最小割的伪确定性算法

Aryan Agarwala
马克斯·普朗克信息学研究所，萨尔州信息学园区，德国
Nithin Varma
科隆大学，
德国

摘要

本文提出了适用于全局最小割问题和最小 s-t 割问题的高效伪确定性算法。对于全局最小割问题，我们算法的运行时间渐近优于目前最快的顺序确定性全局最小割算法（Henzinger、Li、Rao、Wang；SODA 2024）。此外，我们还在顺序、流式处理、并行随机存取机和割查询模型中实现了该算法；在这些模型中，目前尚无高效的确定性全局最小割算法。

2012年 ACM 学科分类计算数学 → 图算法；计算理论 → 流式、次线性与近线性时间算法；计算理论 →伪随机性与去随机化

关键词与短语最小割、伪确定性算法

数字对象标识符10.4230/LIPIcs...94

致谢作者感谢 Danupon Nanongkai 进行了深入讨论。

arXiv:2512.23468v1 [cs.DS] 29 Dec 2025

1 引言

随机化是算法设计的基石之一，在为各种计算问题设计简单快速的算法方面行之有效。另一方面，确定性算法的优势在于其可复现性：对相同输入多次运行同一算法，都会得到相同输出。随机性是否能带来相较于确定性对应算法可证明更快的算法，是理论计算机科学领域的一个根本性问题。

Gat 和 Goldwasser [10] 发起的对伪确定性的研究，对上述研究方向产生了影响 [14, 30]。针对搜索问题的伪确定性算法是一种随机化算法，它以高概率输出相同的答案。对于全局最小割问题，随机化算法只需以高概率输出某个 最小割，而伪确定性算法则需要以高概率输出同一个 最小割。形式化地说，若对所有输入 x，算法 $\mathcal{A}$ 以概率 $\rho \geq \frac{2}{3}$ 满足 $\mathcal{A}(x) = s(x)$，其中$s(x)$ 是 x 的典范解，则称算法为伪确定性的。伪确定性算法兼具确定性算法的可复现性，以及随机化算法的高效与简洁，从而集两者之长。伪确定性算法已被用于构造素数 [35, 7, 34]，二分图匹配 [15]，非二分图匹配 [2]，拟阵交 [12]，无向连通性 [22]，等问题。哪些问题存在高效的伪确定性算法，是一个重要的开放方向。

© Aryan Agarwala 和 Nithin Varma；依据知识共享署名 4.0 国际许可协议授权 莱布尼茨国际信息学会议录 达格施图尔城堡 – 莱布尼茨信息学中心，达格施图尔出版社，德国
LIPICS

1.1 伪确定性全局最小割算法。

在这项工作中，我们研究了如何为一个备受研究的问题——全局最小割——设计伪确定性算法。全局最小割问题的输入是一个带权无向图 G，包含 n 个顶点和 m 条边；输出则是对 G 中顶点的一个划分，使跨越该划分的边的总权重最小。这个问题与最小 s-t 割问题密切相关，后者的输出是对G 中顶点的一个划分，使顶点 s 和 t 位于划分的不同侧。朴素地，全局最小割问题可以通过调用最小 s-t 割算法 $n-1$ 次来求解 [18]。因此，人们最初认为全局最小割是最小s-t 割问题的一个更难变体。后来，Nagamochi 和 Ibaraki [33] 以及 Hao 和 Orlin [23] 证明，全局最小割可以在与当时已知的最佳最小 s-t割算法相匹配的运行时间内求解。随后，Karger 开创了全局最小割问题的随机化 算法研究方向 [25, 28, 26],，最终得到了一种运行时间为 $O(m\log^3 n)$ 的算法，其速度甚至在 25 年后仍快于最佳的最小 s-t 算法。因此，全局最小割成为经典例子，展现了随机性如何带来显著更快、更简单的算法。Gawrychowski 等人 [11] 改进了 Karger 算法中的一个对数因子，给出了一个运行时间为 $O(m\log^2 n)$ 的全局最小割随机化算法。最近，Henzinger 等人 [24] 得到了一个运行时间为 $\tilde{O}(m)$[1] 的全局最小割确定性算法。值得注意的是，他们的算法虽然在运行时间上与随机化算法相差至多多对数因子，但其对数项的指数似乎较高，而且远比 Karger 的近线性时间算法复杂 [26]。

我们提出了一种简单高效的全局最小割问题伪确定性算法，它将任意随机化全局最小割算法用作黑盒。

▶**定理 1.** 考虑一个随机化算法 A，其输入为一个简单图$G=(V,E)$，边权重由 $w\colon E\to\mathbb{Z}^+$ 给定，运行时间为 $t(n,m)$，并以至少$2/3$的概率输出 G 关于 w的一个最小割。那么，存在一个伪确定性算法 A'，其输入为 $G=(V,E,w)$，运行时间为 $O(t(n,m+n)\log(n)\log\log(n))$，并以至少$2/3$的概率输出$G$的一个特定的全局最小割。

我们的结果与 Gawrychowski 等人 [11], 的算法结合后，立即得到一个运行时间为$O(m\log^3 n\log\log n)$的伪确定性全局最小割算法，其速度显著快于目前最佳的确定性算法。我们算法的相对简洁性也是一项额外优势。

1.2 流式处理、割查询与并行模型。

自伪确定性这一概念提出以来，人们已在许多场景中对其进行了研究，例如查询复杂度 [14, 6, 17], 流式算法 [21, 4, 16], 并行算法 [15, 2, 13], 空间受限计算 [22], 等[2]。本文为我们的伪确定性全局最小割算法在流式处理、割查询和并行模型中提供了高效实现。我们的伪确定性算法比相应模型中已知的最佳确定性算法高效得多。事实上，正如我们将在下文讨论的,

[1] $\tilde{O}$ 隐藏了 polylog(n) 因子。[2] 若想更全面地了解该领域，读者可参阅 Grossman 的博士论文 [20]。

在所有这些模型中，随机化算法与确定性算法之间都存在巨大差距，这与顺序模型不同。

1. **流式处理**：在这种模型中，输入图由一连串对边及其相应权重进行插入或删除操作组成。具体而言，数据流中的每个元素要么形如 插入 (e, w_e)，其中 $e \in E$ 且 $w_e \in \mathbb{Z}^+$；要么形如 删除 e。请注意，我们考虑的是简单图的流式处理，因此，一条边插入后，在被删除之前不能再次插入。典型目标是尽可能减少空间用量和对该数据流的遍数，以计算全局最小割。

    Mukhopadhyay 和 Nanongkai [31] 给出了一种用于求全局最小割的随机化精确流式算法，使用 $O(n)$ 空间和 $\log(n)$ 遍。就我们所知，目前还没有使用 $O(n^{1.99})$ 空间且只需 $O(1)$ 遍的该问题确定性算法。

    ▶**定理 2**。 存在一种用于求全局最小割的伪确定性流式算法，它对输入进行 $O(\log^2 n)$ 遍处理，并使用 $\tilde{O}(n)$ 空间。

2. **割查询**：在此模型中，只能通过预言机访问图：输入为将顶点划分为两部分的一个划分，输出为跨越该划分的边的总权重。目标是尽可能少地向该预言机发起查询，以计算全局最小割。

    Mukhopadhyay 和 Nanongkai [31]证明了一种随机精确算法，只需使用 $O(n)$ 次割查询即可求得全局最小割。据我们所知，目前最好的确定性算法是 Grebinski 和 Kucherov [19]提出的算法的推论；该算法通过 $O(n^2/\log\ n)$ 次查询重构整个图。

