

全边增长网络模型的生成树^{*}

王晓敏, 赵喜杨, 姚 兵

(西北师范大学数学与统计学院, 甘肃 兰州 730070)

摘 要: 定义了全边增长网络模型, 给出寻找全边增长网络模型的具有最多叶子生成树的 2 个算法, 分析了全边增长网络模型的小世界性, 并验证具有最多叶子生成树的无标度性, 以及最多叶子生成树在平衡集和最小控制集中的应用。

关键词: 无标度网络; 生成树; 网络模型

中图分类号: O157.5 **文献标志码:** A **文章编号:** 0529-6579 (2016) 01-0048-06

Spanning trees of totally edge-growing network models

WANG Xiaomin, ZHAO Xiyang, YAO Bing

(College of Mathematics and Statistics, Northwest Normal University, Lanzhou 730070, China)

Abstract: Totally edge-growing network models are defined. Two algorithms for finding Maximum Leaf Spanning trees (MLS-trees) of the totally edge-growing models is shown. The small world property of MLS-trees is analysed, and the scale-free with MLS-trees is verified. An application of MLS-trees to determine balanced sets and smallest dominating sets of totally edge-growing network models is given.

Key words: scale-free network; spanning tree; network models

现实生活中处处存在着各种复杂关系网, 如物联网, 地震网络, 食物链网络, 电力网, 科学引文网, 朋友关系网等等, 它们都是由海量个体所组成的, 并且个体之间还存在着相互作用。如果将个体抽象为网络的节点, 将个体之间的相互作用抽象为节点之间的边, 那么上述各种复杂关系网就演化为我们研究的复杂网络。近 20 年来, 人们掀起了研究复杂网络的热潮。复杂网络的拓扑结构及其性质是研究复杂网络的关键, 而且复杂网络的研究以小世界网络和无标度网络为最热点。Barabási 等^[1]描述复杂网络研究中的一个现象: “优先连接的机制常常是线性的。换句话说, 如果一个现存节点的连接数是其相邻节点连接数的两倍, 那么新节点与它连接的可能性, 也是与邻近节点连接可能性的两

倍, 那就容易出现一个获取最多连线的集散节点。我们期待这种标度不变性是许多复杂网络的通用特性”。之后的众多研究证实了他们的期待。

通常, 复杂网络中只连一条边的节点叫做叶子。由于实际网络中含有很多叶子, 运用生成树研究网络自然而简便, 生成树已经被应用于网络的搜索。众所周知, 树在各种网络、程序设计、物流运输、电路设计、化学结构等领域有着广泛的应用。例如, 利用生成树来寻找无标度网络的关键节点以及用树来确定网络的控制集还可以利用无标度生成树来确定动态网络的拓扑结构, 揭示现实网络的内在规律。从而, 生成树的无标度行为与网络的拓扑结构演化之间的关系成为研究的关键点, 少数关键节点的稳定性对整个系统的稳定性起着决定性作

* 收稿日期: 2015-09-08

基金项目: 国家自然科学基金资助项目 (61163054, 61363060)

作者简介: 王晓敏 (1989 年生), 女; 研究方向: 复杂网络; 通讯作者: 姚兵; E-mail: yybb918@163.com

用。最典型的应用之一是: Macdonald 等^[2]研究了无标度权重网络的最小生成树。有些学者研究了无标度网络中的无标度树^[3]。Kim 等^[4]利用生成树来研究无标度网络, 却没有给出生成树的图结构分析, 其侧重点仍然是统计分析和解析计算。他们利用网络模型的介数中心度来研究网络模型的生成树, 并试图证实在无标度网络中存在具有特定指标的生成树, 他们称这种生成树为通讯核。为了证实生成树的无标度行为, 他们还研究了正在成长的网络中节点度的演化。此外, 章忠志等^[5]计算了阿波罗网的生成树的数目。

本文借用文献 [6-8] 中的递归构造思想, 构造了基于任何连通初始网络的全边增长网络模型, 提供 2 个算法来寻找该模型的具有最多叶子的生成树, 研究探讨该模型的无标度生成树。本文所考虑的图均为有限、无向简单图, 文中没有定义的术语和符号均采用于文献 [9]。对于图 G , 记号 $L(G)$ 表示叶子点的集合, $n_d(G)$ 表示图 G 中度数为 d 的顶点的个数。以下记号 $|X|$ 表示集合 X 的元素个数。

