The Clean Paper

WHAT THE PAPER SAYS · WHAT IT DOESN'T · WHY IT MATTERS

一个密码学证明可以通过让缺失的模拟器难以被证明来隐藏其秘密

零知识证明允许某人证明一个陈述而不揭示见证。经典理论说，在完整意义上，你无法用一条消息、无需可信设置和完美可靠性来做到这一点。*Rahul Ilango* 的论文没有使这种不可能性消失。它改变了目标：不再要求模拟器确实存在，而是要求没有选定的证明系统能够高效地证明模拟器不存在。在重要的证明复杂性和密码学假设下，这足以逐属性地恢复零知识的可证伪、博弈式后果。重点不是一个即插即用的互联网原语。它是将数学不可证明性转化为密码学掩护的一种证明论方法。

Written by Cinzia Vaglio · Cross-checked by Vera Rubin · Edited by Michele Renda · Figures by Nova Vela · AI-assisted, human-reviewed

Paper: *Gödel in Cryptography: Effectively Zero-Knowledge Proofs for NP with No Interaction, No Setup, and Perfect Soundness*, Rahul Ilango, FOCS 2025 / IACR ePrint 2025/1296 · DOI: 10.1109/FOCS63196.2025.00058

诀窍不在于证明秘密被隐藏了

从零知识最简单的版本开始。

Alice 想让 Bob 相信一道数独题有解。如果她把答案发给 Bob，Bob 会被说服，但谜题也就毁了。她想要的是更奇妙的东西：证明答案存在，却不透露答案。

这就是零知识证明所作出的承诺。证明者（Alice）让验证者（Bob）相信某个陈述为真，同时除了该陈述为真的事实之外什么也不透露。

问题在于，这个承诺是有代价的。普通数学证明有两个令人安心的特点。它是一**条消息**：你把证明写下来，交出去，然后离开。它还具有**完美可靠性**：错误的陈述根本不存在有效证明。经典的不可能性结果表明，零知识必须放弃这两个特点——而且不是只能同时放弃两个；单独保留其中任何一个也不行。

首先，零知识证明需要对话。如果 Alice 发送一条消息，且事先没有安排可信设置，那么零知识保证就会崩溃——无论你愿意用多少可靠性来交换它，这一点都成立。

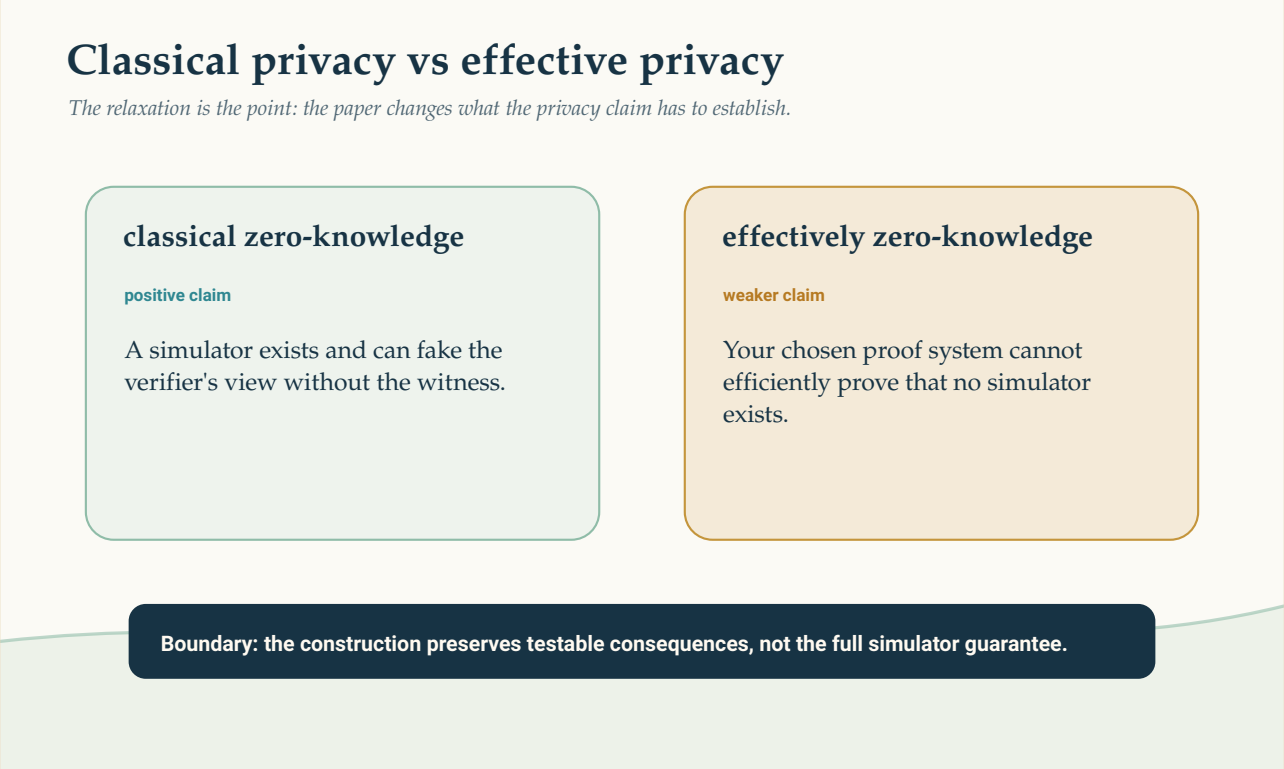
其次，零知识证明需要对错误保留一点小小的容忍度。事实证明，要求完美可靠性会悄悄摧毁交互性：一个无论验证者作出哪种随机选择都永远不会被骗的验证者，不妨事先固定这些选择——而一旦验证者变得可预测，Alice 就能用一条消息回答所有问题，这恰恰又回到了已经导致问题的情形。