    ▶**定理3.** 存在一种伪确定性全局最小割算法，只需进行 $O(n)$ 次割查询。

3. **并行**：我们考虑并行随机存取机（CREW）并行随机存取机）模型，其中有 p 个处理器连接到共享随机存取存储器。任意数量的处理器都可以读取同一个存储单元，而在单个时间步内，只有一个处理器可以向该存储单元写入数据。

    在这一模型中，Anderson 和 Blelloch [3] 给出了一种用于全局最小割的随机精确算法，其工作量为 $O(m)$，深度为 $O(1)$。据我们所知，该问题唯一的确定性 NC 算法由 Karger 和 Motwani [27] 提出。尽管他们没有明确计算该算法的工作量，但它肯定是一个极大的多项式，而且似乎甚至比方说 $O(n^{10})$ 还要差。

    ▶**定理 4**。 存在一种伪确定性全局最小割算法，在 并行随机存取机模型中，工作量为 $O(m)$，深度为 $O(1)$。

1.3 我们的技术

我们算法中的一项关键工具建立在隔离引理之上。Mulmuley、Vazirani 和 Vazirani [32]，提出的隔离引理指出，对于任何基于基集 E 构造的对象族，例如二分图的完美匹配，如果为 E 中的每个元素随机赋予有多项式界的整数权重，那么最小权重的对象，即完美匹配，以高概率是唯一的。此后，对隔离引理进行去随机化一直是一个重要的开放问题，相关研究已有大量积累（例如参见 [5, 8, 9, 37, 1]）。

Goldwasser 和 Grossman [15], 在其二分图完美匹配伪确定性 NC 算法中，提出了一种伪确定性算法，用于为二分图分配边权重，使得最小权完美匹配以高概率是唯一的。随后，

他们使用随机 NC 算法，在输入图上采用构造出的边权重求解最小权完美匹配，从而得到二分图完美匹配的伪确定性 NC 算法。Anari 和 Vazirani [2]对非二分图匹配也采用了相同的方法，Ghosh 等人 [13] 则将其用于线性拟阵交。

在这项工作中，我们采用了与 [15] 的算法类似的路线。我们的伪确定性算法以一个加权图为输入，并对权重进行扰动，使图的全局最小割唯一。我们指出，[15, 2] 中以这种方法为基础的先前算法，比相应的随机化算法慢一个多项式因子。因此，我们的算法是首个在保留计算效率的同时采用这种方法的实例。

我们针对所讨论的四种模型（顺序、流式处理、割查询、并行）实现的算法，在所有这些模型中的效率损失都仅为对数因子。因此，将我们的算法与 [11, 31, 3]，中的随机化算法相结合，我们便能在多种计算模型中获得近乎最优的全局最小割伪确定性算法。此外，尽管隔离引理此前已应用于并行 [32, 8, 9, 37]，空间受限 [36, 38]，催化式 [1, 29] 和顺序 [32] 模型，但据我们所知，这是隔离首次应用于流式处理和割查询模型。

我们指出，我们的精确方法也可以为更一般的次模函数最小化问题给出类似的黑盒陈述。为求简洁，本文不对此作明确说明。

1.4 组织结构

本文组织如下。第 2 节介绍全文所用的基本符号和定义。第 3 节给出一个用于求解最小 s-t 割问题的伪确定性算法，并介绍本文采用的一些关键思路。第 4 节介绍我们的伪确定性全局最小割算法及其分析。最后，第 5 节详细介绍该算法在流式处理、割查询和并行模型中的实现。

2 预备知识

本节介绍一些基本符号和定义。设 $G = (V, E)$ 是一个无向图，其中 $V = \{1, \ldots\ldots, n\}$。设 $\emptyset \subsetneq S \subsetneq V$ 是顶点的一个子集。这样的顶点集称为该图的一个割集。由 S 形成的割是集合

$$\mathrm{cut}_G(S) = \{e \in E \mid e = (u, v), u \in S, v \notin S\}.$$

割集 S 和 $V \setminus S$ 都表示割$_G$(S)；因此，我们有时将这个割记为 (S, $V \setminus S$)。

设 $w : E \to \mathbb{Z}^+$ 为一个边权赋值。对于子集 $E' \subseteq E$，我们用 $w \restriction_{E'}$ 表示权函数 w 在集合 E'上的限制。割 (S, $V \setminus S$) 的权重 （记为w(割$_G$(S))）等于割$_G$(S)中各边的权重之和。如果割 (S, $V \setminus S$) 的权重等于 $\min^{w}_{\emptyset \subsetneq S \subsetneq V}$(割$_G$($S$))，则称其为 G 的全局最小割 。

▶**事实 5.** 上文定义的函数 w(割$_G(\cdot)$) : $2^V \to \mathbb{Z}^+$ 是一个次模函数。也就是说，对于集合 $S, T \subseteq V$，有 w(割 $_G(S \cap T)) + w$(割 $_G(S \cup T)) \leq w$(割 $_G(S)) + w$(割 $_G(T))$。

令 $s,t \in V$为特殊的源点和汇点。割 $(S, V\setminus S)$ 是一个 s-t 割，当且仅当 $s \in S$ 且 $t \notin S$。G 的一个 s-t 割 $(S, V \setminus S)$ 是一个最小 s-t 割，当且仅当其权重等于 $\min_{\emptyset \subsetneq S \subsetneq V, s\in S, t\notin S} w(\text{割}_G(S))$。

现在我们引入拼接权重的概念，它使我们能够在保持整数性的同时扰动权重。这一方法已用于许多先前的工作（例如 [9, 2, 15, 37]）。

▶ **定义 6**（拼接权重）。设 $G = (V, E)$ 是一个图，$w, w' : E \to \mathbb{Z}^+$ 是两个边权赋值。设 $|w'|$ 为满足 $|w'| \geq \sum_{e\in E} w'(e)$ 的任意值。我们将拼接 $[w, w'] : E \to \mathbb{Z}^+$ 定义为如下边权赋值：

$$[w, w'](e) = (|w'| + 1)w(e) + w'(e)$$

为简化记号，在提及子集$A \subseteq E$的拼接权重时，我们可以用元组 $(w(A)\ w'(A))$ 表示 $[w, w'](A)$。这种记号很有用，因为拼接权重的排序恰好与元组 $(w,\ w')$ 的字典序一致。

▶ **观察 7**。令 $A,\ B \subseteq E$，则：

1. $[w, w'](A) \leq [w, w'](B) \iff w(A) < w(B)$ 或 $\{w(A) = w(B)$ 且 $w'(A) \leq w'(B)\}$，以及
2. $[w, w'](A) = [w, w'](B) \iff w(A) = w(B)$ 且 $w'(A) = w'(B)$。

观察 7 立即推出以下事实。

▶ **事实 8**。令 $G = (V, E)$ 为一个图，且 $w, w' :\ E \to \mathbb{Z}^+$ 为边权赋值。在 $[w, w']$ 权重赋值下，G 的任何 *(s-t)* 最小割也是 w权重赋值下 G 的一个 *(s-t)* 最小割。

▶ **定义 9** (E^s_{star})。令 V 为一个图的顶点集，$s \in V$ 为一个源顶点。边集 E^s_{star} 定义为顶点集 V 上以顶点 s 为中心的星图所对应的边的集合。也就是说，$E^s_{\text{star}} = \{(s, v) \mid v \in V, v \neq s\}$。

