[1]Ravi Prakash B

[2]Mohamadi Begum Y

An Efficient and Scalable Framework for Decentralized Finance Application Using Blockchain Interoperability

Abstract: - In recent years, blockchain technology has drawn a lot of attention, especially in the field of decentralised finance (De-Fi). However, scalability problems have come to light as a significant obstacle to the broad use of blockchain-based applications. To solve the issue of scalability, this paper has created a decentralised finance application with three main components: the addition of more liquidity to the swapping application, the implementation of a Polygon Proof of Stake bridge to enable efficient asset transfers, and the ability to transfer tokens between accounts seamlessly regardless of network agnosticism. The first feature, network agnostic capabilities for interoperability, facilitates token transfers between blockchain networks, allowing users to access and transact across them with ease The second component, the Polygon Proof-of-Stake bridge, makes asset transfers more efficient by taking advantage of the Polygon network's scalability advantages, which drastically lower transaction costs and processing times. Finally, adding more liquidity to the swapping programme makes it more scalable by guaranteeing that there is enough money for transactions, which prevents delays and bottlenecks. The scalability issue with blockchain technology is efficiently resolved by adding these three characteristics to the decentralised finance application, creating new opportunities for the mass acceptance and utilisation of blockchain-based financial services.

Keywords: Bridge, Scalability, Bitcoin, Network Agnostic, De-Fi, Swapping, Stake, Polygon, Blockchain.

"**Abbrevations** : - De-Fi – Decentrilzed Finance, dApps – Decentrilized Applications , PoS – Proof of Stake"

I. INTRODUCTION

Blockchain technology is a distributed ledger system that lets various parties administer a single database, doing away with the need for a central authority. Since its initial release in 2009, Bitcoin has attracted interest and developed to serve a wide range of applications beyond only currency [1]. Fundamentally, a blockchain is composed of several linked blocks, each containing a list of transactions. To produce a transparent and unchangeable record of each transaction, these are gathered, hashed, and linked to the preceding block. Because blockchain is decentralised, there is no way for one person to control the network, which increases security and reduces the likelihood of fraud.

Blockchain technology has progressed significantly since Ethereum's launch in 2015 [2]. Ethereum introduced smart contracts, which are contracts automatically executed based on coded clauses [3]. This innovation enabled the creation of decentralized applications (dApps) operating independently without intermediaries, especially in the realm of decentralized finance (De-Fi) [4]. MakerDAO, emerging in 2015, notably popularized decentralized stablecoins, notably Dai, tethered to the US dollar, by using Ethereum as collateral [5]. This marked DeFi's formal initiation in providing financial services in a decentralized and trustworthy manner. The success of Maker DAO spurred the emergence of decentralized lending platforms like

Compound [6] and Aave [7], allowing direct borrowing and lending among users, bypassing intermediaries. Consequently, people globally now have more alternatives for financial services beyond traditional banks. Despite its popularity, De-Fi has encountered challenges, notably in scalability - affecting transaction throughput and fees, crucial factors for De-Fi applications.

Transaction throughput pertains to the quantity of transactions that a blockchain network can handle and authenticate within a designated timeframe. Commonly assessed through metrics like transactions per second (TPS) or transactions per minute (TPM). The significance of transaction throughput lies in its ability to determine the speed and capacity of a blockchain in managing transactions.

However, users must pay transaction fees in order for their transactions to be recorded in the blockchain. These fees serve two purposes: they discourage spam and denial-of-service attacks and incentivize miners or validators to prioritise and add transactions to the blockchain. Usually, the blockchain network's native coin is used to pay transaction fees.

The way that transaction fees and transaction throughput are related affects a blockchain network's scalability directly. Increased transaction fees may encourage miners or validators to give priority to certain transactions, hastening their inclusion into the blockchain. To speed up transaction processing, consumers may, however, bid greater fees in this

[1]Department of Computer Science & Engineering, Presidency University, Bangalore, India

[2]Department of Computer Science & Engineering, Presidency University, Bangalore, India

competitive market. Therefore, high transaction costs may make blockchain technology less usable and accessible, particularly in applications where frequent transfers or microtransactions are involved.

De-Fi, short for decentralized finance, involves restructuring traditional financial institutions using blockchain technology and cryptocurrencies, eliminating intermediaries like banks. Despite its potential for financial inclusion and innovation, De-Fi faces challenges in scaling, referring to a network's ability to handle a growing user base and transactions without compromising service quality or significantly increasing transaction fees. Achieving scalability is crucial for a seamless user experience in De-Fi applications, given the need for fast transaction processing and reasonable pricing. This scalability challenge is particularly pronounced in blockchain systems when dealing with a high volume of transactions or supporting complex applications. One approach to addressing this challenge is through interoperability, which involves enabling communication and interaction between different blockchain networks.

Interoperability, along with cross-chain technology, has the potential to mitigate scalability limitations by leveraging the strengths of multiple chains or platforms. This can be achieved by facilitating the transfer of data and assets between various blockchain networks, reducing network congestion, and increasing overall capacity. Effective interoperability requires seamless interaction and communication between different blockchain networks, fostering a more cohesive and functional blockchain ecosystem. To tackle scalability issues such as network congestion, transaction throughput, and transaction fees – which are interrelated – this study proposes a Scalable Interoperable Decentralized Framework. This framework includes three interoperability procedures i.e. Network Agnostic feature, Polygon Proof of Stake bridge token transfers across multiple chains, and a Swapping application that allows users to swap separate tokens inside the same blockchain.

1.1 Adoption Challenges of De-Fi Applications in Blockchain technology

1.1.1 Expandability

The growth of decentralized applications (dApps) faces a significant hurdle in terms of expandability. During peak usage, traditional blockchains such as Bitcoin and Ethereum experience longer confirmation times and higher prices, resulting in scalability concerns. As dApps gain popularity and attract more users, this challenge intensifies. To tackle scalability problems, it becomes essential to implement solutions like sharding, layer-two protocols, or explore alternative blockchain designs.

1.1.2 User Interaction

The widespread acceptance of dApps relies on delivering a smooth and uncomplicated user interaction. Unfortunately, many dApps feature intricate user interfaces, steep learning curves, and sluggish responsiveness, discouraging non-technical users. Developers must address this challenge by simplifying user interfaces and reducing technical barriers to enhance the overall user experience.

1.1.3 Compatibility

Issues with interoperability in blockchain networks make it difficult for different dApps to engage and communicate with one another. Because of this lack of interoperability, dApps are unable to leverage features and data from other apps or blockchains. Initiatives like as cross-chain protocols and standardisation projects such as the Inter-Blockchain Communication (IBC) protocol are designed to overcome this issue.

1.1.4 Complication

Developing decentralized applications (dApps) poses greater challenges compared to traditional application development. The integration of blockchain technology adds a layer of complexity, requiring developers to grasp fundamentals like distributed consensus, smart contracts, and decentralized data storage. Additionally, navigating the decentralized and trustless environment introduces new challenges related to data privacy, transaction verification, and user authentication. Managing digital asset transactions, often involving the blockchain's native cryptocurrency, further complicates dApp development.

I.1.5 Governance and Agreement

Establishing effective governance structures in decentralized networks poses another challenge. dApps often require decision-making processes for protocol updates, parameter changes, and conflict resolution. Sustainable growth of dApps necessitates robust governance frameworks ensuring inclusivity, fairness, and transparency.

The blockchain community, developers, and businesses are actively examining and resolving these issues. It is anticipated that continuous developments and improvements will get over these obstacles and open the door for the widespread use of decentralised applications. This study presents strategies to address these problems.

