arXiv:2301.03487v1 [cs.CC] 9 Dec 2022

对 Sopin《PH =PSPACE》*的批评

Michael C. Chavrimootoo、Ian Clingerman 和 Quan Luu

计算机科学系 罗切斯特大学 罗
切斯特，纽约州 14627，美国

2022年12月9日

摘要

我们对 Valerii Sopin 的论文《PH = PSPACE》 [Sop14]进行了批评。该论文声称，通过对量化布尔公式中的存在量词进行斯科伦化，证明 QBF（一种广为人知的 PSPACE 完全问题）属于 Π_4^p，从而证明 PH = PSPACE，进而解决理论计算机科学中的一个重大未解决问题。在这篇批评中，我们指出了该论文存在的问题，并得出结论：它未能证明 PH =PSPACE。

1 引言

在本文中，我们批评了 Valerii Sopin 的论文《PH =PSPACE》 [Sop14]。该论文提出了两个定理。第一个定理涉及量化布尔公式的斯科伦化，而第二个定理试图利用第一个定理使多项式层次坍缩，并证明 PH = PSPACE。在进一步展开之前，我们先讨论这一结果的重要性。

在计算复杂性理论中，多项式层次定义为 PH =P ∪NP ∪NP^{NP}∪$\text{NP}^{\text{NP}^{\text{NP}}}$∪···，而 PSPACE 指的是能够用至多多项式空间求解的问题类。二者都是极其重要且受到深入研究的问题类。已知 PH ⊆ PSPACE。然而，PH 是否等于 PSPACE，至今仍是理论计算机科学中的核心未解决问题之一。简而言之，这是因为现有的大多数技术都无法解决这个问题。[1] 证明 PH = PSPACE [Far20]会带来许多推论。假设有人能够证明 PSPACE 的复杂度不高于 PH 的某一层 k，这将意味着该层 k 以上的所有层都会塌缩到第 k 层（因为它们都包含于 PSPACE）。即使 k 的值很大，这种塌缩也有助于推动复杂性理论中其他未解决问题的研究。例如，PH = PSPACE 将意味着 PH 完全问题的存在，这是一个目前尚无定论的问题（事实上，这等价于 PH 是否

*部分由美国国家科学基金会资助，项目编号为 CCF-2006496。[1]这是因为，根据 Baker、Gill和Solovay [BGS75]以及 Yao [Yao85]的结果，解决 PH与PSPACE 的证明无法相对化。

会塌缩，即是否存在某个 i，使得 PH $= \Sigma_i^p$ ）。这一结果的另一个重大影响是厘清 L类与 PH 或 NP.[2] 等类之间的关系

Sopin的论文中用于声称证明其定理的方法如下：首先对任意量化布尔公式应用斯科伦化，然后从一个能够求解 Π_4SAT 的算法构造一个能够求解量化布尔公式的算法。然而，我们将说明这些定理的证明存在缺陷。

在第2节中，我们介绍 Sopin 的证明中使用的定义和符号。在第 3 节和第 4 节中，我们总结定理1和定理2的要点，并分析每个证明。我们将在第 5 节中指出论文语境下使用斯科伦化的一个关键观察。最后，在第 6 节中，我们总结对该论文的批评。

2 预备知识

这里我们介绍复杂性理论中的许多标准概念。在大多数现代复杂性教材中，都可以找到等价的定义 [HO02、AB09、Sip13]。

关于符号，给定一个字符串 w，令 $|w|$ 表示 w的长度。此外，令 $\mathbb{N} = \{0, 1, 2, \ldots\}$，并令 $\mathbb{N}^+ = \{1, 2, 3, \ldots\}$。

本文中，我们将讨论布尔公式和布尔变量。我们假设（布尔）变量的取值始终为 1（真）或 0（假）。含有布尔变量 $x_1, \ldots, x_n$ 的布尔公式 ϕ 记作 $\phi(x_1, \ldots, x_n)$。量化布尔公式的形式为 $(Q_1 x_1) \cdots (Q_n x_n)[\phi(x_1, \ldots, x_n)]$,，其中对每个 $i \in \{1, \ldots, n\}$，均有 Q_i 为 $\exists$或 $\forall$。