1 模型和算法

1.1 全边增长网络模型

构造全边增长网络模型 $N(t)$, 且用记号 $V(t)$ 和 $E(t)$ 分别表示网络 $N(t)$ 在时刻 t 时的节点集合和边集合, 以及它们的元素个数 $n_{v,t} = |V(t)|$, $n_{e,t} = |E(t)|$ 。对于一个连通初始网络模型 $N(0)$ 的一条边 $uv \in E(0)$, 新增加一个节点 w , 将新增节点 w 分别与节点 u 和节点 v 相连, 产生边 wu 和边 wv 。把上述添加节点和边的过程叫做 A -运算。对 $N(0)$ 的每一条边 $uv \in E(0)$ 实施 A -运算后, 所得到的网络模型就是 $t = 1$ 时刻的全边增长网络模型 $N(1)$ 。用 X_1 表示给 $N(0)$ 新增节点的集合, Y_1 表示给 $N(0)$ 新增边的集合, 得到

$$V(1) = V(0) \cup X_1,$$

$$E(1) = E(0) \cup Y_1$$

继续对网络模型 $N(1)$ 的每一条边 $xy \in E(1)$ 实施 A -运算后, 就得到 $t = 2$ 时刻的全边增长网络模型 $N(2)$ 。依次类推, 对全边增长网络模型 $N(t-1)$ 的每一条边 $xy \in E(t-1)$ 实施 A -运算后, 就得到全边增长网络模型 $N(t)$, 见图 1 中的例子。

下面给出了全边增长网络模型 $N(t)$ 的一些基本参数。对于 $t \geq 1$, $N(t)$ 的节点集合和边集合分别表示

$$\begin{cases} V(t) = V(t-1) \cup X_t = V(0) \cup (\cup_{j=1}^t X_j), \\ E(t) = E(t-1) \cup Y_t = E(0) \cup (\cup_{j=1}^t Y_j) \end{cases} \quad (1)$$

其中, X_t 表示 t 时刻新增节点的集合, Y_t 表示 t 时刻新增边的集合。因为

$$\begin{cases} n_{v,1} = n_{v,0} + n_{e,0}, \\ n_{e,1} = n_{e,0} + 2n_{e,0} = 3n_{e,0} \end{cases} \quad (2)$$

根据 (1) 式和 (2) 式, 当 $t \geq 1$ 时, 不难得到 $N(t)$ 的节点数目和边数目分别为

$$\begin{cases} n_{v,t} = n_{v,t-1} + n_{e,t-1} = n_{v,0} + n_{e,0} \sum_{i=0}^{t-1} 3^i = \\ n_{v,0} + (3^t - 1)n_{e,0}/2, \\ n_{e,t} = n_{e,t-1} + 2n_{e,t-1} = 3n_{e,t-1} = 3^t n_{e,0} \end{cases} \quad (3)$$

此外, 可以算出新增加的节点个数 $|X_t| = 3^{t-1}n_{e,0}$ 以及新增加的边的数目 $|Y_t| = 2|X_t|$ 。全边增长网络模型 $N(t)$ 的平均度为 $[k]$ 满足

$$[k] = 2n_{e,t}/n_{v,t} = 2 \cdot \frac{3^t n_{e,0}}{n_{v,0} + (3^t - 1)n_{e,0}/2} \rightarrow 4, t \rightarrow \infty \quad (4)$$

平均度 $[k]$ 说明 $N(t)$ 是一个稀疏网络。Barabasi 等^[10]发现少数高连通网页将整个 WWW 支撑连接在一起。其中, 80% 的网页平均只有 4 个链接。但是, 占网页总数目 0.01 的网页, 每个却有不小 1 000 的链接。这说明本文构造的全边增长网络模型与实际的 WWW 有一致之处。

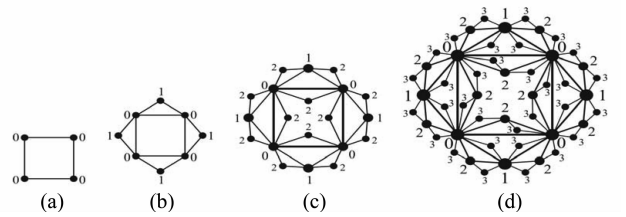


图 1 全边增长网络模型

Fig. 1 Totally edge-growing network models

(a) $N(0) = C_4$; (b) $N(1)$; (c) $N(2)$; (d) $N(3)$

1.2 全边增长网络模型的具有最多叶子的生成树

全边增长网络模型 $N(t)$ 的生成树是一个连通且边数目最少的子网络, 本小节寻找 $N(t)$ 的具有最多叶子的生成树。一般情形下, $N(t)$ 的具有最多叶子的生成树不唯一。用 $L(T^M(t))$ 表示 $N(t)$ 的具有最多叶子的生成树 $T^M(t)$ 的全体叶子的集合。全边增长网络模型 $N(t)$ 的生成树 $T^M(t)$ 具有如下的性质。

