农业大数据学报 2024, 6(2) :205-219
Journal of Agricultural Big Data
DOI: 10.19788/j.issn.2096-6369.200002

开放科学（资源服务）标识码（OSID）：

区块链上的零知识证明技术及其典型算法、工具综述

万巍[1,2]，刘建伟[1,2]，龙春[1,2*]，李婧[1]，杨帆[1]，付豫豪[1]，袁梓萌[1,2]

1.中国科学院计算机网络信息中心，北京 100083；2.中国科学院大学，北京 100190

摘要：在数据安全和隐私保护日益重要的背景下，零知识证明（Zero-Knowledge Proofs，ZKPs）为保护隐私提供了强有力的工具，成为最具应用潜力的核心技术之一。本文综合探讨了零知识证明技术及其在区块链中的应用。首先，详细介绍了零知识证明的相关概念以及三种典型的技术，对 ZK-Snarks 进行了深入探讨，并讨论了 ZK-Stark 和 Bulletproofs 等其他证明机制，深入对比分析了各自的设计、技术特点、性能和应用场景的差异。在此基础上，重点介绍了 ZKPs 在区块链环境下的应用，并分析整理了编写零知识证明的相关工具，这些工具在提升具体应用的性能方面尤为重要。最后，指出了一些潜在的问题和未来的研究方向。

关键词：零知识证明；隐私保护；区块链应用

1 引言

区块链[1-2]通常被认为是一种公共的，分散的和分布式的账本。在区块链的环境下，所有的历史交易数据都被记录和存储。通常，交易数据包括交易的实现细节，如交易金额、账户地址、账户余额等，属于个人隐私。由于区块链的开放性和透明性，任何人都可以访问存档的交易数据，当用户数据请求存储至区块链以及数据被系统验证时，这些数据信息会在一定程度上泄露给运行系统或其他用户，降低链上数据的安全保密性，带来严峻的安全和隐私挑战[3]。随着 2008 年中本聪设计 Bitcoin 开始，区块链技术逐渐受到工业界与学术界的关注。区块链技术的发展历程涉及了多种隐私保护技术，包括同态加密[4]，环签名[5]，安全多方计算[6]。同态加密允许对加密数据执行计算，而无需解密，从而在保持账户余额和交易金额私密性的同时，支持其可用性。然而，同态加密在确保账户地址隐私方面仍存在局限性。同态加密操作通常比非加密操作要复杂得多，这导致处理速度较慢，尤其是在需要大量计算的应用场景中，这种低效率可能成为一个重大障碍。与此相反，环签名技术允许隐藏签名者的身份，为账户地址的保密提供了一种有效手段，但它并不涉及对余额或交易量的保护。安全多方计算则通过分散的计算任务来确保各方数据的隐私，每个参与者仅知晓自己的输入，无法获得其他参与者的详细数据，这在一定程度上保证了交易的安全性，但无法确保账户地址的匿名性[7]。

零知识证明（ZKP）是由 Goldwasser 等人[8]首先提出的，在密码学领域有着重要的应用。它能够保证证明者在不提供任何有用的相关信息的情况下，使验证者相信一个语句是真实的。零知识证明允许证明者产生一个简短的证明 π，可以说服任何验证者相信证明者的公共输入 x 和秘密输入 w 上的公共函数 f 的结果是 y = f（x,w）。w 通常被称为见证输入或辅助输入。零知识证明保证了如果证明者在计算结果时作弊，验证者以压倒性的概率拒绝，而证明过程不会透露关于秘密 w 的额外信息，包括证明者的数据、证明者的身份和验证者的身份等。在区块链应用中，验证者可以使用 ZKP 来验证证明者在区块链环境中是否有足够的交易量，而不会泄露任何私有交易数据。

收稿日期：2024-01-30；**录用日期：**2024-04-14
基金项目：中国科学院网络安全和信息化专项（CAS-WX2022GC-04）
联系方式：第一作者万巍，E-mail：wanwei@cnic.cn。通信作者龙春，E-mail：longchun@cnic.cn。

不同的零知识证明算法在初始设置，证明方式和验证方式等方面具有不同的特性，依据使用的方法和问题需求的不同，这使得它们适用于不同的场景及领域。本文中主要讨论三种在区块链上应用最广的零知识证明算法：ZK-SNARK[8]，ZK-STARK[9]及 Bulletproof[11]。相比而言，ZK-SNARK 提供了一个相对较短的证明长度，而 zk-STARK [9]比其他类型的零知识证明具有更快的验证和证明速度，且不需要可信设置，这两个特性意味着 zkSTARKs 在投票系统和在线身份识别系统等场景中可能具有巨大的潜力[10]，而 Bulletproof 是一种新型的非交互式零知识证明，同样不需要可信的设置过程，适用于范围证明。

本文的组织结构如下：文章首先引入了零知识证明的相关定义，然后讨论了不同类型的零知识证明以及它们的技术特点和应用场景。特别地，本文对 ZK-Snarks 的发展历程进行了深入研究，分析了如何通过 PCP（Probabilistically Checkable Proofs）和 QAP（Quadratic Arithmetic Programs）进行 ZK-Snarks 的构造，然后，重点介绍了 ZKP 在区块链环境下的应用，并分析整理了编写零知识证明的相关工具。最后，本文指出了零知识证明领域一些潜在的问题和未来的研究方向。

2 零知识证明相关基础介绍

本章将介绍零知识证明的相关概念及典型技术。

2.1 零知识证明相关概念

2.1.1 NP 语言：如果对于语言 L，存在一个多项式时间图灵机 R_L 和一个多项式 p(n)使得：x∈L，当且仅当存在 y，$|y| \leqslant p(|x|), R_L(x,y) =1$。

2.1.2 交互证明（Interactive proof）[12]：一对交互式图灵机（P,V），其中 P 表示一个在时间多项式内运行的概率性“诚实”证明者算法，P*表示恶意的证明者算法， V 表示一个在多项式时间内运行且确定性的先验算法，（P,V）被称为语言 L 的一个交互式证明系统，如果以下条件成立：

完备性：$Pr[(P,V)(x)=1]>1-negl(n)$

可靠性：$Pr[(P^*,V)(x)=1]\leqslant 1-negl(n)$

其中：Pr 表示概率，negl(n)表示一个在输入大小 n 的多项式时间内为可忽略的函数。在交互式证明系统中，完备性确保当待证明的命题是真时，诚实的证明者几乎必然能够说服诚实的验证者接受其证明，可靠性则是当命题为假时，任何不诚实的证明者几乎不可能说服诚实的验证者接受其证明。

更详细地说，对于一个 k 轮的消息交互式证明系统，给定一个将$\{0,1\}^n$映射到有限范围的函数 f, V 和 P 都被给定一个共同的输入 $x\in\{0,1\}^n$，在协议开始时，P 提供一个声称等于 f(x)的值 y，由 IP 指定 P，V 中的一方来发送第一条消息 m_1。该方发送消息以后，另一方再发送消息 m_2，此后消息交替发送，当轮到 V 发送消息 m_i 时，V 是基于输入$\{x,m_1,m_2,\ldots m_k,m_{i-1}\}$产生消息 m_i。证明者 P 和验证者 *v* 交换的消息序列与回答 y 称为一份 transcript $=\{m_1,m_2,\ldots m_k,y\}$。在协议的最后，V 必须输出 0 或 1，1 表示接受证明者的陈述 $y=f(x)$，0 表示验证者拒绝了这一声称[13]。

2.1.3 计算不可区分性：设 Dn，En 是两个分布集合，n 表示安全参数，如果对任意的非均匀概率多项式算法 A 满足下列条件，则这两集合被称为计算不可区分：

$$\delta(n) = |Pr_x \in D_n[A(x)=1] - Pr_x \in E_n[A(x)=1]|$$

2.1.4 模拟器：设（P，V）是某语言 L 的一个交互式证明系统，如果对于每个概率多项式时间交互机 V*，存在一个概率多项式时间的算法 M，使得对于每个 x∈L，以下两个随机变量是不可区分的，则 M 被称为 V*与 P 进行交互的模拟器：

(P,V*)(x): 在共同输入 x 下与交互式机器 P 交互后交互式机器 V*的输出。

M*(x)：机器 M*在输入 x 上的输出。

非正式地，模拟器是一台在不同世界中运行的机器，模拟器虽然不能访问交互式机器 P，但能够模拟 V 与 P 的交互。对于正在评估知识是否被泄露的一方来说，这个世界与现实世界具有不可区分性。

2.1.5 零知识：随机变量 $View_{PV^*}(x)$ 表示 V*的随机带内容和 V*在共同输入 x 上与 P 的联合计算中接收的消息。如果对于每一个概率多项式时间交互式机器 *V**，存在一个概率多项式时间算法 *M** 使得集合 $(\{View_{PV^*}(x)\}_{x\in L})$和$(\{M^*(x)\}_{x\in L})$在计算上不可区分，则称（P, V）是零知识的。进一步的，零知识证明可分为计算零知识，统计零知识与完美零知识。