现在我们来定义多项式层次。固定 $i \geq 1$。语言 L 属于 Σ_i^p，当且仅当存在一个多项式 q 和一个多项式时间可计算谓词 R，使得对所有 x，都有

$$x \in L \iff (\exists w_1 : |w_1| \leq q(|x|))(\forall w_2 : |w_2| \leq q(|x|) \cdots (Q_i w_i : |w_i| \leq q(|x|))[R(x, w_1, \ldots, w_i)],$$

其中，当 i 为奇数时， Q_i 为 $\exists$；当 i 为偶数时， $\forall$ 为 L'。此外，语言 L' 属于 Π_i^p，当且仅当 $\overline{L'} \in \Sigma_i^p$。多项式层次定义为类 PH $= \bigcup_{i \in \mathbb{N}^+} \Sigma_i^p$。[3] 此定义得到的类与另一种常见定义相同；我们在引言中采用的正是这种定义，即 $\Sigma_3^p =$NP$^{\text{NP}^{\text{NP}}}$。

对于上述各类（PH除外），我们可以定义以下典型的完全问题。下面的定义改编自Arora和Barak [AB09]给出的定义。对于每个 $i \geq 1$，问题 Σ_iSAT是Σ_i^p 完全的，其定义为：所有形如$(\exists u_1) \cdots (Q_i u_i)[\psi(u_1, \ldots, u_i)]$ 且为真的量化布尔公式所构成的集合，其中 ψ是一个布尔公式， u_1、 u_2、 …、 u_i 分别表示布尔变量列表；当 i为奇数时， Q_i为 $\exists$，而当i 为偶数时， Q_i为 $\forall$。问题 Π_iSAT是Π_i^p 完全的，其定义与此类似，不同之处在于：当 i 为奇数时， Q_i 为$\forall$ ，而当 i 为偶数时， Q_i 为 $\exists$ 。

PSPACE 是由在多项式空间内运行的图灵机所接受的语言类。形式上，PSPACE $= \bigcup_{k \in \mathbb{N}^+}$确定性空间复杂度$[n^k]$。此外，“PSPACE 体现了多项式有界量词的能力” [HO02]。因此，QBF（所有真量化布尔公式的集合），

[2]L类是指可由确定性图灵机在对数空间内求解的判定问题类。已知L $\subseteq$ P。因此，如果L $=$ NP，那么L $=$ P $=$ NP $=$ PH。根据空间层次定理，还可知L $\neq$ PSPACE。因此，如果PH $=$ PSPACE，那么L $\neq$ PH，从而L $\neq$ NP。[3]熟悉这些概念的读者会注意到，我们没有明确提及 Σ_0^p，但它当然包含于 Σ_1^p 之中，因此实际上并未遗漏。这一“遗漏”仅仅是因为本文其余部分不需要再引用该类，所以我们省去了定义它的麻烦。

是一个典型的 PSPACE 完全问题。很容易看出，QBF $= \bigcup_{i\in\mathbb{N}} \Sigma_i\text{SAT} \cup \Pi_i\text{SAT}$，因此多项式层次 $\subseteq$ PSPACE。

斯科伦化是从量化布尔公式中移除存在量化变量，并用斯科伦常量或斯科伦函数替代它们的过程；这些斯科伦函数仅涉及量词顺序中出现在被移除的存在量化变量之前的全称量化变量。感兴趣的读者可参阅 Chang 和 Lee 的教材 [CL73] 以了解更多详情。

3 关于定理 1

现在我们将重点讨论 Sopin的论文中的定理 1。为清晰起见，我们对该定理进行了重新表述。

定理 1（[Sop14]）。对于任意 $n \in \mathbb{N}^+$，令 $\Phi = (\Omega_1 x_1)(\Omega_2 x_2)\cdots(\Omega_n x_n)[\phi(x_1, \ldots, x_n)]$ 为任意布尔公式，其中 $(\Omega_1, \ldots, \Omega_n) \in \{\exists, \forall\}^n$。令 $I = \{i \mid i \leq n \wedge \Omega_i = \exists\}$。则， Φ是一个真的量化布尔公式，当且仅当对于每个 $i \in I$，存在一个关于 $\{x_j \mid j < i \wedge \Omega_j = \forall\}$ 中的变量的布尔函数 y_i，使得将每个 x_i 替换为 y_i 后得到的量化布尔公式是一个永真式。

