arXiv:2301.04131v1 [math.CO] 10 Jan 2023

关于随机有向图中几何出度分布的深度优先搜索中后向弧与前向弧的克努特猜想

斯万特·扬松

To Donald Knuth on his 85th birthday

Abstract. 唐纳德·克努特在即将出版的一卷著作 *The Art of Computer Programming*的草稿中，最近猜想：在具有几何出度分布的随机有向图中进行深度优先搜索时，后向弧与前向弧的数量服从相同的分布。

我们证明，这一猜想等价于两个由不同递推关系定义的生成函数相等。

遗憾的是，我们尚未能利用这一点证明该猜想；这个问题仍未解决。不过，我们希望这篇短文能激励其他人成功证明这一猜想。

1. Introduction

唐纳德·克努特 [2, 在《计算机程序设计艺术》即将出版的一卷草稿第 7.4.1 节中，研究了有向图中的深度优先搜索（DFS）。在该卷中，他针对有向图具有几何出度分布的随机情形提出了一个引人注目的猜想 [2, 问题 7.4.1.2-35] ；见下文第 2 节。精确计算有力地支持了这一猜想，表明它对小规模有向图成立（阶数为 $n \leqslant 9$；我们运用下文的方法将其扩展至 $n \leqslant 18$）。

我们尚未能证明这一猜想，但下文将证明（推论 3.3），该猜想等价于两个生成函数相等，这两个生成函数分别由相似但不同的递推关系定义。（我们还给出了这一猜想的扩展版本。）本文旨在阐述这一改写，希望借此激励其他人证明该猜想。

2. Knuth's conjecture

2.1. 深度优先搜索。关于有向图中深度优先搜索的详细说明以及历史注记，请参见 [2, 第 7.4.1 节2]。简而言之，深度优先搜索从任意一个顶点开始，逐一探索从该顶点出发的弧。当发现一条弧通向一个此前尚未访问过的顶点时，深度优先搜索便以相同方式递归地探索从该顶点出发的弧，然后再返回其父顶点的下一条弧。最终，这会生成一棵包含第一个顶点所有后代的树。如果仍有一些未访问的顶点，深度优先搜索就从其中一个顶点重新开始并生成一棵新树，如此反复，直至所有顶点都被访问。因此，深度优先搜索会在有向图中生成一个生成森林，称为深度优先森林。

Date: 2023年1月10日。由 Knutand Alice Wallenberg 基金会资助。1

舍弃弧（通向已访问顶点的弧）还可以进一步分类； [2,第 7.4.1 节2] 将有向图中的弧分为以下五种类型

- 环；
- 树弧，即深度优先搜索生成的树中的弧；
- 回边，即指向当前树中当前顶点的祖先的弧；
- 前向弧，即指向当前树中当前顶点已发现后代的弧；
- 交叉弧，即所有其他弧（这些弧指向一个已发现的顶点，该顶点既不是当前顶点的后代，也不是其祖先，并且可能位于另一棵树中）。

有向图中的每次深度优先搜索，都可以用按探索顺序排列的弧序列来描述。对于一次深度优先搜索，令L、F、B、C、T分别表示环、前向弧、回边、交叉弧和树弧的数量。

令 $F_n(w、x、y、z、t)$ 为如下生成函数：对顶点标记为 $1,\dots,n$的含有 n 个顶点的有向图，求和所有深度优先搜索实例对应的 $w^L x^F y^B z^C t^T$。为明确起见，我们假设深度优先搜索开始一棵新树时，会选择标号最小的未使用顶点作为根节点。（ [2, 7.4.1.2] 中的算法 D 选择标号最大的未使用顶点；这显然与我们的目的等价，但对我们来说符号表达不太方便。）

2.2. 几何出度分布的随机有向图。从现在起，我们考虑由以下模型给出的随机有向图上的深度优先搜索。

顶点数为 n；另给定参数 $p\in(0,1)$。每个顶点的出度是一个服从几何分布 $\mathsf{Ge}(1-p)$ 的随机变量；换言之，如果顶点 i 的出度为 η_i，那么

$$\mathbb{P}(\eta_i=k)=p^k(1-p),\qquad k=0,1,2,\dots; \tag{2.1}$$

此外，不同顶点的出度相互独立。这样，每个顶点 i 都有 η_i 条出弧；这些弧的另一端点从全部 n 个顶点中均匀随机选取，且各弧的选择相互独立。（因此，允许出现环和重弧，所以该有向图实际上是一个多重有向图。）