▶ **定义 10** (w_s)。设 $G = (V, E)$ 为一个图，$s \in V$ 为一个源顶点。边权赋值 w_s：$E \to \mathbb{Z}^+$ 定义为 $e \mapsto \mathbb{I}[e \in E^s_{\text{star}}]$。也就是说，

$$w_s(e) = \begin{cases} 1, & \text{if } e \in E^s_{\text{star}} \\ 0, & \text{otherwise.} \end{cases}$$

注意，$n \geq \sum_{e\in E} w_s(e)$。因此，在拼接 w_s 时（参见定义 6），我们可以将 $|w_s|$ 设为 n。

3 预备知识：伪确定性最小 s-t 割

本节中，我们介绍用于求解最小 s-t 割问题的伪确定性算法，并证明以下定理。具体而言，我们给出一种将随机化算法转化为伪确定性算法的通用变换。

▶**定理 11**。设 A 是一个随机化算法，满足：
1. 以简单图 $G = (V, E)$ 及正整数边权$w : E \to \mathbb{Z}^+$作为输入，并且2. 输出一个 s-t 割 (S, T)，该割属于 G，且满足

$$\Pr[(S, T) \text{ is a minimum } s\text{-}t \text{ cut of } G \text{ under edge weights } w] \geq \rho$$

那么，算法$\text{PD}^{A}_{s,\ t}$ 是一种伪确定性算法，满足：

1. 输入一个简单图 $G=(V,E)$ 以及正整数边权w： $E\to\mathbb{Z}^+$，并且2. 存在一个固定的 G 的最小 s-t 割 (S^*,T^*)，满足 $\Pr[\mathrm{PD}_{s,t}^{\mathcal{A}}(G,w)=(S^*,T^*)]\geq\rho$。

算法 1 用于求解最小 s-t 割的伪确定性算法 $\mathrm{PD}_{s,t}^{\mathcal{A}}$

输入： $G=(V,E)$ 是边权重为 $w: E\to\mathbb{Z}^+$ 的输入图；可访问用于求解加权最小 s-t 割的随机化算法 $\mathcal{A}$ 的预言机
输出： 最小 s-t 割 (S,T)
1: 通过 $w(e)=0$ 将 w 扩展到定义域 $E\cup E_{\mathrm{star}}^s$，对所有 $e\in E_{\mathrm{star}}^s$ 均成立。2: 在边权重为 $[w,w_s]$ 的 $G'=(V,E\cup E_{\mathrm{star}}^s)$ 上运行 $\mathcal{A}$。设返回的割为 (S,T)。3: 输出 (S,T)。

以下命题仅指出，向图中添加权重为 0 的边不会改变任何割的值。

▷ 命题 12. 设 $G'=(V,E\cup E_{\mathrm{star}}^s)$ 是一个边权重为 w 的图： $E\cup E_{\mathrm{star}}^s\to\mathbb{Z}^+$ 满足 $w(e)=0\ \forall e\notin E$。那么，$(S,T)$ 是 G' 的最小 s-t 割，当且仅当 (S,T) 是 $G=(V,E)$ 中边权重为 $w\restriction_E$ 时的最小 s-t 割。

证明。 只需注意，对于任意割 (S,T)，在 G' 中，

$$w(\mathrm{cut}_{G'}(S))=\sum_{e\in\mathrm{cut}_{G'}(S)\cap E}w(e)+\sum_{e\in\mathrm{cut}_{G'}(S)\setminus E}w(e)=\sum_{e\in\mathrm{cut}_{G'}(S)\cap E}w(e)=\sum_{e\in\mathrm{cut}_G(S)}w\restriction_E(e)$$
$$=w\restriction_E(\mathrm{cut}_G(S))$$

因此，由于在 G' 中，边权重为 w 时的每个割与在 G 中，边权重为 $w\restriction_E$ 时对应的割权重完全相同，所以 G' 中权重为 w 和 G 中权重为 $w\restriction_E$ 时的最小割完全相同。◀

▶ **事实13.** 设 $G=(V,E)$ 是一个边权重为 $w: E\to\mathbb{Z}^+$ 的图，源顶点为 s，汇点为 t。设 $S\subseteq V$ 和 $S'\subseteq V$ 是两个割集，在边权重为 w 时，它们都对应于 G 中的最小 s-t 割，且 $s\in S'\cap S$。那么， $S\cup S'$ 也是一个割集，在边权重为 w 时，它对应于 G 中的最小 s-t 割。

证明。次模函数的一个简单性质是， $w(\text{割}_G(S\cap S'))+w(\text{割}_G(S\cup S'))\leq w(\text{割}_G(S))+w(\text{割}_G(S'))$。由于 S和 S'都是 $w(\text{割}_G(S))$的极小化元，左侧两项也必定都是极小化元。因此， $S\cup S'$ 也是对应于在边权重 w下 G 的最小割的割集。◀

▷ 命题14。设 $G'=(V,E\cup E_{\mathrm{star}}^s)$ 是一个边权重为 $w: E\cup E_{\mathrm{star}}^s\to\mathbb{Z}^+$ 的图。令$s\in V$为固定的源顶点。对于任意 $t\in V\setminus\{s\}$，在边权重为 $[w,w_s]$的 G' 中，存在唯一的 s-t 最小割。

证明。 固定某个 $t\in V\setminus\{s\}$。为反证起见，假设在边权重为 $[w,w_s]$ 时， G' 存在多个权重最小的 s-t 割。令这些割为 (S_1,T_1) 和 (S_2,T_2)。假设它们的权重为 (W,W')。注意 $W'=|T_1|=|T_2|$。现在考虑割 $(S_1\cup S_2,T_1\cap T_2)$。

根据事实 8，我们知道 (S_1,T_1) 和 (S_2,T_2) 也都是 G在边权重 w下的最小割。由此，根据事实13，$(S_1\cup S_2,T_1\cap T_2)$ 也是 G在边权重 w下的最小割。因此，

$$w(\text{割}_G(S_1\cup S_2))=W.$$

此外，

$$w_s(\mathrm{cut}_G(S_1 \cup S_2)) = |T_1 \cap T_2| < |T_1| = |T_2| = w_s(\mathrm{cut}_G(S_1)) = w_s(\mathrm{cut}_G(S_2)).$$

因此，我们有

$$[w, w_s](\mathrm{cut}_G(S_1 \cup S_2)) < [w, w_s](\mathrm{cut}_G(S_1)) = [w, w_s](\mathrm{cut}_G(S_2))$$

这与 (S_1, T_1) 和 (S_2, T_2) 的极小性矛盾。证明至此完成。 ◀

定理 11 的证明。 考虑 $G' = (V, E \cup E^s_{\mathrm{star}})$ 其边权重为 $[w, w_s]$。根据命题14，G' 权重为 $[w, w_s]$ 的图有唯一的最小 s-t 割。记其为 (S^*, T^*)。根据命题 12 和事实 8，(S^*, T^*) 是 G 在权重 w下的最小割。

现在，只需注意到 $\mathrm{PD}^{\mathcal{A}}_{s,t}(G, w) = \mathcal{A}(G', [w, w_s])$。以 $\geq \rho$ 的概率，输出$\mathcal{A}(G', [w, w_s])$ 是 G' 在权重 $[w, w_s]$ 下的一个最小割。唯一这样的最小割是 (S^*, T^*)。因此，以 $\geq \rho$ 的概率，输出 $\mathrm{PD}^{\mathcal{A}}_{s,t}(G, w) = (S^*, T^*)$。证明至此完成。 ◀

