

求解图同构的判定算法

侯爱民

HOU Aimin

东莞理工学院 计算机科学与技术系, 广东 东莞 523808

Department of Computer Science and Technology, Dongguan University of Technology, Dongguan, Guangdong 523808, China

HOU Aimin. Determining algorithm for solving graph isomorphism. *Computer Engineering and Applications*, 2011, 47(16): 52-57.

Abstract: How to determine the isomorphism of graphs is a difficult problem of graph theory, which has not been completely solved so far. From the idea of Ulam conjecture concerning graph isomorphism, a new necessary and sufficient condition of graph isomorphism is presented, which is stated as following: two graphs are isomorphic if and only if their subgraphs are isomorphic and the new vertices as well as their adjacency edges are corresponding. With the help of the technique for unlimitedly generating the corresponding pairs of vertices, this condition is proved with the method of reduction to absurdity. An algorithm for determining graph isomorphism is designed and implemented, whose correctness and validity are tested and verified with some concrete examples.

Key words: subgraph isomorphism; graph isomorphism; technique for unlimitedly generating the corresponding pairs of vertices; determining algorithm

摘要: 图同构的判定性问题是图论理论中的一个难题, 至今没有得到彻底解决。受 Ulam 猜想的启发, 提出了一个新的判定图同构的充分必要条件: 在子图同构的前提下, 根据新增顶点及相应关联边的关系, 利用子图同构函数, 判断父图同构的充分必要条件。基于具有同构关系的对应点无限衍生技术, 采用反证法证明了这个充分必要条件的成立。设计并实现了图同构的一个判定算法, 通过实例验证了算法的正确性和有效性。

关键词: 子图同构; 图同构; 对应点无限衍生技术; 判定算法

DOI: 10.3778/j.issn.1002-8331.2011.16.017 文章编号: 1002-8331(2011)16-0052-06 文献标识码: A 中图分类号: O157.5

1 引言

在研究图论的问题中, 经常会遇到判断两个图是否同构, 即从顶点和边的拓扑图结构上来看, 两个图是否有可能以同样的方式画出。换句话说, 当两个图同构时, 两个图的顶点之间具有保持相邻关系的一一对应。不幸的是, 判断两个图是否同构是一件困难的事情。

Ulam 曾经提出过一个判定图同构的猜想, 也称为图的重构猜想: 若存在双射函数 $f: V(G) \rightarrow V(H)$, 使得对任意的 $v \in V(G)$, 有 $G-v \cong H-f(v)$, 则称 H 是 G 的一个重构。重构问题也是图论中研究的经典问题之一, 这个猜想至今也没有解决, 但是, 围绕这个猜想已有若干论文发表。文献[1]研究了图 G 和 H 作为树出现时 Ulam 猜想的正确性。文献[2-4]扩展了 Kelly^[1]的研究成果, 提出了若干个判定树 G 和 H 同构的充分条件: 所有的最大子树同构; 子树 $T-v$ 同构, 这里 v 是一个外围顶点; 非同构的最大子树不存在。文献[5]研究了非对称树的可重构性。文献[6-9]研究了某些图的重构和边重构问题及参数的重构性。文献[10]研究了边型带权核子图的边可重构性。这些成果对研究图重构问题具有积极的意义。

本文提出了一个新技术, 即具有同构关系的对应点无限

衍生技术。然后, 采用反证法证明了在子图同构的前提下, 判断父图同构的充分必要条件: 同构的两个图, 各自增加一个顶点和与该顶点关联的若干条边, 形成了两个新图。(1) 这两个新图同构, 当且仅当新顶点的新关联边所关联的顶点在原图(即子图)中保持同构对应关系。(2) 这两个新图不同构, 当且仅当新顶点的新关联边所关联的顶点在原图(即子图)中不保持同构对应关系。基于这个充分必要条件, 设计并实现了图同构的一个判定算法, 并用具体的实例进行了算法正确性和有效性的验证。

2 基于子图同构的父图同构

定义 2.1 设两个有限图 G 和 H 同构, 同构函数为 f 。对于任意的 $u \in V(G)$, $v \in V(H)$, 如果 $v=f(u)$, 则称顶点 u 和 v 有对应关系, 记为 $u \leftrightarrow v$ 。否则称顶点 u 和 v 没有对应关系, 记为 $(u \nleftrightarrow v)$ 。如果有向图, 则邻接点的函数关系必须保持相同的有向性。

定义 2.2 设两个有限图 G 和 H 同构, 同构函数为 f 。对于任意的 $e_G=(u_i, u_j) \in E(G)$, $e_H=(v_i, v_j) \in E(H)$, 如果 $v_i=f(u_i)$ 且 $v_j=f(u_j)$ 或者 $v_j=f(u_i)$ 且 $v_i=f(u_j)$, 则称边 e_G 和 e_H 有对应关

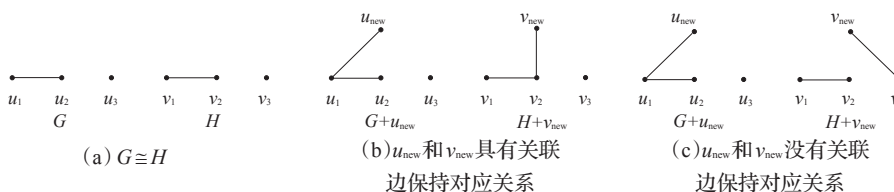


图1 同构图增加新顶点的情况

系,记为 $e_G \leftrightarrow e_H$ 。否则称边 e_G 和 e_H 没有对应关系,记为 $(e_G \nleftrightarrow e_H)$ 。如果是有向图,则边的函数关系必须保持相同的有向性。

定义 2.3 设两个有限图 G 和 H 同构,同构函数为 f 。 G 增加新顶点 u_{new} 及若干条与 u_{new} 关联的边; H 增加新顶点 v_{new} 及若干条与 v_{new} 关联的边。如果 u_{new} 的所有关联边所关联的顶点分别与 v_{new} 的所有关联边所关联的顶点,在同构函数 f 的作用下,保持同构关系,则称 u_{new} 和 v_{new} 具有新增的关联边保持对应关系。否则称 u_{new} 和 v_{new} 没有新增的关联边保持对应关系。如果是有向图,则新增边的函数关系必须保持相同的有向性。

根据定义 2.1 和 2.2 可知,如果 $G \cong H$,对于任意的 $u \in V(G)$, $v \in V(H)$,只要 $u \leftrightarrow v$,则有下列特性成立:(1) $\deg(u) = \deg(v)$ 。(2) u 的邻接点 $u_{i1}, u_{i2}, \dots, u_{ik}$,分别与 v 的邻接点 $v_{i1}, v_{i2}, \dots, v_{ik}$,各自保持对应关系,即 $u_{ij} \leftrightarrow v_{ij} (j=1, 2, \dots, k), v_{ij} = f(u_{ij})$ 。(3) u 的关联边分别与 v 的关联边各自保持对应关系,即 $(u, u_{ij}) \leftrightarrow (v, v_{ij}) (j=1, 2, \dots, k), v_{ij} = f(u_{ij})$ 。