2.3. 猜想。唐纳德·克努特 [2, 问题 7.4.1.2-35] 猜想，对于具有几何出度分布的有向图，对每个 $n\geqslant 1$ 和每个 $p\in(0,1)$，随机变量 F 和 B 具有相同的分布，记作

$$F\overset{\mathrm{d}}{=}B. \tag{2.2}$$

等价地，用概率生成函数表示，

$$\mathbb{E}\,x^F=\mathbb{E}\,x^B. \tag{2.3}$$

（克努特通过显式计算验证了这一点，$n\leqslant 9$。）

2.4. 猜想与生成函数。给定一个描述某个深度优先搜索实例的弧序列，我们按如下方式求观察到的恰好是该序列的概率：对于序列中的每一条弧，发出该弧的顶点必须至少再发出一条弧，超出我们此前可能已观察到的数量（概率为 p），并且该弧的端点必须是一个指定的顶点（概率为 $1/n$）；此外，对于每个顶点，都有一次它本可以继续发出弧、却没有这样做的机会（概率为 $1-p$）。因此，该概率为 $(p/n)^{L+F+B+C+T}(1-$

$p)^n$。因此，在具有 n 个顶点且具有 $\eta \sim \mathsf{Ge}(1-p)$ 的随机有向图中，深度优先搜索的 $(L、\ F、\ B、\ C、\ T)$ 的联合概率生成函数为

$$\mathbb{E}\big[w^L x^F y^B z^C t^T\big] = (1-p)^n F_n\Big(w\frac{p}{n}, x\frac{p}{n}, y\frac{p}{n}, z\frac{p}{n}, t\frac{p}{n}\Big). \tag{2.4}$$

根据（2.4），克努特猜想（2.2）因此等价于

$$F_n\Big(\frac{p}{n}, x\frac{p}{n}, \frac{p}{n}, \frac{p}{n}, \frac{p}{n}\Big) = F_n\Big(\frac{p}{n}, \frac{p}{n}, x\frac{p}{n}, \frac{p}{n}, \frac{p}{n}\Big), \tag{2.5}$$

或者，先将 p/n 替换为 w，再将 xw 替换为 x，

$$F_n\big(w, x, w, w, w\big) = F_n\big(w, w, x, w, w\big). \tag{2.6}$$

事实上，我们提出一个更一般的猜想：对于每个 $n \geqslant 1$,

$$F_n\big(w, x, z, z, t\big) = F_n\big(w, z, x, z, t\big), \tag{2.7}$$

由相同的论证可知，这等价于

$$\mathbb{E}\big[w^L x^F z^{B+C} t^F\big] = \mathbb{E}\big[w^L x^B z^{F+C} t^F\big], \tag{2.8}$$

因此也等价于克努特猜想的以下扩展版本，其中涉及不同变量的联合分布。

猜想 2.1. 对每个 $n \geqslant 1$ 以及每个 $p \in (0、\ 1)$,

$$(L, F, B+C, T) \overset{\mathrm{d}}{=} (L, B, F+C, T). \tag{2.9}$$

注 2.2. 这个弱得多的命题，即 $\mathbb{E}\,F = \mathbb{E}\,B$（也就是说，期望相同），已在 [1, 定理 2.18] 中证明。文中还证明了$\mathbb{E}\,T = \mathbb{E}\,C$，但 T和 C的分布并不相同。□

注 2.3. 容易看出，(2.9) 不能推广到 $(L、\ F、\ B、\ C、\ T) \overset{\mathrm{d}}{=} (L、\ B、\ F、\ C、\ T)$。对于 $n = 3$，这一点就已经不成立；参见第3节中的论证，利用 (3.6) 计算 $G_3(w、\ x、\ y、\ z)$，并注意到 $G_3(w、\ x、\ y、\ z) \neq G_3(w、\ y、\ x、\ z)$。□

3. Recursion formulas

为求得 F_n，现在考虑在第一棵树完成时停止的深度优先搜索，并将 $G_m(w、\ x、\ y、\ z、\ t)$ 定义为：在一次深度优先搜索实例中生成第一棵树、且该树包含给定的 m 个顶点并按给定顺序访问这些顶点（例如，顶点 1、2、.....、m）的所有弧序列中 $w^L x^F y^B z^C t^T$的总和；对于仅构成完整深度优先搜索一部分的弧序列，我们也按上述方式定义 $L、\ F、\ B、\ C、\ T$。

