arXiv:2304.14575v1 [cs.CC] 28 Apr 2023

对 Czerwinski 的“PSPACE 与 EXP 的分离”的评论*

Ian Clingerman 和 Quan Luu

罗切斯特大学计算机科学系，美
国纽约州罗切斯特市 14627

2023年4月28日

摘要

Czerwinski 的论文“PSPACE 与 EXP 的分离” [Cze21] 声称证明 PSPACE $\neq$ EXP，方法是证明不存在从给定的 EXP 完全集到给定的 PSPACE 完全集的长度递增多项式时间归约。然而，在这篇评论中，我们指出该论文的方法存在根本性缺陷，并为其中一个定理提供了一个反例，因此其所提出的 PSPACE $\neq$ EXP 证明并不充分。

1 引言

在这篇评论中，我们总结并分析了 Czerwinski 的《PSPACE 与 EXP 的分离》 [Cze21]。该论文声称通过提出一种新颖的证明技巧，并给出三个主要定理，证明了其核心主张 PSPACE $\neq$ EXP：定理 1 提出，任意图灵机的时间与空间复杂度的任何非平凡性质 P 都是不可判定的；定理 2 声称，如果集合 A 和 B 都是 EXP 完全集，那么它们可以相互进行长度递增的一对一归约；定理 3 则论证，存在一个 EXP 完全集 $\mathcal{E}_0$ 和一个 PSPACE 完全集 $\mathcal{P}_0$，使得 $\mathcal{E}_0$ 不能长度递增地一对一归约到 $\mathcal{P}_0$。如果 Czerwinski 的论文是正确的，那将是计算复杂性理论的一项重大进展。特别是，众所周知，PSPACE $\subseteq$ EXP。另一方面，PSPACE 是否等于 EXP，仍是理论计算机科学中最重要的开放问题之一。此外，如果 Czerwinski 的论文确实提出了一种用于分离 PSPACE 和 EXP 的新颖证明技巧，它可能成为建立其他新分离结果的基础。

Czerwinski 的论文证明 EXP 与 PSPACE 分离所采用的方法如下：利用定理 1 的推论 2 证明定理 3，然后根据定理 2 和定理 3，声称 PSPACE $\neq$ EXP。我们旨在指出这些定理各自存在的缺陷，并给出其中一个定理的反例，从而表明该论文未能证明其核心主张。

在第 2 节中，我们给出原论文和我们的评论中通篇使用的定义和记号。第 3、4、5、6 和 7 节分别概述并讨论 Czerwinski 论文中的一个章节。最后，我们在第 8 节总结我们的评论。

*部分由美国国家科学基金会资助，项目编号 CCF-2006496。

2 预备知识

在本节中，我们向读者介绍复杂性理论中的许多标准概念，以及 Czerwinski 论文中常用的记号。感兴趣的读者可以在大多数现代复杂性教科书中找到这些概念的等价定义 [HO02、AB09、Sip13]。

设 $\mathbb{N} = \{0, 1, 2, \ldots\}$，并且设 $\mathbb{N}^+ = \{1, 2, 3, \ldots\}$。给定一个字符串 w，令 $|w|$ 表示 w 的长度。我们用 ϵ 表示空字符串。对于任意图灵机（图灵机） M，令 $L(M)$ 表示由 M 接受的语言。

我们还引入了一些 Czerwinski 在全文中使用的记号。对于在输入 x 上运行的单带确定性图灵机 M，令 $s_M(x)$ 表示 M 在输入 x 上使用的带空间，并令 $t_M(x)$ 表示 M 在输入 x 上使用的时间。最后，性质 P 是图灵机的集合。当且仅当 $M \in$P 时，称图灵机 M 具有性质 P。关于函数t_M 和 s_M 的性质 P，是满足以下条件的图灵机集合：对于任意两个图灵机 M_1 和 M_2，如果对所有 $x \in \Sigma^*$，$t_{M_1}(x) = t_{M_2}(x)$ 且 $s_{M_1}(x) = s_{M_2}(x)$，那么 M_1 和 M_2 要么都属于 P，要么都不属于 P。当且仅当 P $\neq \emptyset$ 且存在一个图灵机 M 使得 $M \notin$P 时，性质 P 是非平凡的。

