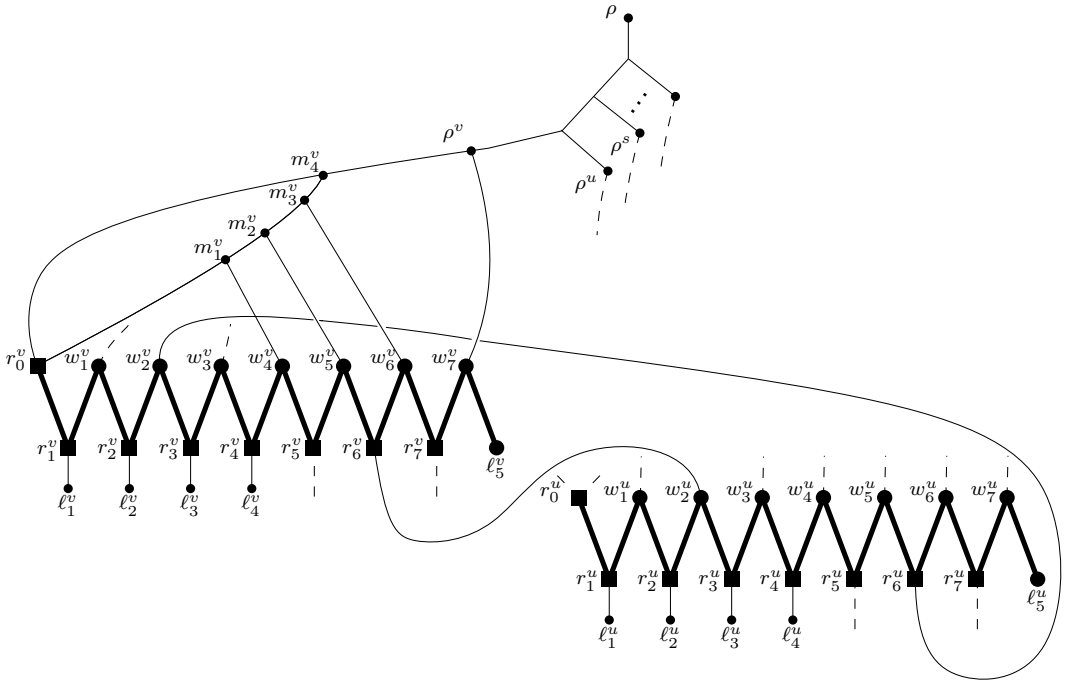


图4示意图　　网络 N_G 在 G 包含一条　　条边 uv时的情况。

在Gad(v)的主体部分上方，添加一组树节点 $m_1^v, m_2^v, m_3^v, m_4^v, \rho^v$ ，其子节点如下： m_1^v 有子节点 r_0^v 和 w_4^v， m_2^v 有子节点 m_1^v 和 w_5^v， m_3^v 有子节点 m_2^v和 w_6^v， m_4^v有子节点 m_3^v和 r_0^v， ρ^v有子节点 m_4^v和 w_7^v（见图4）。

至此，Gad(v) 的构造完成。注意，到目前为止，顶点 w_1^v、 w_2^v、 w_3^v没有入弧，而 r_5^v、 r_6^v、 r_7^v没有出弧。稍后将添加这些弧，以将不同的构件连接起来。

2. 按如下方式连接 ρ^v 中每个 $\mathrm{Gad}(v)$ 的顶点：取顶点 $\{v_1$、……、$v_g\}$ 的一个排列，其中 G。添加一个顶点 ρ 以及顶点 s_i，其中 $i \in [g-1]$。添加弧 ρs_1，以及从集合 $\{s_i s_{i+1}$： $i \in [g-2]\}$ 引出的弧；同时添加从集合 $\{s_i \rho^{v_i}$： $i \in [g-1]\}$引出的弧，最后再添加一条弧 $s_{g-1}\rho^{v_g}$。

3. 接下来，以如下方式在对应于 G 中相邻顶点的构件之间添加弧：对于 G 中每对相邻顶点 u, v ，添加一条弧，连接 Gad(u) 中的一个顶点 r_5^u, r_6^u, r_7^u 与 Gad(v) 中的一个顶点 w_1^v, w_2^v, w_3^v（并对称地，添加一条弧，连接 r_5^v, r_6^v, r_7^v 中的一个顶点与 w_1^u, w_2^u, w_3^u 中的一个顶点）。弧所连接顶点的具体选择并不重要，但应确保每个顶点恰好被这样的一条弧使用。形式化地说：对于 G 中每个邻居为 a, b, c 的顶点 v ，固定两个（任意的）映射 π_v: {a, b, c} → {1, 2, 3} 和 τ_v: {a, b, c} → {5, 6, 7}。然后，对于 G 中每对相邻顶点 u, v ，添加一条从 $r^u_{\tau_u(v)}$ 到 $w^v_{\pi_v(u)}$ 的弧（并对称地，添加一条从 $r^v_{\tau_v(u)}$ 到 $w^u_{\pi_u(v)}$ 的弧）。

4. 最后，对于每个顶点 v 在 G中，将顶点 $\{\ell_i^v$： $i \in [5]\}$ 在 $\mathrm{Gad}(v)$ 标记为 ℓ_i^v。

将所得的图称为 N_G；很容易看出， N_G 是有向无环图，且只有一个根 ρ。因此，它是叶集 $\{\ell_i^v$： $i \in [5]$, $v \in V(G)\}$。 由于 N_G的弧分解为 M-栅栏和 N-栅栏，我们得到以下观察。