一般而言，深度优先搜索会生成包含多棵树的森林。考虑深度优先森林包含 k 棵给定大小为$m_1, m_2, \ldots, m_k$的树，且顶点按给定顺序访问时，该情形对 F_n 的贡献。对第一棵树的遍历给出因子 $G_{m_1}(w, x, y, z, t)$。第二棵树的遍历方式与仅遍历该树相同，但也可能存在返回第一棵树任意 m_1 个顶点的交叉弧；事实上，每当可能出现环时，也可能出现这样的交叉弧，因此第二棵树的遍历产生因子 $G_{m_2}(w + m_1 z, x, y, z, t)$，其中将 w 替换为$w + m_1 z$，以计入可能增加的交叉弧。相同的推理适用于其后的所有树，因此 $F_n(w, x, y, z, t)$的贡献为 $\prod_{i=1}^{k} G_{m_i}(w + M_{i-1} z, x, y, z, t)$，其中 $M_j := \sum_{i \leqslant j} m_i$，从而 $F_n(w, x, y, z, t)$

等于所有可能的此类深度优先森林对应乘积之和。因此，对于每个 $m \geqslant 1$，式（2.7）成立的条件是

$$G_m\big(w,x,z,z,t\big) = G_m\big(w,z,x,z,t\big). \tag{3.1}$$

反过来，通过归纳可知，要使 (2.7) 对所有 n 都成立，(3.1) 是必要条件。在 G_m 的定义中，我们只考虑那些生成含有 m 个顶点的树的弧序列；因此树边的数量为 $T = m-1$。于是我们有

$$G_m(w,x,y,z,t) = t^{m-1} G_m(w,x,y,z,1). \tag{3.2}$$

为简化记号，我们记 $G_m(w,x,y,z) := G_m(w,x,y,z,1)$，并注意到 (3.1) 等价于

$$G_m\big(w,x,z,z\big) = G_m\big(w,z,x,z\big). \tag{3.3}$$

我们为 G_n 求得一个递推公式（为方便起见，现在用 n 替换 m）。对于 $n=1$，G_1 的定义中出现的弧序列，是根节点处零个或多个环组成的序列。因此，$F = B = C = 0$ 和

$$G_1(w,x,y,z) = \sum_{i=0}^{\infty} w^i = \frac{1}{1-w}. \tag{3.4}$$

对于 $n \geqslant 2$，假设深度优先搜索生成了一棵树 τ，其顶点为 $1,\dots,n$（按此顺序）。考虑根节点的最后一个子节点，设其标签为 $m+1$，其中 $1 \leqslant m \leqslant n-1$。令 τ' 为由 τ 中位于 $m+1$ 之前的顶点构成的子树，即 $1,\dots,m$；令 τ'' 为由 $m+1$ 及其后代构成的子树；它们的大小分别为 $|\tau'| = m$ 和 $|\tau''| = n-m$。对 τ 的遍历由以下部分组成

(1) 遍历 τ'；(2) 从根节点 1 到 $m+1$ 的树弧；(3) 遍历 τ''；(4) 返回根节点，并出现若干条仅指向已访问顶点的弧。

为了使一系列弧生成树 τ，只要我们在 τ' 内部进行遍历，就始终拥有与单独遍历树 τ' 时完全相同的可能性；但是在遍历 τ'' 的过程中，我们还可能遇到指向 τ' 中顶点的边；更准确地说，对于任何生成树 τ'' 的遍历，每当可能出现一个环时，在遍历 τ 的过程中还可能有一条指向根节点 1 的回边，或一条指向 τ' 中其他 $m-1$ 顶点之一的交叉边。这对应于在生成函数中将 w 替换为 $w+y+(m-1)z$。此外，在第 4 步中，我们可以有任意数量 $\geqslant 0$ 的弧从 1 指向自身（一个环），或指向 τ 中其他 $n-1$ 顶点之一（一条前向弧）。这使得生成函数具有一个因子

$$\sum_{i=0}^{\infty} (w+(n-1)x)^i = \frac{1}{1-w-(n-1)x}. \tag{3.5}$$

因此，对所有可能的 m 和树 τ' 及 τ''求和，我们得到，对于 $n \geqslant 2$,

$$G_n(w,x,y,z) = \frac{1}{1-w-(n-1)x} \sum_{m=1}^{n-1} G_m(w,x,y,z) G_{n-m}(w+y+(m-1)z,x,y,z). \tag{3.6}$$

我们证明了以下结论：

命题 3.1. 生成函数 G_n 满足递推关系 (3.6)，初始值为 (3.4)。

将参数取为 (3.3) 中的值，可得以下推论。这里

$$\widehat{G}_n(w,x,z) := G_n(w,x,z,z) \quad \text{and} \quad \breve{G}_n(w,x,z) := G_n(w,z,x,z) \tag{3.7}$$