引理 1 全边增长网络模型 $N(t)$ 的任意一棵具有最多叶子的生成树 $T^M(t)$ 满足 $X_t \subset L(T^M(t))$ 。

证明 对任意时刻 $t \geq 1$, 假设存在节点 $w \in X_t$, 但 $w \notin L(T^M(t))$ 。根据全边增长网络模型 $N(t)$ 的构造, 节点 w 的度数 $d(w) = 2$, 且节点 w 与全边增长网络模型 $N(t-1)$ 的一条边 uv 的 2 个端点 u 和 v 分别相连。注意到, 节点 u 和 v 至少有一个不是 $T^M(t)$ 的叶子, 不妨设节点 u 不是叶子。则可以构造全边增长网络模型 $N(t)$ 的另一棵生成树 H 如下: 在 $T^M(t)$ 中删除边 wv , 连接边 uw 。显然 $w, v \in L(H)$, 且生成树 H 的叶子数目

$$|L(H)| \geq |L(T^M(t))| + 1$$

这矛盾于 $T^M(t)$ 是 $N(t)$ 的一棵具有最多叶子的生成树, 本引理得证。

引理 2 当 $k \geq 1$ 时, 全边增长网络模型 $N(k)$ 的节点集 $V(k)$ 中的任何节点不是全边增长网络模型 $N(k+2)$ 的任何一棵具有最多叶子的生成树 $T^M(k+2)$ 的叶子, 即

$$L(T^M(k+2)) \cap V(k) = \emptyset$$

证明 根据全边增长网络模型的构造, $N(k)$ 没有叶子。假设存在节点

$$u \in L(T^M(k+2)) \cap V(k)$$

即

$$L(T^M(k+2)) \cap V(k) \neq \emptyset$$

注意到, 节点 u 在 $N(k)$ 中的度数 $d(u) \geq 2$, 设 $u_1, u_2, \dots, u_{d(u)}$ 是节点 u 在 $N(k)$ 中的邻点。构造 $N(k+1)$ 的过程, 是给边 $uu_i (i = 1, 2, \dots, d(u))$ 添加节点 $u_i^{(k+1)}$, 然后将节点 $u_i^{(k+1)}$ 分别与节点 u 和节点 u_i 连边 $uu_i^{(k+1)}$ 和 $u_i u_i^{(k+1)}$ 。在构造 $N(k+2)$ 的过程中, 要给边 $uu_i^{(k+1)} (i = 1, 2, \dots, d(u))$ 添加节点 $u_i^{(k+2)}$, 然后将节点 $u_i^{(k+2)}$ 分别与节点 u 和节点 $u_i^{(k+1)}$ 连边 $uu_i^{(k+2)}$ 和边 $u_i^{(k+1)} u_i^{(k+2)}$ 。由于节点 u 是 $T^M(k+2)$ 的叶子, 而引理 1 保证, 新增节点集合 X_{k+2} 的每一个节点 $u_i^{(k+2)}$ 必须是 $T^M(k+2)$ 的叶子, 即每一个节点 $u_i^{(k+2)}$ 是 $T^M(k+2)$ 的叶子。那么, 每一个节点 $u_i^{(k+1)} (i = 1, 2, \dots, d(u))$ 就不能是 $T^M(k+2)$ 的叶子。至此, 能够用 $T^M(k+2)$ 来构造 $N(k+2)$ 的另一棵生成树 H 如下: 把 $T^M(k+2)$ 的边 $u_i^{(k+1)} u_i^{(k+2)} (i = 1, 2, \dots, d(u))$ 删去, 再将节点 u 与节点 $u_i^{(k+2)}$ 分别连边。容易计算生成树 H 的叶子数目 $|L(H)|$ 满足

$$|L(H)| = |L(T^M(k+2))| + d(u) - 1 >$$

$$|L(T^M(k+2))|$$

这是一个明显的矛盾。

定理 1 当 $t \geq 3$ 时, 全边增长网络模型 $N(t)$ 的每一棵包含最多叶子的生成树 $T^M(t)$ 的叶子数目

$$|L(T^M(t))| = 4 \cdot 3^{t-2} n_{e,0} \quad (5)$$

证明 先给出 2 个算法。

算法 1 层次优先搜索算法 (Level-first searching algorithm, LFS-算法)

