future internet

MDPI

Review

A Survey on Data Availability in Layer 2 Blockchain Rollups: Open Challenges and Future Improvements

Muhammad Bin Saif †, **Sara Migliorini** *,† and **Fausto Spoto** †

Department of Computer Science, University of Verona, 37134 Verona, Italy; muhammadbin.saif@univr.it (M.B.S.); fausto.spoto@univr.it (F.S.)

* Correspondence: sara.migliorini@univr.it

† These authors contributed equally to this work.

Abstract: Layer 2 solutions have emerged in recent years as a valuable alternative to increase the throughput and scalability of blockchain-based architectures. The three primary types of Layer 2 solutions are state channels, sidechains, and rollups. The rollups are particularly promising, allowing significant improvements in transaction throughput, security, and efficiency, and have been adopted by many real-world projects, such as Polygon and Optimistic. However, the adoption of Layer 2 solutions has led to other challenges, such as the data availability problem, where transaction data processed off-chain must be posted back on the main chain. This is crucial to prevent data withholding attacks and ensure all participants can independently verify the blockchain state. This paper provides a comprehensive survey of existing rollup-based Layer 2 solutions with a focus on the data availability problem and discusses the major advantages and disadvantages of them. Finally, an analysis of open challenges and future research directions is provided.

Keywords: blockchain; layer 2; rollups; data availability

Citation: Saif, M.B.; Migliorini, S.; Spoto, F. A Survey on Data Availability in Layer 2 Blockchain Rollups: Open Challenges and Future Improvements. *Future Internet* **2024**, *16*, 315. https://doi.org/10.3390/fi16090315

Academic Editor: Qiang Qu

Received: 26 July 2024
Revised: 26 August 2024
Accepted: 28 August 2024
Published: 29 August 2024

1. Introduction

Blockchain technology has completely revolutionized the digital landscape since the introduction of Bitcoin in 2009 [1]. It provides a decentralized, transparent, tamper-proof, and publicly verifiable transaction platform. The blockchain network is managed by several peer-to-peer nodes, which helps eliminate the need for a single centralized authority, with all its correlated trust and availability issues, using an underlying consensus algorithm. Since its introduction, several public and private blockchain systems have emerged. However, with more users adopting blockchain, the latter still struggles with scalability in order to achieve mass adoption. For instance, traditional blockchain systems, such as Ethereum and Bitcoin, have low transaction throughput, which in turn, leads to extremely high transaction costs and delays [2]. The throughput of Bitcoin is seven transactions per second. Ethereum can process 15–30 transactions per second, which is a very poor result compared to traditional payment systems such as VISA, which can handle approximately 1500-2000 transactions per second. These limitations are inherent to blockchain design and depend mainly on the mechanism to ensure security and trustworthiness. Indeed, the decentralized consensus protocol requires that each transaction be individually verified by all nodes in the network to be considered valid, and hard work is needed to include it inside a block. This process is important to ensure the security and decentralization of the network; however, it introduces some scalability issues in real-world application scenarios.

Blockchain scalability is an open problem, and both academia and industry have proposed solutions to address this challenge [3–5]. Currently, available solutions for scalability can be categorized into two classes: the former, known as *Layer 1* (L1), and the latter, known as *Layer 2* (L2). L1 solutions increase scalability by modifying the parameters on the main chain, for instance, by increasing the block size, changing the consensus algorithm, or applying network sharding. For instance, Ethereum's transition from Proof-of-Work (PoW)

to Proof-of-Stake (PoS) is an L1 approach aimed at increasing throughput by changing the consensus mechanism. However, an L1 solution involves changing the core design of the blockchain, which tends to raise new issues and leads to forking in the blockchain network. Therefore, the limitations of L1 solutions encourage a new research direction for scaling blockchain transaction throughput, known as L2 protocols. These protocols are developed on top of L1; therefore, they inherit the security and decentralization of the underlying base blockchain. The main idea behind L2 protocols is to process transactions outside of the main blockchain while relying on the base blockchain for security, decentralization, and data availability. For instance, ZK-rollups are a prominent L2 solution where transactions are executed off-chain, and the summary of these transactions, along with cryptographic proofs, are periodically posted to L1. This technique inherits the security and decentralization of L1 while significantly improving scalability. In this way, L2 solutions can increase blockchain scalability and performance without altering the core blockchain design, playing a pivotal role in the evolution of blockchain technology [6].

Several L2 alternatives are available in the literature, and they can be categorized into the following main groups: state channels, sidechains, plasma, rollups, and validiums. Among the various L2 solutions, rollups are considered particularly promising because of their ability to process transactions off-chain, thereby improving scalability, while inheriting security and decentralization of the underlying L1 by posting transaction data and proof of state changes [7]. There are two main categories of rollups: *zk-rollups* and *optimistic rollups*, which differ from each other in terms of transaction execution, verification, and processing time. Although rollups are a promising solution for ensuring scalability, they introduce new challenges. In particular, *data availability* is a challenge in which transaction data need to be accessible to any participant. Rollups post the proofs of off-chain computations to L1 to verify that the computations were performed correctly. However, these proofs are not sufficient for independent verification of state transition. The actual transaction data are required, allowing any participant to reconstruct and verify the state independently without relying solely on the rollup operator. This also prevents potential attacks, such as data withholding. The rollup operator could withhold critical transaction data, preventing users from verifying the correctness of the rollup state. This could lead to situations where users cannot challenge incorrect state transitions or recover funds, making the rollup vulnerable to fraud and censorship. Therefore, posting transaction data is important to maintaining the blockchain's trustless, decentralized, and secure nature. Data availability can be ensured by posting data either on-chain or off-chain. Both solutions have limitations and advantages that need to be carefully considered and addressed in the future.

This paper surveys and compares L2 rollup-based solutions with respect to the data availability problem, comparing their advantages and limitations. In particular, it classifies them based on storage architecture, performance, and security. Finally, it summarizes the open challenges and proposes some future research directions. The remainder of this paper is organized as follows: Section 2 reports background about the different L2 solutions and motivates the choice towards the rollup ones. Section 3 provides a comprehensive overview of the rollup solutions. Section 4 summarizes the various data availability solutions in rollup-based L2. Section 5 discusses open challenges and highlights future improvements. Finally, Section 6 concludes the work.

2. Background on L2 Solutions

This section provides a brief introduction and comparison of the available L2 solutions. The aim is to highlight their main differences and justify the choice of concentrating on the rollup-based ones. Table 1 compares the four main kinds of L2 solutions in terms of transaction throughput, latency, security, and trust assumption.