据我们所知，该定理本质上是在提出斯科伦化保持可满足性，因为定理所描述的过程与斯科伦化相似。然而，众所周知，斯科伦化确实保持可满足性（相关参考可见 Chang 和 Lee [CL73] 以及 Loveland [Lov78]的教材）。为了利用这一技术判定量化布尔公式，至少需要证明斯科伦化的空间复杂度存在一个多项式上界。遗憾的是，目前尚未知晓这样的结果。我们将在第 5 节中进一步阐述这一点。此外，我们还注意到定理 1 的给定证明存在若干问题，并将在本节其余部分对这些问题进行讨论。

在定理 1 的证明中，作者引入了一种递归算法，用于检验量化布尔公式中的成员资格。该算法依次移除量词，并根据被移除的量词生成一个新的逻辑等价公式。随后，作者补充说明“a[boolean] function determines the truth table” [Sop14] of its variables。接着，证明通过两个例子来帮助说明其主要论点。

首先，我们对证明的结构进行评论。证明的各个部分如何彼此关联、又如何与主要论证相联系，并不清楚。因此，在整个分析过程中，我们将尝试理解证明各部分的作用。

Sopin 给出了一个用于判定量化布尔公式的递归算法，并声称定理 1 的证明可由此推出；然而，该算法定义并不明确。算法中唯一得到定义的是递归步骤：取出第一个量化变量，并检查该变量的两个取值。根据量词的不同，利用该变量两种可能取值对应的两个方程形成一个新方程。在该算法的定义中，既没有基本情形，也没有递归调用。要说明算法何时终止，需要基本情形；要说明递归步骤在何处应用，则需要递归调用。因此，我们无法确定所给算法是否确实是递归算法，还是只取出了第一个量词。我们认为，这似乎是一个常见的量化布尔公式判定算法，但不清楚定理 1 的证明如何由此推出。

现在，我们想将注意力转向算法描述末尾的一句话：“请注意，一个 [布尔] 函数决定了真值表（一一对应）” [Sop14]。此处“真值表”一词含义模糊。如果论文确实打算像定理 1 的例 1（稍后讨论）那样用真值表表示斯科伦函数，那么这就令人担忧，因为真值表的大小随其变量数量呈指数增长。另一方面，也有可能引文所述仅仅是布尔函数与真值表之间关系的一个事实。尽管如此，论文仍未展示这种转换可能带来的空间复杂度，而这对于处理 PSPACE 完全问题至关重要。

在定理 1 的证明末尾，作者给出了两个例子，以便更清楚地阐释该定理。例1明确描述了一个斯科伦函数：存在量化变量可以重写为公式中位于其前面的变量的函数。例子最后提到，斯科伦函数“确实是真值表，其中 x_1、.....、x_{k-1} 的取值决定了 x_k 的取值” [Sop14]。如前所述，在此语境中使用真值表会导致规模呈指数级膨胀。例2只是用一个明确的量化布尔公式重述了该定理。例2指出：“当且仅当存在这样的布尔函数 y_1：$\{0,1\} \to \{0,1\}$、y_2：$\{0,1\}^2 \to \{0,1\}$、y_3：$\{0,1\}^3 \to \{0,1\}$，使得 $\phi(x_1, y_1(x_1)\ x_2, y_2(x_1, x_2)\ x_3, y_3(x_1, x_2, x_3))$ 是永真式时，$\forall x_1 \exists z_1 \forall x_2 \exists z_2 \forall x_3 \exists z_3$ 才是一个真命题” [Sop14]。这个例子只是展示了经过斯科伦化后的公式是什么样子。该公式必须是永真式，才能成为真量化布尔公式，因为此时只剩下全称量化变量。用关于恰当变量的布尔函数替换存在量化变量的做法似乎是正确的，但既没有证明这些函数是斯科伦函数，也没有与定理 1 的证明建立直接联系。