分别是 $(L、\ F、\ B+C)$ 和 $(L、\ B、\ F+C)$ 的生成函数。

$\widehat{\ }$ $\breve{\ }$命题 3.2. 设 $G_n(w、\ x、\ z)$ 和 $G_n(w、\ x、\ z)$ 由以下递归关系定义：

$$\widehat{G}_1(w,x,z) := \breve{G}_1(w,x,z) := \frac{1}{1-w} \tag{3.8}$$

并且，对于 $n \geqslant 2$,

$$\widehat{G}_n(w,x,z) := \frac{1}{1-w-(n-1)x} \sum_{m=1}^{n-1} \widehat{G}_m(w,x,z)\widehat{G}_{n-m}(w+mz,x,z), \tag{3.9}$$

$$\breve{G}_n(w,x,z) := \frac{1}{1-w-(n-1)z} \sum_{m=1}^{n-1} \breve{G}_m(w,x,z)\breve{G}_{n-m}(w+x+(m-1)z,x,z). \tag{3.10}$$

当且仅当下式成立时，(2.9)（或等价地，(2.7)）成立：

$$\widehat{G}_n(w,x,z) = \breve{G}_n(w,x,z), \qquad n \geqslant 1. \tag{3.11}$$

$\widehat{\ }$ $\breve{\ }$证明。首先，由 (3.7) 和 (3.6)，G_n 和 G_n 满足 (3.8)–(3.10)，因此可以由这些递归关系定义。

此外，我们还证明了 (2.9) $\iff$ (2.7) $\iff$ (3.1) $\iff$ (3.3)，并且由 (3.7) 可得 (3.3) $\iff$ (3.10)。□

为了证明 (2.6)，即克努特猜想，只需验证 $z = w$ 的 (3.11)。换言之：

推论 3.3. 克努特猜想 $F \overset{\mathrm{d}}{=} B$，或者等价地说 (2.6)，等价于

$$\widehat{G}_n(w,x,w) = \breve{G}_n(w,x,w), \qquad n \geqslant 1, \tag{3.12}$$

$\widehat{\ }$ $\breve{\ }$其中 G_n 和 G_n 由递归关系 (3.8)–(3.10) 给出。

问题 3.4. 证明 (3.11)，或者至少证明 (3.12)！

注 3.5. 克努特通过对 $n \leqslant 9$ 进行精确计算验证了他的猜想。我们通过验证 $n \leqslant 18$ 的 (3.11)（从而也验证了 (2.7)），将这一结果进一步推广。□

$\breve{\ }$注 3.6。$G(w、\ x、\ z)$ 的另一个递推关系为

$$\breve{G}_n(w,x,z) := \frac{1}{1-w} \sum_{m=1}^{n-1} \breve{G}_m(w+x,x,z)\breve{G}_{n-m}(w+mz,x,z). \tag{3.13}$$

其推导方式与上述递推关系类似，但现在定义 τ' 为由根节点的第一个子节点（即 2）及其所有后代组成的子树，并定义 τ'' 为由所有其他顶点（包括根节点 1）组成的子树。对 τ 的遍历包括

(1) 根节点 1 上的环数为 $\geqslant 0$，(2) 从 1 到 2 的树弧，(3) 遍历 τ'，(4) 返回根节点并遍历 τ''。

步骤 (1) 在生成函数中产生因子 $(1-w)^{-1}$。步骤 (3) 相当于遍历树 τ'，但每次我们可能遇到一个环，也可能遇到一条指向 1 的回边；这对应于在生成函数中用 $w+y$ 替换 w。类似地，在步骤 (4) 中，每次我们可能遇到一个环，也可能遇到一条指向 τ' 中某个顶点的边；这要么是前向边（如果当前位于根节点），要么是交叉边；这对应于在生成函数中用 $w+mz$ 替换 w，其中 $m:=|\tau'|$。综合起来，得到 (3.13)。

注意，尽管看起来也可以推出对于递推关系，$\widehat{G}_n(w、x、y、z)$，也可以使用这里采用的分解方法，但这样会更复杂，因为我们还必须持续跟踪从τ'' 到 τ' 的边是前向边还是交叉边。这部分留给读者完成。□

References

[1]菲利普·雅凯和斯万特·扬松。《随机有向图中几何出度分布下的深度优先搜索性能》。预印本，2022年。arXiv:2212.14865[2] 唐纳德·克努特。《计算机程序设计艺术》，第 7.4.1.2 节（初步草稿，2022年2月13日）。http://cs.stanford.edu/~knuth/fasc12a.ps.gz

Department of Mathematics、 Uppsala University、 PO Box 480、 SE-751 06 Uppsala、Sweden*Email address*: svante.janson@math.uu.se*URL*: http://www2.math.uu.se/~svante/