

PQ-树断点距离中心问题的复杂性和精确算法

刘培霞 姜海涛 朱大铭

(山东大学计算机科学与技术学院 济南 250101)

(liupeixia0629@163.com)

Complexity and Algorithm for Minimum Breakpoint Median from PQ-trees

Liu Peixia, Jiang Haitao, and Zhu Daming

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

Abstract A PQ-tree is a tree-based data structure which can represent large sets of permutations. Although the uncertainty of complete ancestral genomes is known, the homologous species provide information to determine the relative locations of partial genomes, whereby the PQ-tree can be designed to efficiently store ancestral genomes. In evolutionary biology, phylogenetic trees represent the evolutionary relationships between species. In the construction of phylogenetic trees, the leaves of the trees represent current species whose genomes are annotated by permutations, and the internal nodes represent ancestral genomes whose genomes are annotated by PO-trees. In order to determine the evolutionary relationships between different species, it is necessary to quantify the distances between the known permutations and the permutations in the constructed PQ-trees. Based on the measurement of breakpoint distance, we investigate p -Minimum Breakpoint Median from the PQ-tree. Given a PQ-tree and the corresponding p permutations, the goal is to identify a permutation generated by the PQ-tree that holds the minimum breakpoint distances relative to the p permutations. Our study shows that when $p \geq 2$, the p -Minimum Breakpoint Median problem from PQ-tree is NP-complete. Moreover, when $p = 1$, the problem is fixed parameters tractable. To solve the 1-Minimum Breakpoint Median from PQ-trees, we propose a Fixed-Parameter Tractable algorithm whose computation complexity is $O(3^K n)$ where K is the optimized breakpoint distance.

Key words PQ-tree; breakpoint distance; fixed-parameter tractable; permutation; NP-complete

摘要 PQ-树是一种树状数据结构,用来表示元素排列集合.虽然消逝物种完整基因组序列具有不确定性,但是根据同源物种可以确定部分基因的相对位置,所以可以利用 PQ-树来存储消逝物种的基因组.在生物学中,进化树用来表示物种之间的进化关系.当构建生物进化树时,叶子结点表示现存物种,其基因组用排列表示;内部结点为祖先物种,其基因组用 PQ-树表示.为了确定物种间的进化关系,需要确定 PQ-树可以产生的排列与已知排列之间的距离.以断点距离为标准,研究了 p -PQ-树断点中心问题,即从给定 PQ-树中产生一个排列,使之与给定的 p 个排列的断点距离之和最小.证明当 $p \geq 2$ 时, p -PQ-树断点中心问题是 NP-完全的.当 $p = 1$ 时, p -PQ-树断点中心问题是参数化可计算的,针对

收稿日期:2014-11-17;修回日期:2015-01-27

基金项目:国家自然科学基金项目(61202014, 61472222);山东省自然科学基金项目(ZR2012FQ008);中国博士后科学基金项目(2011M5001133, 2012T50614)

This work was supported by the National Natural Science Foundation of China (61202014, 61472222), the Natural Science Foundation of Shandong Province of China (ZR2012FQ008), and the China Postdoctoral Science Foundation (2011M5001133, 2012T50614).

通信作者:姜海涛(htjiang@sdu.edu.cn)

1-PQ-树断点中心问题,提出了时间复杂度为 $O(3^K n)$ 的参数化算法,其中 K 为最优解的断点距离。

关键词 PQ-树;断点距离;固定参数可解;排列;NP-完全

中图法分类号 TP301.6

PQ-树是由 Kellogg 和 George^[1] 发现并命名的一种树状数据结构,可以用来表示元素排列的集合. PQ-树主要用来解决一些目标是寻找满足各种约束的排列的问题,在这些问题中,通过一次添加一种约束的方式编辑 PQ-树的结构,使得编辑后的 PQ-树能表示满足约束的排列. PQ-树的主要应用有:验证矩阵是否具有连续 1 性质^[1]、判定图的平面性^[2]等. PQ-树是字符表 $\Sigma = \{1, 2, \dots, n\}$ 上的一棵平面有根树,它有 3 类结点:叶子结点、 P 结点和 Q 结点,每个叶子结点表示字符表中的一个元素,任意 2 个叶子结点表示的元素不同,非叶子结点被标记为 P 结点或 Q 结点.

由于利用现有技术构建祖先基因组时,只能推测出祖先基因组的连续遗传区,不能确定祖先基因组的全序列,如何构建消逝物种的全基因组,已有大量的工作致力于此^[3]. 文献[3]基于多重断点图的概念,提出了算法 MGRA,利用该算法重构祖先基因组. 文献[4]提出了基于现代物种区间邻接关系预测祖先物种区间顺序的算法.