2.1.6 零知识证明（Zero-Knowledge Proofs, ZKPs）：是指具有零知识性的交互式证明系统[14]，具体来说是证明者 P 和验证者 V 之间的一个协议，用于证明一个陈述 x 属于语言 L。非正式地，这样的协议必须满足三个属性：

完备性：一个诚实的验证者总是接受一个诚实的证明者对陈述 x 使用有效见证 w 所产生的证明。

可靠性：没有无界的 PPT 对手可以使一个诚实的验证者接受一个陈述 $x \notin L$ 的证明。

零知识性：对于任何陈述 $x \in L$，在多项式时间内可以模拟验证者和诚实的证明者之间的交互，而不需要知道见证 w。

零知识证明作为一项重要的密码学技术，有着广泛的应用领域，例如隐私保护、区块链智能合约的验证等。为了更深入地理解零知识证明的多样性和适用性，接下来本文将进一步探讨零知识证明的不同类型，包括 SNARKs（Succinct Non-Interactive Arguments of Knowledge）、STARKs（Scalable Transparent ARguments of Knowledge）以及 Bulletproofs 等。每种类型都在特定情境下具有独特的优势和应用。

2.2　零知识证明分类

在当前的密码学研究和实践中，零知识证明（ZKPs）技术已成为确保数据隐私和完整性的关键工具。零知识证明允许一方（证明者）向另一方（验证者）证明某个陈述的正确性，而无需透露除该陈述正确性之外的任何信息。由于零知识证明底层的构造繁杂，本文更强调零知识证明在区块链上的应用，故本节将深入探讨区块链上三种代表性以及使用范围最广的零知识证明构造：ZK-SNARK（Zero-Knowledge Succinct Non-Interactive Argument of Knowledge）、ZK-STARK（Zero-Knowledge Scalable Transparent ARgument of Knowledge）和 Bulletproof。这三种构造方法体现了零知识证明技术在安全性、效率和实用性方面的不同技术特点及发展趋势。ZK-SNARK 是一种高度压缩且非交互式的零知识证明，适用于区块链和隐私保护应用。它们的主要优点是极高的验证效率和低通信开销，但这种优势的代价是需要一个可信的设置阶段，这可能引入了中心化的风险和潜在的安全漏洞。相比之下，ZK-STARK 提供了一种无需可信设置的零知识证明方法，能够在不牺牲透明度和安全性的前提下提供可扩展性。它利用了密码学中的哈希函数和其他非对称技术，因此理论上在对抗量子计算攻击方面具有更强的韧性。然而，这种方法通常会带来更大的证明尺寸和计算开销。最后，Bulletproof 是一种新型的非交互式零知识证明技术，不需要可信的设置过程，适用于范围证明。

综上所述，ZK-SNARK、ZK-STARK 和 Bulletproof 在各自适用的零知识证明领域扮演着关键角色。另外区块链一个显著的特性就是去中心化。虽然 STARK 可以被归为 SNARK 的一种，但 SNARK 需要可信第三方设置，而 STARK 无需此假设。因此，本文仍将 STARK 与 SANRK 分开讨论。

2.2.1　ZK-Snark

2.2.1.1　Snark 定义

记 $R := R_\lambda$ 为一个 NP 语言 L_R 的高效可判定二元关系。对于对 $(u, w) \in R$ 称 u 为陈述，w 为见证。

论证系统：一个对于 R 的非交互式论证是一个概率多项式算法四元组 $\Pi = (G, P, V, Sim)$，其定义如下：

CRS 生成算法$(crs,td) \leftarrow G(1\lambda,R)$：以某些安全参数 λ，关系 R 作为输入，输出一个公共参考字符串 crs 和一个陷门 td。

证明者算法 $\pi \leftarrow P(crs,u,w)$以 crs、一个陈述 u 和一个见证 w 作为输入，输出论证 π。

验证者算法 $b \leftarrow V(ces,u,\pi)$:以一个陈述 u 和论证 π 以及 crs 作为输入，输出 b = 1 表示证明被接受，或输出 b = 0 表示证明被拒绝。

模拟器 $\tau \leftarrow Sim(crs,td,u)$：以一个模拟陷门 td 和一个陈述 u 以及 crs 作为输入，输出一个论证 τ。

ZK-Snark：一个非交互式论证系统 R 的 ZK-Snark 是指满足以下条件的（G, P, V, Sim）：

完备性：对于关系 *R* 的一个真实陈述，一个诚实的证明者 *P* 拥有一个能够说服验证者 V 有效的证据。更正式地说，对于所有 $\lambda \in N$ 和所有 $(u, w) \in R$，

$(Pr[V(crs,u,\pi) = 1 | (crs,td) \leftarrow G(1\lambda,R), \pi \leftarrow P(crs,u,w)] = 1)$

知识可靠性：存在一个提取器，每当对手产生一个有效的论证时，就能计算出一个证据。提取器可以完全访问对手的状态，包括任何随机的硬币。形式上要求对于所有概率多项式时间（PPT）对手 A，存在一个 PPT 提取器(E^A)使得

$Pr[V(crs,u,\pi) = 1 \wedge (u,w) \notin R | ((u, \pi);w) \leftarrow A | EA(crs)]=negl$

简洁性：一个非交互式论证，其中验证者在 $\lambda + |u|$ 的多项式时间内运行，并且证明大小是 λ 的多项式，称为预处理 SNARK。如果公共参考字符串是 λ 的多项式，则非交互式论证是一个完全简洁的 SNARK。

统计零知识：一个论证是零知识的，如果它不泄露除了陈述真实性的任何信息。形式上，如果对于所有 $\lambda \in N$，对于所有$(u, w) \in R$ 和所有 PPT 对手 A，以下两个分布统计上是接近的：

$D_0=\{\pi_0\leftarrow P(crs,u,w): (crs,td)\leftarrow G(1^{\lambda},R)\}$

$D_1=\{\pi_1\leftarrow Sim(crs,td,u): (crs,td)\leftarrow G(1^{\lambda},R)\}$

2.2.1.2 ZK-SNARK 构造

（1）通过 PCP 模型构造 Snark

PCP 模型：全称为 Probabilistically Checkable Proofs（概率可检验证明），是一种用于描述复杂性类别和证明可验证性的数学模型。该模型由 BABAI L 于 1991 年提出[15]。PCP 定理表明，对于 NP 中的每个问题，存在一个概率多项式时间的证明验证器，它只检查证明中的一小部分就能以高概率确定问题的答案。这个发现对于理解 NP 完全问题的困难性和近似算法的设计具有重大意义。Sanjeev Arora 和 Shmuel Safra 在 1998 提供了 PCP 定理的一个更加精确和优化的表述[16]。他们展示了如何构造高效的概率证明检验器，这些检验器能够仅通过检查证明的一小部分来验证 NP 完全问题的解决方案。这些发现为计算复杂性理论和近似算法的研究提供了新的视角和工具。

Random Oracle Model: 随机预言模型（ROM）[17]是一种理想化的密码模型，它假设存在一个真正的随机函数-称为随机预言机（random oracle）的理想化黑盒。协议的所有参与方都可以访问该函数。它可以接收任意长度的输入，并产生相应的随机输出。它能够将任意长度的输入映射到固定长度的输出，并且这个映射是随机的。由于在现实中不存在这样的理想函数，随机预言机一般是用散列函数实例化的。

Fiat-Shamir 启发式：Fiat-Shamir 变换[18]是一种启发式方法，可将多轮交互协议转换为非交互协议或数字签名，广泛用于将交互式零知识证明转换为非交互式零知识证明中。