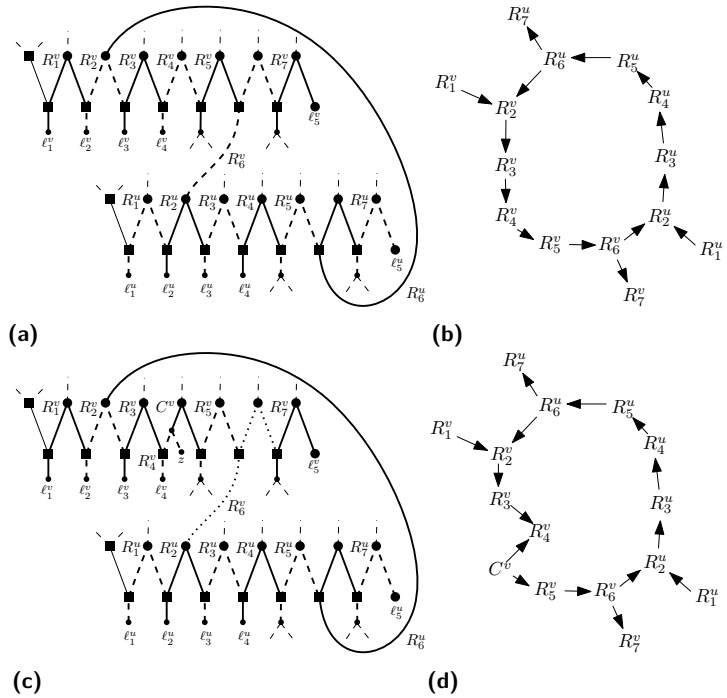


图 5 Gad(v) 和 Gad(u) 的樱桃覆盖。(a) 中展示了 Gad(v) 和 Gad(u) 的主体部分的唯一樱桃覆盖；(b) 给出了 (a) 的樱桃覆盖辅助图。(c) 中，将叶节点 $z \notin X$ 添加到 Gad(v) 的主体部分，并给出了网络同一部分的一种可能的樱桃覆盖。(d) 则给出了 (c) 的樱桃覆盖辅助图。

▶**观察 12**。 设 G 是一个 *3*-正则图， N_G 是通过归约得到的网络。那么 N_G 是基于树的。

根据观察 12 和定理 6，从现在起我们可以直接假设 N_G 有樱桃覆盖。在证明主要结果之前，我们需要一些记号和辅助引理。设 N 是一个网络，且 $\hat{}$ N_i 是 N 的一个 N-栅栏。下文中，我们将写作 $\hat{}$ N_i:=(a_1^i, a_2^i, ..., $a_{k_i}^i$)，并令 c_{2j-1}^i 表示对于 $j \in \left[\frac{k_i-1}{2}\right]$，head($a_{2j-1}^i$) 的子节点。第一个引理指出，尽管一个树基网络可能有不唯一的樱桃覆盖，但覆盖 N-围栏中各弧的网状樱桃形是固定的。

$\hat{}\hat{}$ $\hat{}$▶**引理 13.** 设 N 是一个树基网络，并令 N_1、N_2、.. 、N_n 表示 N 中长度至少为 *3* 的 *N*-栅栏。那么， N 的每个樱桃覆盖都包含网状樱桃形 {(头(a_{2j-1}^i)c_{2j-1}^i) a_{2j}^i, a_{2j+1}^i}，其中 $i \in [n]$ 和 $j \in \left[\frac{k_i-1}{2}\right]$。

注意，对于每个 $v \in V(G)$，构件 Gad(v) 的主体部分都是一个 N-栅栏。对于所有 $v \in V(G)$，我们将构件 Gad(v) 的主体部分记为 ($a_1^v, a_2^v, \ldots, a_{15}^v$)。根据引理 13， a_i^v对于 $i = 2, \ldots, 15$ 和 $v \in E(G)$，在 N_G 的所有可能樱桃覆盖中，它们都以相同的方式被覆盖。对于偶数 $i \in [15]$，我们将包含 a_i^v和 a_{i+1}^v 的网状樱桃形记为 $R_{i/2}^v$。图 5a 和图 5b 展示了一个樱桃覆盖辅助图的局部示例，其中包含 R_i^v 和 R_i^u，其中 $i \in [7]$，且 uv 是 G 中的某条边。注意，这些樱桃形构成了一个环。下一个引理表明，事实上，对于 G 中的任意边 uv，都存在这样一个环。

$\hat{}$▶**引理 14.** 设 N 是一个基于树的网络，并假设对于两个长度至少为 3 的 *N*-栅栏 N_u :=$(a_1^u, a_2^u, \ldots, a_{k_u}^u)$ 和 $\hat{}$ N_v:=$(a_1^v, a_2^v, \ldots, a_{k_v}^v)$，存在有向路径经过其中。

N 从头(a_h^u) 到尾(a_i^v)，以及从头(a_j^v) 到尾(a_k^u)，对于偶数 h, i, j, k，其中有 $k < h$ 和$i < j$。那么， N 的每个樱桃覆盖辅助图都包含一个环。

为了从可能的樱桃覆盖中移除所有可能的环，因此必须破坏任意边 $uv \in G$ 对应的 Gad(u) 或 Gad(v) 的主体部分。

▶**引理 15.** 设 G 为一个 *3*-正则图， N_G 为通过上述归约得到的网络。假设 A 是 N_G的一个弧集，且向 A中的每条弧添加叶节点后得到一个果园网络。对于每条边 $uv \in E(G)$，存在一条弧 $a \in A$，它属于$\mathrm{Gad}(u)$ 或$\mathrm{Gad}(v)$的主体部分。

为了完成对该归约有效性的证明，我们证明：要通过添加叶节点使 N_G成为果园网络，只需（且必须）对顶点覆盖 V_{sol} 中的每个 v，将一个叶节点 z^v 添加到 Gad(v) 的一条适当弧上，其中 V_{sol} 是 G 的顶点覆盖（见图 5c）。关键思路是，这会将 Gad(v) 的主体部分从一个 N-栅栏拆分为一个 N-栅栏和一个 M-栅栏，从而使我们能够避开樱桃覆盖辅助图中的环（见图 5d）。

▶**引理 16.** 设 G 为一个 *3*-正则图，令 N_G 为通过上述归约得到的网络。那么，当且仅当 $L_{\mathcal{OR}}(N_G) = k$ 时， G 具有大小为 k 的最小顶点覆盖。

▶**定理17**。 设 N 是一个网络。判定问题 $L_{\mathcal{OR}}$-距离是 *NP* 完全。计算 $L_{\mathcal{OR}}(N)$ 是 *NP*困难。