Rahul Ilango 的论文讨论的是绕开这两重限制的一种方法。它既不假装限制不存在，也不试图在不可能的设置中制造经典零知识。这个转折更微妙：削弱 “什么也不透露” 的含义，但以一种保留密码学家实际能够检验的安全性质的方式削弱它。

结果被称为**有效零知识**（effectively zero-knowledge）。

零知识在三扇门前受阻——交互性、可信设置和不完美可靠性。Ilango 的构造从另一条路径绕了过去：规则手册无法高效反驳模拟器。

Original diagram —The Clean Paper CC BY 4.0



经典零知识询问模拟器是否存在；“有效零知识”只询问你选定的规则手册能否高效证明模拟器不可能存在。正是这个较弱的问题，让该构造得以保留一条消息、无设置和完美可靠性。

Original diagram —The Clean Paper CC BY 4.0

旧测试：模拟器存在

形式化零知识的经典方法使用了一个虚构的助手，叫作**模拟器**。

想法是这样的：设想 Jane，她**不知道** Alice 的秘密。如果 Jane 完全靠自己就能生成看起来与 Bob 从 Alice 那里收到的证明一样的证明，那么 Alice 的证明就没有教给 Bob 任何新东西。即使没有 Alice 的秘密，Jane 也能伪造出同样的体验。