在本文中,利用 PQ-树来存储祖先基因组^[5]. 在生物学中,进化树用来表示物种之间的进化关系. 生物分类学家和进化论者根据各类生物间的亲缘关系的远近,把各类生物安置在有分枝的树状图表上,简明地表示生物的进化历程和亲缘关系. 当构建生物进化树时,叶子结点表示现存物种,其基因组用排列表示,内部结点为祖先物种,其基因组用 PQ-树表示,祖先基因组跟现存物种基因组的组成元素相同^[6]. 为了确定物种间的进化关系,需要确定 PQ-树可以产生的排列与已知排列之间的距离. 文献[7]证明了 2 棵 PQ-树的最小断点距离问题是 NP-完全的,本文将证明一棵 PQ-树与已知排列的最小断点距离问题也是 NP-完全的.

本文中,以断点距离为标准,研究了 p 排列 PQ-树断点中心问题^[8]. p 排列 PQ-树断点中心问题描述如下:给定一棵 PQ-树 T 和 p 个排列 $s_1, s_2, \dots, s_p$,要求从 T 生成排列 s ,使得 s 与给定的 p 个排列 $s_1, s_2, \dots, s_p$ 的断点距离之和最小. 如果 PQ-树 T 可以生成所有可能的排列,则 p 排列 PQ-树断点中心问题跟传统的断点中心问题就是等价的. 现在已知

3 个排列的断点中心问题是 NP-完全的,从而,当 $p \geq 3$ 时, p 排列 PQ-树断点中心问题是 NP-完全的(这是因为可以构造一棵只有 1 个 P 结点的 PQ-树,所有元素作为该树的叶子结点,这样的 PQ-树可以产生元素的任意排列^[9]).

本文主要工作如下:

- 1) 对于 p 排列 PQ-树断点中心问题,证明当 $p \geq 2$ 时,该问题是 NP-完全的.
- 2) 针对 1-PQ-树断点中心问题,提出时间复杂度为 $O(3^K n)$ 的参数化算法.

1 相关概念

1.1 断点距离

定义 1. 邻接. Σ 表示包含 n 个字符的字符表, s 是 Σ 上的所有元素构成的排列, s 中相邻 2 个元素 a 和 b 构成一个邻接 ab 或 ba . 我们认为邻接 ab 与 ba 是等价的.

定义 2. 公共邻接与断点. 给定 2 个排列 s_1 与 s_2 ,如果 ab 或 ba 在 s_1 与 s_2 中都出现,则称 ab 是公共邻接;如果 ab 在 s_1 中出现,但是 ab 和 ba 在 s_2 中都不出现,则称 ab 是 s_1 的一个断点;如果 ab 在 s_2 中出现,但是 ab 和 ba 在 s_1 中都不出现,则称 ab 是 s_2 的一个断点.

定义 3. 端点与公共端点. 排列 2 端的 2 个元素称为端点,如果元素 a 既是排列 s_1 的端点,又是排列 s_2 的端点,则称 a 是公共端点.

定义 4. 断点距离. 令 $a(s_1, s_2)$ 表示排列 s_1 与 s_2 的公共邻接的数目, $t(s_1, s_2)$ 表示排列 s_1 与 s_2 的公共端点的数目,则排列 s_1 与 s_2 的断点距离定义为

$$d_b(s_1, s_2) = n - a(s_1, s_2) - t(s_1, s_2)/2. \quad (1)$$

在排列 s_1 与 s_2 的两端分别加上元素 0 和 $n+1$,可以得到规整后的断点距离:

$$d_b(s_1, s_2) = n + 1 - a(s_1, s_2). \quad (2)$$

1.2 PQ-树

PQ-树是字符表 $\Sigma = \{1, 2, \dots, n\}$ 上的一棵平面有根树. PQ-树有 3 类结点:叶子结点、 P 结点、 Q 结点. 字符表中的每个元素用一个叶子结点来表示,任

意 2 个子结点表示的元素不同. 非叶子结点被标记为 P 结点或 Q 结点, 1 个 P 结点至少有 2 个子结点, 1 个 Q 结点至少有 3 个子结点. P 结点的子结点的顺序可以随意排列, Q 结点的子结点的顺序只能是正序或者反序. PQ-树只有 2 种可行操作, PQ-树可以通过可行操作实现等价变换:

- 1) 随意重排 P 结点的子结点顺序.
- 2) 将 Q 结点的子结点顺序翻转.

当且仅当存在一系列可行操作能将一棵 PQ-树转化为另一棵 PQ-树, 则称这 2 棵 PQ-树是等价的.

