Provenance Tracking of In-game Virtual Items via the Blockchain

Yiang Lu

Supervisors: Dr. Rich Davison

Prof. Graham Morgan

Dr. Gary Ushaw

School of Computing

Newcastle University

This dissertation is submitted for the degree of

Doctor of Philosophy

Aug 2023

Abstract

This thesis explores the integration of blockchain technology within the gaming industry, with a specific focus on in-game virtual item trading. Through a comprehensive examination of various consensus algorithms, the research identifies optimal solutions that enhance latency and throughput in real-time game trading. A specialized communication model for blockchain-based streamed gaming is developed, offering new insights into transaction delays and enriching our understanding of real-time trading dynamics within gaming environments. A significant contribution is the development of a mathematical model designed to predict system performance and resource allocation, holding substantial potential for enhancing system optimization within the gaming industry. The thesis also delves into the delivery of video games through cloud and edge-based technologies, revolutionizing the gaming industry by enabling players to access and play games on various devices. Experimental aspects are detailed, including in-game trading transactions, communication models, and simulator design, leading to comprehensive results and analysis. The concluding chapter synthesizes the findings, emphasizing the integration of blockchain technology in the gaming industry, the development of a mathematical model, and the potential applications in creating secure platforms for trading in-game assets. Future work includes the exciting avenue of developing a lighting network simulation for cloud game trading. This research not only advances the theoretical understanding of blockchain technology in gaming but also provides practical models and tools that can guide future research and development in this field.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Yiang Lu
Aug 2023

Acknowledgements

I would like to thank my supervisor, Dr. Rich Davison, Prof. Graham Morgan, Dr. Gary Ushaw, for the guidance he has provided me during my research.

I would like to thank my parents, Junfeng Lu and Ziya Chen, for their tremendous support throughout the research process.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

Cryptocurrencies have garnered significant attention and adoption due to their unique properties, which have propelled them into the realms of investment and monetary transactions. The secure and finite nature of cryptocurrencies has made them an appealing choice for investors, who see them as a potentially lucrative asset class traded openly on public markets. Simultaneously, individuals have embraced cryptocurrencies as a means to transfer money securely and efficiently, bypassing the traditional banking system. The added allure of anonymity in monetary transactions outside the digital banking system has further fueled the popularity of cryptocurrencies.

Various technologies related to cryptocurrencies have emerged, including the application of Bitcoin in trade (Böhme et al., 2015), the utilization of blockchain for provenance tracking in Internet of Things (IoT) data collection (Sigwart et al., 2019), the implementation of blockchain for patient-driven record maintenance in the healthcare industry (Harshini et al., 2019), and the adoption of blockchain for supply chain provenance tracking (Montecchi et al., 2019). Blockchain technology has also found applications in the gaming industry, such as improving data exchange in video games (Besancon et al., 2019) and enabling cryptocurrency-based payment for cloud gaming services (Zhao et al., 2020).

Performance evaluation metrics, as outlined by Foytik (2020) (Foytik et al., 2020), include throughput, latency, fault tolerance, and heterogeneity. When considering the utilization of blockchain systems in real-time game trading, latency emerges as the most critical evaluation metric. The network resource requirements also impose limitations if cloud game service providers aim to involve users in the distributed ledger generation process.

This project aims to explore the integration of blockchain technology, cryptocurrency transactions, and public ledger (Besancon et al., 2019) recording in streamed gaming to facilitate content evolution and third-party game asset trading in online gaming environments. Simultaneously, the project seeks to find a balance between blockchain performance and system resource usage through the development of a mathematical model.

Blockchain technology plays a crucial role in this setting by enabling users to exchange third-party content in online games. During these transactions, the system meticulously tracks the sales and past ownership transactions and assigns these to in-game assets. This functionality leads to the creation of a trusted marketplace that gives gamers a secure platform for trading. The innovation of non-fungible tokens (NFTs) and smart contracts can be further leveraged to trace the origin and history of third-party in-game assets, providing additional assurance to gamers about the legitimacy and value of the assets they acquire (Chohan, 2021) (Regner et al., 2019).

1.1. In-Game Virtual Item Trading

As the video game industry grows, the trade of in-game virtual goods has blossomed into a thriving independent economy. These digital commodities, spanning from cosmetic alterations to functionality-enhancing gear, serve as vital components of a player's persona and advancement in the game world. The exchange of such elements has spawned large-scale digital bazaars where individuals purchase, peddle, and swap these assets. Although this marketplace boasts considerable potential, it is not without its shortcomings, including issues such as fraudulent items, lack of clear information, and contention over proprietorship. With the escalating importance and value within these virtual economies, there is an emergent demand for a framework that guarantees security, trustworthiness, and openness.

1.2. Blockchain

Enter blockchain, a technology that has been hailed as a game-changer in numerous sectors for its revolutionary approach to secure data management and transaction handling. A blockchain is a decentralized, distributed digital ledger that records transactions across many computers in such a way that the recorded entries cannot be altered retroactively. This makes it a perfect candidate for tracking and verifying the provenance of virtual items in the gaming world. Its immutable, transparent, and decentralized nature provides a robust solution to the problems plaguing the digital item marketplace.

This research delves into the integration of blockchain technology into the gaming industry to create a reliable and transparent environment for the trading of in-game virtual items. By exploring the potential symbiosis between streamed gaming, virtual item trading, and blockchain technology, we aspire to push the boundaries of the gaming industry towards a safer and more reliable future.

1.3. Streamed Gaming

In the rapidly growing world of digital entertainment, streamed gaming has emerged as a dynamic frontier. Offering a powerful platform where games are hosted on remote servers and delivered to players over the internet, this paradigm shift has democratized gaming by allowing access to high-quality experiences regardless of the user's hardware capabilities. Streaming technology transcends traditional boundaries, enabling players from across the globe to engage in their favourite games virtually anywhere and at any time. However, as we embrace the conveniences of streamed gaming, we are faced with new challenges, such as verifying the authenticity and ownership of in-game virtual items, which forms the focus of this research.

1.4. Challenges

The challenge for streamed gaming is the input feedback delay and network bandwidth requirements. In the context of online gaming, where players interact in real time, even the slightest delay in receiving and responding to input commands can significantly impact the gaming experience. This input feedback delay, often referred to as latency, is a crucial concern that needs to be addressed when integrating blockchain technology into streamed gaming environments.

Latency in streamed gaming is primarily influenced by the time it takes for data to travel between the user's device and the game server. When blockchain technology is introduced into the mix, additional factors come into play. Blockchain systems require consensus among network participants before a transaction can be confirmed and recorded on the ledger. This consensus process inherently introduces an additional delay, as the transaction must be validated by multiple nodes on the network.

Moreover, the bandwidth requirements of blockchain systems also pose a challenge for streamed gaming. Blockchain networks consist of multiple nodes that collectively maintain and update the distributed ledger. Each transaction and associated data must be propagated to all nodes in the network, consuming network bandwidth. In the context of streamed gaming, where a large number of transactions occur within a short timeframe, the network must have sufficient bandwidth and buffer to handle the increased data flow.

To mitigate these challenges, careful consideration must be given to optimizing the blockchain system's performance while minimizing the impact on latency and network bandwidth.

1.5. Thesis Contribution

This thesis contributes significantly to the integration of blockchain technology into streamed gaming environments, with a primary focus on a novel mathematical model designed to predict system performance and resource allocation. The major contributions of this research are outlined below:

1.5.1. Comparison of Different Consensus Algorithms in Streamed Gaming

The thesis investigates various consensus algorithms and their suitability for use in streamed gaming scenarios. Consensus algorithms play a crucial role in achieving agreement among network participants regarding the validity and order of transactions recorded on the blockchain. By studying and evaluating different consensus algorithms, this research aims to identify the most suitable algorithm that minimizes latency and maximizes throughput for real-time game trading. The findings from this study provide valuable insights into the selection and implementation of consensus algorithms tailored to the specific requirements of streamed gaming environments.

1.5.2. Communication Model for Streamed Gaming with Blockchain

Another contribution of this thesis is the development of a communication model specifically designed for streamed gaming environments integrated with blockchain technology. The communication model addresses the challenges of latency and network bandwidth requirements, considering the unique characteristics and demands of real-time gaming interactions. By proposing an optimized communication model, this research aims to enhance the efficiency and responsiveness of data transmission between players, game servers, and blockchain networks. The communication model serves as a framework for designing and implementing communication protocols that minimize delays and maximize the overall gaming experience.

1.5.3. Mathematical Model for Resource and Performance Prediction

The main contribution of this thesis is a mathematical model that enables simulation and prediction of the performance of blockchain-based streamed gaming trading systems. Utilizing mathematical modelling techniques, this research offers an in-depth understanding of the correlation between system parameters, such as block size, network bandwidth, and transaction pending queue size, and their influence on the overall performance, such as transaction delay and transaction handling rate of the gaming environment.

This model presents a unique tool based on FobSim (Baniata and Kertesz, 2021) for analyzing different scenarios and configurations, facilitating informed decision-making processes in optimizing blockchain performance while accommodating the specific demands of streamed gaming. Given its potential to significantly enhance system optimization and design processes, this mathematical model is positioned as the main contribution of the thesis.

1.6. Thesis Outline

This thesis is organized into five chapters. Chapter 2 delves into the delivery of video games through cloud and edge-based technologies, revolutionizing the gaming industry. Chapter 3 focuses on the experimental aspects, detailing in-game trading transactions, communication models, and simulator design. Chapter 4 presents the results and analysis of the experiments, including mathematical modelling related to transaction injection rate limits and average block time. Chapter 6 offers the final conclusions of this thesis and explores potential avenues for future research.

Chapter 2. Background

2.1. Streamed Gaming

Streamed gaming refers to the delivery of video games to players through the use of cloud and edge-based technologies. It has revolutionized the gaming industry by enabling players to access and play games on a wider variety of devices, regardless of their processing power and hardware capabilities. By leveraging the centralized resources of cloud computing and the low-latency capabilities of edge technologies, streamed gaming has opened up new possibilities for immersive and interactive gaming experiences.

Traditionally, playing video games required costly consoles or high-performance PCs with significant hardware requirements. Gamers had to invest in expensive equipment, including powerful graphics processing units (GPUs), to achieve the desired gaming experience. However, streamed gaming has disrupted this model by shifting the computational burden from the player's device to remote servers in the cloud.

With streamed gaming, the game is executed and processed on powerful cloud servers, while the player's device acts as a client that receives the game's video and audio streams. This approach eliminates the need for expensive hardware upgrades and allows players to access the latest games on devices with lower specifications, such as smartphones, tablets, or lightweight laptops. Centralizing computational resources in the cloud enables games to be streamed and played remotely, reducing the reliance on local processing power.

One of the critical advantages of streamed gaming is its ability to overcome the limitations of local hardware. By leveraging the computing power of cloud servers, games can be rendered at high resolutions and frame rates, delivering visually stunning graphics and smooth gameplay even on low-powered devices. Players can enjoy graphically demanding games without investing in expensive gaming hardware.

Furthermore, streamed gaming offers the convenience of on-demand access to a vast library of games. Instead of purchasing physical copies or downloading large game files, players can instantly stream games over the Internet, eliminating the need for installation and reducing storage requirements on their devices. This model provides gamers with greater flexibility and the freedom to explore a wide range of gaming experiences without the limitations of physical media or local storage capacity.

The low-latency capabilities of edge technologies are crucial for enabling real-time interactions and multiplayer gaming experiences in streamed gaming. By leveraging edge servers located closer to the players' geographic locations, the network latency is reduced, resulting in minimal delays between player actions and their effects in the game world. This low-latency infrastructure is essential for delivering fast-paced and responsive gameplay, especially in competitive multiplayer games.

However, despite its numerous benefits, streamed gaming also faces challenges. The reliance on internet connectivity and network stability is paramount for a smooth gaming experience. Any disruptions in the internet connection can result in input delays, visual artefacts, or even game disconnections. Streamed gaming heavily relies on the data centres' infrastructure and the capacity to handle high volumes of concurrent players. Insufficient server resources or network congestion can lead to degraded performance and increased latency, affecting the gaming experience.

2.1.1. Communication Model

Centralized

In a centralized communication model for streamed gaming, the game server is the central hub for all player interactions and data processing. All communication between the client devices and the game server flows through a main point, a data centre or a cloud-based infrastructure.

Under the centralized model, the game server handles game state management, rendering graphics, and audio processing tasks. It receives player input from the client devices, updates the game state, and streams the video and audio content back to the clients. This centralization of resources allows for efficient processing, as the server can leverage powerful hardware and computing capabilities to deliver high-quality gameplay.

The centralized communication model offers several advantages. First, it offloads the resource-intensive tasks from the client devices, enabling players to enjoy graphically demanding games even on low-powered devices. By leveraging the server's processing power, players can experience immersive gaming without the need for expensive gaming hardware.

Furthermore, the centralized model simplifies game updates and maintenance. Game developers can easily deploy updates or bug fixes on the server side, eliminating the need for individual client updates. This centralized control ensures that all players have access to the latest version of the game, fostering a consistent and seamless gaming experience.

However, the centralized communication model also presents challenges. One significant concern is the potential impact of network latency on gameplay. Since all player inputs and video/audio streams pass through the game server, any delays in network transmission can result in input lag or visual artefacts. This latency can degrade the real-time nature of the game, impacting player responsiveness and immersion.

Additionally, the centralized model relies heavily on the stability and scalability of the game server infrastructure. If the server experiences downtime or performance issues, it can disrupt gameplay for all connected players. Adequate server capacity and network bandwidth are essential to accommodate the demands of a large player base and ensure smooth and uninterrupted gameplay.

Distributed

In a distributed communication model for streamed gaming, the traditional centralized architecture is decentralized, distributing the game processing and communication tasks across multiple nodes or servers. This approach aims to improve scalability, reduce latency, and enhance the overall performance of the streamed gaming experience.

Under the distributed model, game servers are strategically located in different geographical regions or data centres, referred to as edge nodes. These edge nodes are positioned closer to the players, reducing network latency and enabling real-time interaction in multiplayer games. By dispersing the processing load across multiple servers, the distributed model can handle a larger number of players and provide a more responsive gaming experience.

In a distributed communication model, player input is sent to the nearest edge node rather than a central game server. The edge node processes the input, updates the game state, and streams the video and audio content back to the player. This distributed architecture ensures that each player experiences minimal latency and benefits from the processing power of the nearby edge node.

One key advantage of the distributed model is improved scalability. By distributing the game servers across multiple locations, the infrastructure can handle a higher number of concurrent players and adapt to fluctuating player demands. This scalability is particularly crucial in multiplayer games with large player bases, where maintaining a smooth and responsive experience for all players is essential.

Furthermore, the distributed model offers enhanced fault tolerance and resilience. If one server or edge node experiences an issue or goes offline, the workload can be automatically shifted to other available nodes, minimizing the impact on gameplay. This redundancy ensures continuity and reduces the risk of service disruptions.

However, implementing a distributed communication model comes with its challenges. The synchronization of game state across multiple servers and ensuring consistent gameplay experiences for all players can be complex. Achieving low-latency communication and synchronization between the distributed nodes requires efficient network protocols and synchronization algorithms.

Additionally, the distributed model may require more significant infrastructure investments and maintenance compared to the centralized approach. Deploying and managing multiple edge nodes across different locations necessitates careful planning, coordination, and ongoing monitoring to ensure optimal performance and reliability.

Edge Computing

Edge computing plays a vital role in the communication model of streamed gaming, enabling low-latency interactions and reducing network bottlenecks. By leveraging edge computing technologies, the processing and data storage tasks are moved closer to the edge of the network, in proximity to the players, resulting in improved performance and responsiveness.

In the context of streamed gaming, edge computing involves deploying computing resources, such as servers or data centres, at the edge of the network, closer to the players' geographic locations. These edge nodes can be strategically positioned in various locations, such as data centres, points of presence (PoPs), or even network edge devices.

With edge computing, the game processing and rendering tasks can be offloaded from the player's device to the nearby edge nodes. This approach reduces the reliance on the player's local hardware, enabling the streaming of high-quality, resource-intensive games on a wider range of lower-powered devices. The edge nodes handle the computation-intensive tasks, such as game physics calculations, rendering, and video encoding, while delivering the video and audio content to the player in real time.

One of the key advantages of edge computing in streamed gaming is the significant reduction in latency. By minimizing the distance between the player and the processing resources, the communication latency is greatly reduced, leading to more responsive gameplay experiences.

This is particularly crucial in fast-paced multiplayer games, where even a slight delay can impact the player's performance and overall gaming experience.

Moreover, edge computing enables dynamic resource allocation and scalability. As the number of players fluctuates, the edge nodes can scale their processing power and capacity to accommodate the changing demands. This flexibility ensures that the gameplay experience remains smooth and uninterrupted, even during peak usage periods.

Additionally, edge computing offers improved fault tolerance and resilience. Since the game processing is distributed across multiple edge nodes, if one node experiences an issue or goes offline, the workload can be seamlessly transferred to other available nodes, ensuring continuous gameplay without disruptions.

However, there are some challenges associated with edge computing in streamed gaming. Ensuring data consistency and synchronization across distributed edge nodes can be complex, especially in multiplayer scenarios where multiple players interact with the same game environment. Efficient data replication, synchronization protocols, and consistency mechanisms need to be implemented to maintain a cohesive gaming experience.

Furthermore, managing the deployment and maintenance of edge nodes requires careful coordination and monitoring. Edge nodes need to be provisioned, updated, and monitored to ensure optimal performance and reliability. Additionally, edge computing infrastructure may require substantial investments in terms of hardware, networking, and management resources.

2.1.2. Video Stream

The successful delivery of streamed gaming relies heavily on the efficient streaming of video content to players' devices. Although closely related to existing streamed media services, streamed gaming has its unique requirements due to the real-time and interactive nature of video games.

Streamed media services involve the delivery of media content over the Internet, where the media is readied for consumption at the user's machine as soon as possible. This approach allows the media to start playing even before it is completely received by the user's device. Video streaming services typically employ a combination of encoding, server-side delivery, and client-side decoding to deliver a smooth playback experience.

In the context of streamed gaming, video streaming becomes even more critical as the entire game needs to be streamed to the player's device to alleviate the need for local installation and resource requirements. The process of video streaming for games shares similarities with streamed media services but requires additional considerations to ensure real-time interactivity and responsiveness.

The video streaming process begins with encoding, where the original audio and video data are compressed and converted into a suitable format for streaming. Commonly used video encoding schemes include MPEG-4, H.264, and AC-1. These encoding techniques aim to strike a balance between maintaining video quality and reducing the overall bitrate of the stream (Seeling et al., 2010).

Once encoded, the video stream is delivered from the server to the client. Various streaming protocols and technologies are employed to ensure the efficient transmission of the video stream over the network. Popular protocols such as MPEG-DASH, RTSP, and proprietary solutions govern the delivery of video streams, taking into account factors like network conditions and available resources (Fecheyr-Lippens, 2010; Schulzrinne et al., 1998; Sodagar, 2011).

Rate control is a crucial aspect of video streaming, allowing for the dynamic adjustment of video quality and bitrate. The quantization parameter (QP) is manipulated to achieve a target bitrate while maintaining a certain level of video quality. Rate control algorithms adjust encoding parameters based on the characteristics of the video frames to allocate bits efficiently within the stream. This ensures a consistent streaming bitrate and optimizes the compression process (all rights reserved copyright 2012, 2012).

Constant bitrate (CBR) encoding provides a steady bitrate, enabling predictable resource usage and bandwidth prediction. However, it may result in fluctuations in the quality of the streamed video. On the other hand, variable bitrate (VBR) encoding allows for variations in the required bandwidth but ensures consistent video quality. Constrained variable bitrate (CVBR) encoding offers a compromise between the steady bitrate of CBR and the consistent quality of VBR, optimizing bandwidth usage when necessary (all rights reserved copyright 2012, 2012).

In the context of streamed gaming, buffering of future content is challenging due to the constant and interactive nature of gameplay. Unlike streaming a pre-recorded movie, where buffering can be done during less complex scenes, streamed gaming requires real-time interaction and dynamic content that is subject to constant change based on user input. This dynamic nature presents a unique challenge in delivering a seamless and responsive gaming experience.

2.1.3. Game Genre and Requirements

The success of streamed gaming as a commercial service is still in its early stages, with significant technological challenges to overcome. Delivering a streamed gaming service requires substantial resources, typically supporting a limited number of titles. The subscription cost for streamed gaming, especially when including premium titles, is often higher compared to traditional streaming services for standard video and audio content. Additionally, the near real-time user interaction required in video games demands significant bandwidth and low latency.

Reviews of resource consumption for streamed gaming services have indicated bandwidth usage of up to 45 Mbit/s (Di Domenico et al., 2020). However, it is worth noting that Sony's streamed gaming offering, PSNow, requires significantly less bandwidth, potentially due to its earlier technology acquisitions and associated patents. Nonetheless, the current high bandwidth requirements, ranging from 25 to 45 Mbit/s, pose a limitation for a large portion of global internet users. Moreover, non-cable access methods like home Wi-Fi or telephony 4G/5G connections can further impact the quality of service (QoS) experienced by players in their homes.

While bandwidth is a primary concern, latency also plays a crucial role in streamed gaming. To ensure near real-time player interaction, a substantial infrastructure, often incorporating edge technologies (Ai et al., 2018; Satyanarayanan, 2017), is required to keep latency at acceptable levels. Latency can increase not only due to network delays but also as a result of processing overheads. The processing tasks involved in streaming gaming content, such as transposing output to a video stream, require significant investments in high-performing technology (Huang et al., 2014). Even with powerful PCs running video games locally, players may still experience jitter and freeze frames due to excessive resource usage.

Compared to traditional online video games, streamed video games experience additional latency primarily due to the downstream network load required to transmit the game video. In a single-server scenario, streaming a 1080p 60FPS game video can consume over 3 Mbit/s of bandwidth per client device (Huang et al., 2014). As the number of players increases, the cloud provider must further invest in network bandwidth to maintain a smooth gaming experience. Furthermore, physical distance poses a significant challenge in the remote server model. Users located far away from the server may experience a suboptimal streamed gaming experience due to the inherent delays introduced by the network (Chen et al., 2019).

The genre of the game also plays a role in determining the requirements and feasibility of streamed gaming. Games with fast-paced, highly interactive gameplay, such as first-person shooters or competitive multiplayer games, require extremely low latency to ensure responsive and enjoyable gameplay. On the other hand, turn-based strategy games or slower-paced single-player experiences may be more forgiving in terms of latency and bandwidth requirements.

Turn-based Games

- Board games (e.g. chess, Go, billiards)
- Card games (e.g. Hearthstone, Slay the Spire)
- Turn-based role-playing games (Tower of the Sorcerer)

Turn-based games are a distinct genre where user interaction is limited to specific moments, and the flow of the game progresses in turns. In these games, players take turns making decisions and executing actions, often without the need for real-time, instantaneous responses to in-game events. Instead, players have ample time to consider their options and make decisions before confirming their actions.

In turn-based games, individual interactions can be completed within a single frame. For example, in a digital chess game, players can use direction buttons to move a cursor between squares and a selection button to choose a chess piece and confirm its destination. These interactions can be acted upon to change the game state within individual frames. The lack of real-time response requirements in turn-based games means that input latency or jitter does not significantly impact gameplay.

Turn-based gaming is a popular genre, especially among players who have transitioned from physical card games to digital platforms (Ito, 2005). However, current commercial streaming platforms have shown limited consideration for delivering such titles. This is primarily because turn-based games often have a different price point associated with their business model (Rayna and Striukova, 2014), and the technical requirements to run these games on local devices are relatively low.

As a result, traditional turn-based games are not a primary concern for streamed gaming platforms at present. These games typically do not require real-time interactivity and are often better suited for local device execution. However, it's important to note that the landscape of streamed gaming is constantly evolving, and future advancements may provide opportunities for streamed turn-based gaming experiences.

Partially Real-time Games

- Sports games (e.g. golf, bowling)
- Puzzle games
- Real-time strategy games (RTS)

Partially real-time games introduce a hybrid gameplay experience that combines elements of turn-based and real-time interactions. In these games, players have the freedom to make decisions and perform actions at specific intervals, similar to turn-based games. However, certain actions or events within the game require real-time responsiveness and precise timing.

One example of a partially real-time game is a golf simulation, where players input their desired power and timing to hit the ball. The user holds a button or key to accumulate power, and the success of their shot depends on the precise release time. This real-time interaction adds a skill-based element to the game. However, in a streamed gaming context, the presence of latency can introduce inconsistencies and impact the player's experience.