1.2 Our Contributions

We contribute the following in this paper:

I.Swapping

Exchanging one cryptocurrency or token for another is known as blockchain swapping. Interoperability facilitates communication and information sharing between several blockchains, enabling users to transfer assets between chains with ease. Interoperability is improved via cross-chain capabilities, which permits transactions between blockchain networks that use various protocols. These ideas are essential to building a transparent and interconnected blockchain ecosystem that gives consumers access to a wide range of assets across many chains while upholding security.

II.Network-Agnostic Feature

We present a novel solution to transcend the constraints of a single blockchain network: network agnosticism. To give customers a cohesive experience, this entails merging various blockchain networks to take advantage of their unique characteristics. We research network agnosticism across the Ethereum and Polygon networks in order to combine Ethereum's security and decentralisation with Polygon's scalability and affordability.

III.Polygon PoS Bridge

Polygon, a layer 2 scaling approach for Crytocurrency, addresses scalability & excessive transaction costs. Our contribution comprises the Polygon PoS Bridge, which serves as a bridge between the Crytocurrency and Polygon networks. This bridge allows for easy asset movement in between two networks while maintaining integrity and security.

II. LITERATURE SURVEY

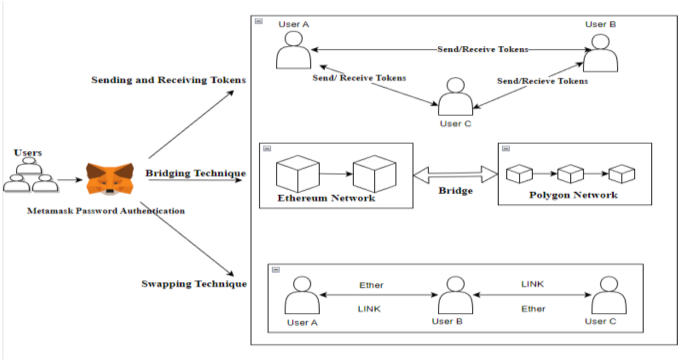
The research paper seeks to explore the enhancement of scalability through various Blockchain interoperability techniques. A comprehensive design for a blockchain De-fi application is introduced, requiring a thorough understanding of existing interoperability solutions, their definitions, and characteristics. They can be broadly divided into three categories by current frameworks: Blockchain of Blockchains, Hybrid Approaches, and Public Connectors. Notary Schemes, Atomic Swaps, and Sidechain/Relays are subcategories of Public Connectors, and Blockchain Agnostic Protocols, Trusted Relays, and Blockchain Migrators are subcategories of Hybrid Approaches [10][11][12].

The study explores many Blockchain scalability review papers in order to construct the suggested framework, extending the first classification to classify all examined publications according to noted qualities. Blockchain interoperability elements are given priority in this new categorization method, which enables a classification based on the features that each solution offers. The advantages of blockchain technology for De-Fi and banking are emphasised; these benefits include improved security, privacy, and efficiency that lowers overhead costs for banks and fosters trust. Nevertheless, difficulties including multiple network connectivity, high processing power requirements, and standards for certifying new blocks are recognised. Through the devaluation of the Interoperable architecture, the study seeks to overcome these issues [13].

The Polygon PoS Bridging technique, which combines a proof-of-stake consensus mechanism with Ethereum and Polygon Networks, is introduced in this study [14][15][16]. Furthermore, to address interoperability concerns between healthcare institutions and guarantee patient privacy, a patient-centric multichain healthcare record (PCMHR) is presented, leveraging the Ethereum blockchain and smart contracts built on the Polygon multichain architecture. Additionally, the Ethereum Network-based decentralised exchange (DEX) Uniswap is examined. A key player in the Decentralized Finance (De-Fi) ecosystem, Uniswap is changing the conventional order book trading model used by centralised exchanges (CEX). Based on liquidity sources, Uniswap, which uses a fixed price mechanism, permits the exchange of cash or tokens/assets. In the decentralised exchange, liquidity providers assume an important role as investors, securing assets in liquidity pools designed for particular currency pairs [17][18].

Therefore, research paper explores diverse aspects of blockchain interoperability, scalability, and applications in De-Fi, providing a comprehensive overview of current solutions and proposing innovative frameworks for future developments.

III. PROPOSED ARCHITECTURE



"Figure 1: Architecture of the Proposed De-Fi Application"

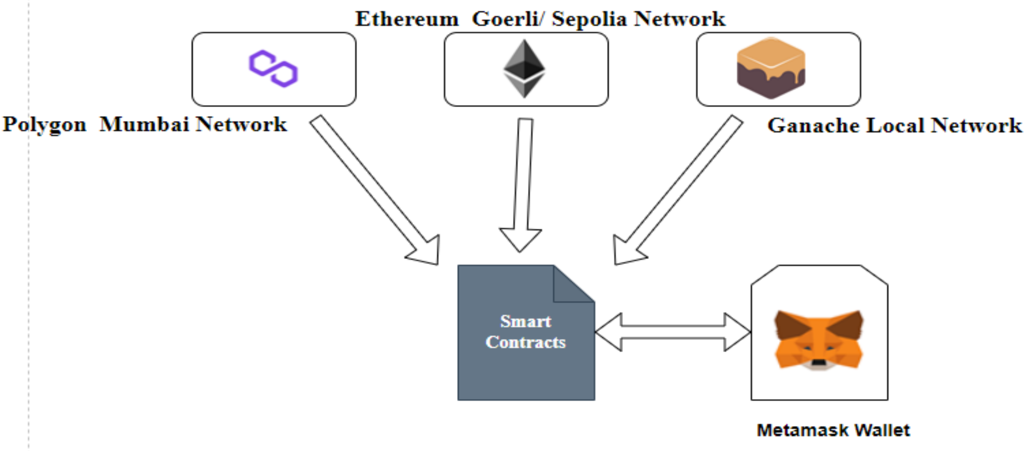
“Figure 1 illustrates the entire architecture of the proposed system, emphasizing its ability to facilitate rapid transactions and cost-effectiveness through the utilization of PoS bridging from Polygon. This architecture goes a step further by incorporating features inspired by Uniswap, offering unparalleled liquidity and decentralized exchange functionalities.”

3.1 Network Agnostic Architecture

The attribute of being network-agnostic implies the capability of a decentralized finance (De-Fi) application (dApp) to operate seamlessly across various blockchain networks. Throughout our inquiry, we utilize Ethereum [19] and the Polygon Blockchain network [20]. This form of network-agnostic dApp is designed to operate on all these blockchain networks without alterations to the smart contracts or supporting code, and without encountering limits specific to any network.

Ethereum, a widely used blockchain platform, leverages smart contract capabilities for decentralized app development and programmable transactions [21]. Ether (ETH), Ethereum's native cryptocurrency, facilitates smart contract execution and serves as a medium of exchange within the Ethereum network. Ethereum offers security, decentralization, and compatibility with existing protocols and smart contracts. Polygon Chain's native cryptocurrency, Matic token (MATIC), serves various functions within the Polygon ecosystem.

In this proposed setup, creating a dApp communicating with both Polygon and Ethereum networks is viable. Users can perform transactions using either Polygon (MATIC) or Ether (ETH) according on their chosen network. To improve the user experience, the dApp may take use of Polygon's fast transaction processing times and cheap costs. Figure 2 depicts the Network Agnostic Model's functioning in a Blockchain Network, demonstrating how the dApp's network-agnostic functionality operates across three various network by flipping networks in Metamask.