Czerwinski 的论文在若干论证中也用到了 S-m-n定理。由于要解释整个定理，就必须定义许多与本评论目的无关的可计算性理论概念，因此我们建议感兴趣的读者直接查阅 S.C. Kleene 的著作《元数学导论》 [Kle52]。

2.1 复杂性类

现在我们将定义本评论中会反复出现的复杂性类。

PSPACE 是能够由确定性图灵机使用多项式空间接受的语言类，形式化定义为 PSPACE $= \bigcup_{k\in\mathbb{N}^+}$ DSPACE$[n^k]$。

P 是能够由确定性图灵机在多项式时间内判定的语言类。形式化地说，P $= \bigcup_{k\in\mathbb{N}^+}$DTIME$[n^k]$。

E 和 EXP 都是能够由确定性图灵机在指数时间内接受的语言类。然而，它们的定义不同，代表两个不同的语言类。形式化地说，E $= \bigcup_{c\in\mathbb{N}^+}$ DTIME$[2^{cn}]$，EXP $= \bigcup_{c\in\mathbb{N}^+}$ DTIME$[2^{n^c}]$。E、EXP 和 PSPACE 之间存在以下关系：(1) E $\subsetneq$ EXP，(2) E $\neq$ PSPACE [Boo74], ，以及 (3) PSPACE $\subseteq$EXP。

预言机可以看作一个用于判定某个集合的隶属关系的子程序。预言机图灵机是一种经过改造的图灵机，具有查询预言机的额外能力。预言机的具体构造通常并不重要，可以将其视为黑箱。如果 A是一个预言机，且 M 是一个能够访问预言机 A的预言机图灵机，那么我们使用记号 M^A。

2.2 归约与完全性

我们将根据 Watanabe [Wat85] 提供的定义，定义若干种归约。其中许多定义可见于早期著作。若存在可计算函数 f，使得 $(\forall x)[x \in A \iff f(x) \in B]$.[1]，则称集合 A 多对一归约到 B，记作 $A \leq_m B$。这种归约称为

[1]Watanabe 使用 $\leq_m$ 表示多项式时间多对一归约；不过，我们沿用了标准记号，以区分多对一归约与多项式时间多对一归约。

若 f 可由确定性图灵机在多项式时间内计算，则称为多项式时间多对一归约；若 f 可由确定性图灵机在线性时间内计算，则称为线性时间多对一归约，分别记作 $A \leq_m^p B$ 和 $A \leq_m^{linear} B$ 。若 f 是一对一函数且可在多项式时间内计算，则称这种归约为多项式时间一对一归约，记作 $A \leq_1^p B$。若 A 可通过一个长度递增函数多项式时间一对一归约到 B ，则称集合 A 可长度递增多项式时间一对一归约到 B ，记作 $A \leq_1^{li} B$。若函数 f 满足 $(\forall x)[|f(x)| > |x|]$，则称其为长度递增函数。

当 B 属于 EXP，且对于所有 $A \in$ 属于 EXP， $A \leq_m^p B$时，语言 B 是 EXP 完全的。当 B 属于 E，且对于所有 $A \in$ 属于 E， $A \leq_m^{linear} B$时，语言 B 是 E 完全的。最后，当 B 属于 PSPACE，且对于所有 $A \in$ 属于 PSPACE， $A \leq_m^p B$时，语言 B 是 PSPACE 完全的。