证明。 假设给定弧集 A_{sol}，属于 N_G，其大小至多为 k。向 A_{sol}中的每条弧添加叶后，我们可以在多项式时间内检查所得网络是否为果园网络（参见 [14] 的第6节）。这意味着 $L_{\mathcal{OR}}$-Distance属于 NP。从三度顶点覆盖 到 $L_{\mathcal{OR}}$-Distance的归约在本节开头已有概述；由于对三度顶点覆盖 实例中的每个顶点，我们都只添加有限个顶点和弧，因此该归约可在多项式时间内完成。根据引理 16 所证明的两个问题之间的等价性， $L_{\mathcal{OR}}$-Distance 是 NP 完全的。因此，$L_{\mathcal{OR}}$-Distance的优化问题，即计算 $L_{\mathcal{OR}}(N)$ 的问题，是 NP困难的。
◀

5 上界

在上一节中，我们证明了计算 $L_{\mathcal{OR}}(N)$ 是 NP困难的。本节中，我们为 $L_{\mathcal{OR}}(N)$ 给出了一个紧确的上界。如果一个网状顶点没有网状节点祖先，我们就称它为最高的。

▶**引理18**。 设 N 为一个网络。假设存在一个最高的网状顶点 r，使得所有其他网状顶点都有一个叶节点兄弟节点。那么 N 是果园网络。

证明。 我们通过对网状结构的数量 k 进行归纳来证明引理。对于基例，注意到只有一个网状顶点的网络是树子网络，因为它没有全父节点。树子网络是果园网络 [14], ，因此该网络必定是果园网络。

现在，假设我们已经证明了所有网状顶点少于 k 的网络都满足该引理，其中 $k > 1$。令 N 为一个网状顶点集为 R 的网络，其中 $|R| = k$；并假设存在 N 中的一个最高网状顶点 r，使得所有其他网状顶点都有一个叶节点兄弟。令 r 表示引理陈述中指定的最高网状顶点。选择一个最低的网状顶点 $r' \in R \setminus \{r\}$。根据假设， r' 有一个叶节点兄弟 c。 r' 下方的每个顶点都必须是树节点或叶节点。不断进行樱桃约简，直到 r' 的子节点 x 成为叶节点。此时，(x, c) 是一对网状樱桃；通过归约这对网状樱桃得到的网络 N' 有 $k-1$ 个网状顶点，并且有一个最高网状顶点 r，使得所有其他网状顶点都有一个叶

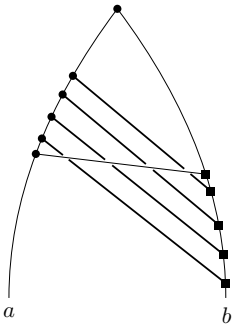


图6 一个有两个叶节点的网络 $N\{a,\ b\}$，有 $r(N) = 5$ 个网状顶点。注意，$L_{\mathcal{OR}}(N) = r(N) - 1 = 4$，因为除非先通过樱桃采摘规约其下方的网状顶点，否则最高的网状顶点无法规约。对于每个非最高网状顶点，我们必须在其一条入弧上添加一个叶节点才能规约它，这便得到 $L_{\mathcal{OR}}(N) = 4$。注意，这种构造可以推广到任意 k 个网状顶点。

节点兄弟。根据归纳假设，N' 必定是果园网络。由于可以对 N 进行一系列樱桃约简以得到 N'，因此网络 N 也必定是果园网络。◀

▶ **定理19.** 设 N 是一个网络，并令 $r(N)$ 表示网状顶点的数量。那么 $L_{\mathcal{OR}}(N) = 0$，如果 N是一棵树；否则，$L_{\mathcal{OR}}(N) \leq r(N) - 1$，其中该界是紧的。

证明。 如果 N 是一棵树，那么它就是果园网络，因此 $L_{\mathcal{OR}}(N) = 0$。因此，假设 $r(N) > 0$。令r 为 N中位置最高的网状顶点，并为每个其他网状顶点任意选取一条入网状弧。在这些网状弧上各添加一个叶节点。根据引理18，所得网络必为果园网络。我们在 N中除一个网状顶点外的每个网状顶点上都添加了一个叶节点。因此 $L_{\mathcal{OR}}(N) \leq r(N) - 1$。图6中的网络表明这个上界是紧的。◀

6 混合整数线性规划公式

为了将计算叶添加邻近度量的问题建模为混合整数线性规划，我们将该度量重新表述为非时间标记。

6.1 指向网状顶点的垂直弧

根据定理2，每个果园网络都可以看作一个网络，其基树上的每条连接弧都是水平弧。回顾非时间标记的定义，这意味着存在一种标记，使得每个网状顶点恰有一条水平入网状弧。按照此定义，我们引入第二种果园网络邻近度量。给定网络 N 的一个非时间标记，我们用仅有垂直入弧的网状结构 来指称 N 中仅有垂直入弧的网状顶点。令 $V_{\mathcal{OR}}(N)$ 表示所有可能的非时间标记中仅有垂直入弧的网状顶点的最小数量。

▶**观察20。** 设 N 为一个网络。当且仅当 $V_{\mathcal{OR}}(N) = 0$时，该网络存在水平基因转移一致标记。换言之，当且仅当 $V_{\mathcal{OR}}(N) = 0$时，该网络为果园网络。

特别是，我们展示了一个更强的结果，即两个接近度量是等价的。

XX:12 通过添加叶节点构造 NetworkOrchard

▶**引理 21.** 设 N为一个网络。那么 $L_{\mathcal{OR}}(N) = V_{\mathcal{OR}}(N)$。

证明。首先，假设我们有一个网络 N，带有某种非时间标记 $t\colon V(N) \to \mathbb{R}$，它会产生 h 个仅有垂直入弧的网状结构。对于每个仅有垂直入弧的网状结构 r，其父节点为 u 和 v，我们在弧 ur 上添加一个叶节点 x（这样添加不失一般性；将叶节点添加到 vr上也可得到同样的论证）。由于 r 是仅有垂直入弧的网状结构，因此必有 $t(u) < t(r)$，且 $t(v) < t(r)$。令 p_x表示 x的父节点，我们将 $t(p_x)$ 标记为 $:= t(r)$，并将 $t(x)$ 标记为 $:= t(p_x) + 1$。这样，包含 x 和 p_x 的映射 t 的扩展便是 $N + x$的一个非时间标记。注意， r 在 $N + x$中不再是仅有垂直入弧的网状结构，因为弧 $p_x r$ 是水平的。因此，在仅有垂直入弧的网状结构的一条入弧上添加叶节点，可以使这类网状结构的数量恰好减少一个。对每个仅有垂直入弧的网状结构重复此过程，可得 $L_{\mathcal{OR}}(N) \leq V_{\mathcal{OR}}(N)$。