State channels [8] is an offline solution that performs micro-payments outside the main chain through an established secure channel and stores them inside the main chain altogether through a single transaction at the end of the payments. The transactions with state channels are performed off-chain, which requires mutual trust, agreement, and secure

communication among participants. The vulnerability arises if one party goes offline or fails to end the channel by broadcasting the final state to the blockchain within the specified time window. In this case, a malicious actor might take advantage by submitting an outdated state or withholding the final state, which results in an incorrect settlement of funds [9]. Therefore, this solution requires trust and the use of secure channels among participants. Conversely, *sidechains* [10] consist of the use of a secondary chain that is created as an attachment of the primary (main) one to allow the transfer of assets from the main chain to the secondary one at predetermined rates. The existence of the secondary chain is strictly related to the existence of the main chain, but the two chains remain separate and can use different consensus protocols. Thus, no security issues are transferred from one chain to the other. Therefore, the integrity and the security of a sidechain solution solely depend on the characteristics of the secondary chains, as the L1 protocol does not guarantee security. *Plasma* [11] is a particular sidechain for the Ethereum network which uses a smart contract as its core. In particular, transactions are performed on the sidechain to reduce the load of the Ethereum network. In contrast, the block headers of the Plasma sidechain are posted on the main chain periodically for verification. Plasma chains operate in an optimistic manner, where transactions are supposed to be correct until an agent, called a watcher, observes a fraudulent transaction and calls for a dispute resolution. The original data are not published on the main chain. Thus, there is a need for trust in the Plasma operator and the availability of at least one watcher at every moment to ensure that transactions published on L1 are not fraudulent.

Table 1. Comparison of Layer 2 Blockchain Solutions.

Solution	Throughput (TPS)	Latency (s)	Security Level	Trust Assumptions
State Channels	Very High (1000–10,000)	Low (0.01–0.10)	High	Requires trust in participants
Sidechains	High (100–1000)	Moderate (10–60)	Varies	Depends on sidechain's security mechanism
Plasma	High (500–5000)	Low to Moderate (0.1–1)	High	Trust in Plasma operator for data availability
Optimistic Rollups	Moderate to High (4000–10,000)	High (0.1–7 days due to fraud-proof period)	High	Trust in fraud-proof mechanism
Zk-Rollups	High (2000–4500)	Low (0.1–1)	Very High	No additional trust assumptions
Validiums	Very High (9000)	Low (0.1)	High	Trust in data availability committee

Note: The reported throughput and latency are approximate and theoretical, depending on the specific implementation and network conditions.

Rollups are similar to plasma since transaction execution is shifted to L2, but all the data derived from that execution are published back on the main chain [7], which allows the L1 blockchain to host the transactions processed by the L2. Barry Whitehat [12] introduced the concept of rollup in 2018; since then, rollups have emerged as a robust solution to overcome scalability challenges, allowing transactions to be processed off-chain while retaining the security and decentralization of the underlying L1 blockchain. *Zk-rollups* and *optimistic rollups* are the two primary types of rollup-based L2 solutions. In particular, zk-rollup is an off-chain solution that employs zk-proofs to bundle multiple transactions into a single light transaction and post them back to the main chain. Zk-rollup employs zero-knowledge proofs to prove the validity of transactions. Conversely, an optimistic rollup eliminates the requirement for zero-knowledge proof to reduce its computational intensity at the expense of additional trust assumptions.

Finally, *validiums* [13] are novel systems similar to rollups, but differ in their data availability mechanism. In contrast to rollups, in validiums transaction data and a state

root hash are stored off-chain, while validity proofs are stored on-chain. A data availability committee, such as a centralized oracle, that needs to be trusted can regulate the availability of off-chain data.

This paper concentrates on rollup-based L2 architectures since they present an optimum balance between efficiency and security (no additional trust assumptions). The following sections also discuss the number of promising solutions developed so far.

3. Overview of Rollups

The main idea underlying rollups is to process transactions off-chain and then subsequently store the result of the transaction processing back on the main L1 chain. This behavior is particularly useful in the context of smart contracts, where the processing of a transaction can involve the execution of complex functionalities and can require a non-negligible amount of space. In this case, rollups off-load the computing effort and reduce the storage requirements from the main blockchain network. This allows the network to scale while inheriting the security from the underlying L1 blockchain.

This section presents a complete overview of the different components of rollups and of their execution interaction, as well as the different types of considered rollups, with the aim of better analyzing the data availability issue in Section 4.

3.1. Components and Workflow of Rollups

An L2 rollup-based solution consists of multiple components that allow the execution of the transactions off-chain and then the posting of the result of the transaction execution on-chain. Respectively, Figures 1 and 2 illustrate the overall architecture of optimistic rollups and zero knowledge rollups. The nature of each of these components can depend on the specific implementation that is considered, but they can be summarized in the following categories:

User—She/he initiates the transaction by sending a transaction request to a rollup sequencer. This step is similar to submitting a normal blockchain transaction but, in this case, the transaction is directed to a specific rollup protocol instead of an L1 solution.

Sequencer—This is one of the most important components in rollups. The sequencer collects the transactions initiated by the user and sequences them in order. This is a critical step to mitigate the risk of fraudulent practices, such as a front-running attack in which a malicious actor can execute a transaction before other pending ones to take financial advantage. The sequencer ensures that all the transactions are processed in the correct order and error-free manner.

Aggregator—This is another crucial rollup component. The aggregator receives the transactions from users, executes the transactions off-chain, and then rolls the results into a single batch. This last activity is where the name rollup is derived. The final step for the aggregator is to post the batched transactions and the resulting blockchain state as Merkle root to the smart contract deployed on the L1 main chain.

On-chain Smart Contract—L2 solutions are based on the presence of a smart contract deployed on the L1 main net. Such a contract acts as a foundation for rollup activities carried out off-chain. In particular, it verifies and monitors the state transition performed off-chain by linking off-chain rollup executions and on-chain data integrity.

Dispute Resolution (specific to Optimistic Rollups)—Unlike zk-rollups, optimistic rollups consider every transaction valid and do not submit the validity proofs on-chain, which is why it is called optimistic. Optimistic rollups define a challenge period in which anyone can challenge a transaction and create a dispute before finalizing the transactions on the main chain. In this case, a fraud proof is provided to the final user to demonstrate the correctness of the data posted on L1.

Prover (specific to Zk-Rollups)—The prover component is specific to zk-rollups. It generates cryptographic proof using zk-SNARK or zk-STARK for off-chain computation. This proof is then posted on-chain to verify that the computation was performed correctly and the state transition followed the pre-defined rules without revealing the underlying

data or the exact computational steps. The prover also stores the transaction data for availability on L1. Therefore, anyone can reconstruct the current state using transaction data and independently verify this state on-chain.

Data Availability—Transaction data are critical for ensuring the integrity and security of rollups. These data are posted to the L1 blockchain, making them available and accessible to everyone. The data availability ensures that any participant can independently reconstruct and validate the current state of the rollup by accessing the data on the L1 blockchain. Different L2 rollup-based solutions can implement data availability differently, which will be discussed in more detail in Section 4.

The following two subsections better detail the differences between optimistic and zk-rollups, discussing the different information stored in each case. A summarized comparison of zk-rollups and optimistic rollups is also presented in Table 2.

Table 2. Comparison of Optimistic and Zk-Rollups.

Feature	Optimistic Rollups	Zk-Rollups
Verification Method	Utilizes fraud proofs, which require a challenge period to contest transactions.	Uses zero-knowledge proofs to validate transactions, ensuring immediate verification without contestation.
Data Handling	Stores full transaction data on-chain to facilitate potential fraud proofs and challenges.	Posts only transaction proofs and state differences on-chain, reducing data load.
Withdrawal Delay	Withdrawals can take up to one week due to the required fraud-proof window.	Enables instant withdrawals due to the immediate finality of transactions.
Transaction Finality	Transaction Finality is delayed, pending the expiration of the fraud-proof challenge period.	Transaction finality is immediate as cryptographic proofs confirm validity at the time of transaction posting.
Security Assumption	Operates under the assumption that validators are honest and will actively challenge incorrect transactions during the dispute period.	Relies on the mathematical validity of zero-knowledge proofs, assuming accuracy in proof generation.
Developer Complexity	Implementation is less complex than ZK Rollups but still requires mechanisms to handle disputes and fraud proofs.	Typically involves more complex cryptographic operations, increasing development complexity because of advanced mathematical concepts.