3 关于 [Cze21]第 1 节

在 Czerwinski 论文的第一节中，定义了四个集合 [Cze21],

> $\mathcal{E} := \{(M, x, k) \mid k$ 以二进制编码，图灵机 M 在 k 步内接受 $x\}$,$\mathcal{P} := \{(M, x, 1^k) \mid$ 图灵机 M 接受 x，至多使用 k 个带格$\}$,$\mathcal{E}_0 := \{(M, k) \mid k$ 以二进制编码，图灵机 M 在 k 步内接受 $\epsilon\}$，以及$\mathcal{P}_0 := \{(M, 1^k) \mid$ 图灵机 M 接受 ϵ，至多使用 k 个带格$\}$,

最后两个分别被列为 EXP 完全集和 PSPACE 完全集的例子。还应指出，很容易证明， $\mathcal{E}$ 和 $\mathcal{P}$ 也分别是 EXP 完全集和 PSPACE 完全集。定义之后是推论 1

推论 1（[Cze21]）。 $\mathcal{E} \leq_1^p \mathcal{E}_0$ 和 $\mathcal{P} \leq_1^p \mathcal{P}_0$。

尽管这项推论是正确的，但后续证明中并未引用它。此外，也很难看出集合 $\mathcal{E}$ 和 $\mathcal{P}$的重要性，因为 $\mathcal{E}_0$ 和 $\mathcal{P}_0$ 已被介绍为各自复杂度类的完全集合。

顺带一提，作者引入了一个引理，其陈述是正确的。虽然给出的证明存在一些小错误，但由于下一节的讨论更为重要，我们不再赘述。

引理之后，作者给出了一条关于相对化障碍的说明。这条说明几乎没有阐明论文打算如何应对相对化障碍。因此，我们认为它唯一的作用是向读者保证，论文之后会在第 5 节中考虑相对化障碍可能带来的问题。

4 关于 [Cze21]第 2 节

Czerwinski 论文的第 2 节主要探讨时间与空间复杂度性质相关的不可判定性。首先，文中引入了两个记号 $s_M(x)$ 和 $t_M(x)$，我们已在评论的第 2 节中对其作出定义。随后，文中给出了关于 $s_M(x)$ 和 $t_M(x)$ 的两个不等式，这些不等式引自 Hopcroft、Paul 和 Valiant 的一篇论文 [HPV77]。对于任意单带确定性图灵机 M 以及任意输入 x ，若 M 在 x上停机，则有：

$$s_M(x) \leq t_M(x) \text{ and}$$
$$t_M(x) = s_M(x) \times 2^{O(s_M(x))}.$$

尽管如此，这一结果如何应用于本文提出的论证尚不清楚，因为似乎没有任何证明用到它。

接着，论文提出了它的第一个定理。

定理 1（[Cze21]）。对于任意图灵机 M，关于函数 s_M和 t_M 的任何非平凡性质 P 都是不可判定的。

该定理似乎是对著名的 Rice 定理 [Ric53]的改编。Rice 定理 [Ric53], 使用非平凡的语义性质，而该定理考虑的是函数 s_M 和 t_M的非平凡性质，它们分别是 M 在某个输入 x上的空间复杂度和时间复杂度。

我们给出定理 1 的一个反例。令 $\mathbf{P_1}(s_M, t_M) = \{M \mid M$为一台图灵机，且 $s_M(\epsilon) \geq 2$和 $t_M(\epsilon) \geq 2\}$，并且 $L_{st2} = \{M \mid M \in \mathbf{P_1}(s_M, t_M)\}$。容易看出，这个性质是非平凡的：存在一些满足 $\mathbf{P_1}$, 的图灵机，这意味着 $L_{st2} \neq \emptyset$；并且存在一台图灵机 M_{s1} ，它在某个输入 x_{s1} 上只使用一个带空间；还存在一台图灵机 M_{t1} ，它在某个输入 x_{t1}上运行一步后就停机，这意味着M_{s1}, M_{t1}不属于 $\mathbf{P_1}(s_M, t_M)$。我们将如下描述判定器 S，用于判定 L_{st2}。