为证明另一个方向，假设我们可以向 N 添加 ℓ 个叶，使其成为果园网络。根据定理11，我们可以假设所有这些叶都添加到集合 $\{e_1, \ldots, e_\ell\}$ 中的网状弧上。根据定理2，得到的网络 N' 有一个水平基因转移一致标记 $t\colon V(N') \to \mathbb{R}$。

我们声称，将标记 $t|_{V(N)}$ 限制在 N 上是非时间标记，并且在 $t|_{V(N)}$下，顶部仅有垂直入弧的网状结构数量至多为 ℓ。假设在网状弧 $e_i = u_i r_i$上添加了一个叶节点 x_i 。令 p_i 表示网络 N'中 x_i 的父节点。根据水平基因转移一致标记的定义，我们必须有 $t(u_i) < t(r_i)$，因为 $u_i p_i r_i$ 是 N'中的一条路径。因此，将标记限制在从 N' 中移除叶节点 x_i后得到的网络上，所得标记是非时间标记。此外，若 v_i 是 r_i 的父节点，且 v_i 不是 u_i，则 $v_i r_i$ 或 $p_i r_i$ 中必有一个在 N'中为水平弧。如果 $v_i r_i$ 是水平弧，那么移除 x_i后， r_i 仍有一条水平入弧，且仅有垂直入弧的网状结构数量不变。另一方面，如果 $p_i r_i$ 是水平弧，那么 $v_i r_i$ 必定是一条垂直弧。删除 x_i后，网状顶点 r_i 变为仅有垂直入弧的网状结构，因为它的另一条入弧 $u_i r_i$ 也是垂直弧。由于叶删除是局部操作，删除一个叶节点最多会使仅有垂直入弧的网状结构数量增加一。对每条网状弧 e_i （其中 $i \in [\ell]$）重复此过程，可知 N 至多包含 ℓ个仅有垂直入弧的网状结构，因此 $V_{\mathcal{OR}}(N) \leq L_{\mathcal{OR}}(N)$。◀

6.2 混合整数线性规划公式

根据引理 21，我们有 $L_{\mathcal{OR}}(N) = V_{\mathcal{OR}}(N)$。本节将介绍一种混合整数线性规划表述，以获得 $V_{\mathcal{OR}}(N)$，从而也获得 $L_{\mathcal{OR}}(N)$。具体做法是搜索网络的非时间标记，使垂直弧的数量最少。

设 N 为给定网络，其顶点集为 V，弧集为 A。令 R 表示 N 的网状顶点集合。我们将决策变量 l_v 定义为顶点 $v \in V$ 的非时间标签。树弧和垂直连接弧 uv 具有性质 $l_u < l_v$。若弧 $a \in A$ 为垂直弧，则定义 x_a 为 1，否则为 0。若 $v \in R$ 是一个仅有入垂直弧的网状顶点，则定义 h_v 为 1，否则为 0。设 $v \in V$ 为 N 的一个顶点。下文中，令 $P_v \subset V$ 为 v 的父节点集合， $C_v \subset V$ 为 v 的子节点集合， X 为叶集合。令 ρ 为 N 的根。于是，混合整数线性规划

表述如下：

$$
\begin{aligned}
\min_{x,h,l} \quad & \sum_{v\in R} h_v & & \\
\text{s.t.} \quad & \sum_{u\in P_v} x_{uv} - 1 \le h_v & \forall v \in R & (1)\\
& \sum_{v\in C_u} x_{uv} \ge 1 & \forall u \in V \setminus X & (2)\\
& \sum_{u\in P_v} x_{uv} \ge 1 & \forall v \in V \setminus \{\rho\} & (3)\\
& l_u \le l_v & \forall uv \in A & (4)\\
& l_u \le l_v - 1 & \forall v \in V \setminus R, \forall u \in P_v & (5)\\
& l_u \le l_v - 1 + |V|(1 - x_{uv}) & \forall v \in R,\ \forall u \in P_v & (6)\\
& l_u \ge l_v - |V| x_{uv} & \forall v \in R,\ \forall u \in P_v & (7)\\
& x_a \in \{0,1\} & \forall a \in A & \\
& h_v \in \{0,1\} & \forall v \in R & \\
& l_v \in \mathbb{R}_+ & \forall v \in V &
\end{aligned}
$$

根据约束 (1)，如果网状顶点 v 的所有入弧都是垂直的，则 h_v 变为 1。根据 (2)，所有顶点都必须至少有一条出垂直弧。接着，(3) 保证每个网状顶点至少有一条入垂直弧。约束 (4) 在网络中创建非时间标记，其中根据 (5)，如果 v 不是网状顶点，则 u 的标记严格小于 v 的标记。然后，对于所有网状顶点 v，如果 uv 是垂直的，则 (6) 将 x_{uv} 设为 1。最后，根据 (7)，如果 x_{uv} 为 0，则 u 和 v 的标记相等。

6.3 实验结果

本节中，我们将上一节介绍的混合整数线性规划应用于一组真实的二叉网络，并对模拟网络进行运行时间分析。这些实验的代码以 Python 编写，可在 `https://github.com/estherjulien/OrchardProximity` 获取。所有实验均在配备 16 GB 内存、主频为 1.8 GHz 的 Intel Core i7 CPU 上运行。求解混合整数线性规划问题时，我们使用开源求解器 SCIP [2]。