3.2. Optimistic Rollups

Optimistic rollups assume that all transactions are valid by default unless challenged, from which the name optimistic comes. In this approach, L2 verifies all transactions and computes the new state Merkle root, ensuring the integrity of the rollup. However, the transactions and state transitions are verified on L1 using fraud proofs only in case a challenge is raised during the dispute period. This approach significantly reduces the on-chain computation and transaction verification on L1. However, it introduces an inevitable challenge period in which transactions can be disputed, potentially taking up to a week. Therefore, latency in transaction finality is a primary challenge in optimistic rollups. This can be a critical issue in applications requiring immediate transaction finality. Furthermore, the security of optimistic rollups relies on an actively participating network of nodes that can detect and challenge fraudulent transactions, therefore, relying greatly on network integrity and node honesty. With reference to Figure 1, the fraud-proofs are maintained in the L2 blockchain, while only the rolled-up data are posted back on the L1 one.

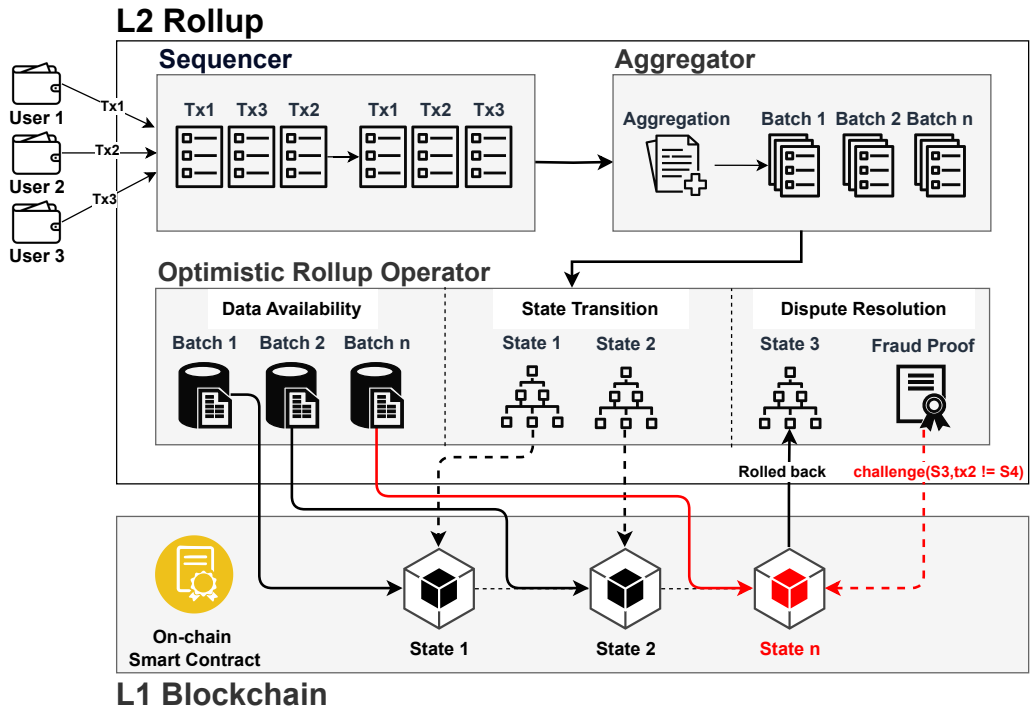

Figure 1. General architecture of Optimistic Rollup.

3.3. Zk-Rollups

Zero-knowledge proofs are the major component of zk-rollups. In cryptography, a zero-knowledge proof is a method by which one party (the prover) can demonstrate to another party (the verifier) that some given statement is true while still avoiding revealing to the verifier any additional information beyond the mere fact of that statement's truth [14]. Generally speaking, a zk-proof can be distinguished by three main characteristics: soundness, completeness, and zero-knowledge. Soundness ensures that no dishonest prover can convince the verifier that a false statement is true. Completeness ensures that a prover can convince the verifier if the statement is true. Finally, zero-knowledge ensures that no information about the statement is revealed except the validity of the statement.

Using zk-proofs, zk-rollups enhance scalability and privacy by allowing off-chain nodes to prove the validity of transactions without revealing any detail. The results of transaction executions are compressed into a single batch and stored in the L1 blockchain together with a cryptographic proof of state transition, as illustrated in Figure 2. In this case, an on-chain verifier smart contract verifies the transactions and updates the blockchain state. This ensures instant transaction finality without a dispute period.

The two main types of zero-knowledge proofs used in zk-rollups are Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARKs) and Zero-Knowledge Scalable Transparent ARguments of Knowledge (zk-STARKs). The following two subsections provide some details about these two alternatives.

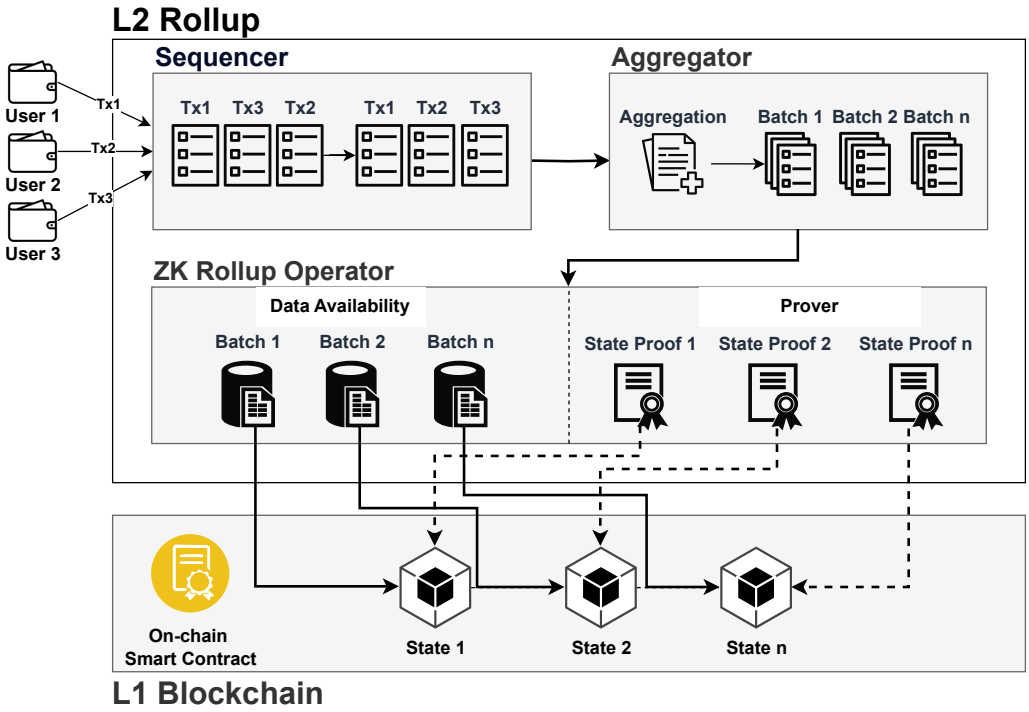

Figure 2. General architecture of Zk-Rollup.