按照前序关系得到的 PQ-树的叶子结点的排列称为 PQ-树的标签, 与某棵 PQ-树 T 等价的所有 PQ-树的标签称为该 PQ-树产生的排列的集合, 记为 $P(T)$.

1.3 问题定义

p 排列 PQ-树断点中心问题 (p -minimum breakpoint median from PQ-trees, p -MBM-PQ).

实例: 字符表 Σ 上的 PQ-树 T 和 p 个排列 $s_1, s_2, \dots, s_p, |\Sigma|=n$, 正整数 K .

询问: 是否可以从 T 产生排列 s , 使得 s 与 p 个排列 $s_1, s_2, \dots, s_p$ 断点距离之和满足 $\sum_{i=1}^p d_b(s, s_i) \leq K$.

2 2-MBM-PQ 是 NP-完全的

本节通过将三正则平面无桥连通图上的哈密顿路径问题规约到 2-MBM-PQ, 证明 2-MBM-PQ 问题是 NP-完全的. 已知三正则平面无桥连通图上的哈密顿路径问题是 NP-完全的^[10].

在文献[6]中, 证明了 2 棵 PQ-树的断点距离问题是 NP-完全的. 由于一棵 PQ-树可以产生排列可能有若干个, 因此, 本文中研究的问题比文献[6]中研究的问题要简单的多, 但本文中证明这个问题仍然是 NP-完全的.

各顶点的度均相同的无向简单图称为正则图. 各顶点度均为 K 的正则图称为 K 正则图. 对于图 G 中的边 e , 若删除边 e 使得图 G 的连通分支数增加, 则边 e 是 G 的桥.

规约的中心思想如下: 对应于一个给定的三正则平面无桥连通图 C , 构造 2-MBM-PQ 问题的 PQ-树 T 及 2 个排列 s_1, s_2 , 其中 PQ-树 T 的叶子结点对应于图 C 的所有顶点, 2 个排列 s_1 与 s_2 对应于图 C 的所有边. 因此图 C 中有哈密顿路径当且仅当可以从 PQ-树 T 生成排列 s , 使得 s 与 s_1, s_2 的公共邻

接数之和为 $n-1$ (其中每个邻接对应于图 C 中的一条边).

给定一个三正则平面无桥连通图 $C=(C(V), C(E))$, 首先构造 2-MBM-PQ 问题的 PQ-树 T : T 有 2 层, 根结点是 P 结点, 它有 2 个子结点: 一个是 P 结点, 存储 $C(V)$ 的所有顶点; 一个是 Q 结点, 存储填充元素 (填充元素是为了间隔非邻接顶点而添加的不同于图 C 中顶点的元素).

然后, 构造 2 个排列 s_1 与 s_2, s_1, s_2 包含 $C(V)$ 中的所有顶点以及填充元素, 对于图 C 中的每对顶点 $v_i v_j$, 如果 $(v_i, v_j) \in C(E)$, 那么 $v_i v_j$ 或 $v_j v_i$ 是 s_1 或 s_2 中的一个邻接, 但不是 s_1 与 s_2 的公共邻接, 如果 $(v_i, v_j) \notin C(E)$, 则用其他顶点或者填充元素将 v_i 与 v_j 间隔开, 使其在 s_1 与 s_2 中不能构成邻接.

引理 1^[11]. 任何三正则平面无桥连通图都有完美匹配, 其完美匹配可以在多项式时间内计算得到.

下面, 将描述构造排列 s_1, s_2 的具体步骤, 该构造过程可以在多项式时间内完成.

步骤 1. 计算三正则平面无桥连通图 $C=(C(V), C(E))$ 的完美匹配 M , 从图 C 中删除 M 中的边, 得到图 $C'=(C(V), C(E)-M)$, 则 C' 是由不相交的环路构成.

步骤 2. 从任意顶点 $v_i (v_i \in C'(V))$ 开始, 遍历 C' 中 v_i 所在环路上的所有顶点, 得到一条路径 $P=\langle v_i, \dots, v_j \rangle$, 对于顶点 v_k (其中 $(v_j, v_k) \in M$), 如果 v_k 已被某条路径遍历过, 则重新选择一个顶点作为开始顶点, 继续遍历 C' 中的环路; 否则, 如果 v_k 还没被任何路径遍历过, 那么将路径 P 通过边 (v_j, v_k) 扩展到其他环路. 循环迭代这个过程直到所有的环路都被遍历. 最后, 利用 M 中的边连接得到的这些路径, 直到没有两条路径可以被连接, 从而构成最长的没有环路的路径, 并把最终得到的路径集合放入 s_1 中, 将得到的路径集合记为 P' .