真实数据集由若干论文中收集的不同二叉网络组成，汇集于 `http://phylnet.univ-mlv.fr/recophync/networkDraw.php`。这些网络的叶节点集大小最多为 39，网状结构数量最多为 9，但有一个例外，其网状结构数量为 32。所有二叉实例均在一秒内完成（最长为 0.072 秒）。根据结果，我们发现 22 个二叉网络中只有两个满足 $L_{\mathcal{OR}}(N) > 0$，因此，只有两个不是果园网络。

模拟数据使用 [22] 的出生-杂交网络生成器生成。该生成器有两个用户定义的参数：λ，用于调节物种形成率，以及ν，用于调节杂交率。根据 [13]，我们设置 $\lambda = 1$，并均匀随机采样$\nu \in [0.0001, 0.4]$。对于每一对值 (L, R)，我们生成了一个大小为 50 的实例组，其中叶的数量为 $L \in \{20, 50, 100, 150, 200\}$，网状结构的数量为 $R \in \{5, 10, 20, 30, 40, 50, 100, 200\}$。在我们的实现中，我们仅为传入的网状弧定义了变量 x_a。因此，二元变量仅取决于网络中的网状结构数量。在表1中，模拟实例的运行时间和 $L_{\mathcal{OR}}(N)$ 值结果均针对网络的网状结构数量进行了展示。时间限制设置为一小时。从这些结果中可以观察到，对于网络规模达到

表1 模拟网络的结果，其中 R 表示网络中的网状顶点数量，完成率（%）表示在一小时时限内求解完成的实例百分比，完成实例运行时间给出已完成实例运行时间的 5%、50% 和 95% 分位数，最后，$L_{\mathcal{OR}}(N)$给出已完成实例的最优目标值的不同分位数百分比。

R	完成率（%）	完成实例运行时间（秒）			$L_{\mathcal{OR}}(N)$		
		5%	50%	95%	5%	50%	95%
5	100.0	0.002	0.008	0.018	0	0	0
10	99.2	0.004	0.011	0.023	0	0	2
20	98.8	0.008	0.016	0.051	0	0	4
30	98.0	0.013	0.023	0.089	0	1	7
40	97.2	0.018	0.038	0.189	0	1	10
50	97.6	0.022	0.049	0.632	0	2	13
100	75.6	0.058	0.233	109.395	2	8	28
200	61.6	0.614	15.783	3,078.631	14	27	56

对于 50 个网状结构，几乎所有实例都能在一秒内求得最优解。接着，对于$R = 100$，200，运行时间会增加，但通常仍能在合理时间内求解这些实例。

7 讨论

本文研究了使一个网络成为果园网络所需的最少叶添加数，并将其作为衡量网络偏离果园网络程度的一种方式。我们证明，计算这一指标是 NP困难的（定理17），并给出了一个紧确上界，即网状结构数减一（定理19）。我们将该指标重新表述为：在所有可能的非时间标记中，最小化仅有垂直入弧的网状结构的数量。在第6节中，我们利用这一重新表述，将计算叶添加指标的问题建模为混合整数线性规划。实验结果表明，真实数据实例可在一秒内求解；该表述在合成实例上也表现良好，几乎所有包含至多50个网状顶点和200个叶节点的实例都能在一秒内求解。然而，对于规模更大的实例，运行时间有所增加。

与不同网络类中相应问题的计算复杂度相比，我们的 NP 困难性结果颇有意思。对于基于树的网络，求出使网络成为基于树的网络所需添加的最少叶节点数，可以在多项式时间内完成 [7] （引理10）；我们还证明，对于树子网络类，情况也相同（引理8）。有趣的是，树子网络类包含于果园网络类 [14] ，而果园网络类又包含于基于树的网络类 [12]。这种 NP 困难性的夹层或许可以归因于禁形的缺失。通过添加叶节点来构造树子网络或基于树的网络，目标都是消除网络中的某些禁形。对于树子网络，我们在每个全能节点的一条出弧上恰好添加一个叶节点；对于基于树的网络，我们则在每个 W围栏的一条弧上恰好添加一个叶节点。迄今为止，以（局部）禁形来刻画果园网络一直难以实现 [14] ——或许，该问题在果园网络这一变体上的 NP 困难性结果表明，要为果园网络找到这样的刻画可能并不可行。

还可以考虑非二叉网络的叶节点添加问题。非二叉网络通过允许顶点具有不同的мунасив性来推广本文所研究的网络。

入度和出度。由于二元版本是一个特例，这个广义问题仍然是 NP 完全的。尝试为非二元版本寻找混合整数线性规划公式可能会很有意思。

另一个自然的研究方向是考察不同的接近度量。其中一个可能尤其值得关注的是基于弧删除的接近度量。也就是说，使一个网络成为果园网络所需删除的网状弧的最少数量是多少？Susanna 证明，这一度量与叶添加邻近度量不可比较 [17], ，但目前尚不清楚计算这一度量是否同样属于 NP困难问题。