算法 1 判定器 S 对 L_{st2}

```
判定器 S（M）：
    模拟 M 在 ϵ上运行
    m ← M的带字母表大小
    如果 M 在 ϵ 上运行第一步后停机，则拒绝
    否则，从第二步到第 m 步：
        如果 M 到达第二个带单元，则接受
        否则，如果 M 在 ϵ 停机，则拒绝
        结束 if
    结束 if
    模拟完第 m 步后，拒绝
判定器结束
```

之所以 S 在运行到第 m 步后能够接受，是因为如果 M 只使用第一个带单元，且有 m 种不同的带字符，那么空间配置就只有 m 种。因此，如果 M 仅使用第一个带单元运行超过 m 步，我们就可以确定 M 会在该单元内无限循环。除此之外，应该很容易看出 S总会停机，并且 $L(S) = L_{st2}$。

定理 1 的证明存在一个错误假设：对于任意图灵机 M 以及函数 s_M和 t_M的任意非平凡性质 P，并且对于任意输入 x，检查性质 P 都需要知道 $s_M(x)$ 和 $t_M(x)$ 的精确值。然而，在我们构造的反例中，判断图灵机 M 是否属于 L_{st2}，既不需要计算精确的空间复杂度，也不需要计算精确的时间复杂度。

随后，论文利用定理 1 的先前结果，并将其推广为推论 2。

推论 2（[Cze21]）。设 $f\colon \mathbb{N} \to \mathbb{N}$ 是一个满足以下条件的函数：

$$\lim_{|x|\to\infty} \frac{f(|x|)}{t_M(x)} = 0$$

and

$$\lim_{|x|\to\infty} \frac{f(|x|)}{\log(t_M(x))} > 0.$$

判断（$\forall x$）[s_M（x） $\leq f$（$|x|$）].是否可判定

遗憾的是，既然我们已经证明定理 1 存在缺陷，那么推论 2 的证明便不完整，因此其正确性无法得到保证。

W我们继续来看本节下一部分也是最后一部分的结论：推论 3 .

推论 3（[Cze21]）。若要证明任意图灵机 M 在 t 步内不接受输入 x，最坏情况下需要 Ω(t) 的计算开销。

阅读该推论的证明时，我们注意到证明并未讨论推论 3 的正确性。相反，该证明试图为任意图灵机 M 的函数 t_M 构造另一种非平凡性质，并依据定理 1 得出该性质不可判定。还应当指出，推论 3 实际上是正确的。简而言之，检查图灵机 M 是否在 t 步内接受输入 x 可能需要 Ω(t) 的运行时间，原因在于模拟 M 在 x 上运行所产生的开销，可能会使总运行时间超过 t。

5 关于 [Cze21]第 3 节

现在我们将评论 Czerwinski 论文的第 3 节。本节介绍了与长度递增函数和归约相关的研究结果。

Czerwinski 的论文对长度递增函数的定义与我们评论的第 2 节类似，但额外要求其具有多项式时间可计算的逆函数。

论文接着介绍了定理 2，该定理改编自 Watanabe 论文《单射多项式时间等价关系》 [Wat85]中的一项结果。

定理 2（[Cze21]）。如果 A 和 B 是 EXP 完全的，则 $A \leq_1^{li} B$ 和 $B \leq_1^{li} A$ 成立。

作为该定理的证明，Czerwinski 引用了 Watanabe 论文 [Wat85]中推论 3.4 的证明。需要注意的是，在 Watanabe 的论文发表时，通常将 EXP 定义为我们所定义的 E。因此，Watanabe 的论文只证明了 A 和 B 是 E 完全的，而不是 EXP 完全的。Czerwinski 的论文没有提供预备知识部分，因此无法确定该论文意图使用 E 还是 EXP。不过，我们认为其主要目标是分离 EXP 和 PSPACE，因为结果 E $\neq$ PSPACE 已是众所周知的 [Boo74]。此外，Watanabe [Wat85] 并未将长度递增函数定义为必然是多项式时间可逆的，这与 Czerwinski 论文中的定义不同。基于这些原因，定理 2 的证明是无效的。这是 Czerwinksi 论文中的一个严重缺陷，因为定理 2 的结果为定理 3 展示 PSPACE $\neq$ EXP 奠定了基础。我们将在第 6 节中进一步讨论这一点。

