arXiv:2512.20939v1 [cs.DC] 24 Dec 2025

随机良结构转移系统

James
Aspnes 耶鲁大学

摘要

将良结构转移系统 [Fin87, FS01] 扩展为包含概率调度规则，我们定义了一类新的随机良结构转移系统，其中包括群体协议 [AAD+06]、化学反应网络 [Gil77], 和许多常见的闲聊模型；还包括通过预言机对这些系统进行扩充的形式，该预言机公开智能体上的全序（如比较模型中的群体协议 [GGJS23] ），或公开等价关系（如带有无序数据的群体协议 [BL23]）。

我们证明，在这些系统中，任何相位时钟的实现要么停止，要么在多项式数量的期望步数后滴答过快；并且，这些系统中的任何终止计算都会在期望多项式时间内完成或失败。后一性质使我们能够精确刻画许多通过智能体上的全序或等价关系扩充的随机良结构转移系统的计算能力：这些系统恰好计算 BPP 中的语言，而相应的未扩充系统则仅计算对称语言中的 BPL。

1 引言

我们定义了一类**随机良结构转移系统**（SWSTS），其中包括许多常见的分布式系统模型，这些模型涉及在随机调度下运行的有限状态主体。我们证明，这类系统若试图将信号生成延迟一段时间，则只能将其延迟至多与主体数量成多项式关系的时间。关键思路是扩展**良结构转移系统**（WSTS）的概念 [Fin87, FS01], ：这类系统的配置来自一个良拟序，且该序相对于转移关系构成模拟；我们在此基础上，为转移添加了关于配置大小度量的多项式概率。

这一扩展使我们能够证明，随机良结构转移系统不擅长等待：给定一个向上封闭目标集，任意配置要么能在常数步内以多项式概率到达该目标集，要么根本无法到达。这一结论对构建时钟（任何时钟协议最终都会失效，或在预期经过多项式数量的滴答后推进过快）和终止协议（任何显式终止的协议最终都会失效，或在预期的多项式时间后终止）都有影响。我们利用这些结果，对多种符合我们框架的模型的计算能力给出了精确刻画。

尽管我们关注随机调度，但本研究受到此前使用良拟序分析采用 [AAD+06]对抗性调度的群体协议的工作的启发。此前的研究已经指出，在这一背景下，群体协议是良结构转移系统的实例 [EGLM16], ；而利用 Dickson 引理 [Dic13] 证明群体协议的配置空间构成良拟序，是 Angluin 等人 [AAER07]经典证明中的关键步骤：该证明表明，群体协议稳定计算的恰好是半线性谓词。其他相关成果包括 Doty 和 Soloveichik 对群体协议中领导者选举的下界 [DS18], 、Chen等人 [CCDS17] 关于许多问题的化学反应网络存在称为**速度故障**的内在瓶颈的证明，以及 Mathur 和 Ostrovsky [MO22] 对自稳定群体协议能力的刻画。据我们所知，将良结构转移系统扩展到能够表示随机化调度的模型，是一项新的工作。

本文的组织结构如下。第2节讨论良结构转移系统，包括良拟序，以及良结构转移系统、群体协议与可达性证明之间的联系。第3节定义随机良结构转移系统。第4节展示关于等待的主要结果，包括核心性质：随机良结构转移系统最多只能等待多项式步数（定理 4.2），以及由此得出的若干关于时钟和终止计算的推论。第5节列出采用随机化调度且可产生 SWSTS 的分布式计算模型，其中包括**群体协议** [AAD+06]、封闭实例的**化学反应网络** [Gil77], 、各种同步**GOSSIP**模型 [BBK09, BCN+14, BNFK18, DKNV24], 、 [AGR22], 的同步令牌传递模型、 [GGJS23], 的**比较模型**、 [BL23], 的**无序群体协议**，以及其他模型。第6节提供了描述

SWSTS 计算所需的定义。第7节给出 SWSTS 的一致性条件，使此类计算能够通过有界错误多项式时间（BPP）或有界错误多项式时间与对数空间（BPL）进行模拟。这两种情况都依赖定理 4.2 来证明，被模拟的协议必然会在多项式数量的期望步数内结束。第8节针对第5节描述的许多模型证明了逆向结果，表明这些模型所计算的谓词恰好是 BPP 中的谓词，或 BPL 中的对称谓词，具体取决于各模型对其组成部分施加的结构限制程度。第9节讨论未来工作的可能方向。

2 良结构转移系统

分布式系统的配置通常可以通过某种嵌入关系赋予拟序，其中 $s \leq t$ 如果 s 可以被视为 t的子配置，而关系 $\leq$是拟序，仅意味着它具有自反性和传递性。[1] 我们尤其关注嵌入关系为**良拟序** [Kru72]或**良拟序**的模型；这种拟序不存在无限下降链或无限反链。[2]

如果嵌入关系是良拟序，且系统中的转移以适当方式遵循该嵌入关系，我们就得到一个**良结构转移系统** [Fin87, FS01] 或**良结构转移系统**。形式上，良结构转移系统（S, $\leq$, $\to$）由一组状态 S 以及两个二元关系 $\leq$ 和 $\to$构成，其中 $\leq$ 是 S 上的良拟序，而 $\to$ 是一个**转移关系**，它与 $\leq$ 相**兼容**，也就是说，只要 $s \to s'$ 和 $s \leq t$成立，就存在 t' 使得 $t \to t'$ 和$s' \leq t'$成立。[3]

当 $s \leq t, s \to s', s' \leq t, t \to t'$ 时，我们称转移 $s \to s'$**发生** 于 $t \to t'$。

我们主要考虑这样的转移系统：其中有有限个智能体，每个智能体的状态都来自某个有限集合 Q。因此，整个系统的状态（我们将其称为**配置** ，以区别于单个智能体的状态）是 Q^* 的一个元素，可能还带有一些额外结构。核心思想是，如果存在一个

[1]这与偏序不同：即使元素**等价** x且 y 满足$x \leq y$ 和 $y \leq x$ ，也不一定意味着 $x = y$。当我们考察那些可以置换智能体、但不会添加或移除任何智能体的嵌入时，就会看到这类例子。[2]关于良拟序的更多细节，请参见 §2.2。[3]这个定义由 Finkel 和 Schnoebelen 提出 [FS01], , 是 Finkel 原始定义的精简版本 [Fin87]。不过，由于如今它已成为标准定义，我们将在此采用它。另请参见 Schmitz 和 Schnoebelen [SS13], , 他们为该定义提供了一些补充说明，并讨论了理论的后续发展。

单射 f，将 s 中的智能体映射到 t 中的智能体，并保留配置中任何必要的结构。在这种情况下，相容性意味着：如果存在单射 $f\colon s \to t$ 和一次转移 $s \to s'$，那么就存在一个配置 t'，以及一个单射 $f'\colon s' \to t'$ 和一次转移 $t \to t'$。对于许多模型，找到转移 $t \to t'$ 的简便方法是：让像集 $f(s)$ 中的每个智能体执行其在 s 中对应的智能体所执行的操作；而让 t 中其余的智能体要么什么都不做（如果模型允许），要么执行不会影响 $f(s)$ 中智能体的操作。在这种情况下，$s \to s'$ 在 $t \to t'$ 中发生这一事实，只意味着 $f(s)$ 中的智能体观察到了一次局部看来与 $s \to s'$ 相同的转移。

2.1 示例：群体协议

良结构转移系统的一个简单示例是（有限状态）**群体协议** [AAD+06]。在群体协议模型中，我们有 n 个状态属于 Q的智能体，从而形成 Q^n中的配置。群体协议的一次**步** $s \to s'$包括在 s中选择两个智能体 i和 j，并根据**联合转移函数** $\delta : Q^2 \to Q^2$更新它们的状态，使得 $(s'_i, s'_j) = \delta(s_i, s_j)$，且 $s'_k = s_k$ 对所有 $k \notin \{i, j\}$均成立。群体协议的一个简单示例是基于抵消的领导者选举协议 [AAD+06]，它有两种状态：L（领导者）和 F（跟随者），以及一个非平凡转移 $L, L \to L, F$，用于裁减多余的领导者。请注意，群体协议中的交互并非对称：一对中的第一个智能体（**发起者**）与第二个智能体（**响应者**）有所区别，转移关系可以将这种区别考虑在内。

群体协议的嵌入 $s \leq t$是从 s 的指标到 t 的指标的保序单射映射 f，满足 $s_i = t_{f(i)}$对每个 i 都成立。根据希格曼引理 [Hig52]，该关系 $\leq$是拟序。为了证明 $\leq$ 和 $\to$ 兼容，对于任意 $s \to s'$ 和$s \leq t$，构造一个转移 $t \to t'$：取出在 s 中进行交互的智能体 i 和 j，并对 t 中的智能体 $f(i)$ 和 $f(j)$ 应用相同的交互。

2.2 识别良拟序

由于良结构转移系统与良拟序密切相关，因此进一步确立良拟序的一些其他刻画将很有帮助。

良拟序 [Kru72] 有几种等价的定义。最常用的定义是：如果一个 $\leq$ 是拟序（即一种自反且传递的关系），并且在任意无限元素序列中

x_0，x_1，x_2，.....都存在指标 $i < j$，使得 $x_i \leq x_j$。这排除了无限反链（其中所有元素彼此不可比较）和无限下降链（其中每个元素都严格小于其前驱）。后一条件体现了它与更常见的良序概念之间的联系。

一个等价的定义可能更难验证，但更容易使用：$\leq$ 是良拟序，当且仅当它是满足**有限基性质** [Hig52], 的拟序，即任何**向上闭**集都是某个有限集的**向上闭包**。[4] 有限基性质是研究良结构转移系统的关键思想。这意味着，即使状态空间是无限的，状态上闭集的可达性也可以归约为这些集合的有限基的可达性。

良拟序由来已久，也有若干经典结果表明，特定的嵌入关系会导出良拟序。我们主要会用到以下众所周知的事实：

- 设 Q 是一个有限集合，且 $\leq$ 是 Q上的一个拟序。那么 Q 是良拟序。（证明：Q 的任意无限序列都包含重复元素$x_i = x_j$，满足 $i < j$。）一个常见的特例是，当 $\leq$ 是最小拟序，且$a \leq b$ 当且仅当 $a = b$时。

- 设 $(Q, \leq)$ 是一个良序。那么 $(Q, \leq)$ 是一个良拟序。（证明：良序中的任何非空集合都有唯一的极小元素，因此任意上闭集都有一个大小为一的有限基。）

- **Dickson 引理** [Dic13]：设 Q 是一个有限集合。那么，在 $\mathbb{N}^Q$ 中向量上的分量逐项序关系是一个良拟序，其中 $x \leq y$ 当且仅当 $x_i \leq y_i$ 对所有 i都成立。

- **希格曼引理** [Hig52]：设 $(Q, \leq)$ 是一个良拟序。给定有限序列 x，$y \in Q^*$，令 x 到 y 的一个**嵌入** 为从 x 的指标到 y 的指标的严格递增函数 f，满足对所有 i，$x_i \leq y_{f(i)}$，并定义：若存在从 x 到 y 的嵌入，则 $x \leq y$。那么 $\leq$ 是一个良拟序。

[4] 这里，如果 $x \in T$ 且 $x \leq y$ 蕴含 $y \in T$，则称集合 T 为向上闭；集合的向上闭包是包含它的最小向上闭超集。

2.3 可达性

按照 [FS01], , 我们将使用记号 $s \xrightarrow{k} t$ 表示存在一个序列 $s = s_0 \to s_1 \to \dots s_k = t$, 并用 $s \xrightarrow{*} t$ 表示对于某个 k, $s \xrightarrow{k} t$ 成立。我们还会在对于某个 $0 \leq \ell \leq k$, $s \xrightarrow{\ell} t$ 成立时写作 $s \xrightarrow{\leq k} t$。等价地, 在上述每种情形下, 我们称 t 在 k 步内可达、可达, 或在至多 k 步内可达。当 T 是一个集合时, 若存在某个 $t \in T$ 使得 $s \xrightarrow{k} t$, 我们将写作 $s \xrightarrow{k} T$; 其他可达性条件也类似。

我们将使用以下记号

$$\begin{aligned}
\mathrm{Pred}(T) &= \{s \mid s \to T\} \\
\mathrm{Pred}^k(T) &= \left\{ s \;\middle|\; s \xrightarrow{k} T \right\}, \\
\mathrm{Pred}^{\leq k}(T) &= \left\{ s \;\middle|\; s \xrightarrow{\leq k} T \right\},
\end{aligned}$$

and

$$\mathrm{Pred}^*(T) = \left\{ s \;\middle|\; s \xrightarrow{*} T \right\}.$$

良结构转移系统的许多用途都源于以下标准引理:

引理 2.1. 设$(S, \leq, \to)$是一个良结构转移系统。那么, 对于任意向上闭的 $T \subseteq S$ 和任意 k, 前驱(T)、前驱$^k(T)$、前驱$^{\leq k}(t)$以及前驱$^*(T)$ 都是向上闭的。

证明。 对于前驱(T), 注意到 $s \in$前驱(T)意味着存在某个$t \in T$ 使得 $s \to t$。但此时, 若 $s' \geq s$, 则存在某个 $t' \geq t$ 满足 $s' \to t'$。由于有 $t' \geq t$, 因此 t' 属于 T (它是向上闭的); 这意味着 s' 属于前驱(T)。

通过直接归纳法, 这一结果可推广到前驱$^k(T)$; 通过观察到向上闭这一性质在并集运算下保持不变, 还可推广到前驱$^{\leq k}(T)$和前驱$^*(T)$。 □

当 T是向上闭的, 集合 $\mathsf{Pred}^{\leq 0}(T) \subseteq \mathsf{Pred}^{\leq 1}(T) \subseteq \dots$构成一个上闭集的递增序列。很容易证明(例如参见 [AvJT96, 引理 3.4]), 对于任意良拟序, 任何这样的序列最终都会稳定, 即对于某个 k, 有 $\mathsf{Pred}^{\leq k+1}(T) = \mathsf{Pred}^{\leq k}(T)$; 这意味着对于任意 T, 都存在某个有限的 k, 使得 $\mathsf{Pred}^*(T) = \mathsf{Pred}^{\leq k}(T)$。

这是一项非常强的性质，构成了良结构转移系统在验证中具有诸多用途的基础（示例请参见 [AvJT96]）。我们可以将其改述为：

引理 2.2。 设 $(S, \lesssim, \rightarrow)$ 是一个良结构转移系统，且$T \subseteq S$ 是向上闭的。那么存在一个固定的 k，使得对于任意 $s \in S$，要么 s 能在至多 k 步内到达 T，要么 s 完全无法到达 T。

3 随机良结构转移系统

标准的良结构转移系统概念并未纳入显式的调度机制：$\rightarrow$关系定义了哪些配置可以 彼此转移，却没有说明这些配置将 彼此转移的可能性有多大。当人们主要关注可达性时，这样做是可以接受的，但我们希望扩展标准定义，以涵盖具有显式转移概率的模型。我们所考虑的许多模型都具有概率调度规则，该规则使每一步的转移选择都按照某种分布随机进行。

定义 3.1。 一个随机良结构转移系统***(SW- STS)*** $(S, \leq, \rightarrow, |\cdot|, \Pr)$ 由以下部分组成：

- 一个良结构转移系统 $(S, \lesssim, \rightarrow)$。
- 一个函数 $|\cdot|$，为每个配置 $s \in S$ 赋予一个权重 $|s|$。
- 一个函数 $\Pr$，为每个转移 $s \rightarrow s\prime$指定转移概率 $\Pr[s \rightarrow s\prime]$，使系统的一次执行 $X_0 \rightarrow X_1 \rightarrow X_2 \rightarrow \ldots$成为马尔可夫链的一条轨迹，其转移概率为 $\Pr[X_{i+1} = s\prime \mid X_i = s] = \Pr[s \rightarrow s\prime]$。

满足以下要求：

- *For each transition $s \rightarrow s'$, there is a constant k, such that in any execution of the system,* $\Pr[X_{i+1} \geq s' \mid X_i \geq s] = \Omega\left(|X_i|^{-k}\right)$.