参考文献

1 Mihaela Baroni、Charles Semple 和 Mike Steel。实时杂交。《系统生物学》，55(1):46–56，2006。2 Ksenia Bestuzheva、Mathieu Besançon、Wei-Kun Chen、Antonia Chmiela、Tim Donkiewicz、Jasper van Doornmalen、Leon Eifler、Oliver Gaul、Gerald Gamrath、Ambros Gleixner、Le-ona Gottwald、Christoph Graczyk、Katrin Halbig、Alexander Hoen、Christopher Hojny、Rolf van der Hulst、Thorsten Koch、Marco Lübbecke、Stephen J. Maher、Frederic Matter、Erik Mühmer、Benjamin Müller、Marc E. Pfetsch、Daniel Rehfeldt、Steffan Schlein、Franziska Schlösser、Felipe Serrano、Yuji Shinano、Boro Sofranac、Mark Turner、Stefan Vigerske、Fabian Wegscheider、Philipp Wellner、Dieter Weninger 和 Jakob Witzig。《SCIP 优化套件 8.0》。技术报告，Optimization Online，2021年12月。URL:http://www.optimization-online.org/DB_HTML/2021/12/8728.html。3 Gabriel Cardona、Francesc Rosselló 和 Gabriel Valiente。树子系统发育网络的比较。《IEEE/ACM 计算生物学与生物信息学汇刊》，6(4):552–569，2008。4 Péter L Erdős、Charles Semple 和 Mike Steel。一类可根据祖先谱系重建的系统发育网络。《数学生物科学》，313:33–40，2019。5 Mareike Fischer 和 Andrew Francis。我的网络有多接近树基？无根系统发育网络的接近度量。《离散应用数学》，283:98–114，2020。6 Mareike Fischer、Tom Niklas Hamann 和 Kristina Wicke。我的网络距离基于边还有多远？无根系统发育网络边基性质的接近度量。arXiv 预印本 arXiv:2207.01370，2022。7 Andrew Francis、Charles Semple 和 Mike Steel。树基网络的新刻画及其接近度量。《应用数学进展》，93:93–107，2018。8 Andrew R Francis 和 Mike Steel。哪些系统发育网络不过是附加了弧的树？《系统生物学》，64(5):768–777，2015。9 Benjamin E Goulet、Federico Roda 和 Robin Hopkins。植物杂交：旧观点，新技术。《植物生理学》，173(1):65–78，2017。10 Carlton Gyles 和 Patrick Boerlin。水平转移的遗传元件及其在细菌性疾病致病过程中的作用。《兽医病理学》，51(2):328–340，2014。11 Momoko Hayamizu。根化二叉系统发育网络的结构定理及其对树基网络的启示。《SIAM 离散数学杂志》，35(4):2490–2516，2021。12 Katharina T Huber、Leo van Iersel、Remie Janssen、Mark Jones、Vincent Moulton、Yukihiro Murakami 和 Charles Semple。无向系统发育网络的定向。arXiv 预印本 arXiv:1906.07430，2019。13 Remie Janssen 和 Pengyu Liu。比较系统发育网络生成器的拓扑结构。《生物信息学与计算生物学杂志》，19(06):2140012，2021。14 Remie Janssen 和 Yukihiro Murakami。关于樱桃挑选与网络包含。《理论计算机科学》，856:121–150，2021。15 Laura Jetten 和 Leo van Iersel。非二叉树基系统发育网络。《IEEE/ACM 计算生物学与生物信息学汇刊》，15(1):205–217，2016。

16 Fabio Pardi 和 Celine Scornavacca。可重建的系统发育网络：不要区分无法区分的对象。《*PLoS* 计算生物学》，11(4):e1004135，2015。17 Merel Susanna。《打造果园网络：判断系统发育网络是否为果园网络，以及将非果园网络转换为果园网络的算法》。学士学位论文，代尔夫特理工大学，2022。http://resolver.tudelft.nl/uuid:724ac2af-e569-4586-b367-288fef890252。18 Gergely J Szöllősi、Adrián Arellano Davín、Eric Tannier、Vincent Daubin 和 Bastien Boussau。基因组尺度的系统发育分析发现真菌间存在广泛的基因转移。《英国皇家学会哲学汇刊 *B*：生物科学》，370(1678):20140335，2015。19 Leo van Iersel、Remie Janssen、Mark Jones 和 Yukihiro Murakami。果园网络是附加了水平弧的树。《数学生物学通报》，84(8):1–21，2022。20 Leo van Iersel、Remie Janssen、Mark Jones、Yukihiro Murakami 和 Norbert Zeh。利用樱桃覆盖统一刻画树基网络和果园网络。《应用数学进展》，129:102222，2021。21 David A Wickell 和 Fay-Wei Li。水平基因转移在植物进化中的意义。《新植物学家》，225(1):113–117，2020。22 Chi Zhang、Huw A Ogilvie、Alexei J Drummond 和 Tanja Stadler。基于多位点序列数据的物种网络贝叶斯推断。《分子生物学与进化》，35(2):504–517，2018。23 Louxin Zhang。关于树基系统发育网络。《计算生物学杂志》，23(7):553–565，2016。

A 附录

▷ **引理 7.** 设 N 是一个网络。那么，$L_{TC}(N)$ 等于全父节点的数量。此外，N 可以通过在每个全父节点的一条出弧上恰好添加一个叶节点，使其成为树子网络。

证明。 根据定义，当且仅当网络不包含任何全父节点时，它才是树子网络。我们证明，每次添加叶节点，都能使所得网络中的全父节点比原网络少一个。设 uv 是一条弧，其中 u 是一个全父节点。在 uv处添加一个叶节点 x 。在所得网络中，u 有一个子节点（ x的父节点），该子节点是树顶点，因此它不再是全父节点。新添加的树顶点有一个叶节点子节点 x；网络其余部分的父子关系保持不变，因此每次添加叶节点至多能移除一个全父节点（本例中为 u）。由此可得，$L_{TC}(N)$ 至少等于 N中的全父节点数量。通过选择尾部为全父节点的弧，每次添加叶节点都至少能移除一个全父节点，因此 $L_{TC}(N)$ 至多等于 N中的全父节点数量。因此，$L_{TC}(N)$ 恰好等于 N中的全父节点数量。◀

▷ **引理8.** 设 N 是一个网络。那么可以在 $O(|N|)$ 时间内计算 $L_{TC}(N)$ 。

证明。 我们首先说明，可以通过检查每个顶点的子节点的入度，在 $O(|N|)$ 时间内计算 N 的全父节点数量。当且仅当一个顶点的所有子节点的入度均为 2 时，该顶点才是全能节点。由于每个顶点的度至多为 3，for 循环中的每次查找都只需常数时间。for 循环遍历大小为 $O(N)$ 的顶点集。根据引理 7，由于 $L_{TC}(N)$ 是 N中的全父节点数量，我们可以在 $O(|N|)$ 时间内计算 $L_{TC}(N)$。◀

已有研究表明，$L_{TB}(N)$ 可以在 $O(|N|^{3/2})$ 时间内计算得出，其中 $|N|$ 是 N [7] 中的顶点数。我们将证明，实际上可以在 $O(|N|)$ 时间内完成计算。

▷**引理9**。 设 N 是一个网络。则 $L_{\mathcal{TB}}(N)$ 等于 W-栅栏的数量。此外，通过在 N 中每个 W-栅栏的任意一条弧上添加一个叶节点，可以使 N 变为基于树的网络。