When playing a partially real-time game through streaming, the player's actions are transmitted to the cloud server, which processes the inputs and generates subsequent frames of gameplay. However, due to the inherent latency in the streaming process, there can be a delay between the player's input and the server's perception of the action. This delay can disrupt the precise timing required for certain interactions, leading to inconsistencies and potentially rendering the game unplayable.

To mitigate this issue, predictive resource assignment can be employed in streamed gaming platforms. Similar to predictive interest management techniques used in non-cloud hosted online games, the server can anticipate the player's actions and allocate resources accordingly. By predicting the player's intended actions and precomputing the associated game state changes, the server can reduce the impact of latency on real-time interactions. This predictive approach aims to maintain consistency and ensure that the player's inputs are accurately reflected in the game's outcome.

Balancing resource allocation and latency reduction is crucial in delivering optimal partially real-time gaming experiences through streaming platforms. As technology continues to advance and streaming infrastructure improves, solutions that minimize latency and enable precise real-time interactions will enhance the feasibility and enjoyment of partially real-time games in a streamed gaming environment.

In summary, partially real-time games offer a unique combination of turn-based decision-making and real-time interactions. However, the latency introduced by streaming can pose challenges to maintaining consistency and responsiveness in these games. Predictive resource assignment techniques can help mitigate these issues by anticipating player actions and reducing the impact of latency on real-time interactions. As streamed gaming technology evolves, providing seamless and immersive partially real-time gameplay experiences will be an important focus for the industry.

Real-time Interaction Games

- Shooter games (e.g. first-person/third-person shooters [FPS/TPS], top-down shooter games, space shooters)
- Action games (e.g. Monster Hunter)
- Racing games (e.g. KartRider, Need For Speed)
- Sports games (e.g. basketball, soccer)

Real-time games are characterized by their constant need for user input, where player actions depend on the current state of the game and can be performed at any moment during the gameplay session. These games require players to react swiftly to in-game events and make decisions in real time. One prominent example of a real-time game genre is the first-person shooter (FPS), where players continuously send commands to control their character's movements, aim, and engage in combat.

In real-time games, player interaction relies heavily on the instantaneous visibility of the game state on the client device. For instance, in an FPS game, the player's ability to aim and shoot accurately depends on the real-time feedback displayed on their screen. Any input latency or delays in the transmission of commands can result in game inconsistency and a diminished gaming experience. Players need precise and responsive controls to effectively navigate the game environment and interact with other players or NPCs (non-player characters).

Supporting real-time games on streamed gaming platforms poses significant challenges. The use of edge computing technologies has become a common and necessary approach in order to minimize latency and ensure smooth gameplay experiences (Bhojan et al., 2020). Jitter, latency, or packet loss issues during message delivery can quickly deteriorate the player's experience, as these games demand fast and accurate responses to maintain competitiveness and immersion (Wahab et al., 2020).

The popularity of real-time games on streamed gaming platforms underscores the importance of providing a high-quality, low-latency streaming experience. Ensuring minimal input latency and delivering a consistent and responsive gameplay environment are critical factors in meeting the expectations of players who engage in real-time game genres. As streaming technology continues to advance, innovations in edge computing, network optimization, and low-latency protocols will be crucial in delivering seamless and immersive real-time gaming experiences to players around the world.

In summary, real-time games require constant user input and depend on the instantaneous visibility of the game state on the client device. The popularity of real-time game genres on streamed gaming platforms highlights the need for low-latency streaming and responsive controls. Edge computing technologies play a vital role in minimizing latency and ensuring smooth gameplay experiences. As technology progresses, further advancements in streaming infrastructure will enable even more immersive and enjoyable real-time gaming experiences.

2.2. Distributed Blockchain

Blockchain technology has gained significant attention and recognition in recent years, revolutionizing various industries and paving the way for new decentralized applications. At its core, a blockchain is a distributed and decentralized ledger that records transactions or any form of digital information in a secure and transparent manner. Unlike traditional centralized databases, a blockchain is not controlled by a single authority but is collectively maintained and verified by a network of participants known as nodes. Each node stores a copy of the entire blockchain, ensuring data redundancy and integrity.

The fundamental concept behind blockchain is the concept of blocks linked together to form a chain. When a transaction occurs, it is grouped with other transactions into a block. Before a block is added to the blockchain, it undergoes a verification process by the network's nodes. This verification involves validating the authenticity and integrity of the transactions, ensuring they meet specific predefined rules or conditions.

Once verified, the block is added to the blockchain, becoming a permanent part of the ledger. Each block contains a unique identifier called a hash, which is generated through a cryptographic function. The hash of each block also includes the hash of the previous block, creating a sequential chain of blocks. This linkage ensures the immutability and integrity of the entire blockchain since altering a block would require changing the hash of that block and all subsequent blocks.

One of the most well-known applications of blockchain technology is cryptocurrencies, such as Bitcoin and Ethereum. Cryptocurrencies leverage blockchain to enable secure peer-to-peer transactions without the need for intermediaries like banks. Blockchain ensures the integrity and transparency of these transactions, making them resistant to fraud and tampering.

2.2.1. Cryptographic Hashing

Cryptographic hashing is a fundamental technique employed in blockchain systems to ensure the integrity, security, and immutability of data. It involves the use of cryptographic hash functions, which are mathematical algorithms that convert an input of any size into a fixed-size output called a hash.

In the context of blockchain, cryptographic hashing serves several essential purposes:

- Data Integrity: Cryptographic hashing provides a way to verify the integrity of data stored in blocks. When data is hashed, even the slightest change to the input data will result in a completely different hash value. By comparing the computed hash with the stored hash, participants in the network can detect any tampering or unauthorized modifications to the data.

- Unique Identifiers: Each block in a blockchain is assigned a unique identifier known as a cryptographic hash. This hash is generated by applying a hash function to the block's data, including transactions and other metadata. The hash serves as a digital fingerprint for the block, ensuring its uniqueness and enabling efficient identification and retrieval.
- Linking Blocks: Cryptographic hashes are used to establish the linkage between blocks in the blockchain. Each block includes the hash of the previous block in its data, creating a chain of blocks. This linking mechanism ensures the chronological order of transactions and provides a tamper-evident structure. Any alteration to a previous block's data would result in a mismatched hash, signalling the tampering attempt.
- Proof-of-Work: Cryptographic hashing plays a crucial role in the consensus mechanism known as Proof-of-Work (PoW). Miners in PoW blockchains compete to find a hash that meets certain predefined criteria. This process involves repeatedly hashing the block's data with different inputs until a hash with the desired properties is discovered. The successful miner's hash serves as proof that they have expended computational work, contributing to the security and decentralization of the blockchain.

Cryptographic hash functions possess several desirable properties, including:

- Determinism: Given the same input, a hash function will always produce the same output. Preimage Resistance: It is computationally infeasible to determine the original input from the hash output.
- Collision Resistance: It is highly improbable for two different inputs to produce the same hash output.
- Efficiency: Hash functions are designed to be computationally efficient, allowing for quick and reliable hashing of data.

Popular cryptographic hash functions used in blockchain systems include SHA-256 (Secure Hash Algorithm 256-bit) and Keccak-256 (part of the SHA-3 family).

2.2.2. Digital Signatures

Digital signatures are crucial in maintaining transaction safety and dependability. They're created by the owner of a transaction with a special private key that's kept secret and is known exclusively to them. Using cryptographic techniques, this private key forms a digital signature that distinguishes and validates the transaction's credibility.

To check the signature, other users or entities use the corresponding public key, which is shared openly by the owner of the transaction. By contrasting the received signature with the transaction message and implementing the public key, the recipient can confirm the transaction's legitimacy. This operation confirms that no alterations or tampering have occurred since the transaction was initially signed.

The security of the digital signature framework hinges on the central concept that only the owner of the transaction knows the private key. The characteristics of asymmetric encryption algorithms make it virtually impossible for others to form a valid digital signature using the owner's public key. As a result, the private key of the transaction owner serves as a unique sign, guaranteeing that the transaction came from the approved party and hasn't been falsified or tampered with by unauthorized persons.

With the aid of digital signatures, transactions can be safely carried out and checked across different fields, such as financial transactions, legal agreements, electronic voting systems, and secure communication protocols. The resilience and tamper-proof feature of digital signatures are essential in building trust and maintaining the integrity of digital transactions in our increasingly digital and interconnected world.

2.2.3. Transactions

Transactions are integral components of blockchain systems, representing the transfer of digital assets or information between participants in the network. Whether it involves cryptocurrencies, smart contracts, or other forms of data exchange, transactions are fundamental to the operation and purpose of blockchain technology.

A transaction in a blockchain typically includes the following elements:

- Sender and Receiver: Transactions involve at least two parties—the sender, who initiates the transaction, and the receiver, who is the intended recipient of the digital assets or information.
- Digital Signature: To ensure the authenticity and integrity of a transaction, the sender utilizes a digital signature. The digital signature is generated using the sender's private key, verifying their identity and providing proof of authorization.
- Transaction Details: This section of the transaction includes specific information about what is being exchanged. In the case of cryptocurrencies, it typically includes the amount of currency being transferred, the recipient's address, and any additional data relevant to the transaction.
- Inputs and Outputs: Transactions consume specific inputs, which are typically unspent outputs from previous transactions. The outputs of a transaction become the inputs for subsequent transactions. This chain of inputs and outputs forms the transaction history, ensuring traceability and accountability within the blockchain.
- Transaction ID: Each transaction is assigned a unique identifier called a transaction ID or hash. The transaction ID is generated by applying a cryptographic hash function to the transaction data, producing a fixed-length string of characters that uniquely represents the transaction. This hash serves as a reference for verifying the transaction's integrity and linking it to subsequent transactions.

In a blockchain, transactions are bundled together within blocks and added to the chain in a sequential order. Miners or validators in the network validate the transactions and ensure their compliance with the predefined rules and consensus mechanism of the blockchain. Once validated, the transactions become a permanent part of the blockchain's ledger, creating an auditable and transparent record of all transactions.

The decentralized nature of blockchain technology ensures that transactions can be verified and validated by multiple participants in the network. This distributed validation process increases security, as it requires a consensus among network participants to accept a transaction as valid and add it to the blockchain.

2.2.4. Blocks

In a blockchain system, blocks serve as the fundamental units for organizing and storing data. Each block contains a collection of transactions or other relevant data, forming a sequential chain of blocks.

A block typically consists of several key components:

- Block Header: The block header contains metadata and crucial information about the block, including a timestamp, a unique block identifier, and a reference to the previous block in the chain.
- Transactions: Transactions are the core elements of a block. They represent the exchange of digital assets, such as cryptocurrencies or other data, between participants in the network. Transactions include details such as the sender's address, the recipient's address, the amount transferred, and additional transaction-specific data.
- Cryptographic Hash: Each block is associated with a unique cryptographic hash generated using a cryptographic hash function. The hash serves as a digital fingerprint of the block's content. Even a slight modification to the data within the block will result in a completely different hash value.
- Merkle Tree Root: In many blockchain systems, including Bitcoin, transactions within a block are organized using a Merkle tree structure (Nakamoto, 2019). The Merkle tree allows for efficient verification of the integrity of the transactions within the block. The root of the Merkle tree, known as the Merkle root, is stored in the block header.

The structure and organization of blocks play a vital role in the security and integrity of the blockchain. By linking blocks together through their unique identifiers and cryptographic hashes, the entire history of transactions becomes tamper-proof. Modifying the data in one block would require recalculating the hashes of subsequent blocks, making it computationally infeasible to alter the blockchain's history.

As new transactions occur within the network, they are collected and grouped into blocks. Miners, who are responsible for maintaining the blockchain's integrity, compete to solve complex mathematical puzzles through a process known as mining. Once a miner successfully solves the puzzle, they can append a new block to the existing chain, incorporating the validated transactions. This process adds new data to the blockchain while preserving the immutability and chronological order of the transactions.

2.2.5. Blockchain

The blockchain acts as a clear, distributed public record that guarantees data reliability and immutability through a sophisticated method. Employing a cryptographic hash operation, every data block within the blockchain receives a distinctive hash value, functioning as a digital identifier for that block. This hash is formed by integrating multiple components, such as the data itself, a nonce (an arbitrary number), and the previous block's hash.

The insertion of a new block into the blockchain triggers a mining operation. During mining, competitors strive to discover a nonce that, combined with the block's data, results in a hash meeting a particular requirement. The hash is often required to have a certain number of zeros at the start. The mining operation, while demanding significant computational resources, offers a consensus approach to authenticate the blockchain's accurate version. By resolving this cryptographic challenge, miners protect the block and confirm the transactions within.

Mining is essential in preserving the safety and immutability of the blockchain. Altering any data from a blockchain block changes the subsequent hash, rendering it incompatible with the consensus rules' stipulated condition. Thus, anyone attempting to manipulate the data would need to redo the hash and find a fresh nonce that fulfils the requirement. Moreover, due to the inclusion of the preceding block's hash in each block, any alteration would propagate through the following blocks, making the re-mining of the entire blockchain necessary. This intricate link ensures that any tampering with the blockchain becomes computationally unfeasible and straightforward to spot.

A notable feature of the blockchain is its capability to offer a reliable and effective method to confirm the data it holds. By contrasting the hash of the final block, often referred to as the "last block hash," on multiple blockchain copies, one can ascertain their exactness. If the last block hashes align, it suggests that the complete transaction history, starting from the initial block, is identical in both blockchain versions. This approach streamlines the task of checking blockchain uniformity and allows users to independently verify the accuracy of the collective ledger.

2.2.6. Peer-to-Peer

The peer-to-peer (P2P) network model has significantly risen in popularity, largely due to its widespread implementation in distributed systems. This network design regards all nodes as peers, forming a web-like linkage among them. With this interconnected framework, any node can establish direct communication with any other within the network, eliminating the requirement for centralized intermediaries.

A salient benefit of P2P architecture lies in its potential for resource distribution among network nodes in a distributed setup. Instead of depending on a central server or governing entity, nodes can directly exchange and gain access to resources with each other. This decentralized modality fosters a more balanced dispersion of resources, allowing peers to contribute their computing prowess, storage potential, and data without relying on specialized infrastructure.

The distributed attribute of the P2P model also leads to augmented system decentralization. Given that each node operates independently, there’s no single failure or control point. This decentralized configuration bolsters system resilience and sturdiness since a single node’s failure doesn’t affect the entire network. Moreover, the lack of central authority nurtures a more egalitarian and inclusive environment, encouraging individual nodes to partake in decision-making and contribute to the system’s overall operations.

Moreover, P2P architecture stimulates collaboration and cooperation among peers. Through forming direct channels of communication between nodes, information and data can be exchanged seamlessly, promoting knowledge sharing and collective problem-solving. This cooperative atmosphere can birth innovative solutions as nodes can harness the varied expertise and resources across the network.

The adaptability of P2P networks has made them a vital element in various fields. Ranging from file-sharing applications to content delivery networks, from blockchain tech to decentralized platforms, P2P architecture presents a flexible and scalable structure for distributed systems. It enables individuals and organizations to build decentralized ecosystems, facilitating peer interactions, efficient resource utilization and strengthened system resilience.

2.2.7. Distributed System

In a distributed system, miners' combined endeavours act as a formidable shield against the insertion of deceitful or fraudulent information into the blockchain. Their collective engagement maintains the veracity and dependability of the distributed ledger.

When a transaction owner broadcasts transactions over the network, miners receive and meticulously validate each transaction. They evaluate various elements such as transaction legitimacy, digital signatures, and adherence to consensus rules, thereby playing a vital role in upholding the blockchain's precision and consistency.

Once a node receives a validated block, it independently checks the block's integrity and authenticity. It ensures the block complies with the consensus rules and conforms to the network's protocol. If the block is verified successfully, it is appended to each node's local blockchain copy.

In some instances, two fresh blocks with distinct hashes may be broadcast across the network at the same time. Nodes are then faced with choosing which block to adopt and integrate into their blockchain. Ordinarily, nodes opt for the first block they receive while keeping the second block as a viable alternative. This approach allows the network to adjust to short-term variations and possible forks in the blockchain.

In order to sustain consensus and synchronization throughout the distributed system, miners persistently strive to elongate the longest chain. They allocate computational resources to mine new blocks, thus enhancing the growth and security of the blockchain. If the blockchain's longest branch changes due to a new block's addition, miners readily shift their attention to the new longest branch. This ensures the network functions based on the most recent and trustworthy blockchain.

2.2.8. Decentralization and Consensus Algorithm

One of the fundamental principles of blockchain technology is decentralization, which refers to the distribution of control and decision-making across a network of participants. In a decentralized blockchain system, there is no central authority or single point of control. Instead, power and authority are shared among multiple participants, making the network more resilient, transparent, and secure.

Decentralization in blockchain is achieved through the use of a distributed consensus mechanism. Consensus algorithms enable participants in the network to agree on the validity and order of transactions added to the blockchain. These algorithms ensure that all nodes in the network reach a consensus on the state of the blockchain, even in the presence of malicious actors or network failures.

Proof-of-Work

Proof of Work (PoW) is a consensus algorithm broadly utilized in blockchain structures, purposed to safeguard the trustworthiness and immutability of the distributed ledger. It is a critical component in achieving consensus among network participants, prohibiting ill-intentioned entities from altering the blockchain. This subsection delves into the principal ideas and processes behind the PoW consensus algorithm.

PoW encompasses miners vying to resolve a computation-heavy puzzle to authenticate and append new blocks to the blockchain. The puzzle necessitates miners to identify a nonce, an arbitrary number, which, when fused with the block's data, produces a hash value satisfying specific conditions. These conditions typically require the hash to have a certain number of zeros at the start. The puzzle's difficulty is adjusted by modifying these conditions to preserve a consistent block production rate.

Miners apply brute-force computational power to cycle through numerous nonce values until a valid solution is obtained. This procedure demands considerable computational resources and electricity usage. Upon finding a valid nonce, the miner broadcasts it to the network, serving as evidence that computational work has been carried out.

Other nodes within the network independently affirm the proof-of-work's validity by applying the same conditions to the received block. This validation process ensures consensus is achieved throughout the network. If the proof-of-work is confirmed as valid, the block is integrated into the blockchain, and the miner who made the discovery is rewarded with cryptocurrency tokens or transaction fees.

The PoW consensus algorithm boasts several remarkable benefits. Firstly, it offers a mechanism for decentralized decision-making, as no single entity or group solely governs the validation process. Also, the computational effort needed to solve the puzzle acts as a hindrance against malicious actors aiming to tamper with the blockchain. PoW further aids in thwarting double-spending attacks, where a user tries to spend the same cryptocurrency tokens multiple times.

Nonetheless, PoW does harbour certain disadvantages, primarily related to its energy usage and scalability. The computational power necessary for mining leads to substantial energy consumption, raising environmental concerns about blockchain networks that rely on PoW. Moreover, the computational intensity of PoW limits the transaction processing speed, potentially causing scalability issues as the network expands.

Lighting Network

The Lightning Network represents a layer-two scalability solution, developed atop blockchain networks, principally devised to resolve the scalability and transaction throughput constraints of cryptocurrencies like Bitcoin. This subsection delves into the basics and relevance of the Lightning Network within the scope of consensus algorithms and blockchain scalability.

The Lightning Network functions as an off-chain payment channel network that facilitates swift and low-cost transactions by minimizing dependence on on-chain transactions. It capitalizes on the inherent security of the underlying blockchain while offering an added layer for immediate payment settlement. Employing bi-directional payment channels, participants can conduct a series of transactions without involving the blockchain for every single transaction.

To establish a Lightning Network payment channel, participants secure a certain amount of cryptocurrency funds in a multi-signature transaction on the blockchain. These funds act as the collateral for the channel. Once the channel is created, participants can execute an unlimited number of off-chain transactions instantly and with negligible cost.

The Lightning Network accomplishes scalability and augmented transaction throughput by enabling direct payment channels between participants and facilitating payments to be routed across the network. By harnessing a network of payment channels, participants can transact with other network participants, even if they don't have a direct channel opened with them. This payment routing capability effectively broadens the network's reach and liquidity, enhancing scalability and enabling near-instant transactions.

The Lightning Network employs a payment channel protocol and a routing algorithm to enable secure and efficient payment routing across the network. Smart contracts and cryptographic techniques ensure the integrity and security of transactions, allowing participants to transact off-chain safely while preserving the finality and security guarantees of the underlying blockchain.

A key advantage of the Lightning Network is its capability to considerably reduce transaction fees and boost transaction speed. By conducting transactions off-chain and settling the final outcome on the blockchain, participants can bypass the need for individual on-chain transactions, which often carry higher fees and slower confirmation times.

Furthermore, the Lightning Network amplifies privacy by reducing the visibility of transaction details on the blockchain. Since most transactions occur off-chain, they are not publicly visible, providing a higher level of privacy for network participants.

However, the Lightning Network also has its challenges. The requirement for an established payment channel and liquidity among participants can create initial entry barriers. Also, the network's routing protocol and liquidity management necessitate constant development and optimization to ensure efficient and reliable payment routing.

Proof-of-Stake

Proof of Stake (PoS) is a distinct consensus algorithm broadly applied in blockchain networks as a scalable, energy-efficient counterpart to Proof of Work (PoW). This section explores the fundamental principles and operations that underlie the PoS consensus algorithm.

Contrary to PoW, where miners engage in a competition based on computational power, PoS chooses block validators in relation to their ownership or "stake" in the cryptocurrency. In a PoS framework, the entities involved are termed validators and are chosen to construct new blocks and authenticate transactions based on the number of tokens they possess and are willing to "stake" as security.

The validator selection process in PoS is typically managed by an algorithm that factors in variables like the number of tokens held and the duration they have been staked. This deterministic selection process is designed to ensure a balanced and secure allocation of the responsibility to create new blocks and authenticate transactions across network participants.

Once chosen, validators propose and validate new blocks according to their stake in the network. The probability of being selected to validate a block is proportional to the validator's stake. This implies that validators with a more substantial stake have a higher chance of being chosen and earning rewards.

To uphold the security and integrity of the blockchain, PoS introduces a notion termed "slashing" to discourage malicious conduct. If a validator attempts to alter the blockchain or acts against the network's interests, their stake might be partially or entirely seized as punishment.

One of the main advantages of PoS is its energy efficiency relative to PoW. By eradicating the need for computationally intensive mining computations, PoS notably reduces the energy consumption linked with block validation. This makes PoS a more eco-friendly consensus algorithm, aligning with the growing emphasis on sustainable blockchain solutions.

PoS also offers enhanced scalability compared to PoW. Since block validation is not dependent on computational power, PoS networks can process transactions more rapidly and efficiently, permitting higher transaction throughput and scalability.

Despite its advantages, PoS faces its own challenges. A key concern is the "nothing at stake" issue, where validators may lack disincentives to validate multiple competing chains, potentially resulting in network instability. To address this problem, PoS algorithms typically introduce penalties for validators who try to validate on multiple chains.

Proof-of-Authority

Proof of Authority (PoA) is a consensus mechanism used in specific blockchain networks to facilitate rapid transaction validation while maintaining a robust level of security. This section delves into the basic principles and features of the PoA consensus algorithm.

In PoA systems, block validators are chosen not based on computational power or token ownership but on their established authority or reputation within the network. These validators, often referred to as authorities or nodes, are usually selected based on their credibility, expertise, or vested interest in the system. By assigning trusted validators, PoA aims to speed up the validation process and enhance the overall efficiency of the blockchain network.

Validators in a PoA system bear the responsibility of validating transactions and generating new blocks. Their established authority and trustworthiness ensure the integrity and security of the network. Unlike PoW or PoS systems, where validators compete or stake tokens, PoA relies on a somewhat centralized model where a limited number of trusted authorities handle block validation.

The process of transaction validation in PoA involves the selected authorities verifying and signing transactions based on their established status of authority. When a transaction receives enough signatures from the assigned authorities, it is deemed valid and added to the blockchain. This deterministic approach guarantees fast transaction finality and eliminates the need for extensive computational resources.

One key advantage of PoA is its high transaction throughput and low latency. By minimizing the dependence on computational puzzles or resource-intensive mining, PoA allows for faster transaction processing, making it suitable for use cases where speed and efficiency are paramount, such as enterprise and private blockchain networks.