我们将最后一个条件称为具有**多项式转移概率**。注意，不必存在一个将 $s \rightarrow s\prime$扩展为 $t \rightarrow t\prime$的单一转移；只需从任意 $t \geq s$ 出发，与 $t' \geq s'$ 对应的所有可能转移 $t \rightarrow t'$ 的总概率为多项式即可。

群体协议是 SWSTS 的一个例子。在随机化调度下，配置 X_i 的权重 $|X_i|$ 是 n中智能体的数量

X_i。由于每一种可能的成对交互发生的概率都相同$n\ ^{1}_{n-1()}$，因此，在一次转移中发生交互的任意一对智能体 $s \to s'$ 也会在任何一个 X_i 中发生交互，且概率相同 $s \leq X_i\, n\ ^{1}_{(n-1)} = \Omega\big(|X_i|^{-2}\big)$ 在这种情况下,一指数 k 是固定的，但一般而言， k 的选择可能取决于转移 $s \to s'$的选择。

我们通常会要求权重函数在转移下保持不变：如果 $s \to s'$，那么 $|s| = |s'|$。具有此性质的权重函数的 SWSTS 称为**封闭**。封闭 SWSTS 的例子包括大多数分布式系统，在这些系统中，转移只改变智能体的状态，而不改变其身份或数量；还包括化学反应网络 [Gil77] ，其中权重定义为总原子数，且任何转移都会保持该量不变。非封闭 SWSTS 的例子可能包括允许新智能体随时间到达的系统，或通过吸收未计入权重的原料来增加体积的化学反应网络。

4 多项式期望步数下的可达性

随机调度使我们能够计算协议到达目标配置所需的期望时间。本节将证明：当完成由到达某个上闭集 T 来刻画，而失败由到达一个永远无法到达 T 的配置来刻画时，封闭 SWSTS 的执行平均会在多项式数量的步骤内完成或失败。

我们先给出一个限制性更强的结果，即任意可达的向上闭集合 T 都能以多项式概率在固定步数内到达：

引理 4.1。设 X 是一个封闭 *SWSTS* $(S, \leq, \to, |\cdot|, \Pr)$ 的执行。设 $T \subseteq S$ 为向上闭集合。设 $X_i \in$ 前驱 $^*(T)$。则存在固定常数 ℓ 和 k，使得 $\Pr[\exists j \leq \ell : \ X_{i+j} \in T] \geq \Omega\Big(|X_t|^{-k}\Big)$。

证明。根据引理 2.1，Pred*(*T*) 是向上闭的；根据引理 2.2，对于某个固定的 ℓ，Pred*(*T*) =Pred$^{\leq \ell}$(*T*)。为 Pred$^{\leq \ell}$(*T*) 选取一个有限基。

任意 t 属于前驱*(*T*) =前驱$^{\leq \ell}$(*T*) 的元素都高于前驱$^{\leq \ell}$(*T*) 的某个基元素 s。因此，存在一个 $\ell_s \leq \ell$ 和一系列转移 $s = s_0 \to s_1 \to \ldots \to s_{\ell_s} \in T$，使 s 属于前驱$^{\leq \ell}$(*T*)。这映射到一族部分执行 $X_i \to X_{i+1} \to \ldots \to X_{i+\ell_s} \in T$，其中每个 $X_{i+j} \geq s_j$。应用多项式转移概率后，存在固定常数 k_j

使得

$$\begin{aligned}\Pr\left[X_{i+\ell_s} \geq s_{\ell_s} \mid X_i \geq s\right] &\geq \prod_{j=1}^{\ell_s} \Pr\left[X_{i+j} \geq s_j \mid X_{i+j-1} \geq s_{j-1}\right] \\ &\geq \prod_{j=1}^{\ell_s} \Omega\Big(|X_i|^{-k_j}\Big) \\ &= \Omega\Big(|X_i|^{-k_s}\Big),\end{aligned}$$

其中

$$k_s = \sum_{j=1}^{\ell_s} k_j.$$

令 $k = \max_s k_s$，其中 s 遍历 $\mathrm{Pred}^{\leq \ell}(T)$ 的有限基。于是

$$\begin{aligned}\Pr\left[\exists j \leq \ell \colon X_{i+j} \in T \mid X_i \in \text{前驱}^*(T)\right] &\geq \text{最小值}_s \Pr\left[X_{i+\ell_s} \in T \mid X_i \geq s\right] \\ &\geq \Omega\Big(|X_i|^{-k}\Big).\end{aligned}$$

□

该引理直接推出以下结论，这是我们关于封闭 SWSTS 的核心结果：

定理 4.2. 设 X 是封闭 *SWSTS* $(S, \lesssim, \to, |\cdot|, \text{概率})$ 的一次执行。设 $T \subseteq S$ 向上闭。设 τ 是 $X_\tau \in T$ 或 $X_\tau \notin \text{前驱}^*(T)$ 首次成立的时刻。则存在常数 k，使得 期望 $[\tau] = O\Big(|X_0|^k\Big)$。

证明。 由定理 4.2，存在常数 ℓ和 k ，使得若 $X_i \in \text{前驱}^*(T)$，则以 $\Omega\Big(|X_i|^{-k}\Big) = \Omega\Big(|X_0|^{-k}\Big)$ 的概率，存在某个 $j \leq \ell$ 使 $X_{i+j} \in T$ 。因此每经过 ℓ 步， X_i 要么以 $\Omega\Big(|X_0|^{-k}\Big)$ 的概率进入 T，要么永远离开前驱$^*(T)$。无论哪种情况，等待该事件发生的期望时间都不超过 $O\Big(\ell |X_0|^k\Big) = O\Big(|X_0|^k\Big)$。 □

4.1 相位时钟的失败

在总体协议的语境下，**相位时钟** [AAE08] 是一种机制，使协议能够以高概率等待若干步，通常用于让其他随机过程（如流行病过程）完成。文献中的相位时钟 [AAE08、KU18、AAG18,

GS18、GS20、GSS21] 反复计算期望为 $\Theta(n \log n)$ 或其他步数的时间间隔，其中每个时间间隔的开始都由一个包含领导者代理的配置标记（有时包含多个领导者代理）：这些领导者代理进入我们称为**滴答**的一类状态，随后领导者恢复到另一类我们称为**嘀嗒**的非滴答状态。对于具有这种结构的相位时钟，引理 4.1 保证它会在多项式数量的滴答内因滴答过快而失效。

论证很简单：包含至少一个滴答状态的配置类 T 和包含至少一个嘀嗒状态的配置类 T' 都是向上闭合的。因此，引理 4.1 表明，从 T' 中的任意配置出发，都有多项式概率在常数步内到达 T 中的某个配置；再次应用该引理，则可得到类似的多项式概率，在常数步内返回 T 中的某个配置。在时钟反复滴答的过程中，避免这一事件的唯一方式就是发生故障：到达某个配置，使得此后 T 和 T' 均不再可达。

我们可以将这一结论正式表述如下：

定理 4.3。 在 *SWSTS* 中定义一个相位时钟协议，使该协议在向上闭集 T 与向上闭集 T' 中的配置之间反复交替；并将相位时钟的一轮定义为一系列转移 $T \xrightarrow{*} T' \xrightarrow{*} T$。那么，对于任意封闭 *SWSTS* 中的相位时钟，其在 $O(1)$ 步内完成一轮或到达某个再也无法抵达 T 的配置之前的期望时间，与其初始配置的大小呈多项式关系。

证明。 从 X_0 开始，运行相位时钟，直到它到达 T 或某个无法到达 T 的配置；根据定理 4.2，这需要的时间关于 $|X_0|$ 为多项式时间。

从 T 中的任意配置出发，应用引理 4.1，以多项式概率在常数步数内到达 T'，随后再次到达 T。如果未能实现，则两次应用定理 4.2，以多项式时间完成一轮相位时钟运行，或到达一个无法再到达 T 的状态。

如果我们再次到达 T，这就提供了另一次以多项式概率获得常数时间轮次的机会。因此，经过期望为多项式的若干次迭代（每次迭代的期望时间也为多项式），我们要么获得一个常数时间轮次，要么到达一个无法到达 T 的配置。□

4.2 SWSTS 计算的多项式时间终止

定理 4.2 的另一个应用，是限制封闭 SWSTS 协议的期望运行时间。在这类协议中，终止由一个特殊的领导者代理进入有限个终止状态 L_{decision}之一来表示。由于任何包含此类状态的配置都是终止配置，终止配置的集合 T 是向上闭的。这意味着，要么到达某个终止配置，要么所有终止配置都将在多项式期望时间内变得不可达。这对计算能力构成了内在限制，例如，基于领导者的群体协议 [AAE08]，无论设计得多么巧妙，其运行步数的期望都不可能超过多项式。

我们将在 §5 中给出许多 SWSTS 的例子。在 §6 中建立一些用于描述这些模型中计算过程的工具之后，我们将应用定理 4.2 证明，在大多数这些模型中以有界双侧错误计算的任何语言，都属于 BPP 类；该类包含由概率多项式时间图灵机以有界双侧错误计算的语言。即使不对 SWSTS 计算施加多项式时间限制，这一结论仍然成立；定理 4.2 会为我们提供多项式时间限制。不过，要做到这一点，计算过程必须显式终止，即进入某个向上闭集所包含的决策配置。

4.3 显式终止与隐式收敛

并非所有模型都假设显式终止。**稳定计算** 由 Angluin等人 [AAD$^+$06]定义，要求最终永久收敛到这样的配置：其中所有智能体都显示正确的输出。这并不要求智能体知道自己显示的是正确输出；事实上，我们可以证明，多项式时间保证并不

f

定理 4.2 不适用于收敛，因为可以构造出一些简单的群体协议，它们只有经过指数级时间后才会收敛。

例如，考虑一个流行病过程，其中智能体的状态为**易感** (S)或**感染** (I)，并加入一个**医生** 代理（D），它会治愈遇到的任何感染智能体。形式上，这是一个具有非平凡转移 $IS \to II$ 和 $ID \to SD$的群体协议。假设我们从一个医生开始，该协议唯一可达的稳定配置是：医生处于状态 D，其余 $n-1$ 个智能体都处于状态S。

然而，当 k 个智能体中有 n 个处于状态 I 时，下一步有新

代理被感染的概率为 $\frac{k}{n}\frac{(n-k-1)}{(n-1)}$，而一个代理停止感染的概率仅为 $\frac{k}{n}\frac{}{(n-1)}$。这使得该过程成为一种有偏随机游走，对于大多数 k 的取值，其偏向显著远离 0。因此，到达 0 需要一段时间：

定理 4.4。从一个包含 $n-1$ 个处于状态 I 的智能体和一个处于状态 D 的智能体的配置出发，$IS \to II$、$ID \to SD$协议收敛到其稳定配置的期望时间关于 n 呈指数级小。

证明。令 X_t 表示经过 t 步后受感染智能体的数量。在给定 $X_{t+1} \neq X_t$ 的条件下，我们有

$$\Pr\left[X_{t+1} = X_t + 1 \mid X_{t+1} \neq X_t\right] = \frac{n-k-1}{n-k}$$

and

$$\Pr\left[X_{t+1} = X_t - 1 \mid X_{t+1} \neq X_t\right] = \frac{1}{n-k}.$$

我们将通过考虑一个更简单的耦合过程 $Y_t \leq X_t$ 来界定 X_t 的收敛时间。该耦合过程表现为一个具有固定偏差的间歇性随机游走，在 $b = \lfloor \frac{n-1}{2} \rfloor$ 处设有反射屏障，在 0 处设有吸收屏障，初始状态为 $Y_0 = b \leq X_0$。

仅当 $X_{t+1} \neq X_t$ 时，Y_t 与 X_t 之间的耦合才会具有 $Y_{t+1} \neq Y_t$。当 Y 确实移动时，我们令 $\Pr\left[Y_{t+1} = Y_t + 1 \mid Y_{t+1} \notin \{0,\ Y_t,\ b\}\right] = p = 1 - \frac{2}{n-1}$。由于存在反射屏障，$Y$ 只有在 $Y_t < b$ 时才能上升。如果 $X_t \geq b+1$，独立选择 Y_{t+1} 和 X_{t+1} 会保持 $Y_{t+1} \leq X_{t+1}$。如果 $X_t \leq b$，那么 X_{t+1} 下降的概率至多为 $\frac{1}{n-b} \leq \frac{2}{n-1} = 1-p$。因此在这种情况下，我们可以安排使 Y_{t+1} 在 X_{t+1} 下降时也随之下降，从而再次保持 $Y_{t+1} \leq X_{t+1}$。

现在我们来考虑 Y_t 到达 0 所需的期望时间。每当 $Y_t = b$ 时，它总是只需一步就能到达 $Y_{t+1} = b-1$。从这一步开始，我们面对的是一个标准的破产问题，其中 Y 下次先到达 b、而非先到达 0 的概率恰好为

$$\frac{(q/p)^{b-1} - (q/p)^b}{1 - (q/p)^b}$$

（例如参见 [Fel68, XIV.2.4]）。在这种情况下，

$$(q/p) = \frac{2/(n-1)}{1 - 2/(n-1)} = \Theta(1/n),$$

这给出了首先到达 0 的概率

$$\frac{\Theta(n^{-b-1}) - \Theta(n^{-b})}{1 - \Theta(n^{-b}} = \Theta(n^{-b}) = \Theta\Big(n^{-\Theta(n)}\Big).$$

由于该事件发生的概率极低，期望需要进行指数次尝试才能从 $b-1$ 到达 0。每次尝试至少需要两步，因此期望步数也是指数级的。

□

这表明，收敛所需的时间可能远长于显式终止，这会给检测收敛带来不利影响。如果我们尝试扩展上述协议，使某个代理拥有一个显式终止状态，那么，根据定理 4.2，到达标记终止配置所需的时间是多项式级的；而根据定理 4.4，实际收敛却需要指数级的时间。两者之间的差距意味着，在除概率呈指数级低的执行之外，收敛检测器都会过早触发。

5 SWSTS 示例

给定一个采用随机化调度的分布式系统模型，我们可以预期会获得 S 和一条随机转移规则 $\rightarrow$。要将其转化为 SWSTS，需要找到一个与 $\rightarrow$兼容的良拟序 $\leq$，以及一个作用于 S 的权重函数 $|\cdot|$，使其给出多项式级的转移概率。

假设任意配置中至少存在一个转移，则可以利用一个**完全**拟序来轻松实现这一点，其中 $s \leq t$ 对所有 s 和 t均成立。这是一个良拟序，因为任意无限序列 x_0，x_1，.....都具有 $x_0 \leq x1$。对于完全良拟序，与 $\rightarrow$ 的相容性之所以成立，是因为对于任意 $s \rightarrow s'$ 和$s \leq t$，以及任意满足 $t \rightarrow t'$、$s' \leq t'$的 t'，均有相应结论。多项式转移概率是显然的，因为对于任意 $s \rightarrow s'$，$\Pr[X_{i+1} \geq s' \mid X_i \geq s] = 1$，这是由于所有 $X_{i+1} \geq s'$。我们永远不会使用这个特定的良拟序，因为它使定理 4.2变得微不足道：每个向上封闭目标集 T 要么是空集，要么是所有配置的集合。但这确实表明，我们可能需要谨慎处理，以确保子配置关系 $\leq$除了满足 SWSTS 的技术要求之外，还能反映所研究系统的有趣性质。

在本节中，我们将说明，文献中许多已知模型都可以描述为具有相当自然的子配置关系的 SWSTS。我们还将举例介绍一些此前未曾研究过的较为特别的 SWSTS。

5.1 群体协议

到目前为止，我们一直以群体协议 [AAD+06]作为标准示例，下面简要回顾一下，以说明这种方法：

- 含有 n个 s智能体的配置，是 Q^n中的智能体状态向量。其权重函数 $|s|$ 就是 n。转移是封闭的，因为它们不会改变智能体的数量。

- 子配置关系 $s \leq t$ 是一个嵌入 $f : [|s|] \to [|t|]$，将 s 的指标映射到 t的指标，其中 f 严格递增，并且对于所有 i，都有$s_i = t_{f(i)}$。根据希格曼引理，使用 Q 上的平凡良拟序（其中 $q \leq q'$ 当且仅当 $q = q'$），这构成一个良拟序。