“Figure 2: Architecture of Network Agnostic Mechanism”

To achieve network agnostic capabilities, the dApp needs to connect with both the Ethereum and Polygon networks using their respective developer toolkits or APIs. It is essential to develop smart contracts and backend logic for the dApp to smoothly manage interactions and transactions across both networks. Tokens depending on ether & matic behave on their respective networks just like utility tokens. Polygon (MATIC) tokens are used for staking, payment of transaction fees, and participation in the Polygon proof-of-stake consensus process [15]. Ethereum's ecosystem of decentralised apps and protocols is accessible using Ether, which is also used to pay gas fees and carry out smart contracts. An Ethereum and Polygon network-agnostic dApp can achieve interoperability by putting similar smart contract methods into practise.

The network agnostic feature of the dApp allows for the use of both tokens. For example, when using the dApp on Polygon, users can interact with other users on the Ethereum network and pay for transactions with Matic tokens in addition to Ether. The DApp's smart contracts would have to handle the right token transfers and ensure that the features of both networks were seamlessly integrated.

There are several benefits that users of the De-Fi protocol dApp can take advantage of thanks to this network-independent feature. By permitting users on both Polygon and Ethereum, it largely boosts the application's liquidity and adoption potential by expanding the user base. Additionally, it provides users with the freedom to choose the blockchain network that best suits their requirements with regard to transaction fees, speed, and security. It also provides alternatives that lessen the risks associated with depending solely on one blockchain network, like network instability or congestion.

Finally, by leveraging the benefits of both blockchains, a Blockchain De-Fi DApp built with Polygon and Ethereum can operate flawlessly across both networks thanks to a feature that makes it network agnostic. It makes interoperability possible, increases the user base, and gives consumers more options and flexibility in how they use the service.

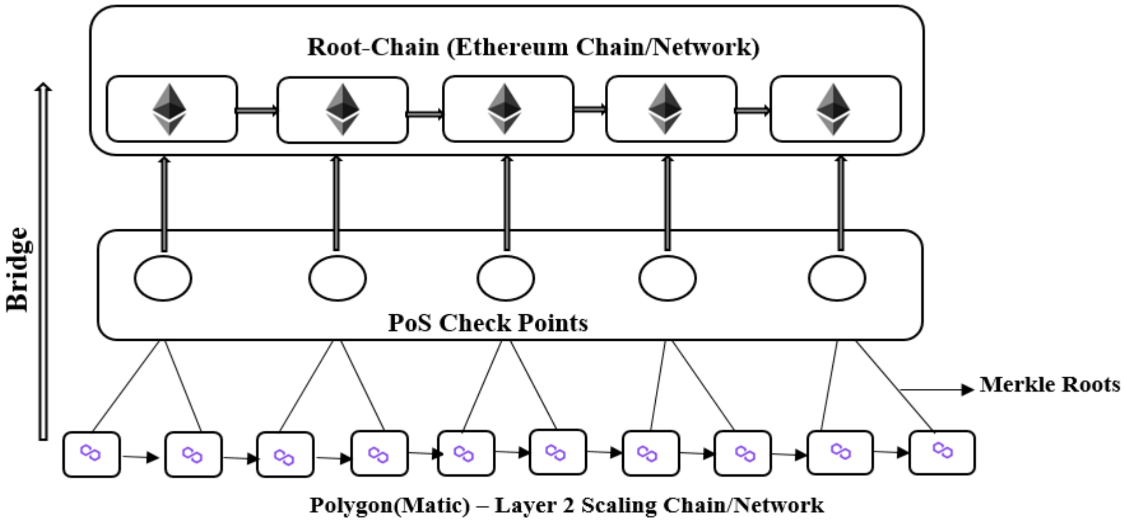
3.2 Polygon PoS Bridge

The Polygon chain, which provides a trustless two-way transaction channel, is seamlessly integrated with the Ethereum chain using our De-Fi DApp. This integration is made feasible by the implementation of a cross-chain bridge proof-of-stake, which allows users to move tokens across polygons without being constrained by market liquidity or third-party risks [22]. This design provides a rapid, inexpensive, and adaptive scaling alternative. Polygon optimises speed and decentralisation by combining Proof-of-Stake with a dual consensus architecture.

It is simple to integrate Polygon with Ethereum within an EVM-capable system. Tokens on the Polygon network are burned using the consensus proof of burn process, which also unlocks them on the Ethereum network. To maintain a 1:1 pegging of the tokens, the Polygon chain allows for arbitrary state transitions on the sidechain, where tokens leaving the Ethereum network are locked. An equal number of tokens are minted on Polygon in response.

This methodology establishes a consensus layer known as dual consensus architecture by utilising the essential components of Proof-of-Stake [23]. Popular token standards such as ERC-20 [24], ERC1155 (a multi-token standard), and ERC721 (a non-fungible token standard) [25] are examples of asset standards that are supported by Proof-of-Stake. Widely acclaimed for its complex architecture, Polygon features a generic validation layer that is separate from the various execution contexts it supports in an elegant architecture. These environments include side chains that are EVM-based, chains that are plasma-enabled, and chains that are compatible with ZK-Rollups and optimistic rollups [26, 27].

To summarise, the adoption of a cross-chain bridge proof-of-stake allows for safe token transfers between polygons while reducing third-party risks and limitations on market liquidity. Using a dual consensus architecture with Proof-of-Stake to maximise speed and decentralisation, this architecture offers an instantaneous, affordable, and adaptable scaling solution. The fact that Polygon's architecture integrates many execution contexts adds to its well-known status in the blockchain industry.


Root-Chain (Ethereum Chain/Network)
Bridge
PoS Check Points
Merkle Roots
Polygon(Matic) – Layer 2 Scaling Chain/Network

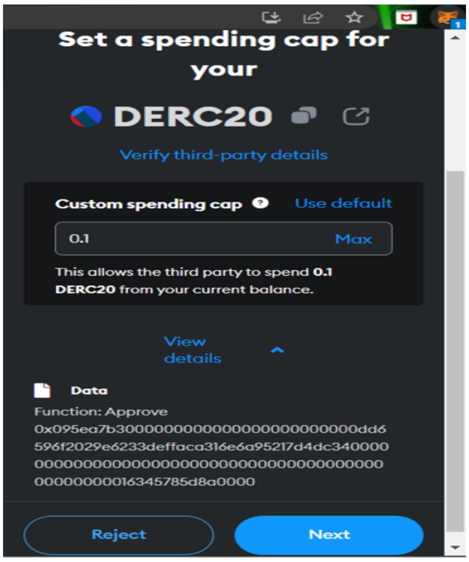
“Figure 3: Architecture of Polygon Proof of Stake Bridging Mechanism”

A bridge is a collection of contracts that allow assets to be transferred from the Roots chain to the child chain. Bridges also allow assets to be transferred between the Ethereum and Polygon chains. Smooth asset transfers between the “Ethereum network” and the “Polygon (formerly Matic) chain” are made possible by the “Polygon PoS (Proof of Stake) Bridge”. Token transfers between the Polygon network and Ethereum are depicted in Figure 3, which offers a thorough overview of the Polygon PoS Bridge's architecture. The following is a breakdown of the bridge operation's sequential flow:

- Deposit on Ethereum: To begin the process, a user must transfer assets (tokens or ETH) to the Ethereum network's PoS Bridge from their Ethereum wallet. To accomplish this, a transaction must be created on the Ethereum network, and the PoS Bridge contract must be designated as the beneficiary.
- Ethereum PoS Bridge: The smart contract of the bridge holds custody of assets transferred to the Ethereum PoS Bridge contract. Asset deposits are tracked by the PoS Bridge, which also creates a corresponding Polygon chain representation.
- Ethereum Confirmation: To complete the deposit transaction and make it irreversible, the PoS Bridge contract on Ethereum needs a certain amount of block confirmations. After obtaining the necessary confirmations, the assets are regarded as deposited and made available on the Polygon chain.