The PoA consensus algorithm also offers strong protection against Sybil attacks, as the system depends on a limited number of trusted validators. Since validators are chosen based on their reputation and authority, the probability of malicious actors gaining control over the network is significantly lowered. This enhances the security and resilience of the blockchain system.

However, PoA comes with some compromises. The centralization of authority can create potential single points of failure or lead to collusion among validators. The trust placed in the selected authorities necessitates a high level of confidence in their integrity and accountability. Also, depending on a limited number of validators may reduce the degree of decentralization and censorship resistance typically associated with public blockchain networks.

2.2.9. Immutable and Transparent Nature of Blockchain

One of the defining characteristics of blockchain technology is its immutable and transparent nature. These properties contribute to the trust and integrity of blockchain networks, making them suitable for a wide range of applications, including financial transactions, supply chain management, and data provenance.

Immutability refers to the inability to alter or tamper with data stored on the blockchain once it has been added to a block and confirmed by the network. Each block in the blockchain contains a cryptographic hash of the previous block, creating a chain of blocks that is resistant to modification. This cryptographic linking ensures that any changes to a block would require modifying subsequent blocks, making it computationally infeasible and highly improbable. Consequently, the immutability of blockchain provides a secure and tamper-proof record of transactions and information.

The transparency of blockchain stems from its decentralized and distributed nature. In a public blockchain, such as Bitcoin or Ethereum, the entire transaction history is visible to all participants in the network. Every transaction and subsequent block added to the blockchain is recorded and replicated across multiple nodes. This transparency allows anyone to independently verify and audit the transactions, providing a high level of trust and accountability. It also reduces the reliance on intermediaries and central authorities, as the decentralized network collectively maintains the integrity of the blockchain.

The transparency of blockchain enables a variety of use cases. In financial transactions, it allows participants to trace the flow of funds and verify the legitimacy of transactions, reducing the risk of fraud and enhancing financial transparency. In supply chain management, blockchain can provide a verifiable record of the origin, movement, and authenticity of goods, increasing trust and improving accountability throughout the supply chain. Additionally, in areas such as data provenance and intellectual property rights, blockchain's transparent nature enables the tracking and verification of the ownership and history of digital assets.

However, while blockchain offers transparency, it also presents challenges in terms of privacy. Public blockchains, by design, reveal transaction details to all participants, which may not be desirable in certain scenarios where privacy is paramount. To address this, various privacy-enhancing techniques, such as zero-knowledge proofs and off-chain transactions, have been developed to allow for selective disclosure of information while preserving the integrity of the blockchain.

2.2.10. Security and Trust in Blockchain Networks

Security and trust are fundamental pillars of blockchain technology, ensuring the integrity, reliability, and authenticity of data within a blockchain network. The design and cryptographic mechanisms employed by blockchain systems contribute to the robust security and trustworthiness of these networks.

Blockchain networks are secured through the use of cryptographic techniques, such as hash functions and digital signatures. Hash functions are mathematical algorithms that convert data into a fixed-length string of characters, known as a hash. Each block in a blockchain contains a unique hash that is generated based on the data within the block and the hash of the previous block. This cryptographic linking creates a chain of blocks that ensures the immutability and tamper-proof nature of the data.

Digital signatures play a crucial role in verifying the authenticity and integrity of transactions in a blockchain. Each participant in the network possesses a unique digital signature, which is generated using their private key. When a transaction is initiated, the sender's digital signature is attached to it, serving as proof of their identity and ensuring that the transaction has not been altered during transit. The recipient can verify the signature using the sender's public key, providing assurance that the transaction originated from the expected sender and has not been tampered with.

The decentralized and distributed nature of blockchain networks enhances security and trust. Instead of relying on a central authority, blockchain relies on a network of participants, or nodes, that collectively validate and verify transactions. Consensus algorithms, such as Proof of Work or Proof of Stake, ensure that the majority of participants agree on the validity of transactions before they are added to the blockchain. This consensus mechanism makes it extremely difficult for malicious actors to manipulate or tamper with the data, as it would require control of a majority of the network's computational power or stake.

Additionally, the transparency of blockchain networks contributes to security and trust. Public blockchains allow anyone to view and audit the transaction history, providing transparency and accountability. This transparency fosters trust among participants, as they can independently verify the integrity of the transactions and the state of the blockchain.

Despite the robust security measures, blockchain networks are not entirely immune to security threats. While the technology itself is highly secure, vulnerabilities can arise from implementation flaws, human errors, or external attacks on individual nodes or wallets. It is crucial to follow best practices for key management, secure wallet storage, and network security to mitigate these risks and maintain the integrity of blockchain systems.

2.3. Challenges of Streamed Gaming In-Game Trading

The growth of the video game industry has given rise to a thriving economy centred around the trade of in-game virtual goods. These digital commodities, ranging from cosmetic alterations to functional enhancements, hold significant value for players in terms of personalisation and game progression. However, the in-game trading market faces several challenges that can hinder its potential, including fraudulent items, lack of clear information, and disputes over ownership. This section explores the challenges associated with in-game trading and the need for a secure, trustworthy, and open framework to address them.

2.3.1. Ensuring Security in Transactions

As in-game trading involves real or virtual currencies, ensuring secure transactions is paramount. Users should be able to receive or purchase virtual assets with confidence, knowing that they are engaging in legitimate transactions. The secure method employed should guarantee the authenticity and integrity of the assets and track the provenance of transactions. Blockchain technology offers a promising solution by providing a decentralised and immutable ledger that records ownership and transaction history. By leveraging blockchain-like approaches, it becomes possible to establish a secure and transparent framework for in-game trading, reducing the risk of inappropriate or criminal behaviour.

2.3.2. Addressing Inappropriate and Criminal Behavior

To foster a trustworthy in-game trading ecosystem, it is crucial to identify and address inappropriate and criminal behaviour. This includes activities such as hacking, cheating, or the sale of illicit virtual goods. Implementing mechanisms to detect and prevent such behaviour is essential to maintain the integrity of the trading environment. By leveraging blockchain technology's transparency and traceability, it becomes possible to track ownership and transactions, making it easier to identify and address any fraudulent or malicious activities.

2.3.3. Maintaining Real-Time Interactivity

In a stream interactive media environment, where low-powered hardware enables access to high-fidelity graphical simulations and real-time interactions, maintaining stable and acceptable latency is crucial. Most video games require high interactivity to deliver a satisfactory player experience. Any significant delays or variations in latency can disrupt the immersion and control of the game, leading to player frustration and potential abandonment. Ensuring real-time interaction in a distributed network environment poses a challenge that needs to be effectively addressed to provide a seamless gaming experience.

2.3.4. Balancing Resource Requirements and Robustness

Creating a framework for in-game trading that is both resource-efficient and robust is another challenge to consider. The solution should be able to handle a significant volume of transactions while maintaining low latency and ensuring the security of the system. Balancing the resource requirements, including computing power and network bandwidth, is crucial to provide a smooth and responsive trading experience for users. An evaluation model can be developed to analyse and assess different solutions, considering factors such as transaction latency, computing resource requirements, and security support.

2.4. Enhancing In-Game Trading in Streamed Gaming through Blockchain Technology

This section will delve into the transformative potential of blockchain technology in refining in-game trading mechanisms within streamed gaming. By providing a robust framework for ensuring security, establishing trust and transparency, and enabling the creation of decentralised marketplaces, blockchain technology promises to revolutionise the current paradigm. This comprehensive discussion will illustrate how the technology fosters a more secure and fair gaming environment while also expanding trading opportunities for gamers globally.

2.4.1. Ensuring Security and Provenance

Blockchain technology, known for its transparent and immutable nature, can provide a secure framework for in-game transactions and asset ownership. By leveraging the blockchain's distributed ledger, each transaction and asset transfer can be recorded, creating an unchangeable history of provenance. This ensures that users can verify the authenticity and ownership of in-game items, reducing the risk of fraudulent activities. The decentralised nature of blockchain also mitigates the reliance on centralised authorities, enhancing security and preventing unauthorised modifications or tampering.

2.4.2. Establishing Trust and Transparency

Streamed gaming platforms integrated with blockchain can introduce trust and transparency into the in-game trading ecosystem. Through smart contracts, predefined rules and conditions can govern transactions, ensuring that both parties adhere to the agreed-upon terms. Smart contracts automatically execute transactions once the conditions are met, eliminating the need for intermediaries and reducing the possibility of disputes. The transparency provided by blockchain allows users to validate the fairness and legitimacy of in-game transactions, fostering trust among players and improving the overall trading experience.

2.4.3. Enabling Decentralized Marketplaces

Blockchain technology can facilitate the creation of decentralised marketplaces within streamed gaming platforms. These marketplaces allow players to trade virtual assets directly with each other, eliminating the need for third-party intermediaries or centralised control. By utilising blockchain's decentralised infrastructure, users can engage in peer-to-peer transactions, expanding the range of available trading opportunities and empowering players to have more control over their virtual assets. This decentralised approach promotes a more open and inclusive environment for in-game trading, benefiting both individual players and the overall gaming community.

2.5. Blockchain Based Applications

This section delves into the wide-ranging world of blockchain-based applications, exploring the depth and diversity of their transformative potential across multiple domains. We will navigate the digital realms of Bitcoin and Ethereum, including the intricacies of smart contracts, which lay the foundation for decentralised applications. Venturing into the realm of the Internet of Things, we'll examine the critical role of provenance tracking. The potential of blockchain in healthcare will also be highlighted, particularly focusing on patient-driven record maintenance. Further, we'll dissect the integral function of blockchain in ensuring supply chain provenance tracking, its burgeoning role in the world of digital arts through non-fungible tokens, and its growing impact on the gaming industry. Finally, we'll explore how blockchain is reshaping data exchange in video games and investigate the next generation of gaming through 'CloudArcade'.

2.5.1. Bitcoin

Bitcoin, created by the pseudonymous individual or group known as Satoshi Nakamoto, stands as the first and most well-recognised cryptocurrency built on blockchain technology (Nakamoto, 2019). Operating as a decentralised peer-to-peer electronic cash system, it enables individuals to transact directly without needing intermediaries such as banks or financial institutions.

The defining innovation of Bitcoin resides in its underlying blockchain technology, which functions as a transparent and immutable ledger of all transactions. Every Bitcoin transaction is verified and appended to a block, generating a chronological chain of blocks that cannot be retroactively altered. This safeguards against double-spending and assures the integrity of transaction history, fostering a high degree of security and trust.

Bitcoin transactions are confirmed and validated by a network of participants known as miners. These miners compete to solve intricate mathematical puzzles in a process referred to as mining. Upon solving a puzzle, the miner appends a new block of transactions to the blockchain and receives a reward of newly created Bitcoins for their work.

Another crucial aspect of Bitcoin is its built-in scarcity. With a finite supply capped at 21 million Bitcoins and the rate of new Bitcoin issuance halved periodically in an event known as "halving," this scarcity has contributed to the digital currency's perceived value, framing Bitcoin as a store of value akin to digital gold.

Beyond its role as a digital currency, Bitcoin has incited a worldwide movement around blockchain technology, spurring the development of countless cryptocurrencies and blockchain-based applications. Bitcoin's decentralised, permissionless nature has inspired innovations in financial systems, decentralised finance (DeFi), remittances, and cross-border transactions.

Despite its considerable impact, Bitcoin is not devoid of challenges. Its public blockchain, while delivering transparency, grapples with scalability issues, which restrict transaction throughput and can lead to elevated fees during times of high demand. Furthermore, Bitcoin's substantial energy consumption and resulting environmental impact, a byproduct of the mining process, have attracted criticism.

Nonetheless, as a trailblazer in blockchain technology providing decentralised, censorship-resistant digital currency, Bitcoin has been a significant driving force for the widespread acceptance and exploration of blockchain-based applications.

2.5.2. ETH and smart contract

Ethereum, sometimes called ETH, is a key blockchain-based platform that builds upon the foundational concepts established by Bitcoin. This section sheds light on the vital characteristics and the significance of Ethereum within the field of blockchain-centric applications.

First suggested by Vitalik Buterin in 2013, Ethereum functions as a decentralised ecosystem that enables the implementation of smart contracts. It uses its inherent cryptocurrency, Ether (ETH), as the medium for value transfer within its network. Ethereum's blockchain not only permits Ether transfers but also grants developers the ability to design and roll out decentralised applications (DApps) through the use of smart contracts.

Smart contracts, which autonomously operate on the Ethereum blockchain, automatically honour the stipulations they are programmed with, thereby eradicating the necessity for third parties while boosting transparency and fostering trust. With the help of smart contracts, Ethereum paves the way for the development of a broad array of applications in a multitude of sectors, such as finance, supply chain management, decentralised exchanges, gaming, and beyond.

A hallmark characteristic of Ethereum is its Turing-complete programming language. This provides developers with the capacity to build sophisticated applications and smart contracts capable of carrying out intricate operations. This feature sets Ethereum apart from other blockchain platforms and has driven its widespread acceptance and usage.

Ethereum's blockchain is built on a consensus mechanism known as Proof of Stake (PoS). Unlike Bitcoin's energy-intensive Proof of Work (PoW) system, PoS depends on validators who pledge, or "stake," their Ether to maintain the security of the network. This shift to PoS aims to amplify scalability, minimise energy use, and increase transaction speed on the Ethereum network.

In addition to its programmability and flexibility, Ethereum has brought the concept of decentralised finance (DeFi) to the blockchain landscape. DeFi encompasses a suite of financial applications and protocols constructed on Ethereum that aspire to deliver traditional financial services in a decentralised and transparent way. This includes decentralised exchanges, lending and borrowing platforms, stablecoins, and yield farming protocols. DeFi has made considerable inroads and has significantly disrupted conventional financial structures by eliminating intermediaries and facilitating peer-to-peer financial transactions.

Despite these benefits, Ethereum faces hurdles such as scalability and elevated transaction fees during periods of high network use. To address these challenges, Ethereum is in the midst of a major transformation known as Ethereum 2.0, which will introduce sharding and a more effective consensus mechanism to boost scalability and decrease transaction expenses.

2.5.3. Provenance Tracking in the Internet of Things

Provenance tracking, or the ability to trace the origin and history of data, is a critical requirement in the context of the Internet of Things (IoT). This subsection explores the significance of using blockchain technology for provenance tracking in the IoT, drawing upon the research paper "Blockchain-based data provenance for the internet of things" as a foundation (Sigwart et al., 2019).

The Internet of Things comprises a vast network of interconnected devices that generate and exchange data. With the increasing adoption of IoT technologies across various domains, the need for trustworthy and verifiable data becomes paramount. Provenance tracking enables stakeholders to understand the origin, transformations, and movements of data throughout its lifecycle, ensuring data integrity and reliability.

Blockchain technology offers a promising solution to address the challenges associated with provenance tracking in the IoT. The immutable and decentralised nature of blockchain allows for transparent and tamper-evident records, ensuring the integrity and authenticity of data provenance.

The research paper highlights the application of blockchain-based data provenance in the IoT. By leveraging blockchain, data producers can securely record metadata and information about the origin, ownership, and modifications of IoT-generated data. Each data transaction or modification is captured as a transaction on the blockchain, creating an immutable and auditable trail of data provenance.

Using blockchain technology for provenance tracking in the IoT offers several advantages. Firstly, it enhances data trustworthiness by providing an auditable and transparent record of data origins, transformations, and transfers. This promotes accountability and ensures the verifiability of data sources.

Secondly, blockchain-based provenance tracking can facilitate data integrity verification, allowing data consumers to verify the integrity of received data by tracing its path and ensuring that it has not been tampered with during transmission or storage. This is particularly critical in applications where data accuracy is crucial, such as healthcare, supply chain management, and environmental monitoring.

Moreover, blockchain technology enables data owners to have control over their data and determine who can access and modify it. Smart contracts can be utilised to enforce data usage policies and access permissions, ensuring data privacy and security within the IoT ecosystem.

However, implementing blockchain-based data provenance in the IoT also presents challenges. The scalability of blockchain networks and the resource constraints of IoT devices need to be considered to achieve efficient and practical deployment. Additionally, the interoperability of diverse IoT systems and standardisation of data formats pose additional complexities in designing a robust and interoperable blockchain-based provenance solution.

2.5.4. Patient-Driven Record Maintenance

Patient-driven record maintenance is a significant application of blockchain technology in the healthcare sector, focusing on empowering patients to take control of their medical records. This subsection explores the concept of patient-driven record maintenance based on the research paper

"Health record management through blockchain technology" as a foundation (Harshini et al., 2019).

Traditional healthcare systems often involve fragmented and centralised record-keeping processes, leading to challenges such as data silos, limited patient access, and privacy concerns. Patient-driven record maintenance, facilitated by blockchain technology, aims to address these issues by providing patients with ownership and control over their medical data.

The research paper highlights how blockchain technology can revolutionise the management of health records. By leveraging the decentralised and immutable properties of blockchain, patients can securely store their medical information on the blockchain, granting them sole ownership and control over their data. This patient-centric approach ensures privacy, data integrity, and accessibility.

In a patient-driven record maintenance system, healthcare data is stored in a blockchain-based distributed ledger, where each patient maintains control over their own data. Each interaction or modification of the health records is recorded as a transaction on the blockchain, creating an auditable and tamper-proof history of the data.

Blockchain technology enhances patient-driven record maintenance in several ways. Firstly, it eliminates the need for intermediaries and central authorities, enabling patients to directly manage and grant access to their medical records. This eliminates the dependence on healthcare providers or institutions for data access, streamlining processes and improving patient autonomy.

Secondly, blockchain's immutability ensures that patient data remains tamper-proof and transparent. Any changes or modifications made to the records are recorded on the blockchain, creating an auditable trail of data provenance. This enhances data integrity and reduces the risk of unauthorised alterations or data breaches.

Moreover, patient-driven record maintenance allows patients to share their medical data securely with healthcare providers, researchers, or other stakeholders. By utilising smart contracts, patients can define access permissions and consent requirements, ensuring privacy and data confidentiality. This enables efficient and secure data sharing for clinical trials, research, and healthcare collaborations.

However, challenges persist in implementing patient-driven record maintenance through blockchain technology. Interoperability remains a significant hurdle, as healthcare systems often employ diverse formats and standards for data representation. Efforts towards standardisation and data interoperability are crucial to ensure seamless integration and exchange of medical records across different blockchain platforms and healthcare providers.

2.5.5. Supply Chain Provenance Tracking

Supply chain provenance tracking is a crucial application of blockchain technology that aims to enhance transparency, traceability, and trust within supply chain networks. This subsection explores the concept of supply chain provenance tracking based on the research paper "TrustChain: Trust Management in Blockchain and IoT supported Supply Chains" as a foundation (Malik et al., 2019).

Supply chains involve complex networks of multiple participants, including suppliers, manufacturers, distributors, and retailers. The lack of transparency and visibility in traditional supply chain systems often leads to challenges such as counterfeit products, fraud, and inefficient processes. Blockchain technology offers a solution by providing a decentralised and immutable ledger that enables end-to-end tracking of products and components throughout the supply chain.

The research paper highlights the use of blockchain technology to create a trust-based system for supply chain provenance tracking. By leveraging blockchain's transparent and tamper-evident properties, stakeholders can record and verify every step of the supply chain journey, ensuring the authenticity and integrity of products and components.

In a supply chain provenance tracking system, each participant in the supply chain records key information, such as the origin, manufacturing process, quality checks, and transportation details, on the blockchain. These records form a distributed ledger that captures the entire lifecycle of a product, allowing for transparent and auditable traceability.

Blockchain technology enhances supply chain provenance tracking in several ways. Firstly, it provides a shared and decentralised platform that allows participants to securely exchange and access supply chain data. This eliminates the need for intermediaries and establishes trust among participants, fostering collaboration and information sharing.

Secondly, blockchain's immutability ensures that the recorded information remains tamper-proof and cannot be retroactively modified. Each transaction or event recorded on the blockchain becomes a permanent and verifiable part of the supply chain history, enabling stakeholders to validate the authenticity and integrity of products and components.

Moreover, blockchain enables the integration of Internet of Things (IoT) devices and sensors within the supply chain. IoT devices can generate real-time data, such as temperature, humidity, and location, which can be securely recorded on the blockchain. This integration enhances supply chain transparency and enables proactive monitoring and quality control.

However, challenges exist in implementing supply chain provenance tracking using blockchain technology. Interoperability and standardisation of data formats across different supply chain systems and blockchain platforms are crucial to ensure seamless integration and information sharing. Additionally, scalability concerns and the efficient handling of a large volume of supply chain data need to be addressed for practical deployment.

2.5.6. Non-Fungible Token

Non-Fungible Tokens (NFTs) form a revolutionary class of blockchain-based applications with significant implications in the digital asset world. Unlike fungible tokens, which are interchangeable and identical to each other, such as cryptocurrencies, non-fungible tokens are unique and distinguishable. They offer verifiable proof of ownership and authenticity for digital assets, secured via the decentralised nature of blockchain technology (Chohan, 2021).

One of the most pronounced use cases of NFTs is within the video gaming industry. Unlike traditional gaming environments, where game developers hold complete control over in-game assets, NFTs bring about a paradigm shift, giving users the power to create and trade in-game assets. These NFT-backed in-game assets extend beyond the confines of the gaming platforms, opening the door for enhanced gameplay experiences.

These user-generated assets are securely encoded as NFTs and can be traded on third-party platforms, such as online marketplaces. This approach transcends the traditional boundaries set by the gaming developers, allowing assets to be exchanged freely without their direct oversight or control. The trading of these assets is made possible due to the unique coding underlying NFTs, enabling seamless and transparent transactions while ensuring digital ownership and scarcity.

The value proposition of NFTs, especially in gaming, is anchored in the concept of digital scarcity. The limited availability of unique, user-generated assets heightens their value, driving their desirability and market worth. This underscores the increasing acceptance and potential of NFTs as a significant component in the landscape of blockchain-based applications.

2.5.7. Improving Data Exchange in Video Games

The exchange of data in video games is a critical aspect of enhancing gameplay experiences and enabling interoperability between different gaming platforms. This subsection explores the concept of improving data exchange in video games through blockchain technology, drawing upon the research paper "Towards blockchain interoperability: Improving video games data exchange" as a foundation (Besancon et al., 2019).

Video games generate vast amounts of data, including player profiles, in-game assets, achievements, and virtual currencies. However, the current centralised approach to data management in the gaming industry often results in data silos, limited data portability, and challenges in cross-platform interactions. Blockchain technology offers a decentralised and secure solution to address these issues and improve data exchange in video games.

The research paper highlights the potential of blockchain technology to enhance data exchange and interoperability in video games. By utilising blockchain, gaming data can be stored, shared, and verified in a transparent and tamper-proof manner, enabling seamless data transfer between different gaming platforms.

Blockchain technology improves data exchange in video games through several key mechanisms. Firstly, blockchain provides a decentralised and trustless environment, removing the need for intermediaries or centralised game servers. This allows players to have direct control and ownership of their in-game assets, ensuring transparency and reducing the risk of fraudulent activities.

Secondly, blockchain enables the creation and management of non-fungible tokens (NFTs) representing unique in-game assets. NFTs on the blockchain provide a verifiable and secure way to prove ownership, scarcity, and authenticity of digital items, enabling players to trade and exchange assets across different games and platforms.

Moreover, blockchain facilitates peer-to-peer transactions and microtransactions in video games. By utilising cryptocurrency tokens or smart contracts, players can engage in secure and efficient in-game transactions, such as buying, selling, and trading virtual items or accessing premium game content.

Blockchain technology also supports the concept of decentralised game platforms and marketplaces. These platforms leverage blockchain's transparency and security to create open ecosystems where developers, players, and content creators can collaborate, share resources, and monetise their creations. This fosters innovation and empowers players to participate in the game economy more directly.

However, challenges exist in implementing blockchain-based data exchange in video games. Scalability, as games generate large volumes of data and transaction costs, needs to be carefully considered. Additionally, ensuring the compatibility and interoperability of blockchain solutions across different games and platforms is essential for widespread adoption.

2.5.8. CloudArcade

CloudArcade is a blockchain-based cloud gaming system that combines the power of blockchain technology with cloud gaming infrastructure. This subsection explores the concept and significance of CloudArcade based on the research paper "CloudArcade: A blockchain empowered cloud gaming system" as a foundation (Zhao et al., 2020).

Cloud gaming has gained significant traction in recent years, allowing players to stream games from remote servers without needing high-end hardware. However, traditional cloud gaming platforms often rely on centralised infrastructure and face challenges such as data privacy, ownership of in-game assets, and transparency. CloudArcade aims to address these issues by leveraging blockchain technology.