▶注 15. 据我们所知，在所有模型中，效率最高的最小 s-t 割算法都使用 s-t 最大流算法。给定一个 s-t 最大流，只需找出由最大流所形成的残量图中从 s 可达的顶点构成的割集 S，即可轻松得到一个规范的最小 s-t 割。因此，我们转而关注全局最小割，因为已知存在远比最大流算法高效的算法。

4 伪确定性全局最小割

I本节中，我们为全局最小割问题设计伪确定性算法 .
S具体而言，我们证明如下结论。

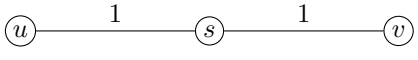
▶**定理 16.** 设 A 是一个随机化算法，它满足：
1. 输入一个简单图 $G = (V, E)$ 及其正整数边权$w : E \to \mathbb{Z}^+$，并且**2.** 输出一个割 (S, T)，其属于 G，满足

$$\Pr[S \textit{ is a minimum cut of } G \textit{ under edge weights } w] \geq \rho$$

那么，算法$\mathrm{PD}^{\mathcal{A}}_{\mathrm{global}}$ 是一种伪确定性算法，满足：
1. 输入一个简单图 $G = (V, E)$ 以及正整数边权w： $E \to \mathbb{Z}^+$，并且**2.** 存在一个固定的最小割 (S^*, T^*) 属于 G，使得

$$\Pr[\mathrm{PD}^{\mathcal{A}}_{\mathrm{global}}(G, w) = (S^*, T^*)] \geq 1 - (1 - \rho) \cdot O(\log n).$$

就我们的目的而言，全局最小割的结构比最小 s-t 割稍微复杂一些。下面的简单路径示例说明了为什么我们针对最小 s-t 割的方法无法直接奏效。



只需注意，$\{s, v\}$ 和$\{s, u\}$ 都是变换$G' = (V, E \cup E^s_{\mathrm{star}})$在边权重 $[w, w_s]$下的全局最小割。

下面的引理揭示了这些反例的结构。

▶ **引理 17.** 设 $G=(V,E)$ 是一个边权重为 $w\colon E\to\mathbb{Z}^+$ 的简单图。设 $s\in V$ 是一个特殊的源顶点。考虑变换图 $G'=(V,E\cup E^s_{\text{star}})$，其边权重为 $[w,w_s]$。设 $\{(S_1,T_1)\dots,(S_k,T_k)\}$ 是 G' 在边权重 $[w,w_s]$ 下的全局最小割，且满足 $s\in S_i\ \forall i\in[k]$。那么：

1. 对于所有 $i,\ j\in[k]$ 满足 $i\neq j$，有 $T_i\cap T_j=\emptyset$。
2. 对于所有 $i,\ j\in[k]$，有 $|T_i|=|T_j|$。

证明。 命题14意味着这些 T必定两两不相交。否则，设 $(S_i,\ T_i)$ 和 $(S_j,\ T_j)$都是满足 $s\in S_i\cap T_j$ 且 $t\in T_i\cap T_j$的最小割。它们都是最小 s-t割，这与最小 s-t 割的唯一性矛盾。

现在，考虑任意的 $(S_i,\ T_i)$ 和 $(S_j,\ T_j)$。注意到，由于它们都是最小割，因此相对于 w 和 w_s，它们的权重相同（由观察 7 可知）。因此，$w_s(\text{割}_{G'}(S_i))=w_s(\text{割}_{G'}(S_j))\implies|T_i|=|T_j|$。证明完毕。◀

为了利用这一性质，我们现在引入第二种边集和权重赋值。

▶ **定义 18** ($w_s^{<i}$)。设 $G=(V,E)$ 为一个图，且 $s\in V$ 为一个源顶点。权重赋值 $w_s^{<i}$: $E\to\mathbb{Z}^+$ 定义为

$$w_s^{<i}(e)=\begin{cases}1, & \text{if } e=(s,v)\ \text{and}\ v<i\\ 0, & \text{otherwise.}\end{cases}$$

注意到 $n\geq\sum_{e\in E}w_s^{<i}(e)$。因此，在拼接 $w_s^{<i}$时，我们可以设定 $|w_s^{<i}|=n$ （见定义 6）。

现在我们需要使用三种权重赋值而不是两种，因此引入重复拼接的概念。

▶ **定义 19** (重复拼接)。设 $G=(V,E)$ 为一个图，且 $w_1,w_2,w_3\colon\ E\to\mathbb{Z}^+$为边权重。权重赋值 $[w_1,w_2,w_2]$ 定义为重复拼接$[[w_1,w_2],w_3]$。

▷ 断言 20。设 $G=(V,\ E)$ 为一个带有边权重 w 的图，且设 s 为一个源顶点。设 $G'=(V,\ E\cup E^s_{\text{star}})$，并设 $\{(S_1,\ T_1)\dots,\ (S_k,\ T_k)\}$ 为根据权重 $[w,w_s]$ 在 G' 中的全局最小割集合，使得 $s\in S_i\ \forall i\in[k]$。此外，设$\mathcal{T}=\bigcup_{i\in[k]}T_i=\{t_1,\ t_2,\ \dots,\ t_m\}$。

对于任意 $x\in V$:
1. 若 $x>t_m$，则在权重 $[w,w_s,w_s^{<x}]$（其中 $s\in S$）下，G' 的每个最小割 (S,T) 都满足 $w_s^{<x}(\text{割}_{G'}(S))=|T|$。 2. 若 $x\leq t_{m-1}$，则在权重 $[w,w_s,w_s^{<x}]$（其中$s\in S$）下，G' 的每个最小割 (S,T) 都满足 $w_s^{<x}(\text{割}_{G'}(S))<|T|-1$；或者，$G'$ 在权重 $[w,w_s,w_s^{<x}]$ 下有多个最小割（也可能两者皆是）。3. 若 $t_{m-1}<x\leq t_m$，则在权重 $[w,w_s,w_s^{<x}]$ 下，G' 的最小割 (S,T) 是唯一的，且满足 $w_s^{<x}(\text{割}_{G'}(S))=|T|-1$。此外，$(S,T)$ 恰好是满足 $t_m\in T_i$ 的割 (S_i,T_i)。

证明。 因此，根据引理 17，所有 T_i 两两不相交且大小相同。根据事实 8，任意 （S，T）在 G' 权重 $[w,w_s,w_s^{<x}]$下的最小割，也是 G' 在权重 $[w,w_s]$下的最小割。因此，

$(S,\ T)\ =(S_i,\ T_i)$ 对某个 $i\in[k]$ 成立，更重要的是 $w_s^{<x}(S)\ =\min_{i\in[k]} w_s^{<x}(S_i)$。

对于任意割 S_i，都有 $w_s^{<x}(\text{cut}_{G'}(S_i))=|T_i\cap\{v\in V\mid v<x\}|=|T_i\setminus\{v\in V\mid v\geq x\}|=|T|-|\{v\in T\mid v\geq x\}|$。