基于 PCP 构造 Snark 的经典协议：Kilian 在其 1992 年的研究[19]中，提出了一种针对 NP 问题的简洁零知识论证方法。在这个方法中，证明者 P 利用 Merkle 树为验证者 V 提供对 PCP 证明 π 的访问。具体来说，证明者使用抗冲突哈希函数（CRHF）H 来计算对长字符串 $\pi \in \{0,1\}^{q(\lambda)}$的简洁承诺，并且能够以简洁的方式对 π 的任何比特进行局部开放。证明者首先基于私有输入和见证 w 构建一个 PCP 证明 π，然后使用验证者提供的 CRHF H 构建一个以 π 为叶节点值的 Merkle 树，从而生成一个根值。证明者将这个根值发送给验证者，作为对 π 的承诺。验证者随后抛出固定多项式数量的随机硬币并发送给证明者。证明者 *P* 和验证者 V 通过内部运行 PCP 验证器对输入 *x* 和根值进行 PCP 查询计算。之后，证明者 *P* 返回对这些查询的回答，并附带“证明”称为认证路径—每个回答都与 Merkle 树的根保持一致。如果所有回答都与根值一致并且符合PCP验证器的判断，验证者就接受这个证明。由于验证者V在调用PCP验证器时只做固定多项式数量的查询，每个查询都用固定多项式长度的认证路径回答，而这些都不依赖于见证的长度，因此 Kilian 的协议是简洁的。LIPMAA[20]通过 ROM 模型将 Kilian 的四轮交互式协议降低至两轮交互。Micali[21]通过 Fiat-Shamir 启发式将交互式论证转换成了非交互式论证[19]，其核心思想是使用哈希函数（被模型化为随机预言机）处理其概率可检验证据（PCP）字符串，这既是一种承诺形式，也用于非交互式地生成验证者的 PCP 查询。Micali 结合计算上的可靠性，进一步提高了证明系统的效率，但该方案需要存储和处理大量数据，导致验证过程既耗时又占用大量空间。Paul Valiant[22]提出了一个新颖的概念：“增量可验证计算”。这种方法允许证明者逐步构建证明，证明长时间运行的计算过程是正确的。这种方法适用于复杂或长时间的计算任务，使得在计算的每个阶段都能验证其正确性，而不仅仅在最终阶段，能在维持计算正确性的同时减少资源消耗。但上述方案基于 PCP 的零知识协议构造仅限于理论研究，难以高效实现。Zkboo[23]通过模拟安全多方计算协议的思想，直接构造了高效零知识 PCP，协议的零知识性由安全多方计算协议的隐私性来保障。Zkboo 的核心思想本质上是一种基于加性秘密分享机制的具有二元隐私性的多方计算（MPC）协议。Chase 减少了验证者在挑战响应阶段回复的消息数量，在不增加的计算复杂度的同时，将 Zkboo 的通信复杂度降低了约一半[24]。Chase 等同时提出了一种新型的零知识证明和数字签名方案，这些方案进一步具有后量子安全性，是对传统的、大多基于非对称密码学的零知识证明和签名方法的重要补充和扩展。此外，构造实现了通信复杂度和验证者的时间复杂度的降低，其由安全参数中的多项式、实例的大小以及验证实例的有效证人所需时间的对数限制，从而获得完全简洁的 SNARK[25]。

（2）通过 QAP 模型构造 Snark

在本节中将探讨如何通过 QAP 来构建 SNARK 以及一些经典协议。PCP 和 QAP 在目标上有着共同之处：它们都旨在为复杂的计算过程提供一种高效且可验证的证明机制，同时确保证明的简洁性和非交互性。

PCP 提供了强大的理论基础和广泛的应用前景，而 QAP 在某些方面，尤其是在处理算术电路方面，提供了更高的效率和实用性。基于 QAP 方法的 SNARKs 用于各种实际应用，包括 Zcash [26]等加密货币，以通过 ZK 属性保证匿名性以及防止双花问题。

电路满足问题 Circuit-SAT：针对电路 $C: I_u \times I_w \to \{0, 1\}$，通过关系 $RC = \{(u, w) \in I_u \times I_w : C(u, w) = 1\}$ 描述，其语言被定义为 $L_C = \{u \in I_u : \exists w \in I_w, C(u, w) = 1\}$。

C-SAT 问题属于 NPC（非确定性多项式完全）问题类别，这意味着任何 NP（非确定性多项式时间）问题都可以在多项式时间内被转化为 C-SAT 问题。同时，多数实际应用中的问题都可以通过电路的方式表达，因此，当前流行的简洁非交互式零知识证明通常采用 C-SAT 问题作为待证明命题的表达形式。

二次算术程序 QAP：在域 F 上的二次算术程序 Q 包含三个多项式集合 V={vi(x)}，W={wi(x)}和 Y={yi(x)}其中 i∈{0, 1, . . . , m}和一个目标多项式 t(x)。假设 F 是一个算术函数，它以 n 个域 F 的元素作为输入并输出 n′ 个元素，总共 N = n + n′ 个元素。那么$(c1, . . . , cN) \in F_N$ 是 F 的有效赋值当且仅当存在系数$(c_{N+1}, . . . , c_m)$使得 t(x)除以 p(x)，其中

$$p(x) := \left(v_0(x) + \sum_{i=1}^{m} c_i v_i(x)\right)\left(w_0(x) + \sum_{i=1}^{m} c_i w_i(x)\right) - \left(y_0(x) + \sum_{i=1}^{m} c_i y_i(x)\right).$$

基于 QAP 构造的 ZKSNARK 思路[8]一般是将待证明的陈述规约到 C-SAT 问题上，再将 C-SAT 规约至 QAP 中，直接为 C-SAT 问题构建零知识证明往往难以实现简洁性，然后证明者根据 CRS 模型与其他密码学方法如承诺的构造，通过找到 QAP 的解决方案并编码额外的零知识项来生成证明。其中 CRS 模型假设存在一个由可信第三方生成的公共字符串，证明者和验证者可通过访问该公共字符串生成对应的证明密钥与验证密钥完成非交互的证明与验证过程。

2010 年，Groth[27]基于 CRS 模型将 C-SAT 问题简化为一组方程，并利用双线性配对来验证这些方程的有效性。实现了第一个通信量为常数个群元素的 zk-SNARK。2012 年 Lipmaa[28]成功将协议的 CRS 大小由电路规模的平方级别降低至 O（ClogC）级别。然而证明复杂度仍未降低。在 2013 年，Gennaro[29]提出了一种新的、有影响力的定义 NP 复杂性类的方法：Quadratic Span Programs（QSPs）。这是基于 Karchmer 和 Wigderson[30]定义的 span programs 的自然扩展。QSPs 为加密学和理论计算机科学领域提供了一种新的视角，特别是在理解和构造高效的计算表示方式方面。随后，Lipmaa 对 QSPs 进行了一些变体和改进，他结合现有技术和线性纠错码，提出了一类更高效的二次跨度程序[31]。2013 年，Gennaro 等人[29]提出了 QAP，可将算术电路可满足问题快速归约为 QAP 可满足问题，同时将 CRS 规模降低至电路的线性级别。Parno 提出了 Pinocchio[32]，将通信量进一步降低到了 8 个群元素，且验证复杂度仅与输入输出的大小呈线性关系。Pinocchio 协议的优良性能促进了零知识证明的在区块链隐私的第一次大规模落地应用-Zcash。针对布尔电路提出了改进版本的 Square Span Programs（SSP）[33]，这自然引导出一种简化的算术电路版本，即后来的 Square Arithmetic Programs（SAP）[34]。这些方法通过紧凑编码计算，从而获得高效的零知识 SNARKs。它们的核心思想是将每个门的输入和输出表示为变量，将每个门重写为某些表示门输入和输出线路的变量的方程。只有符合门逻辑或算术规范的线路值才能满足这些方程。通过为电路中的所有门组合这样的约束，任何电路的满足赋值首先可以被指定为一组二次方程，然后作为一组多项式跨度的约束，定义相应的二次/方形跨度程序。因此，证明者需要说服验证者，所有的二次方程都是可满足的，并通过等效多项式找到问题的解决方案。SNARK for QAP 使用范围最广的技术是 Groth 在 2016 年的著名成果[35]，该方案具有完美完备性与完美零知识性。方案在 Generic Group Model 中实现了三个组元素的证明大小以及 3 个配对运算的验证开销。与 Pinocchio 中的构造相比，该构造被简化，且证明元素相对于一些随机因子并没有重复且安全性依赖于更强的模型 GGM。通用群模型[36,37]是一种理想化的密码模型，其中算法不利用群元素表示的任何特殊结构，因此可以应用于任何循环群。在这个模型中，攻击者只能访问随机选择的组编码，而不是有效的编码，例如实际使用的有限域或椭圆曲线组所使用的编码。该模型包括一个执行（加法）组操作的 oracle。因此，可以有效地提取用于将预言机的输出表示为初始组元素的线性组合的系数。此外，Groth 提出基于通用非对称双线性群模型构建

ZK-SNARK 的通信度下限，即无法构造通信复杂度仅为 1 个群元素的 zk-SNARK。然而上述 ZK-Snark 的构造在初始阶段需要一个可信第三方基于某一秘密值生成公共参考串，该秘密值应当在初始阶段完成后被诚实丢弃。一旦该秘密值被攻击者获取，整个协议就不具备可靠性：证明者可以借助该秘密值伪造证明通过验证，从而在实际使用中造成巨大的威胁。Ben 等人提出第一个 ZKP 的 MPC 协议[38]，证明只要至少有一个贡献方是诚实的，协议生成的 CRS 就是安全的。