根据定义 2.3 可知,如果 $G \cong H$,且 $u_i \leftrightarrow v_i, u_i \in V(G), v_i \in V(H) (i=1, 2, \dots, n)$ 。现在, G 增加新顶点 u_{new} ,且与 u_{i1}, u_{i2}, u_{i3} 邻接。 H 增加新顶点 v_{new} ,且与 v_{j1}, v_{j2}, v_{j3} 邻接。新顶点 u_{new} 和 v_{new} 没有新增的关联边保持对应关系,意味着 u_{i1}, u_{i2}, u_{i3} 决不与 v_{j1}, v_{j2}, v_{j3} 保持对应关系,即不存在任何的同构函数 $f (G \cong H \text{ 下的同构函数})$,使得:(1) $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j2}, u_{i3} \leftrightarrow v_{j3}$;(2) $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j3}, u_{i3} \leftrightarrow v_{j2}$;(3) $u_{i1} \leftrightarrow v_{j2}, u_{i2} \leftrightarrow v_{j1}, u_{i3} \leftrightarrow v_{j3}$;(4) $u_{i1} \leftrightarrow v_{j2}, u_{i2} \leftrightarrow v_{j3}, u_{i3} \leftrightarrow v_{j1}$;(5) $u_{i1} \leftrightarrow v_{j3}, u_{i2} \leftrightarrow v_{j1}, u_{i3} \leftrightarrow v_{j2}$;(6) $u_{i1} \leftrightarrow v_{j3}, u_{i2} \leftrightarrow v_{j2}, u_{i3} \leftrightarrow v_{j1}$ 。这六种情况中的任何一种情况出现。

例 1 考虑图 1 所描述的情况。在图 1(a)中, $G \cong H$,有两个同构函数:(1) $f_1: v_1 = f_1(u_1), v_2 = f_1(u_2), v_3 = f_1(u_3)$ 。(2) $f_2: v_1 = f_2(u_2), v_2 = f_2(u_1), v_3 = f_2(u_3)$ 。在图 1(b)中, G 新增顶点 u_{new} 及对应边 (u_{new}, u_1) , H 新增顶点 v_{new} 及对应边 (v_{new}, v_2) 。因为存在 $G \cong H$ 下的同构函数 f_2 ,使得 $u_1 \leftrightarrow v_2, (u_{\text{new}}, u_1) \leftrightarrow (v_{\text{new}}, v_2)$,所以 u_{new} 和 v_{new} 具有新增的关联边保持对应关系。在图 1(c)中, G 新增顶点 u_{new} 及对应边 (u_{new}, u_1) , H 新增顶点 v_{new} 及对应边 (v_{new}, v_3) 。因为不存在 $G \cong H$ 下的同构函数,使得 $u_1 \leftrightarrow v_3$ 成立。即 $G \cong H$ 下的所有同构函数 f_1 和 f_2 ,都不能使得 $u_1 \leftrightarrow v_3$ 成立,所以 u_{new} 和 v_{new} 没有新增的关联边保持对应关系,即 u_{new} 和 v_{new} 新增的关联边不保持对应关系。

定理 2.1 同构的两个图,各自增加一个顶点和与该顶点关联的若干条边。如果新顶点的所有关联边依然保持对应关系,则形成的新图必同构。

证明 不失一般性,假设 $G \cong H, |V(G)| = |V(H)| = n$,且 $G \cong H$ 的一个同构函数 f ,使得 $u_i \leftrightarrow v_i, u_i \in V(G), v_i \in V(H)$

($i=1, 2, \dots, n$)。 $u_i \leftrightarrow v_i$ 表示 u_i 与 v_i 保持对应关系。

现在, G 增加新节点 u_{new} ,且 u_{new} 与 $u_{i1}, u_{i2}, \dots, u_{ik}$ 邻接。 H 增加新顶点 v_{new} ,且 v_{new} 与 $v_{i1}, v_{i2}, \dots, v_{ik}$ 邻接。因为新顶点的关联边保持对应关系,所以必存在 $G \cong H$ 的一个同构函数 f ,使得 $u_{i1} \leftrightarrow v_{i1}, u_{i2} \leftrightarrow v_{i2}, \dots, u_{ik} \leftrightarrow v_{ik}$ 。且 $u_s \leftrightarrow v_s$ (这里 $u_s \in V(G) - \{u_{i1}, u_{i2}, \dots, u_{ik}\}, v_s \in V(H) - \{v_{i1}, v_{i2}, \dots, v_{ik}\}$)。

现在,令 $v_{\text{new}} = f(u_{\text{new}})$ 。很显然, G 和 H 的原有边在新图 $G + u_{\text{new}}$ 和 $H + v_{\text{new}}$ 中依然保持对应关系。即在 $G \cong H$ 下, $\forall (u_i, u_j) \in E(G)$,当且仅当 $(f(u_i), f(u_j)) \in E(H)$ 。这种关系在 $G + u_{\text{new}}$ 和 $H + v_{\text{new}}$ 中依然成立。

另一方面,在新图 $G + u_{\text{new}}$ 和 $H + v_{\text{new}}$ 中,新增加的边在原同构函数 f 的作用下,也保持对应关系,即 $\forall (u_{\text{new}}, u_{ij}) \in E(G + u_{\text{new}})$,当且仅当 $(f(u_{\text{new}}), f(u_{ij})) \in E(H + v_{\text{new}})$,这里 $u_{ij} \in \{u_{i1}, u_{i2}, \dots, u_{ik}\}$ 。

所以, $G \cong H$ 的同构函数 f ,补充一个顶点对应关系 $v_{\text{new}} = f(u_{\text{new}})$ 后,使得 $G + u_{\text{new}} \cong H + v_{\text{new}}$ 。证毕。

例 1 (继续) 在图 1(b)中, $G \cong H$ 的同构函数 f_2 ,也使得 $G + u_{\text{new}} \cong H + v_{\text{new}}$ 成立。这时,顶点对应关系如下: $v_{\text{new}} = f_2(u_{\text{new}}), v_1 = f_2(u_2), v_2 = f_2(u_1), v_3 = f_2(u_3)$ 。由此可见,在 $G \cong H$ 中的顶点对应关系,在 $G + u_{\text{new}} \cong H + v_{\text{new}}$ 中依然保持。此外,在 $G + u_{\text{new}} \cong H + v_{\text{new}}$ 中新增加了顶点对应关系 $v_{\text{new}} = f_2(u_{\text{new}})$ 。

另一方面, $G \cong H$ 中的同构函数 f_1 ,在 $G + u_{\text{new}} \cong H + v_{\text{new}}$ 中不再是同构函数。进一步地, $G + u_{\text{new}} \cong H + v_{\text{new}}$ 也可能产生一个新的同构函数,例如 $f_3: v_{\text{new}} = f_3(u_2), v_1 = f_3(u_{\text{new}}), v_2 = f_3(u_1), v_3 = f_3(u_3)$ 。

定理 2.2 同构的两个图,各自增加一个顶点和与该顶点关联的若干条边。如果新顶点至少存在一条关联边不保持对应关系,则形成的新图必不同构。

证明 假设原图为 G 和 $H, G \cong H$ 。 $V(G) = \{u_i | i=1, 2, \dots, n\}$, $V(H) = \{v_i | i=1, 2, \dots, n\}$ 。现在,图 G 新增顶点 u_{new} ,图 H 新增顶点 v_{new} 。

首先,无需考虑新增顶点 u_{new} 和 v_{new} 都不带边,或都带 n 条边的情况。即 $\deg(u_{\text{new}}) = \deg(v_{\text{new}}) = 0$,或 $\deg(u_{\text{new}}) = \deg(v_{\text{new}}) = n$ 。因为在这两种情况下,它们违反“新顶点至少存在一条关联边不保持对应关系”的前提条件。此时,当然有 $G + u_{\text{new}} \cong H + v_{\text{new}}$ 。