现在，我们分别处理各种情况：
1. 如果 $x>t_m$，则有 $|T\cap\{v\in V\mid v\geq x\}|=\emptyset$，这意味着 $\forall i\in[k], w_s^{<x}(\text{cut}_{G'}(S_i))=|T_i|$。因此，$w_s^{<x}(\text{cut}_{G'}(S))=|T|$。2. 如果 $x\leq t_{m-1}$，则有 $|T\setminus\{v\in V\mid v\geq x\}|\geq 2$。a. 如果存在 $i\in[k]$ 使得 $T_i\cap\{v\in V\mid v\geq x\}\geq 2$，那么 $w_s^{<x}(\text{cut}_{G'}(S))\leq w_s^{<x}(\text{cut}_{G'}(S_i))\leq|T_i|-2<|T_i|-1=|T|-1$。因此，我们回到了第一种情况。b. 如果反之有 $\forall i\in[k], |T_i\cap\{v\in V\mid v\geq x\}|\leq 1$，那么由于这些 T_i 互不相交，必定存在 $i,j\in[k]$ 使得 $|T_i\cap\{v\in V\mid v\geq x\}|=|T_j\cap\{v\in V\mid v\geq x\}|=1$，因此 S_i 和 S_j 都是权重为 $[w,w_s,w_s^{<x}]$ 时 G' 的最小割。这是第二种情况，此时最小割并不唯一。

3. 若 $t_{m-1}<x\leq t_m$。令 (S_i,T_i) 为满足 $t_m\in T_i$ 的割。注意，对于所有 $j\neq i$，$w_s^{<x}(\text{cut}_{G'}(S_j))=|T_j|=|T|$，而 $w_s^{<x}(\text{cut}_{G'}(S_i))=|T_i|-1=|T|-1$。因此，我们有 $(S,T)=(S_i,T_i)$ 是权重 $[w,w_s,w_s^{<x}]$ 下 G' 的唯一最小割，且满足 $w_s^{<x}(\text{cut}_{G'}(S))=|T|-1$。

证明至此完成。 ◀

我们需要的最后一个组成部分是一个用于检查图的最小割是否唯一的算法。我们在算法 2 中对其进行描述。

算法 2 唯一性测试$^{\mathcal{A}}$

输入：边权重为 $w:\ E\to\mathbb{Z}^+$ 的图 $G=(V,E)$；针对简单加权图上的全局最小割问题的随机化算法 A 的预言机访问。输出：若存在，则返回 $G=(V,E)$ 中关于 w 的唯一割；否则返回 False。1：在 G 上运行 A，边权重为 w。2：令返回的割为 (S,T)，满足 $s\in S$。3：令 $G'=(V,E\cup E^s_{\text{star}})$。通过令所有 $e\in E^s_{\text{star}}\setminus E$ 都满足 $w(e)=0$，将 w 扩展到定义域 $E\cup E^s_{\text{star}}$。4：令 $w_1:\ E\cup E^s_{\text{star}}\to\mathbb{Z}^+$ 为如下定义的边权赋值：若 $e=(s,v)\in E^s_{\text{star}}\cap$ 割 $_{G'}(S)$，则 $w_1(e)=1$；否则 $w_1(e)=0$。5：任意选择一个顶点 $t\in T$。6：令 $G=(V,E\cup E^t_{\text{star}})$。7：令 $w_2:\ E\cup E^t_{\text{star}}\to\mathbb{Z}^+$ 为如下定义的边权赋值：若 $e=(t,v)\in E^t_{\text{star}}\cap$ 割 $_{G'}(S)$，则 $w_2(e)=1$；否则 $w_2(e)=0$。8：在 G' 上运行 A，边权重为 $[w,w_1]$。令返回的割为 (S',T')，满足 $s\in S'$。9：在 G 上运行 A，边权重为 $[w,w_2]$。令返回的割为 (S,T)，满足 $s\in S$。10：若 $S=S'=S$，则返回 (S,T)；否则返回 False。

▶**定理 21.** 设 A 是一个随机化算法，它：
1. 以简单图作为输入 $G=(V,\ E)$ 以及正整数边权$w:\ E\to\mathbb{Z}^+$，并且2. 输出一个割 (S,T)，其属于 G，满足

$$\Pr[S\text{ 是 }G\text{ 在边权重 }w]\geq\rho\text{ 下的一个最小割}$$

那么，*Uniqueness-Test*$^{\mathcal{A}}$ 是一种随机化算法，它：

1. 输入一个简单图 $G = (V, E)$ 以及正整数边权w： $E \to \mathbb{Z}^+$，并且**2.** 以概率 G 判断 $\geq 1 - 3(1-\rho)$的最小割是否唯一，**3.**仅使用 $O(1)$ 次对 $\mathcal{A}$的调用。

证明。 首先，注意到 唯一性测试$^{\mathcal{A}}$ 恰好调用 $\mathcal{A}$3 次。假设两次调用都返回一个最小割。这种情况发生的概率为 $\geq 1 - 3(1-\rho)$。

¯ ¯如果在权重 w 下， G 的最小割 (S, T) 是唯一的，那么割 (S', T') 和 (S, T) 必定都等于 (S, T)。对于 (S', T')，这一点成立，因为它是在 $[w, w_1]$ 下 $(V, E \cup E^s_{\text{star}})$ 的最小割；根据事实 8，它是在 w 下 $(V, E \cup E^s_{\text{star}})$ 的最小割；而根据命题 12，它是在 w 下 G 的最小割。对 $(\bar{S}, \bar{T})$ 的证明也相同。因此，在这种情况下，唯一性测试$^{\mathcal{A}}$正确地判定，在权重 w 下，G的最小割是唯一的。

假设在权重 w下， G 的最小割不唯一。令 (S, T) 为其中一个，$(\hat{S}, \hat{T})$ 为另一个。如果 $(S', T') \neq (S, T)$，那么 Uniqueness-Test$^{\mathcal{A}}$ 会正确判定在权重 w下， G 的最小割不唯一。因此，假设 $(S', T') = (S, T)$。我们将证明，在这种情况下，$(\bar{S}, \bar{T}) \neq (S, T)$。首先，我们有 $w($割$_{G'} S) = w($割$_{G'} \hat{S})$。因此，如果 $(S', T') = (S, T)$，则必有 $w_1($割$_{G'} S) \leq w_1($割$_{G'} \hat{S}) \Leftrightarrow |T| \leq |\hat{T} \cap T| \implies T \subsetneq \hat{T}$。因此，注意到我们有 $t \in \hat{T}$ 和 $|\hat{S}| < |S|$。这意味着 $w_2($割$_{\bar{G}} S) = |S| > |\hat{S}| = w_2($割$_{\bar{G}} \hat{S})$。因此，$(\bar{S}, \bar{T}) \neq (S, T)$。这说明 Uniqueness-Test$^{\mathcal{A}}$ 会正确判定相对于权重 w， G 的最小割不唯一。◀

接下来，我们在算法 3 中描述我们的伪确定性全局最小割算法。它同时运行一个任意的全局最小割算法$\mathcal{A}$ 以及唯一性测试$^{\mathcal{A}}$。然后我们完成定理 16 的证明。

算法 3 用于全局最小割的伪确定性算法 $\text{PD}^{\mathcal{A}}_{\text{global}}$

输入：带有边权重 w 的图 $G = (V, E)$： $E \to \mathbb{Z}^+$；可访问预言机 A，A 是一种用于简单加权图全局最小割问题的随机化算法。输出：全局最小割 (S, T) 1：任取一个顶点 $s \in V$ 作为源顶点。2：通过 $w(e) = 0$，将 w 扩展到定义域 $E \cup E^s_{\text{star}}$，对所有 $e \in E^s_{\text{star}} \setminus E$ 均如此。3：定义 $G' = (V, E \cup E^s_{\text{star}})$。4：如果 G' 在权重 $[w, w_s]$ 下有唯一的全局最小割 (S, T)，则输出 (S, T) 并终止。5： $l \leftarrow 1, u \leftarrow n$6：当 $l \leq u$ 时，执行▷ 在范围 $[l = 1, u = n]$ 内进行二分搜索。7：令 $x = \lfloor (l + u)/2 \rfloor$，并令 $(S, T) =$ A$(G', [w, w_s, w_s^{\leq x}])$。8：如果 (S, T) 是 G' 在权重 $[w, w_s, w_s^{\leq x}]$ 下的唯一全局最小割，且$w_s^{\leq x}$(割$_{G'}(S)$) $= |T| - 1$，则输出 (S, T) 并终止。9：如果 $w_s^{\leq x}$(割$_{G'}(S)$) $= |T|$，则令 $u = x - 1$ 并继续二分搜索。10：否则，令 $l = x + 1$ 并继续二分搜索。11：循环结束