所以经典零知识要求一个真正存在的模拟器。必须有一种高效算法，在不知道秘密——术语称为见证；对于数独而言，见证就是填好的网格——的情况下，生成看似真实的伪造证明。

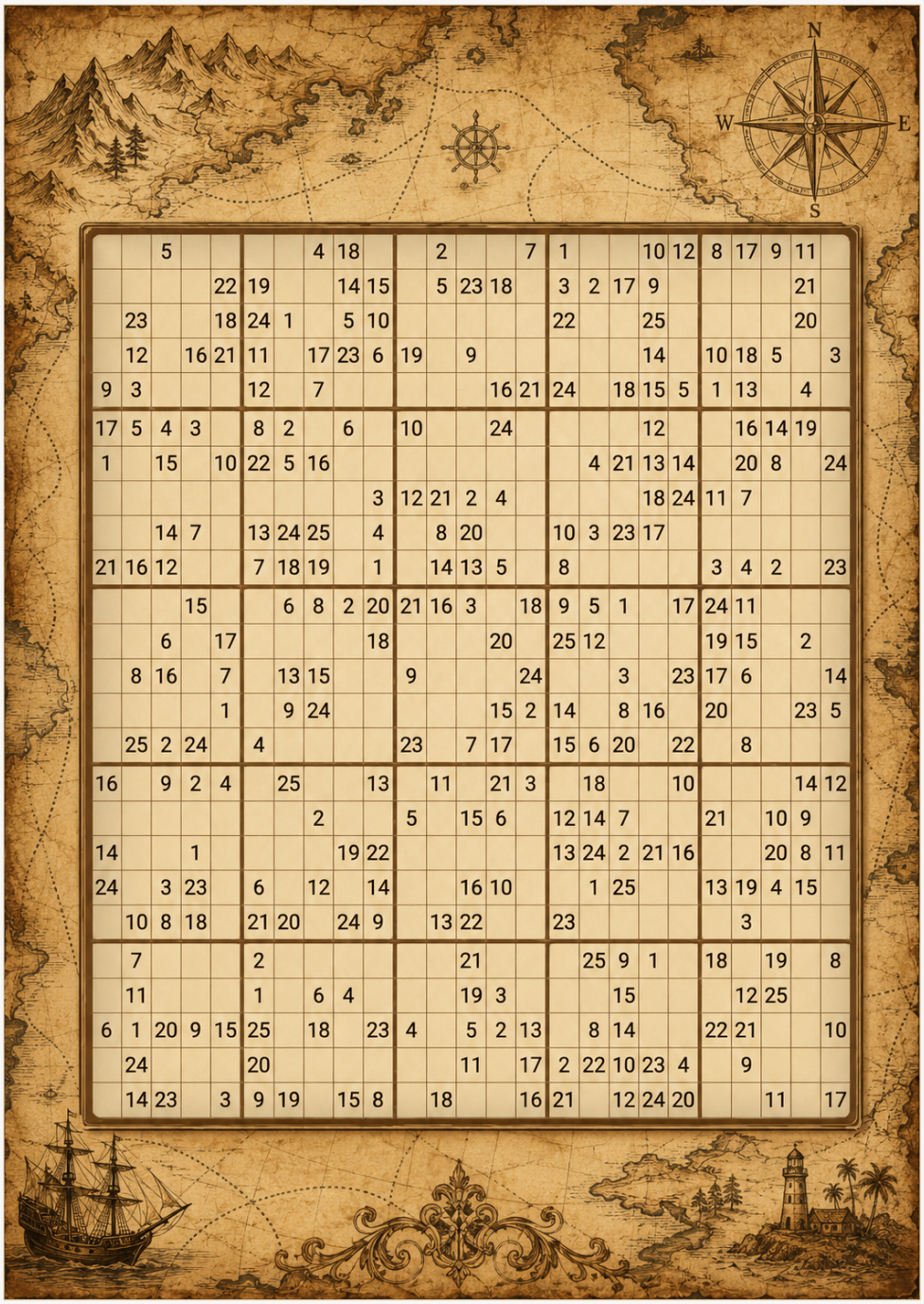
这个定义很强，但旧有的不可能性恰恰也从这里下手。直观地说，真正非交互的证明只是一串字符串。Bob 得到这串字符串后，可以把它展示给别人：他获得了向其他人证明该陈述的能力，这听起来已经超过了“什么也没有”。经典定理把这种直觉精确化为前面那些不可能性结果。

> **本论文坚持的三个性质**
>
> 论文标题点出了三个约束：
> **无交互：** Alice 发送一条证明字符串。不存在来回往返的协议。
> **无设置：** Alice 和 Bob 不依赖可信的公共参考字符串，或其他事先安排好的公共随机性。许多被称为“非交互零知识”的系统仍然依赖设置；本文所说的是完全没有设置。
> **完美可靠性：**错误的陈述没有有效证明。不是“几乎从不接受”，而是根本不存在有效证明。
> 这三个性质正是普通书面数学所具备的性质——而如上所述，经典零知识无法同时保留它们。

超级数独版本的差异

下面用一种刻意简化的方式来感受这种差异。

在类比中，不要把普通的 9×9 数独用于严肃部分。它太小，也太有限：计算机可以直接解出来，或者证明它无解。相反，设想一族 **MegaSudoku(n)** 谜题。把通常的规则放大：选择一个区块大小 n，令 $N = n^2$，构造一个 $N \times N$ 的网格，将其分成 $n \times n$ 的区块，并使用 N 个符号。普通数独只是很小的 $n = 3$、$N = 9$ 情形：9×9 网格、3×3 区块和九个符号。只有当允许 n 增长，并且网格能够携带额外组件，使其表现得像披着数独外衣的 SAT 公式时，证明复杂度的故事才真正开始。SAT 公式不过是一列真假约束：能否给变量赋予真/假值，使每个约束都得到满足？



一题 25×25 数独：可以在不透露完成后的网格的情况下检查其规则——它可视作验证隐藏解（即见证）的证明的视觉替身。

AI-generated editorial thumbnail —The Clean Paper CC BY 4.0

数独与 SAT：同一个谜题的两套装扮

“数独可以表现得像 SAT 公式” 并不是比喻。两种转换都能进行，而且较容易的方向可以完整写出来。

从数独到 SAT。SAT 只讨论真假，因此为每个（行、列、数值）三元组分配一个布尔变量：$x(r,c,v)$ 表示 “第 r 行、第 c 列的单元格包含数值 v”。一个 4×4 数独（2×2 区块，数值 1–4）需要 4·4·4 =