2017 年，Bowe 等人提出一种 MPC-MMORPG 协议，该协议专门针对基于 pairing 配对的 zk-SNARK[39]，如 Groth16。在 MMORPG 协议中，由中央“协调者”管理参与者之间的消息。相较于之前的 MPC 方案，MMORPG 协议不需要提前选择贡献者，也并不总是需要贡献者随时在线。此外该协议还确保协调器的公开可验证性。近些年 MMORPG 协议已成为区块链的行业标准。以 Ethereum 为代表的众多项目已使用该协议为系统生成 CRS。2018 年 Groth 等人的论文提供了一种创新的 zk-SNARK 协议[40]，它仍利用 QAP 作为基础，并具备全局性和可更新性的 CRS。在传统的 SNARK 协议中，通常需要为每个不同的问题实例生成一个独立的 CRS。这意味着如果有多个不同的问题，就需要为每个问题分别创建和维护一个 CRS。2018 年，Groth 等人引入了 CRS 的全局性概念[40]，这意味着一个 CRS 可以用于多个不同的问题实例，而不需要为每个实例重新生成 CRS。这种全局性降低了系统的复杂性和计算开销，使协议更具可扩展性和实用性。另外在传统的 SNARK 协议中，CRS 是静态的，一旦生成就不能更改。这导致了一个问题，即如果协议需要适应新的问题实例或更新，就需要重新生成整个 CRS，这可能非常耗时和资源消耗。同时 Groth 还引入了 CRS 的可更新性，这意味着 CRS 可以动态地更新，而不需要重新生成。这使得协议能够在运行时向 CRS 中添加新的信息，以适应新的问题实例或协议的演化。这种可更新性使协议更加灵活，并且能够应对不断变化的需求。但该类 CRS 的更新在实际中需要额外的预处理过程与相应的更新计算过程，计算开销往往过大。2019 年 Maller，Bowe 等通过置换论证在代数群模型提高了全局可更新 CRS 构造的效率，其 CRS 与电路规模为线性扩展，这使得它在处理复杂计算时更加高效[41]。2020 年 Plonk 进一步改进了 Maller 等人的成果，显著提高了证明者的效率，计算要求大大减少[42]。在 PLONK 中，证明者只需要处理少量的多项式承诺和打开证明，工作量得到了大幅的减少。这种改进对于计算资源受限的应用尤为重要。PLONK 的设计允许使用 Kate 承诺方案有效地验证多项式方程。这个方案有助于维护零知识特性，并使验证者能够有效地检查一定输入值范围内的多项式方程。所以 PLONK 特别适合于像以太坊上的 zk-rollups 这样的应用，解决了主网的吞吐量问题。它已经被用于 Aztec[43] 项目，该项目专注于以太坊上的隐私保护解决方案。

2.2.2 ZK-Stark

Eli Ben-Sasson 于 2018 年提出了一种称为 zk-STARKs 的新型零知识证明[9]。ZK-STARK 是 ZK-SNARK 协议的改进版本。“STARK”这个缩写代表“Scalable Transparent ARgument of Knowledge”－即可扩展透明知识论证。“可扩展”指的是证明者的运行时间最多是计算大小的准线性级别、验证时间是计算大小的对数级别。也就是说，ZK-STARKs 是一种针对可用对数空间，使用可计算电路表示陈述的非交互零知识论证。“透明”指的是所有验证者信息只是公开抽样的随机硬币。ZK-STARK 不需要可信设置程序来实例化证明系统，而是依赖于基于哈希冲突的对称加密算法，这种特性使其更加高效，并且完全摆脱了 ZK-SNARK 中可信阶段产生的参数，能够有效抗击量子计算机对算法的威胁。如之前所说，非交互式 STARK 是 SNARK 的一个子类，用来指特定的可扩展透明 SNARK 构造。ZK-Stark 通过 AIR（algebraic intermediate representation，代数中间表示）进行约束的表示，STARK 证明系统将在任何时间计算的状态都包含在从有限域取值的寄存器元组中，在每个周期更新状态。而代数执行轨迹（AET）则是按时间顺序排列的所有状态元组的列表。通过 AIR 定义了对代数执行轨迹的两种类型的约束：边界约束（在计算的开始或结束时，指示的寄存器具有给定的值）与转换约束（任何两个连续的状态元组都按照状态转换函数演变）。FRI（Fast Reed-Solomon IOP）是一种协议，其中证明者发送对应于编码的 Merkle 根序列，该编码的长度在每次迭代中减半；验证者通过证明者提供的叶子及其认证路径检查连续轮的 Merkle 树以测试简单的线性关系。对于诚实的证明者，所表示的多项式的次数同样在每一轮中减半，并且因此比编码的长度小得多。然而，对于恶意的证明者，这个长度仅比编码

的长度小一些。不断重复该过程，在最后一步中，证明者发送对应于常数多项式的非平凡编码。这样通过 AIR 与 FRI，得到了一个交互式的证明系统，再通过 Fiat-Shamir 变换可最终得到一个非交互式的 STARK 证明。ZK-STARKs 可以有一个非常快的证明时间和验证时间，但证明大小过大。因此，它在投票系统、在线系统和其他一些需要识别步骤才能访问的服务中有着光明的前景。

2.2.3　Bulletproof

Bulletproof 的构造思路如下：首先将电路中的乘法门约束和乘法门之间的线性约束利用 Schwartz-Zippel 引理[44]归约为一个多项式的某一特定项系数为零的问题，然后将该问题转化为内积论证（IPA）[45]的陈述表示形式，最后调用内积论证实现零知识证明。

Bulletproof 提供了一种更有效的机密交易（CT）范围证明，主要应用于在加密货币领域如 Zcash 中。防弹技术建立在实现通信高效的零知识证明的技术之上，它们可以用来扩展多方协议，如多重签名或零知识紧急支付等，事实上，Bulletproof 可以认为是基于 IPA 的 Snark 构造的一种。Bootle 将 C-SAT 问题归约为一个多项式是否为零多项式的问题[46]，然后调用内积论证实现对数级别的通信复杂度，但在进行对目标多项式承诺这一步骤时，Bulletproofs 通过消除 Bootle 工作[46]的一些多项式计算过程，只需分别对每一项系数进行承诺，从而降低证明者的计算开销，将目标多项式的次数降低至常数。Hoffmann 指出 Bulletproofs 中门的约束关系可转化 R1CS 的表述形式，再通过二次等式（quadratic equations） 表达约束来降低证明者的计算开销，并通过实验证明计算开销约是 Bulletproofs 的 3/4[47]。然而上述协议的验证者复杂度都是线性级别，而且需要大量的群幂运算，在实际应用中的开销过大。Daza 将验证者的复杂度从线性降到了对数级别，增加了实际的可用性[48]。

2.3　区块链上经典零知识证明协议对比

本节对上述三种零知识证明协议进行了分析与对比，如表 1 所示，在证明者的算法复杂度方面，SNARKs 和 Bulletproofs 都需要 O（N*log(N)）的复杂度，而 STARKS 仅需要 O（N*poly-log(N)）的复杂度，这表明 STARKs 在证明的构建上可能有更高的效率，尤其是在处理大量数据时。对于验证者，SNARKs 拥有最低的复杂度，几乎是常数时间（～O(1)），这使得它们在验证证明时非常高效。通信复杂度是指生成的证明大小，SNARKs 在这方面也表现出极高的效率，其证明大小几乎是常数（～O(1)），而 STARKs 和 Bulletproofs 的证明大小随着数据量的增加而缓慢增长。在零知识证明的安全性方面，需要可信设置的协议可能存在中心化的风险，SNARKs 在这方面需要可信设置，而 STARKs 和 Bulletproofs 不需要，这使得后两者在去中心化和安全性方面更为可靠。另外，只有 STARKs 是后量子安全的，这意味着它们能够抵抗未来量子计算机的攻击，而 SNARKs 和 Bulletproofs 在这方面存在潜在的安全隐患。

表 1　经典零知识证明协议对比

Table 1　Comparison of classic zero-knowledge proof protocols

	SNARKs	STARKs	Bulletproofs
证明者复杂度	O(N * log(N))	O(N * poly-log(N))	O(N * log(N))
验证者复杂度	~O(1)	O(poly-log(N))	O(N)
通信复杂度	~O(1)	O(poly-log(N))	O(log(N))
需要可信设置	是	否	否
后量子安全	否	是	否

每种协议都有其优势和劣势，SNARKs 在证明者和验证者的算法复杂度以及通信复杂度上表现出色，但需要可信设置，并且不是后量子安全。STARKs 虽然在证明构建时复杂度较高，但不需要可信设置且是后量子安全，是一种在安全性和未来兼容性方面表现良好的协议。Bulletproofs 在证明大小上具有优势，但在验证者的复杂度上不如 SNARKs 和 STARKs，且同样不是后量子安全。