The research paper introduces CloudArcade as a blockchain-powered cloud gaming system. By utilising blockchain technology, CloudArcade offers enhanced security, data privacy, and ownership of in-game assets, ultimately providing a more transparent and player-centric gaming experience.

One of the key features of CloudArcade is its utilisation of blockchain to ensure secure and verifiable transactions. Blockchain enables transparent and tamper-proof recording of in-game transactions, such as purchases, trades, and ownership transfers. This empowers players with true ownership of their in-game assets, as ownership records are stored on the blockchain and can be independently verified.

Additionally, CloudArcade leverages smart contracts, self-executing blockchain agreements, to automate various aspects of gaming transactions and interactions. Smart contracts enable secure and transparent execution of game-related processes, such as matchmaking, rewards distribution, and dispute resolution. This reduces the need for intermediaries and increases the efficiency and fairness of gameplay.

Moreover, CloudArcade utilises blockchain to enhance data privacy and security. By leveraging blockchain's decentralised nature, player data is securely stored and managed without relying on a centralised authority. This reduces the risk of data breaches and ensures that players have greater control over their personal information.

Blockchain also enables the creation and management of non-fungible tokens (NFTs) in CloudArcade. NFTs represent unique in-game assets, such as virtual items, characters, or achievements.

By utilising NFTs, players have verifiable ownership and can freely transfer or trade their investments across different games and platforms.

However, challenges exist in implementing CloudArcade and blockchain-based cloud gaming systems. Scalability and network latency are crucial to ensure smooth and responsive gameplay experiences. Integrating blockchain technology into existing cloud gaming infrastructures also requires careful planning and coordination.

2.6. Consensus Algorithm Comparison in Streamed Gaming Environment

In the realm of blockchain technology, various consensus algorithms have been developed to address the challenges of scalability, security, and energy efficiency. This section provides an overview and comparison of four prominent consensus algorithms: Bitcoin's Proof-of-Work (PoW) Scalability Problem, Proof-of-Work with Lightning Network, Proof-of-Stake (PoS), and Proof-of-Authority (PoA).

2.6.1. Bitcoin(Proof-of-Work) Scalability Problem

One of the notable challenges faced by the Bitcoin blockchain is its scalability problem. The Bitcoin blockchain operates on a proof-of-work (PoW) consensus algorithm, which requires a significant amount of computational power to generate new blocks. However, the scalability issue arises due to the increasing demands of transaction volume and the subsequent growth of the blockchain.

As stated in the research paper by (Poon and Dryja, 2016), it has been estimated that in order to achieve a transaction volume comparable to Visa's peak volume, the Bitcoin network would need to generate 8 gigabytes per Bitcoin block every ten minutes. If this transaction volume were to persist for a year, the total amount of data generated would exceed 400 terabytes.

The increase in data capacity poses a challenge as larger block sizes are required to accommodate more transaction data. However, the growing block size creates a dilemma. With larger blocks, fewer miners may possess the necessary computational resources, including sufficient bandwidth and storage capacity, to mine new blocks efficiently. This concentration of mining power leads to a more centralised network, which raises concerns about security.

The centralisation of mining power in the Bitcoin network introduces security risks. A higher degree of centralisation means that fewer entities have control over the validation and addition of new blocks. This concentration of power not only undermines the decentralised nature of the blockchain but also increases the vulnerability to potential attacks or manipulation.

Efforts to address Bitcoin's scalability problem have led to exploring various solutions. The Lightning Network, for instance, is an off-chain solution that aims to improve scalability and reduce transaction fees by conducting transactions outside of the main Bitcoin blockchain. By establishing payment channels between participants, the Lightning Network enables faster and cheaper transactions without burdening the main blockchain.

2.6.2. Proof-of-Work with Lighting Network

The Lightning Network (LN) has emerged as a scalable solution for Bitcoin instant payments and is already being utilised as a daily life payment method. In a video demonstration (https://www.youtube.com/watch?v=48uSd4eQfZs&t=69s), it was shown that the Lightning Network can process transactions in under one second, surpassing even the speed of Visa transactions. The Lightning Network's short transaction time has opened up possibilities for integrating Bitcoin into multiplayer online games.

Platforms like ZEBEDEE (https://zebedee.io/) have facilitated game developers in building games where players can earn or spend Bitcoins within the gaming environment. With the help of the Lightning Network and platforms like ZEBEDEE, game developers can easily incorporate cryptocurrencies into their games without the need to create their own cryptocurrency or worry about the safety and security issues associated with blockchain techniques.

In traditional streamed gaming trading systems, transactions are typically sent from the client device to the main server. The main server then verifies and approves the transactions before updating the game state and streaming the gaming video back to the client. In this process, the main server assumes control over all the trading activities and is responsible for verifying all the transactions, necessitating a substantial amount of computing power and network bandwidth to ensure acceptable trading delays.

However, the Lightning Network can serve as the main server to verify all transactions in a decentralised manner. The client device can directly communicate with the payment channel of the Lightning Network. Once the cloud confirms that a transaction has been accepted by the payment channel, it can update the game state and inform the player that the trading is complete. This decentralised approach alleviates the stress on the main server and distributes the trading process in a more efficient and decentralised manner.

Despite its advantages, the Lightning Network does have two main drawbacks when applied in a streamed gaming environment. As mentioned in section 2.2.8, players need to deposit a certain amount of Bitcoin into the payment channel when opening it. This on-chain action requires waiting until the mining process is complete and the Bitcoin main chain approves the transaction. If developers are unable to mask this waiting time through creative techniques, it may result in a less engaging user experience. In regular games, players are accustomed to waiting for game file downloads and installations, which provides an opportunity for developers to utilise that time for players to create the payment channel and wait for the mining process to finish.

However, unlike regular games, cloud games offer the advantage of not requiring players to wait for game file downloads and local disk setups. The setup time for a cloud game is typically much shorter than the mining time required for the Lightning Network (approximately 10 minutes, citation needed), making it challenging for cloud game developers to hide the mining time from players.

Additionally, the Hashed Time Lock Contract (HTLC) used in the Lightning Network introduces potential delays when routing is not optimal or when some nodes along the path are uncooperative. Delayed payments can cause the trading process to become unstable and impact real-time trading between players.

Moreover, the requirement for a certain amount of Bitcoin to be placed into the payment channel may confuse players. It necessitates special game design elements to guide players in finding a reasonable way to deposit money before playing or engaging in in-game trading.

In conclusion, the combination of Proof-of-Work with the Lightning Network offers a scalable solution for Bitcoin instant payments and has found application as a payment method in various real-life scenarios. The integration of the Lightning Network into streamed gaming environments presents opportunities for decentralised trading processes and improved transaction speeds. However, considerations such as on-chain waiting times, potential delays in HTLCs, and user confusion regarding Bitcoin deposits require careful attention, and thoughtful game design approaches to ensure a seamless and user-friendly experience.

2.6.3. Proof-of-Stake

Proof-of-Stake (PoS) is a consensus algorithm that offers several advantages, including low energy consumption and minimal computing power requirements. In a PoS blockchain, all nodes have the opportunity to become alternative validators, unlike traditional proof-of-work systems where only a select few miners have the computational power to validate new blocks. As mentioned in section 4.1, only the chosen validator needs to dedicate more computing power to approve and append the new block to the blockchain. Other alternative validators require minimal computing power to keep track of the blockchain. Players interested in becoming validators can utilise their loading time or away-from-keyboard (AFK) time to validate new blocks. When a player is selected as the validator for the next block, they can choose to participate by either dropping or computing. This approach increases computing power utilisation and contributes to the overall activity of the cryptocurrency.

In a streamed gaming environment, the client device may not have sufficient computing power to act as a validator. However, the cloud can assume this role on behalf of the client. The client device can operate as an alternative validator, utilising minimal computing power to keep track of the blockchain, while the cloud handles the validator process when selected to validate the next block. This distributed approach allows for greater participation and scalability in the blockchain network.

However, one of the main drawbacks of integrating a proof-of-stake blockchain into video games is the security issue. If a developer creates a new cryptocurrency or its own chain to support in-game trading, it becomes susceptible to a "51% attack." This attack occurs when an attacker controls more than 50% of the stakes on the network, enabling them to manipulate the blockchain. In a streamed gaming environment, if players are allowed to use the cloud for the validation process, it can lead to a more centralised verification system since most of the verification results will come from the cloud.

One possible solution to overcome this drawback is to utilise existing proof-of-stake blockchain platforms such as ETH2.0. However, the feasibility of implementing such platforms depends on their compatibility and ease of integration with game development frameworks and technologies.

In conclusion, the proof-of-stake consensus algorithm offers advantages such as low energy consumption and increased participation opportunities for validators. In a streamed gaming environment, the cloud can play a crucial role in assuming the validation process, while client devices act as alternative validators. However, the security issue of potential 51% attacks and the centralisation of verification results from the cloud must be carefully considered. Exploring established proof-of-stake blockchain platforms like ETH2.0 may offer potential solutions, but it requires assessing their suitability for game development purposes.

2.6.4. Proof-of-Authority

Proof-of-Authority (PoA) is a consensus algorithm that offers unique advantages and possibilities for integrating blockchain technology in the gaming industry, particularly in the context of cloud gaming providers that predominantly use distributed servers to deliver streamed gaming services. These distributed servers can serve as the initial authorised nodes in a blockchain network. While players have the ability to view the blockchain, only authorised players can participate in the verification and addition of new transactions to the blockchain. Developers have the flexibility to determine whether players can become authorised nodes or not. The Proof-of-Authority algorithm grants developers greater control over the block generation process, resulting in a more centralised trading system. However, this centralisation can serve as a stepping stone for transitioning the game trading system from a fully centralised main-server-based system to a decentralised blockchain-based system. As the number of trusted players who become authorised nodes increases, the blockchain becomes more decentralised, making it more challenging for attackers to manipulate the voting process.

The Proof-of-Authority algorithm utilises a voting-based scheme to add new transactions to the blockchain, ensuring a predictable and stable block generation process. Transactions are consistently accepted within a steady timeframe, providing the trading system with a stable trading latency. In a streamed gaming environment, a stable trading latency can be effectively concealed within animations or loading times. Developers can program calculations to determine the trading delay based on networking latency from authorised nodes, enabling them to dynamically choose how to hide the trading processing time. Moreover, the inherent time delay in streamed game videos, which occurs as data is transmitted from the cloud to the client device, can also serve as a cover for the trading delay. Transaction confirmation messages can be sent directly from the blockchain to the client device, allowing the client device to handle the user interface and inform players that the transaction has been confirmed.

In summary, the Proof-of-Authority consensus algorithm offers a range of possibilities for incorporating blockchain technology into the gaming industry. By utilising distributed servers as authorised nodes and implementing the Proof-of-Authority algorithm, developers gain greater control over the block generation process, albeit in a more centralised manner. However, this centralisation can serve as a catalyst for transitioning towards a decentralised blockchain-based trading system. The voting-based scheme of PoA ensures a predictable and stable block generation process, resulting in a steady trading latency that can be effectively concealed within animations, loading times, or the inherent time delay of streamed game videos. Developers can leverage the advantages of the Proof-of-Authority algorithm to create a seamless and secure trading experience for players, where transaction confirmations are efficiently handled by the blockchain and relayed to the client devices.

2.6.5. Comparison and Summary

In this section, we have examined and compared four prominent consensus algorithms and layer two solutions – Proof-of-Work (PoW), Proof-of-Work with Lightning Network, Proof-of-Stake (PoS), and Proof-of-Authority (PoA) – in the context of streamed gaming environments. Each algorithm offers unique advantages and presents specific considerations for integrating blockchain technology into the gaming industry.

PoW, as employed by Bitcoin, provides a secure and decentralised network. However, its scalability limitations and energy consumption make it less suitable for real-time trading and interactive gameplay. The Lightning Network addresses some of these challenges by facilitating faster and cheaper off-chain transactions. It introduces decentralisation and improves transaction speeds, but certain drawbacks, such as on-chain waiting times and potential delays in the HTLC, need to be managed.

PoS offers a more energy-efficient and scalable alternative, allowing for greater participation and lower computational requirements. It can be a viable solution for low-powered client devices in streamed gaming environments. However, the security issue of potential 51% attacks needs careful consideration. Integrating established PoS platforms, like ETH2.0, may provide enhanced security features and compatibility with game development frameworks.

PoA presents a centralised approach with increased control over the block generation process. It can act as a transitional solution for moving from fully centralised trading systems to decentralised blockchain-based systems. PoA ensures a predictable and stable block generation process, resulting in steady trading latency. However, careful attention must be given to security risks associated with centralisation and the distribution of authorised nodes.

To summarise, as Table2.1 shows, each consensus algorithm has its strengths and limitations in the context of streamed gaming environments. The table assigns a rating of 1 to 3 to each evaluation parameter, with 1 indicating a negative rating, 2 indicating an average rating, and 3 indicating a positive rating. The Lightning Network offers faster transactions and decentralisation, while PoS provides energy efficiency and greater participation. PoA balances control and decentralisation, acting as a transition towards decentralised trading systems. Developers must carefully evaluate their specific requirements and considerations to select the most suitable consensus algorithm for integrating blockchain technology into streamed gaming applications. For this project, the decision has been made to use the Proof of Stake (PoS) consensus algorithm for the final simulator.

	scalability	transactional latency	initialize time	decentralized	security	energy consumption
PoW	1	1	3	3	3	1
LN	2	3	1	3	3	3
PoS	3	2	3	3	2	3
PoA	3	2	3	1	1	2

Table 2.1 Consensus Algorithm Comparison

2.7. Blockchain Simulator

In the field of blockchain research and development, simulation tools play a crucial role in understanding and evaluating the behaviour and performance of blockchain systems. Several notable blockchain simulators have been developed, including Bitcoin-Simulator, CLoTH, PCNsim, BlockSim, and FobSim. These simulators provide researchers and developers with powerful platforms to explore different aspects of blockchain technology, such as consensus algorithms, scalability, payment networks, and integrated fog-blockchain systems.

2.7.1. Bitcoin-Simulator

To study and evaluate the security and performance aspects of Proof-of-Work (PoW) blockchains like Bitcoin, researchers and developers have created simulation tools known as Bitcoin simulators. These simulators aim to replicate the behaviour and dynamics of the Bitcoin network, allowing for in-depth analysis and experimentation.

One notable Bitcoin simulator is the "Bitcoin-Simulator", developed based on the research conducted in the paper titled "On the Security and Performance of Proof of Work Blockchains" in 2016 (Gervais et al., 2016). This simulator provides a valuable platform for investigating various aspects of the Bitcoin network, including security vulnerabilities, transaction confirmation times, and block generation processes.

The Bitcoin-Simulator enables users to simulate different attack scenarios, such as 51% attacks or double-spending attacks, to assess the resilience and security of the Bitcoin network under such circumstances. By altering parameters like the hash rate, block size, or transaction fee policies, researchers can analyse the impact of these factors on the overall performance and security of the blockchain system.

Furthermore, the simulator allows for the examination of transaction confirmation times and the scalability of the Bitcoin network. It provides insights into the relationship between network parameters, such as block size and block interval, and the speed at which transactions are confirmed. This information is crucial for understanding the limitations and potential improvements of the Bitcoin network in terms of transaction throughput and latency.

The Bitcoin-Simulator also aids in evaluating proposed modifications or enhancements to the Bitcoin protocol. Researchers can simulate the effects of changes like adjusting the difficulty level, implementing different transaction fee structures, or introducing alternative consensus mechanisms. This allows for the exploration of potential improvements to the scalability, security, and efficiency of the Bitcoin network.

Overall, the Bitcoin-Simulator serves as a valuable tool for studying and analysing the security and performance characteristics of PoW blockchains, specifically focusing on the Bitcoin network. By providing a simulated environment, researchers and developers can gain insights into the behaviour and limitations of the blockchain system, facilitating the development of strategies and solutions to address its challenges.

2.7.2. CLoTH

Researchers and developers have created CLoTH (Cloth: A Lightning Network Simulator) as a simulation tool to analyse and study the Lightning Network, a scaling solution for blockchain networks like Bitcoin. CLoTH specifically focuses on the Lightning Network and provides a platform for in-depth experimentation and analysis.

Derived from the research paper titled "CLoTH: A Lightning Network Simulator" published in 2021 (Conoscenti et al., 2021), CLoTH offers a comprehensive simulation environment to understand the behaviour and dynamics of the Lightning Network. It enables researchers and developers to explore various aspects of its performance, scalability, and security.

By utilising CLoTH, users can simulate different scenarios and configurations within the Lightning Network, gaining valuable insights into its routing mechanisms, payment channel establishment, fee policies, and transaction dynamics. Through the manipulation of parameters and variables, researchers can assess the impact of different factors on the network's behaviour and performance.

A notable feature of CLoTH is its ability to evaluate the scalability of the Lightning Network. Researchers can simulate large-scale Lightning Network deployments, examining the network's behaviour with an increasing number of nodes, channels, and transactions. This enables the identification of potential bottlenecks and congestion issues, providing insights into the network's overall scalability.

The CLoTH also facilitates the study of routing algorithms and strategies within the Lightning Network. Researchers can explore different routing protocols, fee management techniques, and efficient payment path strategies. This allows for the evaluation and comparison of various routing mechanisms, aiding in the optimisation of the network's performance and reliability.

Additionally, CLoTH supports the assessment of security and resilience within the Lightning Network. Researchers can simulate different attack scenarios, such as channel hijacking or payment fraud, to understand vulnerabilities and explore potential countermeasures. By enhancing security protocols and identifying weaknesses, CLoTH contributes to the ongoing improvement and secure implementation of the Lightning Network.

In summary, CLoTH serves as a powerful simulation tool for studying and analysing the Lightning Network. It offers a simulated environment where researchers and developers can explore its behaviour, scalability, and security. Through CLoTH, valuable insights into the Lightning Network's performance characteristics, routing strategies, and scalability limitations can be gained, driving advancements in this innovative technology.

2.7.3. PCNsim

In the context of studying and analysing Payment Channel Networks (PCNs), researchers and developers have developed PCNsim (Payment Channel Network Simulator), a simulation tool specifically designed for PCNs. PCNsim provides a flexible and modular platform for in-depth analysis and experimentation with payment channel networks.

PCNsim, introduced in the research paper titled "PCNsim: A Flexible and Modular Simulator for Payment Channel Networks" in 2022 (Fontes Rebello et al., 2022), offers a powerful simulation environment to understand the behaviour and dynamics of PCNs. It enables researchers and developers to explore various aspects of PCNs, including network topology, payment routing, channel management, and scalability.

One key feature of PCNsim is its flexibility and modularity. It allows users to customise and configure different parameters of PCNs, such as the number of nodes, network connectivity, payment policies, and channel capacities. This flexibility enables researchers to simulate and evaluate various PCN scenarios, assessing the impact of different factors on network performance and efficiency.

The simulator provides insights into payment routing algorithms and strategies within PCNs. Researchers can study and compare different routing protocols, fee management mechanisms, and strategies for finding efficient payment paths. This allows for the evaluation of routing efficiency, network congestion, and overall performance of PCNs.

Furthermore, PCNsim facilitates the investigation of channel management strategies in PCNs. Researchers can analyse the dynamics of channel opening, closing, and maintenance, considering factors such as transaction fees, channel lifetimes, and participants' behaviours. This provides valuable insights into channel utilisation, liquidity management, and the overall stability of PCNs.

PCNsim also aids in assessing the scalability of PCNs. Researchers can simulate large-scale PCN deployments, studying how the network behaves with an increasing number of nodes and channels. This enables the examination of scalability challenges, potential bottlenecks, and the impact of network growth on performance.

Overall, PCNsim serves as a valuable simulation tool for studying and analysing Payment Channel Networks. Its flexibility, modularity, and customisable features provide researchers and developers with a platform to explore various aspects of PCNs, including network topology, payment routing, channel management, and scalability. Through PCNsim, valuable insights into the behaviour and performance characteristics of PCNs can be gained, driving advancements in the design and implementation of efficient and scalable payment channel networks.

2.7.4. BlockSim

In the realm of studying and analysing blockchain systems, researchers and developers have created BlockSim, an extensible simulation tool designed specifically for blockchain systems. BlockSim serves as a powerful platform for conducting in-depth analysis and experimentation within the field of blockchain technology.

BlockSim, introduced in the research paper titled "BlockSim: An Extensible Simulation Tool for Blockchain Systems" in 2020 (Alharby and van Moorsel, 2020), provides researchers and developers with a comprehensive simulation environment to understand the behaviour, dynamics, and performance of blockchain systems. This versatile tool allows for the customisation and extension of various blockchain components, enabling the exploration of different blockchain architectures, consensus algorithms, and network topologies.

One key feature of BlockSim is its extensibility, allowing users to incorporate and test new blockchain protocols and algorithms. Researchers can simulate different consensus mechanisms, such as Proof of Work (PoW), Proof of Stake (PoS), or Byzantine Fault Tolerance (BFT), and analyse their impact on security, scalability, and performance. This flexibility enables the evaluation of various blockchain design choices and the comparison of different consensus algorithms.

The simulator provides insights into transaction processing and validation within blockchain systems. Users can analyse the impact of block size, block interval, and network latency on transaction throughput and confirmation times. By simulating different scenarios and configurations, researchers can study the trade-offs between decentralisation, security, and scalability in blockchain systems.

Furthermore, BlockSim enables the examination of network connectivity and topology within blockchain networks. Users can model different network structures, such as fully connected, random, or scale-free networks, to assess their impact on information propagation, consensus, and overall system resilience. This allows for the exploration of network robustness and the evaluation of potential vulnerabilities.

BlockSim also supports the analysis of blockchain performance under various attack scenarios. Researchers can simulate double-spending attacks, 51% attacks, or Sybil attacks to understand their impact on the integrity and security of the blockchain system. This aids in identifying potential vulnerabilities and devising countermeasures to enhance the system's resilience.

Overall, BlockSim serves as a versatile and extensible simulation tool for studying and analysing blockchain systems. It provides researchers and developers with a customisable environment to explore different blockchain architectures, consensus algorithms, and network topologies. Through the use of BlockSim, valuable insights into the behaviour, performance, and security aspects of blockchain systems can be gained, driving advancements in the design and implementation of efficient and secure blockchain solutions.

2.7.5. FobSim

Researchers and developers have developed FobSim, an extensible open-source simulation tool to explore and analyse integrated fog-blockchain systems. FobSim is a versatile platform for conducting detailed analysis and experimentation within fog computing and blockchain integration.

FobSim, introduced in the research paper titled "FobSim: An Extensible Open-Source Simulation Tool for Integrated Fog-Blockchain Systems" in 2021 (Baniata and Kertesz, 2021), provides researchers and developers with a comprehensive simulation environment to understand the behaviour and performance of fog-blockchain systems. This tool explores various aspects of these integrated systems, including resource management, data processing, and consensus mechanisms.

One key feature of FobSim is its extensibility, allowing users to customise and extend different components of fog-blockchain systems. Researchers can simulate diverse fog computing architectures, such as hierarchical or decentralised models, and integrate them with various blockchain frameworks. This flexibility enables the evaluation of different deployment scenarios and the exploration of optimal configurations for fog-blockchain integration.

The simulator provides insights into resource management and allocation within fog-blockchain systems. Users can analyse the impact of fog node distribution, computing power, and network connectivity on system performance and efficiency. By simulating different scenarios and configurations, researchers can evaluate resource utilisation, load balancing, and overall system scalability.

Furthermore, FobSim enables the examination of data processing and storage mechanisms within integrated fog-blockchain systems. Researchers can simulate various data distribution strategies, caching algorithms, and data replication schemes to evaluate their impact on latency, data availability, and reliability. This facilitates the investigation of optimal data management techniques for fog-blockchain systems.

FobSim also supports the analysis of consensus mechanisms and their impact on system performance. Users can simulate different consensus algorithms, such as Proof of Work (PoW), Proof of Stake (PoS), or Practical Byzantine Fault Tolerance (PBFT), and assess their efficiency, scalability, and fault tolerance. This allows for the evaluation of consensus protocol selection for fog-blockchain systems.

Overall, FobSim serves as an extensible open-source simulation tool for studying and analysing integrated fog-blockchain systems. It provides researchers and developers with a customisable environment to explore various aspects of fog computing and blockchain integration, including resource management, data processing, and consensus mechanisms. Through the use of FobSim, valuable insights into the behaviour, performance, and optimisation of fog-blockchain systems can be gained, driving advancements in the design and implementation of efficient and resilient integrated systems.