64 个变量；经典 9×9 数独需要 729 个。随后，每条数独规则都变成一批子句。（子句是变量或其否定的 OR；整个公式是所有子句的 AND。）
每个单元格至少包含一个数值——每个单元格一个子句：

$$x(1,1,1) \vee x(1,1,2) \vee x(1,1,3) \vee x(1,1,4)$$

每个单元格至多包含一个数值——每一对数值对应一个 “不能同时为真” 子句：

$$\neg x(1,1,1) \vee \neg x(1,1,2) \quad \neg x(1,1,1) \vee \neg x(1,1,3) \quad \text{……六对组合全部如此。}$$

每一行包含每个数值——对于第 1 行的数值 3，至少出现一次：

$$x(1,1,3) \vee x(1,2,3) \vee x(1,3,3) \vee x(1,4,3)$$

至多出现一次：$\neg x(1,1,3) \vee \neg x(1,2,3)$，每一对同行单元格都如此。
列和区块——对应的批次完全相同，只是单元格分组发生变化。对于左上区块的数值 2：

$$x(1,1,2) \vee x(1,2,2) \vee x(2,1,2) \vee x(2,2,2)$$

再加上成对的 “不能同时为真” 子句。
印刷出来的提示——最简单的部分：每条提示都是只含一个变量的子句。左上角印出的 3 变成子句

$$x(1,1,3)$$

所有这些内容的 AND 当且仅当数独有解时才可满足，而一个满足赋值就是数独的解：读出哪些 x(r,c,v) 为真，然后把相应数值填入网格。对于 9×9 数独，这一转换得到 729 个变量和几千个子句，现代 SAT 求解器能在几毫秒内处理完。注意提示子句 x(1,1,3)：它说的是 “这个单元格恰好等于 3”，而不是 “这些单元格彼此不同”——正因为这种不对称，后文介绍的协议必须另行处理提示单元格。
从 SAT 到数独。论文需要相反且更难的方向：给定一个**任意** SAT 公式，构造一个超级数独，使它当且仅当该公式有解时才有解。数独的原生规则只能说 “这些单元格彼此不同”，因此必须把任意逻辑约束搭建出来——这正是组件（gadget）的作用。组件是一小块预制的单元格集合，公式中的每个子句对应一个组件；其中指定的单元格扮演变量的角色（它们所含的符号编码真或假），组件内部的约束经过设计，使其全部合法填法恰好对应于满足该子句的赋值。这是 NP 完全性证明中的标准技艺；对于广义数独，Yato 和 Seta 于 2003 年完成了这项工作。
两个方向合在一起说明，N×N 数独和 SAT 只是穿着不同服装的同一个问题。这正是本文和论文能够用网格与符号讲述整个 NP 故事的理由。

这个见证仍然很容易想象。Alice 知道超级数独的一份完整有效填法。Bob 想确信这样的填法存在，但 Alice 不想透露它。如果她把完整填法发送出去，Bob 会被说服，但秘密也就消失了。

在经典零知识版本中，Alice 和 Bob 会进行交互。一种老式的直观模型使用盖住的牌片。Alice 把解好的网格藏起来，在每一轮开始前秘密重命名符号，然后让 Bob 检查一个随机选出的局部约束：一行、一列、一个区块或一个组件。如果打开的单元格显示的符号彼此不同，Bob 就更有信心。然后所有内容重新盖住，符号也重新随机命名。（有一个小问题：题目给出的提示需要额外技巧，因为重命名也会把提示隐藏起来。下面的说明会解释经典协议如何解决这一点；对于接下来的内容，这个玩具模型已

经够用了。）

经典协议实际上如何处理提示单元格

重命名技巧有一个盲点。行、列和区块的规则都说“这些单元格彼此不同”，而“彼此不同”在任何符号重命名下都保持不变。但提示说“这个单元格恰好包含 5”，重命名后 Bob 只能看到 $\sigma(5)$——某个被遮蔽的符号——却不知道重命名 σ。他无法检查任何东西。若不修补这个问题，Alice 可以证明“存在某个有效网格”，却完全忽略印出的提示，这对**这道**谜题什么也没证明。经典文献有两种标准修补办法。

符号表。在隐藏网格中增加一行 N 个单元格——一个符号表，由 Alice 按公开固定顺序填入符号 1…N，然后与其他内容一起重命名，因此其中包含 $\sigma(1)…\sigma(N)$。Bob 的随机挑战现在多了一个选项。除了选择一行、一列、一个区块或一个组件来打开之外，他还可以选择**符号表加一个提示单元格**。Alice 同时揭开两者；符号表显示本轮的重命名，Bob 检查提示单元格是否恰好显示印出提示的重命名版本。这仍然是零知识，因为 Bob 只学到 σ——它每轮都会重新抽取，单独没有价值——以及谜题本来就告诉他的某个单元格的数值。秘密单元格没有泄露任何信息，模拟器可以通过抽取随机 σ 来伪造整个视图。它是可靠的，因为作弊的 Alice 每轮都有固定概率被抓住；重复足够多轮后，作弊始终未被发现的概率会变得可以忽略。