3.3.1. Zk-SNARKs

The name zk-SNARK stays for Zero-Knowledge Succinct Non-Interactive Argument of Knowledge [15]. As suggested by the name, the proof size of zk-SNARK is very compact, regardless of the complexity of the statement. This makes such solutions widely adopted in currently available L2 projects due to their efficiency and small proof size in comparison to the length of the actual computation. Moreover, the non-interactive nature of zk-SNARKs eliminates the need for continuous communication between the prover and verifier in the proof generation process. However, zk-SNARK requires a one-time trusted setup for proof parameters generation, which can pose a security risk.

Formally, a zk-SNARK protocol consists of three steps: Setup (S), Prove (P), and Verify (V), as defined below:

- Setup S—The first step involves the key generation, which takes a secret parameter λ also known as toxic waste, to generate a proving key p and a verification key v.

$$(v, p) \longleftarrow S(1^{\lambda}) \tag{1}$$

  This phase relies on elliptic curve pairings and requires a trusted environment.
- Prove P—The prover generates a proof π using the proving key p, the public statement y and the secret information also known as witness ω.

$$(\pi) \longleftarrow P(p, y, \omega) \tag{2}$$

- Verify V—The verifier can verify the public statement y by using proof π and verification key v, without revealing any information about ω. Verification will return *true* if proof of the statement y is correct.

$$V(v, y, \pi) = true|false \tag{3}$$

3.3.2. Zk-STARKs

Zk-STARK stands for Zero-Knowledge Scalable Transparent Argument of Knowledge [16]. It is an evolution of zk-SNARK since it eliminates the need for a trusted setup, which enhances security and transparency. Indeed, while zk-SNARK relies on elliptic

curve pairings and requires a trusted setup phase for generating public parameters, zk-STARK employs polynomial commitments and interactive oracle proofs, which not only ensure quantum-resistant proofs but also allow scalability for larger computations and transparency. In particular, zk-STARK leverages the Fast Fourier Transform (FFT) for efficient computation and verification of large datasets, while the collision-resistant hashing functions help to achieve transparency and quantum resistance and provide long-term cryptographic security. However, this results in larger proof sizes, in comparison to zk-SNARK, which can pose some problems in terms of transaction costs.

The proof generation and verification of zk-STARK involves the usual three main steps described for zk-SNARK: Setup (S) to initialize public parameters; Prove (P) to generate proofs; and Verify (V) to check the validity of the proofs. The main difference resides in the characteristics and assumptions made in the setup phase.

4. State of Art about Data Availability in L2 Rollups

Blockchain operates in a trustless environment where every peer-to-peer network node can verify the global state independently, without the need for a central authority. In this regard, the term *data availability* is used to denote the fact that any participant can access all the required information needed to verify the transactions in a trustless environment, independently. With reference to the rollup-based L2 solutions, this means that data used for state transitions in L1 need to be accessible to everyone to ensure that any participant can validate the rollup state independently [17].

Data availability is critical in rollups as it directly affects security, functionality, and interoperability. It is essential to ensure that participants interact with a valid blockchain state and trust the rollup protocol. For instance, if data are unavailable and in case of a malicious rollup operator, the users may not be able to withdraw their funds from the main L1 chain because nodes can not correctly compute the new blockchain state and the user's current funds [18]. The main reasons for which data availability is important can be summarized as follows:

- Security—Users can independently verify all the transactions and state transitions, maintaining the trustless nature of the blockchain.
- Integrity—Transactions and states are reliable, correct, and resilient to modifications.
- Functionality—It ensures that all required data are available for processing and validation and ensures the proper functionality of the rollup.
- Interoperability—It ensures consistent data access to allow the interaction between multiple blockchain layers and applications.
- Censorship Resistance—It ensures that transaction data are processed and stored without involving centralized authority or malicious rollup operator. Users can directly submit a transaction on the L1 zk-rollup contract to bypass the rollup operator censorship.

Despite the importance of data availability, this property can lead to problems related to costs and performance due to the amount of data to be posted back on the L1 chain. Currently, there are two main approaches to data availability: on-chain and off-chain. The first one typically guarantees greater security while inducing more costs for the end-user. Conversely, the second one typically reduces costs and improves performance at the expense of smaller security or greater trust requirements.

This section discusses in detail the current state of the art about data availability in rollups-based L2 solutions, which includes on-chain and off-chain data posting. The on-chain includes calldata and blob. The *calldata* refers to a location where function arguments are stored. It is a read-only data area that does not alter the Ethereum state directly. This makes it a cost-effective way to store essential transaction data on-chain. On the other hand, *blob* is a new technique introduced with Ethereum's EIP-4844 (Proto-Danksharding), which allows large amounts of data to be posted on-chain more efficiently. Blob are not directly accessible by the EVM and stored temporarily for verification. Section 4.1 presents a detailed explanation of on-chain data availability solutions. This section also discusses

the impact of data availability on scalability, security, and the overall performance of blockchain. Tables 3 and 4 present a detailed comparison of data availability in the most prominent rollup-based L2 solutions. Specifically, Table 3 summarizes the main technical characteristics of these solutions, while Table 4 highlights their scalability and security properties, as well as the strengths and weaknesses with respect to data availability.

Table 3. L2 Rollup-based solutions to data availability: technical characteristics.

Rollup Name	Data Availability Solution	Provider	Verification Mechanism
ZKSync [19]	On-chain (Calldata)	Ethereum	zk-SNARKs
Starknet [20]	On-chain (Calldata or Blob)	Ethereum	zk-STARKs
Aztec (Private Blockchain) [21]	Off-chain (Initial Plan for Validium)	Off-chain	zk-SNARKs
Scroll [22]	On-chain (Calldata or blob)	Ethereum	zk-SNARKs
Linea [23]	On-chain (Blob)	Ethereum	zk-SNARKs
DeGate V1 [24]	On-chain (calldata)	Ethereum	zk-SNARKs
Taiko [25]	On-chain (calldata)	Ethereum	zk-SNARKs
Arbitum [26]	On-chain (blob) and off-chain (anytrust)	Ethereum/DAC	Fraud Proofs (Optimistic Rollup)
Optimism [27]	On-chain (calldata or blob)	Ethereum	Fruad Proofs (Optimistic Rollup)
Metis [28]	Off-chain	Memolabs	Hybrid (Fraud Proofs and Zk Proofs)
Polygon (zkEVM) [29]	On-chain (Calldata)	Ethereum	zk-SNARKs
Loopring (L2+Dex) [30]	On-chain (calldata)	Ethereum	zk-SNARKs
Immutable X (NFT Solution) [31]	Onchain (calldata) or Off-chain (Validium)	Ethereum/DAC	zk-STARKs
Manta Pacific [32]	Off-chain (Celestia)	Celestia	zk-SNARKs
ZKFair [33]	Off-chain (Celestia)	Celestia	zk-SNARKs

Table 4. L2 Rollup-based solutions to data availability: strengths and weaknesses.