2.7.6. Comparison and Summary

When comparing these simulation tools, several aspects stand out. Firstly, "On the Security and Performance of Proof-of-Work Blockchains" focuses specifically on analysing the security and performance aspects of proof-of-work (PoW) blockchains. It provides researchers with insights into the impact of various parameters on PoW blockchains. On the other hand, "CLoTH: A Lightning Network Simulator" and "PCNsim: A Flexible and Modular Simulator for Payment Channel Networks" concentrate on specific scalability solutions for blockchain systems, namely the Lightning Network and payment channel networks, respectively. These simulators allow researchers to analyse the behaviour and performance of these specific solutions.

In terms of generality and extensibility, "BlockSim: An Extensible Simulation Tool for Blockchain Systems" and "FobSim: An Extensible Open-Source Simulation Tool for Integrated Fog-Blockchain Systems" offer broader capabilities. BlockSim provides a flexible framework for modelling and evaluating various blockchain protocols, consensus mechanisms, and transaction processing strategies. It allows researchers to assess the performance, scalability, and security of different blockchain architectures. FobSim, on the other hand, focuses on the integration of fog computing and blockchain technologies. It enables researchers to study the behaviour and performance of fog-blockchain systems in edge computing environments.

Another aspect to consider is the release year of these simulation tools. "On the Security and Performance of Proof of Work Blockchains" was published in 2016, making it one of the earlier contributions in this field. The more recent simulators, such as "CLoTH: A Lightning Network Simulator" (2021), "PCNsim: A Flexible and Modular Simulator for Payment Channel Networks" (2022), "BlockSim: An Extensible Simulation Tool for Blockchain Systems" (2020), and "FobSim: An Extensible Open-Source Simulation Tool for Integrated Fog-Blockchain Systems" (2021), benefit from advancements in blockchain research and technology.

Since in 2.6.5 the decision has been made to focus on Proof-of-Stake (PoS) in this simulation, the development of the simulator will be based on the foundation provided by FobSim.

2.8. Summary

This chapter has provided a comprehensive exploration of blockchain technology and its potential applications in the gaming industry, with a specific focus on streaming gaming and the provenance tracking of in-game virtual items. The discourse started with a deep dive into the core principles of blockchain technology, highlighting its decentralised nature and the pivotal role of distributed ledger technology. This technology, secured by robust cryptographic principles, ensures the immutability and transparency of transactions.

The concept of streaming gaming was introduced, and the challenges it poses were discussed. Particularly, the complications related to the management of in-game assets in streaming gaming platforms and the difficulties in controlling and tracking the trade of these assets have been identified as significant challenges. Blockchain technology, with its decentralised and immutable characteristics, has been proposed as a viable solution for these problems.

Furthermore, this chapter sheds light on the consensus algorithms that drive the operation of blockchain systems, maintaining their data integrity and facilitating efficient functioning. The two major algorithms, Proof of Work (PoW) and Proof of Stake (PoS), were analysed in depth. While PoW has been foundational in the operation of several blockchain systems, it is noted for its substantial energy requirements. PoS, however, is an energy-efficient alternative that presents promising potential in terms of scalability and security. Hence, this thesis chose to concentrate on this algorithm for the forthcoming study.

In the context of virtual economies, this chapter elucidated the notion of in-game virtual items and the related challenges, particularly those concerning provenance and ownership. The potential of blockchain technology as an effective solution to these problems was explored. Non-fungible tokens (NFTs), uniquely identifiable cryptographic tokens, emerge as a promising avenue for representing virtual items on the blockchain, thereby ensuring their uniqueness and facilitating provenance tracking.

As a precursor to the development of a simulation tool, this chapter examined five significant existing blockchain simulation tools: Bitcoin-Simulator, CLoTH, PCNsim, BlockSim, and FobSim. Each tool has its unique strengths, modelling and analysing various aspects of blockchain systems. Bitcoin-Simulator and CLoTH provide detailed analysis of PoW blockchains and Lightning Network, respectively, whereas PCNsim offers a modular platform for Payment Channel Networks' analysis.

Among the tools, BlockSim offers extensibility, providing a flexible platform to experiment with different blockchain protocols, consensus algorithms, and transaction processing strategies. FobSim, the most recent tool in this series, is an open-source solution designed to simulate the integration of fog computing and blockchain technologies. Considering this thesis's emphasis on PoS, the decision was made to construct the proposed simulator on the foundation provided by FobSim.

In conclusion, this background chapter has set the stage for this thesis, highlighting the complexities of blockchain technology, the potential of PoS as a consensus algorithm, the challenges and potential solutions for asset management and provenance tracking in streaming gaming, and the importance of simulation tools. This foundational understanding will guide the ensuing steps towards the development of a PoS-based blockchain simulator for the efficient tracking of the provenance of in-game virtual items, thereby contributing to the evolution of virtual economies in the gaming industry.

Chapter 3. Experimentation

3.1. Focused Requirements For In-Game Trading Transactions

This section establishes the groundwork for grasping the distinctive requirements and challenges associated with in-game trading transactions. It emphasizes the role of transaction delay, a pivotal aspect that can greatly affect the gaming experience. The conversation concedes that the acceptable latency might differ based on the game's nature and the importance of real-time trading. This portion acts as a cornerstone for the ensuing design and assessment of a blockchain-centric system intended to fulfil these prerequisites and elevate the in-game trading experience.

3.1.1. Transaction Delay

Transaction delay refers to the amount of time it takes for a transaction to be processed and completed within a game's trading system. This delay is a crucial aspect in ensuring smooth and responsive trading experiences for players within the game. The acceptable delay for in-game transactions can vary depending on the type of game and the significance of real-time trading.

In fast-paced games such as multiplayer shooters or competitive esports titles, it is crucial to have minimal transaction delays to maintain the fluidity and responsiveness of trading interactions. Players engaged in intense battles or time-sensitive gameplay scenarios rely on quick transaction processing to react swiftly to changing game situations.

On the other hand, slower-paced games like turn-based strategy or simulation games may have more flexibility in terms of delay requirements. These games are typically not reliant on real-time interactions, allowing players to tolerate slightly longer transaction processing times without significant impact on their overall gameplay experience. However, it is still important to ensure that transaction delays do not become excessively burdensome, as this can lead to player frustration and hinder the overall trading ecosystem.

Game developers and system architects need to carefully analyze the gameplay dynamics of their specific game and determine appropriate transaction delay thresholds that strike a balance between responsiveness and the overall game experience. By optimizing transaction processing algorithms and leveraging efficient blockchain technologies, it is possible to minimize transaction delays and enhance the in-game trading experience for players.

3.1.2. Pending Queue Size

The size of the pending queue is a critical consideration in the design and implementation of a trading system for in-game transactions. It refers to the maximum number of transactions that the trading system can accommodate simultaneously. The size of the pending queue directly impacts the system's capacity to handle concurrent transactions and ensures smooth and efficient trading interactions.

Game developers must carefully determine an appropriate size for the pending queue based on various factors. One important consideration is the expected trading volume within the game. If the game experiences high trading activity, a larger pending queue size may be necessary to prevent transaction backlogs and delays. On the other hand, games with lower trading volumes may require a smaller pending queue size to optimize system resources.

System resources are another important factor to consider. The pending queue size should align with the available computational power, memory, and storage capacity of the trading system. Insufficient resources can lead to performance issues and hinder the overall responsiveness of the trading system.

To handle overflow situations when the pending queue exceeds its capacity, developers must implement appropriate strategies. These strategies can include prioritization mechanisms, where certain types of transactions or higher-value transactions are given precedence in processing. Temporary storage solutions can be employed to store excess transactions until space becomes available in the pending queue. Additionally, queuing mechanisms can be implemented to ensure fair and orderly processing of transactions once sufficient resources are freed up.

By carefully managing the pending queue size and implementing effective overflow strategies, game developers can ensure that the trading system can handle varying transaction volumes while maintaining the desired level of responsiveness. This contributes to a seamless and efficient in-game trading experience for players.

3.1.3. Average Injection Rate

The average injection rate is a crucial factor to consider when designing and optimizing a trading system for in-game transactions. It refers to the average number of transactions processed per unit of time within the game's trading system. This metric becomes particularly relevant in massively multiplayer online games (MMOs) or games with a large player population where a significant volume of trading activities occur.

The required average injection rate can vary depending on several factors. The scale of the game, including the number of active players and the overall trading activity, plays a significant role. Games with a larger user base and extensive trading systems may require a higher average injection rate to effectively handle the increased transaction volume.

Additionally, the importance of trading to the overall gameplay experience influences the desired average injection rate. In games where trading is a central aspect and directly impacts player progression, a higher injection rate might be necessary to ensure a smooth and engaging trading experience. Conversely, games where trading plays a lesser role may have more relaxed requirements in terms of the average injection rate.

Game developers must carefully analyze the anticipated trading patterns and player behaviour to determine the appropriate average injection rate. They can optimize the trading system by fine-tuning the system's capacity, resource allocation, and transaction processing algorithms to meet the required injection rate.

By ensuring that the trading system can handle the expected average injection rate, game developers can provide players with a seamless and efficient trading experience. This contributes to player satisfaction and enhances the overall dynamics of the virtual economy.

3.1.4. Peak Injection Rate

The peak injection rate is a crucial factor to consider when designing a trading system for in-game transactions. It represents the maximum number of transactions that the trading system can handle during periods of high activity or peak player engagement. These peaks in transaction volume often occur during special events, new content releases, or when a game experiences a surge in popularity.

During these peak periods, it is essential for the trading system to efficiently process the significant influx of transactions. The system must maintain its responsiveness and ensure a seamless trading experience for players, even under high-load conditions. Performance degradation or service interruptions should be minimized or avoided altogether.

To prepare for peak injection rates, game developers must carefully anticipate and plan for these periods of heightened activity. They need to assess the expected transaction volume based on factors such as the game's popularity, promotional events, and content releases. By accurately predicting peak injection rates, developers can ensure that the trading system is designed to handle the maximum transaction load without compromising its performance.

Scalability and stability are key considerations in addressing peak injection rates. The trading system should be designed with the necessary infrastructure and resources to scale up and down as needed during peak periods. This may involve implementing load-balancing mechanisms, optimizing server capacity, and leveraging cloud-based solutions to accommodate the increased transaction volume.

Additionally, developers must consider techniques such as caching, queuing, and prioritization to efficiently handle the high transaction influx during peak times. These strategies help manage the transaction flow, prioritize critical transactions, and ensure that all transactions are processed in a timely manner.

By effectively addressing peak injection rates, game developers can create a robust and resilient trading system that can handle the demands of high player engagement. This enhances the overall player experience, fosters a thriving in-game economy, and contributes to the success of the game.

3.2. Commuication Model

Figure 3.1 provides a visual representation of the functioning of a blockchain-based trading system specifically tailored for cloud gaming. In this system, users utilize cryptocurrencies to acquire services within the game or assets in the game environment. Once initiated, this transaction is sent over to the overarching blockchain network.

Upon receiving confirmation that the transaction has been duly recorded on the blockchain, which hosts the gaming service, proceeds to update the current state of the game. Subsequently, this updated gaming environment is streamed back to the user. One of the unique advantages of such a cloud gaming service is its ability to mitigate issues related to the unauthorized copying or piracy of local game data, an issue prevalent in conventional gaming systems. Moreover, the blockchain aspect of the system allows for meticulous tracking and recording of all transactions, thus ensuring a high degree of transparency.

The architecture of this model is illustrated in Figure 3.2. The model comprises a streaming server, which sends requests and fetches the requisite gaming data from data servers. These data servers are repositories where content creators and game developers store valuable gaming assets and games themselves. Each piece of content stored has a unique content ID that is usable within the blockchain system, which in turn enables the content's effective tracking and verification.

Players in this system have the ability to purchase the content directly using the blockchain network, thus ensuring security and transparency. Once the cloud system receives transaction confirmations from the blockchain network, it seamlessly streams the updated game state to the players. This ensures players always have access to the most recent game state, contributing to an uninterrupted and enhanced gaming experience.

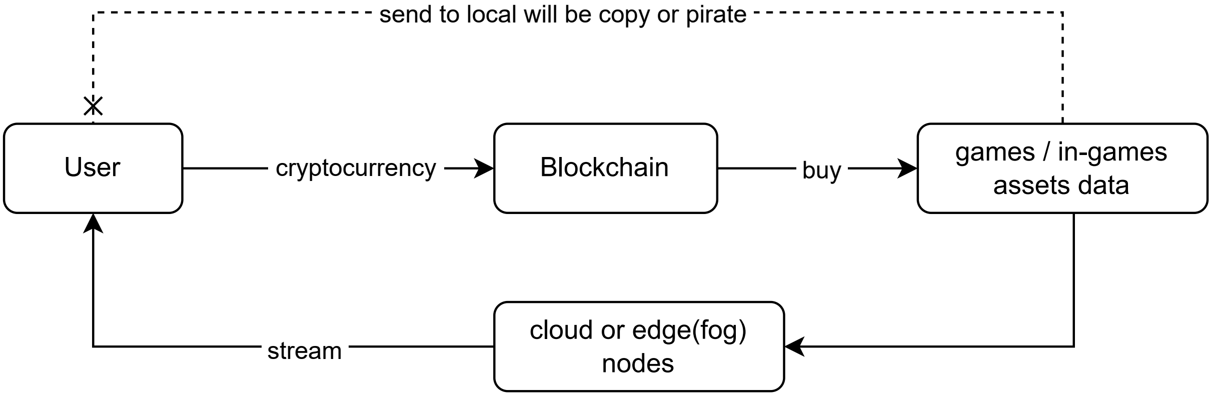
send to local will be copy or pirate
User
cryptocurrency
Blockchain
buy
games / in-games assets data
cloud or edge(fog) nodes
stream

Figure 3.1 communication model

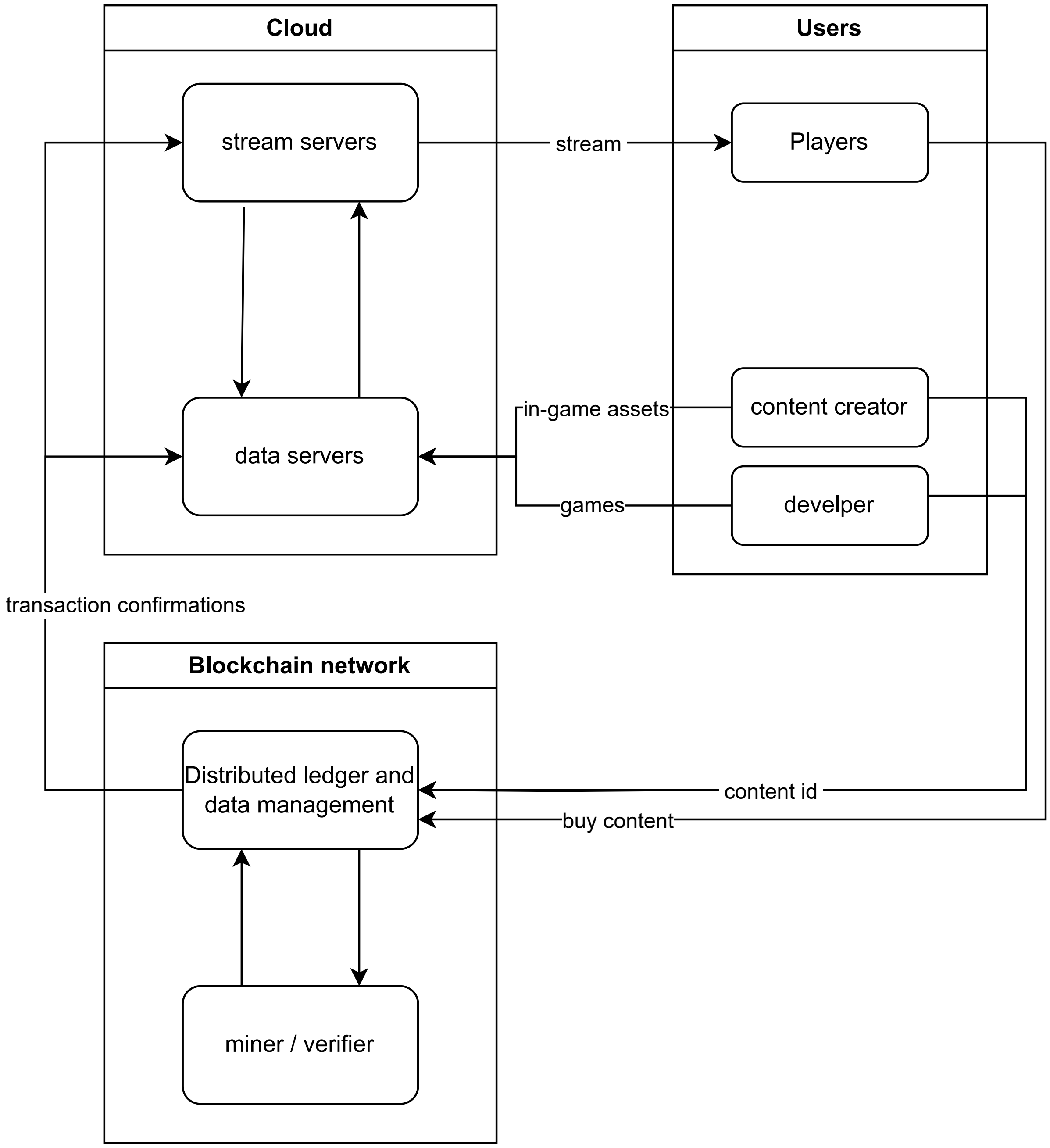

Figure 3.2 structure of communication model

3.3. Simulator Design and Implementation

The simulator, while modelled on the framework of FobSim as articulated in Baniata's 2021 research (Baniata and Kertesz, 2021), presents significant advancements in several domains. An intricately designed system has been integrated to precisely compute simulation time. Data collection mechanisms from networking systems are embedded, promoting a comprehensive understanding of various network behaviours and their performance metrics. Furthermore, a detailed recording system for simulation data has been incorporated, aiding in more thorough data analyses and decision-making processes. The simulator also encompasses features to handle cloud gaming transactions, thus broadening its application spectrum.

3.3.1. Refine Time Calculation

The updated design disaggregates the overall simulation time into several key components, thereby allowing for a more detailed analysis of the system's performance. Each of these time components is visually represented and explained in figure 3.3:

- T1: block prepare time - This is the amount of time taken to decide and notice the validator to be ready to generate the next block.
- T2: block generation time - This refers to the time spent by the validator node to generate a new block.
- T3: block transfer time - This is the time required to propagate a fully formed block across the network.
- T4: transaction pending time - This is the duration for which a transaction waits in the pool before it gets included in a block on the blockchain.
- T5: block interval time - This refers to the time between the generation of consecutive blocks in the blockchain.

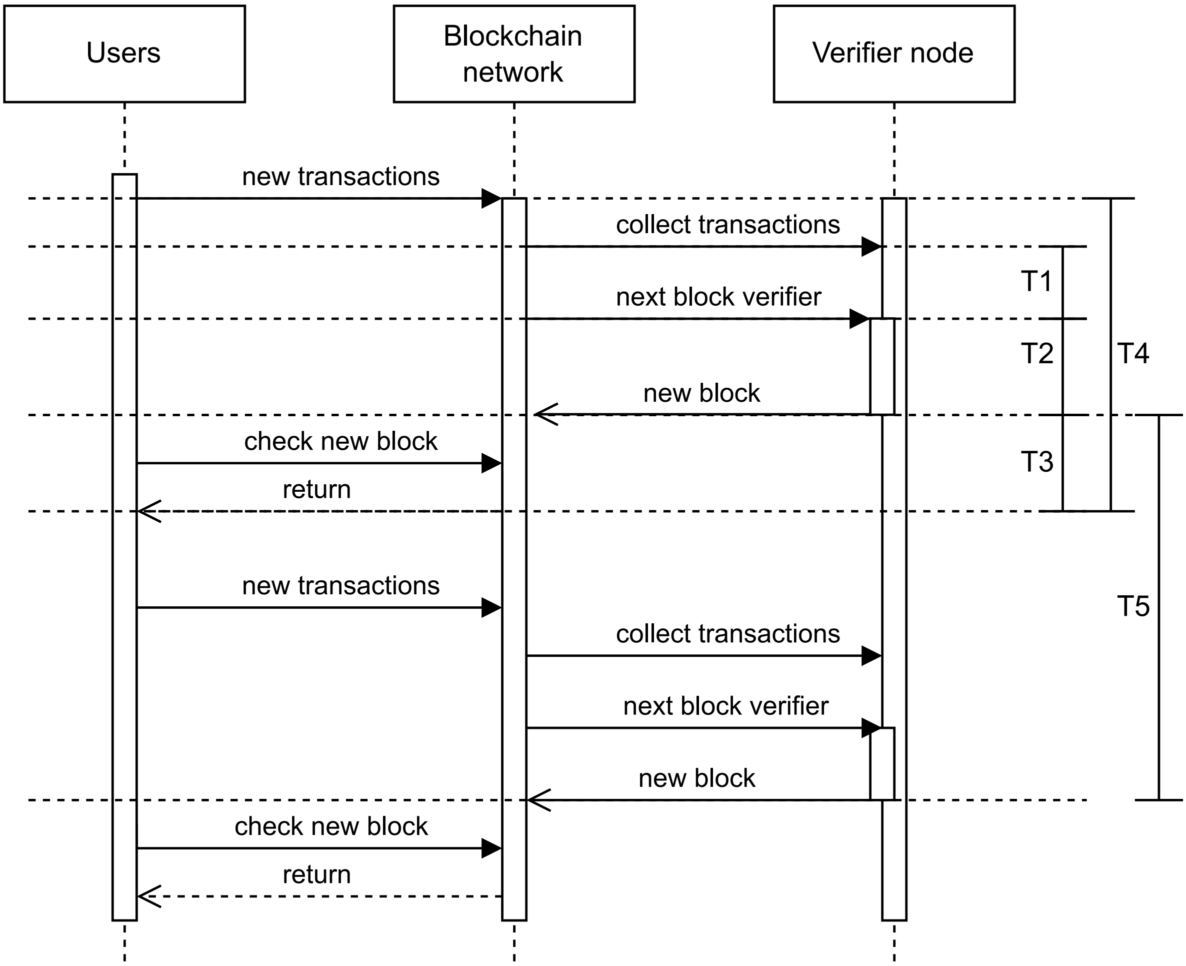

Figure 3.3 sequence diagram and time calculation

3.3.2. Networking Data Calculation

The current implementation of the simulator does not assume any specific communication protocol, such as TCP or UDP, when modeling interactions between nodes. Instead, it relies on an abstracted average communication time per node. While this approach simplifies the model, it limits the ability to simulate more realistic networking scenarios, such as variable latencies, packet loss, or protocol-specific behaviors.

In the communication model under consideration, the most significant delay arises during the specific period of data transfer, labelled as T3 in Figure 3.3. This delay occurs when a verifier node broadcasts a validated block, and other nodes subsequently download this newly verified block.

Each transaction encapsulates information about the previous owner, the incoming owner, and the item in question. This item could be a standard in-game asset or a non-fungible token (NFT). The specific layout of these transactions is illustrated in Figure 3.4.

The block and the overarching blockchain structures have been preserved from their initial setup. Figures 3.5 and 3.6 provide clear depictions of the structure of a block and the overall blockchain, respectively.

To assess the time taken for network data transfer, we have gathered the upload and download data usage statistics from blockchain nodes. These data points enable us to compute the network transfer time accurately.