步骤 3. 将 C 中剩余的没被遍历的所有路径 (不包含在路径集合 P' 中) 放入 s_2 .

引理 2. $C(V)$ 中的每个顶点在 s_1 中恰好出现 1 次.

证明. 由于图 C 是一个三正则平面无桥连通图, M 是图 C 的完美匹配, 那么 $C'=(C(V), C(E)-M)$ 是由不相交的环路构成. 步骤 2 中, 遍历图 C' 中所有环路得到的路径集合放入 s_1 中, 其中, 每个环路遍历且只遍历 1 次, 所以, $C(V)$ 中所有顶点在 s_1 中恰好出现 1 次. 证毕.

从图 C 中删除路径集合 P' 中的边, 得到图 C'' .

引理 3. 图 C'' 中没有环.

证明. 对于图 C'' , 可以证明结论: 图 C'' 中每个度为 2 的顶点都有 1 个度为 1 的邻居顶点, 若该结论成立, 则图 C'' 中没有环.

注意到 C'' 中每个度为 2 的顶点必定是 P' 中某条路径的端点. 假设: 1 个度为 2 的顶点 v_r 连接到 2 个度为 2 的顶点 v_s, v_t , 则 v_s, v_t 中至少有 1 个顶点跟 v_r 不在同一条路径上, 而根据上述构造排列 s_1 与 s_2 的过程, 这样的 2 条路径在步骤 2 中应该被连接起来, 这与每个度为 2 的顶点必定是 P' 中某条路径的端点相矛盾, 所以图 C'' 中每个度为 2 的顶点都有 1 个度为 1 的邻居顶点, 所以图 C'' 中没有环. 证毕.

引理 4. $C(V)$ 中的每个顶点在 s_2 中恰好出现 1 次.

证明. 由于 $C(V)$ 中的每个顶点在 s_1 中恰好出现 1 次(引理 2), 所以 C'' 中的顶点度数为 1 或 2. 又由于 C'' 中没有环(引理 3), 所以放入 s_2 中的都是不相交的路径, 所以 $C(V)$ 中的每个顶点在 s_2 中恰好出现 1 次. 证毕.

引理 5. 如果边 $(v_i, v_j) \in C(E)$, 则在 s_1 或 s_2 中, 存在邻接 $v_i v_j$ 或 $v_j v_i$, 但 $v_i v_j$ 或 $v_j v_i$ 不是 s_1 与 s_2 的公共邻接.

证明. 根据 s_1, s_2 的构造过程, 对于图 C 中的每条边 (v_i, v_j) , 存在且只存在 2 种情况中的 1 种:

1) (v_i, v_j) 或 (v_j, v_i) 在步骤 2 中被放入 s_1 中, 则在 s_1 中存在邻接 $v_i v_j$ 或 $v_j v_i$.

2) (v_i, v_j) 或 (v_j, v_i) 在步骤 3 中被放入 s_2 中, 则在 s_2 中存在邻接 $v_i v_j$ 或 $v_j v_i$.

(v_i, v_j) 或 (v_j, v_i) 或者在步骤 2 中被放入 s_1 中, 或者在步骤 3 中被放入 s_2 中, 但不会同时放入 s_1 与 s_2 中, 所以邻接 $v_i v_j$ 或 $v_j v_i$ 只能在 s_1 或 s_2 中存在, 但不是 s_1 与 s_2 的公共邻接, 所以该引理成立. 证毕.

引理 6. 如果边 $(v_i, v_j) \notin C(E)$, 则在 s_1 和 s_2 中, 不存在邻接 $v_i v_j$ 或 $v_j v_i$.

证明. 根据 s_1, s_2 的构造过程, 放入 s_1, s_2 中的只能是图 C 中的边, 所以如果 $(v_i, v_j) \notin C(E)$, 则在 s_1, s_2 中不存在邻接 $v_i v_j$ 或 $v_j v_i$, 所以该引理成立. 证毕.

下面, 我们完成三正则平面无桥连通图上的哈密顿路径问题到 2-MBM-PQ 问题的规约, 证明 2-MBM-PQ 问题是 NP-完全的:

1) 利用不同的填充元素将 s_1 以及 s_2 中的路径分别间隔开.

2) 构建 4 个子排列 $x_1 x_2 x_3 x_4, x_2 x_4 x_1 x_3,$

$x_5 x_6 x_7 x_8, x_6 x_8 x_5 x_7$, 分别把 $x_1 x_2 x_3 x_4$ 与 $x_5 x_6 x_7 x_8$ 放在排列 s_1 与 s_2 的两端; 把 $x_2 x_4 x_1 x_3$ 放在 PQ-树 T 的子 Q 结点的最左边, 把 $x_6 x_8 x_5 x_7$ 放在 PQ-树 T 的子 Q 结点的最右边.