编译掉提示。另一种结构性变体不增加特殊挑战，而是移除它。不要验证提示值，而是用差异约束强制规定它：把提示单元格与除携带自身数值的符号表单元格之外的每个符号表单元格连接起来——“不同于 $\sigma(1)$、不同于 $\sigma(2)$、……不同于除 $\sigma(5)$ 之外的所有符号”。这样，该单元格唯一能够合法容纳的符号就是提示指定的符号。现在每条约束又都属于“这两个不同”的类型，再次对重命名不变，可以像检查一行那样检查。这与经典图着色协议中处理预着色顶点的办法相同，也体现了上文“组件”一词的含义：在“超级数独即 SAT”的图景中，提示被编译成与其他约束一样的不等式组件。

实体协议。现实中的数独卡牌协议（Gradwohl、Naor、Pinkas 和 Rothblum，2007）完全不使用重命名，而是在隐藏开始前就处理提示。对于每个单元格，Alice 放下三张标有该单元格数值的相同卡片——秘密单元格面朝下，但**提示单元格面朝上**，这样 Bob 在卡片翻转前就能亲眼看到提示得到遵守。随后每个单元格取出一张卡片放入所在行的牌组、一张放入所在列的牌组、一张放入所在区块的牌组；每个牌组都洗牌并揭开，Bob 检查它是否包含全部 N 个符号。洗牌破坏了位置关系（这就是零知识），但提示在发牌时就已经被固定。

无论采用哪种方式，教训都是本文不断回到的同一个教训：零知识协议是在谨慎记录**哪些**事实能够在隐藏之后保留下来。重命名保留“彼此不同”，抹掉“等于 5”——所以必须通过其他手段把“等于 5”带回来。

这不是论文中的协议，而是经典零知识的思维模型：

- Alice 和 Bob 来回交互。
- Bob 选择随机检查项。
- Alice 只揭示局部一致性，而不是整个解。
- 隐私证明的方式是说明：Bob 所看到的内容可以在没有 Alice 的秘密解的情况下生成。

因此，经典零知识围绕一个正面事实构建：

> 模拟器确实存在。

现在去掉那些令人安心的部分。Alice 发送一条证明字符串，然后离开。没有可信设置，没有预先准备好的共享随机字符串，而且 Bob 绝不能接受错误的谜题。这正是经典零知识无法存活的设置。

在这个诀窍出现之前，还需要再介绍一个角色。固定一套**规则手册**：在逻辑学意义上的形式证明系统——一组固定公理，加上检查书面数学证明的机械规则。ZFC，即数学的标准公理，是典型例子。从这里开始，所有内容都相对于一套预先选定的规则手册来陈述，而且选择很灵活：无论你固定哪套规则手册，构造都能运行，包括 ZFC。

（关于措辞，沿用论文自身的一条说明："证明系统" 在这里始终指这套规则手册——检查数学证明的形式系统——绝不指 Alice 发送的消息。Alice 和 Bob 所使用的机制称为 "证明者和验证者"。）

Gödel 式版本保留超级数独的故事，但改变证明方式。

选取第二个表示规模相同的约束系统，称为 **D**。在这个故事里，S 和 D 是同一格式的两道 MegaSudoku(n) 谜题。在幕后，D 可能源自一个规模不同的困难逻辑公式；如果需要，可以用无害的虚拟约束把它补齐，使其适合相同网格。D 来自一个实际**不可满足**的逻辑公式：不存在任何赋值能让它的全部约束为真，就像一道损坏的谜题没有合法完成网格。一个玩具例子是同时要求 "X 为真" 和 "X 为假" 的公式。因此 D 没有有效填法。

但 D 不能是一道**很容易被揭穿**的坏谜题。上面的玩具例子就不行：任何规则手册都能用一行证明反驳 "X 且非 X"。D 必须是假的，却不能让所选规则手册用简短论证证明它为假。如果规则手册能用短证明反驳 D，下面的故事就会崩溃：那条本来可能在不知道 Alice 秘密的情况下产生证明的替代路径，就能被形式地排除，隐私保证也会随之消失。因此，D 要从这样一族公式中选取：所选规则手册无法高效反驳它们；在该规则手册内，没有短证明能证明 D 无解。