输入: 一个全边增长网络模型 $N(t), t \geq 3$ 。

输出: $N(t)$ 的全体具有最多叶子的生成树。

(初始化 Initialization)

步骤 1 当 $k = 1, 2$ 时, $F^M(k) \leftarrow \{N(k) \text{ 的全体具有最多叶子的生成树}\}, F(k) \leftarrow \{N(k) \text{ 的全体生成树}\}。$

步骤 2 $k \leftarrow 1$ 若 t 为奇数, 否则 $k \leftarrow 2; F^M(k+2) \leftarrow \emptyset$ 。对每一棵生成树 $T_0 \in F(k)$, 有节点集合

$$V(T_0) = \{u_i^{(k)} \mid i = 1, 2, \dots, n_{v,k}\};$$

$$F_0^M(k+2) \leftarrow \emptyset, \text{ 做: 从 } i = 1 \text{ 到 } i = n_{v,k},$$

$$E'(T_{i-1}) \leftarrow \{u_i^{(k)} u_s^{(k+1)} \in E(k+1),$$

$$u_i^{(k)} u_l^{(k+2)} \in E(k+2) \mid u_s^{(k+1)} \in$$

$$V(k+1) \setminus V(T_{i-1}),$$

$$u_l^{(k+2)} \in V(k+2) \setminus V(T_{i-1})\},$$

$$E(T_i) \leftarrow E(T_{i-1}) \cup E'(T_{i-1});$$

$$V(T_i) \leftarrow V(T_{i-1}) \cup \{x \mid xu_i^{(k)} \in E'(T_{i-1})\};$$

$$T^M(k+2) \leftarrow T_{n_{v,k}};$$

$$F_i^M(k+2) \leftarrow \{T^M(k+2)\};$$

$$F^M(k+2) \leftarrow F^M(k+2) \cup F_{n_{v,k}}^M(k+2);$$

$$F(k+2) \leftarrow \{N(k) \text{ 的全体生成树}\}$$

(循环 Iteration)

步骤 3 如果 $t = k+2$, 到步骤 6; 否则, 到步骤 4。

步骤 4 如果 $s < t-2$, 做: $F^M(s+1) \leftarrow \emptyset$; 对每一棵生成树 $T_0 \in F(s)$, 节点集合

$$V(T_0) = \{u_i^{(s)} \mid i = 1, 2, \dots, n_{v,s}\}$$

有 $F_0^M(s+1) \leftarrow \emptyset$, 从 $i = 1$ 到 $i = n_{v,s}$, 做:

$$E'(T_{i-1}) \leftarrow \{u_i^{(s)} u_r^{(s+1)} \in E(s+1),$$

$$u_i^{(s)} u_r^{(s+1)} \in E(s+1) \mid u_r^{(s+1)} \in$$

$$V(s+1) \setminus V(T_{i-1})\};$$

$$E(T_i) \leftarrow E(T_{i-1}) \cup E'(T_{i-1});$$

$$V(T_i) \leftarrow V(T_{i-1}) \cup \{y \mid yu_r^{(s)} \in E'(T_{i-1})\};$$

$$T^M(s+1) \leftarrow T_{n_{v,s}};$$

$$F_i^M(s+1) \leftarrow F_{i-1}^M(s+1) \cup \{T^M(s+1)\};$$

$$F^M(s+1) \leftarrow F^M(s+1) \cup F_{n_{v,s}}^M(s+1);$$

$$F^M(s+1) \leftarrow \{N(s+1) \text{ 的全体生成树}\}$$

步骤 5 如果 $s \neq t-2$, 到步骤 4。如果 $s = t-2$, 做: $F^M(t) \leftarrow \emptyset$; 对每一棵生成树 $T_0 \in F(t-2)$,