定理 16 的证明。 假设每当 $\text{PD}^{\mathcal{A}}_{\text{global}}$ 将随机化算法用作子程序时，它都会返回正确答案。对于 $\geq \rho$ 的每次使用，这种情况发生的概率为$\mathcal{A}$，而对于唯一性测试，概率为 $\geq 1 - (1-\rho) \cdot 3$。由于我们使用唯一性

测试，并且$\mathcal{A}$ 至多执行 $O(\log n)$ 次；根据并合界，所有子程序均能正确运行的概率为 $\geq 1-(1-\rho)\cdot O(\log n)$。

现在，设 $(S_1,T_1)\ldots,(S_k,T_k)$ 是权重$[w,w_s]$ 下 $G'=(V,E\cup E^s_{\text{star}})$ 的最小割（满足 $\forall i\in[k]$, $s\in S_i$），并设 $\mathcal{T}=\bigcup_{i\in[k]}T_i=\{t_1,\ldots,t_m\}$。设 $i\in[k]$ 满足 $t_m\in T_i$。引理 17 表明 T_j 两两不相交，因此 i 的这种选择是唯一的。我们将证明 $\text{PD}^{\mathcal{A}}_{\text{global}}$ 返回 $(S^*,T^*)=(S_i,T_i)$。

首先，若 $k=1$，则算法 3 的第 4 节必然返回 $(S^*,T^*)=(S_i,T_i)$。因此，假设 $k>1$。在这种情况下，我们执行二分搜索。现在，只需注意，断言 20 的条件保证我们的二分搜索过程会找到 $x\in(t_{m-1},t_m]$；此时，关于权重 $[w,w_s,w_s^{\leq x}]$ 的 G' 具有唯一的最小割，即 (S_i,T_i)，而 $\text{PD}^{\mathcal{A}}_{\text{global}}$ 算法会在第 4 节输出该割。

因此，如果每个随机化子程序都正确运行，$\text{PD}^{\mathcal{A}}_{\text{global}}$ 会输出 $(S^*,\ T^*)$。正如证明第一段所讨论的，这种情况发生的概率为 $\geq 1-(1-\rho)\cdot O(\log n)$。证明至此完成。◀

5 流式处理、割查询与并行模型

本节将简要说明，算法 3 可以在顺序、流式处理、割查询和并行模型中高效实现。

5.1 顺序

设 $\mathcal{A}$ 在一个有 n 个顶点和 m 条边的图 G 上运行，耗时为 $T(n,\ m)$。显然，可以实现算法 2，使其运行时间为 $O(m+n+T(n,\ m+n))$。然后，可以实现算法 3，使其运行时间为 $O((m+n+T(n,\ m+n))\log n\log\log n))$。结合 [11], 的随机化 $O(m\log^2 n)$ 时间算法，我们得到一个用于全局最小割的伪确定性 $O(m\log^3 n\log\log n)$ 时间算法，从而证明定理 1。额外的 $\log\log n$ 因子是为了将 $\mathcal{A}$ 的成功概率从 $2/3$ 提高到 $1-1/\Omega(\log n)$。在其他模型中，我们也会付出类似的代价，但该代价被波浪号记号隐藏。

5.2 并行

算法 3 和算法 2 都只涉及以下非平凡步骤：

~1. 将星图 E^s_{star} 的边添加到图 G中。这只需通过并行地对所有 $v\in V$添加(s,v)，即可在 $O(n+m)$~工作量和 $O(1)$深度内完成。通过维护一个以 $v\in V$为索引的查找表，指示(s,v)是否 $\in E$，即可轻松避免添加重复边。 2. 将权重赋值 w' 与权重赋值 w拼接。我们的算法仅在 n 是 $\sum_{e\in E}w'(e)$的上界时才这样做。因此，我们只需将每条边的权重 w 替换为$(n+1)\cdot w(e)+w'(e)$。对每个 $e\in E$ 并行执行此操作，~ ~即可在 $O(n+m)$ 工作量和 $O(1)$深度内完成。 3. 测试两个割集 S 和 S' 是否相等。有很多简单的方法可以做到这一点~ ~，工作量为 $O(n+m)$，深度为 $O(1)$。

4. 计算割相对于边权赋值的权重 w。只需并行遍历所有边，然后以˜ ˜树形方式将它们对该割的贡献相加，即可在 $O(n+m)$ 工作量和 $O(1)$ 深度内完成。

假设 $\mathcal{A}$ 在具有 n 个顶点和 m 条边的图上，所需工作量为 $W(n,m)$，深度为 $D(n,m)$。根据上述观察，很容易看出，算法 2 和算法 3˜ ˜ ˜可通过 $O(n+m+W(n,m+n))$ 的工作量和 $O(D(n,m+n))$ 的时间实现。 O˜吸收了算法 3 中二分搜索产生的 log n 因子。结合 $O(n+m)$˜ ˜ ˜工作量和 $O(1)$ 深度的算法 [3], ，我们得到一个工作量为 $O(m)$、深度为 $O(1)$ 的伪确定性并行算法，从而证明了定理 4。

5.3 流式处理

算法 3 和算法 2 都只涉及以下非平凡步骤:

1. 向图中添加星图的边 E^s_{star} G。只需将边 $E^s_{\text{star}} \setminus E$ 以权重 0追加到输入流末尾即可完成。为此，需要跟踪输入流中的边 $E \cap E^s_{\text{star}}$。使用 $O(n \log n)$ 空间即可轻松实现，因为 $|E^s_{\text{star}}| = n$ – 只需维护 E^s_{star} 的所有边及其是否已在数据流中出现的列表即可。

2. 拼接权重赋值 w' 与权重赋值 w。只有当 n 是 $\sum_{e\in E} w'(e)$ 的上界时，我们的算法才会执行此操作。此外，对于任意边 e ，只要知道 e的端点，我们的算法就总能计算出 $w'(e)$。因此，每当我们在数据流中遇到一条边 e ，且其边 $w(e)$ 时，只需将其权重改为 $w(e)\cdot(n+1) + w'(e)$ 即可。这一步隐含地需要占用空间来存储计算 w' 所需的信息：如果 $w' = w_s$，则只需存储 s；如果 $w' = w_s^{<x}$，则需存储 x 和 s；如果 w' 是 w_1 或 w_2，则需存储割集 S 以及顶点 s 和 t 。存储割集 S 的开销最大，只需 $O(n\log n)$ 空间。

3. 测试两个割集 S 和 S' 是否相等。由于流式模型没有时间限制，一旦我们使用 S和 S' 占用 $O(n\log n)$ 空间存储好这些集合后，就可以完全不占用额外空间，轻而易举地完成比较。