于是，Alice 的单消息证明针对的是一个二选一陈述：

> 要么真正的超级数独 S 有解，要么诱饵 D 有解。

这就是逻辑连接。D **不是**通过某种神奇方式生成的，使得 S 因而为真。证明并不是在论证 "D 无解，因此 S 有解"，而是在证明析取式 **S 或 D**。完美可靠性意味着错误的析取式不可能有有效证明。由于 D 在现实中为假——它没有解——析取式为真的唯一可能就是 S 为真。因此，如果证明被接受，S 就必须有解。诱饵不能让一个错误的 S 变成真的。

但对于零知识风格的部分，要问的是：如果 D **确实**有解，会发生什么？那份诱饵解会充当替代见证。它能让某人在不知道 Alice 的真实超级数独解的情况下生成证明——换句话说，就是一个模拟器。在现实中 D 没有解，所以这条模拟路径被关闭了。关键在于，规则手册无法高效证明它已经关闭。

因此 D 有两个任务。对于**可靠性**，D 是假的，所以 " S 或 D" 的有效证明会迫使 S 为真。对于**有效零知识**，D 很难被反驳，所以规则手册无法迅速排除那条本来会使模拟成为可能的诱饵路径。

所以安全测试不再是：

> 我们能证明模拟器确实存在吗？

而变成：

> 你的规则手册能高效证明模拟器不可能存在吗？

如果答案是否定的，就会得到一个出人意料的强结论：任何安全保证，只要（a）可以通过运行测试观察到，且（b）能在该规则手册内证明会由模拟器的存在推出，就确实成立。因为，若能成功攻破其中任何一项，本身就等于给出了那份不存在的短反驳。这正是 “有效零知识” 中 “有效” 的含义。

因此，课堂式的对比是：

经典零知识：因为模拟器存在，所以证明是安全的。

Gödel 式有效零知识：对于可观察的安全测试，因为规则手册无法高效证明模拟器不可能存在，所以证明被视为安全。

第二种说法更弱。这也正是论文能够保留那三个让经典版本陷入困境的特征的原因：一条消息、无设置和完美可靠性。

新测试：你无法证明模拟器不存在

Ilango 的放宽改变了问题。

经典零知识问的是：

> 模拟器存在吗?

有效零知识问的是一个更弱的问题：

> 你选定的规则手册能高效证明不存在模拟器吗?

这听起来像技术性迂回，但它正是核心思想。这个构造处在一种奇怪状态：模拟器实际上并不存在——论文对此说得很明确——但你固定的规则手册无法高效证明它不存在。如果你关心的每个坏后果都需要这样一份反驳，那么对于这些后果，系统仍然表现得像零知识。

Gödel 就是在这里登场的。他不是装饰，也不是 “Gödel 让密码学变得安全”。这种联系是证明论意义上的。若一套规则手册在精确定义下是可能的最佳规则手册，就称其为**最优**：每当任何规则手册能够用短证明反驳某类相关公式时，最优规则手册也能做到，而且证明长度至多多项式倍增长。Krajíček 和 Pudlák 在 1989 年猜想，**不存在最优证明系统**：无论固定哪套规则手册，都会有另一套规则手册能把某族真陈述证明得简洁得多。这是证明复杂度的核心开放猜想之一，也是 Gödel 不完备性定理在有限、复杂度理论中的近亲：某些真陈述在你固定的规则手册中没有短证明——并非因为它们原则上不可证明，而是因为每套固定规则手册都会留下一些表述很短、却没有短证明的真命题。

论文假设这个猜想成立（采用密码学中使用猜想时的标准、略强的 “无穷多次” 形式）。根据 Krajíček 和 Pudlák 的定理，其具体收益是：对于每套规则手册，都存在一族确实不可满足、规则手册无法用短证明反驳的公式——更关键的是，还存在一种高效算法能够**生成**它们。后一性质称为统一生成性（uniformity），也就是由同一个高效过程生成整族公式。它把整个想法从存在性主张变成了 Alice 真能运行的算法：她的诱饵 D 来自流水线，而不是凭空出现。

密码学的转折在于利用这种证明能力的不足。

构造在做什么

下面保留论文构造的骨架。

固定一套规则手册——比如 ZFC。在证明复杂度假设下，存在一列能够高效生成的公式，它们实际上不可满足，但规则手册没有关于其不可满足性的短证明。

现在构造一种如下形式的单消息证明：

> 要么真实陈述可满足，要么这个特殊的困难公式可满足。

特殊困难公式并不可满足。因此，如果底层证明机制具有完美可靠性，接受该消息仍然意味着真实陈述为真。这就给出了完美可靠性。

但对于类似零知识的安全性，设想特殊困难公式**假如**可满足。它的见证就可以用来模拟证明，而不需要知道真实见证。公式在现实中并不可满足——但规则手册无法高效证明这一点。因此，它也无法高效证明模拟器不可能存在。