3) 把 s_2 中的所有填充元素放在 s_1 中的 x_1 的左边, 把 s_1 中的所有填充元素放在 s_2 中 x_8 的右边; 把所有既在 s_1 中又在 s_2 中的填充元素放在 PQ-树 T 的子 Q 结点的 $x_2 x_4 x_1 x_3$ 与 $x_6 x_8 x_5 x_7$ 之间, 并且保证在 s_1 或 s_2 中邻接的两个填充元素在 Q 结点中不能相邻.

引理 7. 三正则平面无桥连通图 C 存在一条哈密顿路径, 当且仅当 PQ-树 T 可以生成排列 s , 使得 s 与 s_1 的公共邻接数加上 s 与 s_2 的公共邻接数之和为 $n-1$.

证明. 显然, 没有公共邻接涉及填充元素.

充分性. 如果三正则平面无桥连通图 C 有一条哈密顿路径, PQ-树可以产生一个排列 s , 排列 s 中的元素顺序与哈密顿路径中顶点的顺序一致, 那么排列 s 中每一个邻接对应图 C 的哈密顿路径中的一条边. 由于图 C 中的每条边被放入且只放入排列 s_1 或 s_2 中的 1 个, 所以排列 s 与 s_1, s_2 的公共邻接数之和为 $n-1$.

必要性. 如果 T 可以生成排列 s 使得 s 与 s_1 的公共邻接数加上 s 与 s_2 的公共邻接数之和为 $n-1$, 由于 T 最多可以生成 $n-1$ 个公共邻接, 每个邻接对应于 C 中的一条边, 所以, 所有的这些边构成一条哈密顿路径. 证毕.

现在举例说明这个规约. 给定一个三正则平面无桥连通图 C , 如图 1(a) 所示, 图 C 的完美匹配如图 1(b) 所示:

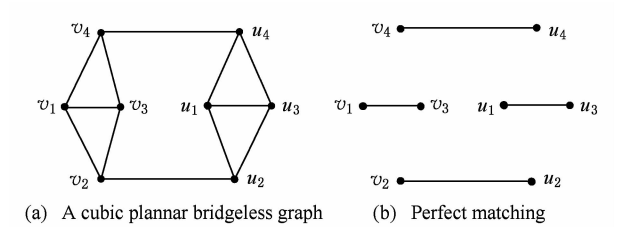


Fig. 1 A cubic planar bridgeless graph and its perfect matching.

图 1 三正则平面无桥连通图及其完美匹配

根据排列 s_1, s_2 的构造过程: 从顶点 v_1 开始, 遍历环路 $\langle v_1 v_2 v_3 v_4 v_1 \rangle$ 上所有顶点, 得到路径 $P = \langle v_1 v_2 v_3 v_4 \rangle$, 对于顶点 u_4 , 由于 $(v_4, u_4) \in M$ 且 u_4 还没被遍历过, 将路径 P 通过边 (v_4, u_4) 扩展到环路 $\langle u_4 u_1 u_2 u_3 u_4 \rangle$, 得到 $P' = \langle v_1 v_2 v_3 v_4 u_4 u_1 u_2 u_3 \rangle$, 将路

径 P' 放入 s_1 , 将剩余的没被遍历的路径集合 $\{v_4 v_1 v_3, v_2 u_2, u_4 u_3 u_1\}$ 放入 s_2 , 添加填充元素 #、\$ 将 s_2 中的路径间隔开, 在 s_1, s_2 中添加子排列 $x_1 x_2 x_3 x_4, x_5 x_6 x_7 x_8$, 在 s_1 中添加填充元素 #、\$, 最终得到:

$$s_1 = \# \$ x_1 x_2 x_3 x_4 v_1 v_2 v_3 v_4 u_4 u_1 u_2 u_3 x_5 x_6 x_7 x_8,$$
$$s_2 = x_1 x_2 x_3 x_4 v_4 v_1 v_3 \# v_2 u_2 \$ u_4 u_3 u_1 x_5 x_6 x_7 x_8.$$

字符 #、\$ 是填充元素. PQ-树 T 的根结点是 P 结点, 根结点有 2 个子结点, 1 个是 P 结点, 1 个是 Q 结点. P 结点的叶子结点为: $\{v_1, v_2, v_3, v_4, u_1, u_2, u_3, u_4\}$, Q 结点的叶子结点为: $\langle x_2, x_4, x_1, \#, x_3, \$, x_6, x_8, x_5, x_7 \rangle$.