其次,考虑 u_{new} 和 v_{new} 新增不相同数目的关联边的情况。因为 $G \cong H$,所以 $E(G) = E(H)$ 。现在新增边数不相等,所以 $E(G + u_{\text{new}}) \neq E(H + v_{\text{new}})$,所以 $G + u_{\text{new}} \not\cong H + v_{\text{new}}$ 。因为新增关联边的数目不相等,所以满足“新顶点至少存在一条关联边不保持对应关系”的前提条件。

因此,只需要考虑 u_{new} 和 v_{new} 新增相同数目的关联边的情况。假设新增边数为 m ,则有 $1 \leq m \leq n-1$ 。

下面利用反证法进行证明。设新增关联边的数目相等,且满足“新顶点至少存在一条关联边不保持对应关系”的前提下,有 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 。

不失一般性,设在 $G+u_{\text{new}}$ 中, u_{new} 与 $u_{i1}, u_{i2}, \dots, u_{im}$ 邻接;在 $H+v_{\text{new}}$ 中, v_{new} 与 $v_{j1}, v_{j2}, \dots, v_{jm}$ 邻接。且不存在任何的 $G \cong H$ 的同构函数 f , 使得: (1) $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j2}, \dots, u_{im-1} \leftrightarrow v_{jm-1}, u_{im} \leftrightarrow v_{jm}$, 其余顶点一一对应。(2) $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j2}, \dots, u_{im-1} \leftrightarrow v_{jm}, u_{im} \leftrightarrow v_{jm-1}$, 其余顶点一一对应。……。(m!) $u_{i1} \leftrightarrow v_{jm}, u_{i2} \leftrightarrow v_{jm-1}, \dots, u_{im-1} \leftrightarrow v_{j2}, u_{im} \leftrightarrow v_{j1}$, 其余顶点一一对应。这 $m!$ 种情况中的任何一种成立。

下面分两种情况进行反证法证明。

情况1 在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 假设下,有 $u_{\text{new}} \leftrightarrow v_{\text{new}}$ 。

则必存在 $V(H)$ 中的一个顶点 v_t , 其邻接点为 $v_{t1}, v_{t2}, \dots, v_{tm}$, 使得在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 下, 有 $u_{\text{new}} \leftrightarrow v_t, u_{i1} \leftrightarrow v_{t1}, u_{i2} \leftrightarrow v_{t2}, \dots, u_{im} \leftrightarrow v_{tm}$ 。同时, 必存在 $V(G)$ 中的一个顶点 u_s , 其邻接点为 $u_{s1}, u_{s2}, \dots, u_{sm}$, 使得在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 下, 有 $u_s \leftrightarrow v_{\text{new}}, u_{s1} \leftrightarrow v_{j1}, u_{s2} \leftrightarrow v_{j2}, \dots, u_{sm} \leftrightarrow v_{jm}$ 。同时, 其余顶点一一对应。

现在, 删掉 u_{new} 和 v_{new} 后, 在 $G \cong H$ 中, 必有 $u_{i1} \leftrightarrow v_{t1}, u_{i2} \leftrightarrow v_{t2}, \dots, u_{im} \leftrightarrow v_{tm}; u_{s1} \leftrightarrow v_{j1}, u_{s2} \leftrightarrow v_{j2}, \dots, u_{sm} \leftrightarrow v_{jm}$ 。 $u_{i1}, u_{i2}, \dots, u_{im}$ 和 $v_{j1}, v_{j2}, \dots, v_{jm}$ 中至少有一个顶点(对)不保持对应关系。所以, 在 H 中必然存在另外 m 个顶点 $v'_{i1}, v'_{i2}, \dots, v'_{im}$, ($v'_{i1}, v'_{i2}, \dots, v'_{im}$ 中的某些顶点可以是 $v_{j1}, v_{j2}, \dots, v_{jm}$ 中的某些顶点; 但是, 不可能全部都是), 使得在 $G \cong H$ 中, 有 $u_{i1} \leftrightarrow v'_{i1}, u_{i2} \leftrightarrow v'_{i2}, \dots, u_{im} \leftrightarrow v'_{im}$ 。

因为在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 中, 有 $u_{i1} \leftrightarrow v'_{i1}, u_{i2} \leftrightarrow v'_{i2}, \dots, u_{im} \leftrightarrow v'_{im}$ 。所以在 G 中必存在另外 m 个顶点 $u'_{i1}, u'_{i2}, \dots, u'_{im}$, 使得在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 中, 有 $u'_{i1} \leftrightarrow v_{i1}, u'_{i2} \leftrightarrow v_{i2}, \dots, u'_{im} \leftrightarrow v_{im}$ 。

重复上述过程, 删掉 u_{new} 和 v_{new} 后, 在 $G \cong H$ 中, G 中有两组 m 个顶点 $u_{i1}, u_{i2}, \dots, u_{im}$ 和 $u'_{i1}, u'_{i2}, \dots, u'_{im}$, 分别与 H 中一组 m 个顶点 $v_{i1}, v_{i2}, \dots, v_{im}$ 保持一一对应关系。所以, 在 H 中必然存在另外一组 m 个顶点 $v''_{i1}, v''_{i2}, \dots, v''_{im}$, 分别与 G 中两组 m 个顶点 $u_{i1}, u_{i2}, \dots, u_{im}$ 和 $u'_{i1}, u'_{i2}, \dots, u'_{im}$ 中的一组保持一一对应关系, 以便使得具有同构关系的对应点保持平衡。

这样一来, 由于 $G+u_{\text{new}} \cong H+v_{\text{new}}$, 又导致在 G 中存在另外一组 m 个顶点 $u''_{i1}, u''_{i2}, \dots, u''_{im}$, 使得新图中具有同构关系的对应点保持平衡。 G 中多出的 m 个顶点, 又导致 H 中多出 m 个顶点, 反过来又导致 G 中再多出 m 个顶点。

如此这样, 一直进行下去, G 和 H 为了保持旧图关系中和新图关系中具有同构关系的对应点的平衡, 会导致出无穷多个顶点。因为 $|V(G)|=|V(H)|=n$ 是有限图, 所以导致出无穷多个顶点是不可能的事情。

所以, 在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 下, 假设 $u_{\text{new}} \leftrightarrow v_{\text{new}}$ 是不成立的。

情况2 在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 假设下, 有 $u_{\text{new}} \leftrightarrow v_{\text{new}}$ 。

因为 u_{new} 的邻接点为 $u_{i1}, u_{i2}, \dots, u_{im}$; v_{new} 的邻接点为 $v_{j1}, v_{j2}, \dots, v_{jm}$ 。现在, $G+u_{\text{new}} \cong H+v_{\text{new}}$, 且 $u_{\text{new}} \leftrightarrow v_{\text{new}}$ 。

所以, $u_{i1}, u_{i2}, \dots, u_{im}$ 必然与 $v_{j1}, v_{j2}, \dots, v_{jm}$ 存在一种一

一对应关系。不妨假设在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 下, $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j2}, \dots, u_{im} \leftrightarrow v_{jm}$, 其余顶点一一对应。