接下来，我们将评论引理 2。

引理 2（[Cze21]）。设 $A, B \subseteq \{(M,k) \mid$图灵机 M接受 $k\}$和 $A \leq_1^{li} B$。那么存在一个长度递增归约[2] g，使得：

$$(M,k) \in A \iff g(M,k) \in B, \text{ and}$$

$$g(M,k) = (M_g, k_g)$$

其中，M_g 仅取决于 M。

在开始证明之前，我们想先介绍作者使用的一些记号。函数 g_1 和 g_2 定义为 $g_1(M,k) = M_g$ 和 $g_2(M,k) = k_g$，因此$(M_g, k_g) = g(M,k) = (g_1(M,k), g_2(M,k))$。根据S-m-n定理，$g_1(M,k) = g_{1,S(M)}(k)$，且 $g_2(M,k) = g_{2,S(M)}(k)$。此外，根据Czerwinski对长度递增函数的定义，g 可在多项式时间内求逆，因此$(M,k) = (g_{1,S(M)}^{-1}(k_g), g_{2,S(M)}^{-1}(k_g))$。

我们想重点关注证明末尾给出的伪代码。

算法 2

```
def M_g(k_g):
    k̂ ← g_{2,S(M)}^{-1}(k_g)
    if k̂ 定义后
        返回 g_{1,S(M)}(k̂)(k_g)
    结束 if
```

该伪代码作为一种算法给出，其中可以构造 M_g 并在不使用 k 的情况下对其进行模拟，从而使其仅依赖于 M。据我们理解，给定 k_g 作为某个输入，该算法将构造 k；如果其有定义，则满足 $(M,k) \in A$。之后，可以通过 $g_{1,S(M)}(k)$ 构造 M_g^。然而，这一构造并不能证明该引理的表述。引理声称 M_g 仅依赖于 M，而所提供的伪代码在计算中还使用了 k_g。

6 关于 [Cze21]的第4节

主要结果发表于 Czerwinski 论文的第4节。

定理 3（[Cze21]）。不存在从 $\mathcal{E}_0$ 到 $\mathcal{P}_0$的 $\leq_1^{li}$-归约。

必须明确一点：要使该定理能够推出 EXP $\neq$ PSPACE，定理 2 就必须成立。定理 2 声称，所有 EXP 完全集都可以通过长度递增多项式时间一对一归约彼此归约。假设定理 2 确实成立，那么要证明 $\mathcal{P}_0$ 不是 EXP 完全的，从而证明 EXP $\neq$ PSPACE（这正是定理 3 的陈述），只需证明不存在从某个 EXP 完全集 $\mathcal{E}_0$ 到某个 PSPACE 完全集 $\mathcal{P}_0$ 的长度递增多项式时间一对一归约。正如我们已经指出的，定理 2 存在问题

[2]在 Czerwinski 的论文中，g 最初被表述为长度递增函数。然而，如果 g 只是长度递增函数，那么后续的当且仅当条件就有缺陷，因为它未能正确处理语法上不符合 $(\cdot,\cdot)$ 形式的字符串。因此，我们推测作者本意是让 g 成为长度递增归约。由于这不会对评论产生重大影响，我们将此作为脚注处理。

因此，如果定理 3 成立，它也不足以推出 PSPACE $\neq$ EXP。此外，该定理的证明还用到了引理 2 和推论 2，而我们已经证明它们是无效的。因此，我们可以说，这个证明同样不完整，归根结底，论文未能证明 PSPACE $\neq$EXP。