有节点集合

$$V(t-2) = \{u_i^{(t-2)} \mid i = 1, 2, \dots, n_{v, t-2}\},$$

$$F_0^M(t) \leftarrow \emptyset$$

从 $i = 1$ 到 $i = n_{v, t-2}$,

$$E'(T_{i-1}) \leftarrow \{u_i^{(t-2)} u_r^{(t-1)} \in E(t-1),$$

$$u_i^{(s)} u_l^{(t)} \in E(t) \mid u_r^{(t-1)} \in V(t-1) \setminus V(T_{i-1}),$$

$$u_l^{(t)} \in V(t) \setminus V(T_{i-1})\};$$

$$E(T_i) \leftarrow E(T_{i-1}) \cup E'(T_{i-1});$$

$$V(T_i) \leftarrow V(T_{i-1}) \cup \{y \mid y u_i^{(t-2)} \in E'(T_{i-1})\};$$

$$T^M(t) \leftarrow T_{n_{v, t-2}};$$

$$F_i^M(t) \leftarrow F_{i-1}^M(t) \cup \{T^M(t)\};$$

$$F^M(t) \leftarrow F^M(t) \cup F_{n_{v, t-2}}^M(k+2)$$

(终止 Termination)

步骤 6 返回 $F^M(t)$ 。

算法 2 大度优先层次搜索算法 (BDFLS - 算法, Big - degree - first level searching algorithm)

输入: 一个全边增长网络模型 $N(t)$, $t \geq 3$ 。

输出: $N(t)$ 的全体具有最多叶子的生成树。

(初始化 Initialization)

步骤 1 当 $k = 1, 2$ 时, $F^M(k) \leftarrow \{N(k)$ 的全体具有最多叶子的生成树 $\} F(k) \leftarrow \{N(k)$ 的全体生树 $\}$ 。

步骤 2 $k \leftarrow 1$ 若 t 为奇数, 否则 $k \leftarrow 2$; $F^M(k+2) \leftarrow \emptyset$ 。对每一棵生成树 $T_0 \in F(k)$, $F_0^M(k+2) \leftarrow \emptyset$, 做: 使得节点集合

$$V(T_0) = \{u_i^{(k)} \mid i = 1, 2, \dots, n_{v, k}\}$$

的节点度数为递减数列

$$d(u_{i+1}^{(k)}) \leq d(u_i^{(k)}), i = 1, 2, \dots, n_{v, k} - 1$$

从 $i = 1$ 到 $i = n_{v, k}$,

$$E'(T_{i-1}) \leftarrow \{u_i^{(k)} u_s^{(k+1)} \in E(k+1),$$

$$u_i^{(k)} u_l^{(k+2)} \in E(k+2) \mid u_s^{(k+1)} \in$$

$$V(k+1) \setminus V(T_{i-1}),$$

$$u_l^{(k+2)} \in V(k+2) \setminus V(T_{i-1})\};$$

$$E(T_i) \leftarrow E(T_{i-1}) \cup E'(T_{i-1});$$

$$V(T_i) \leftarrow V(T_{i-1}) \cup \{x \mid x u_i^{(k)} \in E'(T_{i-1})\};$$

$$T^M(k+2) \leftarrow T_{n_{v, k}};$$

$$F_i^M(k+2) \leftarrow F_{i-1}^M(k+2) \cup \{T^M(k+2)\};$$

$$F^M(k+2) \leftarrow F^M(k+2) \cup F_{n_{v, k}}^M(k+2);$$

$$F(k+2) \leftarrow \{N(k) \text{ 的全体生成树}\}$$

(循环 Iteration)

步骤 3 如果 $t = k+2$, 到步骤 6; 否则, 到步骤 4。

步骤 4 如果 $s < t-2$, 做: $F^M(s+1) \leftarrow \emptyset$; 对每一棵生成树 $T_0 \in F(s)$, 使得节点集合

$$V(T_0) = \{u_i^{(s)} \mid i = 1, 2, \dots, n_{v, s}\}$$

的节点度数为递减数列

$$d(u_{i+1}^{(s)}) \leq d(u_i^{(s)}), i = 1, 2, \dots, n_{v, s} - 1;$$

$$F_0^M(s+1) \leftarrow \emptyset$$

从 $i = 1$ 到 $i = n_{v, s}$, 做:

$$E'(T_{i-1}) \leftarrow \{u_i^{(s)} u_r^{(s+1)} \in$$

$$E(s+1) \mid u_r^{(s+1)} \in V(s+1) \setminus V(T_{i-1})\};$$

$$E(T_i) \leftarrow E(T_{i-1}) \cup E'(T_{i-1});$$

$$V(T_i) \leftarrow V(T_{i-1}) \cup \{y \mid y u_r^{(s)} \in E'(T_{i-1})\};$$

$$T^M(s+1) \leftarrow T_{n_{v, s}};$$

$$F_i^M(s+1) \leftarrow F_{i-1}^M(s+1) \cup \{T^M(s+1)\};$$

$$F^M(s+1) \leftarrow F^M(s+1) \cup F_{n_{v, s}}^M(k+2);$$

$$F^M(s+1) \leftarrow \{N(s+1) \text{ 的全体生成树}\}$$

步骤 5 如果 $s \neq t-2$, 到步骤 4。如果 $s = t-2$, 做: $F^M(t) \leftarrow \emptyset$; 做: 每一棵生成树 $T_0 \in F(t-2)$, 有节点集合