Rollup Name	Scalability	Security Features	Strengths	Weaknesses
ZKSync [19]	High	Strong	Secure, Cost-effective	State Difference Only
Starknet [20]	High	Strong	Scalable, Secure	State Difference Only
Aztec (Private Blockchain) [21]	High	Strong	Privacy-preserving	Pending EIP-4844 Decision
Scroll [22]	High	Strong	Scalable, Secure	Centralized Operator
Linea [23]	High	Strong	Efficient, Scalable	Data Expiry After 18 Days
DeGate V1 [24]	Medium	Strong	Trustless, Decentralized	Unclear Merkle Tree Storage
Taiko [25]	High	Strong	Decentralized, Secure	Relies on Ethereum Validators
Arbitum [26]	High	Strong	Low cost, high throughput	Dispute Delays
Optimism [27]	High	Strong	EVM Equivalence	Dispute Delays
Metis [28]	High	Strong	Cheapest Transactions	Less Community Adaptation and Data Withholding Attack
Polygon (zkEVM) [29]	Medium to High	Strong	EVM Equivalence	Pending EIP-4844 Implementation
Loopring (L2+Dex) [30]	Medium to High	Strong	Flexible, Secure	Centralized Operator
Immutable X (NFT Solution) [31]	Medium to High	Strong	Decentralized, Flexible	Trust Issues in Validium Mode
Manta Pacific [32]	High	Moderate	Scalable, Cost-effective	Relies on Celestia
ZKFair [33]	High	Moderate	Scalable, Cost-effective	Relies on Celestia

4.1. On-Chain (Rollup Mode)

The on-chain data posting, or rollup mode, is a basic functioning mode for data availability in rollup. To ensure the security, transparency, and verifiability of blockchain transactions, all related data must be publicly available on-chain to all network participants for verification. This approach leverages the security and consensus mechanisms of the L1 blockchain, ensuring data integrity and information accessibility. Compared to optimistic rollups, the zk-rollups do not require storing much data on-chain, as validity proofs can

validate the authenticity of state transitions without revealing them. However, on-chain data availability still accounts for 80% of the total transaction cost [34]. To overcome this challenge, different data compression techniques are used, which ultimately reduces the Gas fees for end-users. These techniques include posting only state differences and employing data compression algorithms such as Huffman coding, Run-Length Encoding (RLE), and delta encoding. Moreover, the state compression using Merkle trees and redundancy reduction using erasure coding can significantly reduce the on-chain data while ensuring data availability and integrity.

Posting only the state difference instead of complete transaction data helps to achieve scalability without overburdening the underlying L1. For instance, ZKSync [19] compresses the transaction data and computes the state differences along with zk-proof to post on Ethereum L1. StarkEx [35], which employs the zk-STARK for the zk-rollup solution, is another prominent example of posting only state difference on-chain for data availability.

Calldata and blob storage are the two main techniques to ensure on-chain data availability in rollups. Both solutions have their benefits and limits, as discussed below.

4.1.1. Calldata

In a smart contract, calldata is a read-only data area that works similarly to memory and is used to pass arguments to a function. Therefore, calldata remains on-chain as a part of the Ethereum chain history logs and is not stored as a part of the Ethereum state. Since calldata does not change the Ethereum state, it results in a cost-effective way to store information. Zk-rollups employ calldata to publish compressed transaction data on-chain; the rollup operator creates a new batch by calling the relevant function in the contract and passing the bytes array encoded data as payload to the function arguments. $T_1, T_2, \ldots T_n$ are the individual transactions in the rollup batch. The rollup operator computes proof π and the state transition S of batch B. The calldata for B is as follows:

$$B_S = CE(T_1, T_2, \ldots T_n) \tag{4}$$

where CE is the function that compresses the transaction data and encodes the compressed data in a byte array. B_S along with proof π are then passed as parameters in the function call F_{verify} of the rollup smart contract R_S. An L1 smart contract verifies the proof, updates the state on-chain, and finalizes the batch.

$$F_{verify}(\pi, B_S) \rightarrow \text{Updated state in contract } R_S \tag{5}$$

Loopering [30], a decentralized exchange, employs Ethereum calldata for data availability and zk-SNARK for verification. This ensures that all the essential data for verification are available without preserving the entire transaction history on-chain. In contrast, the Immutable X [31], a zk-rollup-based NFT solution, employs zk-STARK for proof generation and posts the state differences as calldata in L1 or off-chain. Conversely, Optimism [27] is an optimistic rollup employing the calldata or blob to ensure data availability. The batch includes the transaction data and the result of the state transition to ensure verification during the dispute period. Indeed, optimistic rollups assume that the transaction batch is valid unless someone challenges the validity of the batch.

4.1.2. Blob-Carrying Transaction

Initially, the calldata was the only available option for rollups to ensure on-chain data availability. However, using calldata for transaction data storage led to higher Gas fees and limited scalability. For instance, storing a single non-zero byte costs 16 units of Gas, and a single zero byte is 4 units of Gas [36]. Furthermore, Ethereum currently has a restriction of 30 M Gas per block, allowing a maximum size of 1.8 MB per block. To overcome these challenges, on 13 March 2024, Ethereum implemented EIP-4844 (Ethereum Improvement Proposal), also known as Proto-Danksharding [37]. This improvement is part of the Ethereum scaling roadmap, addressing both the cost and performance limitation of

the previous calldata method. The EIP-4844 proposal introduced a new type of transaction, called carrying transaction in short blob [38]. The blob persists in the beacon node for a short time (almost two weeks), significantly enhancing the scalability of the Ethereum network and data availability in L2 [39]. The beacon nodes are special nodes in the network mainly responsible for coordinating validators and maintaining consensus in Ethereum's Proof of Stake (PoS) beacon chain. Blobs do not allow direct access to Ethereum Virtual Machine (EVM) or smart contracts because their design enables efficient data storage and retrieval without requiring direct interaction with the EVM. Ethereum nodes store blobs separately from main transaction data for efficient processing and validation of transactions. Each blob holds 4096 field elements, and each element size is 32 bytes; therefore, a single blob can hold $4096 \times 32 = 128$ KB of data. Furthermore, each block can hold up to 16 blobs or 2 MB of data, significantly improving storage and cost compared to calldata. A zk-rollup called Linea [23] is the first to implement blob for data availability fully, while other zk-rollups, such as Starknet [20], Optimism [27], and Scroll [22], provide both blob and calldata options.

4.2. Off-Chain (Validium Mode)

Off-chain data availability, also known as validium mode, is an alternative option in the rollup architecture, designed to ensure that critical transaction data are available for state transition verification without being stored on-chain. Depending on cost and security tradeoffs, it may be suitable to store the information off-chain and employ other techniques to ensure data availability. Off-chain data storage can significantly increase the transaction throughput. Offloading the data storage to an off-chain solution leads to low data footprints on L1, which can significantly increase the scalability and performance of rollups. The number of transactions can reach up to 9000 per second or more. Moreover, shifting data availability to off-chain can significantly reduce the Gas cost associated with on-chain data storage, making transactions cheaper for end users.

For instance, zk-SNARK-based rollup Polygon CDK [34] provides both options to store the transaction data off-chain or on-chain. Polygon CDK off-chain data availability ensures that state transitions can be verified without extensive on-chain storage by storing transaction data off-chain and submitting zk-SNARK proofs to the Ethereum mainnet. Similarly, Arbitrum Orbit [40] is an optimistic rollup that leverages an off-chain data availability mechanism for storing transaction data. Arbitrum batches transactions and stores the data off-chain to maintain the integrity and security of the rollup and prevent the Ethereum mainnet from overloading. This ensures that the required data for state verification are available during the challenge period.