毫无疑问, 删掉 u_{new} 和 v_{new} 后, 在 G 和 H 中, 依然保持: $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j2}, \dots, u_{im} \leftrightarrow v_{jm}$, 其余顶点一一对应。所以, 一定存在 $G \cong H$ 的一个同构函数 f , 使得(不失一般性): $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j2}, \dots, u_{im} \leftrightarrow v_{jm}$, 其余顶点一一对应。显然, 这与前提“不存在任何的 $G \cong H$ 的同构函数 f , 使得(1) $u_{i1} \leftrightarrow v_{j1}, u_{i2} \leftrightarrow v_{j2}, \dots, u_{im} \leftrightarrow v_{jm}$, 其余顶点一一对应。这 $m!$ 种情况之一成立”产生矛盾。

所以, 在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 下, 假设 $u_{\text{new}} \leftrightarrow v_{\text{new}}$ 是不成立的。

综合上述两种情况, 在 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 的假设下, 既有 $u_{\text{new}} \leftrightarrow v_{\text{new}}$ 不成立, 又有 $u_{\text{new}} \leftrightarrow v_{\text{new}}$ 不成立, 显然自相矛盾。

所以, 假设 $G+u_{\text{new}} \cong H+v_{\text{new}}$ 不成立, 即形成的新图必不同构。证毕。

例1(继续) 在图1(c)中, $G \cong H$ 的同构函数 f_1 和 f_2 , 都不能使得新增边 (u_{new}, u_1) 和 (v_{new}, v_3) 保持对应关系, 所以, 必有 $G+u_{\text{new}} \not\cong H+v_{\text{new}}$ 。进一步地, 假设 u_{new} 新增两条边 (u_{new}, u_1) 和 (u_{new}, u_2) ; v_{new} 新增两条边 (v_{new}, v_1) 和 (v_{new}, v_3) 。显然, 新顶点至少存在一条关联边不保持对应关系, 即 (v_{new}, v_3) 不存在对应边; 所以, 必有 $G+u_{\text{new}} \not\cong H+v_{\text{new}}$ 。

推论2.1 同构的两个图, 各自增加一个顶点和与该顶点关联的若干条边, 形成了两个新图。

(1) 这两个新图同构, 当且仅当新顶点的所有关联边依然保持对应关系。

(2) 这两个新图不同构, 当且仅当新顶点至少存在一条关联边不保持对应关系。

证明 由定理2.1和定理2.2的否定命题, 可以得出推论2.1的(1)成立。

另一方面, 由定理2.2和定理2.1的否定命题, 可以得出推论2.1的(2)成立。证毕。

3 图同构的判定算法

众所周知, 如果两个图 G 和 H 同构, $V(G)=\{u_i | i=1, 2, \dots, n\}$, $V(H)=\{v_i | i=1, 2, \dots, n\}$, 同构函数为 f 。不失一般性, 假定 $v_i=f(u_i)$ 。那么, 必有: $G-u_1 \cong H-v_1, (G-u_1)-u_2 \cong (H-v_1)-v_2, \dots, (G-u_1-\dots-u_{n-2})-u_{n-1} \cong (H-v_1-\dots-v_{n-2})-v_{n-1}$ 。

反过来说, 如果 $G \cong H$ 为真, 那么, 根据定理2.1, 一定可以逆向证明上述一系列子图同构。即有: $(G-u_1-\dots-u_{n-2})-u_{n-1} \cong (H-v_1-\dots-v_{n-2})-v_{n-1}, (G-u_1-\dots-u_{n-3})-u_{n-2} \cong (H-v_1-\dots-v_{n-3})-v_{n-2}, \dots, (G-u_1)-u_2 \cong (H-v_1)-v_2, G-u_1 \cong H-v_1, G \cong H$ 。

在这个逆向证明过程中, $G \cong H$ 可以看作是 $(G-u_1)+u_1 \cong (H-v_1)+v_1$, 即子图 $G-u_1 \cong H-v_1$ 的前提下, 新增顶点 u_1 和 v_1 , 以及相应的关联边, 并且满足定理2.1的判定条件。以此类推, $G-u_1 \cong H-v_1$ 可以看作是 $((G-u_1)-u_2)+u_2 \cong ((H-v_1)-v_2)+v_2$ 。

由于两个顶点的无向图和有向图已经是唯一的, 所以, 可以先判定两个顶点的子图是否同构, 然后再判定三个顶点的子图是否同构。如此这样, 一直进行下去, 直至判定 n 个顶点的子图(即图 G 和 H)是否同构。

进一步分析,可以发现:如果 $G-u_1 \not\cong H-v_1$, 新增顶点和关联边后,却有可能导致 $G \cong H$ 。此时,定理2.1不能发挥作用。另一方面,如果真有 $G \cong H$, 且 $G-u_1 \not\cong H-v_1$; 那么,就一定存在 $v'_1 \in V(H)$, 使得 $G-u_1 \cong H-v'_1$ 。所以,在遇到这种情况时,需要不断回溯到父图,再选择其他的顶点进行删除,获得新子图。再判断新子图是否同构;由新子图同构来判定父图是否同构。当然了,如果对于 $\forall v'_1 \in V(H)$, 都有 $G-u_1 \not\cong H-v'_1$; 那么,就可以判定 $G \not\cong H$ 。

为了正确地判定在某一次迭代中子图是真正地不同构,根据定理2.2,必须在前一次迭代中,找到该子图的子图同构时的所有同构函数。这种寻找并保留所有同构函数的操作在整个逆向判定过程中的每一次迭代中都要进行,因此,提出来的图同构的判定算法不是多项式时间复杂度。

根据逆向判定过程的实施原理,不难发现,在每一次迭代中新增顶点及关联边时,如果新增的关联边的边数越少,对算法时间效率的提高就越有利。因此,在正向删除顶点,获得一系列子图的过程中,可以按照顶点的度数从小到大的次序来删除顶点,从而保证在逆向判定过程中每一次新增的关联边的边数都不太多。

根据以上分析,设计基于子图同构来判定父图同构的判定算法 GIBSGI (Graph Isomorphism Based-on Subgraph Isomorphism) 如下:

算法 GIBSGI

输入 图 G 的顶点集 $V(G)$ 和边集 $E(G)$, 图 H 的顶点集 $V(H)$ 和边集 $E(H)$ 。

输出 True—图 G 和 H 同构, False—图 G 和 H 不同构。

步骤1 计算图 G 和 H 的所有顶点的度数,并按从小到大的次序重新编号为 $u_1, u_2, \dots, u_n; v_1, v_2, \dots, v_n$ 。如果两个图的顶点度数序列不一致,则说明两个图 G 和 H 不同构,算法结束,返回 False 值。

步骤2 不断删除相同度数的、且最小度数的一个顶点,获得子图,直至子图仅含两个顶点为止。在这个过程中,每一次删除顶点时,应当保证新生成的子图具有顶点度数序列一致性。如果不能得到仅含两个顶点的子图 $G^{(n-2)}$ 和 $H^{(n-2)}$, 不妨设得到子图 $G^{(n-i)}$ 和 $H^{(n-i)}$ (这里 $i \geq 2$) 后中断删除操作,则必有 $G^{(n-i)}$ 和 $H^{(n-i)}$ 的顶点度数序列不一致。所以,必有 $G^{(n-i)} \not\cong H^{(n-i)}$, 转步骤7的后半部分操作。否则,得到了仅含两个顶点的子图 $G^{(n-2)}$ 和 $H^{(n-2)}$, 转步骤3。不失一般性,假设删除顶点后,获得的一系列子图为: $G^{(1)} = G - u_1, H^{(1)} = H - v_1; G^{(2)} = G^{(1)} - u_2 = (G - u_1) - u_2, H^{(2)} = H^{(1)} - v_2 = (H - v_1) - v_2; \dots; G^{(n-2)} = G^{(n-3)} - u_{n-2} = (G - u_1 - \dots - u_{n-3}) - u_{n-2}, H^{(n-2)} = H^{(n-3)} - v_{n-2} = (H - v_1 - \dots - v_{n-3}) - v_{n-2}$ 。