$$V(t-2) = \{u_i^{(t-2)} \mid i = 1, 2, \dots, n_{v, t-2}\}$$

的节点度数为递减数列

$$d(u_{i+1}^{(t-2)}) \leq d(u_i^{(t-2)}), i = 1, 2, \dots, n_{v, t-2} - 1;$$

$$F_0^M(t) \leftarrow \emptyset$$

从 $i = 1$ 到 $i = n_{v, t-2}$,

$$E'(T_{i-1}) \leftarrow \{u_i^{(t-2)} u_r^{(t-1)} \in E(t-1),$$

$$u_i^{(s)} u_l^{(t)} \in E(t) \mid u_r^{(t-1)} \in V(t-1) \setminus V(T_{i-1}),$$

$$u_l^{(t)} \in V(t) \setminus V(T_{i-1})\};$$

$$E(T_i) \leftarrow E(T_{i-1}) \cup E'(T_{i-1});$$

$$V(T_i) \leftarrow V(T_{i-1}) \cup \{y \mid y u_i^{(t-2)} \in E'(T_{i-1})\};$$

$$T^M(t) \leftarrow T_{n_{v, t-2}};$$

$$F_i^M(t) \leftarrow F_{i-1}^M(t) \cup \{T^M(t)\};$$

$$F^M(t) \leftarrow F^M(t) \cup F_{n_{v, t-2}}^M(k+2)$$

(终止 Termination)

步骤 6 返回 $F^M(t)$ 。

根据引理 1 和引理 2, 上述 2 个算法找到的集合 $F^M(t)$ 包含了全边增长网络模型 $N(t)$ 的所有具有最多叶子的生成树, 且上述 2 个算法指明每一棵具有最多叶子的生成树 $T^M(t) \in F^M(t)$ 的叶子数目为

$$|L(T^M(t))| = |X_t| + |X_{t-1}| =$$

$$3^{t-1} n_{e, 0} + 3^{t-2} n_{e, 0} = 4 \cdot 3^{t-2} n_{e, 0}$$

即 (5) 式成立。

使用层次优先搜索算法和大度优先层次搜索算法可以证明下面的结论。

定理 2 当 $t \geq 3$ 时, 全边增长网络模型 $N(t)$ 的任意 2 棵具有最多叶子的生成树 $T_i^M(t)$ 和 $T_j^M(t)$ 拥有相同的叶子集合, 即 $L(T_i^M(t)) = L(T_j^M(t))$ 。

用记号 $G - L(G)$ 表示去掉图 G 的所有叶子后

得到的图。在层次优先搜索算法和大度优先层次搜索算法中,将语句“每一棵生成树 $T_0 \in F(k)$ ” 换为“每一棵生成树 $T_0 \in F^M(k)$ ”,将语句“每一棵生成树 $T_0 \in F(s)$ ” 换为“每一棵生成树 $T_0 \in F^M(k)$ ”,则可得到 $N(t)$ 的特殊的具有最多叶子生成树 $T^M(t)$, 这棵生成树满足:使得去掉 $T^M(t)$ 的所有叶子后得到的树是 $N(t-2)$ 的一棵具有最多叶子的生成树 $T^M(t-2)$ 即

$$T^M(t) - L(T^M(t)) = T^M(t-2)$$

继续下去,

$$T^M(s) - L(T^M(s)) = T^M(s-2)$$

是 $N(s-2)$ 的一棵具有最多叶子的生成树 $s=3, 4, \dots, t_0$ 。

定理 3 全边增长网络模型 $N(t)$ 包含一类特殊的具有最多叶子的生成树 $T^M(t)$ 使 $s=t, t-1, \dots, 4, 3$ 时,删去树 $T^M(s)$ 的叶子集合 $L(T^M(s))$ 中的所有叶子后得到的树 $T^M(s) - L(T^M(s))$ 是 $N(s-2)$ 的一棵具有最多叶子的生成树 $T^M(s-2)$, 见图 2。此外, $T^M(t)$ 的直径 $D(T^M(t))$ 满足