4. 根据权重赋值 w' 计算割 (S,T) 的权重。在我们的算法中，只有当 $w' = w_s^{<x}$ 时才需要这样做，并且算法已经知道 x。因此，这很容易实现：我们维护 W，并将其初始化为 0。每当数据流中有一条边 e 跨越割 (S,T) 并被加入图中时，我们就修改 $W = W + w'(e)$；每当一条边 e 从图中删除时，我们就修改 $W = W - w'(e)$。这里所需的唯一空间是存储 x 和 W。这需要 $O(\log n)$ 空间。

设 $\mathcal{A}$ 是一个使用 $S(n,m)$ 空间、并对数据流进行 $P(n,m)$ 遍的流式算法。结合上述观察，算法 2 和算法 3 可以用 $O((S(n,m)+n)$ 空间实现，并且需要对数据流进行 $O(P(n,m)\cdot\log(n))$ 遍。再结合 [31], 的 $O(n)$ 空间、 $O(\log n)$ 遍算法，我们得到一个使用伪确定性 $\tilde{O}(n)$ 空间、对数据流进行 $O(\log^2 n)$ 遍的流式算法，从而证明定理 2。

5.4 割查询

对于割查询算法，这个论证总体上非常简单，因为模型唯一的限制是对图的查询，而不涉及任何计算方面。唯一非平凡的实现问题，是如何在割查询模型中拼接权重 $[w,w']$ 。形式上，给定一个割预言机 $\mathcal{O}$，其中 $\mathcal{O}(S) = w$(割$_G(S)$)，我们希望设计一个预言机 $\mathcal{O}'$ ，其中 $\mathcal{O}'(S) = [w,w']$(割$_G(S)$)。由于我们对拼接权重和图的选择，这很容易做到。

在算法 2 和算法 3 中， w'(割$_G$(S)) 对任意 $\emptyset \subsetneq S \subsetneq V$，只需给定 S 和权函数 w' 的参数即可计算。假设 $s \in S$:

1. w_s(割$_G$(S)) 等于 $n - |S|$。 2. $w_s^{<x}$(割$_G$(S)) 等于 $|\{v \in V \mid v < x\} \cap (V \setminus S)|$。 ^ ^ 3. 设 S 为定义 w_1 时所依据的割集，使得 $s \in S$。那么w_1(^S) $= |(V \setminus S) \cap (V \setminus \hat{S})|$。同样的论证也适用于 w_2 和 t。

因此，预言机 $\mathcal{O}'$ 只需返回 $\mathcal{O}'(S) = \mathcal{O}(S) \cdot (n+1) + w'$(割$_G$($S$)) 作为答案即可

设 $\mathcal{A}$ 是一个进行 $C(n, m)$ 次割查询的割查询算法。根据上述观察，很容易看出，可以实现算法 3 和算法 2，使它们需要 $O(C(n, m) \cdot \log\ n)$ 次查询。因此，结合 [31], 的算法~，我们得到一个需要 $O(n)$ 次割查询的算法，从而证明定理3。

参考文献

1 Aryan Agarwala 和 Ian Mertz。二分图匹配属于催化对数空间。载于 IEEE计算机科学基础研讨会（FOCS），2025年。2 Nima Anari 和 Vijay V. Vazirani。在 NC模型中，匹配与判定问题一样容易。载于 Thomas Vidick（编），第11届理论计算机科学创新会议（ITCS 2020），美国华盛顿州西雅图，2020年1月12日至14日，第151卷 LIPIcs，第54:1–54:25页。达格施图尔城堡 - 莱布尼茨信息学中心，2020年。doi:10.4230/LIPIcs.ITCS.2020.54。3 Daniel Anderson 和 Guy E. Blelloch。以 O(m $\log^2$ n) 工作量和较低深度实现并行最小割。ACM并行计算汇刊，10(4):18:1–18:28，2023年。doi:10.1145/3565557。4 Vladimir Braverman、Robert Krauthgamer、Aditya Krishnan 和 Shay Sapir。数据流中伪确定性计数的下界。载于 Kousha Etessami、Uriel Feige 和 Gabriele Puppis（编），第50届国际自动机、语言与程序设计学术研讨会（ICALP 2023），德国帕德博恩，2023年7月10日至14日，第261卷 LIPIcs，第30:1–30:14页。达格施图尔城堡 - 莱布尼茨信息学中心，2023年。doi:10.4230/LIPIcs.ICALP.2023.30。5 Suresh Chari、Pankaj Rohatgi 和 Aravind Srinivasan。随机性最优的唯一元素隔离及其在完美匹配和相关问题中的应用。载于第二十五届ACM计算理论研讨会论文集，第458–467页，1993年。doi:10.1145/167088.167213。6 Arkadev Chattopadhyay、Yogesh Dahiya 和 Meena Mahajan。搜索问题的查询复杂度。载于 Jérôme Leroux、Sylvain Lombardy 和 David Peleg（编），第48届国际计算机科学数学基础研讨会（MFCS 2023），第272卷《莱布尼茨国际信息学会议录》（LIPIcs），第34:1–34:15页，德国达格施图尔，2023年。达格施图尔城堡 – 莱布尼茨信息学中心。网址：https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.MFCS.2023.34，doi:10.4230/LIPIcs.MFCS.2023. 34。7 Lijie Chen、Zhenjian Lu、Igor C. Oliveira、Hanlin Ren 和 Rahul Santhanam。多项式时间伪确定性素数构造。载于第64届 IEEE 计算机科学基础年会（FOCS 2023），美国加利福尼亚州圣克鲁斯，2023年11月6日至9日，第1261–1270页。IEEE，2023年。doi:10.1109/FOCS57990.2023.00074。8 Samir Datta、Raghav Kulkarni 和 Sambuddha Roy。确定性地隔离二分平面图中的一个完美匹配。Theory of Computing Systems，47(3):737–757，2010年。doi:10.1007/s00224-009-9204-8。9 Stephen Fenner、Rohit Gurjar 和 Thomas Thierauf。二分图完美匹配属于准 NC。载于第四十八届年度 ACM计算理论研讨会论文集，第754–763页，2016年。doi:10.1145/2897518.2897564。10 Eran Gat 和 Shafi Goldwasser。具有唯一答案的概率搜索算法及其密码学应用。Electron. Colloquium Comput. Complex.，TR11-136(18:136)，2011年。网址：https://eccc.weizmann.ac.il/report/2011/136，arXiv:TR11-136。11 Pawel Gawrychowski、Shay Mozes 和 Oren Weimann。在 $O(m\log^2 n)$ 时间内求最小割。Theory Comput. Syst.，68(4):814–834，2024年。doi:10.1007/s00224-024-10179-7。12 Sumanta Ghosh 和 Rohit Gurjar。拟阵交：从搜索到加权判定的伪确定性并行归约。载于 Mary Wootters 和 Laura Sanità（编），《近似、随机化与组合优化：算法与技术》，APPROX/RANDOM 2021，2021年8月16日至18日，美国华盛顿州西雅图华盛顿大学（线上会议），第207卷 LIPIcs，第41:1–41:16页。达格施图尔城堡 - 莱布尼茨信息学中心，2021年。doi:10.4230/LIPIcs.APPROX/RANDOM.2021.41。13 Sumanta Ghosh、Rohit Gurjar 和 Roshan Raj。从加权拟阵交搜索到判定问题的确定性并行归约。Algorithmica，86(4):1057–1079，2024年。doi: 10.1007/s00453-023-01184-2。14 Oded Goldreich、Shafi Goldwasser 和 Dana Ron。论伪确定性算法的可能性与局限性。载于 Robert D. Kleinberg（编），理论计算机科学创新会议