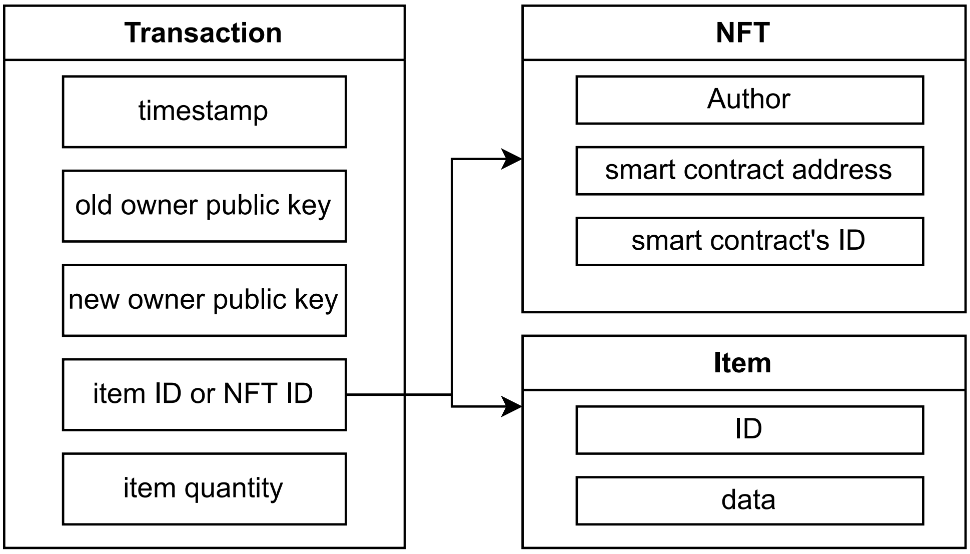

Figure 3.4 transaction structure

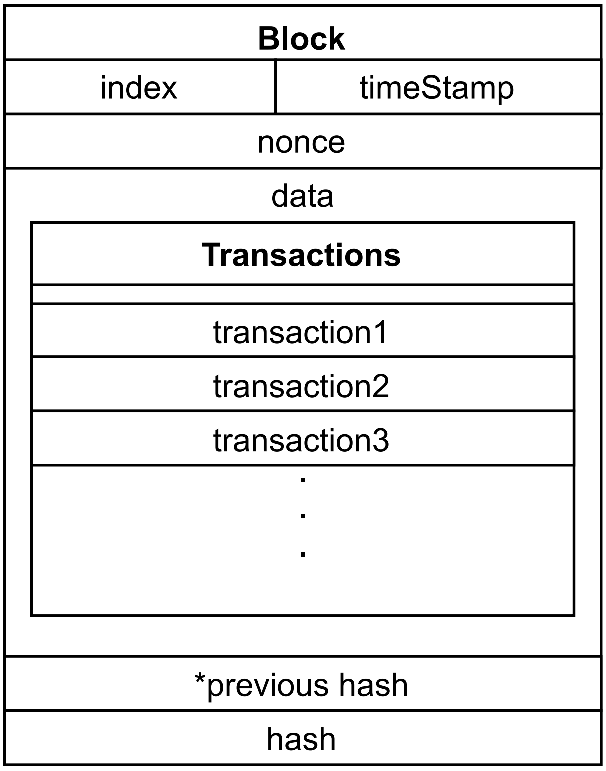

Figure 3.5 block structure

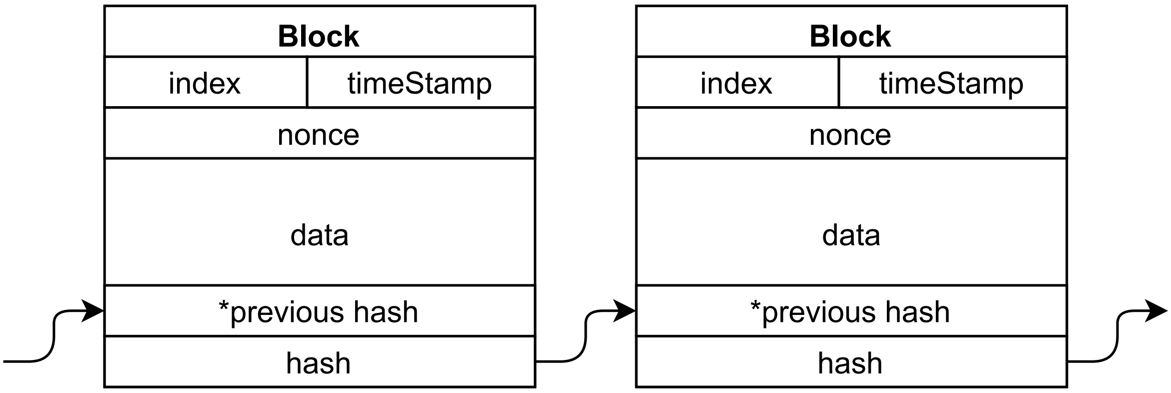

Figure 3.6 blockchain structure

3.3.3. Transaction Pending Queue and Inject Transactions

The original design of the transaction memory pool, which functioned as a multiprocessing queue, has been developed into a standard queue that serves as a transaction pending queue. This queue can introduce random new transactions. It also allows for determining the waiting time for the longest-pending transaction that remains in the queue.

The system's original design for block management involved pre-calculating the number of blocks based on the total number of transactions and the number of transactions per block. However, due to the addition of transaction injection, the system now accommodates a dynamic number of blocks.

3.3.4. Data Collection

The configurations and results for each simulation run will be computed and stored in '.csv' files. These configurations encompass:

- No. of users: the count of user nodes.
- No. of miners: the count of miner nodes.
- Initial No. of tx: the count of transactions at system initialization.
- Upload bandwidth(KB/S): the total upload speed of all nodes in the blockchain network.
- Download bandwidth(KB/S): the total download speed of all nodes in the blockchain network.
- Queue limit: the maximum limit for pending transactions. A run is considered failed when the number of pending transactions surpasses this limit.
- Injection rate(per sec): the quantity of transactions added to the pending transaction queue every second.
- Tx per block: the highest possible number of transactions that a single block can hold.

The results consist of:

- Final block count: the total number of blocks produced at the end of a simulation run.
- Failure time(secs): the duration a simulation continues until the pending transactions surpass the queue limit, leading to a failed run. This value will be -1 if the simulation doesn't fail.

- Average block time(ms): the mean time interval between blocks.
- Average upload time(ms): the mean block upload time for each node.
- Average download time(ms): the mean block download time for each node.
- Total upload time(sec): the sum of all block upload times for each node.
- Total download time(sec): the sum of all block download times for each node.
- Simulated time(sec): the duration of the simulation.
- Elapsed time(secs): the real-world time spent on a single simulation run.

3.3.5. Simulation Flow

The simulation flowchart in Figures 3.7 represents the sequential operational logic of the blockchain simulation, beginning with the initialization phase and advancing through transaction management and block validation. The process is initiated by setting up the simulation environment, which involves the sequential initialization of user nodes, transaction pools, miner nodes, and the genesis block. Following this setup, the system evaluates whether the transaction pool is empty. If no transactions are present, the simulation concludes successfully. Conversely, if transactions are detected, they are collected from user nodes, with the quantity determined by the delta time and transaction injection rate, which dictate the pace of transaction addition to the pool over time.

The volume of transactions collected is subject to variation based on different game genres, as each genre may exhibit distinct user counts and transaction injection rates. This variability is essential to accurately simulate interactions within the blockchain network, capturing the inherent dynamics of diverse gaming environments. The simulation then assesses whether the transaction pool has reached or exceeded its capacity. In cases of overflow, the simulation is deemed to have failed, signaling that the number of pending transactions has surpassed the designated queue limit. If the transaction pool remains within capacity, a block validator is selected from the pool of miner nodes to generate the next block. Once created, the block is propagated to all participating miners and users, thereby completing a single simulation cycle. The process subsequently reiterates, continuing until either the transaction pool is emptied or an overflow condition is encountered.

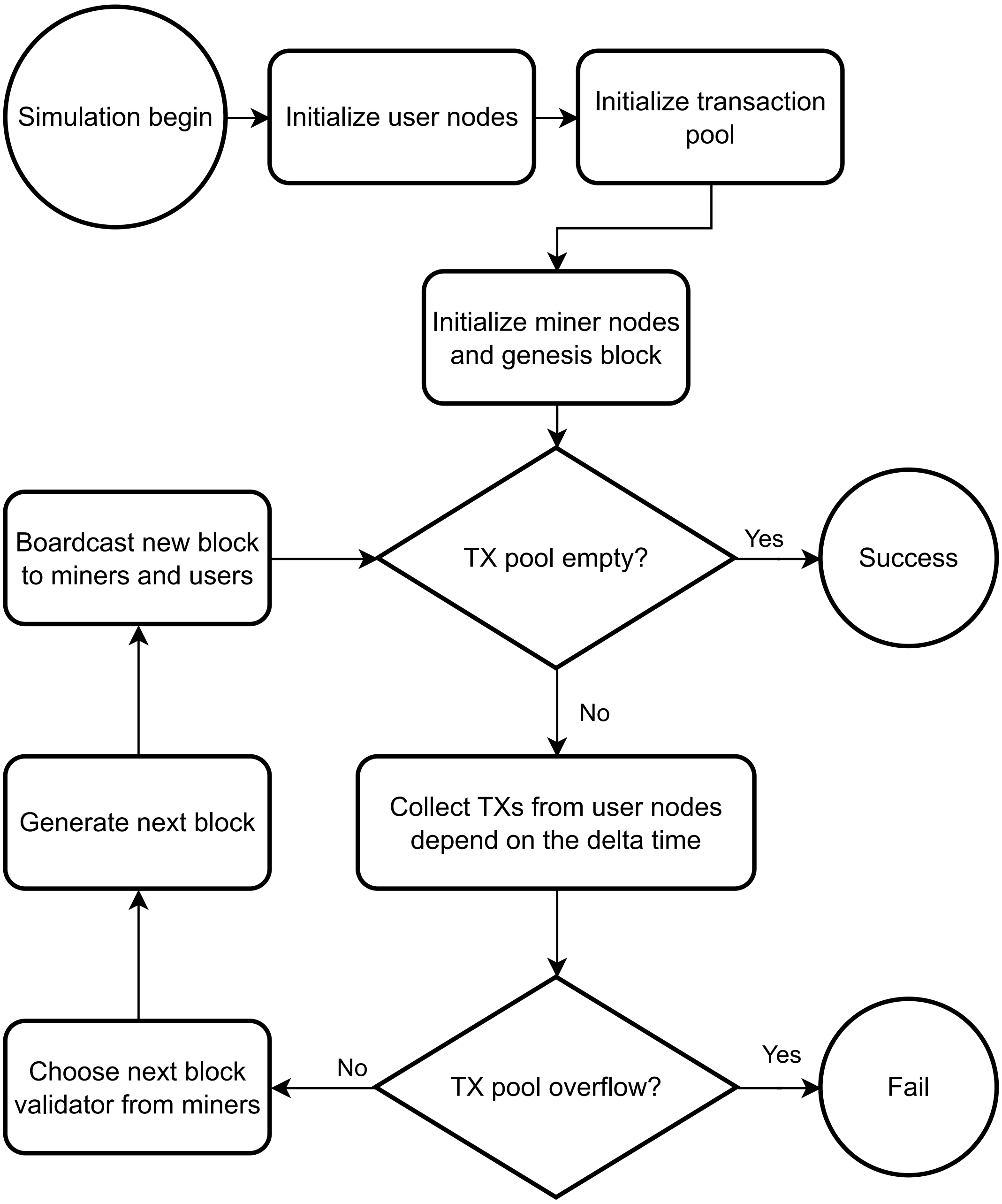
Simulation begin
Initialize user nodes
Initialize transaction pool
Initialize miner nodes and genesis block
Boardcast new block to miners and users
TX pool empty?
Yes
Success
No
Generate next block
Collect TXs from user nodes depend on the delta time
No
Choose next block validator from miners
TX pool overflow?
Yes
Fail

Figure 3.7 simulator flowchart

3.4. Analysis Tools

This section will delve into the specific tools and methodologies used to analyze the data collected during the simulation. The tools discussed include extra data tables, curve fitting, and the bisection method, each playing a unique role in the analysis process. The use of these tools ensures a comprehensive and accurate understanding of the system's performance under various conditions.

3.4.1. Extra Data Tables

The experiment repeatedly conducts each simulation setting to gather an expansive dataset for in-depth analysis. This data is used to generate several different tables, each offering unique insights into the performance and reliability of the system under various conditions.

Failrate Table

The fail rate table documents the number of unsuccessful and successful runs for each unique setting and then calculates the fail rate for all given injection rates. This enables researchers to identify configurations that are prone to failure, helping them optimize the system for better reliability.

Max Injection Rate Table

The maximum injection rate table ascertains the highest injection rate that maintains the fail rate below a specified limit. By default, the fail rate thresholds are set at 0%, 5%, and 10%. The maximum injection rate at a 0% fail rate also indicates the transaction handling rate of the system under current settings. This information is crucial for identifying system performance thresholds and preemptively mitigating potential system overloads.

Average Block Time Table

The average block time table computes the weighted average block time for each configuration. This is achieved using the average block time for a single run and the number of generated blocks for that run. This table offers insights into the speed at which blocks are generated and processed under different conditions.

3.4.2. Curve Fitting

Curve fitting is the process of constructing a curve or a mathematical function that has the best fit to a series of data points (Arlinghaus, 1994). The objective of curve fitting is to define a curve that describes the behavior of a system or a trend within a dataset. The constructed curve allows us to predict the behavior of the system for inputs that haven't been observed yet.

Curve fitting often involves the use of optimization algorithms to find the set of parameters that minimize the difference between the predicted outputs of the function (i.e., the curve) and the actual outputs in the dataset.

Logarithmic Curve Fitting

Logarithmic curve fitting is used when data demonstrates a rapid initial change that slows over time. This situation is common in many natural and biological processes, such as the cooling of an object in a warmer environment or the growth of a population.

A logarithmic function has the general form: $y = a + b * ln(x) + c * x$. The parameter a represents the y-intercept, b adjusts the vertical stretch and direction of the logarithm, and c adjusts the slope of the linear term.

To fit a logarithmic curve to data, we determine the parameters a, b, and c that minimize the difference between the function's output and the actual data points. This can be done with optimization algorithms like the least-squares method, which minimizes the sum of the squares of the residuals (the differences between the actual and predicted values).

Exponential Curve Fitting

Exponential curve fitting is applicable when data exhibits exponential growth or decay - this is common in many scientific, financial, and demographic contexts.

An exponential function has the general form: $y = a + b * x + c * exp(d * x)$. The parameter a represents the y-intercept, b adjusts the slope of the linear term, c influences the vertical stretch of the exponential function, and d determines the base of the exponent.

Fitting an exponential curve to data also involves determining the parameters a, b, c, and d that minimize the difference between the function's output and the actual data points. This is often a more complex process than fitting a linear or logarithmic curve due to the nonlinear nature of the exponential function.

For both types of fitting, once the optimal parameters are found, the resulting function can be used for data interpolation or prediction. The quality of the fit can be assessed using measures like R-squared, which gives the proportion of the variance in the dependent variable that is predictable from the independent variable(s).

R-squared Score

In statistics, the coefficient of determination, also known as R-squared, is a measure that assesses how well the model approximates the real data points. An R-squared of 100% indicates that all changes in the dependent variable are completely explained by changes in the independent variable(s).

In the context of curve fitting, the R-squared score tells you how closely the fitted values follow the trend of the actual data values. An R-squared score of 1 or close to 1 indicates that the model provides a good fit to the data.

3.4.3. Bisection Method To Solve Equation

The bisection method is a root-finding algorithm that applies to any continuous function for which one knows two values with opposite signs. The method consists of repeatedly bisecting the interval defined by these values and then selecting the subinterval in which the function changes sign and, therefore, must contain a root.

Here's the basic concept:

1. Start with an interval that contains a root (this interval can be found by analyzing the graph or using the intermediate value theorem).
2. Calculate the midpoint (c) of the interval.
3. If the function at c is zero, then c is the root of the solution.
4. If not, the function at c is either positive or negative. Replace either the lower bound (a) or upper bound (b) of the interval with c, such that the function at the new bounds a and b is still of opposite sign.
5. Repeat the process until the value of c is sufficiently accurate.

3.5. Simulation Detail

The aim of the simulation experiment is to formulate a mathematical model capable of accurately forecasting the peak transaction injection rate into the system. The model factors in variables such as block sizes and network bandwidth of validators to offer insights into the optimal transaction injection rate that won't destabilize or hamper the system's performance.

The parameters of the mathematical model are defined by studying the performance of validators (also referred to as miners) within the system and the unique configurations of the blockchain peer-to-peer (p2p) network. By integrating these parameters, the model can shed light on the system's abilities and constraints in terms of transaction velocities.

The simulation targets a specific circumstance where the blockchain p2p network is comprised of 10 validators, with each validator linked to three adjacent validators. This setup echoes the network structure and interconnectivity among validators, allowing the simulation to faithfully mimic real-world conditions.

To implement the simulation, a computer equipped with an i9-12950HX CPU is employed, harnessing the Windows Subsystem for Linux (WSL2) to operate a Linux operating system within the Windows 10 environment. This computational framework supplies the requisite resources and settings to effectively and efficiently conduct the simulation.

By constructing and applying this mathematical model, game developers and blockchain system designers can glean valuable insights into the best transaction injection rates for in-game trading systems. This information guides the conception and execution of trading platforms, ensuring they can accommodate high transaction volumes while preserving system performance and stability. Ultimately, the simulation outcomes aid in the creation of resilient and efficient in-game trading systems, enriching the overall gaming experience for players.

3.5.1. Transaction Per Block (Block Size)

The experiment concentrates on tweaking the block size attribute by altering the count of transactions per block within the data section. Each simulation cycle launches with a predetermined peak number of transactions that can be incorporated within an individual block.

The aim is to scrutinize the influence of block size on the system's performance and efficiency. This parameter enables an investigation of how varying block sizes impact the transaction processing duration, usage of network bandwidth, and the overall scalability of the system.

By setting a fixed maximum count of transactions for each block, it becomes viable to contrast the results from different simulation iterations and perceive the impact of varying block sizes on a range of performance measures. This methodology assists in pinpointing the optimal block sizes that balance transaction throughput with resource usage.

Adjusting the block size during the simulation allows an exploration of the balance between larger blocks, capable of accommodating more transactions but potentially causing longer processing times, against smaller blocks, which provide quicker transaction processing but limit the transaction count within each block.

By performing experiments with diverse block sizes, priceless insights can be derived regarding the ideal configuration for the trading mechanism. This involves identifying a balance that optimizes transaction throughput while ensuring tolerable levels of delay and resource usage. This information is of great importance to game developers and blockchain system architects as they aim to refine their in-game trading systems and provide a smooth and efficient trading experience for participants.

3.5.2. Bandwidth and Network Delay

Within this process, it is crucial to consider two types of network delay time: data transmission time and physics delay time. However, for the purpose of this study, the attention is solely on the data transmission time, while the influence of physics delay time is disregarded.

To ensure the simulations' consistency, the physics delay time is set as a constant value for all runs. This approach allows attributing any observed variations or differences in the results solely to the data transmission time.

In the simulation, the data transmission time is affected by two primary factors: the block size and the network bandwidth. The block size determines the volume of data to be transmitted, while the network bandwidth influences the speed of data transfer.

To simulate the validators' networking capabilities, it is assumed that the download bandwidth consistently exceeds the upload bandwidth by a factor of two. This assumption aims to replicate realistic networking conditions and accurately represent the data transmission process in the simulation.

Additionally, it is essential to note that all validators have the same network bandwidth throughout each simulation run. This ensures fairness and consistency when evaluating the impact of different block sizes on data transmission time.

By focusing on the data transmission time and carefully controlling the physics delay time, this study seeks to offer valuable insights into the relationship between block size, network bandwidth, and the overall efficiency of the in-game trading system.

3.5.3. Transaction Injection Rate

The transaction injection rate, as explored in this experiment, refers to the speed at which incoming transactions are introduced to the transaction pending queue per second. It represents the flow of transactions entering the system and serves as a significant determinant of the system's capacity and performance.

To ensure consistency and control in the simulation, the transaction injection rate will be maintained at a constant value throughout each simulation run. This implies that the number of incoming transactions added to the system per second will remain unchanged.

By keeping the transaction injection rate consistent, the experiment can focus on examining the impact of other variables, such as block size and network bandwidth, on the system's performance and efficiency. This methodology allows for a more controlled analysis of how different factors affect transaction processing and the overall functionality of the in-game trading system.

3.5.4. Simulation Process

The experiment's objective is to model the operation of a blockchain infrastructure in processing transactions associated with in-game asset exchanges. The simulation commences with system initialization, with a predetermined bunch of transactions being placed into the awaiting transaction queue. These transactions are representative of the trading interactions happening in the virtual gaming world.

Within the simulation framework, a validator shoulders the task of gathering and authenticating transactions from the queue of awaiting transactions. The quota of transactions a validator can amass in each block is governed by the block size configuration. This setting delineates the upper boundary on the count of transactions that can be assimilated within a block.

Following the collection and validation of transactions, the validator generates a fresh block encompassing these transactions. This block is subsequently disseminated to all associated validators within the decentralized network. This mechanism is perpetuated at constant intervals, aligning with block creation.

The duration taken for every block interval is assessed to reckon the volume of incoming transactions, which is premised on the predefined rate of transaction injection. These incoming transactions are added to the queue of awaiting transactions, poised to be processed during the next block creation cycle.

The simulation is sustained until either of the two criteria is satisfied. The first situation arises when the transaction queue awaiting processing becomes vacant, implying that all transactions have been effectively processed. This denotes a successful simulation operation. The second scenario transpires when the queue of transactions awaiting processing surpasses its holding capacity, indicating the system's inefficiency in effectively processing incoming transactions. This eventuality leads to a flawed simulation operation.

By executing the simulation under an array of situations and configurations, the experiment endeavours to appraise the blockchain infrastructure's performance and efficacy in processing transactions for in-game asset trading. The results yielded will bestow crucial insights into the system's capabilities and inform the optimization of its design and specifications to augment the overall in-game trading experience.

3.6. Transaction Injection Rate vs Transaction Handling Rate

For each combination of block size, bandwidth, and injection rate, the experiment will be repeated 100 times to ensure statistical significance. In each run, the system's ability to handle in-game item trading transactions will be assessed, determining whether it succeeds or encounters failures.

The outcomes of these runs, regardless of success or failure, will be recorded and subjected to analysis. The fail rate, which represents the proportion of failed runs out of the total number of runs, will be computed for each configuration.

Through the examination of these results, an estimation of the maximum injection rate that the system can handle without experiencing failures will be derived. This maximum injection rate signifies the upper limit at which the system can efficiently process incoming transactions, ensuring that the transaction pending queue is not overwhelmed or exceeds its capacity. It serves as a critical metric for evaluating the system's transaction processing capacity.

The achievement of a fail rate of zero while estimating the maximum injection rate is a significant milestone. It indicates the optimal transaction rate that the system can sustain without encountering failures. This insight provides valuable guidance for system designers and developers in fine-tuning parameters such as block size and network bandwidth to achieve the desired performance and scalability.

By conducting extensive simulations and analyzing the results, the experiment aims to deliver a comprehensive understanding of the system's capabilities and limitations in managing in-game item trading transactions. This knowledge will serve as a foundation for designing and optimizing blockchain-based systems that facilitate seamless and reliable in-game trading experiences.

3.7. Block Time Interval vs Transaction Delay

Within an actual in-game trading scenario, the pace of transaction initiation and the rate of transaction management are vital elements dictating the system's efficacy and functionality. Assuming the initiation rate is subpar compared to the management rate, it is feasible to swiftly gather and integrate all awaiting transactions into a fresh block. This strategy ensures a quicker transaction management duration than the block interval, facilitating efficient and punctual transaction execution.

On the flip side, if the initiation rate overtakes the management rate, new transactions are forced to linger in the waiting queue, resulting in a lapse surpassing the block interval. The size of the queue in limbo directly contributes to the transaction delay, with an elongated queue causing an extended waiting duration. This lag may interfere with the system's reactivity and the cumulative trading experience for participants.

For the purpose of scrutinizing and deciphering the system's reactions under varied setups, this study will document the mean block time for each permutation of block capacity and network throughput. The mean block time embodies the time necessary for a new block's inception within the blockchain structure. Through accumulating this data, we can construct a mathematical blueprint capable of estimating the mean block time rooted in the system's parameters.

This mathematical scheme will yield priceless perspectives into the system's performance attributes and augment our comprehension of how elements like block capacity and network throughput sway the block creation procedure. It will arm us with the ability to gauge the mean block time and assess the system's proficiency in managing transactions under diverse situations.

Merging experiential data gathered from the study with the mathematical model enables us to enrich our understanding of the interplay among block capacity, network throughput, and mean block time. Such comprehension will be a pivotal asset in fine-tuning the system's setup and elevating its functionality to efficiently accommodate in-game trading transactions' requirements.

3.8. Transaction Pending Queue Size vs Peak(Dynamic) Injection Rate

The transaction pending queue serves as a vital component in managing fluctuations in trading transactions within a real game environment. As transactions can vary in volume over time, it is essential to have a queue size that can accommodate these fluctuations and prevent transaction overflow or failure.