$$D(T^M(t)) \leq D(T^M(0)) + 2t - 3$$

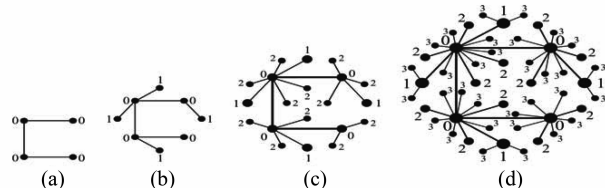


图 2 使用大度优先层次搜索算法得到图 1 中的全边增长网络模型的具有最多叶子的生成树

Fig. 2 The MLS-trees of totally edge-growing

network models in Figure-1 by applying

big-degree-first level searching algorithm

(a) $T^M(0)$; (b) $T^M(1)$; (c) $T^M(2)$; (d) $T^M(3)$

2 具有最多叶子生成树的应用

在 1998 年, Watts 和 Strogatz 介绍了小世界网的特性, 小世界网络是具有较高的聚集系数和较小的平均路径的稀疏网络。一般采用较小的平均路径来说明网络是小世界网络。

$N(t)$ 的小世界性 根据定理 3, 因为

$$D(N(t)) \leq D(T^M(t)) \leq D(T^M(0)) + 2t - 3$$

所以, $N(t)$ 是小世界的。

$N(0) = C_4$ 时, $N(t)$ 的具有最多叶子的无标度生成树 当初网络模型 $N(0) = C_4$ 时, 我们用大度优先层次搜索算法找到全边增长网络模型 $N(t)$

的一棵具有最多叶子的生成树 $T^M(t)$ ($t \geq 1$), 用记号 d 表示度, 用记 $n_d(t)$ 表示度为 d 的顶点的个数。 $T^M(t)$ 的度谱为

$$n_1(t) = 4 \cdot 3^{t-2}, n_3(t) = 3^{t-3}$$

当 $d = 13 \cdot 2^{t-4-r} - 4$ ($r = 0, 1, \dots, t-4$) 时, $n_d(t) = 3^r$;

当 $d = 7 \cdot 2^{t-3} - 2$ 时, $n_d(t) = 1$;

当 $d = 2^t$ 时, $n_d(t) = 1$ 。

定义 $N(t)$ 的一个节点标号函数 f 为: 若 $u \in V(0)$, $f(u) = 0$; 当 $j \in [1, t]$ 时, 若节点 $u \in X_j$, 并且边 $ux, uy \in Y_j$ 以及边 $xy \in Y_{j-1}$, 则 $f(u) = j$ 。由于 $T^M(t)$ 的度谱是离散型的, 则可以计算它的随机选择恰好 k 条边的节点的概率 $P(k)$ 。当

$$k = 13 \times 2^{t-4-\tau} - 4 \quad (0 < \tau < t)$$

时, 可解出

$$\tau - t = -[\ln(k+4)(\ln(13))/\ln 2]$$

有

$$P(k^* > k) = \frac{1}{n_{v,t}} \left[n_{v,0} + \sum_{i=1}^{\tau} |X_i| \right] = \frac{2 + 3^0 + 3^1 \dots + 3^{\tau-1}}{(3^t - 1)/2 + 2} \propto 3^{\tau-t} \quad (6)$$

以及

$$P(k^* > k-1) \propto 3 \times 3^{\tau-t}$$

进而得到

$$P(k) = P(k^* > k-1) - P(k^* > k) \propto 2 \cdot 3^{\tau-t} = 2 \cdot 3^{-[\ln(k+4) - \ln(13)]/\ln 2} \quad (7)$$

(7) 式说明, 具有最多叶子的生成树 $T^M(t)$ 服从幂律分布, 即 $T^M(t)$ 是 $N(t)$ 的一棵无标度生成树。

最小控制集 生成树被用于确定网络的最小控制集。

定义 1 设 S 是连通图 G 的一个顶点集合, 如果 G 的每一个顶点在 S 中或者与 S 中的一个顶点相邻, 则称 S 是连通图 G 的一个控制集。如果 G 的任何一个控制集 S' 均满足 $|S| \leq |S'|$, 则称 S 是连通图 G 的一个最小控制集。

定义 2 连通图 G 的非空顶点集合 S 是 G 的一个平衡集, 如果它满足

$$n_1(T) - |L(T \cap S)| = n_1(H) - |L(H) \cap S|$$

其中, T, H 为图 G 的任意两棵生成树, 并且 $L(T), L(H)$ 分别是树 T, H 的叶子集合。当 $S \neq V(G)$ 时, 则称 S 为真平衡集。