步骤3 令子图 $SG = G^{(n-2)} = (G - u_1 - \dots - u_{n-3}) - u_{n-2}, SH = H^{(n-2)} = (H - v_1 - \dots - v_{n-3}) - v_{n-2}, i=2$ 。

步骤4 直接判断两个顶点的子图 SG 和 SH 是否同构。若同构,则转步骤5。若不同构,则转步骤9。

步骤5 寻找 $SG \cong SH$ (即 $G^{(n-i)} \cong H^{(n-i)}$ 的所有的同构函数)。

步骤6 在子图 SG 和 SH 中,新增顶点 u_{n-i} 和 v_{n-i} , 以及相应的关联边(即回溯到上一级子图 $G^{(n-(i+1))}$ 和 $H^{(n-(i+1))}$)。

步骤7 应用定理2.1,判断 $SG+u_{n-i}$ 和 $SH+v_{n-i}$ 是否同构

(即 $G^{(n-(i+1))}$ 和 $H^{(n-(i+1))}$ 是否同构)。若同构,则 $SG+u_{n-i} \Rightarrow SG, SH+v_{n-i} \Rightarrow SH, i+1 \Rightarrow i$ (即 $G^{(n-(i+1))} \Rightarrow SG, H^{(n-(i+1))} \Rightarrow SH$), 转步骤8。若不同构,则 $SG+u_{n-i} \Rightarrow SG, SH+v_{n-i} \Rightarrow SH, i+1 \Rightarrow i$, 转步骤9。

步骤8 若 $|V(SG)|=|V(G)|$, 则判定图 G 和 H 同构,算法结束,返回 True 值。否则,转步骤5。

步骤9 若 $|V(SG)|=|V(G)|$, 则说明 $G^{(1)} \cong H^{(1)}$ 而 $G \not\cong H$ 。因此,可以判定图 G 和 H 不同构,算法结束,返回 False 值。否则,由于已经执行完步骤7的后半部分操作,使得 $H^{(n-i)}$ 已经回溯到父图(即上一级子图) $H^{(n-(i+1))}$ 。因此,可以重新选择相同度数的其他顶点 v'_{n-i} 进行删除,重复步骤2。如果回溯到父图 $H^{(n-(i+1))}$, 考虑完了所有相同度数的其他顶点 v'_{n-i} 进行删除后,仍然有父图的子图(即 $G^{(n-i)}$ 和 $H^{(n-i)}$)不同构,则继续回溯到祖父图(即上上一级子图) $H^{(n-(i+2))}$, 重新选择相同度数的其他顶点 $v'_{n-(i+1)}$ 进行删除,重复步骤2。这样的回溯过程可以一直进行下去,直至考虑完了原图 H 的所有的相同度数的其他顶点 v'_1 后,仍然有 $G-u_1$ 和 $H-v'_1$ 不同构,则判定图 G 和 H 不同构,算法结束,返回 False 值。

例2 图2给出了两个图 G 和 H 。 $\deg(u_1)=\deg(u_2)=\dots=\deg(u_8)=4, \deg(v_1)=\deg(v_2)=\dots=\deg(v_8)=4$ 。

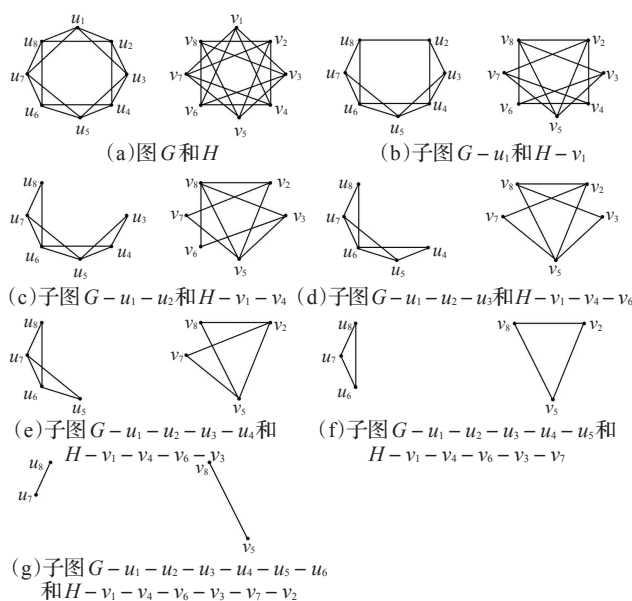


图2 图 G 和 H 以及依次删除顶点的系列子图

图2(a)给出了原图 G 和 H 。图2(b)给出了删除顶点 u_1 和 v_1 后,获得的第1代子图 $G-u_1$ 和 $H-v_1$ 。图2(c)给出了删除顶点 u_2 和 v_4 后,获得的第2代子图 $G-u_1-u_2$ 和 $H-v_1-v_4$ 。在 $G-u_1$ 中, $\deg(u_2)=3$ 。因此,在 $H-v_1$ 中,可以删除顶点 v_3 , 或 v_4 , 或 v_6 , 或 v_7 。又由于要保证删除后的子图具有相同的顶点度数序列,因此,只能删除顶点 v_4 , 或 v_6 。不失一般性,假设选择顶点 v_4 进行删除。其余子图的获得以此类推。

第0次迭代: $i=2$, 只有两个顶点的子图同构, $G-u_1-u_2-u_3-u_4-u_5-u_6 \cong H-v_1-v_4-v_6-v_3-v_7-v_2$ 。见图2(g)。所有的同构函数为: $f_1^{(0)}: u_8 \leftrightarrow v_8, u_7 \leftrightarrow v_5$ 。 $f_2^{(0)}: u_8 \leftrightarrow v_5, u_7 \leftrightarrow v_8$ 。

第1次迭代: $i=3$, 新增顶点 u_6 和 v_2 , 以及关联边 (u_6, u_7) ,

$(u_6, u_8), (v_2, v_5), (v_2, v_8)$ 。显然,三个顶点的子图同构, $G-u_1-u_2-u_3-u_4-u_5 \cong H-v_1-v_4-v_6-v_3-v_7$ 。见图2(f)。依据定理2.1,同构函数 $f_1^{(0)}$ 和 $f_2^{(0)}$ 起作用,见图3(a)。更新后所有的同构函数为: $f_1^{(1)}:u_8 \leftrightarrow v_8, u_7 \leftrightarrow v_5, u_6 \leftrightarrow v_2$ 。 $f_2^{(1)}:u_8 \leftrightarrow v_5, u_7 \leftrightarrow v_8, u_6 \leftrightarrow v_2$ 。 $f_3^{(1)}:u_8 \leftrightarrow v_2, u_7 \leftrightarrow v_5, u_6 \leftrightarrow v_8$ 。 $f_4^{(1)}:u_8 \leftrightarrow v_2,$

$u_7 \leftrightarrow v_8, u_6 \leftrightarrow v_5$ 。 $f_5^{(1)}:u_8 \leftrightarrow v_8, u_7 \leftrightarrow v_2, u_6 \leftrightarrow v_5$ 。 $f_6^{(1)}:u_8 \leftrightarrow v_5, u_7 \leftrightarrow v_2, u_6 \leftrightarrow v_8$ 。