证明。根据引理 4，当且仅当网络中不含 W-栅栏时，该网络才是基于树的。我们证明，每次添加叶节点都能使网络中的 W-栅栏数量比原网络少一个。假设 N 至少含有一个 W-栅栏。否则，根据引理 4，我们可以得出该网络是基于树的。设 $(a_1, a_2, \ldots, a_k)$ 是 N 中的一个 W-栅栏，其中 $a_i = u_i v_i$ 对应于 $i \in [k]$，并将一个叶节点 x 添加到 a_1；令 p_x 为 x 的树顶点父节点。在所得网络中，$\{u_1 p_x, p_x v_1, p_x x, a_2, a_3, a_4, \ldots, a_k$ 中的弧被分解为其唯一的极大之字形迹（定理 5），即两个 N-栅栏 $(u_1 p_x)$ 和 $(a_k, a_{k-1}, \ldots, a_3, a_2, p_x v_1, p_x x)$。所有其他弧仍处于与 N 相同的极大之字形迹中。因此，W-栅栏的数量恰好减少了一个。对网络中的所有 W-栅栏重复此过程即可；由此可知，$L_{\mathcal{T}}$B(N) 是 N 中 W-栅栏的数量。

快速检查可知，在 W围栏的任意弧上添加一个叶节点，会将 W围栏分解为两个 N-栅栏。◀

▷**引理10**。 设 N 为一个网络。那么 $L_{\mathcal{TB}}(N)$ 可以在 $O(|N|)$ 时间内计算得到。

证明。 求出极大的之字形分解需要 $O(|N|)$ 时间（[11] 的命题5.1）。根据引理9，计算该分解中 W-栅栏的数量可得 $L_{\mathcal{TB}}(N)$。◀

▷**观察 12.** 设 G 为一个 *3*-正则图， N_G 为通过该归约得到的网络。那么 N_G 是基于树的。

证明。 很容易检查， N_G 的弧可分解为 M-栅栏和 N-围栏（每个构件 $\mathrm{Gad}(v)$ 的主体部分是一个 N-栅栏；离开构件主体部分的每条弧 $\mathrm{Gad}(v)$ 都是长度为 1 的 N-栅栏；其余弧可分解为长度为 2 的 M-栅栏）。根据引理 4， N_G 必须是基于树的。◀

▷**引理 15.** 设 G 为一个 *3*-正则图，且设 N_G 为通过上述归约得到的网络。假设 A 是 N_G 的一个弧集，通过向 A 中的每条弧添加叶，得到一个果园网络。对于每条边 $uv \in E(G)$，存在一条弧 $a \in A$，它是 $\mathrm{Gad}(u)$ 或 $\mathrm{Gad}(v)$的主体部分中的一条弧。

证明。 我们通过逆否命题证明该引理。假设存在一条边 $uv \in E(G)$，使得 $\mathrm{Gad}(u)$ 或 $\mathrm{Gad}(v)$ 的主体部分中没有任何弧属于 A。我们将证明，向 $a \in A$ 中的所有 N_G 添加叶所得到的网络——记为 $N_G + A$ ——不是果园网络。

根据定理 1，我们知道 N_G 是果园网络，当且仅当 N_G 有一个无环樱桃覆盖。我们在此证明 $N_G + A$ 不会有无环樱桃覆盖，从而证明 $N_G + A$ 不是果园网络。

由于没有向 Gad(u) 或 Gad(v) 的主体部分添加弧，这些主体部分在 $N_G + A$ 中仍然是 N-栅栏。此外，根据构造，对于偶数 $h \geq 10$ 和偶数 $i \leq 6$， N 中存在一条从某个 head(a_h^v) 到 tail(a_i^v) 的弧，因此 $N_G + A$ 中存在一条从 head(a_h^v) 到 tail(a_i^v) 的路径。类似地，对于某个偶数 $j \geq 10$ 和 $h \leq 6$， $N_G + A$ 中存在一条从 head(a_j^v) 到 tail(a_k^v) 的路径。于是，引理 14 蕴含 $N_G + A$ 的任意樱桃覆盖的辅助图都包含一个环。根据定理 1，我们有 $N_G + A$ 不是果园网络。◀

^^ ^▷**引理 13**。 设 N 是一个树基网络，并令 N_1、N_2、……、N_n 表示 N 中长度至少为 *3* 的 *N*-栅栏。那么， N 的每个樱桃覆盖都包含网状樱桃形 *{*（头部a^i_{2j-1}）c^i_{2j-1}） a^i_{2j}、a^i_{2j+1}*}*，其中 $i \in [n]$ 和 $j \in \left[\binom{k_i - 1}{2}\right]$。 ——

XX:18 通过添加叶构造 NetworkOrchard

证明。令 $\hat{}\ N_i = (a^i_1、\ a^i_2、\ \ldots\ldots、\ a^i_{k_i})$ 为一个长度为 $k_i \geq 3$ 的 N-栅栏。注意，在每个樱桃覆盖中，每个网状顶点恰有一条入弧被网状樱桃形作为中间弧覆盖（因为该网络是二叉的；对于非二叉网络，一般而言并非如此 [20]）。由于 head(a^i_1) 是一个网状顶点，因此 a^i_1 或 a^i_2 中必有一个属于某个网状樱桃形的中间弧。但 tail(a^i_1) 是一个网状顶点；因此，a^i_2 必须属于某个网状樱桃形的中间弧。同一个网状樱桃形的另外两条弧随即确定为 head(a^i_1)c^i_1 和 a^i_3。对 $j \in \left[\frac{k_i-1}{2}\right]$ 的网状顶点 head(a^i_{2j+1}) 重复此论证，即可得到 N-栅栏 $\hat{}\ N_i$ 的所需结论；再对每个 N- 栅栏重复此论证，即可得到所需结论。◀