重排 P 结点的叶子结点可以得到一个序列: $[v_1, v_2, v_3, v_4, u_4, u_1, u_2, u_3]$. 该序列对应 C 中的一条哈密顿路径, 在得到的序列基础上添加填充元素 #、\$, 可以得到一个排列 s , 使得 s 与 s_1, s_2 的公共邻接数之和为 7.

引理 8. 2-MBM-PQ 问题是 NP-完全的.

证明. 显然, 2-MBM-PQ 是 NP 的. 根据引理 5, 可以看到本文的规约可以在多项式时间内完成, 所以 2-MBM-PQ 是 NP-完全的. 证毕.

由于 $p \geq 3$ 时, p 排列 PQ-树断点中心问题是 NP-完全的, 综合引理 6, 可以得出如下结论:

推论 1. 当 $p \geq 2$ 时, p -MBM-PQ 问题是 NP-完全的.

3 1-MBM-PQ 问题的参数化算法

本节提出一个参数化算法解决 1-MBM-PQ 问题, 该算法的参数是问题最优解的断点距离.

判定问题 Π 、参数 p 、存在算法可以在 $O(f(p)n^c)$ 时间内解决该问题, 其中 f 是只与 p 有关的函数, n 是问题输入规模, c 是与 p 无关的常数^[12]. 这样的问题称为参数化问题, 可以在 $O(f(p)n^c)$ 时间内解决该问题的算法称为参数化算法.

3.1 PQ-树的图表示

首先介绍怎样图形化表示 PQ-树. 给定 1-MBM-PQ 问题的 PQ-树 T 与排列 s_1 , PQ-树 T 的每个结点对应图 G 的一个顶点. 对应于叶子结点的顶点称为元素, 对应于 P 结点的顶点称为超 P 顶点, 对应于 Q 结点的顶点称为超 Q 顶点. 2 个顶点之间存在边, 当且仅当它们是某个 Q 结点的相邻子结点. 图 G 的边称为黑边.

根据 T 的层次关系, 在图 G 中增加顶点嵌入信息. 如果 PQ-树中一个结点 X 是另一个结点 Y 的子

结点, 则在图 G 中, X 对应的顶点嵌入到 Y 对应的顶点中, 如图 2 所示. 因此, 当且仅当 X 是 Z 的子孙结点时, X 对应的顶点嵌入到 Z 对应的顶点中.

将排列 s_1 的信息增加到图 G 中. 对于 s_1 中每个邻接 xy , 如果图 G 中不存在黑边 (x, y) , 则在图 G 中增加一条蓝边 (x, y) , 令得到的新的图为 G' , 如图 3 所示. 图 G' 中超顶点 X 的度定义为: 连接 X 内顶点与 X 外顶点的蓝边的数目.

可以看到 1-MBM-PQ 问题跟最小路径覆盖问题关系密切. 最小路径覆盖问题描述如下: 给定一个图, 寻找数目最少的路径, 使之覆盖图的所有顶点, 并且任何一个顶点有且只有一条路径与之关联^[14].

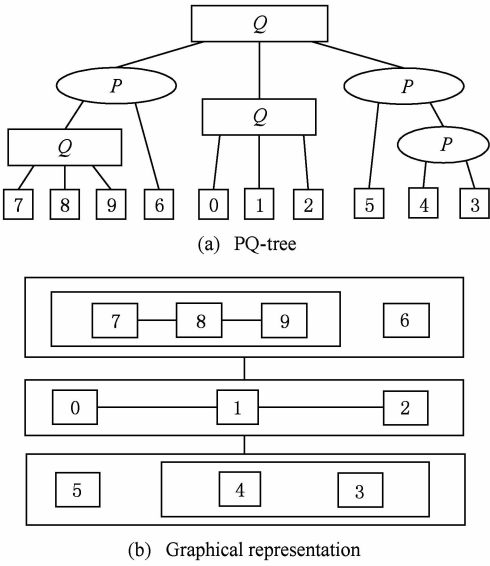


Fig. 2 A PQ-tree and its graphical representation.
图 2 PQ-树及其图表示

3.2 1-MBM-PQ 问题的参数化算法

通过下面的引理来描述如何求解问题的最优解.

引理 9. 通过对图 G' 完成以下操作, 可以得到 1-MBM-PQ 问题的最优解:

- 1) 如果元素 x 是 Q 结点 Y 的中间子结点 (其中 Y 的所有子结点都是叶子结点), 则删除所有与 x 关联的蓝边.
- 2) 如果元素 x 的度数超过 2, 则至多保留 2 条与 x 关联的蓝边, 其余蓝边都删除.
- 3) 如果超顶点 X 的度数超过 2, 则至多保留 2 条与 X 内元素相关联的蓝边, 其余蓝边都删除.