因此，尽管该陈述为真，给出的证明却不足以支持这个定理。鉴于我们的观察表明，目前尚不清楚如何在多项式空间内计算斯科伦化，因此我们不清楚这个定理将如何在 Sopin 的论文其余部分发挥作用（事实上，我们将在下一节讨论，我们看不到这种联系）。

4 关于定理 2

本节将重点讨论 Sopin 的论文中的以下定理。

定理 2 ([Sop14])。 Π_4^p =PSPACE。

正如预期的那样，所谓的证明从任意一个量化布尔公式 Φ 出发，并据此在多项式时间内构造一个量化布尔公式 Φ'（它具有一种特定的语法形式，我们将在本节稍后说明），使得 $\Phi \in$ QBF $\iff$ $\Phi' \in \Pi_4$SAT。先说明一点：论文中的证明往往含义不清，并且包含若干逻辑跳跃，因此我们将基于自己对作者意图的最佳理解来探讨证明中的论证。

我们的第一条评论涉及表述清晰度，具体来说是量化布尔公式的形式。作者假设量化布尔公式具有如下形式（在本文中，我们经常将其称为“标准”形式）

$$(\forall x_1)(\exists y_1) \cdots (\forall x_n)(\exists y_n)[\phi(x_1, y_1, \ldots, x_n, y_n)],$$

对某个 $n \in \mathbb{N}^+$，乍看之下这似乎不正确。例如，公式 $(\exists x, y)(\forall z)[(x \vee y \vee z)]$ 显然属于量化布尔公式，但论文并未讨论它。事实上，文中给出的通用形式似乎遗漏了变量个数为奇数的公式、以存在量词开头的公式，以及多个变量受同一个量词约束的公式。然而，论文没有明确说明，所有此类公式都可以在多项式时间内转换为“标准”形式。将任意量化布尔公式转化为“标准”形式的关键是使用哑变量。[4] 我们的例子，

$$(\exists x, y)(\forall z)[(x \vee y \vee z)],$$

本段前面介绍的形式体现了“标准”形式似乎存在的所有问题。因此，我们将以非正式的方式（因为这个问题相当简单，用例子说明即可，无须赘述）展示如何将上述公式转换为“标准”形式。首先，我们引入虚变量α，使第一个量词成为全称量词。由此得到公式

$$(\forall \alpha)(\exists x, y)(\forall z)[(x \vee y \vee z)].$$

接下来，我们引入虚变量 β，将变量 x 和 y 分开，使每个量词只约束一个变量。由此得到公式

$$(\forall \alpha)(\exists x)(\forall \beta)(\exists y)(\forall z)[(x \vee y \vee z)].$$

最后，由于我们需要偶数个变量，因此引入虚变量 γ ，得到

$$(\forall \alpha)(\exists x)(\forall \beta)(\exists y)(\forall z)(\exists \gamma)[(x \vee y \vee z)].$$

不难看出，如果原公式有 n 个变量，那么新公式最多有 $2n+2$ 个变量。

现在我们来关注定理 2 的证明是否正确。在本节余下部分，固定某个 $n \in \mathbb{N}^+$ 以及一个包含 $2n$ 个变量的布尔公式（不含量词） ϕ ，考虑以下量化布尔公式

$$\Phi = (\forall x_1)(\exists y_1) \cdots (\forall x_n)(\exists y_n)[\phi(x_1, y_1, \ldots, x_n, y_n)].$$

本文试图从 Φ 构造出以下公式（声称在逻辑上等价，但并非“标准”形式）

$$\begin{aligned}\Phi' =& (\forall x_1, \ldots, x_n)(\exists y_1, \ldots, y_n)[\phi(x_1, y_1, \ldots, x_n, y_n) \wedge \\ & (\forall \hat{x}_n)(\exists z_n)[\phi(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, \hat{x}_n, z_n)] \wedge \\ & (\forall \hat{x}_{n-1}, \hat{x}_n)(\exists z_{n-1}, z_n)[\phi(x_1, y_1, \ldots, x_{n-2}, y_{n-2}, \hat{x}_{n-1}, z_{n-1}, \hat{x}_n, z_n)] \wedge \cdots \wedge \\ & (\forall \hat{x}_2, \ldots, \hat{x}_n)(\exists z_2, \ldots, z_n)[\phi(x_1, y_1, \hat{x}_2, z_2, \ldots, \hat{x}_n, z_n)]],\end{aligned}$$