- Asset Minting on Polygon: Following confirmation of an Ethereum deposit, an equivalent representation of assets on the Polygon chain is generated using the Ethereum PoS Bridge contract (asset minting). These freshly created Polygon assets are identical to those being kept on Ethereum.
- Asset Availability on Polygon: When assets are created on Polygon, they can be used throughout the Polygon ecosystem. Users can access and make use of these resources, transacting, interacting with smart contracts, and taking part in decentralised apps (dApps).
- Withrawl on Polygon: To transfer assets back to the Ethereum network, a user must first submit a withdrawal request on the Polygon chain. The PoS Bridge contract's unique function on Polygon is used to submit the withdrawal request.
- Asset Burning on Polygon: The PoS Bridge contract on Polygon confirms the user's ownership of the required amount and makes sure it is available for withdrawal after receiving the withdrawal request. The contract burns (destroys) the corresponding assets on the Polygon chain after successful verification.
- Asset Release on Ethereum: After the assets undergo elimination on the Polygon chain, the PoS Bridge contract on Ethereum frees up the connected assets that were formerly in custody. Once these liberated assets reach the user's Ethereum wallet, they become available for unrestricted transfer or use within the Ethereum ecosystem.

The Proof of Stake Bridge's deposit procedures: The two tokens used in proof of stake bridging are called root and child tokens, and they are mapped on the proof of stake bridge. This means that in order to transfer assets, the token contracts for the root chain and child chain must connect. Consequently, there are two distinct steps involved in the entire cycle of transferring assets from Ethereum to Polygon and back to Ethereum. The ERC20, ERC721, and ERC1155 tokens in our experiment are the owners of the asset. In order for the tokens to be transferred, a smart contract on the proof of stake bridge must be approved. Figures 4 and 5 depict the Bridge transaction's initiation and confirmation.



Set a spending cap for your

DERC20

Verify third-party details

Custom spending cap

Use default

0.1

Max

This allows the third party to spend **0.1 DERC20** from your current balance.

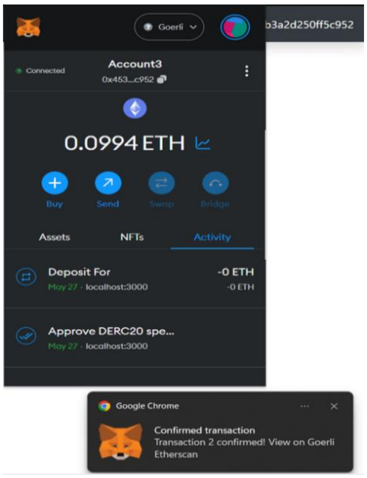
View details

Data

Function: Approve

```
0x095ea7b3000000000000000000000000dd6
596f2029e6233deffaca316e6a95217d4dc340000
000000000000000000000000000000000000
00000000016345785d8a0000
```

“Figure 4 : - Transacting ERC20 in Bridging Method”



Goerli

b3a2d250ff5c952

Connected

Account3

0x453...c952

0.0994 ETH

Deposit For -0 ETH

May 27 · localhost:3000 -0 ETH

Approve DERC20 spe...

May 27 · localhost:3000

Google Chrome

Confirmed transaction

Transaction 2 confirmed! View on Goerli Etherscan

“Figure 5 : - Confirmed Transactions in Metamask through Goreli Network”

The deposit process to the PoS Bridge comprises three essential steps:

1. **Predicate Contract**

- The first step is to use a predicate contract, which is a smart contract launched on the Ethereum Network that acts as a bridge between the two chains.
- This contract establishes the circumstances and criteria for moving tokens from the source chain to the PoS bridge, assuring the process's security and integrity. By getting permission, the predicate contract secures the required quantity of tokens for deposit.

2. **Depositing Asset**

- The depositing asset method allows tokens to be transferred from the parent blockchain to the PoS bridge when the child chain management triggers the transfer.
- Status synchronisation is used in this procedure, in which the current state of tokens on the source chain is captured and sent to the PoS bridge.
- State synchronisation guarantees that tokens on the PoS bridge are accurately represented and accessible for staking or other functions inside the PoS network.
- The mechanics of the state synchronisation procedure may vary, such as establishing a token representation, minting new tokens on the bridge mirroring the source chain, or utilising other ways to synchronise states.

3. **Child Chain Manager**

- After successful asset deposition, the child chain manager assumes control.
- The child chain manager manages the deposited assets on the PoS blockchain's sidechain or child chain.
- Notably, only the child manager has access to the deposit function on the child token contract.
- Once users acquire the tokens, they can promptly transfer them with minimal fees on the Polygon chain.
- The child chain management confirms the deposit on the PoS blockchain and produces a matching representation of the deposited tokens within a child chain, often known as a wrapped or pegged asset.
- The child chain manager guarantees that the token balance within the child chain appropriately reflects the deposited amount on the PoS blockchain.
- Wrapped tokens on the child chain become useable inside the PoS ecosystem, allowing users to engage in activities including as staking, voting, and engaging with decentralised apps (DApps) operating on the PoS blockchain.

Algorithm for “depositERC20”:

1. Set up the “depositERC20” function.

2. Call the “posClientParent” function and assign the returned value to “maticPoSClient”.

3. Calculate the value of “x” by multiplying “inputValue” with “1000000000000000000”.

4. Convert “x” to a string and assign it to “x1”.

5. Call the “approveERC20ForDeposit” function of “maticPoSClient” with parameters “config. posRootERC20” and “x1”, providing the “from” account.

6. Await the completion of the “approveERC20ForDeposit” function.

7. Call the “depositERC20ForUser” function of “maticPoSClient” with parameters “config. posRootERC20”, “account”, and “x1”, providing the “from” account.

8. Await the completion of the “depositERC20ForUser” function.

“Note: The ‘from’ account refers to the sender's account for the transactions.”

Algorithm for Withdrawing assets from PoS Bridge:

To start the withdrawal process, two calls are required. The Polygon chain is where the token must be burned, and the Ethereum chain is where the burned token proof must be posted. The transaction is validated by a Proof-of-Stake validator. Upon adding the transaction to the checkpoints, the {exit} function can be used to submit the proof of burn transaction to the Ethereum chain's rootchain manager contract. When this function is called, the predicate contract—which contains the locked asset tokens that were initially deposited—is triggered and the inclusion of the checkpoint is verified. These tokens will be released by the predicate contract and refunded to the user's Ethereum network account.

Algorithm: Burn ERC20 Tokens

Inputs:

- “inputValue”: The amount of ERC20 tokens to burn.

- “account”: The Ethereum account address initiating the burn.

- “config”: Configuration object containing necessary parameters like “posChildERC20”.

Outputs:

-“burnHash”: The transaction hash generated after burning ERC20 tokens.

1. Start the 'burnERC20' function as follows:

2. Call the “posClientChild” function to obtain an instance of the Matic PoS client. Store it in the “maticPoSClient” variable.

3. Multiply the “inputValue” by 10^18 (to convert it to Wei), store the result in the “x” variable.

4. Convert “x” to a string and store it in the “x1” variable.

5. Use the Matic PoS client (“maticPoSClient”) to burn ERC20 tokens with the following parameters:

- Token address: “config.posChildERC20”

- Amount to burn: “x1”

- Sender address: “account”

6. Await the outcome of the burn procedure.

7. After finishing, get the transaction hashes from the result and save it in the variable "burnHash."

8. Put an end to the function.

Exit Function:

1. Begin the function “exitERC20”.

2. Call the function “posClientParent” and assign its result to the variable “maticPoSClient”.

3. Asynchronously wait for “maticPoSClient” to exit the ERC20 token with the given “inputValue”.

4. Provide additional parameters to the “exitERC20” function, including the “from” parameter set to the value of the “account” variable.