In Figure 3.8, the blue line represents the system's transaction handling rate, indicating the rate at which transactions can be processed and added to blocks. On the other hand, the red area represents the difference between the system's handling rate and the transaction injection rate, reflecting the potential backlog of transactions that cannot be immediately processed.

During periods of high transaction injection rates or spikes in trading activity, the red area becomes larger, indicating an accumulation of transactions waiting to be processed. To ensure successful handling of all incoming transactions, especially during peak times, it is crucial to have a pending queue size that exceeds the area represented by the red region.

By having a sufficiently sized pending queue, the system can absorb and store the incoming transactions until they can be processed and added to blocks. This buffer capacity allows the system to handle fluctuations in transaction volume, maintaining smooth operation and preventing overload.

Having an appropriately sized pending queue provides several benefits. Firstly, it helps avoid transaction overflow, where the system becomes overwhelmed with incoming transactions that exceed its capacity to handle. This prevents transaction loss or rejection, ensuring that all valid transactions can be processed and included in the blockchain.

Secondly, a well-sized pending queue improves the overall responsiveness and efficiency of the system. By accommodating fluctuations in transaction volume, it ensures that transactions can be processed in a timely manner, minimizing delays and enhancing the trading experience for players.

Lastly, a properly sized pending queue contributes to the stability and reliability of the trading system. It helps mitigate the impact of sudden increases or spikes in transaction injection rates, allowing the system to handle varying transaction volumes without compromising its performance or risking transaction failures.

Therefore, in the context of in-game trading, it is crucial to carefully determine and configure the size of the transaction pending queue based on factors such as expected trading volume, system resources, and desired responsiveness. This ensures that the system can effectively handle fluctuations in transaction activity, maintaining smooth operation and providing a seamless trading experience for players.

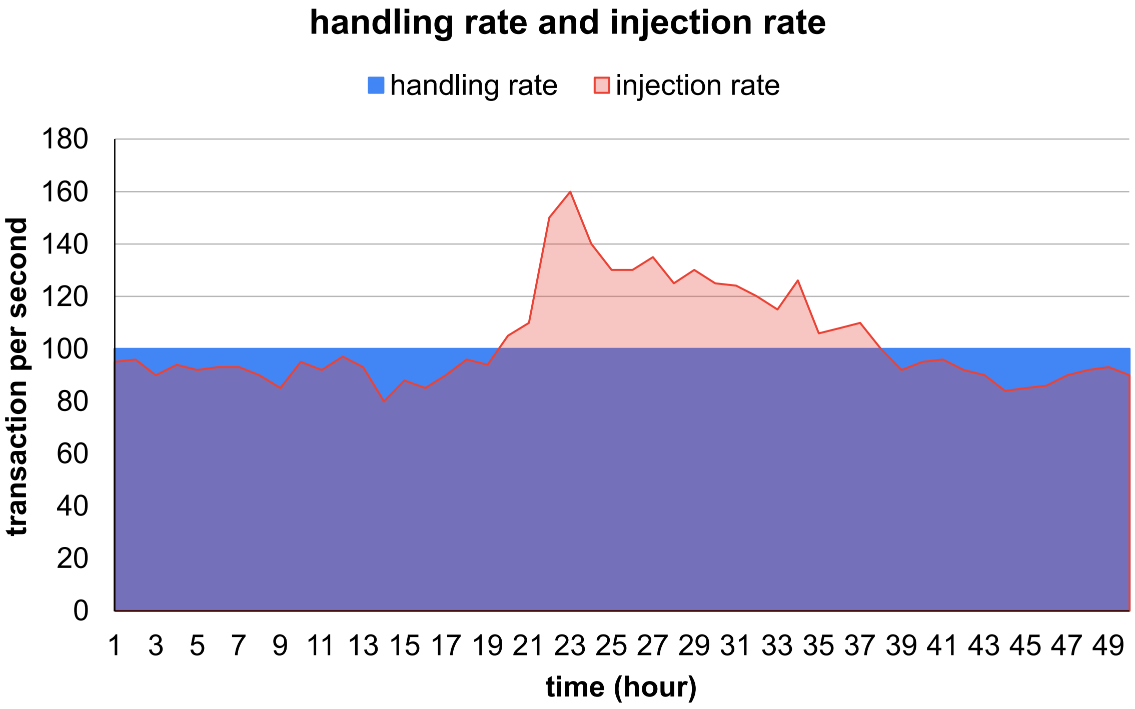

Figure 3.8 handling rate and injection rate

3.9. Pre-test Result

Pre-test findings related to block times are presented in Figure 3.9. This histogram depicts the block times for 1000 blocks, all produced under identical settings. The average block time comes out to be 14.94451629, with a standard deviation of 3.08578081, indicating significant instability in block times. As a result, further experimentation is necessary to obtain a more reliable mean value by running more times.

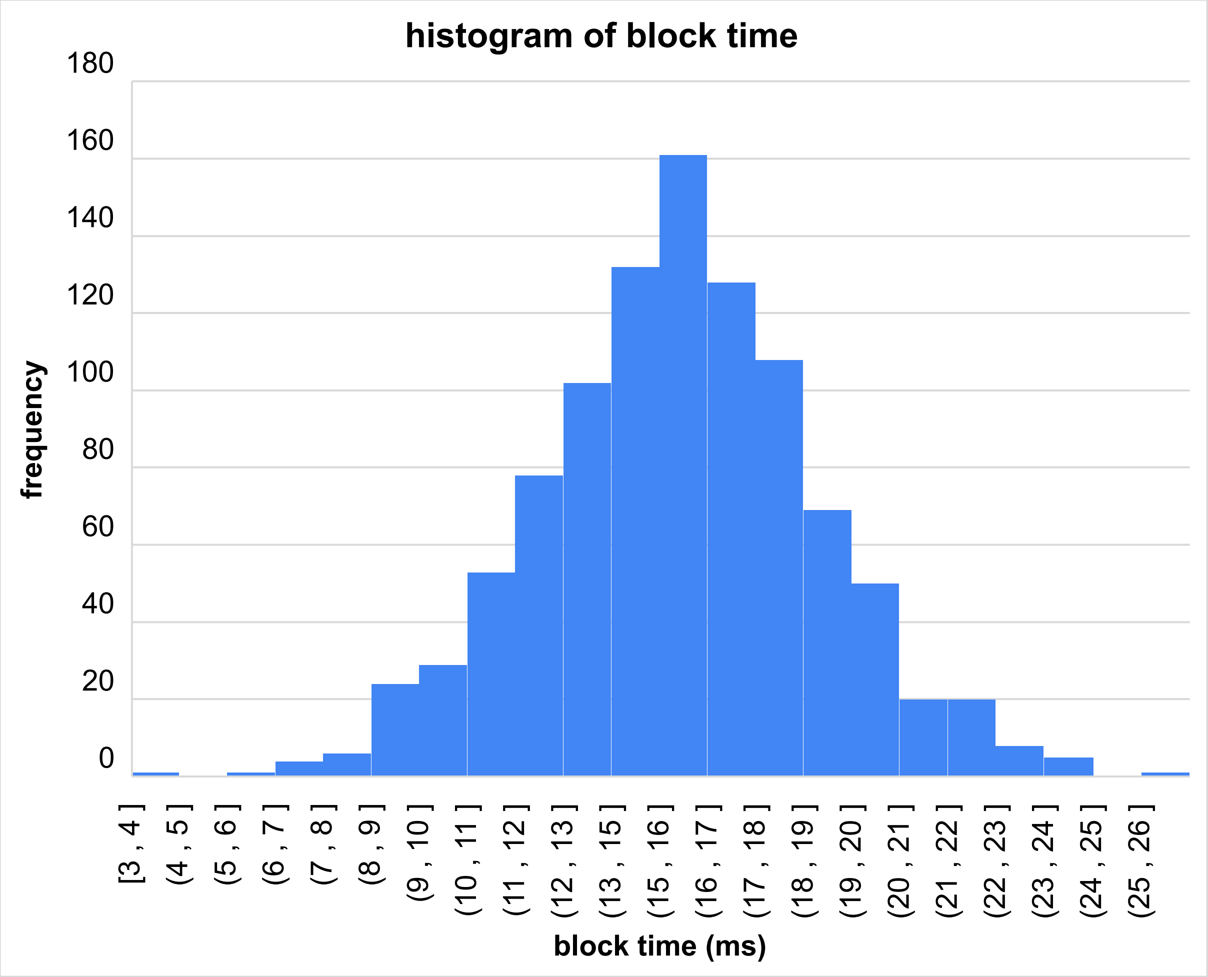

Figure 3.9 histogram of block time

3.10. Summary

The chapter delves deep into the intricate dynamics of in-game trading transactions, outlining key factors that contribute to a successful trading experience for players. The concept of transaction delay is extensively explored, emphasizing that the speed at which transactions are processed can significantly impact the gameplay experience. This requirement for low transaction delay varies across different game genres and is crucial for games demanding real-time trading.

The chapter then introduces a communication model designed specifically for blockchain-based streamed gaming. Subsequently, the discussion shifts to the design and implementation of a simulator that analyzes the time taken for network data transfer, contributing valuable insights into transaction delay.

The use of comprehensive analysis tools such as an extra data table, curve fitting, and the bisection method underpins the data interpretation stage. These tools aid in translating complex data into meaningful insights that can further optimize in-game transactions. The intricacies of the simulation process are elaborated upon, covering essential parameters like block size, bandwidth, network delay, and the transaction injection rate.

The latter sections of the chapter investigate the interplay between the transaction injection rate and the transaction handling rate, underscoring that a disparity between these rates may result in a backlog of unprocessed transactions. Similarly, the relationship between the transaction pending queue size and the peak injection rate is analyzed, highlighting the need for a queue size that can sustain transaction volume fluctuations. The chapter concludes by sharing preliminary test findings related to block times, which suggest significant instability and imply the need for additional experimentation.

Chapter 4. Result and Analysis

4.1. Transaction Injection Rate Limit vs Bandwidth and Block Size

4.1.1. Experiment Setting and Result

Experiments were conducted to analyze the impact of various network bandwidths (measured in KB/s) on the maximum injection rate in the context of in-game trading. The experiment utilized different configurations of transactions per block, and the results were analyzed using the data presented in Table 4.1.

In each simulation run, the initial pending queue consisted of 100 transactions. After the creation of each block, additional transactions were injected into the pending queue based on the product of the injection rate and the time elapsed since the previous block. The upload bandwidth for miners was varied from 100 KB/s to 800 KB/s, with increments of 100 KB/s. The download bandwidth for miners was always twice the upload bandwidth, maintaining a consistent ratio.

To evaluate the success of each simulation run, a criterion was set where a run would be considered unsuccessful if the pending queue reached a capacity of 300 transactions. Conversely, a successful run was defined as the pending queue reaching zero transactions at any point during the simulation.

The experiment was repeated 100 times for each configuration, allowing for the collection of fail rates and mean of block times corresponding to different combinations of network bandwidth and transactions per block.

The illustration depicted in Figure 4.1 outlines the impact that varying network bandwidths have on the maximum injection rate, particularly when combined with distinct transaction per block settings. This information can be found tabulated in Table 4.1. Each line represents a logarithmic progression. Figures 4.2 further demonstrate a linear trend as network bandwidth escalates.

Overall, the findings indicate that increasing the network bandwidth allows for a higher maximum injection rate, enabling the system to handle a larger volume of incoming transactions. This insight can help game developers and blockchain system designers optimize the network infrastructure for in-game trading, ensuring smooth and efficient transaction processing.

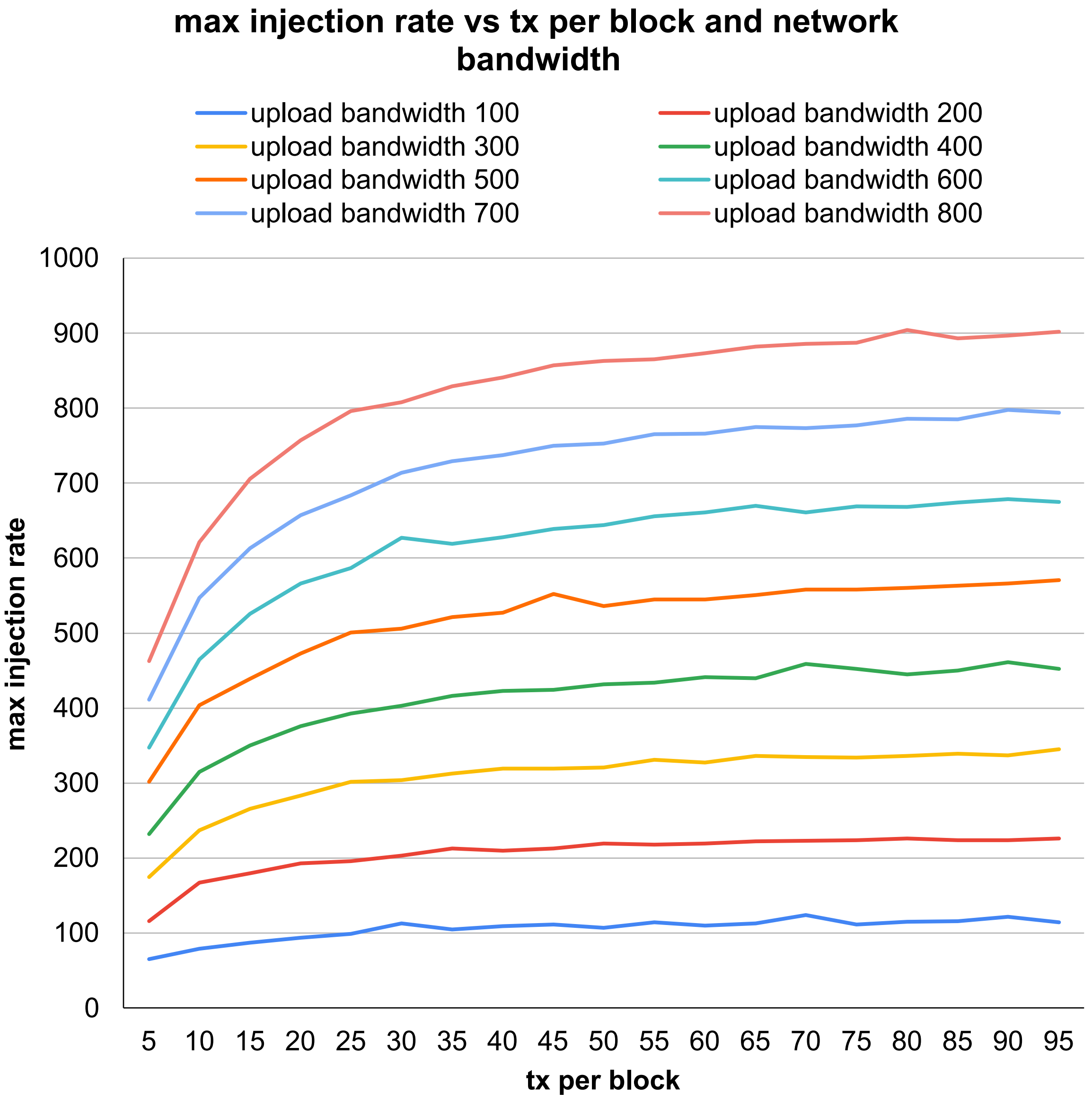

Figure 4.1 transaction injection rate limit vs bandwidth and block size

max injection rate		upload bandwidth(KB/S)							
		100	200	300	400	500	600	700	800
tx per block	5	65	116	175	232	302	347	411	463
	10	79	167	237	315	404	465	547	621
	15	87	180	266	350	439	526	613	706
	20	94	193	283	376	473	566	657	757
	25	99	196	302	393	501	587	684	796
	30	113	203	304	403	506	627	714	808
	35	105	213	313	416	521	619	729	829
	40	109	210	319	423	527	628	737	841
	45	111	213	319	424	552	639	750	857
	50	107	219	321	432	536	644	753	863
	55	114	218	331	434	545	656	765	865
	60	110	219	327	441	545	661	766	873
	65	113	222	336	440	551	670	775	882
	70	124	223	335	459	558	661	773	886
	75	111	224	334	452	558	669	777	887
	80	115	226	336	445	560	668	786	904
	85	116	224	339	450	563	674	785	893
	90	122	224	337	461	566	679	798	897
	95	114	226	345	452	571	675	794	902

Table 4.1 original max injection rate data

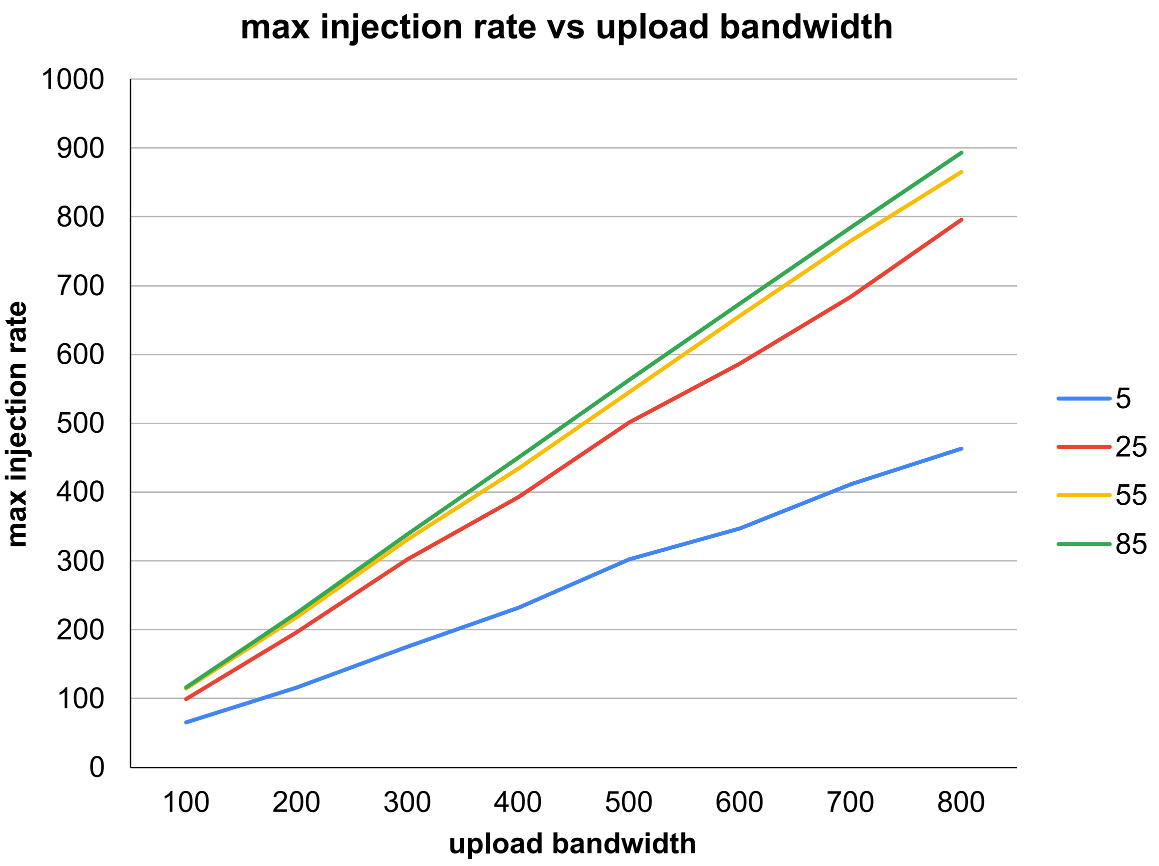

Figure 4.2 transaction injection rate limit vs bandwidth

4.1.2. Mathematical Model for Fixed Bandwidth

Two mathematical models, namely the logarithmic model and the exponential model, were selected to perform curve fitting analysis between the maximum injection rate and the number of transactions per block for a given network bandwidth.

The logarithmic model is represented by the equation:

$$y = a + b * ln(x) + c * x \tag{4.1}$$

where y represents the maximum injection rate, x represents the number of transactions per block, and a, b, and c are the model parameters.

The exponential model is represented by the equation:

$$y = a + b * x + c * e^{d*x} \tag{4.2}$$

where y represents the maximum injection rate, x represents the number of transactions per block, and a, b, c, and d are the model parameters.

Figures 4.3 to 4.10 depict the results of the curve fitting analysis for upload bandwidth ranging from 100 to 800 (KB/S), with increments of 100. The coefficient of determination (R^2) scores were used to evaluate the goodness of fit of the models. The R^2 scores indicate that there is not a significant difference between the two models, and the exponential model only offers a marginal advantage over the logarithmic model in terms of fitting the data.

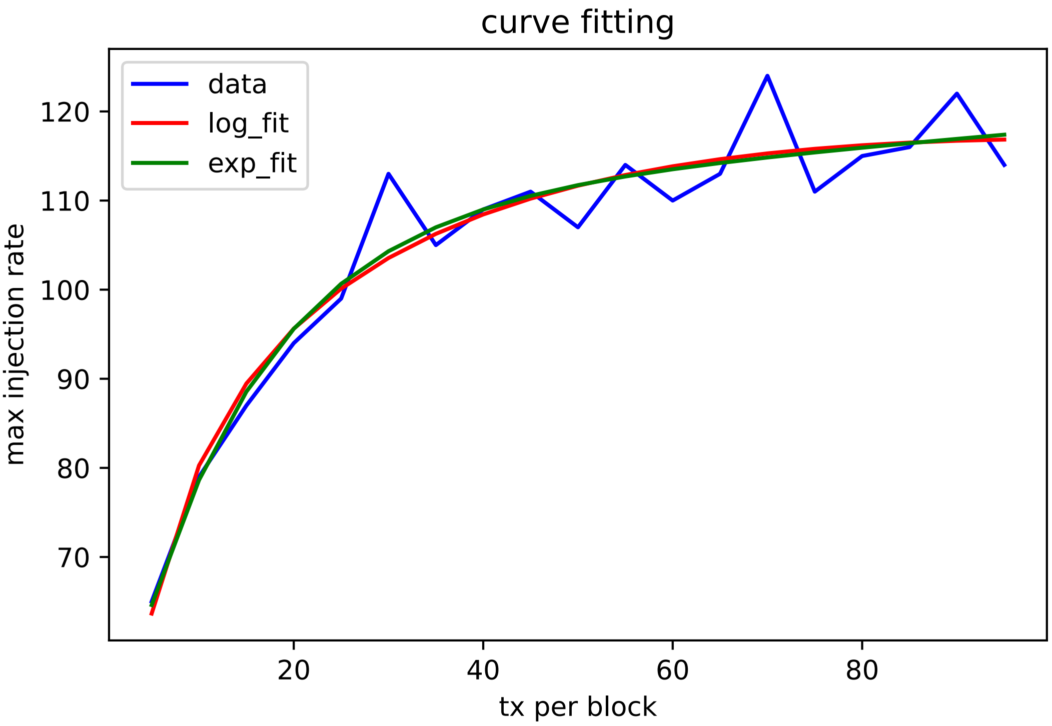

Figure 4.3 curve fitting result (upload bandwidth = 100KB/S)

logarithmic curve fitting result(upload bandwidth = 100KB/S), $R^2 = 0.9298528464043808$:

$$y = 23.39691600219724 + 25.799083116814046 * ln(x) - 0.2530450242144163 * x \tag{4.3}$$

exponential curve fitting result(upload bandwidth = 100KB/S), $R^2 = 0.9335947199679591$:

$$\begin{aligned} y = \; & 109.06486064832542 + 0.08835448225862452 * x \\ & - 64.3898819912871 * e^{-0.07210888438967783 * x} \end{aligned} \tag{4.4}$$