3　零知识证明的应用

现有的关于零知识证明的研究综述[7,14]，往往在应用研究方面不够深刻，有时过于关注理论的完整性，

而忽视了零知识证明在实际应用中的具体功能和实现，无法深入到具体的应用层面，或者在实际应用案例中无法提供详尽的分析。比如对于区块链的扩容问题，已成为增强区块链网络可扩展性的关键方案。Rollups 通过在二层协议上处理交易并将结果传回主链，能够在提高性能和降低交易费用的同时，保持去中心化和安全性。在这一过程中，零知识证明尤为关键，因为它们允许在主链上对多笔交易的真实性进行一次性验证，而无需逐个验证每笔交易的细节。这降低了计算和存储成本，并且可以确保所有的交易都经过了正确的验证，避免了虚假交易的问题。而在跨链技术方面，ZK Bridge 展示了零知识证明技术在实现不同区块链之间资产与信息传递的潜力。与传统的跨链桥相比，ZK Bridge 的优势在于它不需要引入额外的信任假设，并且可以实现高效率的交易验证，从而降低了计算和存储成本。所以总体而言，零知识证明在区块链应用中的实际落地不仅涉及技术实现，还包括了设计理念和效率问题。当前的零知识证明相关的应用综述文献[49]缺少对于零知识证明在跨链的应用场景的分析与讨论，而跨链已成为零知识证明在区块链中的重要应用场景之一。因此，本节将会对于扩容与跨链这两个具体的应用领域进行分析。另外 Bulletproof 往往应用在机密交易的场景中，如 Zcash，Monero[50]中，它能隐藏交易金额的同时验证交易的合法性，机密交易通过隐藏交易双方的金额来提高隐私性，但传统的方法往往会导致证明的大小与交易数量线性增长，这对区块链的扩展性和效率构成挑战。Bulletproofs 技术的引入改善了这一状况，因为它生成的范围证明既短小也高效，证明的大小与交易金额的位数呈对数关系，而非线性关系。这意味着即使在处理大量交易时，也能显著减少数据的存储和处理需求。此外，如上文所说 Bulletproofs 在保证交易隐私的同时无需 trusted setup，也就是说，在生成零知识证明时无需一个可信的第三方来初始化系统，这消除了中心化的风险，并且在某种程度上提高了系统的安全性。由于本节专注于零知识证明在扩容与跨链的应用领域，故本节不再单独针对 Bulletproof 的应用进行说明。

3.1 扩容

Layer 1 是指基础的区块链协议如比特币和以太坊，构成了区块链的基础架构，它们负责处理网络的基本功能，包括交易验证、共识机制的实施以及保证数据的不可逆性。然而，随着网络参与者的增加和应用的丰富，原有的 Layer 1 技术开始显得不够用，主链的拥堵和高昂的交易费用成了限制区块链大规模应用的瓶颈。

Layer 2 是建立在 Layer 1 基础之上的网络，旨在通过处理链外交易来提升扩展性。Layer 2 解决方案可以在不直接修改 L1 协议的情况下，实现交易速度的提升和成本的降低。Layer 2 技术通常包括状态通道、侧链和 Rollup 等。

随着以太坊生态的日渐繁荣，以太坊主链无法承受庞大的生态，导致整个以太坊网络拥堵。Rollup 是为了缓解 Layer1 扩容问题所提出的可扩展性的方案，通常被称为链下解决方案。它扩展了以太坊并继承了以太坊的安全保证。它的主要目的是在提高以太坊的性能并且降低 Gas 费用的同时，保留分布式协议的去中心化和安全性特点。Rollup 通过将 Layer1 的部分数据转移到二层协议上进行处理，然后将处理结果返送到 Layer1 上，从而增强区块链网络的可扩展性。Rollups 会在其上的网络中将交易打包在一起并进行压缩，然后将打包后的交易发送到 Layer1 主网进行验证，通过一次性验证多笔交易，使得网络效率得到提高，同时增加了可被执行的交易数量，实现了网络扩容。但在这个过程中，需要保证 L1 的节点没有作弊上传虚假交易。根据证明方法，Rollup 可以大致分为两类：乐观（optimistic）—Rollup 和 ZK（零知识）-Rollup。Optimistic-Rollup 的前提是所有交易均有效，除非另有证明。如果交易的有效性受到质疑，验证者需要提供欺诈证明，然后将其发送到主网络进行验证。如果发现无效，交易将被恢复。这种方法依赖于网络参与者彼此保持诚实，从而建立信任和警惕的平衡。但是当用户提供欺诈证据时，主网上的解决方案不会立即得到解决。这可能会导致从 Optimistic-Rollu 链中提取资产时出现延迟，等待时间从几天到甚至几周不等。而零知识证明可以很好地完成上述需求，在 L1 打包多笔交易后同时为这个过程生成零知识证明，验证者在 Layer1 上通过验证该证明后打包生成共识，完成扩容功能。ZK-Rollup 则确保所有交易都经过验证，同时保持交易详细信息完全私密。这不仅增强了安全性，而且提供了所有用户都非常赞赏的更高程度的隐私。ZK-rollups 还提供近乎即时的事务验证，使 Optimistic-Rollup 在速度方面无法与之相比。ZK-rollups 的落地应用包括基于 Snark 的 Scroll[51]，基于 Starknet 的 Starknet[52]，混合 Snark 与 Stark 证明机

制的 Taiko[53]等项目。

3.2　跨链

跨链技术是一种使得加密资产在不同的区块链之间移动和储存的技术。当前市场上存在众多独立运作的区块链，例如比特币和以太坊，但它们之间缺乏直接的互通机制。若无跨链技术，资产将无法在不同链间转移。跨链技术旨在实现不同区块链间资产与信息的互传递，打破不同区块链间的壁垒，促进了资产和数据的交流，同时提高了网络的性能和功能，更好地满足用户需求。

ZK Bridge 作为使用零知识证明技术的跨链桥梁，其最大特点是不需要引入额外的信任假设就可以适应多种不同类型的区块链。在这个解决方案当中，零知识证明是在区块链之外生成的，实际的验证则是在区块链上进行的。这样的做法大幅降低了区块链上的计算和存储成本，是当今市场上一种相当前沿且有潜力的跨链技术。目前，有几个项目正在发展 ZK Bridge 的生态系统，也就是开发基于零知识证明技术的跨链桥解决方案，但皆处于开发阶段。例如，Succinct Labs[54]、Electron Labs[55]、zkIBC[56]、Polyhedra Network[57]的 zkBridge[58] 等。Succinct Labs 推出了 Tendermint X，这是第一个开源的、高性能的 Tendermint ZK 轻客户端，它为 Cosmos 和 Ethereum 之间提供了一个无需信任的 ZK 桥接，标志着将 Cosmos 连接到 Ethereum 的实现，为超过 4000 万美元的 TVL 和超过 15 亿美元的稳定币资产流动提供了安全保障。ZKIBC，即基于零知识证明的区块链间通信协议，是一项创新技术，专门设计来解决现有区块链互操作性方案中的隐私和安全问题。ZKIBC 的核心，是对跨链交易中数据的处理方式的重新思考。传统的区块链互操作方案通常需要在链上公开交易的某些细节，以便进行验证。而 zkIBC 通过构造特定的零知识证明，只向验证者证明交易的合法性（例如，证明者拥有足够的资金进行交易），而不暴露交易的具体细节（如交易金额、资产来源等）。这种方法的一个关键优点是，它大大减少了链上的信息泄露风险，提高了参与双方的隐私保护。相比较与前两者，Polyhedra Network 的 zkBridge 利用其独创的 deVirgo 协议，一种高效的分布式零知识证明协议，实现了令人印象深刻的性能优化和线性可扩展性。deVirgo 协议的核心优势在于它几乎完美的线性可扩展性—在一个分布式计算网络中，随着计算资源的线性增加，证明的生成时间将成倍减少。具体来说，如果网络拥有 M 台计算机，那么生成证明的时间可以减少近 M 倍。这样的设计使得 Polyhedra Network 的 Layer1 能力得到显著地扩展，为区块链应用提供了前所未有的扩展性和效率。deVirgo 协议的这一特性特别适合处理大量数据或高频交易，使得 zkBridge 在处理跨链交易时，不仅保持了零知识证明的隐私和安全性优势，同时也确保了极高的吞吐量和低延迟，这对于金融交易和复杂的去中心化应用（dApps）来说至关重要。

这些项目都充分利用 zk-SNARKS 技术来革新跨链桥的设计。此外，ZK Bridge 在效率方面带来了明显的优势。传统的跨链交易验证往往需要大量的计算和存储资源，导致交易速度缓慢且耗能巨大。然而，通过使用 zk-SNARKs 技术，ZK Bridge 在跨链交易验证过程中能够实现高效率地验证，减少了计算和存储的负担。这种高效率使得交易得以快速验证和确认，从而加快整个跨链交易过程。同时 ZK Bridge 还可以在不牺牲安全性的前提下实现快速验证，这在高频交易环境下尤其有利。传统的跨链解决方案常常受限于交易的规模和速度，很难应对不断增长的用户需求。而通过应用 zk-SNARKs 技术，ZK Bridge 能够实现较小的验证负荷，从而实现更好的扩展性。这种扩展性使得 ZK Bridge 能够应对大规模交易的处理需求，并且在未来的发展中能够轻松扩展以适应不断变化的市场。