第2次迭代: $i=4$,新增顶点 u_5 和 v_7 ,以及关联边 $(u_5, u_6), (u_5, u_7), (v_7, v_2), (v_7, v_5)$ 。显然,四个顶点的子图同构, $G-u_1-u_2-u_3-u_4 \cong H-v_1-v_4-v_6-v_3$ 。见图2(e)。依据定

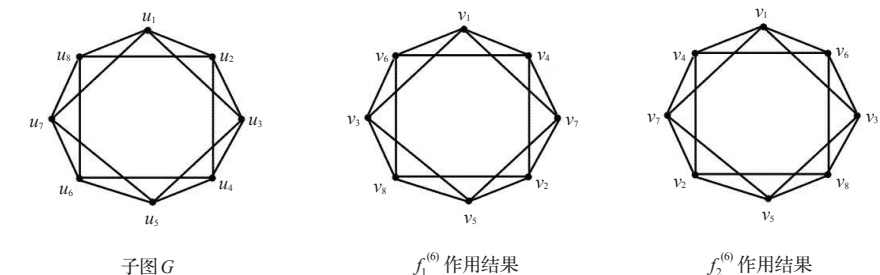
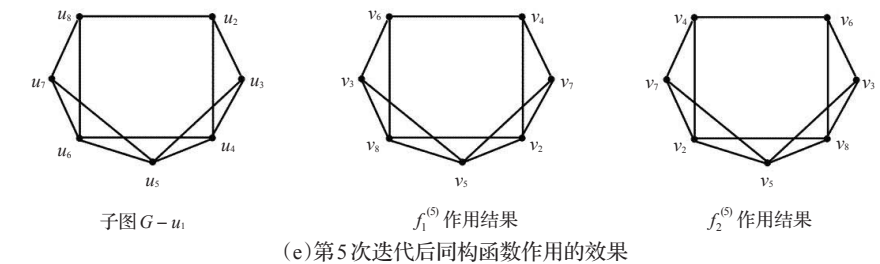
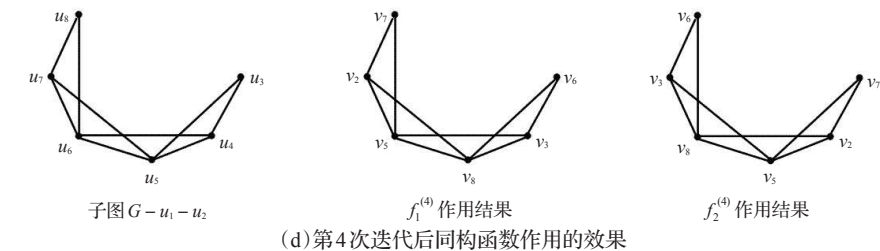
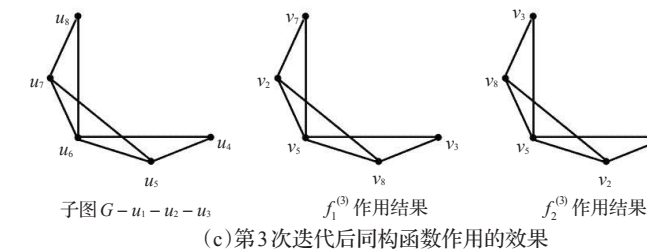
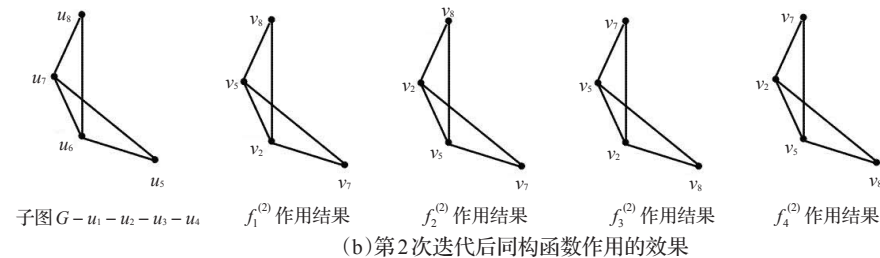
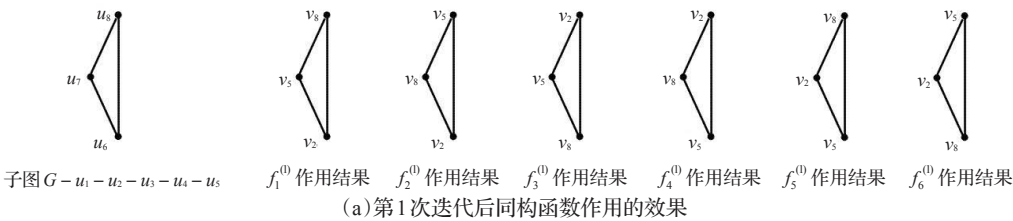


图3 每次迭代后经过函数变换得到的同构效果图

理 2.1, 同构函数 $f_1^{(1)}$ 和 $f_5^{(1)}$ 起作用, 见图 3(b)。更新后所有的同构函数为: $f_1^{(2)}: u_8 \leftrightarrow v_8, u_7 \leftrightarrow v_5, u_6 \leftrightarrow v_2, u_5 \leftrightarrow v_7$ 。
 $f_2^{(2)}: u_8 \leftrightarrow v_8, u_7 \leftrightarrow v_2, u_6 \leftrightarrow v_5, u_5 \leftrightarrow v_7$ 。
 $f_3^{(2)}: u_8 \leftrightarrow v_7, u_7 \leftrightarrow v_5, u_6 \leftrightarrow v_2, u_5 \leftrightarrow v_8$ 。
 $f_4^{(2)}: u_8 \leftrightarrow v_7, u_7 \leftrightarrow v_2, u_6 \leftrightarrow v_5, u_5 \leftrightarrow v_8$ 。

第 3 次迭代: $i=5$, 新增顶点 u_4 和 v_3 , 以及关联边 (u_4, u_5) , (u_4, u_6) , (v_3, v_5) , (v_3, v_8) 。显然, 五个顶点的子图同构, $G-u_1-u_2-u_3 \cong H-v_1-v_4-v_6$ 。见图 2(d)。依据定理 2.1, 同构函数 $f_4^{(2)}$ 起作用, 见图 3(c)。更新后所有的同构函数为: $f_1^{(3)}: u_8 \leftrightarrow v_7, u_7 \leftrightarrow v_2, u_6 \leftrightarrow v_5, u_5 \leftrightarrow v_8, u_4 \leftrightarrow v_3$ 。
 $f_2^{(3)}: u_8 \leftrightarrow v_3, u_7 \leftrightarrow v_8, u_6 \leftrightarrow v_5, u_5 \leftrightarrow v_2, u_4 \leftrightarrow v_7$ 。

第 4 次迭代: $i=6$, 新增顶点 u_3 和 v_6 , 以及关联边 (u_3, u_4) , (u_3, u_5) , (v_6, v_3) , (v_6, v_8) 。显然, 六个顶点的子图同构, $G-u_1-u_2 \cong H-v_1-v_4$ 。见图 2(c)。依据定理 2.1, 同构函数 $f_1^{(3)}$ 起作用, 见图 3(d)。更新后所有的同构函数为: $f_1^{(4)}: u_8 \leftrightarrow v_7, u_7 \leftrightarrow v_2, u_6 \leftrightarrow v_5, u_5 \leftrightarrow v_8, u_4 \leftrightarrow v_3, u_3 \leftrightarrow v_6$ 。
 $f_2^{(4)}: u_8 \leftrightarrow v_6, u_7 \leftrightarrow v_3, u_6 \leftrightarrow v_8, u_5 \leftrightarrow v_5, u_4 \leftrightarrow v_2, u_3 \leftrightarrow v_7$ 。