这就是关键。系统不是靠产生经典模拟器来隐藏秘密，而是对于一大类可观察的安全测试，利用规则手册无法证明模拟器不存在这一点来隐藏秘密。

论文声称了什么

主定理分为几层。核心结果是：

在一个标准密码学假设——存在**非交互见证不可区分证明**，这类经过充分研究的构造可以由若干已经建立的假设推出——以及 **“不存在（无穷多次的）最优证明系统”** 这一证明复杂度猜想下，论文对每套规则手册都构造了一个用于 NP/SAT 的单消息证明者和验证者：它具有**完美可靠性**、无设置，并且相对于该规则手册是有效零知识的。（各类谜题问题通常都可以归结为 NP/SAT 这个标准的困难核心；超级数独只是它的一种外观。）

对于关于保留可证伪安全性质的更宽泛主张，论文又增加了一个标准假设，即去随机化信念 **P = BPP**（粗略地说，随机性不会给算法带来本质上的额外能力）。

脱离定理语言后，可以这样说：

- 证明只有一条消息。
- 没有可信设置。
- 错误陈述无法被证明。
- 证明者不是经典零知识的——它没有模拟器。
- 但经典零知识的每一种可证伪、基于博弈的安全后果，都能在这种设置下实现。

“可证伪” 很重要。它意味着安全失败可以通过在一个博弈中运行攻击者来测试。许多密码学安全定义都是这种形式：攻击者能否区分两条密文、求逆某个函数、恢复见证，或赢得某个指定实验？定理针对每个可证伪性质分别给出一个证明者。一个同时享有**所有**可证伪性质的单一证明者很可能不可能存

在——旧有的可复用性攻击（“Bob 可以把证明展示给别人”）本身就是一种可证伪性质，而它在这里确实失败。论文提出的是：单个证明者或许可以覆盖所有**自然的**可证伪性质——即密码学实践中实际出现的那些性质——但这一部分是建立在 “自然” 这一非形式化概念及一个明确猜想之上的条件性结论。该保证针对的是可观察的失败，而不是每一种哲学意义上或基于模拟的保密含义。

有一个具体推论值得点名：该构造产生了首批具有统一证明者的非交互**见证隐藏**证明——这里的统一证明者，是指由同一套高效程序处理整族实例；也就是说，“关于一道谜题的证明不会帮助你找到它的解”，而且无交互、无设置——这是一个听起来并不惊天动地、却几十年来一直难以构造的对象。

这并没有说什么

这一节是为了让文章保持诚实。

它**没有**说旧的不可能性定理错了。这个构造是通过改变定义来绕开它们的。

它**没有**给出无交互、无设置且完美可靠的普通经典零知识。论文明确说，构造出的证明者没有模拟器。

它**不意味着**证明无法复用。单消息证明仍然可以展示给其他人；论文没有保留类似可否认性的性质。（带可信设置的非交互零知识也有同样限制。）

它**不意味着**这是可以立即部署的实用协议。这是复杂度理论和密码学基础研究。结果依赖证明复杂度和密码学中的重大假设，构造讨论的是原则上什么能够实现。

它**没有把** “Gödel” 变成神奇的安全原语。Gödel 的联系来自证明系统、最优证明系统以及不完备性的有限类比。可用的直觉不是 “不完备性保护你的密码”，而是：如果规则手册无法高效证明模拟器不可能存在，那么需要这份证明的攻击就能在安全定义层面被阻断。

为什么它仍然有趣

密码学经常把困难转化为安全性。因式分解很难，所以 RSA 式假设变得有用。格问题很难，所以格密码学变得有用。这里的困难更奇特：不是 “计算秘密很难”，而是 “证明某个证明对象不存在很难”。

这就是论文让人觉得不寻常的原因。它几乎把公理和规则手册当成了密码学资源。通常的不可能性结果说，可靠性和模拟之间存在张力。Ilango 的做法是把这种张力放到证明论的帷幕后面：模拟器并不存在，但形式系统无法高效揭露它的不存在。

对读者来说，令人惊讶的地方不在于它会取代今天的零知识系统。至少不会直接取代。令人惊讶的是，数学逻辑中的一种限制可以被建设性地利用：它不只是墙，也可以是一种遮蔽物。

证据有多强?