Data Availability Committees (DACs) and Data Availability Layers (DALs) are the most prominent techniques to ensure off-chain data availability in rollups.

4.2.1. Data Availability Committee (DAS)

A Data Availability Committee (DAS) is a group of trusted nodes that store transaction data off-chain and provide cryptographic guarantees for data availability. This approach addresses the data availability problem while significantly reducing the costs and complexities of storing all transaction data directly on L1. A DAC is selected by forming a group of nodes to store and verify transaction data. These nodes are often determined based on network resilience, performance, and trustworthiness. The specific implementation of the rollup determines if the committee member selection process is permissioned or decentralized. Once a rollup processes a transaction batch, it transmits the corresponding data to the DAC members. Each member maintains a full copy of the transaction data, which ensures that DAC can maintain the data availability even if some nodes fail or become compromised. This helps to achieve redundancy and fault tolerance.

To ensure data integrity and tamper resistance, DAC members generate cryptographic proofs that validate that the stored data are still available. DAC members regularly post these proofs on-chain to confirm the availability of data. The on-chain smart contract verifies proof provided by the committee members and validates the data availability and integrity.

This approach implies that the rollup can trust the DAC. The cryptographic techniques can vary depending on the specific implementation of the rollup. Furthermore, DAC implements a challenge-response technique to improve the security. Any network member can challenge the batch if they believe the DAC is not correctly storing the data. After that, the DAC members must submit additional evidence to validate the data availability. Immutable X [31] supports both on-chain and off-chain solutions, allowing users to store data according to security and cost requirements. The DAC members sign and retain each batch of transactions. Users can still withdraw from the protocol if a compromised member is present. The Immutable DAC includes Immutable, StarkWare, Deversifi, Consensys, Nethermind, Iqlusion, Infura, and Cephalopod [41].

4.2.2. Data Availability Layer (DAL)

The traditional monolithic blockchains are designed with a single layer that handles the transaction execution, consensus, data availability, and settlement. However, this tight integration has led to issues such as low throughput, scalability, and transaction cost. For instance, Bitcoin has low transaction throughput, and Ethereum has high Gas prices. These challenges led to a novel modular blockchain design with separate layers for execution, consensus, data availability, and settlement [42]. Figure 3 illustrates the difference between modular and monolithic blockchains. This modularity allows for improved throughput and lower transaction costs, resulting in more scalable systems. Therefore, a dedicated Data Availability Layer (DAL) can be a storage and consensus layer for rollup, including transaction data, settlements, and off-chain data. A DAL concept is similar to DAC; instead of relying on permissioned nodes, DALs are permissionless networks and employ the Proof-of-Stake (PoS) validator systems to store data off-chain without relying on trusted third parties. DAL is a dedicated layer that ensures the integrity and availability of transaction data without burdening the underlying L1 blockchain.

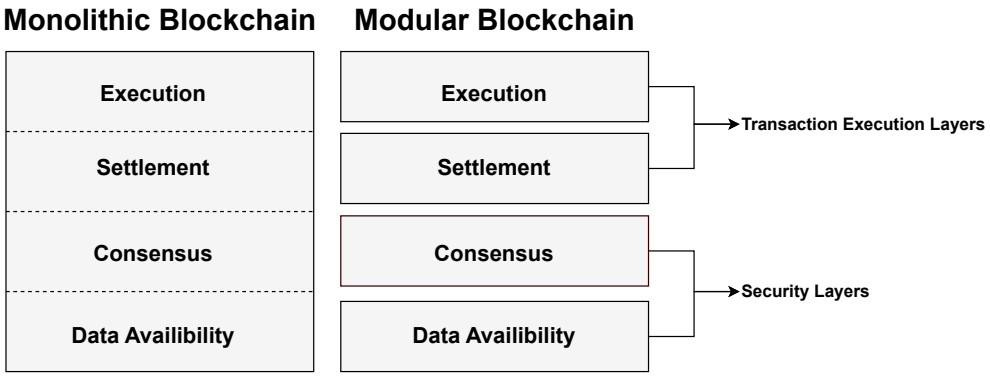

Figure 3. Modular vs Monolithic Blockchains.

DAL distributes transaction data across multiple nodes in a decentralized network while ensuring data redundancy and fault tolerance through techniques such as erasure coding. Erasure coding splits data into multiple segments and allows the original data to be recovered from a subset of these fragments. This ensures that the data are still recoverable even if certain fragments are lost or distorted. A DAL ensures security by making all data available to nodes and preventing malicious activities. Moreover, a dedicated DAL helps achieve a more flexible and modular blockchain design, significantly reducing transaction costs and increasing scalability.

Celestia [43] is a modular pluggable and pre-configured DAL solution that offers rollups to store large datasets off-chain. This allows the rollups to integrate DAL into existing solutions easily. Celestia incorporates data availability sampling (DAS), allowing light nodes to verify data availability without downloading the entire block. Randomly sampling small portions of data ensures that the expected data are consistent. A consistent dataset indicates that the entire dataset is available. Moreover, Namespaced Merkle Trees

(NMTs) allow the node to download only relevant transaction data, which leads to an efficient settlement and execution. These two novel approaches enable rollups to achieve a significant cost reduction of approximately 100x and improved scalability.

Another example of DAL is Avail [44], a spin-off of Polygon. Avail splits the network into light peer-to-peer nodes and employs a sampling technique to verify data availability. Moreover, Avail uses the innovative polynomial KZG commitment schema [45] and the erasure coding for proof validity. The erasure coding distributes data across a decentralized network, ensuring it remains available and verifiable. This separation improves the overall efficiency and scalability of the rollup.

The NEAR protocol [46] is another DAL introduced by the NEAR foundation in 2023, which employs a sharding mechanism called Nightshade. Each shard splits the data into chunks and then integrates them into a single block. These data are available for around 60 h and are then transferred to archival nodes to retain them. In this way, the network distributes the responsibility of data availability, which enhances scalability and reliability.

5. Open Challenges and Future Directions

By considering what has been discussed and summarized in the previous sections, this section identifies some open challenges and proposes some future directions.

The main challenges that still need to be tackled are essentially related to finding the right compromise between the costs of storing data on-chain and the level of security that can be achieved with off-chain storage. In particular, we can summarize the main challenges with the following points:

Cost—The high cost of storing data on-chain is a significant challenge for the broader adoption of rollup solutions, making many rollup solutions less economically viable than their L1 counterparts. For instance, storing data on calldata can cost 16 units of Gas for non-zero bytes and 4 units for zero bytes. Blob aims to reduce on-chain storage costs. However, in some cases, such as during periods of high demand, blobs can be more expensive than traditional calldata, preventing rollups from implementing blob-based solutions. Moreover, another significant challenge in the blob is that rollups have to pay the full blob fee regardless of the data size, which can be expensive for smaller rollups. In this case, rollups need to wait until they have enough data to fill a blob slot, potentially delaying transaction finality or submitting smaller data segments, increasing user costs. Blob still needs to improve in terms of efficient implementation and scalability.