- $\leq$ 与 $\to$ 之间的相容性成立，因为每当 $s \to s'$将 s_i 和s_j 的状态更新为新值（s'_i，s'_j）$= \delta$（s_i，s_j）时，对于任意 $t \geq s$，都存在一个匹配转移 $t \to t'$，它对$t_{f(i)} = s_i$ 和$t_{f(j)} = s_j$执行相同的更新，并使用相同的嵌入得到 $t' \geq s'$。

- 多项式转移概率源于每个可能的转移$t \to t'$ 以概率 $\frac{1}{n(n-1)} = \Omega(|t|^{-2})$发生，这为任意 $s \to s'$ 上的 $\Pr[X_{i+1} \geq s' \mid X_i \geq s]$ 提供了一个下界，因为至少存在一个可能的转移 $X_i \to X_{i+1} \geq s'$，且它以 $\Omega(|X_i|^{-2})$的概率发生

由此可知，采用随机化调度的群体协议构成了一类封闭 SWSTS。我们将这类系统称为 POP。

5.2 带有边预言机的群体协议

最近的两个模型 [BL23, GGJS23] 为总体增添了额外的结构信息，不难证明，这些增强模型也构成封闭 SWSTS。我们将先介绍这两个具体模型，然后介绍一种利用携带这些额外信息的**边预言机**来涵盖这两种模型的方法。

5.2.1 无序模型与有序模型

Blondin and Laceur [BL23] 定义了一种具有受限无限状态的群体协议变体，称为**无序数据群体协议**。在该模型中，每个智能体的状态由有限状态空间中的一个可变状态 Q 组成，该状态可由转移关系直接访问，

以及一个不可变的**数据**或**颜色**，它来自一个固定且可能无限的域$\mathbb{D}$，只能对其进行相等性测试。此时，转移关系的形式为 $\delta \in Q^2 \times \{=, \neq\} \to Q^2$，其中第三个输入用于报告两个智能体的颜色是否相同。我们将此模型简称为 POP$^{=}$。

Gańczorz 等人 [GGJS23] 定义了一种类似的**比较模型**，其中每个智能体都有唯一的隐藏密钥；但现在，密钥不再通过相等性测试，而是按照全序进行比较。该模型中的转移关系现在具有如下形式：$\delta \in Q^2 \times \{<, >\} \to Q^2$。

由于比较模型对所有智能体进行了排序，因此协议的输入可以视为一个字，而不只是状态的多重集。这意味着我们可以讨论此类协议所接受的语言。

为了与无序群体协议保持对称，我们将比较模型中的协议称为**有序群体协议**，并将该模型简称为 POP$^{<}$。

5.2.2 通过边预言机进行表示

将这些模型推广的一种方式是：智能体x和 y之间的任何交互，都由如下形式的转移规则控制：$\delta(q_x, q_y, A(x, y)) = \left\langle q'_x, q'_y \right\rangle$ 其中 A 是一个**边预言机**，它为有序对$\langle x, y\rangle$ 提供信息，且该信息不依赖于智能体的状态 q_x 和 q_y。这种边预言机模型本质上是 Michail等人 [MCS11], 提出的**中介群体协议**的一种静态版本，区别在于后者允许协议更新每条边上的标签。我们将由特定边预言机增强的 POP A 记作 POP^{A}；之后也会用相同的上标记法表示由相应边预言机增强的其他模型。

POP$^{=}$ 和 POP$^{<}$ 都可以用边预言机来表示，因为预言机可以根据具体情况，对智能体的隐藏数据进行相等性测试，或提供智能体的排序信息。

5.2.3 定向

我们采用了 POP 的约定：交互的发起者向边预言机提供第一个参数。

这种选择在某种程度上是任意的。我们考虑的大多数预言机都是**可逆的**，也就是说，存在一个函数 f，可以计算 $A(y, x) = f(A(x, y))$。这样一来，只要在不同交互中始终采用一致的选择，交互中的任一智能体都可以作为预言机的第一个参数，而不会改变模型的能力。不过，

为稳妥起见，我们会尽量对边预言机的参数采用一种在其他模型中也适用的一致排序方式：对于双向通信模型，我们假定其中一个智能体总是被标记为发起者，并作为边预言机的第一个参数；对于单向通信模型，我们则将传递信息的智能体作为边预言机的第一个参数。

5.2.4 后继预言机

遗憾的是，对于任意边预言机，群体协议并不一定能给出有用的 SW-STS。例如，假设全序群体的后继预言机仅当响应者代理是发起者的直接后继时才返回 true。这足以实现一种不会出错的图灵机模拟，其中每个代理持有一个磁带单元，因为移动磁带头只需等待预言机识别出磁带头当前位置的直接前驱或后继。此类实现可以通过计数等待指数级时间，从而违反定理 4.2。这并不一定意味着不存在与 $\to$ 兼容的良拟序 $\leq$，因为如前所述，完备良拟序与任何转移关系都兼容；但这确实意味着，任何与 $\to$ 兼容的良拟序 $\leq$ 都存在一个包含终止状态的配置 s 和一个不包含终止状态的配置 $t \geq s$，也就是说，转移到更大的配置时可能失去终止性。

类似结果也适用于任何能够在不出错的情况下用足够多的磁带单元模拟图灵机的模型。这包括运行在有界度交互图上的群体协议 [AAC+05]、社区协议 [GR09], 、中介群体协议 [MCS11], 以及空间群体协议 [GKL+25], 因为在某些启动配置下，这些模型都能通过利用或构造代理之间某种形式的邻接关系来模拟线性图灵机磁带。

5.2.5 全拟序预言机

然而，对于 $\mathrm{POP}^{=}$ 和 $\mathrm{POP}^{<}$，存在能够导出良拟序的嵌入。我们可以通过定义一种同时将这两种模型作为特例的边预言机模型来证明这一点。在这种模型中，我们将其称为<加载>全拟序群体协议</加载>，每个智能体都有一个可变状态，以及一个取自全序集 $\mathbb{D}$ 的隐藏颜色；边预言机则报告智能体颜色的相对顺序，从而得到形式为 $\delta \in Q^2 \times \{<,\ =,\ >\} \to Q^2$ 的转移关系。一个包含 n 个智能体的配置是

$(Q \times \mathbb{D})^n$ 中的一个向量；由于边预言机仅取决于颜色的顺序，而不取决于智能体位置，为方便起见，我们可以要求任一配置中的智能体按颜色递增顺序排列。由于两个智能体可能具有相同的颜色，$\mathbb{D}$ 上的全序会在智能体上诱导出一个全拟序，因此得名。我们将由颜色全序诱导的智能体拟序记作 $\lesssim$，并将该模型称为 POP$^{\lesssim}$。

POP$^{=}$ 和 POP$^{<}$ 模型都是 POP$^{\lesssim}$的特例。对于 POP$^{=}$，假设转移关系不区分 $<$和 $>$，而是将二者都视为 $\neq$。对于 POP$^{<}$，假设每个代理的初始颜色都不相同。

现在我们论证，对于 POP$^{\lesssim}$中的任意协议，都存在一个配置上的良拟序 $\leq$，它与该协议的转移关系 $\rightarrow$ 相容。这将表明 POP$^{\lesssim}$、 POP$^{=}$ 和 POP$^{<}$ 都构成良结构转移系统。我们从以下引理开始:

引理 5.1。 设一个集合 Q 上的拟序列是由 Q 中的元素构成的序列s，并配有一个定义在 s 指标上的拟序 $\lesssim$，满足$i < j$ 蕴含 $i \lesssim j$。给定良拟序集合 Q上的两个拟序列 s 和 t，当存在从 s 到 t 的严格递增嵌入 f，使得 $i \lesssim j$ 当且仅当 $f(i) \lesssim f(j)$，且对所有 i、 $s_i \leq t_{f(i)}$均成立时，令 $s \leq t$。那么 $\leq$ 是良拟序。

证明。 令 $i \approx j$ 表示当 $i \lesssim j$ 和$j \lesssim i$时成立的等价关系，并令 $[i]$ 表示 i 在该关系下的等价类。注意，在任意拟序序列中，每个等价类的元素都是连续的。因此，我们可以将 s 重新编号为 s_{ci}，其中 c 是一个等价类，而 i 是该等价类中某个元素的位置。

令 $\hat{}\ f(c)$ 表示 f 将 c的每个元素映射到的等价类。对于每个 c， $f|_c$都是从 s_c到 $t_{\hat{}\ f(c)}$的递增嵌入，并且对于所有 i，都有 $s_{ci} \leq t_{f(c)i}$。根据希格曼引理，这类嵌入构成一个良拟序 $\hat{}$，并且相对于该拟序有 $s_c \leq t_{f(c)}$。但这样一来， f 就是从 s中的等价类到 t中的等价类的递增嵌入，对于所有 c，都有 $s_c \leq t_{\hat{f}(c)}$；这再次满足希格曼引理的条件。

□

对于 POP$^{\lesssim}$，这使我们能够证明:

引理 5.2。 POP$^{\lesssim}$ 中的每个协议都会产生一个良结构转移系统。

证明。 固定 POP$^{\lesssim}$中的某个协议，并设 s 和 t是该协议的配置。若存在满足引理 5.1条件的嵌入 f，则记作 $s \leq t$。这在配置上给出了一个良拟序。

$\to$ 与 $\leq$ 的相容性基本可由与 POP 相同的论证得出：给定一个转移 $s \to s'$，其中智能体 i 和 j 进行交互，并给定一个嵌入 $f\colon s \to t$，那么就存在一个转移 $t \to t'$，其中智能体 $f(i)$ 和 $f(j)$ 进行交互，并观察到与 s中相同的状态和相对颜色顺序，从而使它们能够选择相同的新状态 $t'_{f(i)} = s'_i$ 和 $t'_{f(j)}$。由此可得，对所有 k，均有 $t'_{f(k)} = s'_k$，因此 f 也构成一个嵌入 $s' \leq t'$。□

这表明，$\mathrm{POP}^{\lesssim}$（因此 $\mathrm{POP}^{=}$ 和 $\mathrm{POP}^{<}$也一样）是良结构转移系统的一类。假设 δ 是确定性的，并采用与 POP 相同的均匀随机调度规则，则可得到多项式转移概率，因此这些系统也都是 SWSTS。

5.2.6 祖先预言机

克鲁斯卡尔树定理 [Kru60] 指出，带有来自良拟序的标签的有限树在同胚嵌入下构成良拟序，其中 $s \leq t$ 如果存在一个映射 f，将 s 的顶点映射到 t 的顶点，并且满足：（a）对标签保序，即对于所有 i，s_i 的标签与 i 的标签 $t_{f(i)}$ 和$f(i)$ 满足 $s_i \leq t_{f(i)}$；以及（b）对边保持结构，即 s 中的每条边都映射为 t中的一条不相交路径。

Nash-Williams [NW63]为有根树的同胚提供了一种更便捷的刻画。在 Nash-Williams 的定义中，如果 s和 t是有根树，则 $f\colon s \to t$ 是从 s 到 t 的一个同胚，如果对于 s中的每个节点 i，i 的子节点分别映射到 f的不同子节点的后代(i)。这一性质具有保持祖先关系的优点：$f(i)$是 j 的祖先，当且仅当 i 是 j的祖先。还很容易看出，它等价于克鲁斯卡尔更一般定义中的（b）部分，因为经过不同子节点的下降链映射，会为 s中从某个父节点出发的边在 t 中形成不相交路径。

记作 $i \sqsubseteq j$ if i 是 j的祖先。每个代理都拥有一个唯一的隐藏字符串，因此可以在本地表示这一信息：当i的字符串是 j字符串的前缀时，$i \sqsubseteq j$。我们可以定义一个模型 $\mathrm{POP}^{\sqsubseteq}$，其中配置是一棵以某个有限集合 Q 中的状态为标签的有根树，转移函数的形式为 $\delta\colon Q^2 \times \{\sqsubseteq, \perp, \sqsupseteq\} \to Q^2$，其中第三个参数表示发起者是响应者的祖先、与响应者无关，还是响应者的后代。

$\mathrm{POP}^{\sqsubseteq}$ 中的配置现在是一棵带标签的有根树，而嵌入$s \leq t$ 是有根树之间的保标签同胚。根据克鲁斯卡尔树定理（与通常情况一样，在 Q上采用平凡的良拟序），$\mathrm{POP}^{\sqsubseteq}$ 中的配置构成良拟序。与 $\to$ 的相容性源于 $\sqsubseteq$ 在嵌入下保持不变

。多项式转移概率来自所有转移都具有相同的$n^{-1}{}_{(n-1)}$概率，与其他所有 POP 变体中的情况一样。因此，POP$^{\sqsubseteq}$ 构成一类封闭 SWSTS。

5.3 化学反应网络

群体协议模型是**化学反应网络**（CRN）模型的一种特例 [Gil77]。除了形式为 $A + B \to C + D$ 的成对交互外，化学反应网络还可以包含这样的反应：左侧有任意正数的反应物，右侧有相应的生成物。直观来说，在这种模型中，代理对应分子，而步骤对应化学反应。化学反应网络通过列出允许的反应来定义，例如反应 $H_2+H_2+O_2 \to H_2O+H_2O$，它将两个 H_2 分子和一个 O_2 分子转化为两个 H_2O分子。反应中可能出现的分子类型称为**物种**，对应于其他模型中的智能体状态。每种可能发生的反应都有一个**速率**，其值等于某个常数乘以可提供相应反应物的方式数。某一反应在给定步骤中发生的概率，等于其速率与所有可能反应的速率总和之比。

由于化学反应网络不会保持智能体的数量，因此我们无法使用群体协议所采用的 Q^n 表示法。相反，我们将化学反应网络的配置建模为 $\mathbb{N}^Q$ 中的有限维向量，其中记录了每种物种的分子数量。子配置关系是逐分量序：当且仅当对所有 $q \in Q$都有 $s_q \leq t_q$ 时，$s \leq t$ 成立。一个反应由一对向量 $\langle x,\ y\rangle$指定，其中当 $x \leq s$时，$s \to s - x + y$。

根据 Dickson 引理，逐分量序是一种良拟序。只需观察到 $s \to s' = s - x + y$且 $s \leq t$ 蕴含 $x \leq s \leq t$，便可轻易证明它与 $\to$ 相容，由此得到一个满足$s' \leq t'$的转移 $t \to t' = t - x - y$。

配置的权重通过为每种物种 $q \in Q$指定一个总原子数 w_q，并计算 $|s| = \sum_q w_q s_q$得出。对于包含所有反应物和生成物的实际化学反应，这一数值会保持不变；但对于任意化学反应网络则未必如此，因为这类网络可能允许生成新的生成物，例如消耗未在反应物中明确列出的原料来生成生成物。

还需证明该模型给出的转移概率是多项式的。任意转移 $s \to s'$ 都对应某个反应 $\langle x, y$。该反应在 $t \geq s$ 中的速率 r_{xy} 由 $c_{xy} \prod_{q \in Q} (t_q)^{x_q}$ 给出，其中 c_{xy} 是

该反应的基准速率，而 $(t_q)_{x_q} = t_q(t_q - 1) \ldots (t_q - x_q + 1)$ 统计了在配置 t中供应 x_q 个分子 q 的方式数。假设没有物种的原子序数小于 1，则该乘积中的每一项都至少为 1，至多为 $|t|^{x_q}$。由此可知，r_{xy} 是关于 $|t|$的多项式，而该反应成为下一次发生的反应的概率 $\sum_x r_{xy} \sum_{y'} r_{x'y'}$ 也关于 $|t|$呈多项式。