▷引理 *14.* 设 N 是一个树基网络，并假设对于两个长度至少为 *3* 的 *N*-栅栏 $N_u := (a^u_1, a^u_2, \ldots, a^u_{k_u})$ 和 $\hat{}\ N_v := (a^v_1, a^v_2, \ldots, a^v_{k_v})$，在$N$中存在从 *head*($a^u_h$) 到 *tail*($a^v_i$) 以及从 *head*($a^v_j$) 到 *tail*($a^u_k$) 的有向路径，其中 h, i, j, k 为偶数，且满足 $k < h$ 和$i < j$。那么， N 的每个樱桃覆盖辅助图都包含一个环。

证明。再次将包含 a^u_h 和 a^u_{h+1} 的网状樱桃形记为 $R^u_{h/2}$，同理将 $R^v_{i/2}$、 $R^v_{j/2}$和 $R^u_{k/2}$ 分别记为相应的网状樱桃形。根据引理 13， $R^u_{h/2}$、 $R^v_{i/2}$、 $R^v_{j/2}$、 $R^u_{k/2}$ 均出现在樱桃覆盖辅助图中。此外， $R^u_{k/2}$ 位于 $R^u_{h/2}$之上，且 $R^v_{i/2}$ 位于 $R^v_{j/2}$之上。现在注意，从 head(a^u_h) 到 tail(a^v_i) 的路径上任意一对连续的弧，要么属于樱桃覆盖中的同一个网状樱桃形，要么属于两个不同的樱桃形，且其中一个樱桃形直接位于另一个之上。这意味着在樱桃覆盖辅助图中存在一条从 $R^u_{h/2}$ 到 $R^v_{i/2}$ 的路径。类似的论证表明，从 $R^v_{j/2}$ 到 $R^u_{k/2}$ 也存在一条路径。但这样一来， $R^u_{h/2}$ 位于 $R^v_{i/2}$之上，而 $R^v_{i/2}$位于 $R^v_{j/2}$之上， $R^v_{j/2}$又位于 $R^u_{k/2}$之上， $R^u_{k/2}$再位于 $R^u_{h/2}$ 之上，因此形成了一个环。◀

▷**引理 16.** 设 G 是一个 *3*-正则图， N_G 是通过上述归约得到的网络。那么，当且仅当 $L_{\mathcal{OR}}(N_G) = k$ 时， G 有一个大小为 k 的最小顶点覆盖。

证明。 首先假设 V_{sol} 是 G 的一个顶点覆盖，且至多包含 k 个顶点。我们将证明，对每个 Gad(v) ，在其主体部分的一条弧上添加一个叶节点， $v \in V_{sol}$ 就能使 N_G成为果园网络。这表明 G 的最小顶点覆盖至少为 $L_{\mathcal{OR}}(N_G)$。在本证明的其余部分，我们将把 N_G 的顶点和弧称为上述归约中引入的顶点和弧。

对于每个 $v \in V_{sol}$，我们都在 Gad(v) 中的弧 $w^v_4 r^v_4$ 上添加一个叶节点 z^v （见图 5c）。设 q^v 是 z^v 的父节点。关键思路是，这会将 Gad(v) 的主体部分中的一个 N-栅栏拆分为一个 N-栅栏和一个 M-栅栏，从而使我们能够避开樱桃覆盖辅助图中的环（见图 5d）。

我们将新网络称为 M。为了形式化地证明 M 是果园网络，我们给出一个水平基因转移一致标记 $t: V(M) \to \mathbb{R}$。

首先设置 $t(\rho) = 0$，对于 s_1、……、 s_{g-1} 或 ρ^v、 m^v_4、……、 m^v_2 中的任意顶点，若 v 在 $V(G)$ 中，则令该顶点的标签等于其父节点的标签加 1。令 h 为目前分配给顶点的最大值，现在调整 t，将每个标签减去 $(h+1)$。因此，现在可以假设，对于 ρ、 s_1、……、 s_{g-1} 或 m^v_4、……、 m^v_2 中的所有顶点，若 v 在 $V(G)$ 中，则其标签均为 ≤ -1。现在，对 $V(G)$ 中的每个 v，设置 $t(m^v_1) = 0$ 和 $t(r^v_0) = 0$。

很容易看出，到目前为止， t 满足水平基因转移一致标记的性质。接下来还需要为每个构件 Gad(v) 主体部分中的顶点、每个构件的叶，以及为 $v \in V_{sol}$ 新增的顶点 q^v 和 z^v 进行标记。我们按如下方式进行。

对于 $v \in V_{sol}$，设置 $t(r^v_1) = t(w^v_1) = 12, t(r^v_2) = t(w^v_2) = 13, t(r^v_3) = t(w^v_3) = 14$，以及 $t(r^v_4) = t(q^v) = 15$。设置 $t(w^v_4) = 1, t(r^v_5) = t(w^v_5) = 2, t(r^v_6) = t(w^v_6) = 3$，以及 $t(r^v_7) = t(w^v_7) = 4$。

对于 $v \notin V_{sol}$，为每个 i 设置 $t(r_1^v) = t(w_1^v) = 5$，以及 $t(r_i^v) = t(w_i^v) = i + 4$，直至 $t(r_7^v) = t(w_7^i) = 11$。

最后，对于每个父节点为 p 的叶节点 ℓ，令 $t(\ell) = t(p) + 1$。

还需注意，t 是 M 的非时间标记，并且对于 M中的每个网状顶点 r，r 恰有一个父节点 p 满足 $t(p) = t(r)$。因此，t 是M的水平基因转移一致标记，由定理2可知，M 是果园网络。

现在假设我们有一个大小至多为 k的 N_G 的弧集 A_{sol}，使得在弧集 A_{sol} 中的弧上添加叶节点后，N_G 成为果园网络。根据引理15，对于每条边 $uv \in E(G)$，存在一条弧 $a \in A_{sol}$，它是 Gad(u) 或 Gad(v) 的主体部分中的一条弧。因此，集合 $\{v \in V(G)$: A_{sol} 包含 Gad(v) 主体部分中的一条弧$\}$显然是 G的一个顶点覆盖。由于这对任何这样的弧集 A_{sol}都成立，因此，如果存在这样一个大小至多为 k的 A_{sol}，那么必定存在一个大小至多为 k的 G 的顶点覆盖。◀