对连通图 G 的任意两棵具有最多叶子的生成树 T_i^M 和 T_j^M , 若总有

$$|L(T_i^M) \cap S| = |L(T_j^M) \cap S|$$

则称 S 为 G 的 T^M -平衡集。设连通图 G 的全体最多

叶子生成树 $T_1^M, T_2^M, \dots, T_p^M$ ，存在三个节点集合，

$$L_{\cap}(G) = \cap L(T_i^M),$$

$$L_{\cup}(G) = \cup L(T_i^M),$$

$$\bar{L}_{\cup}(G) = V(G)/L_{\cup}(G)$$

分别称 $L_{\cap}(G)$ 为 T^M -平衡交集, T^M -平衡并集, T^M -平衡余集。文献 [11] 讨论了无标度网络的平衡集，借助于此方法，我们有如下的定理。

定理 4 对 $t \geq 3$, 有全边增长网络模型 $N(t)$ 的 T^M -平衡交集 $L_{\cap}(N(t))$ 和 T^M -平衡并集 $L_{\cup}(N(t))$ 均等于 $X_{t-1} \cup X_t$ 。

3 结 论

本文给出全边增长网络模型 $N(t)$ 的构造方法，并讨论了全边增长网络模型 $N(t)$ 的具有最多叶子的生成树的基本性质。当初始网络为特殊网络时，我们确定了具有最多叶子的生成树 $T^M(t)$ 的小世界性和无标度性。本文的全边增长网络模型具有很好的性质：当 $t \geq 3$ 时，全边增长网络模型 $N(t)$ 的任意 2 棵具有最多叶子的生成树 $T_i^M(t)$ 和 $T_j^M(t)$ 拥有相同的叶子集合也就是说 $N(t)$ 为 $T^M(t)$ 保持全边增长网络，并得到这类图的刻画。证明了全边增长网络模型在每一时刻 t 的叶子的集合为 $X_{t-1} \cup X_t$ 。即对每一个 $N(t)$ 的节点都需要增加新的连线。如果考虑全边增长网络模型 $N(t)$ 的随机增加连线或者删除一些线，对一些网络模型的模拟会更好，但是，确定它们的性质以及找到最多叶子的生成树，将是研究实际网络连线的关键。此外，基于全边增长网 $N(t)$ 及其具有最多叶子的生成树 $T^M(t)$ 的无标度性，对于任何一个时刻 t 的网络模型 $M(t)$ ，且满足 $T^M(t) \subset M(t) \subset N(t)$ ，是 $M(t)$ 也具有无标度性？这些都是今后值得研究的问题。

参考文献：

[1] BARABASI A L, ALBERT R. Emergence of scale-free networks [J]. Science, 1999, 286: 509 – 512.

[2] MACDONALD P J, ALMAAS E, BARABASI A L. Minimum spanning trees of weighted scale-free networks [J]. Europhys Lett, 2005, 72 (2): 308 – 314.

[3] KATONA Z. Levels of a scale-free tree [J]. Random Structures and Algorithms, 2006, 29(2): 194 – 207.

[4] KIM D H, NOH J D, JEONG H. Scale-free trees: The skeletons of complex networks [J]. Physical Review E, 2004, 70(4): 046126.

[5] ZHANG Z Z, WU B, COMELLAS F. The number of spanning trees in Apollonian networks [J]. Discrete applied mathematics, 2014, 169: 206 – 213.

[6] COMELLAS F, FERTIN G, RASPAUD A. Recursive graphs with small-world scale-free properties [J]. Physical Review E, 2004, 69: 037104.

[7] ZHANG Z Z, ZHOU S G, FANG L J, et al. Maximal planar scale-free Sierpinski networks with small-world effect and power-law strength-degree correlation[J]. Europhysics Letters, 2007, 79: 38007.

[8] YAO B, YAO M, CHEN X E, et al. Research on edge-growing models related with scale-free small-world networks [J]. Applied Mechanics and Materials, 2014: 2444 – 2448.

[9] BONDY J A, MURTY U S R. Graph theory [M]. London and Basingstoke, New York: The Macmillan Press Ltd, 2008.

[10] BARABASI A L, BONABEAU E. Scale-free networks [J]. Scientific American, 2003, 288: 60 – 69.

[11] 刘霞, 姚东任, 姚兵, 等. 探讨无标度网络的平衡集 [J]. 中山大学学报(自然科学版), 2015, 54 (1): 19 – 23.