5. Once the “exitERC20” operation is completed, execute the “.then()” method.

6. Inside the “.then()” block, log the message "exit" to the console along with the “res” parameter.

7. Put an end to the function.

Swapping Application:

We have described the Uniswap v3 technique and introduced the Swapping Method, a different sort of Blockchain Interoperability, in this study [29]. A decentralised trade mechanism on the Ethereum blockchain, Uniswap v3 offers several enhancements over its previous iterations, including improved compatibility and better scalability.

Uniswap Preliminaries:

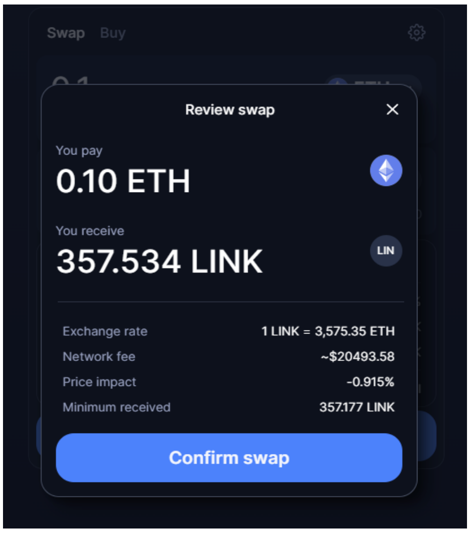
1. Liquidity Pools: Users provide money to Uniswap's liquidity pool system, which establishes a pool for every trading pair [30]. Two distinct tokens of equal value are included in each pool, increasing liquidity and enabling quicker trades.
2. Automated Market Maker (AMM): An essential part of a decentralised exchange protocol, Uniswap makes use of an automated market maker mechanism. Because there is no longer a requirement for an order book, trades are conducted directly against the liquidity pool, resulting in increased liquidity [31].

3. Time-Weighted Average Price, or TWAP, is a trading algorithm that seeks to execute trades at an asset's average price during a given time frame. By uniformly distributing smaller orders throughout the trading session, it lessens the influence of larger orders and guarantees a steady execution price [32].

Liquidity providers (LPs) contribute money to liquidity pools, which are smart contract repositories of different tokens, according to the AMM paradigm. Fees are paid to LPs based on the amount of liquidity they provide. The token reserve ratio is used by the AMM to determine a token's price. Concentrated liquidity is introduced by Uniswap v3, which enables LPs to concentrate money within particular price ranges, improving capital efficiency and protocol effectiveness. This approach promotes compatibility with other decentralised applications (dApps) and protocols and is compatible with the Ethereum Virtual Machine (EVM), which encourages innovation in the field of decentralised finance.

Scalability is enhanced by the concentrated liquidity strategy, which maximises capital efficiency. Interoperability is improved by compatibility with layer 2 solutions and the Ethereum ecosystem, allowing for easy integration with other dApps and protocols in the decentralised finance space. The liquidity in the pool determines scalability and transaction speed. with increased liquidity leading to faster transactions and lower fees. The relationship between liquidity (a) and transaction fees/gas fees (b) is inversely proportional, denoted by the equation ab = k, where k is a constant. Increasing liquidity in the pool reduces transaction fees, emphasizing the importance of attracting more users to enhance efficiency.

“The presented figures (6 and 7) illustrate the initiation and confirmation of a Swapping transaction, showcasing how liquidity is allocated within a custom price range for Uniswap and Eth (Sapolia ETH Token).”

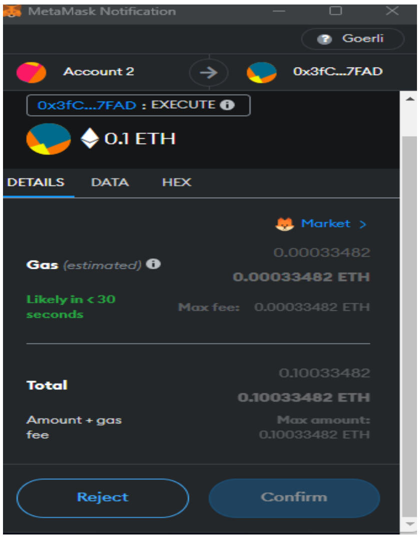


You pay
0.10 ETH

You receive
357.534 LINK

Exchange rate	1 LINK = 3,575.35 ETH
Network fee	~$20493.58
Price impact	-0.915%
Minimum received	357.177 LINK

“Figure 6: Swapping in from ETH – Link”



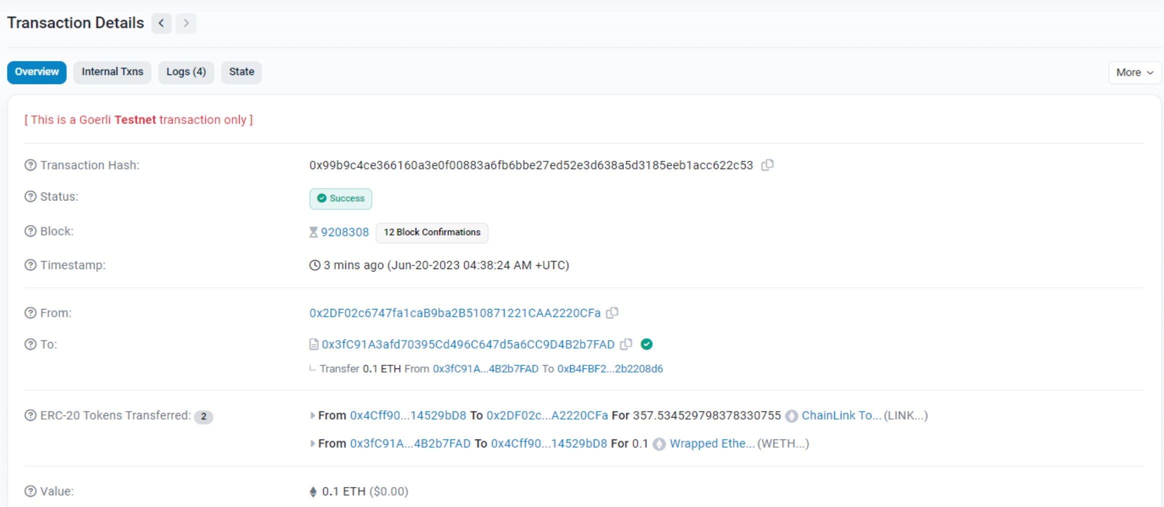
Account 2 → 0x3fC...7FAD

0x3fC...7FAD : EXECUTE

0.1 ETH

Gas *(estimated)*	0.00033482 0.00033482 ETH
Likely in < 30 seconds	Max fee: 0.00033482 ETH
Total	0.10033482 0.10033482 ETH
Amount + gas fee	Max amount: 0.10033482 ETH

“ Figure 7: Confirming Swapping mechanism”



Transaction Details

[This is a Goerli **Testnet** transaction only]

Transaction Hash: 0x99b9c4ce366160a3e0f00883a6fb6bbe27ed52e3d638a5d3185eeb1acc622c53

Status: Success

Block: 9208308 12 Block Confirmations

Timestamp: 3 mins ago (Jun-20-2023 04:38:24 AM +UTC)