证明. 这里只讨论第 3 种情况. 如果 X 度数超过 2, 为了将 X 嵌入某条路径, 则必须删除与 X 相关联的多余边. 如果连接到 X 内元素的蓝边超过 2 条, 则不可能将 X 嵌入某条路径. 证毕.

在执行了引理 9 的步骤 1 操作之后,令 r 表示图 G' 中超顶点的最大度数,如果 $r \leq 2$,则问题是平凡的,所以假定 $r \geq 3$. 该 FPT 算法利用有限搜索树的原理,处理度数超过 2 的超顶点. 令 K 表示 1-MBM-PQ 问题的最优解的断点距离, $f(K)$ 表示搜索树的叶子结点数目,可以得到如下递推关系:

$$f(K) = \begin{cases} 0, & K = 0; \\ 1, & K = 1; \\ \leq \binom{r}{r-2} f(K-r+2), & K > 1. \end{cases} \tag{3}$$

主要递推关系式可以简化为

$$f(K) \leq \binom{r}{2} f(K-r+2) = \frac{r(r-1)}{2} f(K-r+2). \tag{4}$$

利用生成函数求解递推关系的方法,对式(4)进一步化简得到:

$$f(K) \leq \left(\frac{r(r-1)}{2}\right)^{\frac{K}{r-2}}. \tag{5}$$

可以看到,当 $r=3$ 时,取得最大值,则:

$$f(K) \leq 3f(K-1), \tag{6}$$

因此 $f(K) \leq 3^K$.

算法描述如下:

算法 1. *One-MBP-PQ*(T, s_1).

输入:PQ-树 T 、排列 s_1 、整数 K ;

输出:由 T 生成的排列 s , s 与 s_1 之间的断点距离不超过 K .

步骤 1. 根据 T , 计算得到图 G ;

步骤 2. 根据 T 的层次关系,在图 G 中增加嵌入信息;

步骤 3. 对应于 s_1 中的所有邻接,在图 G 中添加蓝边,得到图 G' ;

步骤 4. 令 U 表示从 G' 中删除的蓝边的集合;

步骤 5. while $|U| \leq K$

步骤 5.1. 如果元素 x 是 Q 结点 Y 的中间的子结点(其中 Y 的所有子结点都是叶子结点),则删除所有与 x 关联的蓝边,并把删除的蓝边放入 U ;

步骤 5.2. 如果元素 x 的度数超过 2,保留 2 条与 x 关联的蓝边,删除其余蓝边,并把删除的蓝边放入 U ;

步骤 5.3. 如果超顶点 X 的度数超过 2,保留 2 条与 X 内元素关联的蓝边,删除其余蓝边,并把删

除的蓝边放入 U ;

步骤 6. 如果最终得到的图由路径组成,则返回排列 s , 否则,返回“无解”.

当从图 G' 中删除 K 条蓝边后,要做的就剩下验证所得的图是否由路径组成,如果在删除 K 条蓝边后,所有顶点度数都不超过 2,那么验证所得的图是否由路径组成,就可以转化为验证图中是否有环路存在. 在删除 K 条蓝边后,如果没有找到可行解,则返回“无解”. 这可以在 $O(n)$ 的时间内完成. 因此,一旦计算得到 G' ,可以利用有限搜索树^[15]原理实现一个时间复杂度为 $O(3^k n)$ 的算法.

在图 3 中,针对上述算法举一个简单的例子. 图 3(a)中,给定 PQ-树 T ,给定排列 $s_1 = "0123456789"$, 图 3(b)中,图 G' 是计算所得(图中虚线边代表算法中添加的蓝边). 该实例最优解的断点距离 $K=1$. 根据上述算法,需要在图 G' 中删除一条蓝边,例如,对于超顶点“105”,其度数为 3,所以至少删除(2,1)(5,6)(4,5)的一条边,为了使最终得到的图由路径组成,删除蓝边(4,5),此时可以得到最优解排列 $s = "4321056789"$,且 s 与 s_1 之间恰好有一个断点.