Security and Trust—The security and trustworthiness of data availability solutions are crucial. The decentralized solutions aim to reduce dependency on trusted third parties; however, these solutions still face new security challenges. For instance, DAC relies on a few trusted nodes, which poses a risk in case of malicious or compromised nodes. Moreover, data withholding attacks are significant challenges in the data availability layer, where malicious actors can withhold data to disrupt the rollup. Furthermore, maintaining the integrity and verifiability of off-chain data is still challenging, as it requires robust mechanisms to ensure that data are always available and valid.

Governance and Regulation—Another challenge is the constantly evolving governance and regulations of blockchain systems. Blockchain technologies and their applications have to handle increasingly complex regulatory and governance rules, which can ensure compliance while maintaining decentralization and user trust in the system. Therefore, rollup solutions need to adapt to regulatory challenges and incorporate robust governance mechanisms to address these dynamics.

Interoperability—The interoperability between various data availability solutions and blockchain networks is essential for developing a robust ecosystem. However, several technical and standardization challenges exist in achieving interoperability in the blockchain ecosystem. Furthermore, different blockchains and rollup solutions often employ different protocols and data formats, which can further complicate the efforts needed to ensure the interaction and integration of multiple rollups. Therefore, standardized frameworks and protocols for interoperability are still an open challenge.

Given such main challenges, several future research directions need to be investigated to improve the quality of L2 solutions from an economic, performance, and security point of view.

Robust and Scalable Solutions—Future research can focus on developing more robust and scalable data availability solutions. Off-chain data availability techniques, such as more decentralized and robust DACs and DALs, can also be crucial. Moreover, new data repetition and identification methods can improve storage efficiency and ensure that unique transaction data are posted to the blockchain. In addition, novel techniques, such as enhanced data compression, erasure coding, and advanced cryptographic methods, could significantly reduce Gas costs and improve transaction throughput. Furthermore, implementing modular blockchains can address the performance and scalability challenges of monolithic blockchains. Decoupling transaction execution, consensus, data availability, and settlement layers allows modular systems such as Celestia and Avail to provide more flexible and scalable solutions. Future research can focus on further enhancing the modular blockchain layers and improving the existing ones for more efficient and resilient networks.

Security and Trust Models—The resilient security frameworks for data availability solutions are critical, which involve improving cryptographic proof techniques to ensure data integrity and developing more robust protocols to protect against different attacks. Moreover, hybrid techniques combining on-chain and off-chain data availability can balance security and efficiency. Advanced cryptographic techniques such as homomorphic encryption and Zero-Knowledge Proofs can improve data integrity and privacy while reducing computing overhead. Future work can also investigate the integration of quantum-resistant cryptographic algorithms in blockchain architectures.

Cross-Layer and Cross-Chain Interoperability—The interoperability between different blockchain layers and across various blockchain networks is crucial for the broader adoption of blockchain technology. Future research can investigate developing standardized protocols and frameworks to facilitate the interaction between rollup solutions and mainnet blockchains. Additionally, focusing on developing communication standards and universal data formats to ensure data validation and exchange between different blockchains can mitigate the interoperability issue.

Hybrid solutions—Given all the previous considerations, we can also conclude that a hybrid solution integrating on-chain and off-chain data storage with real-time price analysis can significantly advance the data availability landscape. Hybrid frameworks can dynamically assign data to either calldata or blobs based on cost and storage requirements. The concept of shared blobs, in which multiple rollups collaborate to share data slots, provides an innovative approach to high-cost problems while improving transaction finality. Further research on these concepts will be critical for improving the performance and scalability of rollup solutions.

6. Conclusions

Layer 2 architectures have emerged in recent years as a valuable solution to improve the scalability of traditional blockchains, such as Ethereum while retaining the security and reliability of the main network. Several kinds of L2 alternatives have been proposed and developed, and they can be categorized into the following main groups: state channels, sidechains, and rollups. Rollups are certainly the most promising solution since they provide the best combination of security and performance. However, the introduction of rollup-based L2 solutions introduces additional challenges that need to be carefully considered in real-world applications. One of the most important is data availability, which refers to the need to store back in the L1 the information required to validate the work conducted in the L2.

This paper provides a detailed survey of the currently available L2 solutions by comparing them with respect to data availability and the solutions they provide to this problem. The various alternatives are carefully considered and described, and an analysis of their strengths and weaknesses is also provided. In particular, a main distinction is made with respect to on-chain and off-chain solutions and, for each of them, a further distinction

is made based on the solution applied to implement them. The paper concludes with a discussion of open problems and future research improvements.

Author Contributions: Conceptualization, M.B.S. and S.M.; methodology, M.B.S. and S.M.; writing—original draft preparation, M.B.S., S.M. and F.S.; writing—review and editing, S.M.; supervision, S.M. and F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This study was partially carried out within the Interconnected Nord-Est Innovation Ecosystem (iNEST) and the initiative "Innovative PhDs which respond to the companies demand of innovation". It received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.5 – D.D. 1058 23/06/2022 ECS00000043, and MISSIONE 4 COMPONENTE 2, INVESTIMENTO 3.3—DM 352/2022 progetto M4C2 Investimento 3.3). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

Acknowledgments: During the preparation of this work, the authors used Grammarly's generative AI writing feature and ProWritingAid AI writing assistant in order to check grammar, improve writing, and fix passive verbs to active verbs. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

L1	Blockchain Layer 1
L2	Blockchain Layer 2
ZK	Zero Knowledge
zk-SNARK	Zero-Knowledge Succinct Non-Interactive Argument of Knowledge
zk-STARK	Zero-Knowledge Scalable Transparent Argument of Knowledge
DAS	Data Availability Committee
DAL	Data Availability Layer

References

1. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2009. Available online: http://www.bitcoin.org/bitcoin.pdf (accessed on 27 August 2024).
2. Weber, I.; Gramoli, V.; Ponomarev, A.; Staples, M.; Holz, R.; Tran, A.B.; Rimba, P. On Availability for Blockchain-Based Systems. In Proceedings of the 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS), Hong Kong, China, 26–29 September 2017; pp. 64–73. [CrossRef]
3. Sanka, A.I.; Cheung, R.C. A systematic review of blockchain scalability: Issues, solutions, analysis and future research. *J. Netw. Comput. Appl.* **2021**, *195*, 103232. [CrossRef]
4. Yang, D.; Long, C.; Xu, H.; Peng, S. A Review on Scalability of Blockchain. In Proceedings of the 2020 2nd International Conference on Blockchain Technology, ICBCT '20, New York, NY, USA, 12–14 March 2020; pp. 1–6. [CrossRef]
5. Zhou, Q.; Huang, H.; Zheng, Z.; Bian, J. Solutions to Scalability of Blockchain: A Survey. *IEEE Access* **2020**, *8*, 16440–16455. [CrossRef]
6. Sguanci, C.; Spatafora, R.; Vergani, A.M. Layer 2 blockchain scaling: A survey. *arXiv* **2021**, arXiv:2107.10881.
7. Thibault, L.T.; Sarry, T.; Hafid, A.S. Blockchain Scaling Using Rollups: A Comprehensive Survey. *IEEE Access* **2022**, *10*, 93039–93054. [CrossRef]
8. Dziembowski, S.; Faust, S.; Hostáková, K. General State Channel Networks. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS'18, Toronto, ON, Canada, 15–19 October 2018; pp. 949–966. [CrossRef]
9. Negka, L.D.; Spathoulas, G.P. Blockchain State Channels: A State of the Art. *IEEE Access* **2021**, *9*, 160277–160298. [CrossRef]
10. Singh, A.; Click, K.; Parizi, R.M.; Zhang, Q.; Dehghantanha, A.; Choo, K.K.R. Sidechain technologies in blockchain networks: An examination and state-of-the-art review. *J. Netw. Comput. Appl.* **2020**, *149*, 102471. [CrossRef]
11. Poon, J.; Buterin, V. Plasma: Scalable Autonomous Smart Contracts. 2017. Available online: https://www.plasma.io/plasma.pdf (accessed on 27 August 2024).
12. Whitehat, B. Roll Up: Scale Ethereum with SNARKs. 2018. Available online: https://github.com/barryWhiteHat/roll_up (accessed on 15 July 2024).
13. Lavaur, T.; Lacan, J.; Chanel, C.P.C. Enabling Blockchain Services for IoE with Zk-Rollups. *Sensors* **2022**, *22*, 6493. [CrossRef] [PubMed]