第 5 次迭代: $i=7$, 新增顶点 u_2 和 v_4 , 以及关联边 (u_2, u_3) , (u_2, u_4) , (u_2, u_8) , (v_4, v_2) , (v_4, v_6) , (v_4, v_7) 。显然, 七个顶点的子图同构, $G-u_1 \cong H-v_1$ 。见图 2(b)。依据定理 2.1, 同构函数 $f_2^{(4)}$ 起作用, 见图 3(e)。更新后所有的同构函数为: $f_1^{(5)}: u_8 \leftrightarrow v_6, u_7 \leftrightarrow v_3, u_6 \leftrightarrow v_8, u_5 \leftrightarrow v_5, u_4 \leftrightarrow v_2, u_3 \leftrightarrow v_7, u_2 \leftrightarrow v_4$ 。
 $f_2^{(5)}: u_8 \leftrightarrow v_4, u_7 \leftrightarrow v_7, u_6 \leftrightarrow v_2, u_5 \leftrightarrow v_5, u_4 \leftrightarrow v_8, u_3 \leftrightarrow v_3, u_2 \leftrightarrow v_6$ 。

第 6 次迭代: $i=8$, 新增顶点 u_1 和 v_1 , 以及关联边 (u_1, u_2) , (u_1, u_3) , (u_1, u_7) , (u_1, u_8) , (v_1, v_3) , (v_1, v_4) , (v_1, v_6) , (v_1, v_7) 。显然, 八个顶点的子图(即原图)同构, $G \cong H$ 。见图 2(a)。依据定理 2.1, 同构函数 $f_1^{(5)}$ 和 $f_2^{(5)}$ 起作用, 见图 3(f)。此时, 至少存在两个函数: $f_1^{(6)}: u_8 \leftrightarrow v_6, u_7 \leftrightarrow v_3, u_6 \leftrightarrow v_8, u_5 \leftrightarrow v_5, u_4 \leftrightarrow v_2, u_3 \leftrightarrow v_7, u_2 \leftrightarrow v_4, u_1 \leftrightarrow v_1$ 。
 $f_2^{(6)}: u_8 \leftrightarrow v_4, u_7 \leftrightarrow v_7, u_6 \leftrightarrow v_2, u_5 \leftrightarrow v_5, u_4 \leftrightarrow v_8, u_3 \leftrightarrow v_3, u_2 \leftrightarrow v_6, u_1 \leftrightarrow v_1$ 。它们是 $G \cong H$ 的同构函数。

因为已经判断出八个顶点的子图是同构的, 即 $G \cong H$ 。所以, 算法结束, 不需要进行第 7 次迭代, 返回 True 值。

在每一次迭代中都可以发现以下规律: (1) 第 1 次迭代后得到的同构函数 $f_1^{(1)}$ 和 $f_2^{(1)}$, 是第 0 次迭代后得到的同构函数 $f_1^{(0)}$ 和 $f_2^{(0)}$, 直接补充新的顶点对应关系 $u_6 \leftrightarrow v_2$ 得到。(2) 第 2 次迭代后得到的同构函数 $f_1^{(2)}$ 和 $f_2^{(2)}$, 是第 1 次迭代后得到的同构函数 $f_1^{(1)}$ 和 $f_5^{(1)}$, 直接补充新的顶点对应关系 $u_5 \leftrightarrow v_7$ 得到。(3) 第 3 次迭代后得到的同构函数 $f_1^{(3)}$, 是第 2 次迭代后得到的同构函数 $f_4^{(2)}$, 直接补充新的顶点对应关系 $u_4 \leftrightarrow v_3$ 得到。(4) 第 4 次迭代后得到的同构函数 $f_1^{(4)}$, 是第 3 次迭代后得到的同构函数 $f_1^{(3)}$, 直接补充新的顶点对应关系 $u_3 \leftrightarrow v_6$ 得到。(5) 第 5 次迭代后得到的同构函数 $f_1^{(5)}$, 是第 4 次迭代后得到的同构函数 $f_2^{(4)}$, 直接补充新的顶点对应关系 $u_2 \leftrightarrow v_4$ 得到。(6) 第 6 次迭代后得到的同构函数 $f_1^{(6)}$ 和 $f_2^{(6)}$, 是第 5 次迭代后得到的同构函数 $f_1^{(5)}$ 和 $f_2^{(5)}$, 直接补充新的顶点对应关系

$u_1 \leftrightarrow v_1$ 得到。

上述这个规律反映了定理 2.1 揭示的一个事实真相, 即 $G-u \cong H-v$ 的一个同构函数 f , 通过直接补充新的顶点对应关系 $v=f(u)$, 必然将 f 升阶为 $G \cong H$ 的一个同构函数。从而, 也通过直观的方式证明了定理 2.1 的正确性。

例 3 图 4 给出了两个图 G 和 H 。 $\deg(u_1)=\deg(u_2)=\dots=\deg(u_8)=4$, $\deg(v_1)=\deg(v_2)=\dots=\deg(v_8)=4$ 。

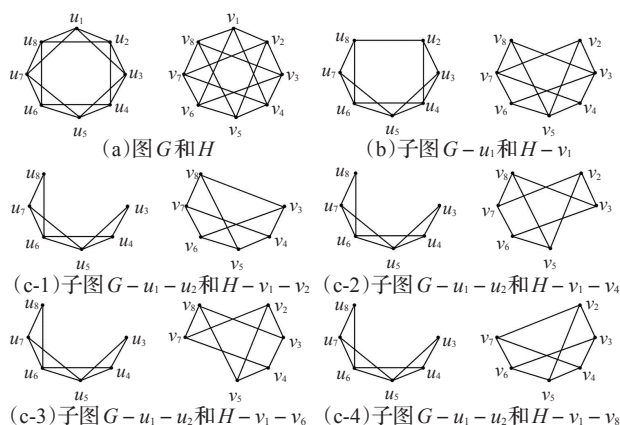


图 4 图 G 和 H 以及依次删除顶点的系列子图

图 4(a) 给出了原图 G 和 H 。图 4(b) 给出了删除顶点 u_1 和 v_1 后, 获得的第 1 代子图 $G-u_1$ 和 $H-v_1$ 。图 4(c-1) 给出了删除顶点 u_2 和 v_2 后, 获得的第 2 代子图 $G-u_1-u_2$ 和 $H-v_1-v_2$ 。显然, 子图 $G-u_1-u_2$ 和 $H-v_1-v_2$ 不具有相同的顶点度数序列, 所以, 它们不同构。此时, 正在执行算法的步骤 2; 且没能进行不断删除顶点, 直至得到两个顶点的子图。因此, 转步骤 7 的后半部分, 设置回溯点; 再转步骤 9, 回溯到父图 $H-v_1$, 再选择 v_4 , 或 v_6 , 或 v_8 进行删除。结果都是新子图不具有相同的顶点度数序列(见图 4(c-2)~图 4(c-4)所示)。由此, 可以判定 $G-u_1$ 和 $H-v_1$ 不同构。 $H-v_1-v_2$ 再回溯到祖父图(即 $H-v_1$ 回溯到父图 H), 再选择相同度数的另外顶点(如 v_2, v_3, \dots, v_8)进行删除, 得到 $H-v_i$ ($2 \leq i \leq 8$)。重复上面的“正向删除+判定子图具有相同的顶点度数序列”过程, 发现都不同构。即 $G-u_1 \not\cong H-v_i$ ($2 \leq i \leq 8$)。至此, 有 $G-u_1 \not\cong H-v_i$ ($1 \leq i \leq 8$)。所以, 可以判定图 G 和 H 不同构。算法结束, 返回 False 值。在这个例子中, 由于不能不断删除顶点, 直至得到两个顶点的子图。因此, 可以判定图 G 和 H 不同构。