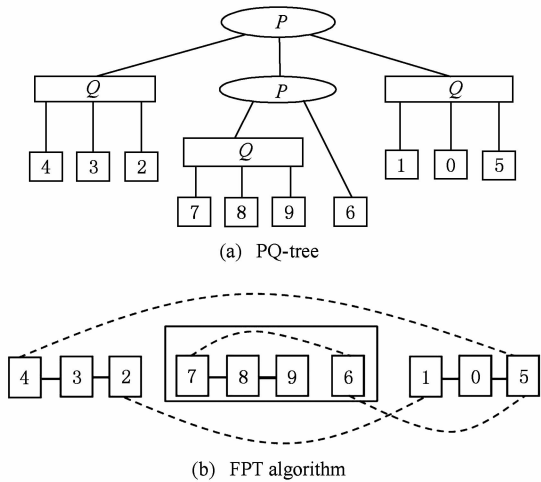


Fig. 3 An example for the FPT algorithm.

图 3 参数化算法示例

根据上述算法,可以得到结论:1-MBM-PQ 问题存在时间复杂度为 $O(3^k n)$ 的算法,其中 n 是元素个数, K 是最优解的断点距离.

4 结束语

本文主要分析讨论了 p -PQ-树断点中心问题,在介绍了 PQ-树、断点距离等相关概念后,证明了当 $p \geq 2$ 时, p -PQ-树断点中心问题是 NP-完全的,然后

针对 1-PQ-树断点中心问题,提出了时间复杂度为 $O(3^k n)$ 的参数化算法.

目前,1-MBM-PQ 问题的复杂性还是未知,设计 PQ-树断点距离问题的有效算法也值得继续研究.

参 考 文 献

[1] Booth K S, Lueker G S. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms [J]. Journal of Computer and System Sciences, 1976, 13(3): 335-379

[2] Ou Liancheng. Testing for graph planarity using PQ-tree algorithms [J]. Computer Engineering and Science, 1990, 12 (2): 44-50 (in Chinese)
(欧连成. 用 PQ-树算法判定图的平面性[J]. 计算机工程与科学, 1990, 12(2): 44-50)

[3] Alekseyev M A, Pevzner P A. Breakpoint graphs and ancestral genome reconstructions [J]. Genome Research, 2009, 19(5): 943-957

[4] Ma J, Zhang L, Suh B B, et al. Reconstructing contiguous regions of an ancestral genome [J]. Genome Research, 2006, 16(12): 1557-1565

[5] Rascol V L, Pontarotti P, Levasseur A. Ancestral animal genomes reconstruction [J]. Current Opinion in Immunology, 2007, 19(5): 542-546

[6] Chauve C, Tannier E. A methodological framework for the reconstruction of contiguous regions of ancestral genomes and its application to mammalian genomes [J]. PLoS Computational Biology, 2008, 4(11): e1000234

[7] Jiang H, Chauve C, Zhu B. Breakpoint distance and PQ-trees [C] //Proc of the 21st Annual Symp on Combinatorial Pattern Matching. Berlin: Springer, 2010: 112-124

[8] Watterson G A, Ewens W J, Hall T E, et al. The chromosome inversion problem [J]. Journal of Theoretical Biology, 1982, 99(1): 1-7

[9] Sankoff D, Blanchette M. Multiple genome rearrangement and breakpoint phylogeny [J]. Journal of Computational Biology, 1998, 5(3): 555-570

[10] Diwan A A. Disconnected 2-factors in planar cubic bridgeless graphs [J]. Journal of Combinatorial Theory, 2002, 84(2): 249-259

[11] Biedl T C, Bose P, Demaine E D, et al. Efficient algorithms for petersen's matching theorem [J]. Journal of Algorithms, 2001, 38(1): 110-34

[12] Downey R G, Fellows M R. Parameterized Complexity [M]. Berlin: Springer, 1999

[13] Gramm J, Niedermeier R. Breakpoint medians and breakpoint phylogenies: A fixed-parameter approach [J]. Bioinformatics, 2002, 18(Suppl 2): S128-S139

[14] Bergeron A, Blanchette M, Chateau A, et al. Reconstructing ancestral gene orders using conserved intervals [C] // Proc of the 4th Int Workshop on Algorithms in Bioinformatics. Berlin: Springer, 2004: 14-25

[15] Marron M, Swenson K M, Moret B M E. Genomic distances under deletions and insertions [J]. Theoretical Computer Science, 2004, 325(3): 347-360



Liu Peixia, born in 1990. Received her master degree form School of Computer Science and Technology, Shandong University in 2015. Her main research interests include algorithm design and analysis.



Jiang Haitao, born in 1982. Associate professor in the School of Computer Science and Technology, Shandong University. Member of China Computer Federation. His main research interests include algorithm design and analysis.



Zhu Daming, born in 1963. Professor in the School of Computer Science and Technology, Shandong University. Member of China Computer Federation. His research interests include algorithms, bioinformatics, and computational biology.