因此，我们得到另一类 SWSTS，我们将称之为化学反应网络。

5.3.1 非封闭化学反应网络

如前所述，此类中的 SWSTS 可能是封闭的，也可能不是。封闭化学反应网络满足定理 4.2 的条件，因而与其他封闭 SWSTS 一样，具有在多项式期望时间内完成或失败的相同性质。但非封闭化学反应网络可用于证明，这些结果对于非封闭的 SWSTS 并不必然成立。

下面的例子基于 Soloveichik 等人 [SCWB08] 的一个构造。考虑具有物种 a、b和 c以及转移的化学反应网络

$$\begin{aligned} a + a &\to c, \\ a + b &\to a + b + b, \\ b + b &\to b + b + b, \end{aligned}$$

它们都以相同的恒定速率发生。令 T 为至少包含一个 c的配置上闭集。对于每个 $i \in \mathbb{N}$，令 s_i 由物种 a的两个分子和物种 b的 i 个分子组成。从 s_i出发，有 $2i + i(i - 1) = i(i + 1)$ 种可能的转移到 s_{i+1}（通过应用 $a + b$ 或 $b + b$ 规则），以及一种可能的转移到 T（通过应用 $a + a$ 规则）。转移到 T以概率 $\frac{1}{i\,(i+1)\,()+1)} < \frac{1}{i^{-2}}$ 发生。在 s_2中启动一次执行，并令 X_i 为 s_i 转移到 T 这一事件的指示变量。则 $\mathrm{E}[X_i] < i^{-2}$ 且 $\mathrm{E}[\sum X_i] < \frac{\pi^2}{6} - 1 < 1$。因此，并集界给出一个非零概率，即该执行永远不会到达 T，或到达一个无法到达 T的状态。这与定理 4.2 的结论相矛盾，表明对权重施加某种界是必要的。

5.3.2 化学反应网络与边预言机

由于化学反应网络中的智能体数量可能随时间变化，因此如何将边预言机应用于化学反应网络模型并不明显。对于某些特殊情况，或许可以为化学反应网络定义一种扩展，使输出分子继承特定输入分子的隐藏属性，但这样做

可能得不偿失。因此，我们将只关注不含预言机的基础化学反应网络模型。

5.4 同步闲聊模型

群体协议和化学反应网络是异步模型的例子，在这类模型中，每次只发生一次交互或反应。这些模型自然会形成良结构转移系统，因为当 $s \leq t$ 表示 s 是 t 中智能体的一个子集时，我们可以通过让 t 中任何额外的智能体不采取行动，将任意转移 $s \to s'$ 映射为转移 $t \to t'$ 。

在同步模型中则并非如此，因为每个智能体都会在每一步更新其状态。幸运的是，我们通常仍然可以得到一个良结构转移系统：让配置 $t \geq s$ 中对应于 s 的智能体以一种不受 t 中任何额外智能体状态影响的方式进行交互。

本节将介绍若干受经典谣言传播 gossip 算法启发的模型（关于这一领域早期文献的综述，参见 [HHL88]）。gossip 算法有许多变体，因此我们并不声称这些模型必然涵盖此类算法的所有方面；相反，我们借鉴了以往通用闲聊模型的示例，包括 Bertier **等人提出的基于采样的通用匿名闲聊协议** [BBK09]，以及 Bechetti 等人提出的同步均匀 *GOSSIP* 模型 [BCN+14]。这些模型以一种允许进行更一般化计算的方式，将经典谣言传播算法的一些基本假设形式化。

我们始终假设智能体是有限状态的，并且与 POP 一样，将配置定义为 Q^* 某个有限状态空间 Q中的一个序列。子配置关系 $\leq$ 是有序嵌入，因此构成良拟序。加入边预言机后，子配置关系还要求保留边预言机给出的输出；对于特定边预言机，证明这种受限的 $\leq$仍然成立，论证与 POP 的情形相同，因此此处略去。

这些模型大多涉及信息的单向传输。如 §5.2.3 所述，我们假设，若提供了边预言机，则该信息的发送方将作为第一个参数传递给预言机。

5.4.1 GOSSIP

在 Bechetti 等人的模型中，每一步中，每个 n智能体都会抽取另一个智能体，然后根据转移函数 $\delta \quad Q \times Q \to Q$ 更新自身状态；该函数作用于智能体自身的状态以及它所抽取的智能体的状态。采用例如 [KSSV00、KDG03], 中的术语，这是一个**PULL** 模型，因为每个智能体都会从自己抽取的智能体那里拉取信息。

与 POP 中一样，该模型中的配置可以表示为由代理状态构成的向量，其中 $\leq$ 由有序嵌入给出；当 Q 有限时，这使得 $\leq$成为良拟序。

转移关系 $\to$ 与 $\leq$兼容，因为如果 $s \to s'$ 且$s \leq t$，则存在一个转移$t \to t'$，使得在该转移中， s 中的所有代理仅从 s中的其他代理采样，且 $s' \leq t'$。

多项式转移概率可通过如下计算得到：给定转移 $s \to s'$ 和包含关系 $s \leq t$，计算 s中的代理在 s 和 t中采样到相同其他代理的概率；该概率恰为 $|t|^{-|s|}$。由于对于任何固定转移 $s \to s'$， $|s|$ 都是常数，因此该概率关于 $|t|$ 是多项式。

遵循 Bechetti 等人的做法，我们将这种模型称为 GOSSIP。

与 POP 一样，我们可以为 GOSSIP 扩展一个边预言机；在这种情况下，它会告知更新状态的代理自身与其采样的代理之间的关系。具体而言，当 x 对 y进行采样时，它会计算出一个新状态 $\delta(q_x, q_y, A(y, x))$，其中 $A(y, x)$的参数顺序遵循信息发送方在前的惯例。为每个预言机选择恰当的良拟序后，GOSSIP$^{=}$、GOSSIP$^{<}$、GOSSIP$^{\lesssim}$和 GOSSIP$^{\sqsubseteq}$ 都属于 SWSTS 类。

5.4.2 PUSH

再次借用 [KSSV00 的术语，KDG03], GOSSIP 中使用的拉取模型与**推送** 模型形成对照；在推送模型中，信息沿相反方向流动，即从一个代理流向它所抽样的代理。在这里，每一步中，每个代理都根据转移函数 δ: $Q \times \mathbb{N}^Q \to Q$更新其状态，其中 δ 的输入是该代理的状态以及推送给它的状态的无序多重集。我们将这种模型称为 PUSH。

采用有序嵌入子配置关系的 PUSH 模型是一个良结构转移系统，因为如果 $s \to s'$ 和 $s \leq t$，那么存在一个转移 $t \to t'$，其中 $s' \leq t'$，且 s 中的每个代理只接收来自 s 中代理的推送。

证明多项式转移概率比在 GOSSIP 中稍微复杂一些。给定一个转移 $s \to s'$ 和嵌入 f： $s \to t$，仅有

要求 s 中的每个代理都向 s和 $f(s)$中对应的代理 PUSH 还不够；我们还需要保证 $t \setminus f(s)$中的任何新代理都不会向 $f(s)$中的代理 PUSH。记作 $n = |t|$ 和 $m = |s|$。那么，对于 $s \to s'$ 转移中向 $y \in s$ PUSH 的 m 个代理 $x \in s$，每个代理 x 都恰好以 $1/n$的概率向 $f(y)$ PUSH；而对于 $n - m$个代理$z \in t \setminus f(s)$，每个代理都恰好以 $1 - m/n$ 的概率向 $t \setminus f(s)$中的另一个代理 PUSH。这些事件彼此独立，因此它们全部发生的概率恰好为

$$\begin{aligned} n^{-m}\left(1 - \frac{m}{n}\right)^{n-m} &\geq n^{-m}\left(1 - \frac{m}{n}\right)^{n} \\ &= \Theta(n^{-m}), \end{aligned}$$

因为右侧因子在 n 较大时收敛到 $e^{-m} = \Theta(1)$。

给定一个输出值为 V 的边预言机 A，PUSH^A 将转移函数替换为函数 $\delta : Q \times \mathbb{N}^{Q \times V} \to Q$。对于代理 x，计数现在同时跟踪每个代理的状态 y（该代理将内容推送到 x）以及边预言机 A 的输出（y，x）。用于 POP 的相同良拟序可为 $\mathrm{PUSH}^{=}$、$\mathrm{PUSH}^{<}$、$\mathrm{PUSH}^{\lesssim}$ 和 $\mathrm{PUSH}^{\sqsubseteq}$ 生成随机加权转移系统。

PUSH 及其边预言机变体存在的一个问题是，转移函数需要无界空间来存储，因为观察到的状态的多重集可能任意大。对于某些应用，施加一致性条件会很有帮助，例如要求转移函数使用的空间相对于智能体数量呈对数级增长即可完成计算。另一种方案（我们不作探讨）是使用一个固定大小的转移函数，按照任意顺序逐个处理传入的状态，并可能额外使用一个标记来指示在某个步骤中获得的状态何时处理完毕。

5.4.3 匹配模型

这里假设总体规模为偶数，并且在每一步中，对总体应用有向匹配，每对匹配的智能体根据转移函数 $\delta\colon Q^2 \to Q^2$ 更新其状态。

我们将**有向匹配** 定义为将智能体集合划分为若干对，并为每一对指定一个方向。对这类匹配进行采样的一种直接方法是，从智能体的 $n!$ 种可能排列中任选一种，将所得序列拆分成若干对，然后忽略这些对的排列顺序。这样便得到 $\frac{n!}{(n/2)!}$ 种可能的有序匹配，且每种出现的概率相同。

与通常一样，我们令 $s \leq t$ 当存在保持状态的有序嵌入 $f\colon s \to t$ 时成立。这类协议构成良结构转移系统，因为当 $s \to s'$ 和$s \leq t$ 时，至少存在一个转移 $t \to t'$，其中 $s' \leq t'$ 可通过让 $f(s)$ 中的所有智能体按照与 s 中相同的模式彼此匹配而得到。

当 $|t| = n$ 和 $|s| = m = O(1)$ 时，对于 s 上任意固定的有向匹配，若其产生特定转移 $s \to s'$，我们可以通过以下方式构造 t 上的有向匹配，使其将相同的有向匹配应用于 $f(s)$：为所有不在 $f(s)$ 中的智能体选择一个有向匹配，共有 $\frac{(n-m)!}{((n-m)/2)!}$ 种可能的选择。由此，从配置 t 到达配置 ~~$t' \geq s'$~~ 的概率至少为

$$\begin{aligned}\frac{(n-m)!/((n-m)/2)!}{n!/(n/2)!} &= \frac{(n/2)!/((n-m)/2)!}{n!/(n-m)!} \\ &= \frac{\Theta\Big((n/2)^{m/2}\Big)}{\Theta(n^m)} \\ &= \Theta\Big(n^{-m/2}\Big),\end{aligned}$$

对于固定的 m，它关于 n 是多项式的。

我们将此模型称为 MATCHING。加入边预言机后，便得到常见的 SWSTS 变体 MATCHING$^{=}$、MATCHING$^{<}$、MATCHING$^{\lesssim}$ 和 MATCHING$^{\sqsubseteq}$。与其他采用双向通信的模型一样，应用每个预言机时的约定是：每对中的第一个智能体向预言机提供第一个参数。

5.4.4 令牌洗牌模型

Alistarh 等人 [AGR22] 定义了一种同步类 gossip 模型：对于某个固定的 $k \geq 1$，每个 n 智能体根据其当前状态生成 k 个令牌，所有 nk 个令牌在智能体之间均匀洗牌，然后每个智能体根据其先前状态以及其接收到的令牌，按照转移函数 $\delta\colon Q \times Q^k \to Q$ 更新状态。这与 PUSH 模型类似，但保证每个智能体恰好从 k 个智能体处接收令牌，而且智能体也可能接收到来自自身的令牌。

与其他模型一样，当存在从 s 到 t 的有序嵌入 f 时，设 $s \leq t$。给定一个转移 $s \to s'$ 和嵌入 $f\colon s \to t$，至少存在一个相容的转移 $t \to t'$，其中对于每个 $x \in s$，$f(x)$ 接收与 x 相同的令牌。我们希望证明，对于任意固定的转移 $s \to s'$，这类转移发生的概率关于 $n = |t|$ 为多项式级别。和之前一样，我们令 $m = |s|$。

为证明转移概率是多项式级别的，请注意，将 nk 个令牌洗牌分配到 n 个有序序列中、每个序列含 k 个令牌，其可能方式总数恰好为 $(nk)!$，因为我们只需选取一个排列，再将其切分成长度为 k 的若干段。要得到一个与 $s \to s'$ 对应的转移，我们需要 (a) 按照 $s \to s'$ 转移所体现的方式，将 $f(s)$ 中各代理的所有令牌发送给 $f(s)$ 中的其他代理；(b) 将剩余的 $(n-m)k$ 个令牌任意发送给其余的 $n-m$ 个代理。第一部分至少有一种方式，第二部分恰有 $((n-m)k)!$ 种方式，因此

$$\begin{aligned}\Pr\left[X_{i+1} \geq s' \mid X_i = t \geq s\right] &\geq \frac{((n-m)k)!}{(nk)!} \\ &= \frac{1}{(nk)_{mk}} \\ &\geq (nk)^{-mk} \\ &= \Theta(n^{-m}).\end{aligned}$$

我们将这类 SWSTS 称为 SHUFFLE，并定义相应的边预言机类 SHUFFLE$^{=}$、SHUFFLE$^{<}$、SHUFFLE$^{\lesssim}$ 和 SHUFFLE$^{\sqsubseteq}$；按照约定，由 y 发送给 x 的令牌会标记为 $A(x,\ y)$。我们还将 SHUFFLE$_k$ 记作 SHUFFLE 的子类，其中的协议满足每个代理发出 k 个令牌。

5.5 基于图浸入的模型

由于良拟序是良结构转移系统定义中的核心，寻找更多良结构转移系统模型的一个直接方法，是考虑其他类别的良拟序。其思路是利用已知的良拟序在配置上定义拟序，然后寻找与该拟序相容的转移规则。本节将通过一个例子说明这种方法，该例子基于定义在图上的一种特定良拟序。

Robertson and Seymour [RS04]的图小定理表明，在取小图这一操作下，图构成一个良拟序；如果 G 是 H 的小图，那么 G 可以通过收缩 H的某个子图中的边得到。遗憾的是，要提出一个与该良拟序相容且合理的转移规则并不容易，主要是因为不清楚如何处理因边收缩而合并的节点状态。

然而，同一作者后来的一项结果 [RS10,Theorem 1.5]表明，图在**浸入**关系下构成良拟序：该关系将 G 的顶点映射到 H 的顶点，并将 G 的边映射到 H 中互不相交的路径，同时保持

顶点标签上的良拟序（通常，我们将其限制为有限状态集 Q 上的相等关系）。浸入保持**介于关系**：如果 $f : G \to H$ 是一次浸入，且 t 是从 s 到 u 的一条简单路径上的中间顶点，那么 f（t）就是从 f（s）到 f（u）的一条简单路径上的中间顶点。

我们可以据此得到一类 SWSTS，并将其称为 BETWEEN。在这个模型中，每次转移都包括选择一个顶点三元组 s, t, u，使得 t 位于从 s 到 u 的某条简单路径上，并且所有满足条件的三元组被选中的概率均相同。随后，智能体根据转移函数 δ：$Q^3 \to Q^3$ 更新其状态。由于任何这样的三元组 s, t, u 经由任意浸入 f：$G \to H$ 映射后，都会对应于一个由具有相同状态的智能体组成的三元组 $f(s)$ $f(t)$ $f(u)$，且该三元组同样满足介于关系条件，因此任何转移 $G \to G'$ 都会映射为相应的转移 $H \to H'$，其中 $f(s)$ $f(t)$，而 $f(u)$ 会在 H 中根据 s、t 和 u 在 G 中应用的同一转移来更新其状态。我们还有多项式转移概率，因为含有 n 个智能体的图中，可能的转移至多有 $n(n-1)(n-2)$ 个，因此最小转移概率为 $\Omega(n^{-3})$。

重要的是，BETWEEN 应通过限制哪些智能体能够交互来构造，而不是额外提供一个参数，告知任意三个智能体彼此之间的关系。这是因为浸入不保留非介于关系：通过向图 G 添加边，我们可以创建一条新路径，使得在某个 $H \geq G$ 中，t 介于 s 和 u 之间，即便在 G中，t 并不介于 s 和 u 之间。但是，如果我们只允许满足以下条件的三元组$\langle s, t, u\rangle$ 参与交互：t 介于 s和 u之间，那么增加此类三元组只会增加可能的交互，这符合 $\leq$与$\to$ 之间的相容性。使用边预言机扩展 BETWEEN 也会遇到同样的问题，因此我们不考虑该模型的预言机变体。

不可否认，BETWEEN 模型是人为构造的；其动机更多是希望使用某种特定的良拟序，而非声称它代表了一种现实的计算模型。但它确实说明，即使是不同寻常的良拟序也能产生 SWSTS。

6 通过 SWSTS 进行计算

要将 SWSTS 用作计算模型，我们必须能够对输入进行编码并对输出进行解码。这引出了以下扩展定义：**定义 6.1**。一个***SWSTS*** 协议$(S, \leq, \to, |\cdot|, \Pr \Sigma, I, V_0, V_1)$ 由以下部分组成：

- *An SWSTS* $(S, \leq, \to, |\cdot|, \Pr)$.