From: 0x2DF02c6747fa1caB9ba2B510871221CAA2220CFa

To: 0x3fC91A3afd70395Cd496C647d5a6CC9D4B2b7FAD

└ Transfer 0.1 ETH From 0x3fC91A...4B2b7FAD To 0xB4FBF2...2b2208d6

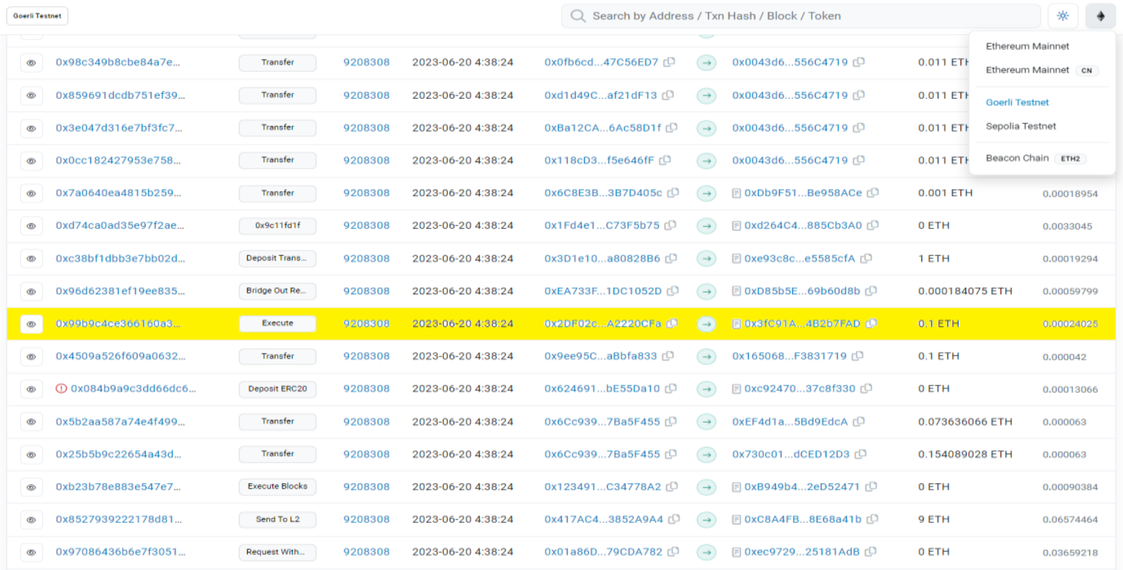
ERC-20 Tokens Transferred: 2

▸ **From** 0x4Cff90...14529bD8 **To** 0x2DF02c...A2220CFa **For** 357.534529798378330755 ChainLink To... (LINK...)

▸ **From** 0x3fC91A...4B2b7FAD **To** 0x4Cff90...14529bD8 **For** 0.1 Wrapped Ethe... (WETH...)

Value: 0.1 ETH ($0.00)

“Figure 8: Confirmed Swapped Transaction”



0x98c349b8cbe84a7e...	Transfer	9208308	2023-06-20 4:38:24	0x0fb6cd...47C56ED7	0x0043d6...556C4719	0.011 ETH	
0x859691dcdb751ef39...	Transfer	9208308	2023-06-20 4:38:24	0xd1d49C...af21dF13	0x0043d6...556C4719	0.011 ETH	
0x3e047d316e7bf3fc7...	Transfer	9208308	2023-06-20 4:38:24	0xBa12CA...6Ac58D1f	0x0043d6...556C4719	0.011 ETH	
0x0cc182427953e758...	Transfer	9208308	2023-06-20 4:38:24	0x118cD3...f5e646fF	0x0043d6...556C4719	0.011 ETH	
0x7a0640ea4815b259...	Transfer	9208308	2023-06-20 4:38:24	0x6C8E3B...3B7D405c	0xDb9F51...Be958ACe	0.001 ETH	0.00018954
0xd74ca0ad35e97f2ae...	0x9c11fd1f	9208308	2023-06-20 4:38:24	0x1Fd4e1...C73F5b75	0xd264C4...885Cb3A0	0 ETH	0.0033045
0xc38bf1dbb3e7bb02d...	Deposit Trans...	9208308	2023-06-20 4:38:24	0x3D1e10...a80828B6	0xe93c8c...e5585cfA	1 ETH	0.00019294
0x96d62381ef19ee835...	Bridge Out Re...	9208308	2023-06-20 4:38:24	0xEA733F...1DC1052D	0xD85b5E...69b60d8b	0.000184075 ETH	0.00059799
0x99b9c4ce366160a3...	Execute	9208308	2023-06-20 4:38:24	0x2DF02c...A2220CFa	0x3fC91A...4B2b7FAD	0.1 ETH	0.00024025
0x4509a526f609a0632...	Transfer	9208308	2023-06-20 4:38:24	0x9ee95C...aBbfa833	0x165068...F3831719	0.1 ETH	0.000042
ⓘ 0x084b9a9c3dd66dc6...	Deposit ERC20	9208308	2023-06-20 4:38:24	0x624691...bE55Da10	0xc92470...37c8f330	0 ETH	0.00013066
0x5b2aa587a74e4f499...	Transfer	9208308	2023-06-20 4:38:24	0x6Cc939...7Ba5F455	0xEF4d1a...5Bd9EdcA	0.073636066 ETH	0.000063
0x25b5b9c22654a43d...	Transfer	9208308	2023-06-20 4:38:24	0x6Cc939...7Ba5F455	0x730c01...dCED12D3	0.154089028 ETH	0.000063
0xb23b78e883e547e7...	Execute Blocks	9208308	2023-06-20 4:38:24	0x123491...C34778A2	0xB949b4...2eD52471	0 ETH	0.00090384
0x8527939222178d81...	Send To L2	9208308	2023-06-20 4:38:24	0x417AC4...3852A9A4	0xC8A4FB...8E68a41b	9 ETH	0.06574464
0x97086436b6e7f3051...	Request With...	9208308	2023-06-20 4:38:24	0x01a86D...79CDA782	0xec9729...25181AdB	0 ETH	0.03659218

“Figure 9: Swapped Transaction in the Block”

The final transaction is depicted in Figure 9, presented within the Block. Figure 8's output from Etherscan showcases the transacted transactions, revealing all three accounts involved in the token transfer.

Given that Uniswap's scalability relies heavily on liquidity, we exemplify the swapping of Ether and ChainLink in our experiment.

This paper has delved into how Time-Weighted Average Price (TWAP) contributes to scalability.

Scalability and TWAP:

Uniswap v3's scalability is indirectly facilitated by the TWAP computations, which promote effective capital allocation and reduce the frequency of rebalancing. Liquidity providers (LPs) attain a more equitable distribution of liquidity throughout the market by deliberately placing their funds in price ranges determined by TWAP research.

With little price slippage, Uniswap v3 can manage higher trade volumes thanks to its well distributed liquidity. “Given that institutional traders and larger market participants demand deep liquidity and little price effect for their deals, this scalability benefit is especially important in luring them in.”

“For traders and liquidity providers, TWAP calculations in Uniswap v3 are an invaluable tool that helps them execute deals more quickly, maximise liquidity provision, and contribute to the platform's scalability.”

T – Tick

A and b are the tokens

P – Current price

“In Uniswap V3 the average price of token b over token a from time t_k to t_n is calculated by taking $P_{avg} = 1.0001^{Tick(avg)}$ where the tick average is equal to the timeweighted Average of each tick per second”

$$\text{Tick}_{\text{avg}} = T_{\text{avg}} = \frac{\sum_{i=k}^{n-1} T_i(t_{(i+1)} - t_i)}{t_n - t_k} \text{ [33]}$$

“P_i = Price of token b in terms of token a at time i . In Uniswap the price is represented as 1.0001^{Tick_i} ,However this tick is used to compute the current price is getting tracked in the mechanism

And we have taken virtual reserve concept to derive Timeweighted Average price by dividing the amount of tokens in the liqudity pool”

$\frac{y_i}{x_i}$ = y_i = “Virtual reserve of Token b at time i and x_i is the Virtual reserve of Token a at time i.”

“In Uinswap these virtual reserves are not tracked there is no variables inside the contract thet stores yi and xi instead uniswap tracks the price and liqudity i.e. Pi, By presenting example of Link/ETH that we transacted while experimenting.”