4　零知识证明相关工具库

在探讨零知识证明（ZKP）相关的开发库和领域特定语言（DSL）时，理解它们在电路生成中的作用至关重要。这些方案的能效往往取决于电路大小，通常以门的数量来衡量。因此，电路生成是影响整体性能的关键因素之一。计算的转化可以通过手动电路构造工具或利用编译器自动完成。尽管手动转化可能产生更优化的电路，高级编译器则为开发者提供了更多便利。在性能敏感的应用中，低级电路构造工具通常更为重要。

4.1　高级编译器

高级编译器为开发人员提供了一种将计算转换为电路的简单方法。这些编译器接受用高级语言编写的代码。因此，新的和现有的算法都可以很容易地转换。然而，为了产生足够大小的电路，它们可能已经对代码的结构施加了一些限制。

TinyRAM：这是一种用于表达非确定性计算正确性的随机存取机器，特别是在简洁的零知识证明环境中。它旨在将计算任务表示为可以高效验证的形式。TinyRAM 设计为一个简化指令集计算机（RISC），具有字节级和字级可寻址的随机存取存储器。

Geppetto：这是一个全面的可验证计算框架，它实现了一个可扩展的编译器和运行时环境，可以处理从各种源 C 程序和密码学库生成的 LLVM 代码。这个框架在由云计算引发的对可验证计算协议的兴趣中诞生，这些协议允许计算能力较弱的客户端将计算任务安全地外包给远程方。Geppetto 引入了互补技术来减少证明者的开销并增加证明者的灵活性。通过 Multi-QAPs，Geppetto 大幅降低了在计算之间（例如，MapReduce）或在单个计算内共享状态的成本，其降低程度高达两个数量级。通过仔细选择密码学原语，Geppetto 也降低了验证外包加密计算的成本（例如，可验证地计算签名数据）。

ZoKrates：这是一个面向以太坊的 zkSNARKs 工具箱，它帮助用户在 DApp 中使用可验证计算，从用高级语言编写程序的规范，到生成计算证明，再到在 Solidity 中验证这些证明。它通过一个特定领域的语言、一个编译器以及用于生成证明和验证智能合约的生成器，简化了零知识证明固有的复杂性，为开发人员提供了更熟悉和更高级的编程接口。ZoKrates 将离链程序与以太坊区块链联系起来，扩展了 DApp 的可能性。

genSTARK：这是一个基于 JavaScript 的库，用于生成基于 STARK（Scalable Transparent ARguments of Knowledge）的零知识证明。它旨在帮助用户快速轻松地生成计算的 STARK 证明。genSTARK 尽可能地处理模板代码，让用户能够专注于计算的具体细节。

Sdiehl：这是一个纯 Rust 语言实现的 Bulletproofs 库，使用了 Ristretto 来提高安全性和性能。适用于证明关于承诺值的语句，例如范围证明、可验证混洗、算术电路等，支持单范围和多范围证明。

4.2 低级电路构造工具

在零知识证明方案中，性能是一个关键因素，尤其当方案的效率至关重要时，低级电路构造工具成为必不可少的资源。相较于高级编译器，这些工具要求开发者直接手动编写电路或约束，这一过程虽然复杂度较高，但由此生成的电路往往更为精简和高效。通过细致地设计电路，可以显著减少所需的门数量，进而提升整个零知识证明系统的性能。

libSNARK：这是一个 C++库，为 zkSNARKs 证明的创建和验证提供了高效地实现，使得证明的创建和验证过程非常迅速。它还包含了一套灵活的构建工具，允许开发者从底层开始构建复杂的约束系统实例。libsnark 的设计注重于性能和安全性，虽然它是研究级别的概念验证，并且未经过广泛的审查或测试，但其理论安全性建立在详尽分析过的数学构造上。此外，libsnark 还提供了一系列“gadget”库，可以方便地构建可重用的“gadget”以及额外的显式方程。这些库基于模板设计，以便高效地支持在多个椭圆曲线上工作。对于开发者而言，libsnark 是一个强大且灵活的工具，能够帮助他们在保护隐私的同时，实现复杂的密码学协议。

jsnark：它允许用户用 Java 编程语言直接构建和操作 zk-SNARK 电路。jsnark 的主要目的是简化在 Java 环境中开发 zk-SNARK 应用程序的过程。作为一个研究工具，它为零知识证明提供了一个可访问和灵活的开发平台。

Bellman：这是一个基于 Rust 的库，用于约束系统的公式化。它提供了电路特性和基本结构，以及诸如布尔值和数字抽象等基础 gadget 的实现。Bellman 使用 ff 和 group crates 在标量字段类型上通用构建电路。Bellman 的目的是使 zk-SNARK 的使用和实验对普通公众更加易于访问，同时提高下一代 Zcash 的安全性和性能。在 Bellman 中，所有的电路抽象都是通用编写的，涵盖了椭圆曲线和有限域算术。电路合成的核心是 ConstraintSystem trait，负责变量的分配、赋值和约束的执行。

Circom：这是一个为零知识证明（ZKPs）设计的领域特定语言（DSL），专门用于在区块链和密码学应用中构建和验证算术电路。它为开发者提供了一个强大的工具链和生态系统，以满足零知识密码学的具体需求，特别是在以太坊生态系统中的应用。Circom 的核心在于能够将计算问题转换成算术电路的形式，这些电路随后可以用于生成零知识证明。Circom 通过其特有的语法和结构，使得开发者能够方便地创建和表示计算逻辑。它支持信号和约束系统，其中计算被模型化为一组多项式方程，称为 Rank-1 Constraint System（R1CS）。这使得 Circom 不仅能够处理简单的算术运算，还能处理更复杂的密码学函数和算法。此外，Circom 引入了模板和组件的概念，允许开发者

以模块化和可扩展的方式构建电路。这些模板提供了一种参数化结构，使得特定值的电路实例化成为可能，而组件则用于定义具有输入和输出信号的算术电路。Circom 的另一个重要特点是支持并行化计算，这在处理大型电路时特别有益，可以提高见证生成的效率。Circom 与 zkSNARK 生成器（如 snarkjs）紧密协作，将复杂的高级约束翻译成适合 zkSNARK 的格式，从而创建零知识证明系统中的证明者和验证者。

gnark：这是适用于 Go 语言的开源库，用于编写零知识参数协议的电路。

目前已有多个零知识证明的开发框架已被基准测试，以比较它们的性能。例如，Circom、gnark 和 Arkworks 都采用相同的 R1CS 算术化，而 Halo2 和 Plonky2 则采用 Plonkish 算术化。Starky 使用 AIR 算术化，而 Boojum 则基于特定优化达到了较少的约束数量。

5　挑战与未来展望

零知识证明仍然面临许多挑战，同时也带来了许多研究方向。具体如下：

较弱假设的挑战：ZKP 的一个挑战是，是否可以在一些较弱假设下有效实施。例如，Zerocash 中使用了 zkSNARK，但它需要一个受信任的第三方来进行设置和系统初始化。ZKP 可以在没有受信任第三方的情况下实施，但这会影响 ZKP 的效率。因此，研究在没有受信任第三方的情况下有效实施 ZKP 是值得的。Spartan 是一个引人注目的成果[59]，它提供了一种无需可信设置的 zk-SNARKs，特别适用于解决算术电路满足性问题（R1CS）。它的特点在于，它能够在验证证明时产生低于线性的成本，而且不要求 NP 陈述的结构具有一致性。此外，Spartan 实现了时间最优化的证明者，这在先前的 zk-SNARKs 文献中几乎未被实现。Spartan 应用了新技术，如计算承诺和一个加密编译器 SPARK，用于将现有的可提取多项式承诺方案转换为有效处理稀疏多项式的方案，这对于实现时间最优化的证明者至关重要。Spartan 作为 Rust 库实现，并与最新 zkSNARKs 进行了实验比较，表现出多方面的优势，包括在无可信设置方案中具有较快的证明者速度，生成更短的证明，验证时间低，综合效率优秀。

不同机制的融合：每种 ZKP 模型都有其独特的优势，当前研究的趋势是寻找将这些不同 ZKP 模型的优势结合起来的方法。比如 Orion[60]等，正在探索如何更有效地结合不同的 ZKP 模型来提高交叉链互操作性和整体系统性能。

效率优化：在现有的 ZKP 模型中，效率优化方法通常适用于在足够大的域上的算术电路。研究是否存在一种新的效率优化方法，适用于在一些小域或布尔电路上的算术电路，是值得的。同时，这种可能的方法不应该需要任何额外的计算开销。此外，这种方法与减小域大小相关联，且不会影响证明的正确性。

其他数学问题：目前，为了提高 ZKP 的效率，大多数优化方法专注于双线性群的计算研究，研究依赖于其他数学问题来构建高效的非交互式 ZKP 模型的可能性是值得的。一个突破性的进展是 QuickSilver 协议[61]。该协议在电路基础模型中，将计算表示为一个场上的电路，实现了每个非线性门仅需一个场元素的通信复杂性，同时保持了非常低的计算成本。QuickSilver 的实现表现出极高的效率和可负担性。相比之前最佳的实现，它在计算上实现了 6 倍至 7 倍的提升，在通信上实现了 3 倍至 7 倍的提升。这表明在处理小域或布尔电路上的算术电路时，QuickSilver 提供了一种高效的优化方法。