这是一篇定理论文，所以“证据”的含义不同于生物学或天文学论文。问题不是实验是否得到复现，而是定义、假设和证明链是否支持这一主张。

证明是形式化的，论文也明确列出了自己的假设。这些假设并不随意。非交互见证不可区分证明是密码学中经过充分研究的标准构造，可以由若干已经建立的假设推出。“不存在最优证明系统”是证明复杂度中的核心猜想。P = BPP 是一种标准去随机化信念，论文只在关于可证伪性质的更宽泛定理中使用它。

论文还论证了这些假设是合理的代价，而非随意搭设的脚手架：它证明了一个逆向结论，说明这些假设在本质上是必要的——如果这类构造确实存在，那么非交互见证不可区分证明必须存在，而且（假设标准的单向函数存在）最优证明系统不能存在。并且这些假设具有“双赢”性质：反驳其中任何一个，都将本身成为证明复杂度、密码学或复杂度理论中的里程碑发现。

但由于结果是有条件的，对它的信心也必须是有条件的。如果这些假设不成立，定理的解释就会改变。即使假设成立，这一保证也不是完整的经典零知识，而是论文所提出的、基于证明论的放宽版本。

因此，恰当的信心程度是：对于论文确立了一种连贯的条件性可能性结果，我们有较高信心；对于这些假设是否描述了我们实际生活的密码学世界，我们有中等信心；对于任何眼下的实用后果，我们的信心较低。

为什么重要

论文打开了一条本来被认为已经关闭的道路。

经典理论说：没有设置时，完整零知识不可能只有一条消息，也不可能具有完美可靠性。Ilango 的论文说：如果我们只要求那些能够在安全博弈中测试的零知识后果，并允许安全定义依赖于规则手册能够或不能高效反驳的内容，那么许多有用行为都可以恢复——同时保留一条消息、无设置和完美可靠性。

这不是一个微小的定义调整，而是思考密码学保证的一种不同方式。不要只问什么存在，也问你的规则手册能够排除什么。不要把不可证性当成哲学上的麻烦，而要把它当成一种结构。

现实世界明天或许不会改变。但概念地图变了。现在有一种形式化意义上的说法：对于我们希望从“秘密没有泄露”中得到的许多基于博弈的保护，“没人能高效证明秘密已经泄露”可能已经足够强。

这就是为什么 Gödel 会出现在标题里。

简明总结

零知识证明让证明者能够在不透露见证的情况下，使验证者相信某个陈述为真。经典不可能性结果表明，零知识无法在无设置的情况下压缩成一条消息，也无法具有完美可靠性。Rahul Ilango 的论文没有反驳这些不可能性。它定义了一个较弱的概念，即有效零知识：它不要求模拟器确实存在，而要求

一个选定的证明系统——像 ZFC 这样的形式规则手册——无法高效证明不存在模拟器。在密码学的重要假设（非交互见证不可区分证明）和证明复杂度假设（不存在最优证明系统）下，论文为 NP/SAT 构造了无设置、完美可靠的单消息证明者，并且逐一实现零知识性质的可证伪、基于博弈的后果。一个覆盖所有“自然”此类性质的单一证明者，是更进一步、部分依赖猜想的扩展——而覆盖字面意义上的每个可证伪性质很可能不可能，因为证明仍然可以复用。这个结果是理论性的、有条件的，不是已经部署的原语；但它展示了把证明论上的不可证性当作密码学资源的一种新方式。

不忽悠检查

论文展示了什么：在给定假设下，可以构造用于 NP/SAT 的单消息、无设置、完美可靠证明者；相对于任意选定的证明系统，它们是有效零知识的，并实现经典零知识的每一种可证伪、基于博弈的后果。

合理但尚未无条件证明的内容：所需的证明复杂度和密码学假设确实成立。这些是严肃且经过充分研究的假设——论文也说明它们既基本必要又足够——但它们仍然是假设。

它没有展示什么：无交互、无设置且完美可靠的经典零知识；可立即部署的实用系统；证明的可否认性或不可复用性；或者 Gödel 不完备性定理本身就能保护密码学。

主要局限：这一保证是零知识的放宽版本；最宽泛的版本依赖多个假设；关于单一通用证明者的主张仍有一部分属于猜想；而且这一结果主要是基础性的。

普通读者应有多大信心？如果接受这些定义，应当高度相信这是一个重要的条件性理论结果；对这些假设是否符合现实，保持中等信心；对立即实际部署，保持低信心。稳妥的结论是：论文没有打破零知识不可能性，而是找到了一条新的证明论路径，绕过其中与许多安全博弈有关的部分。

来源

Ilango, IACR ePrint 2025/1296
https://eprint.iacr.org/2025/1296

FOCS 2025, DOI 10.1109/FOCS63196.2025.00058
https://doi.org/10.1109/FOCS63196.2025.00058