4 结束语

图同构的判定性问题, 由于缺乏有效而简洁的充分必要条件, 因此不容易解决。Ulam 提出了一个判定图同构的猜想, 也称为图的重构猜想: 若存在双射函数 $f: V(G) \rightarrow V(H)$, 使得对任意的 $v \in V(G)$, 有 $G-u \cong H-f(v)$, 则称 H 是 G 的一个重构。虽然围绕这个猜想, 产生了许多有积极意义的研究成果^[1-10]; 但是, 至今没有完全解决 Ulam 猜想。

为了寻找图同构的判定方法, 受 Ulam 猜想的启发, 提出了一个新技术, 即具有同构关系的对应点无限衍生技术。利用这个技术, 采用反证法证明了下述思想: 同构的两个图, 各自增加一个顶点和与该顶点关联的若干条边, 形成了两个新

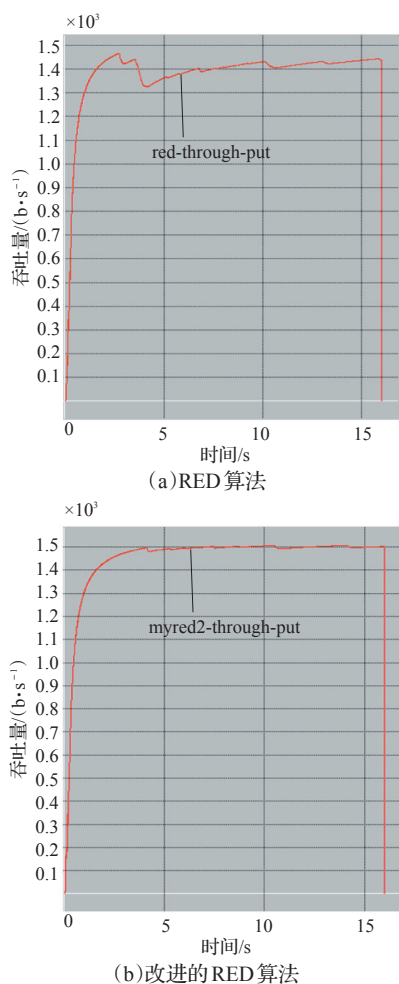


图4 两种算法的吞吐量变化曲线

包。队列最小阈值取5,最大阈值取15,权值 w_q 取0.002。在仿真过程中分别于3s和10s时加入数据源流量突增和骤减两种情况。分别使用原始RED和改进后的RED算法做对比。

4.2 仿真结果与分析

仿真结果给出两组算法性能对比图,分别为队列长度曲线图和吞吐量变化曲线图。

从图3可以看出,改进后的RED算法的队列长度波动较为平稳,波动不大,算法的强壮性和稳定性有所增强。在吞吐

量方面,从图4可以看出改进后的算法也有明显优势。总的来说,改进后的算法在稳定性和吞吐量方面都有一定优势。

5 结束语

对RED算法进行研究并对其优缺点进行了分析,在原有基础上对算法进行改进,针对线性丢弃、平均队列不能完全反映网络拥塞状况等问题,提出相应改进算法。该算法对丢弃概率进行非线性平滑,并且在丢弃策略中加入当前队列长度这一考虑因素,用以判断实际拥塞情况。仿真实验表明,改进算法弥补了原始算法的不足之处,无论是稳定性还是吞吐量均要优于原算法。

参考文献:

- [1] Ohta K, Yoshikawa T, Nakagawa T, et al. Design and implementation of mobile grid middleware for handsets[C]//Proceedings of the 11th International Conference on Parallel and Distributed Systems. Washington D C, USA: IEEE Computer Society, 2005: 679-683.
- [2] Floyd S, Jacobson V. Random early detection gateways for congestion avoidance[J]. IEEE/ACM Transactions on Networking, 1993, 1(4): 397-413.
- [3] 林闯, 单志广, 任丰原. 计算机网络的服务质量[M]. 北京: 清华大学出版社, 2004: 157-158.
- [4] 徐雷鸣, 庞博, 赵耀. NS 与网络模拟[M]. 北京: 人民邮电出版社, 2003.
- [5] 方路平, 刘世华, 陈盼, 等. NS-2 网络模拟基础与应用[M]. 北京: 国防工业出版社, 2008.
- [6] Abbasov B, Korukoglu S. Effective RED: an algorithm to improve RED's performance by reducing packet loss rate[J]. Network Compute Appl, 2008.
- [7] Chen W, Yang S H. The mechanism of adapting RED parameters to TCP traffic[J]. Comput Commun, 2009.
- [8] Xiong D N. Design and analysis of a self-tuning active queue management scheme for the Internet[J]. Comput Netw, 2009.
- [9] Domańska J, Domański A. Adaptive RED in AQM[C]//16th Conference on Computer Networks, Wisła, Poland, 2009.
- [10] Grochla K. Simulation comparison of active queue management algorithm in TCP/IP networks[J]. Telecommunication Systems, 2008, 39(2).

(上接57页)

图。(1)这两个新图同构,当且仅当新顶点的新关联边所关联的顶点(即新顶点在原图中的邻接点)在原图(即子图)中保持同构关系。(2)这两个新图不同构,当且仅当新顶点的新关联边所关联的顶点(即新顶点在原图中的邻接点)在原图(即子图)中不保持同构关系。

基于这个充分必要条件,设计并实现了一个具有递归特性(即迭代特性)的、基于子图同构判定父图同构的判定算法(GIBSGI),并用具体的实例进行了算法正确性和有效性的验证。

参考文献:

- [1] Kelly P J. A congruence theorem for trees[J]. Pacific J Math, 1957, 7: 961-968.
- [2] Bondy J A. On Kelly's congruence for trees[J]. Proc Cambridge Philos Soc, 1969, 65: 1-11.

- [3] Harary F, Palmer E. The reconstruction of a tree from its maximal subtrees[J]. Canadian J Math, 1966, 18: 803-810.
- [4] Manvel B. Reconstructing of trees[J]. Canadian J Math, 1970, 22: 55-60.
- [5] Nešetřil J. A congruence theorem for asymmetric trees[J]. Pacific J Math, 1971, 37: 771-778.
- [6] Kocay W L. Some new methods in reconstruction theory[C]//Lecture Notes in Math. Berlin: Springer, 1982: 89-114.
- [7] Xie Litong, Fan Hongbing. On reconstructible local subgraphs[J]. J Advances in Math, 1996, 25: 65-70.
- [8] Xie Litong, Fan Hongbing. A new result on reconstructible local subgraphs[J]. J Advances in Math, 1997, 26: 440-444.
- [9] 谢力同, 刘桂真. 领域伪相似点的可重构性[J]. 数学物理学报, 1997 (12): 225-228.
- [10] 龙和平, 谢力同, 严谨, 等. 边型带权核子图的边可重构性[J]. 山东大学学报: 理学版, 2002, 37: 105-107.