7 关于 [Cze21]第 5 节

在本节中，Czerwinski 试图论证该证明技巧并未违反相对化障碍。为此，他引入了命题 1。

命题 1（[Cze21]）。存在一个可计算函数 f，使得（$\forall(M,x,k)$)[$(M,x,k) \in \mathcal{E} \iff f((M,x,k)) \in \mathcal{P}$]，且具有指数计算时间。

作为该命题的证明，论文给出了伪代码：它从 $\mathcal{E}$ 中取出一个问题，并在 $\mathcal{P}$ 中计算一个集合，所需时间为指数时间。该伪代码似乎是正确的，然而命题本身是平凡的。目前尚不清楚该命题如何说明所谓的证明技巧并未违反相对化障碍。

论文声称，这个函数 f 是一个可计算但无法在多项式时间内计算的多对一归约，“意味着 $\mathcal{E} \leq_m \mathcal{P}$，但 $\mathcal{E} \not\leq_m^p \mathcal{P}$，因此 $\text{PSPACE}^{\text{P}} \neq \text{EXP}^{\text{P}}$” [Cze21]。在此之前，论文并未直接声称 $\mathcal{E} \not\leq_m^p \mathcal{P}$，因此尚不清楚指数时间归约的存在如何阻止多项式时间归约的存在。事实上，多项式时间多对一归约的存在将意味着指数时间多对一归约的存在。

“$\mathcal{E} \not\leq_m^p \mathcal{P}$ 意味着 $\text{PSPACE}^{\text{P}} \neq \text{EXP}^{\text{P}}$ ”这一论断是正确的，但尚不清楚论文为何使用带有 P 预言机的复杂度类，而不是直接使用这些复杂度类，因为 $\mathcal{E} \not\leq_m^p \mathcal{P}$ 更直接地意味着 PSPACE $\neq$ EXP，进而意味着 $\text{PSPACE}^{\text{P}} \neq \text{EXP}^{\text{P}}$，这是因为 $\text{PSPACE}^{\text{P}} = \text{PSPACE}$ 且 $\text{EXP}^{\text{P}} =$EXP。

8 结论

在这篇评论中，我们指出了“PSPACE 与 EXP 的分离” [Cze21] 中的错误，并得出结论：该论文未能证明 PSPACE $\neq$ EXP。论文的推理中存在若干错误，因此无法为其关键论断提供充分的证明。因此，PSPACE $\neq$ EXP 是否成立仍是一个未解决问题。

致谢 我们特别感谢 Michael C. Chavrimootoo 在本评论撰写过程中给予的帮助，以及他对早期稿件提出的宝贵意见。我们也感谢 Ethan Ferland、Lane A. Hemaspaandra 和 David E. Narv´aez 对早期稿件提出的宝贵意见。对于仍然存在的任何错误，作者承担全部责任。

参考文献

[AB09] S. Arora 和 B. Barak。《复杂性理论：一种现代方法》。剑桥大学出版社，2009年。

[Boo74] R. Book。比较复杂性类。《计算机与系统科学杂志》，3(9):213–229，1974年。[Cze21] R. Czerwinski。PSPACE 与 EXP 的分离。技术报告 arXiv:2104.14316 [cs.CC]，计算研究文库，arXiv.org/corr/，2021年4月。版本 1。[HO02] L. Hemaspaandra 和 M. Ogihara。《复杂性理论指南》。施普林格出版社，2002年。[HPV77] J. Hopcroft、W. Paul 和 L. Valiant。时间与空间之比较。《ACM会刊》，24(2):332–337，1977年4月。[Kle52] S. Kleene。《元数学导论》。D. van Nostrand公司，1952年。[Ric53] H. Rice。递归可枚举集的类及其判定问题。《美国数学会会刊》，74(2):358–366，1953年。[Sip13] M. Sipser。《计算理论导论》。圣智学习，第3版，2013年。[Wat85] O. Watanabe。关于单射多项式时间等价关系。《理论计算机科学》，38:157–165，1985年。