- 一个有限的输入字母表 Σ。
- 一个输入映射 I： $\Sigma^* \to S$ 对于任意 x, $|I(x)|$ 是关于 $|x|$ 的多项式；并且对于任意 x 和 y，若 $x \leq y$ 在子序列序中成立，则 $I(x) \leq I(y)$。

两个输出集 V_0 和 V_1，它们对 $\leq$ 和 $\to$ 均封闭，分别表示输出0 和1的终止配置。

现在，我们可以通过从配置 $I(x)$ 启动并运行 SWSTS，直到到达 V_0 或 V_1 中的某个配置，来计算输入 x 上的布尔函数 f： $\Sigma^* \to \{0, 1\}$。通常这会涉及较小的出错概率（见第 6.1 节），因此，如果对于每个 x， P 输出 $f(x)$ 的正确值的概率至少为 $1-\epsilon$，我们就称 SWSTS 协议 P 以错误率 ϵ 计算 f。类似地，如果对于每个 x，当 $x \notin L$ 时到达 V_0，或当 $x \in L$ 时到达 V_1 的概率至少为 $1-\epsilon$，我们就称该协议以错误率 ϵ 接受语言 L。

例如，群体协议可以通过让 I 将输入字母映射到一系列智能体的输入状态，同时添加一个领导者代理，并可能添加一些提供额外存储空间的智能体，然后按照 [AAE08] 中的方式运行寄存器机模拟，直到领导者到达一个嵌入输出 0 或 1 的终止状态，从而计算某个函数 f。集合 V_0 和 V_1 将由所有至少包含一个输出为 0 或 1 的领导者代理的配置组成。应用 [AAE 08], 中的误差分析，可以使错误概率 ϵ 在多项式意义上足够小。

分析 SWSTS 协议时，我们希望应用引理 4.1 和定理 4.2，但它们仅适用于封闭 SWSTS。**封闭 SWSTS 协议**是指其 SWSTS 为封闭的协议。

定义 6.1 对输入映射和输出集施加了若干限制。

对于输入映射，要求其将输入规模保持在多项式范围内，这意味着封闭 SWSTS 协议的多项式转移概率会成为输入规模的多项式。输入映射保持子序列序的要求稍显人为，但直观上反映了这样一种想法：初始配置应尽可能体现输入的结构，并且允许将关于初始配置的一些陈述转化为关于输入的陈述（例如，见定理 6.2）。

我们能对输入映射施加的约束，受配置集合 S 的抽象层次所限。例如，我们可能希望通过限制 I （比如限制为有限状态转导器或对数空间转导器），来确保 I 无法替我们完成计算；但由于 S 不一定是一个集

f

字符串集合，我们必须允许更一般的输入映射。我们或许会希望，要求 I 保持子序列序能有所帮助，但一般而言并非如此：给定任意布尔函数f：$\Sigma^* \to \{0, 1\}$，输入映射 I 将 x 发送到 $0^{2|x|+f(x)}$，满足多项式扩展和序约束，并且任何足够聪明、能够计算奇偶性的 SWSTS 都可以借此计算 f。因此，我们必须克制行事，避免选择不合理的输入映射。

对输出集的限制旨在体现这样一种思想：输出可以通过检查配置的一小部分来获得，因此，考虑更多组件或在终止后继续运行协议都不会改变输出。对于涉及智能体的 SWSTS 协议，每个输出集 V_i 通常由所有包含某个标记有输出 i的特殊智能体的配置组成。

然而，坚持要求 V_0 和 V_1 是向上闭的，意味着我们现在通常无法要求它们互不相交。例如，某个群体协议配置若以某种方式多出一个领导者代理，其中的两个领导者代理可能会决定不同的值。通常，我们只能要求：任何可从初始配置到达的配置，至多属于 V_0 和 V_1中的一个。

6.1 错误

与大多数概率计算模型一样，SWSTS 协议也容易出错。SWSTS 方法具有一般性，因此无需参照任何特定模型的细节，我们就能证明这一点。

定理 6.2。 设 $P = (S, \leq, \to, |\cdot|, \Pr \Sigma, I, V_0, V_1)$ 是一个封闭 *SWSTS* 协议。那么，要么 P 在所有执行中都输出相同的值，要么存在一个非空的向上闭集 A，使得对于每个输入 $x \in A$， P以至少多项式概率为 x计算出错误的输出。

证明。 令 $A = I^{-1}$(前驱* $V_0 \cap$ 前驱*(V_1)). Then for any $x \in A$ and $i \in \{0,1\}$，$I(x) \in$ 前驱*(V_i)，由引理 4.1，从 $I(x)$ 启动的执行以至少多项式概率到达 V_i。由于这对V_0 和 V_1都成立，而这两个输出中只有一个对 x 是正确的，因此对任意 $x \in A$， P 都有多项式概率为 x计算出错误的输出。

根据引理 2.1，由于每个输出集 V_i 都是向上闭的，Pred*(V_i) 也向上闭。因此，Pred*(V_0) ∩Pred*(V_1) 作为两个向上闭集的交集，也是向上闭的；又因为 I 是保序的，所以 $A =I^{-1}$(Pred* $V_0 \cap$ Pred*(V_1)) 是向上闭的。

现在假设存在输入 x_0 和 x_1，它们在某些执行中分别产生输出 0 和 1。那么 $I(x_0) \in \mathrm{Pred}^*(V_0)$，且 $I(x_1) \in \mathrm{Pred}^*(V_1)$。令 y 为 x_0 与 x_1 的拼接 x_0x_1。那么 x_0，$x_1 \leq y$ 在子序列序下，因此 $I(x_0)$ $I(x_1)$ $\leq I(y)$。由于每个 $\mathrm{Pred}^*(V_i)$ 都是向上闭的，因此 $I(y)$ 同时属于 $\mathrm{Pred}^*(V_0)$ 和 $\mathrm{Pred}^*(V_1)$，这意味着 y 属于 A。所以，要么 P 在所有执行中都输出相同的值，要么 A 非空。□

定理 6.2 表明，对于几乎所有足够大的输入，非恒定 SWSTS 协议的错误概率都是多项式级的。这是因为，如果 x 是字母表 Σ 上某个非空字符串集合 A 中的元素，且该集合关于子序列关系是向上闭的，那么从 Σ 中随机抽取，直到得到 x 的超序列，所需随机抽取次数的期望值仅为常数 $|x| \cdot |\Sigma|$；根据马尔可夫不等式，这意味着长度为 n 的随机字符串 y 以至少 $1 - \frac{|x| \cdot |y|}{n}$ 的概率具有 $x \leq y$（从而属于 A），而该概率在极限下趋近于 1。这为假设 SWSTS 协议的计算必须允许双边错误提供了某种依据。

7 使用图灵机模拟 SWSTS 协议

SWSTS 协议在达到 V_0 或 V_1 中的某个配置时终止。由于这些集合被要求是向上闭的，定理 4.2 适用：经过多项式期望步数后，封闭 SWSTS 协议要么终止，要么达到一个配置，使其不再可能终止。我们将利用这一事实证明，只要能够高效地表示配置，概率多项式时间图灵机就能以多项式级的小错误概率计算 SWSTS 协议的输出。

我们给出这一结果的两个版本。第 7.1 节给出了在概率多项式时间内模拟 SWSTS 协议的充分条件，而第 7.2 节则给出了更强的条件，使协议能够在概率对数空间内得到模拟。

7.1 在 BPP 中模拟 SWSTS 协议

如果存在一个从配置 s 到某个有限字母表 Σ 上表示形式 $\hat{}s \in \sum^*$ 的映射，使得 (a) $|\hat{}s|$ 是关于 $|s|$的多项式；(b) 存在一台概率多项式时间图灵机，以配置 s 的表示形式 $\hat{}s$ 作为输入，并对每个 $s \to t$ 输出 $\hat{}t$，其概率至少为 $(1 - o(n^{-c}))\ \Pr[s \to t]$（对于任意常数），则称 SWSTS 是**一致的**

$c > 0$；并且 (c) 对任意固定的 s，存在一台图灵机，在给定 ˆt的情况下计算 $s \le t$，所用时间关于 ⌈t|为多项式。

如果一个 SWSTS 协议除了满足上述要求外，还存在一台多项式时间图灵机，以 x 和 $\widehat{\ }$ 为输入并输出 I（x），则称该协议是统一的。

定理 7.1. 设 L 是一个由具有有界双侧错误 $\epsilon < 1/2$ 的统一封闭 *SWSTS* 协议所接受的语言。那么 L 属于 BPP。

证明。设 $P = \langle S, \to, \le, |\cdot|, \Pr\ I, V_0, V_1\rangle$ 是一个接受 L、错误界为 $\epsilon < 1/2$ 的协议。

根据定理 4.2，存在一个常数 k，使得 P 在输入 x 上执行时，要么在期望 $O\left(|I(x)|^k\right)$ 步内到达 V_0 或 V_1 中的某个配置，要么在同样的步数内到达一个无法到达 V_0 或 V_1 的配置。该步数是关于 $n = |x|$ 的多项式。选取一个常数 c，使得该多项式为 $O(n^c)$。

注意到，由于 V_0 和 V_1 都是向上闭的，因此我们可以通过检查 t 是否位于有限个基元素 s中的任一个之上，根据 ˆs在多项式时间内计算某个配置 t是否属于任一集合。

给定输入 x，通过计算 $X_0 = I(x)$，然后不断采样转移 $X_i \to X_{i+1}$，我们可以在 $\widehat{\ }$ 多项式时间内模拟 P 的一次长度至多为 n^{c+1} 的执行，直到以下情况之一发生：(a) 模拟到达 V_0 或 V_1中的某个配置；或 (b) 模拟步骤数超过 n^{c+1}。在第一种情况下，输出相应的值 0 或 1；在第二种情况下，输出 0。第二种情况可能会因过早终止计算而引入额外的错误源，但马尔可夫不等式表明，该情况发生的概率为 $O(1/n)$。

为了界定模拟协议产生正确答案的概率，注意到对于 SWSTS 协议的任意特定执行 $s_0 \to s_1 \ldots s_m$，其发生概率为 $p = \prod_{i=0}^{m-1} \Pr[s_i \to s_{i+1}]$，模拟中选择每个转移 ˆ$s_i \to$ ˆs_{i+1} 的概率至少为$1 - o(n^{-c-2})$ $\Pr[s_i \to s_{i+1}]$，因此采样到该执行的概率至少为 $(1 - o(n^{-c-2}))^t \prod_{i=0}^{m-1} \Pr[s_i \to s_{i+1}] \ge (1 - o(n^{-c-2}))^{n^{c+1}} p = (1 - o(1))p$，因为错误因子在极限情况下趋近于 1。将这些错误汇总到所有接受/拒绝执行上，至多会在我们的错误预算中增加 $o(1)$。

因此，当 n 足够大时，总误差的上界为 $\epsilon + O(1/n) + o(1) < 1/2 - \Omega(1)$，从而使 L 属于 BPP。 □

例如，任何具有多项式时间可计算输入映射 I 的 $\text{POP}^<$ 协议都是一致的，因为我们已经将配置表示为

从有限集合中取值的智能体状态序列，因此满足 (a) $\lceil s \rceil = |s|$；(b) 对 $s \to t$ 进行多项式时间采样，因为我们只需从序列 [5] 中选取一对智能体，并根据可在常数时间内计算的转移函数 δ 更新它们的状态；以及 (c) 对 $s \leq t$ 进行多项式时间判定，因为我们可以用有限状态机检验固定的 s 是否为 t 的子序列。因此，定理 4.2 告诉我们，任何由具有有界双侧错误且输入映射可在多项式时间内计算的 POP$^<$ 协议所接受的语言都属于 BPP。

类似的分析也适用于基础 POP 模型以及任何能产生 SWSTS 的预言机变体 POP^A 。对于具有有理反应速率和多项式时间可计算的输入映射的封闭 CRN 协议，标准模拟算法 [Gil77, GB00] 证明了一致性。类似地，GOSSIP、MATCHING、SHUFFLE 和（假设 δ 可在多项式时间内计算的）PULL 均满足一致性，它们的预言机变体也一样。因此，所有这些协议类都可以在 BPP 中模拟。

目前尚不清楚 BETWEEN 通常是否会产生一致 SWSTS。问题在于，尽管已知图浸入判定是固定参数可处理的 [GKMW11], ，但这一结果是否适用于带标记图的浸入，并非显而易见。因此，定理 7.1 的证明中的模拟可能无法高效地判定任意上闭集中的成员资格 V_0 和 V_1。对于某些特定的 BETWEEN 协议，我们可以避开这个问题：这些协议中， V_0 和 V_1 只需检测某个合适的输出状态中是否存在代理，而这可以通过多项式时间图灵机轻松检查。对于采用此输出约定的协议，我们得到的语言属于 BPP。

7.2 在BPL中模拟对称 SWSTS 协议

在某些情况下，我们可以利用对称性，在 BPL 而非 BPP 中模拟 SWSTS。与 BPP 一样，我们依靠定理 4.2 来保证在多项式时间内终止，但还需要依靠 SWSTS 的其他特定性质来证明它可以在对数空间中进行模拟。其核心思想是：如果 SWSTS 协议中的智能体可以互换，那么我们就可以将配置的多项式大小字符串表示 Σ^* 替换为多重集的对数大小表示 $\mathbb{N}^\Sigma$。

如果存在一个定义在构型上的等价关系 $\sim$ ，使得若 $s \sim s'$，则对于任意 t、$\sum_{t' \sim t}$ ，$\Pr[s \to t'] = \sum_{t' \sim t} \Pr[s' \to t']$，那么称 SWSTS 是**对称的**。

[5] 如果假设我们的概率图灵机只能访问随机比特，那么除非 n 和 $n-1$ 都是 2 的幂，否则无法保证在多项式时间内准确完成这一操作；但在一致性的定义中允许存在指数级小的相对误差 $1 - o(n^{-c})$，因此这一点得到了处理。

令 $[s]$为包含 s的等价类，这意味着，从 $[s]$ 中的一个元素转移到 $[t]$ 中的一个元素的概率 $\Pr[[s] \to [t]] = \sum_{t' \sim t} \Pr[s \to t']$ 是良定义的，且不依赖于代表元 s的选择。