14. Aad, I. Zero-Knowledge Proof. In *Trends in Data Protection and Encryption Technologies*; Mulder, V., Mermoud, A., Lenders, V., Tellenbach, B., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2023; pp. 25–30. [CrossRef]
15. Bitansky, N.; Canetti, R.; Chiesa, A.; Tromer, E. From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again. In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. Association for Computing Machinery, ITCS'12, Cambridge, MA, USA, 8–10 January 2012; pp. 326–349. [CrossRef]
16. Ben-Sasson, E.; Bentov, I.; Horesh, Y.; Riabzev, M. Scalable, transparent, and post-quantum secure computational integrity. *IACR Cryptol. ePrint Arch.* **2018**, 46.
17. Huang, C.; Song, R.; Gao, S.; Guo, Y.; Xiao, B. Data Availability and Decentralization: New Techniques for zk-Rollups in Layer 2 Blockchain Networks. *arXiv* **2024**, arXiv:2403.10828.
18. Tas, E.N.; Adler, J.; Al-Bassam, M.; Khoffi, I.; Tse, D.; Vaziri, N. Accountable safety for rollups. *arXiv* **2022**, arXiv:2210.15017.
19. ZKsync Era Documentation. 2024. Available online: https://docs.zksync.io/build (accessed on 27 August 2024).
20. Starknet Documentation. 2024. Available online: https://docs.starknet.io/ (accessed on 27 August 2024).
21. Aztec Protocol Documentation. 2024. Available online: https://docs.aztec.network/ (accessed on 27 August 2024).
22. Scroll Protocol Documentation. 2024. Available online: https://scroll.io/blog/blobs-are-here-scrolls-bernoulli-upgrade (accessed on 27 August 2024).
23. Linea Documentation. 2024. Available online: https://docs.linea.build/ (accessed on 27 August 2024).
24. Degate Documentation. 2024. Available online: https://docs.degate.com/ (accessed on 27 August 2024).
25. Labs, T. TAIKO: A Type-1 Ethereum ZK-Rollup. 2022. Available online: https://github.com/code-423n4/2024-03-taiko/blob/main/packages/protocol/docs/taiko-whitepaper.pdf (accessed on 27 August 2024).
26. Arbitrum White Paper. 2024. Available online: https://github.com/OffchainLabs/nitro/blob/master/docs/Nitro-whitepaper.pdf (accessed on 27 August 2024).
27. Optimism Documentation. 2024. Available online: https://docs.optimism.io/stack/transactions/fees (accessed on 27 August 2024).
28. Metis Whitepaper. 2024. Available online: https://drive.google.com/file/d/1-hGL4mj8hLtWV8jlt6zRz63yKY14cvyr/view (accessed on 27 August 2024).
29. Polygon zkEVM Documentation. 2024. Available online: https://docs.polygon.technology/zkEVM/ (accessed on 27 August 2024).
30. Wang, D.N.; Zhou, J.; Wang, A.; Finestone, M. Loopring: A Decentralized Token Exchange Protocol. 2018. Available online: https://docs-protocol.loopring.io/ (accessed on 27 August 2024).
31. Immutable X Documentation. 2024. Available online: https://docs.immutable.com/docs/ (accessed on 27 August 2024).
32. Manta Network Documentation. 2024. Available online: https://docs.manta.network/docs/Introduction (accessed on 27 August 2024).
33. ZKFair Documentation. 2024. Available online: https://docs.zkfair.io/ (accessed on 27 August 2024).
34. Technology, P. Polygon CDK: All About Data Availability. 2024. Available online: https://polygon.technology/blog/polygon-cdk-all-about-data-availability (accessed on 15 July 2024).
35. *StarkEx Documentation*. 2024. Available online: https://docs.starkware.co/starkex/con_data_availability.html (accessed on 27 August 2024).
36. Kotzer, A.; Gandelman, D.; Rottenstreich, O. SoK: Applications of Sketches and Rollups in Blockchain Networks. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 3194–3208. [CrossRef]
37. EIP-4844 Blobs Documentation. 2024. Available online: https://ethereum.org/en/developers/docs/data-availability/blockchain-data-storage-strategies/#eip-4844-blobs (accessed on 27 August 2024).
38. Park, S.; Mun, B.; Lee, S.; Jeong, W.; Lee, J.; Eom, H.; Jang, H. Impact of EIP-4844 on Ethereum: Consensus Security, Ethereum Usage, Rollup Transaction Dynamics, and Blob Gas Fee Markets. *arXiv* **2024**, arXiv:2405.03183.
39. Meister, B.K.; Price, H.C. Gas Fees on the Ethereum Blockchain: From Foundations to Derivatives Valuations. *arXiv* **2024**, arXiv:2406.06524.
40. Arbitrum Orbit Documentation. 2024. Available online: https://docs.arbitrum.io/launch-orbit-chain/orbit-sdk-introduction (accessed on 27 August 2024).
41. Immutable X Whitepaper. 2024. Available online: https://uploads-ssl.webflow.com/646557ee455c3e16e4a9bcb3/6499367de527dd82ab7475a3_Immutable%20Whitepaper%20Update%202023%20(3).pdf#page=5.21 (accessed on 27 August 2024).
42. Cohen, S.; Goren, G.; Kokoris-Kogias, L.; Sonnino, A.; Spiegelman, A. Proof of availability and retrieval in a modular blockchain architecture. In *Financial Cryptography and Data Security*; Springer: Cham, Switzerland, 2023; pp. 36–53. [CrossRef]
43. Celestia. Celestia Documentation. 2024. Available online: https://docs.celestia.org/learn/how-celestia-works/data-availability-layer (accessed on 27 August 2024).
44. Avail. Avail Documentation. 2024. Available online: https://docs.availproject.org/docs/ (accessed on 27 August 2024).
45. Kate, A.; Zaverucha, G.M.; Goldberg, I. Constant-Size Commitments to Polynomials and Their Applications. In *Advances in Cryptology—ASIACRYPT 2010*; Abe, M., Ed.; Springer: Berlin/Heidelberg, Germany, 2010; pp. 177–194. [CrossRef]
46. *NEAR White Paper*. 2024. Available at Available online: https://discovery-domain.org/papers/the-official-near-white-paper.pdf (accessed on 27 August 2024).