Link/ETH = 3575/1ETH

“Where 3575 Links is equals to 1Ether, In the experiment we have swapped 0.1 ether which is equal to 357 LINKS. P = 3575 (357.5310) = 1.0001^{81822} by Computing the tick for each price at

$P_{k =} 1.0001^{Tk}$, $P_{k+1} = 1.0001^{tk+1}$ ….$P_{k-1=}1.0001^{n-1}$”

“By Calculating Time weighted Average price using the experimented results and gathered parameters.”

$$\text{Tick}_{\text{avg}} = T_{\text{avg}} = \frac{\sum_{i=k}^{n-1} T_i(t_{(i+1)} - t_i)}{t_n - t_k} \qquad [33]$$

$$T_{avg} = \frac{S_T(n) - S_T(k)}{t_n - t_k}$$

“Once We can get the Time weighted average Tick, we can get the average price by raising $1.0001^{T_{avg}}$”

t	Price	Tick
0	3575	81822
5	3000	80067
9	1500	73135
10	3575	81822

“Table 1 : - List of Twap price and Tick”

$P_{avg} = 1.0001^{T(avg)}$

“We have calculated If t=0, price=3575 and the tick = 81822”

$T_{\text{avg}} = \frac{81822(5-0)+80067(9-5)+73135(10-9)}{10-0} = 802513 = 1.0001^{80251-"}3055$ is the Average

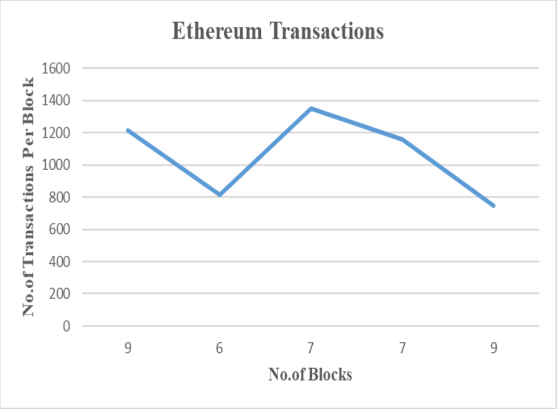
“This Average may varies according to the market price of the tokens”

1. Experimental setup and Results

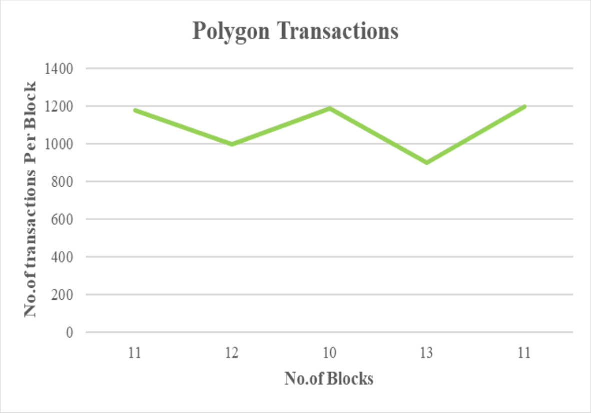
In our study, we introduced a new De-Fi concept on blockchain that accommodates various interoperability approaches within a unified dApp. To demonstrate this, we developed a prototype on the public blockchain, merging Ethereum and Polygon networks. Our tests were carried out on Ethereum's Goreli test net and Polygon Matic's Mumbai test net, concentrating on block time and transactions per second.

In the absence of a specialized application for accurately gauging transaction speed on the Polygon network, we executed around 5000 transactions across 150 transaction blocks for our experiments. Across various blocks of both Polygon and Ethereum testnets, we transacted around 300 transactions and recorded the results. The average block time for Ethereum was observed to be around 10-12 seconds, while Polygon exhibited a faster block time ranging from 2-4 seconds.

“Figures 10 and 11 visually represent the transaction processes in Ethereum and Polygon, respectively. The graphs clearly illustrate that Polygon provides better and more effective results compared to Ethereum.”



“Figure 10 Analyzing Ethereum Transactions”



“Figure 11 Analyzing Polygon Transactions”

We carried out more than 300 transactions in our test setup, all of which were painstakingly documented by smart contract terms and bundled into Blockchain blocks. Using the Ethereum and Polygon Blockchain platforms for the Network Agnostic functionality, the dApp was tested on the testRPC platform. Polygon Mumbai and Ethereum Goreli testnets were used to evaluate this feature. The MetaMask Cryptocurrency wallet was used to distribute system activities over the Blockchain network, guaranteeing the safety and security of tokens.

The second interoperability approach in our concept, the Bridging mechanism, was evaluated using Ethereum Goerli testnet variations, including ERC20, ERC721, and ERC1155, which are tokens that support proof of stake. As for the third interoperability approach in our concept, Uniswap, transactions were carried out with both the Goreli and Uniswap testnets for the swapping mechanism. The web3.js library and the Solidity programming language were utilised to create smart contracts, and the Hardhat development environment was used to support the Ethereum and Polygon networks. Front-end work made use of React.js, HTML, and CSS. The two bridges were able to interact more easily thanks to the matic.js package.

In addition, we examined the 3329 Polygon transactions for deposits and withdrawals that occurred during 27 distinct blocks. Apart from the transactions, we also looked at the block rewards for every block. To do this, we divided the 27 blocks into 9 blocks, using each 9 blocks as an epoch. Every epoch's transaction, including money deposits and withdrawals, were tracked, and examined, and block rewards were also examined. The number of transactions per block and the block rewards that nodes receive are shown in Tables 2, 3, and 4. 1235, 746, and 1348 transactions were reported for each epoch, with an average of 137.22, 82.88, 149.77, and 1.09, 0.96, and 1.8 Matic tokens awarded to validators as block rewards for each period, respectively.

Blocks	Number of Transactions	Block Rewards
1	75	0.70
2	68	0.61
3	175	0.89
4	260	1.21
5	110	1.24
6	58	0.55
7	65	0.56
8	77	0.66
9	347	3.18

“Table 2:- Epoch 1 of Polygon Transactions and Block rewards”

Blocks	Number of Transactions	Block Rewards
1	90	0.75
2	92	1.30
3	49	0.50
4	87	0.89
5	142	1.45

6	84	0.72
7	105	0.94
8	103	0.95
9	596	8.7

“Table 3:- Epoch 2 of Polygon Transactions and Block rewards”

Blocks	**Number Transactions**	**Block Rewards**
1	140	1.18
2	81	0.72
3	71	0.80
4	62	1.53
5	54	0.96
6	110	0.92
7	75	0.95
8	68	0.70
9	85	0.89

“Table 4: Epoch 3 of Polygon transactions and Block rewards”

In the three tables, we highlighted transactions and block rewards, enabling a comparison that reveals changes in rewards and the number of transactions in each block. “For instance, in Epoch 1 (Table 2), the 4th block consists of 260 transactions with 1.21 rewards, whereas in Epoch 3 (Table 4), the 1st block recorded 140 transactions and received 1.18 rewards. There are noticeable variations in both the number of transactions and block rewards for successful block creations with wrapped transactions”.

The changes in block rewards on the Polygon network can arise from various factors:

1. Network Activity: Block rewards may vary depending on the overall network activity. Increased transaction volume can lead to heightened competition among validators for block rewards, potentially resulting in decreased individual rewards.

2. Validator Participation: The number of active validators is a key determinant of block rewards. With fewer validators participating, individual rewards may be higher, while a larger validator pool could lead to rewards being distributed among a more extensive group.

3. Epoch Changes: PoS networks, including Polygon, operate in epochs—defined periods where validator sets and rewards are determined. After each epoch, alterations in the validator set based on factors like stake size or performance can cause fluctuations in individual rewards.