若封闭 SWSTS 满足以下条件，则称其为**对称对数空间一致**：(a) 它具有对称性；(b) 存在一个从配置 s 到某个有限字母表 Σ 上的表示 $\hat{s} \in \Sigma^*$ 的映射，并且该映射保留对称性，即若 $s \sim t$，则 $\hat{s}$ 是 $\hat{t}$ 的一个排列；(c) 存在一个概率对数空间转导器，以配置s 的表示 $\hat{s}$ 作为输入，并输出一个配置 $\hat{t}$，且对于任意常数$c > 0$，在 $[t]$ 中输出某个 t'的概率至少为 $(1 - o(n^{-c}))\Pr[[s] \to [t]]$；(d) 对任意固定的 s，存在一台图灵机，给定 $\hat{t}$ 后，使用关于 $|\hat{t}|$的对数空间计算 $s \le t$。

若封闭 SWSTS 协议满足对称对数空间一致的定义，除上述要求外，还需存在一个对数空间转导器，它以 $\widehat{x}$ 为输入，并生成一个属于 $I(x)$的排列的输出。

定理 7.2。 任何由对称的对数空间一致 *SWSTS* 协议接受的语言都是对称的，且属于BPL。

证明。 对称性显然成立：对于协议在某个输入 x 上进行并到达 V_0 或 V_1 的任意计算，在 x 的任意排列上都存在一个等价计算，以相同的概率到达同一集合。

证明其包含于 BPL 与定理 7.2 的证明基本相同，不同之处在于我们将全程把多项式时间步骤替换为对数空间步骤，并且必须谨慎表示中间配置，以免使用超过对数空间的空间。

由于 $\Pr[[s] \to [t]]$ 不依赖于我们从等价类中选取哪些成员，因此我们可以将 $[\hat{s}]$ 表示为 $\mathbb{N}^{\Sigma}$ 中的一个元素 $\widehat{[s]}$，方法是统计 s 中每种符号出现的次数。给定这些计数（所需空间为关于 $|\hat{s}|$ 的对数空间），我们可以模拟对数空间转导器的输入：通过选取 s 的一个有序排列，并根据 $[s]$ 中的计数，计算每个模拟输入带位置上的 $\widehat{\ }$ 符号，从而进行采样 $\to$。类似地，每当转导器向其输出带写入一个符号时，我们就将该操作替换为递增一个计数器，最终得到一个由计数构成的向量 $\widehat{[t]}$。

应用输入映射 I 并执行终止测试 V_0 和 V_1 时，采用相同的技术来表示配置的等价类。SWSTS 的对称性意味着，将每种表示替换为

其排序后的版本，不会影响模拟中这些部分的任何行为。

误差分析与定理 7.1 的证明相同。 □

不难证明，POPs 中的任何协议都满足必要的对称性，并且其转移关系可以由概率对数空间转导器计算。我们也可以在对数空间内轻松地针对固定的 s 检验 $s \leq t$。因此，任何由 POP 接受且具有对称、对数空间可计算输入映射的语言，都是 BPL 中的对称语言。

假设输入映射适当，则对于具有有理反应速率的封闭 CRN 协议，以及 GOSSIP、MATCHING 和 SHUFFLE，也可以证明类似的结果。对于同步 gossip 模型，抽样比 POP 或 CRN 稍微复杂一些，因为我们必须谨慎，确保每个代理或令牌至多使用一次。不过，可以通过迭代适当的无放回抽样来实现：使用计数器记录初始配置中尚未使用的代理数量，并在使用后递减；同时累积记录看到特定值或令牌组合的代理数量，并在每次迭代中按需递增。这之所以可行，只是因为在这些模型中，每一步代理可能具有的视图和先前状态的集合大小都是常数。

对于 PUSH，我们必须更加谨慎。显而易见的方案是跟踪哪些状态被推送给了哪些智能体，但这种方法行不通，因为每个智能体看到的状态数可能从 0 到 n 不等，因此可能产生超出对数空间存储能力的视图。不过，如果我们假设单个智能体的转移函数仅取决于所见状态的数量，那么就可以通过对每个此前未见过的智能体进行采样，判断它是否推送到 a，从而分别对每个目标代理 a 的视图进行采样；推送到 a 的智能体计入相应的计数，而没有推送的智能体则放回池中，留待下一个智能体采样。对于每个智能体上对称且可在对数空间内计算的转移函数，这满足定理的要求，表明这一特定受限版本的 PUSHi 仅接受 BPL 中的对称语言。

8 使用 SWSTS 协议模拟图灵机

群体协议 [AAE08] 和化学反应网络 [CSWB09]已知能够模拟概率图灵机，通常是通过

寄存器机等中间表示来实现的。在本节中，我们将展示，许多在智能体上具有或能够编码线性结构的 SWSTS 协议，可以直接以每一步较小的错误概率模拟图灵机，从而能够计算 BPP 中的任意谓词。

我们首先聚焦于 POP 及其边预言机变体，然后通过展示如何在其他各种 SWSTS 模型中模拟 POP^A ，将问题归约到这些情形；这些模型使用相同的边预言机，或使用在能力上等价于 BETWEEN 中适当边预言机的其他结构。

8.1 在POP^A中模拟图灵机

对于基本的群体协议模型，已有结果 [AAD$^+$06、AAE08]表明，可以实现一种寄存器机，其计算能力几乎等同于工作带长度为对数级的概率图灵机，并且每一步发生错误的概率都小到多项式级别。这里的“几乎”是因为无法表示输入字符串中符号的顺序：由于智能体之间具有对称性，这些模拟机器只能在任意排列下观察输入。尽管如此，这些结果表明，POP 可以计算 BPL 中的任意对称语言。结合定理 7.2 可知，未经修改的 POP 恰好计算 BPL 中的对称语言。

加入边预言机可能会显著增强模型的能力。在 $\mathrm{POP}^<$ 中，大多数智能体都有一个直接前驱和后继，我们可以利用这一结构来模拟图灵机纸带，并将每个单元格分配给某个智能体。由于智能体无法直接观察后继关系（如 §5.2.4 中所述，这会产生一个非良构转移系统），因此它们需要反复运行一个子协议，以找到当前磁带单元的前驱或后继。这并不困难，但需要等待子协议完成；而这又需要构建某种时钟机制，并允许时钟在子协议收敛之前滴答时出现多项式概率的失败。另一个问题是，我们既需要指定一名领导者，也需要数量与输入规模成多项式关系的智能体来表示整条纸带，不过这些都可以由输入映射提供。相关构造的详细信息见第 8.1.1 节，其中证明了 BPP $\subseteq \mathrm{POP}^<$。

事实证明，$\mathrm{POP}^=$ 也足以模拟概率多项式时间图灵机。由于边预言机不会为智能体提供顺序，我们可以改为将第 i 个磁带单元表示为同一等价类中的 i 个智能体的集合，并使用比较协议在

$b<0 \ \rightarrow -<0$	生成 $-$
$-<0 \ \rightarrow -<0$	
$+<0 \ \rightarrow -<0$	
$0<b \ \rightarrow \ 0<+$	生成 $+$
$0<- \ \rightarrow \ 0<+$	
$0<+ \ \rightarrow \ 0<+$	
$0<- \ \rightarrow \ 0<b$	移除错误的 $-$
$+<0 \ \rightarrow \ b<0$	移除错误的 $+$
$-<- \ \rightarrow \ b<-$	修剪 $-$
$+<+ \rightarrow +<b$	修剪 $+$

表 1：在 POP 中查找前驱和后继$^{<}$

需要时在 i和 $i-1$或 $i+1$之间移动。这需要在输入映射中进行更多一些设置，以提供编码 m 个磁带单元所需的 $\Theta(m^2)$个智能体，并设置它们的颜色以便适当地将其分组。完整构造见第8.1.2节，其中证明了 BPP $\subseteq$POP$^{=}$。

8.1.1 在POP中模拟图灵机$^{<}$

给定一台在 n^c 步内完成运行的概率图灵机，我们将使用输入映射构造一条由 n^c 个智能体组成的模拟纸带，外加少数几个辅助智能体。这些辅助智能体包括一个领导者代理L，负责协调模拟并保存图灵机读写头的状态；一个计时器智能体 T，用于实现一个简单时钟，如 [AAD^{+}06] 中所示；以及两个智能体 R_0 和 R_1，用于实现一个简单的抛硬币机制。很容易看出，此输入映射既是保序的，也是可在对数空间内计算的。

大部分工作由领导者代理完成，它将直接负责模拟图灵机读写头并更新纸带智能体的内容。尽管如此，纸带智能体仍需执行一个协议，以查找当前位置的前驱或后继。

该协议在表 1 中进行了描述。为简便起见，表中仅列出了发起者所在位置低于响应者所在位置的转移；但我们认为，每个这样的转移都对应一个对称转移，其中智能体的位置互换。我们还省略了不起作用的转移，以及涉及领导者代理的所有转移；领导者代理的角色仅限于在经过适当时长后切换中央智能体的状态。

最初，当前读写头位置上的智能体被赋予状态 0。这会使它分别以状态 $-$ 和 $+$ 招募其左侧和右侧的智能体。等待足够长的时间，使紧邻的前驱和后继很可能已被招募后，中央智能体会被切换为空白状态 b。随后会逐步剔除多余的 $-$ 和 $+$ 智能体，直到只剩下紧邻的前驱和后继。

由于第一阶段的每一步都会以 $\Theta(N^{-2})$ 的概率招募紧邻的前驱，等待 $\Omega(N^3)$ 步后，紧邻的前驱仍未被招募的概率会指数级低。[6] 紧邻的后继也同样如此。对于第二阶段，我们可以利用 [AAE08] 中对通过消除进行的领导者选举的分析，得出将 $-$ 或 $+$ 智能体的数量减少到一个所需的期望时间为 $O(N^2)$。同样，等待 $\Omega(N^3)$ 步后，失败概率会指数级小。

遗憾的是，引理 4.1 告诉我们，在任何 SWSTS 中，任何试图等待超过常数步数的做法，失败概率至少为多项式级。因此，要等待 $\Omega(N^3)$ 步，我们能做的最多也不过是采用 [AAD+06] 中的简单时钟：领导者代理等待连续 k 次遇到唯一的计时器智能体，且期间不见到任何其他智能体。正如 [AAD+06 中观察到的，定理 24]，表明，平均而言，这需要 $O(N^{k+1})$ 步；简单的并集界也表明，它在 $O(N^{\ell})$ 步内完成的概率为 $O(N^{\ell-k-1})$。因此，适当选择 k 后，表 1 中所需的 $\Omega(N^3)$ 步延迟可以在任意指定的多项式级失败概率下实现。

现在我们可以描述图灵机模拟。每一步中，领导者首先通过等待看到 R_0 或 R_1来抛硬币。然后，它等待遇到标记为持有当前读写头位置的纸带智能体，根据被模拟的图灵机的转移函数更新存储在该处的纸带符号，并将读写头标记替换为 0 标记，以便执行表 1 中描述的邻居查找协议。随后，它执行该协议，找到前一读写头位置的直接前驱和后继。根据读写头向左还是向右移动，领导者等待遇到 $-$代理或 $+$ 代理，并将其标记为新的读写头位置。

[6]如果智能体被标记了错误的方向，新招募的前驱或后继可能会被尚未移除的错误标记智能体剪除。不过，标准的优惠券收集问题界表明，经过 $\widetilde{O}(N^2)$ 步后仍有任何错误智能体残留的概率极小，呈指数级衰减；而在我们的图灵机模拟中，这个问题并不会出现，因为从来不会创建这类错误的旁侧智能体。

因此，我们可以用任意多项式级小的失败概率模拟概率图灵机的一步。通过适当调整指数并采用并集界，即使经过 n^c 步，失败概率仍然是多项式级小的。这表明 BPP $\subseteq$ POP$^<$。

由定理 7.1 可立即得出反方向的包含关系。因此我们有：

定理 8.1. $\mathrm{POP}^<$ =BPP。

8.1.2 在$\mathrm{POP}^=$中模拟图灵机

在这一构造中，我们将特别不吝使用智能体。回想一下，使用 n 个智能体的 POP 能够模拟一个空间为 $O(\log n)$、每步错误概率为 [AAE08] 的概率图灵机。为了在 POP$^=$ 中实现一台运行 n^c 步后停止的图灵机，我们将创建 n^c 个等价类，每个等价类包含 Θ个（n^c）智能体；每个等价类将实现一台独立的图灵机，该图灵机除了存储内容和可能的读写头状态外，还能存储特定磁带单元的数值指标。与 POP$^<$ 构造一样，可以通过保序对数空间可计算输入映射来搭建这些图灵机，并初始化指标和状态。

为了模拟完整的图灵机，当前持有读写头的那个智能体等价类 i 负责计算下一步，并相应地更新本地磁带值。随后，它必须找到自己的前驱或后继，以移交读写头的控制权。这可以通过以下方式实现：(a) 随机选择另一个等价类作为候选新位置，然后 (b) 根据需要，测试这个其他等价类的指标是否为 $i-1$ 或 $i+1$。如果测试失败，就重试。

随机选择候选等价类，只需让 i 等待与其他某个等价类中的领导者相遇即可，预期需要多项式次交互。将该候选领导者标记以供后续参考后，可以使用 [AAE08] 的技术，在预期多项式次交互内测试另一类是否具有指标$i \pm 1$：让领导者协调，对各自存储的指标副本进行递减，直到其中一个减至零，再测试另一个副本的值。这会带来显著开销，因为每次递减后，领导者都必须等待才能直接通信，但总开销仍是多项式的。如果测试失败，就取消对候选领导者的标记，并选择新的候选者。经过预期多项式次尝试后，就能找到正确的前驱类或后继类。

误差分析与 POP$^{<}$ 情形类似：我们只需将每一步的多项式错误概率设得足够小，使其即使乘以很大但仍为多项式的步数，结果也仍然是多项式小量。这得到 BPP $\subseteq$ POP$^{=}$；而利用定理 7.1 得到另一个方向的结果

定理 8.2. $\mathrm{POP}^{=}$ =BPP。

8.2 在其他模型中模拟 POP^{A}

为了证明其他 SWSTS 模型能够计算 BPL 或 BPP 中的函数，我们模拟 POP^{A}。通常，这需要证明某个模型能够执行一个循环，在循环中反复选择两个智能体，分别作为 POP 交互的发起者和响应者，并利用该模型可用的预言机来模拟 POP^{A} 预言机。

这有多困难，在一定程度上取决于这些模型与 POP 的差异有多大。我们可能需要克服的两个主要困难是：模型可能不包含智能体之间的直接交互，因而无法同时更新两个智能体的状态；而在具有间接观察的模型（如 GOSSIP）中，可能有多个智能体观察同一个智能体。我们先（在 §8.2.1 中）说明在 GOSSIP^{A} 中模拟 POP^{A} 时如何克服这两个问题。其他模型的模拟（将在后续各节中介绍）将允许采用更简单的构造。

8.2.1 在POP^{A} 中模拟GOSSIP^{A}

在GOSSIP^{A} 中模拟POP^{A} 的一步操作，需要由唯一的领导者代理协调一系列阶段。该领导者代理使用唯一的计时器智能体来实现 [AAD^{+}06] 中计时器机制的一种变体。其余的 n跟随者代理表示POP^{A} 协议中的代理。输入映射负责设置这些代理的初始状态。

为避免混淆 GOSSIP 协议与其所模拟的协议，我们将每个 GOSSIP 智能体状态中控制其行为的组成部分称为其角色。

由于 GOSSIP 允许多个智能体从单个智能体拉取数据，因此在每个阶段挑选出单个发起者和响应者并非易事。相反，领导者可以为这些角色招募多个候选者，然后应用类似领导者选举的算法来消除重复者。

另一个复杂之处在于，选定的发起者和响应者智能体并不直接交互；相反，响应者会等待采样发起者，并计算两个智能体的新状态，然后利用 GOSSIP

Role	观测状态	更新规则
idle	招募发起方	角色 ←发起者
idle	招募响应方	角色 ←候选响应方
发起者	招募响应方	角色 ←候选响应方
发起者	发起者	角色 ← 空闲
候选响应方	候选响应方	角色 ←空闲
候选响应方	更新	角色 ←响应者
响应者	发起者， q_i,A	角色 ← 发送；（q'， q） ← δ（q_i， q， A）
发起者	发送， q'	角色 ←空闲； $q \leftarrow q'$
发起者	重置	角色 ← 空闲
候选响应方	重置	角色 ← 空闲
响应者	重置	角色 ← 空闲
发送	重置	角色 ← 空闲