使得 $\Phi \in \mathrm{QBF} \iff \Phi' \in \Pi_4\mathrm{SAT}$。

尝试给出的证明是对公式中的变量数量进行归纳。我们不再重复其中的完整论证，并建议感兴趣的读者直接查阅原

[4]对我们的目的而言，虚变量是指被量化、但未出现在布尔公式中的变量。这当然是合法的。例如，我们可以说公式 $(\forall x)(\exists y)(\forall z)[(x \vee y)]$ 是关于变量集 $\{x, y, z\}$ 的。在这种情况下， z 是一个虚变量。

论文。所谓归纳证明的要点如下：我们可以交换原公式中量词的位置，然后在多项式时间内判断这些改动是否影响公式的真值。我们首先指出，这种方法有两个主要问题：（1）它没有利用定理 1 陈述所蕴含的结论；（2）归纳法实际上并未证明逻辑等价。

我们先讨论（1）。在所谓的定理 2 证明中，唯一提及定理 1 的地方，是变量数量 m 大于或等于 3 的情形。论文称，只要“去掉第一个量词，并按照我们在定理 1 中的 [方式] 检查 [中第一个变量的两种可能取值，就会得到 $m-1$ 情形” [Sop14]。值得再次强调的是，定理 1 的尝试性证明不过是给出了一种（众所周知的）递归算法，用于在多项式空间内判定量化布尔公式。因此，尽管这种一次“消去”一个变量的方法或许有助于判断公式是否为真，但它可能会使用指数级的（非确定性）时间，而且论文中并没有明确说明如何使用该算法。

现在讨论（2）。在所谓的归纳证明中，论文声称，对于含两个变量 ψ 、 x 和 y 的任意量化布尔公式，有 $(\forall x)(\exists y)[\psi(x,y)] \equiv (\exists y)(\forall x)[\psi(x,y)]$ 当且仅当 ψ 既不是异或函数，也不是异或函数的否定；这一点确实成立。因此，论文中的论证指出，只要在固定除两个变量之外所有变量的取值（其中一个变量作全称量化，另一个作存在量化）时，ψ 不可能是异或函数（或其否定），归纳就成立。Φ' 中的附加子句旨在为这一论证提供支持。然而，论文不仅没有解释这些附加子句如何发挥作用、为何能够发挥作用，似乎还遗漏了一些情形。事实上，论文称上述检验可以在多项式时间内完成，因为“这样的公式只有 [个”] n^2 [Sop14]。我们认为，这一结论是通过从 Φ 中选出一个作全称量化的 n 个变量之一，以及一个作存在量化的 n 个变量之一而得出的。然而，论证遗漏了一点：其余每个 $2n-2$ 个变量都可以取两个值之一（0 或 1），因此会产生 $n^2 2^{2n-2}$ 个需要检验的公式。（值得注意的是，上述检验很容易在非确定性多项式时间内完成。然而，论文只写了“多项式时间”，按惯例这意味着“确定性多项式时间”。）因此，我们认为该证明中的归纳过程并不正确。

顺带简要提及一下，还有其他一些小错误，但我们对此不予关注，因为它们的重要性不及我们指出的那些错误。此外，在该证明的最后部分，论文提到公式的代数正规形对论证至关重要，并给出了一个例子，但并未明确说明代数正规形为何至关重要。因此，我们无法确定作者是否希望实现与这份技术报告所传达内容不同的目标。

5 关于斯科伦化的注记

我们增加本节，希望能更好地解释斯科伦化的用法。我们认为，斯科伦函数的大小必然会在证明中发挥作用。事实上，如果这类函数的大小不受多项式上界约束，那么就很难理解如何能在多项式空间内计算出任何斯科伦化公式（并使其在此处成为一种有用的方法）。另一方面，如果斯科伦函数的大小有多项式上界，那么我们就可以