4. Slashing and Rewards Penalties: Validators may face penalties for malicious actions or downtime. Any violation of network rules or failure to fulfill duties could result in penalties, ultimately reducing the rewards for the affected validators.

5. Inflation Rate: The block rewards and overall supply dynamics of the cryptocurrency play a role. If the inflation rate of the network changes (e.g., due to protocol upgrades), it can affect the amount of new tokens minted as rewards and distributed to validators.

6. Protocol Upgrades: Changes to the network's protocol or economic model can lead to variations in block rewards. Upgrades that modify the reward distribution mechanism or validator selection process can impact how rewards are allocated.

Blockchain networks are complex systems influenced by various parameters and participants' actions. As a result, rewards on PoS networks like Polygon can fluctuate over time, depending on the network's activity, validator participation, and other dynamic factors. For the most accurate and up-to-date information about Polygon's rewards, it's recommended to refer to the official documentation or community resources.

IV. CONCLUSION

In conclusion, addressing the scalability challenges in blockchain networks is paramount for their widespread adoption and efficient functioning. One promising approach to overcome this challenge involves implementing an interoperable and network-agnostic framework. Combining this framework with Polygon's proof-of-stake bridge and enhancing the scalability of platforms like Uniswap offers a robust solution to scalability issues. The integration of the Interoperable Network Agnostic feature, Polygon Proof of Stake bridge, and improved Uniswap scalability forms a powerful ecosystem that not only boosts the performance and efficiency of decentralized applications but also encourages broader adoption through a seamless and user-friendly experience. Overcoming scalability hurdles is a significant stride towards realizing the full potential of blockchain technology, unlocking new possibilities across diverse industries. Although scalability issues have posed challenges for widespread adoption and extended use cases beyond basic token transfers, ongoing efforts in interoperability methods such as network-agnostic features, Polygon PoS bridging, and Uniswap v3 showcase innovative solutions in the blockchain space. As technology evolves, we anticipate further progress in scalability and interoperability, paving the way for a more seamless and efficient decentralized ecosystem.

REFERENCES

[1] Nakamoto, S., 2009. Bitcoin: A Peer-to-Peer Electronic Cash System. [online] Bitcoin.org https://bitcoin.org/bitcoin.pdf

[2] Ethereum Whitepaper | ethereum.org. (n.d.). In ethereum.org. https://ethereum.org

[3] GAVIN WOOD, "Ethereum: A secure decentralised generalised transaction ledger", Ethereum project yellow paper, vol. 151, no. 2014, pp. 1-32, 2014.

[4] Ramos, D., & Zanko, G. (2020). A review of decentralized finance as an application of increasing importance of blockchain technology. Mobileyour Life.

[5] MakerDAO (2017), "The Maker Protocol: MakerDAO's Multi-Collateral Dai (MCD) System". Retrieved from https://makerdao.com/en/whitepaper/

[6] Compound, 2019, [online] Available: https://compound.finance

[7] Aave, 2020, [online] Available: https://aave.com/.

[8] Jensen, Johannes Rude, Victor von Wachter, and Omri Ross. "An introduction to decentralized finance (defi)." Complex Systems Informatics and Modeling Quarterly 26 (2021): 46-54.

[9] Pillai, B., Biswas, K., & Muthukkumarasamy, V. (2020). Cross-chain interoperability among blockchain-based systems using transactions. The Knowledge Engineering Review, 35, e23.

[10] Gorkhali, Anjee, Ling Li, and Asim Shrestha. "Blockchain: A literature review." Journal of Management Analytics 7.3 (2020): 321-343.

[11] [11] Zhou, Qiheng, et al. "Solutions to scalability of blockchain: A survey." Ieee Access 8 (2020): 16440-16455.

[12] Sanka, Abdurrashid Ibrahim, and Ray CC Cheung. "A systematic review of blockchain scalability: Issues, solutions, analysis and future research." Journal of Network and Computer Applications 195 (2021): 103232.

[13] Ozcan, R. (2021). Decentralized Finance. In: Hacioglu, U., Aksoy, T. (eds) Financial Ecosystem and Strategy in the Digital Era. Contributions to Finance and Accounting. Springer, Cham. https://doi.org/10.1007/978-3-030-72624-9_4

[14] Kapengut, Elie, and Bruce Mizrach. "An event study of the ethereum transition to proof-of-stake." Commodities 2.2 (2023): 96-110.

[15] Ethereum↔Polygon PoS Bridge | Polygon Wiki. (2023, August 6). Ethereum↔Polygon PoS Bridge | Polygon Wiki. https://wiki.polygon.technology/docs/pos/design/bridge/ethereum-polygon/getting-started

[16] Rai, Bipin Kumar, Sumrah Fatima, and Kumar Satyarth. "Patient-centric multichain healthcare record." International Journal of E-Health and Medical Communications (IJEHMC) 13.4 (2022): 1-14.

[17] Lo, Yuen C., and Francesca Medda. "Uniswap and the Emergence of the Decentralized Exchange." Journal of Financial Market Infrastructures 10.2 (2021): 1-25.

[18] Aigner, Andreas A., and Gurvinder Dhaliwal. "Uniswap: Impermanent loss and risk profile of a liquidity provider." arXiv preprint arXiv:2106.14404 (2021).

[19] Wood, Gavin. "Ethereum: A secure decentralised generalised transaction ledger." Ethereum project yellow paper 151.2014 (2014): 1-32.

[20] Kanani, Jaynti, Sandeep Nailwal, and Anurag Arjun. "Matic whitepaper." Polygon, Bengaluru, India, Tech. Rep., Sep (2021).

[21] Dannen, Chris. Introducing Ethereum and solidity. Vol. 1. Berkeley: Apress, 2017.

[22] Architecture Overview | Polygon Wiki. (2023, August 6). Architecture Overview | Polygon Wiki. https://wiki.polygon.technology/docs/pos/polygon-architecture

[23] Responsibilities | Polygon Wiki. (2023, August 6). Responsibilities | Polygon Wiki. https://wiki.polygon.technology/docs/pos/design/validator/responsibilities

[24] Cuffe, Paul. "The role of the erc-20 token standard in a financial revolution: the case of initial coin offerings." (2018).

[25] Bauer, Davi Pedro. "Erc-721 nonfungible tokens." Getting Started with Ethereum: A Step-by-Step Guide to Becoming a Blockchain Developer. Berkeley, CA: Apress, 2022. 55-74.

[26] Poon, Joseph, and Vitalik Buterin. "Plasma: Scalable autonomous smart contracts." White paper (2017): 1-47.

[27] Sigwart, Marten, et al. "Decentralized cross-blockchain asset transfers." 2021 Third International Conference on Blockchain Computing and Applications (BCCA). IEEE, 2021.

[28] Thibault, Louis Tremblay, Tom Sarry, and Abdelhakim Senhaji Hafid. "Blockchain scaling using rollups: A comprehensive survey." IEEE Access (2022).

[29] Adams, Hayden, et al. "Uniswap v3 core." Tech. rep., Uniswap, Tech. Rep. (2021).

[30] Neuder, Michael, et al. "Strategic liquidity provision in uniswap v3." arXiv preprint arXiv:2106.12033 (2021).

[31] Loesch, Stefan, et al. "Impermanent loss in uniswap v3." arXiv preprint arXiv:2111.09192 (2021).

[32] Adams, Austin, Xin Wan, and Noah Zinsmeister. "Uniswap v3 TWAP Oracles in Proof of Stake." Available at SSRN 4384409 (2022).

[33] S. (n.d.). GitHub - stakewithus/notes. GitHub. https://github.com/stakewithus/notes