表 2：在 GOSSIP^A 中模拟 POP^A 的跟随者转移

机制将这一变化传达给发起者。让响应者执行计算是边预言机定向所要求的：按照预言机报告 $A(x,\ y)$、其中 x 是信息发送方的约定，只有响应者能够获取预言机的正确值。

对于领导者代理，角色将依次循环经过招募发起方、招募响应方、剪除多余者、更新和重置这些取值；其中，前两个角色各维持一步，而其余角色的预期维持时间为 $\Theta(n^k)$ 步，方法是让领导者等待，直到连续 k 次观测到唯一的计时器智能体。领导者中还包含一个额外状态分量，用于记录连续观测计时器智能体的当前次数。

对于跟随者代理，每个代理都维护三个分量：

- 角色是其在模拟协议中的当前角色。
- q 是其在模拟的 POP 协议中的状态。
- $q\prime$ 是一个额外的状态分量，仅供模拟发起者使用，以向模拟响应者传达更新后的状态。

在初始配置中，每个跟随者都处于空闲角色，其状态q 等于输入映射所选 POP^A 中的状态。

表 2 展示了这些组件如何在跟随者智能体中根据对领导者及彼此的观测而演化。

引理 8.3。 固定 $c \geq 2$，并令 $k = 3c$。那么，上述协议以 $O(n^{-c})$ 的失败概率，在期望时间 $O(n^{4c})$ 内，模拟具有转移函数 δ 的 POP^A 协议的 $O(n^c)$ 步。

证明。 让我们看看由领导者的五种角色对应的五个阶段组成的一个周期内会发生什么。

每个周期可能以两种方式之一失败。**全局失败**会不可逆地破坏模拟：如果发生全局失败，协议就会结束，且对 POP^A 的模拟也会失败。我们将证明，每个周期发生全局失败的概率仅为 $O(n^{-2c})$，而在全部 $\Theta(n^c)$ 个模拟步骤中发生全局失败的概率为 $O(n^{-c})$。**局部失败**的破坏性较小，只会阻止当前周期取得进展。局部失败可能以恒定概率在每个周期发生，平均而言会使模拟速度减慢一个恒定倍数。

每个周期的阶段按如下方式进行：

1. 在领导者处于招募发起方角色的唯一一步中，每个 n 个跟随者代理都有独立的 $\frac{1}{n+2}$ 概率观察到这一角色并切换为发起者。至少一个智能体切换为发起者的概率为 $1 - \left(1 - \frac{1}{n+2}\right)^n$，在极限情况下趋于常数 $1 - 1/e$。

   这一阶段不会导致全局失败，但如果最终配置中没有角色为发起者的智能体，则可能导致局部失败。

2. 在接下来的步骤中，领导者的角色为招募响应方，同样的论证表明，至少有一个智能体获得候选响应方角色的概率为常数。

   允许将发起者覆盖为候选响应方的规则，仅仅是为了使这一论证成为可能。然而，它也带来了一个不幸的副作用：上一阶段的所有发起者都可能被覆盖。只有当某个智能体在两个步骤中都观察到领导者时，这一事件才会在该智能体处发生，因此其概率为 $\frac{1}{(n+2)^2}$，。由此，根据并集界，所有发起者都被覆盖的概率至多为 $\frac{n}{(n+2)_2} = O(1/n)$。覆盖所有发起者会导致局部失败。

   这一阶段也不会导致全局失败，但它有一定概率（至多为常数）以如下状态结束：没有智能体担任发起者角色，或没有智能体担任候选响应者角色。这两种结果都会导致局部失败，使当前周期不起作用。

3. 在接下来的预期 $\Theta(n^k) = \Theta(n^{3c})$ 步中，只要领导者担任剪除多余者角色，且候选响应者代理不止一个，那么每个此类代理每步至少有 $\frac{1}{n+2}$ 的概率观察到另一个担任相同角色的代理，并切换回空闲状态。因此，对于每个候选者，经过预期 $O(n)$ 步后，它要么退出，要么成为唯一剩下的候选者。不过，这个过程仍有几种可能出错的情况：

(a) 移除多余候选者所需的实际时间超过 $O(n^c)$。根据马尔可夫不等式，每个多余候选者在每个长度为 $O(n)$步的时间间隔内退出的概率至少为一个常数，因此经过 $O(n^2) = O(n^c)$ 步后，仍有任何多余候选者存活的概率会呈指数级小。这类失败将被视为全局失败。

(b) 到达更新阶段（从而将候选响应者代理切换为响应者）之前的时间太短。这里可以使用并集界来说明这种情况发生的概率为 $O(n^c \cdot n^{-k}) = O(n^{-2c})$。

这类失败也将被视为全局失败。

(c) 最后剩下的候选者在同一步观察到彼此，并同时退出。令 X_t 表示经过 t 步后存活的候选者数量，并令 τ 表示满足 $X_\tau > 1$的最后一步。于是

$$\begin{aligned}
\Pr[X_{\tau+1} = 0] &= \sum_{\ell=2}^{n} \Pr[X_\tau = \ell] \cdot \Pr[X_{\tau+1} = 0 \mid X_\tau = \ell] \\
&= \sum_{\ell=2}^{n} \Pr[X_\tau = \ell] \cdot \left(\frac{\left(\frac{\ell-1}{n+2}\right)^{\ell}}{\left(\frac{\ell-1}{n+2}\right)^{\ell} + \ell\left(\frac{\ell-1}{n+2}\right)^{\ell-1}\left(1 - \frac{\ell-1}{n+2}\right)} \right) \\
&\leq \sum_{\ell=2}^{n} \Pr[X_\tau = \ell] \cdot \frac{1}{2} \\
&= \frac{1}{2}.
\end{aligned}$$

相同的分析也适用于担任发起者角色的智能体。假设我们启动剪除多余者阶段时，候选响应方和发起者这两个角色各至少有一个智能体，那么在该阶段结束时，这两个角色各有多个智能体留下的概率为 $O(n^{-2c})$，且至少有 $1/2$ 的概率使得至少有一个

智能体留在每个角色中，因此至少有 $1/4$的概率使得每个角色中至少有一个智能体留下。
这会以 $O(n^{-2c})$ 的概率导致全局失败；在本周期此前未发生失败的条件下，局部失败的概率至多为 $3/4$ 。

4. 在接下来的 $\Theta(n^k) = \Theta(n^{-3c})$ 个期望步骤中，领导者执行角色更新。假设之前的阶段都按预期运行，因此恰有一个智能体 i 担任发起者角色，另有一个智能体 r 担任候选响应方角色。

请注意，根据对称性，跟随者代理的每个有序对成为 $(i,\ r)$ 的可能性都相同。因此，模拟过程符合 POP^A 模型的均匀调度假设。

我们可以考虑三个连续的子阶段：

(a) 经过期望为 $O(n)$ 的步骤后，r 将观察到更新，并切换为响应者。(b) 再经过期望为 $O(n)$ 的额外步骤后，r 将观察到 i，计算联合转移函数 $\delta(q_i,\ q_r,\ A(i,\ r))$ 的输出，并相应地更新自身状态。额外的分量 q' 将存储为 i更新后的状态，而 r 的角色将切换为发送，以告知 i 该值已是当前值。(c) 再经过期望为 $O(n)$ 的步骤后，i 将观察到处于发送角色的 r ，把自身状态更新为 q'，并切换为空闲角色。此时，i 和 r 的状态都与它们在 POP^A中会选择的状态相同。

这三个时间间隔合计为 $O(n)$ 个预期步骤。按照前一阶段的同样论证，在这些时间间隔结束之前该阶段完成的概率为 $O(n^{-2c})$。

如果候选响应方角色中有多个代理，或发起者角色中有多个代理，这种情况已经被计为全局失败，因此我们无需再次考虑。

这样就剩下 i 或 r 缺失的情况。如果只有 i缺失，r 看到发起者时不会因此更新其状态，因此该子阶段结束时，它的角色要么是候选响应方，要么是响应者；无论哪种情况，它都不会更新状态。如果只有 r 缺失，i 将永远看不到角色为发送的代理，因此也不会执行任何更新。如果两者都缺失，则什么也不会发生。这三种情况都属于局部失败。

5. 在最后的 $\Theta(n^k)$ 个预期步骤中，领导者的角色为重置，任何仍处于候选响应方、响应者、发送或发起者角色的智能体，都会在 $O(n)$ 个预期步骤内重置为空闲。在成功的阶段中，处于发送角色的智能体只会有一个；但如果前一阶段导致局部失败，该阶段的规则也会清理其他角色中残留的任何智能体。

假设此前没有发生全局失败，则每个目标角色中最多有一个代理，并且此阶段的失败概率为 $O(n^{-2c})$。

我们可以用并集界对所有全局失败概率求和，从而得到每个周期发生全局失败的概率为 $O(n^{-2c})$。对于局部失败，在此前未发生失败的条件下，每个阶段或子阶段避免局部失败的概率至少为一个常数；将这些条件概率相乘，可得整个周期中任何地方都不发生局部失败的概率为一个常数。

局部失败只是漏掉步骤，因此只会影响模拟速率中的常数项。由此可知，模拟 POP^A 的 $\Theta(n^c)$ 个步骤只需要 $\Theta(n^c)$ 个 GOSSIP^A 协议周期，对应于 $O(n^{c+k}) = O(n^{4c})$ 个步骤。这 $\Theta(n^c)$ 个周期内发生全局失败的总概率为 $O(n^c) \cdot O(n^{-2c}) = O(n^{-c})$。 □

不难看出，GOSSIP$^<$ 和 GOSSIP$^=$ 都满足定理 7.1 的一致性条件，因此这两个类都属于 BPP。我们还可以将引理 8.3 与定理 8.1 和 8.2 中的模拟结合起来，证明反向包含关系。由此可得：

定理 8.4。 GOSSIP$^<$ =GOSSIP$^=$ =BPP。

没有预言机时，引理 8.3 表明 POP $\subseteq$ GOSSIP。结合定理 7.2，可知 GOSSIP 能够计算 BPL 中的对称语言。

8.2.2 模拟 POP^A 在PUSH^A 或$\mathrm{SHUFFLE}^A$中

在 PUSH^A 中，每一步中，每个智能体只能被另一个智能体看到。这样就可以通过在智能体之间传递一个活动令牌，简单地模拟基于 POP^A 的协议。我们仍然需要一个领导者代理，但其角色仅限于充当中立的仲裁者，在每个周期中均匀地选择一个发起者，从而避免因发起者是由前一周期中因某种角色而被选中的智能体选出而可能产生的偏差。本质上

同一算法也适用于 SHUFFLE^A：只需将模拟令牌视为显式令牌即可。不过，我们还必须处理智能体接收到自己发送的令牌的情况。

该模拟的转移表见表3。表中每组转移都用于实现活动令牌的移动，而活动令牌由角色表示：start、initiator、wait+restart、restart、send、wait+send 或 done。附加角色 wait 和 receive 标记具有特殊角色的代理，这些角色会影响它们处理传入令牌的方式。每个代理还会存储其模拟状态 q 以及可选的额外状态 q'，以便在本周期中传递给模拟发起者。

转移由代理的当前角色以及它观察到的活动令牌（如果有）控制。在 PUSH^A 模型中，这种观察表现为接收到一次推送。在 SHUFFLE^A 模型中，这种观察表现为接收到一个令牌。在后一种情况下，我们假设只有持有活动令牌的代理会发出非空令牌，而其他代理则发出默认的空令牌。在 PUSH^A 模型中，代理无法选择是否发出推送，但我们假设，与活动令牌不对应的角色所发出的推送会被忽略。

我们省略了表中代理观察到与自身当前角色相同角色的转移；在这种情况下，代理不会更新其角色，而是会在下一步重试。这些转移只会在 SHUFFLE^A 模型中造成问题。

在初始配置中，领导者代理的角色为启动，跟随者代理的角色为空闲，每个跟随者代理的 q 字段均设为其所模拟的 POP^A 代理的状态。

协议的一个正常周期模拟 POP^A 的一步，并按以下方式进行：

1. 领导者将其启动令牌发送给一个跟随者；该跟随者采用发起者角色，成为模拟步骤的发起者。领导者切换到等待角色，直至收到周期完成的通知。

2. 发起者将其发起者令牌发送给一个模拟响应者；模拟响应者使用随附的状态和预言机 A 的输出，计算模拟群体协议的转移关系 $(q', q) \leftarrow \delta(q_i, q, A)$。响应者进入发送($q'$)角色，将更新后的发起者状态发送回发起者；发起者则切换到接收角色，等待接收该状态。

3. send(q') 令牌在总体中游走（如果被领导者持有，可能会暂时切换为 wait+send(q') 令牌），直到它

当前角色	观察到的状态	更新规则	新角色
启动 idle	启动		wait 发起者(q)
发起者 wait idle	发起者(q_i) 发起者(q_i) A	(q, q') $\leftarrow \delta(q_i, q, A)$	接收 等待+重新启动 发送(q')
等待+重新启动 idle 接收	等待+重新启动 等待+重新启动		wait 重新启动 发起者(q)
重新启动 idle wait 接收	重新启动 重新启动 重新启动		idle 重新启动 等待+重新启动 发起者(q)
发送(q') idle wait 接收	发送(q') 发送(q') 发送(q')	$q \leftarrow q'$	idle 发送(q') 等待+发送(q') done
等待+发送(q') idle 接收	等待+发送(q') 等待+发送(q')	$q \leftarrow q'$	wait 发送(q') done
done wait	done		idle 启动

表3：通过令牌传递模拟 POP^A

到达发起者。此时，发起者更新其状态并发出一个完成令牌。

4. 完成令牌同样会在总体中游荡，直到到达领导者；领导者通过将其角色切换为启动来开始下一个周期。

不难看出，协议的正常周期会以相同的概率选择每个可能的发起者和响应者 $\frac{1}{n(n+1)}$，并且会在开始下一个周期之前正确更新两个智能体的状态。

一种复杂情况是，如果发起者令牌返回到领导者，而不是传递给另一个智能体，那么领导者就不能充当响应者，因此需要重新启动发起者—响应者交互。为此，领导者会切换到等待+重新启动角色，该角色会以等待+重新启动或重新启动的形式在总体中传播，直到到达先前选定的发起者；该发起者随后从接收切换回发起者。

图1展示了该协议的配置类别及其之间的转移。每个类别首先以持有活动令牌的智能体的角色标记，后面跟着总体中存在的其他非空闲角色。链中的每次转移发生的概率至少为 $\frac{1}{n+1}$，因此起始配置以概率 1 被无限次重新访问，这意味着模拟的 POP^A 协议会执行无限多步。稍加推导还可以证明，期望减速比为 $O(n)$，其中大部分耗时都花在等待发送令牌或完成令牌到达正确的智能体上。

与 GOSSIP^A 模拟不同，这种模拟不会遇到无法恢复的错误，并且可以无限期地继续模拟 POP^A 在 PUSH^A 或 SHUFFLE^{A}中。我们可以将其表述如下：

引理8.5。对于任意边预言机 A，$\mathrm{POP}^A \subseteq \mathrm{PUSH}^A$ 以及$\mathrm{POP}^A \subseteq \mathrm{SHUFFLE}^A$。

由此可得

定理8.6。$\mathrm{PUSH}^<$ =$\mathrm{PUSH}^=$ = $\mathrm{SHUFFLE}^<$ = $\mathrm{SHUFFLE}^=$ = BPP，假设 $\mathrm{PUSH}^<$ 和$\mathrm{PUSH}^=$ 被限制为多项式时间可计算的转移函数。