基于格的密码学：公钥密码算法是在区块链环境中构建 ZKP 模型的关键因素。不幸的是，常见算法无法抵抗量子计算攻击，特别是考虑到量子计算对传统公钥密码算法的潜在威胁。例如，RSA 算法可以被量子计算中的 Shor 算法在多项式时间内破解。然而，基于格的密码学问题，如模块-SIS 和模块-LWE 问题，被认为对量子攻击具有抵抗力。最近的研究提出了一个改进的实用协议[62]，用于证明知道满足 $As=t \bmod q$ 的短向量 s。该协议提供了一种更直接且更高效的方式来证明 s 的系数具有小的 2 范数，不需要转换到 CRT 表示。这项新的证明系统可以被黑箱方式插入到各种基于格的隐私原语的构造中，例如可验证的加密方案和群签名方案，使得解决方案比以前的最佳解决方案紧凑两倍以上。

零知识证明的硬件加速：零知识证明技术虽然被广泛认为是解决区块链主要问题的关键方案，但长期受制于其本身的高计算密集性导致的计算效率问题。正是基于这种背景，ZKP 硬件加速成为解决 ZKP 效率问题的一个重要创新方向。ZKP 硬件加速涉及在专用硬件（如 GPU、FPGA 和 ASIC）上实现 ZKP 算法的优化，使其能够更快地处理复杂计算，从而大幅提高 ZKP 的生成和验证速度。在 ZKP 的不同证明系统及

其相关实现中，计算需求与资源开销各不相同。在众多证明系统中，有两种计算操作尤为耗时与昂贵，分别是多标量乘法（Multiscalar Multiplication-MSM）和快速傅里叶变换（Fast Fourier Transform-FFT）。CUZK[63]中提出 MSM 算法占据了证明生成总运行时间的 70%以上。随着新的 ZKP 框架 STARK 的发展，也有更多的证明是基于 FTT 算法为主。

多标量乘法（MSM）是一种在椭圆曲线密码学中常见的操作，它涉及对多个标量和椭圆曲线点的乘法与求和运算。虽然 MSM 可以通过并行处理来加速，但即使在多核心的系统上，对于复杂的应用 MSM 的运算仍然需要消耗大量的资源与时间。MSM 算法需要处理大量的元素与重复执行相同的操作。

以 STARK 为代表的零知识证明系统大量用到了快速傅里叶变换（FFT）。这个算法用于高效计算序列的离散傅里叶变换（DFT）及其逆变换。FFT 的运行过程严重依赖于数据的频繁交换，数据交换过程中需要从大数据集中“随机”地传输元素，这在硬件内存有限的情况下尤为困难。尽管硬件操作本身非常快，但传输数据的时间却显著降低了整体操作速度。除此之外，FFT 算法通常需要将输入数据重新排列成特定顺序以执行变换，这可能需要大量的数据移动，对于大型 FFT 算法规模来说，这可能成为性能瓶颈。FFT 虽然是一种强大且广泛应用的算法，但在大型数据处理和分布式计算环境中，其性能和效率受到数据交换、带宽限制的显著影响。

基于 SNARK 的证明系统主要依赖于 MSM 算法，而 STARK 类证明则主要使用 FFT 算法。因此，目前的硬件加速主要是面向这两种加密算法的需求进行优化。MSM 对硬件的需求包括强大的并行处理能力、较大的内存容量。相比之下，FFT 对硬件的需求则包括高带宽、大内存容量、高效的数据访问模式。

6 结语

本文中首先深入剖析了零知识证明的核心原理，并概述了 Snark、Stark 等不同类型的 ZKPs，展现了它们独特的技术特性和应用场景。本研究特别深入探讨了 ZK-Snarks 的理论基础，如 PCP 和 QAP，并分析了它们在构建零知识证明过程中的关键作用。同时，也将 ZK-Snarks 和 ZK-Stark、Bulletproofs 等其他证明机制进行了比较，揭示了它们在设计和性能上的独特优势。随后，详细介绍了 ZKP 在区块链技术中的应用，包括在确保交易隐私和数据安全性方面的潜力，并综合分析了开发零知识证明相关工具的实践方法，如 Circom、ZoKrates 以及其他低级电路构造工具的使用。特别强调，尽管零知识证明技术已经取得了显著进展，但在实现广泛应用方面仍面临挑战，这包括对零知识证明算法的进一步优化、对电路构造工具的改进以及对新应用场景的开发。文章最后提出了一些潜在的研究问题，并展望了零知识证明技术在加强数据隐私保护和推动区块链技术发展方面的未来方向。

参考文献

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[J]. Computer Science,2008. DOI:10.2139/ssrn.3440802.

[2] Buterin V. A next-generation smart contract and decentralized application platform[J]. White Paper, 2014, 3(37): 2-1.

[3] Badreddine W, Zhang K, Talhi C. Monetization using blockchains for IoT data marketplace[C]//2020 IEEE International Conference on Blockchain and Cryptocurrency. IEEE, 2020: 1-9. DOI:10.1109/ICBC48266.2020.9169424.

[4] Rivest R L, Adleman L, Dertouzos M L. On data banks and privacy homomorphisms[J]. Foundations of Secure Computation, 1978, 4(11): 169-180.

[5] Rivest R L, Shamir A, Tauman Y. How to leak a secret[C]//Advances in Cryptology—ASIACRYPT 2001: 7th International Conference on the Theory and Application of Cryptology and Information Security Gold Coast, Australia, December 9–13, 2001 Proceedings 7. Springer Berlin Heidelberg, 2001: 552-565.

[6] Yao A C. Protocols for secure computations[C]//23rd Annual Symposium on Foundations of Computer Science .IEEE, 1982: 160-164.

[7] 李一聪,周宽久,王梓仲.基于零知识证明的区块链隐私保护研究[J].空间控制技术与应用,2022,48(1):44-52.

[8] Goldwasser S, Micali S, Rackoff C. The knowledge complexity of interactive proof systems[J]. SIAM Journal on Computing, 1989, 18(1): 186-208.

[9] Ben-Sasson E, Bentov I, Horesh Y, et al. Scalable zero knowledge with no trusted setup[C]//Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part III 39. Springer International Publishing, 2019: 701-732.

[10] Gong Y, Jin Y, Li Y, et al. Analysis and comparison of the main zero-knowledge proof scheme[C]//2022 International Conference on

Big Data, Information and Computer Network (BDICN). IEEE, 2022: 366-372.

[11] Bünz B, Bootle J, Boneh D, et al. Bulletproofs: Short proofs for confidential transactions and more[C]//2018 IEEE symposium on security and privacy (SP). IEEE, 2018: 315-334.

[12] Goldreich O. Foundations of Cryptography, Volume 2[M]. Cambridge: Cambridge University Press, 2004.

[13] Nitulescu A. zk-SNARKs: a gentle introduction[J]. Computer Science, 2020.

[14] 李威翰,张宗洋,周子博等.简洁非交互零知识证明综述[J].密码学报,2022,9(3): 379-447.

[15] Babai L, Fortnow L, Levin L A, et al. Checking computations in polylogarithmic time[C]//Proceedings of the twenty-third annual ACM symposium on Theory of computing[J]. Computer Science, 1991: 21-32.

[16] Arora S, Safra S. Probabilistic checking of proofs: A new characterization of NP[J]. Journal of the ACM (JACM), 1998, 45(1): 70-122.

[17] Bellare M, Rogaway P. Random oracles are practical: A paradigm for designing efficient protocols[C]//Proceedings of the 1st ACM Conference on Computer and Communications Security. 1993: 62-73.

[18] Fiat A, Shamir A. How to prove yourself: Practical solutions to identification and signature problems[C]//Conference on the theory and application of cryptographic techniques. Berlin, Heidelberg: Springer Berlin Heidelberg, 1986: 186-194.

[19] Kilian J. A note on efficient zero-knowledge proofs and arguments[C]//Proceedings of the twenty-fourth annual ACM symposium on Theory of computing. 1992: 723-732.

[20] Di Crescenzo G, Lipmaa H. Succinct NP proofs from an extractability assumption[C]//Logic and Theory of Algorithms: 4th Conference on Computability in Europe, CiE 2008, Athens, Greece, June 15-20, 2008 Proceedings 4. Springer Berlin Heidelberg, 2008: 175-185.

[21] Micali S. CS proofs[C]//Proceedings 35th Annual Symposium on Foundations of Computer Science. IEEE, 1994: 436-453.

[22] Valiant P. Incrementally verifiable computation or proofs of knowledge imply time/space efficiency[C]//Theory of Cryptography: Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008. Proceedings 5. Springer Berlin Heidelberg, 2008: 1-18.