证明一些令人震惊的结论。

命题3. 如果存在一个多项式 p： $\mathbb{N} \to \mathbb{N}^+$，使得对于每个量化布尔公式Φ，对 Φ 进行斯科伦化后生成的斯科伦函数大小都不超过 $p(|\Phi|)$，那么 Σ_2^p =PSPACE。

证明。$\subseteq$ 关系是众所周知的，因此只需证明在上述陈述的假设下，PSPACE $\subseteq \Sigma_2^p$。我们将通过证明 QBF $\in \Sigma_2^p$ 来做到这一点。令 p 为命题陈述中定义的多项式。不失一般性，假设输入是一个量化布尔公式（如果不是，我们可以在多项式时间内轻松检测出来，并立即拒绝）。固定一个任意的布尔公式 Φ，其中有 n 个变量，作为输入。令 E表示 Φ中被存在量化的变量集合。由于斯科伦化保留可满足性，因此可知 $\Phi \in$ QBF $\iff$ 存在一些斯科伦函数，使得对于每个不属于 E 的变量赋值，将 E 中的每个变量替换为相应的斯科伦函数后得到的布尔公式，在当前赋值下为真。由于每个斯科伦函数的大小都不超过 $p(|\Phi|)$，将斯科伦函数代入 Φ 并在给定特定赋值的情况下检查所得公式的真值，可以在多项式时间内完成。根据我们在第2节对 Σ_2^p 的定义，这意味着 QBF $\in \Sigma_2^p$。

□

这强化了Akshay等人的一项结果。[ACG+18] 他们在类似假设下得出结论：$\Sigma_2^p = \Pi_2^p =$ PH（借助Karp–Lipton定理）。

6 结论

在这篇批评中，我们指出了“Sopin的“PH =PSPACE” [Sop14] ”中的错误，并得出结论：Sopin的论文未能证明PSPACE与多项式层次相等。主要问题在于，Sopin的论文没有考虑执行其中一项检查可能需要的指数级工作量，而该论文声称这项检查可以在多项式时间内完成。此外，我们还发现，尝试中的证明对斯科伦化的使用并不清楚。事实上，如果一台机器只需使用多项式空间就能计算斯科伦化，那将令人震惊，因为这会带来突破性结果（正如我们在命题3中所证明的那样）。

致谢 我们感谢Erin Gibson、David E. Narv´aez和Lane A. Hemaspaandra对早期稿件提出的宝贵意见。对于任何遗留错误，作者概不推卸责任。

参考文献

[AB09] S. Arora 和 B. Barak。《复杂性理论：现代方法》。剑桥大学出版社，2009年。[ACG+18] S. Akshay、S. Chakraborty、S. Goel、S. Kulal 和 S. Shah。布尔函数综合的难点是什么? 载于《计算机辅助验证》，第251–269页。Springer 国际出版，2018年7月。[BGS75] T. Baker、J. Gill 和 R. Solovay。P=?NP问题的相对化。《SIAM计算杂志》，4(4):431–442，1975年。

[CL73] C. Chang 和 R. Lee。《符号逻辑与机械定理证明》。Academic Press，1973年。[Far20] A. Farago。如果 PH =PSPACE，会有什么后果？理论计算机科学 Stack Exchange，https://cstheory.stackexchange.com/questions/21191，2020年。[HO02] L. He maspaandra 和 M. Ogihara。《复杂性理论指南》。Springer-Verlag，2002年。[Lov78] D. Loveland。《自动定理证明：逻辑基础》。North-Holland，1978年。[Sip13] M. Sipser。《计算理论导论》。Cengage Learning，第3版，2013年。[Sop14] V. Sopin。PH = PSPACE。技术报告 arXiv:1411.0628v20 [cs.CC]，计算研究资料库，arXiv.org/corr/，2014年11月。修订于2022年11月。[Yao85] A. Yao。利用预言机分离多项式时间层次。载于第26届 IEEE 计算机科学基础研讨会论文集，第1–10页。IEEE计算机学会出版社，1985年10月。