计算机科学，ITCS ’13，美国加利福尼亚州伯克利，2013年1月9日至12日，第127–138页。ACM，2013年。doi:10.1145/2422436.2422453。15 Shafi Goldwasser 和 Ofer Grossman。伪确定性 NC 中的二分图完美匹配。载于 Ioannis Chatzigiannakis、Piotr Indyk、Fabian Kuhn 和 Anca Muscholl（编），第44届国际自动机、语言与编程研讨会，ICALP 2017，波兰华沙，2017年7月10日至14日，第80卷 LIPIcs，第87:1–87:13页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2017年。doi:10.4230/LIPIcs.ICALP.2017.87。16 Shafi Goldwasser、Ofer Grossman、Sidhanth Mohanty 和 David P. Woodruff。伪确定性流式处理。载于 Thomas Vidick（编），第11届理论计算机科学创新会议，ITCS 2020，美国华盛顿州西雅图，2020年1月12日至14日，第151卷 LIPIcs，第79:1–79:25页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2020年。doi:10.4230/LIPIcs.ITCS.2020.79。17 Shafi Goldwasser、Russell Impagliazzo、Toniann Pitassi 和 Rahul Santhanam。论 NP 搜索问题的伪确定性查询复杂度。载于第36届计算复杂性会议（CCC 2021），第200卷莱布尼茨国际信息学会议录（LIPIcs），第36:1–36:22页，2021年。doi:10.4230/LIPIcs.CCC.2021.36。18 R. E. Gomory 和 T. C. Hu。多端网络流。工业与应用数学学会期刊，9(4):551–570，1961年。arXiv:https://doi.org/10.1137/0109047，doi:10.1137/0109047。19 Vladimir Grebinski 和 Gregory Kucherov。加性模型下图的最优重构。Algorithmica，28(1):104–124，2000年。doi:10.1007/s004530010033。20 Ofer Grossman。伪确定性。博士论文，麻省理工学院，美国，2023年。URL: https://handle. net/1721.1/151209。21 Ofer Grossman、Meghal Gupta 和 Mark Sellke。伪确定性近似计数的紧空间下界。载于第64届 IEEE 计算机科学基础年会，FOCS 2023，美国加利福尼亚州圣克鲁斯，2023年11月6日至9日，第1496–1504页。IEEE，2023年。doi:10.1109/FOCS57990.2023.00091。22 Ofer Grossman 和 Yang P. Liu。对数空间中的可复现性与伪确定性。载于第30届 ACM-SIAM 离散算法年会论文集（SODA），第606–620页，2019年。doi:10.1137/1.9781611975482.38。23 Jianxiu Hao 和 James B. Orlin。求解有向图最小割的更快算法。《算法杂志》，17(3):424–446，1994年。doi:10.1006/jagm.1994.1043。24 Monika Henzinger、Jason Li、Satish Rao 和 Di Wang。加权图中的确定性近线性时间最小割。载于 David P. Woodruff（编），2024年 ACM-SIAM 离散算法研讨会论文集，SODA 2024，美国弗吉尼亚州亚历山德里亚，2024年1月7日至10日，第3089–3139页。SIAM，2024年。doi:10.1137/1.9781611977912.111。25 David R. Karger。割、流和网络设计问题中的随机采样。载于 Frank Thomson Leighton 和 Michael T. Goodrich（编），第26届 ACM 计算理论年会论文集，1994年5月23日至25日，加拿大魁北克省蒙特利尔，STOC ’94，第648–657页，美国纽约州纽约，1994年。ACM。doi:10.1145/195058.195422。26 David R. Karger。近线性时间内的最小割。《ACM 杂志》，47(1):46–76，2000年。doi: 10.1145/331605.331608。27 David R. Karger 和 Rajeev Motwani。最小割的 NC 算法。《SIAM 计算杂志》，26(1):255–272，1997年。doi:10.1137/S0097539794273083。28 David R. Karger 和 Clifford Stein。最小割问题的一种新方法。《ACM 杂志》，43(4):601–640，1996年。doi:10.1145/234533.234534。29 Jiatu Li、Edward Pyne 和 Roei Tell。区分、预测与认证：部分伪随机性概念的深远影响。载于第65届 IEEE 计算机科学基础年会，FOCS 2024，美国伊利诺伊州芝加哥，2024年10月27日至30日，第1–13页。IEEE，2024年。doi:10.1109/FOCS61266.2024.00095。30 Zhenjian Lu、Igor C. Oliveira 和 Rahul Santhanam。伪确定性算法与概率时间的结构。载于 Samir Khuller 和 Virginia Vassilevska Williams（编），

STOC '21：第53届 ACM SIGACT 计算理论研讨会，线上活动，意大利，2021年6月21日至25日，第303–316页。ACM，2021年。doi:10.1145/3406325.3451085。31 Sagnik Mukhopadhyay 和 Danupon Nanongkai。加权最小割：顺序、割查询和流式算法。载于 Konstantin Makarychev、Yury Makarychev、Madhur Tulsiani、Gautam Kamath 和 Julia Chuzhoy（编），第52届 ACM SIGACT计算理论年度研讨会论文集，STOC 2020，美国伊利诺伊州芝加哥，2020年6月22日至26日，STOC 2020，第496–509页，美国纽约州纽约，2020年。ACM。doi:10.1145/3357713.3384334。32 Ketan Mulmuley、Umesh V Vazirani 和 Vijay V Vazirani。匹配与矩阵求逆一样简单。载于第19届ACM计算理论年度研讨会论文集，第345–354页，1987年。doi:10.1145/28395.383347。33 Hiroshi Nagamochi 和 Toshihide Ibaraki。计算多重图和容量图中的边连通度。《SIAM 离散数学杂志》，5(1):54–66，1992年。doi:10.1137/0405004。34 Igor C. Oliveira。多项式时间伪确定性构造（特邀报告）。载于 Olaf Beyersdorff、Mamadou Moustapha Kanté、Orna Kupferman 和 Daniel Lokshtanov（编），第41届国际计算机科学理论问题研讨会，STACS 2024，2024年3月12日至14日，法国克莱蒙费朗，第289卷 LIPIcs，第1:1–1:1页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2024年。doi:10.4230/LIPIcs.STACS.2024.1。35 Igor C. Oliveira 和 Rahul Santhanam。亚指数时间的伪确定性构造。载于 Hamed Hatami、Pierre McKenzie 和 Valerie King（编），第49届ACM SIGACT计算理论年度研讨会论文集，STOC 2017，加拿大魁北克省蒙特利尔，2017年6月19日至23日，第665–677页。ACM，2017年。doi:10.1145/3055399.3055500。36 Klaus Reinhardt 和 Eric Allender。使非确定性变得无歧义。《SIAM计算杂志》（SICOMP），29(4):1118–1131，2000年。doi:10.1137/S0097539798339041。37 Ola Svensson 和 Jakub Tarnawski。一般图中的匹配问题属于准NC。载于 Chris Umans（编），第58届IEEE计算机科学基础年度研讨会，FOCS 2017，美国加利福尼亚州伯克利，2017年10月15日至17日，第696–707页。IEEE计算机学会，2017年。doi:10.1109/FOCS.2017.70。38 Dieter van Melkebeek 和 Gautam Prakriya。空间受限情形下的隔离去随机化。《SIAM计算杂志》（SICOMP），48(3):979–1021，2019年。doi:10.1137/17M1130538。