[23] Giacomelli I, Madsen J, Orlandi C. {ZKBoo}: Faster {Zero-Knowledge} for Boolean Circuits[C]//25th USENIX Security Symposium (USENIX Security 16). 2016: 1069-1083.

[24] Chase M, Derler D, Goldfeder S, et al. Post-quantum zero-knowledge and signatures from symmetric-key primitives[C]//Proceedings of the 2017 ACM Sigsac Conference on Computer and Communications Security. 2017: 1825-1842.

[25] Bitansky N, Canetti R, Chiesa A, et al. The hunting of the SNARK[J]. Journal of Cryptology, 2017, 30(4): 989-1066.

[26] Sasson E B, Chiesa A, Garman C, et al. Zerocash: Decentralized anonymous payments from bitcoin[C]//2014 IEEE symposium on security and privacy. IEEE, 2014: 459-474.

[27] Groth J. Short pairing-based non-interactive zero-knowledge arguments[C]//Advances in Cryptology-ASIACRYPT 2010: 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings 16. Springer Berlin Heidelberg, 2010: 321-340.

[28] Lipmaa H. Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments[C]//Theory of Cryptography: 9th Theory of Cryptography Conference, TCC 2012, Taormina, Sicily, Italy, March 19-21, 2012. Proceedings 9. Springer Berlin Heidelberg, 2012: 169-189.

[29] Gennaro R, Gentry C, Parno B, et al. Quadratic span programs and succinct NIZKs without PCPs[C]//Advances in Cryptology-EUROCRYPT 2013: 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings 32. Springer Berlin Heidelberg, 2013: 626-645.

[30] Karchmer M, Wigderson A. On span programs[C]//[1993] Proceedings of the Eigth Annual Structure in Complexity Theory Conference. IEEE, 1993: 102-111.

[31] Lipmaa H. Succinct non-interactive zero knowledge arguments from span programs and linear error-correcting codes[C]//Advances in Cryptology-ASIACRYPT 2013: 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I 19. Springer Berlin Heidelberg, 2013: 41-60.

[32] Parno B, Howell J, Gentry C, et al. Pinocchio: Nearly practical verifiable computation[J]. Communications of the ACM, 2016, 59(2): 103-112.

[33] Danezis G, Fournet C, Groth J, et al. Square span programs with applications to succinct NIZK arguments[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014: 532-550.

[34] Groth J, Maller M. Snarky signatures: Minimal signatures of knowledge from simulation-extractable SNARKs[C]//Annual International Cryptology Conference. Cham: Springer International Publishing, 2017: 581-612.

[35] Groth J. On the size of pairing-based non-interactive arguments[C]//Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II 35. Springer Berlin Heidelberg, 2016: 305-326.

[36] Shoup V. Lower bounds for discrete logarithms and related problems[C]//Advances in Cryptology-EUROCRYPT'97: International Conference on the Theory and Application of Cryptographic Techniques Konstanz, Germany, May 11–15, 1997 Proceedings 16. Springer Berlin Heidelberg, 1997: 256-266.

[37] Maurer U. Abstract models of computation in cryptography[C]//Cryptography and Coding: 10th IMA International Conference, Cirencester, UK, December 19-21, 2005. Proceedings 10. Springer Berlin Heidelberg, 2005: 1-12.

[38] Ben-Sasson E, Chiesa A, Green M, et al. Secure sampling of public parameters for succinct zero knowledge proofs[C]//2015 IEEE Symposium on Security and Privacy. IEEE, 2015: 287-304.

[39] Bowe S, Gabizon A, Miers I. Scalable multi-party computation for zk-SNARK parameters in the random beacon model[J]. Cryptology ePrint Archive, 2017.

[40] Groth J, Kohlweiss M, Maller M, et al. Updatable and universal common reference strings with applications to zk-SNARKs[C]//Annual International Cryptology Conference. Cham: Springer International Publishing, 2018: 698-728.

[41] Maller M, Bowe S, Kohlweiss M, et al. Sonic: Zero-knowledge SNARKs from linear-size universal and updatable structured reference strings[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. 2019: 2111-2128.

[42] Gabizon A, Williamson Z J, Ciobotaru O. Plonk: Permutations over lagrange-bases for oecumenical noninteractive arguments of knowledge[J]. Cryptology ePrint Archive, 2019.

[43] https://github.com/AztecProtocol

[44] Schwartz J T. Fast probabilistic algorithms for verification of polynomial identities[J]. Journal of the ACM (JACM), 1980, 27(4): 701-717.

[45] Bootle J, Cerulli A, Chaidos P, et al. Efficient zero-knowledge arguments for arithmetic circuits in the discrete log setting[C]//Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II 35. Springer Berlin Heidelberg, 2016: 327-357.

[46] Bootle J, Cerulli A, Chaidos P, et al. Efficient zero-knowledge arguments for arithmetic circuits in the discrete log setting[C]//Advances in Cryptology-EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II 35. Springer Berlin Heidelberg, 2016: 327-357.

[47] Hoffmann M, Klooß M, Rupp A. Efficient zero-knowledge arguments in the discrete log setting, revisited[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. 2019: 2093-2110.

[48] Daza V, Ràfols C, Zacharakis A. Updateable inner product argument with logarithmic verifier and applications[C]//Public-Key Cryptography-PKC 2020: 23rd IACR International Conference on Practice and Theory of Public-Key Cryptography, Edinburgh, UK, May 4–7, 2020, Proceedings, Part I 23. Springer International Publishing, 2020: 527-557.

[49] 宋英齐,冯荣权.零知识证明在区块链中的应用综述[J].广州大学学报(自然科学版),2022,21(04):21-36.

[50] https://www.getmonero.org/

[51] https://github.com/scroll-tech

[52] https://github.com/starknet-io

[53] https://github.com/taikoxyz

[54] https://github.com/succinctlabs

[55] https://github.com/Electron-Labs

[56] https://www.zkibc.com/

[57] https://github.com/topics/polyhedra

[58] https://zkbridge.com/

[59] Setty S. Spartan: Efficient and general-purpose zkSNARKs without trusted setup[C]//Annual International Cryptology Conference. Cham: Springer International Publishing, 2020: 704-737.

[60] Xie T, Zhang Y, Song D. Orion: Zero knowledge proof with linear prover time[C]//Annual International Cryptology Conference. Cham: Springer Nature Switzerland, 2022: 299-328.

[61] Yang K, Sarkar P, Weng C, et al. Quicksilver: Efficient and affordable zero-knowledge proofs for circuits and polynomials over any field[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. 2021: 2986-3001.

[62] Lyubashevsky V, Nguyen N K, Plançon M. Lattice-based zero-

knowledge proofs and applications: shorter, simpler, and more general[C]//Annual International Cryptology Conference. Cham: Springer Nature Switzerland, 2022: 71-101.

[63] Lu T, Wei C, Yu R, et al. Cuzk: Accelerating zero-knowledge proof with a faster parallel multi-scalar multiplication algorithm on gpus[J]. Cryptology ePrint Archive, 2022.

引用格式：万巍,刘建伟,龙春,李婧,杨帆,付豫豪,袁梓萌.区块链上的零知识证明技术及其典型算法、工具综述[J].农业大数据学报,2024,6(2):205-219. DOI: 10.19788/j.issn.2096-6369.200002.

CITATION: WAN Wei, LIU JianWei, LONG Chun, LI Jing, YANG Fan, FU YuHao, YUAN ZiMeng. An Overview of Zero-Knowledge Proof Technology and Its Typical Algorithms and Tools[J]. Journal of Agricultural Big Data, 2024,6(2):205-219. DOI: 10.19788/j.issn.2096-6369.200002.

An Overview of Zero-Knowledge Proof Technology and Its Typical Algorithms and Tools

WAN Wei[1,2], LIU JianWei[1,2], LONG Chun[1,2*], LI Jing[1], YANG Fan[1], FU YuHao[1], YUAN ZiMeng[1,2]

1. Computer Network Information Center, Chinese Academy of Sciences, Beijing 100083, China;2. University of Chinese Academy of Sciences, Beijing 100049, China

Abstract: In the context of the increasing importance of data security and privacy protection, Zero-Knowledge Proofs (ZKPs) have provided a powerful tool for protecting privacy. This article comprehensively discusses the technology of zero-knowledge proofs and their application in modern cryptography. First, the article introduces the basic concepts of zero-knowledge proofs, as well as different types of ZKPs such as Snarks and Starks, along with their technical characteristics and application scenarios. In particular, the article conducts an in-depth study of ZK-Snarks. At the same time, the article also discusses other proof mechanisms such as ZK-Stark and Bulletproofs, comparing their differences in design and performance. Then, it focuses on the application of ZKPs in the blockchain environment and analyzes the related tools for writing zero-knowledge proofs. Finally, it points out some potential problems and future research directions in the field of zero-knowledge proofs.

Keywords: zero-knowledge proof; privacy protection; blockchain applications