与 GOSSIP 一样，通过类似的归约到 POP 可以证明，在没有预言机并施加适当限制的情况下，PUSH 和 SHUFFLE 在 BPL 中恰好接受对称语言。

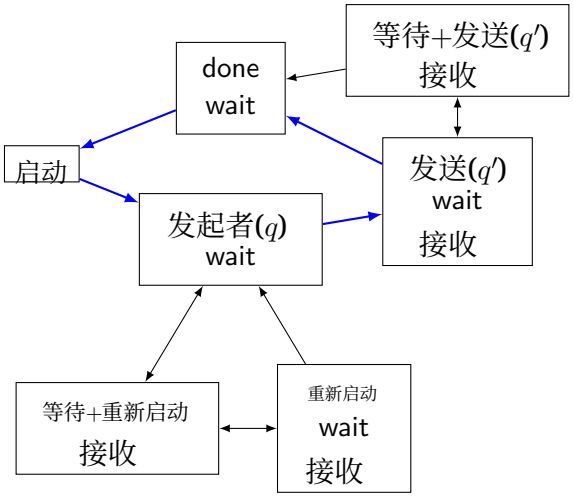


图1：在 PUSH^A 或 SHUFFLE^A 中模拟 POP^A 的转移图。典型执行周期以蓝色突出显示。

ℓ_0，空闲(q) $\to \ell_1$，发起者(q)ℓ_1，空闲(q) $\to \ell_2$，响应者(q) 发起者(q_1)，响应者(q_2) $A \to$ 完成(q_1')，完成(q_2') 其中 $q' = \delta(q_1, q_2, A)\ell_2$，完成($q$) $\to \ell_3$，空闲(q)ℓ_3，完成(q) $\to \ell_0$，空闲(q)

表4：在 MATCHING^A 中模拟 POP^A 的转移规则

8.2.3 在POP^A 中模拟MATCHING^A

可以像PUSH^A 或SHUFFLE^{A}中那样，使用相同的令牌传递方法来模拟 POP^A 中的 MATCHING^A ，不过还有一种更简单的模拟方法，它利用了智能体之间的直接交互。

表4给出了该模拟的转移规则。其思路是使用一个领导者代理，在每个模拟步骤中随机标记一个发起者和一个响应者。随后，这些智能体直接交互，并将自身标记为完成。领导者收集这些标记后，再选择新的发起者和响应者。由于该协议不需要任何时序假设，因此无需单独的计时器智能体。

很容易证明，这模拟了一个协议

POP^A，没有错误，且减速幅度为 $O(n^2)$。

将此模拟与我们先前的结果结合起来可得：

定理 8.7。 MATCHING$^<$ =MATCHING$^=$ =BPP。

此外，假设输入映射可在对数空间内计算，MATCHING 也能接受 BPL 中的对称语言。

8.2.4 模拟 $\mathrm{POP}^<$ 在BETWEEN中

模拟 POP$^<$ 在 BETWEEN 中很简单，因为我们使用输入映射来选择一个路径形式的图，其中每个智能体在路径上的位置都对应于全序 $<$。

给定一个使用 n 个智能体、转移函数为 δ 的 POP$^<$ 协议，通过扩展输入映射来添加新智能体 L 和 R，并添加 POP$^<$ 配置的左右端点，然后按照全序 $<$将这些节点组织成一条路径，即可构造一个包含 $n+2$ 个智能体的 BETWEEN 协议。

回顾一下，BETWEEN 中的任何转移都涉及三个智能体的三元组 s,t,u，其中 t 位于从 s 到 u 的一条简单路径上。令任何涉及此类三元组但不恰好包含 L 或 R 中一个的转移均不起作用。对于三元组 L,a,b，将 a 和 b 的状态更新为 $\delta(a,b,<)$。对于三元组 a,b,R，将 a 和 b 的状态更新为 $\delta(b,a,>)$。在限定所选择的转移恰好包含 L 或 R 中一个的条件下，所模拟的群体协议中的每个 $n(n-1)$ 个智能体有序对被选中的概率相等。因此，除去因无效转移导致的预期 $O(n)$ 倍减速外，所得的 BETWEEN 协议恰好执行与原始 POP$^<$ 协议相同的转移。

可以采用类似的方法来模拟 POP$^=$：利用屏障标记 $|$分隔等价类，使用输入为 $\langle a,|,b\rangle$ 的转移来检测 a 和 b 是否不在同一等价类中，并通过在足够长的时间内未观察到此类转移，以概率方式检验 a 和 b 是否处于同一等价类中。不过，细节较为繁琐，而包含关系 POP$^<$ $\subseteq$BETWEEN 已经足以得到：

定理 8.8。$\mathrm{BETWEEN} = \mathrm{BPP}$。

如 §7.1 所述，BETWEEN $\subseteq$BPP 的包含关系要求 BETWEEN 协议采用一种特定的输出约定，即由单个智能体在其状态中报告输出。该定理假设采用这一约定。

9 结论

我们已经证明，将良结构转移系统的定义扩展为包含配置的权重以及关于这些权重呈多项式关系的转移概率后，便可以证明一大类采用有限状态主体和随机调度的分布式计算模型的一般性结果。特别是，我们可以证明，这类随机良结构转移系统中的计算若能终止，就必定会在多项式时间内终止；这也根据标准复杂度类，精确刻画了其中许多模型的计算能力。

一个自然的问题是：群体协议、化学反应网络和八卦模型的哪些其他性质能够推广到足够抽象的统一模型中？例如，Angluin等人 [AAER07] 关于刻画群体协议中稳定计算的计算能力的研究，很大程度上依赖于此类协议配置的良拟序，但还需要额外的加法结构，而这种结构由更一般的向量加法系统模型 [KM69]所捕获。或许可以针对更加一般的良结构转移系统模型得到类似的结果。同样，Mathur 和 Ostrovsky [MO22] 关于自稳定群体协议的结果，归根结底也建立在配置的良拟序之上；似乎很可能也能针对一般的良结构转移系统得到类似的结果。

对于采用随机化调度的系统，我们的结果只能给出时间上界。能否对良结构转移系统施加某种概率调度约束，从而获得执行特定计算任务所需时间的有用的一般下界，类似于已知的群体协议和化学反应网络下界 [DS18, CCDS17]？一种可能的思路是找到一种抽象的密度概念（Doty [Dot14]曾对其进行探讨），使其在足够短的转移序列下保持不变；这样或许可以避免通过给配置指定显式权重、给转移指定概率而给良结构转移系统增加的某些复杂性。。

参考文献

[AAC+05] Dana Angluin、James Aspnes、Melody Chan、Michael J. Fis- c her、Hong Jiang 和 René Peralta。网络图的稳定可计算性质。载于 Viktor K. Prasanna、Sitharama

Iyengar、Paul Spirakis 和 Matt Welsh（编），传感器系统中的分布式计算：第一届 *IEEE* 国际会议，*DCOSS 2005*，玛丽娜德尔雷伊，加利福尼亚州，美国，*2005*年*6*月*/7*月，论文集，计算机科学讲义第3560卷，第63–74页。施普林格出版社，2005年6月。[AAD+06] Dana Angluin、James Aspnes、Zoë Diamadi、Michael J. Fischer 和 René Peralta。被动移动有限状态传感器网络中的计算。*Distributed Computing*，第235–253页，2006年3月。[AAE08] Dana Angluin、James Aspnes 和 David Eisenstat。带领导者的群体协议快速计算。*DistributedComputing*，21(3)：183–199，2008年9月。[AAER07] Dana Angluin、James Aspnes、David Eisenstat 和 Eric Rup- pert。群体协议的计算能力。分布式计算，20(4)：279–304，2007年11月。[AAG18] Dan Alistarh、James Aspnes 和 Rati Gelashvili。群体协议中的空间最优多数。载于 Artur Czumaj（编），第二十九届 *ACM-SIAM* 离散算法年会论文集，*SODA 2018*，新奥尔良，路易斯安那州，美国，*2018*年*1*月*7*日至*10*日，第2221–2239页。SIAM，2018年。[AGR22] Dan Alistarh、Rati Gelashvili 和 Joel Rybicki。快速图形群体协议。载于第二十五届分布式系统原理国际会议，2022年。[AvJT96] Parosh Aziz Abdulla、K¯arlis Čer¯ans、Bengt Jonsson 和 Yih-Kuen Tsay。无限状态系统的一般可判定性定理。载于第十一届 *IEEE* 计算机科学逻辑年会论文集，第313–321页。IEEE，1996年。[BBK09] Marin Bertier、Yann Busnel 和 Anne-Marie Kermarrec。论八卦与群体。载于结构信息与通信复杂性：第十六届国际研讨会，*SIROCCO 2009*，皮兰，斯洛文尼亚，*2009*年*5*月*25*日至*27*日，修订精选论文*16*，第72–86页。Springer，2009年。[BCN+14] L Becchetti、A Clementi、E Natale、F Pasquale 和 R Silvestri。八卦模型中的相对多数共识。载于论文集

第二十六届 *ACM-SIAM* 离散算法年度研讨会，第 371–390 页。SIAM，2014 年。[BL23] Michael Blondin 和 François Ladouceur。无序数据群体协议。*arXiv* 预印本 *arXiv:2305.00872*，2023 年。[BNFK18] Lucas Boczkowski、Emanuele Natale、Ofer Feinerman 和 Amos Korman。随机群体中可靠信息流的限制。《*PLoS* 计算生物学》，14(6)，2018 年。[CCDS17] Ho-Lin Chen、Rachel Cummings、David Doty 和 David Soloveichik。化学反应网络计算中的速度故障。*Distributed Computing*，30(5)：373–390，2017 年。[CSWB09] Matthew Cook、David Soloveichik、Erik Winfree 和 Jehoshua Bruck。化学反应网络的可编程性。载于 Anne Condon、David Harel、Joost N. Kok、Arto Salomaa 和 Erik Winfree 编，《算法生物过程》，自然计算系列，第 543–584 页。Springer，2009 年。[Dic13] Leonard Eugene Dickson。具有 n 互异素因子的奇完全数和本原盈数的有限性。*American Journal of Mathematics*，35(4)：413–422，1913 年。[DKNV24] Niccolò D'Archivio、Amos Korman、Emanuele Natale 和 Robin Vacus。含噪 PULL 模型中的快速、鲁棒信息传播。*CoRR*，abs/2411.02560，2024 年。[Dot14] David Doty。化学反应网络中的定时。载于第二十五届 *ACM-SIAM* 离散算法年度研讨会论文集，第 772–784 页。SIAM，2014 年。[DS18] David Doty 和 David Soloveichik。群体协议中的稳定领导者选举需要线性时间。*Distributed Comput.*，31(4)：257–271，2018 年。[EGLM16] Javier Esparza、Pierre Ganty、Jérôme Leroux 和 Rupak Majumdar。群体协议的模型检测。载于第*36*届 *IARCS* 软件技术与理论计算机科学基础年度会议（*FSTTCS 2016*），第 27–1 页。Schloss Dagstuhl–Leibniz-Zentrum für Informatik，2016 年。[Fel68] William Feller。概率论及其应用导论，第 1 卷。Wiley，第三版，1968 年。

[Fin87] Alain Finkel。将 Karp 和 Miller 的过程推广到良结构转移系统。载于自动机、语言与编程：第*14*届国际研讨会，德国联邦共和国卡尔斯鲁厄，*1987*年*7*月*13–17*日，论文集 *14*，第 499–508 页。Springer，1987年。[FS01] Alain Finkel 和 Philippe Schnoebelen。无处不在的良结构转移系统！理论计算机科学，256(1-2)：63–92，2001 年。[GB00] Michael A Gibson 和 Jehoshua Bruck。对含有大量物种和大量通道的化学系统进行高效、精确的随机模拟。《物理化学杂志*A*》，104(9)：1876–1889，2000 年。[GGJS23] Adam Gańczorz、Leszek Gasieniec、Tomasz Jurdzinski 和 Grzegorz Stachowiak。选择性群体协议。$CoRR$，abs/2305.08460，2023 年。[Gil77] Daniel T Gillespie。耦合化学反应的精确随机模拟。《物理化学杂志》，81(25)：2340–2361，1977 年。[GKL+25] Leszek Gąsieniec、Łukasz Kuszner、Ehsan Latif、Ramviyas Parasuraman、Paul Spirakis 和 Grzegorz Stachowiak。简短公告：通过空间群体协议实现匿名分布式定位。载于第四届动态网络算法基础研讨会（*SAND 2025*），第 19–1 页。Schloss Dagstuhl–Leibniz-Zentrum für Informatik，2025 年。[GKMW11]Martin Grohe、Ken-ichi Kawarabayashi、Dániel Marx 和 Paul Wollan。寻找拓扑子图是固定参数可处理的。载于第四十三届 *ACM* 理论计算研讨会论文集，第 479–488 页，2011 年。[GR09] Rachid Guerraoui 和 Eric Ruppert。名字胜过恶意：微型移动代理能够容忍拜占庭故障。载于自动机、语言与编程国际研讨会，第 484–495 页。Springer，2009 年。[GS18] Leszek Gasieniec 和 Grzegorz Stachowiak。群体协议中快速且空间最优的领导者选举。载于论文集

第二十九届 *ACM-SIAM* 离散算法年会，第 2653–2667 页。SIAM，2018 年。[GS20] Leszek Gąsieniec 和 Grzegorz Stachowiak。增强型相位时钟、群体协议与快速且空间最优的领导者选举。《*ACM*学报》（*JACM*），68(1):1–21，2020 年。[GSS21] Leszek Gasieniec、Paul Spirakis 和 Grzegorz Stachowiak。新型时钟、最优直线形成与自复制群体协议。*arXiv* 预印本 *arXiv:2111.10822*，2021 年。[HHL88] Sandra M Hedetniemi、Stephen T Hedetniemi 和 Arthur L Liestman。通信网络中的闲聊传播与广播综述。*Networks*，18(4):319–349，1988 年。[Hig52] Graham Higman。抽象代数中按整除关系排序。伦敦数学学会会议录，3(1):326–336，1952 年。[KDG03] D. Kempe、A. Dobra 和 J. Gehrke。基于闲聊的聚合信息计算。载于第 *44* 届 *IEEE* 计算机科学基础年会论文集，*2003* 年，第 482–491 页，2003 年。[KM69] Richard M. Karp 和 Raymond E. Miller。并行程序模式。《计算机与系统科学学报》，3(2):147–195，1969 年。[Kru60] Joseph B Kruskal。良拟序、树定理与 Vazsonyi猜想。《美国数学会汇刊》，95(2):210–225，1960 年。[Kru72] Joseph B Kruskal。良拟序理论：一个屡被独立发现的概念。《组合理论学报》*A*辑，13(3):297–305，1972 年。[KSSV00] R. Karp、S. Schindelhauer、S. Shenker 和 B. V ocking。随机化谣言传播。载于第 *41* 届计算机科学基础年会论文集，第 565–574 页。IEEE，2000 年。[KU18] Adrian Kosowski 和 Przemysław Uznański。群体协议速度很快。*arXiv* 预印本 *arXiv:1802.06872*，2018 年。

[MCS11] Othon Michail、Ioannis Chatzigiannakis 和 Paul G. Spirakis。中介群体协议。理论计算机科学，412(22):2434–2450，2011 年。[MO22] Shaan Mathur 和 Rafail Ostrovsky。自稳定群体协议的组合刻画。信息与计算，285:104829，2022年。[NW63] C. St. J. A. Nash-Williams。有限树的良拟序。剑桥哲学学会数学论文集，59(4):833–835，1963 年。[RS04] Neil Robertson 和 P.D. Seymour。图小量 XX：Wagner猜想。《组合理论学报》*B*辑，92(2):325–357，2004 年。纪念 W.T. Tutte 教授的专刊。[RS10] Neil Robertson 和 Paul Seymour。图小量 XXIII：Nash-Williams 的浸入猜想。《组合理论学报》*B*辑，100(2):181–205，2010 年。[SCWB08] David Soloveichik、Matthew Cook、Erik Winfree 和 Jehoshua Bruck。有限随机化学反应网络中的计算。自然计算，7:615–633，2008 年 12 月。[SS13] Sylvain Schmitz 和 Philippe Schnoebelen。良结构系统的威力。载于并发理论国际会议，第 5–24 页。Springer，2013 年。