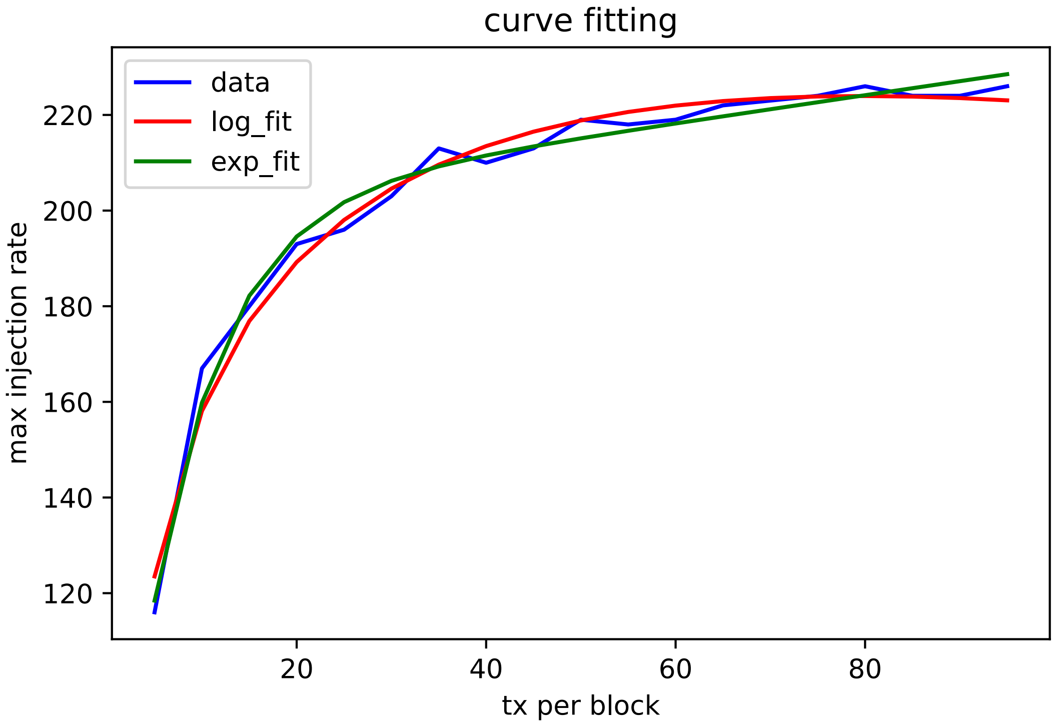

Figure 4.4 curve fitting result (upload bandwidth = 200KB/S)

logarithmic curve fitting result(upload bandwidth = 200KB/S), $R^2 = 0.9828016819852146$:

$$y = 38.641465401622675 + 54.86997581844493 * ln(x) - 0.6891083196980368 * x \tag{4.5}$$

exponential curve fitting result(upload bandwidth = 200KB/S), $R^2 = 0.9870739775871669$:

$$\begin{aligned} y = \; & 200.70814129714228 + 0.292763061337288 * x \\ & - 160.05898131252215 * e^{-0.12958806009229326 * x} \end{aligned} \tag{4.6}$$

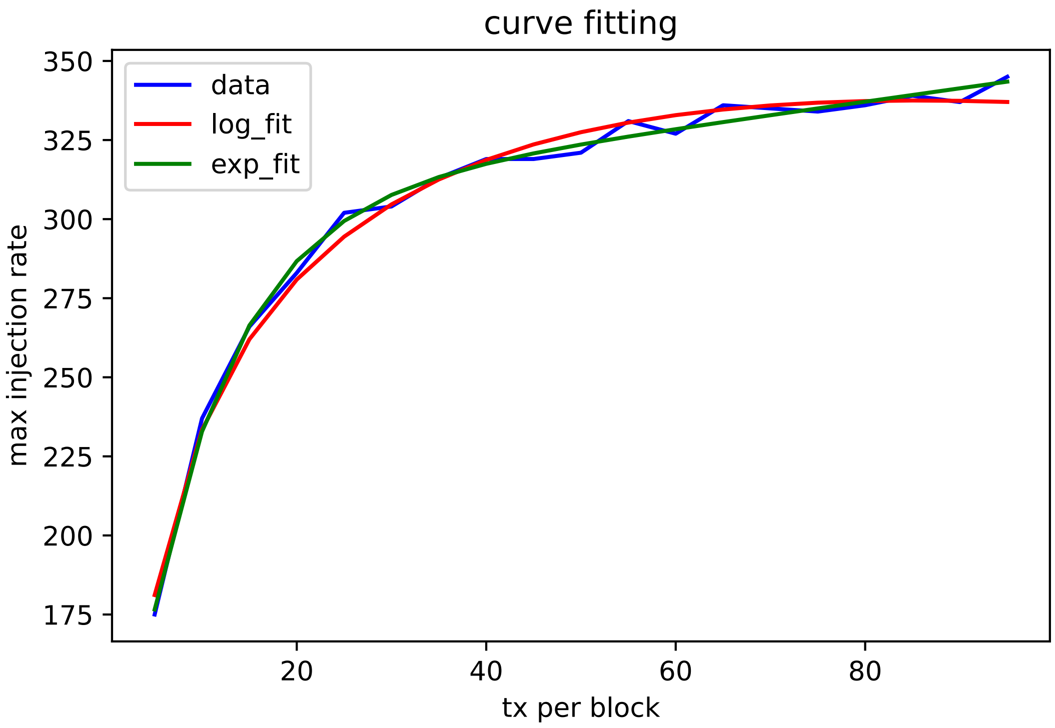

Figure 4.5 curve fitting result (upload bandwidth = 300KB/S)

logarithmic curve fitting result(upload bandwidth = 300KB/S), $R^2 = 0.9906580855588207$:

$$y = 53.42994241212728 + 82.35245467347293 * ln(x) - 0.9623021338451719 * x \tag{4.7}$$

exponential curve fitting result(upload bandwidth = 300KB/S), $R^2 = 0.9954382071701438$:

$$\begin{aligned} y = \; & 303.5009305781111 + 0.4207557261817233 * x \\ & - 222.39161777928467 * e^{-0.10890900096322592 * x} \end{aligned} \tag{4.8}$$

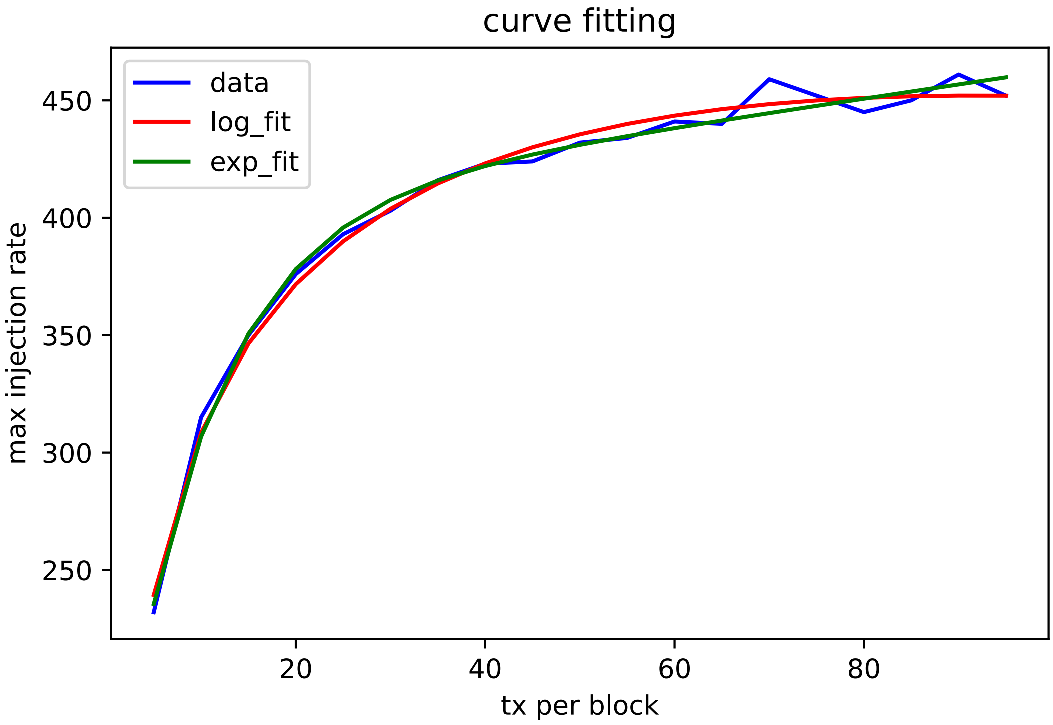

Figure 4.6 curve fitting result (upload bandwidth = 400KB/S)

logarithmic curve fitting result(upload bandwidth = 400KB/S), $R^2 = 0.9916347518778726$:

$$71.10599819761434 + 108.20465150538665 * ln(x) - 1.1774706575181368 * x \tag{4.9}$$

exponential curve fitting result(upload bandwidth = 400KB/S), $R^2 = 0.9919314822882328$:

$$\begin{aligned} y = \ & 402.6690783542761 + 0.601514480599832 * x \\ & - 284.1579810586587 * e^{-0.10255341907431291 * x} \end{aligned} \tag{4.10}$$

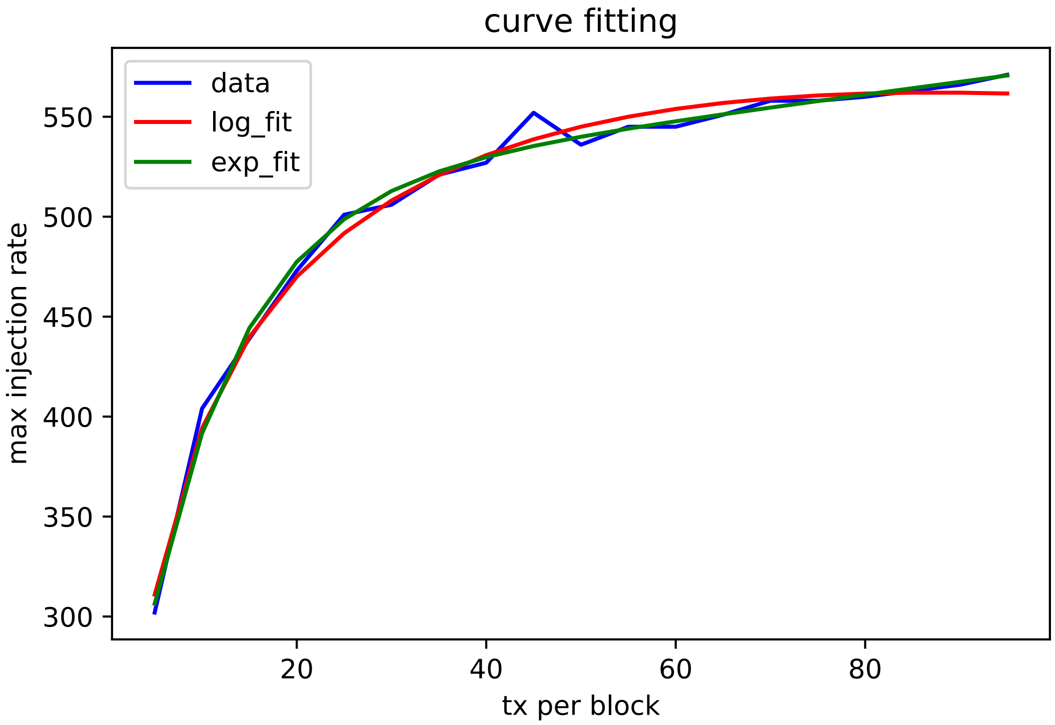

Figure 4.7 curve fitting result (upload bandwidth = 500KB/S)

logarithmic curve fitting result(upload bandwidth = 500KB/S), $R^2 = 0.9904865478246843$:

$$y = 107.63560832194128 + 130.96509790128476 * ln(x) - 1.4990021303031633 * x \quad (4.11)$$

exponential curve fitting result(upload bandwidth = 500KB/S), $R^2 = 0.9928629046926254$:

$$\begin{aligned} y = \ & 510.6170993088231 + 0.6320828073828646 * x \\ & - 343.0965147287491 * e^{-0.10069544556937 * x} \end{aligned} \quad (4.12)$$

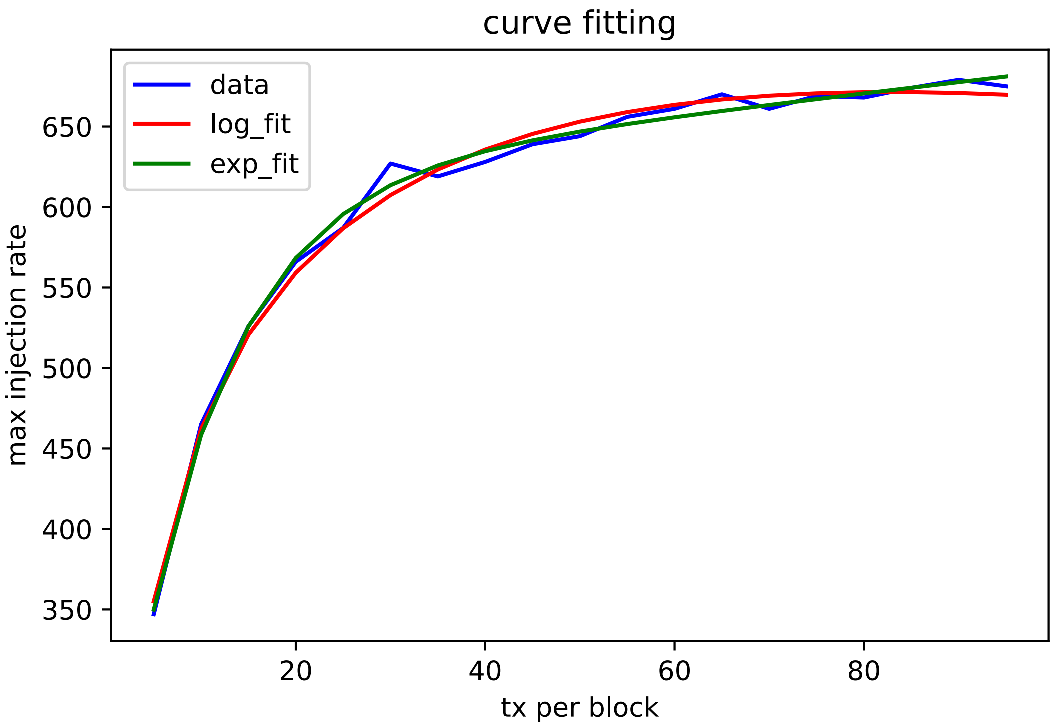

Figure 4.8 curve fitting result (upload bandwidth = 600KB/S)

logarithmic curve fitting result(upload bandwidth = 600KB/S), $R^2 = 0.992946458841731$:

$$y = 92.74760879359415 + 169.3456882424849 * ln(x) - 2.0439161869714537 * x \tag{4.13}$$

exponential curve fitting result(upload bandwidth = 600KB/S), $R^2 = 0.9952829794769869$:

$$\begin{aligned} y = \; & 615.3772485850039 + 0.6917477297826456 * x \\ & - 441.3913046973328 * e^{-0.09908541425110448 * x} \end{aligned} \tag{4.14}$$

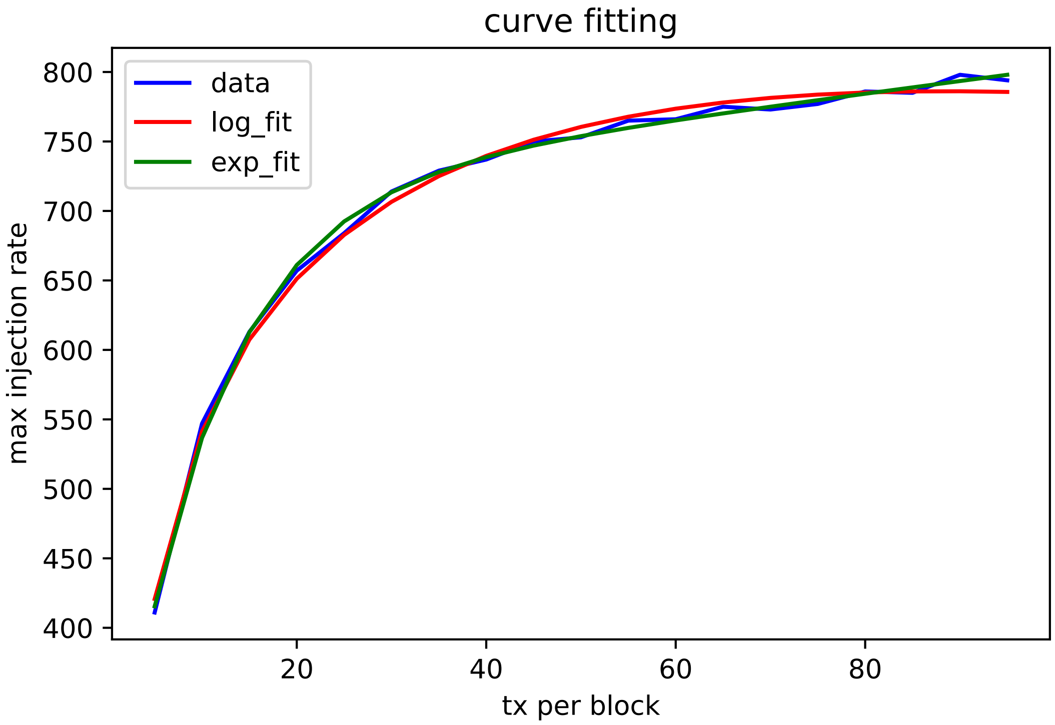

Figure 4.9 curve fitting result (upload bandwidth = 700KB/S)

logarithmic curve fitting result(upload bandwidth = 700KB/S), $R^2 = 0.9958836555088965$:

$$y = 126.57617470495748 + 189.42355889990964 * ln(x) - 2.142412500108811 * x \quad (4.15)$$

exponential curve fitting result(upload bandwidth = 700KB/S), $R^2 = 0.9980539027999326$:

$$\begin{aligned} y = \ & 712.5127397151589 + 0.9003161712062256 * x \\ & - 492.01124864402084 * e^{-0.09790012530521683 * x} \end{aligned} \quad (4.16)$$

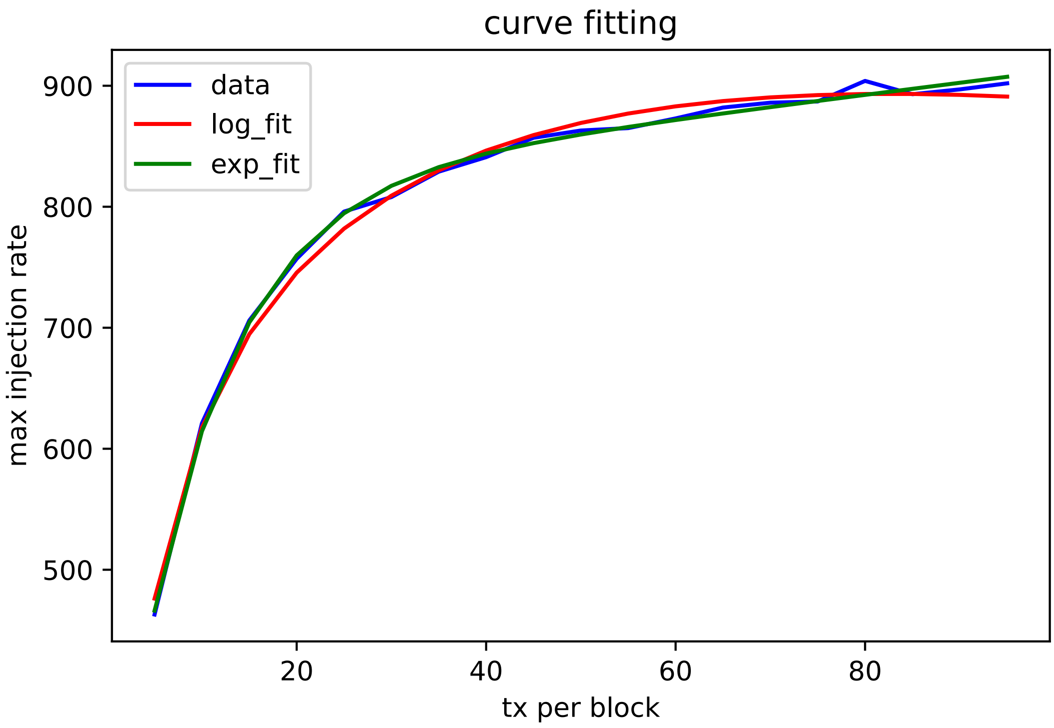

Figure 4.10 curve fitting result (upload bandwidth = 800KB/S)

logarithmic curve fitting result(upload bandwidth = 800KB/S), $R^2 = 0.9943918905712384$:

$$y = 129.74736792851814 + 223.620598495395 * ln(x) - 2.7058677387701016 * x \tag{4.17}$$

exponential curve fitting result(upload bandwidth = 800KB/S), $R^2 = 0.9980214387435754$:

$$\begin{aligned} y = & \ 813.4157106549335 + 0.9897947644791162 * x \\ & - 593.879470705328 * e^{-0.10451576469539568 * x} \end{aligned} \tag{4.18}$$

4.1.3. Mathematical Model Extend for Different Bandwidth

As depicted in Figure 4.11, all the parameters of the logarithmic model exhibit a linear trend with increasing bandwidth. Consequently, the values of a, b, and c in the logarithmic model were subjected to linear regression analysis. The data and results of this linear regression analysis are presented in Table 4.2.

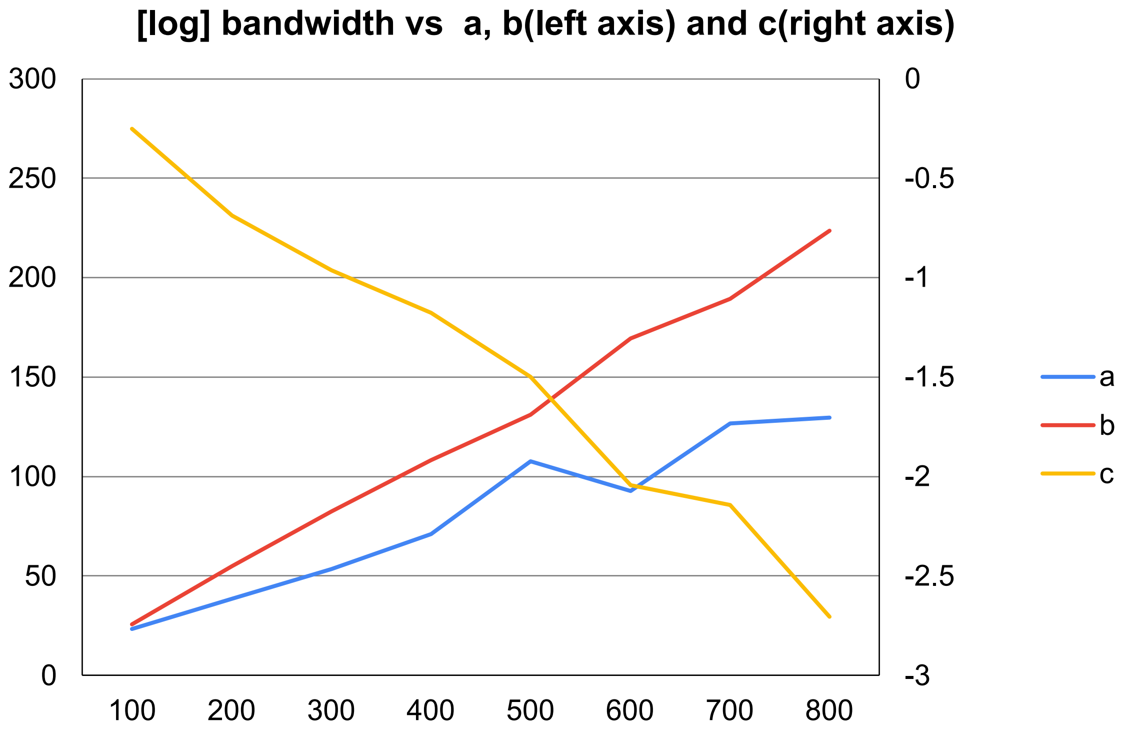

Figure 4.11 logarithmic model parameter vs bandwidth

y = a + b * ln(x1) + c * x1			
bandwidth	a	b	c
100	23.396916	25.79908312	-0.2530450242
200	38.6414654	54.86997582	-0.6891083197
300	53.42994241	82.35245467	-0.9623021338
400	71.1059982	108.2046515	-1.17747066
500	107.6356083	130.9650979	-1.49900213
600	92.74760879	169.3456882	-2.043916187
700	126.5761747	189.4235589	-2.1424125
800	129.7473679	223.6205985	-2.705867739
x2	0.1594*x2 + 8.699	0.2787*x2 - 2.352	-0.003334*x2 + 0.06600

Table 4.2 logarithmic model parameter linear regression result

As bandwidth increases, Figure 4.12 shows that the exponential model's parameters a, b, and c demonstrate a linear trend. However, parameter d remains nearly constant. As a result, the mean value obtained from all curve fitting results is used as a constant parameter for d.

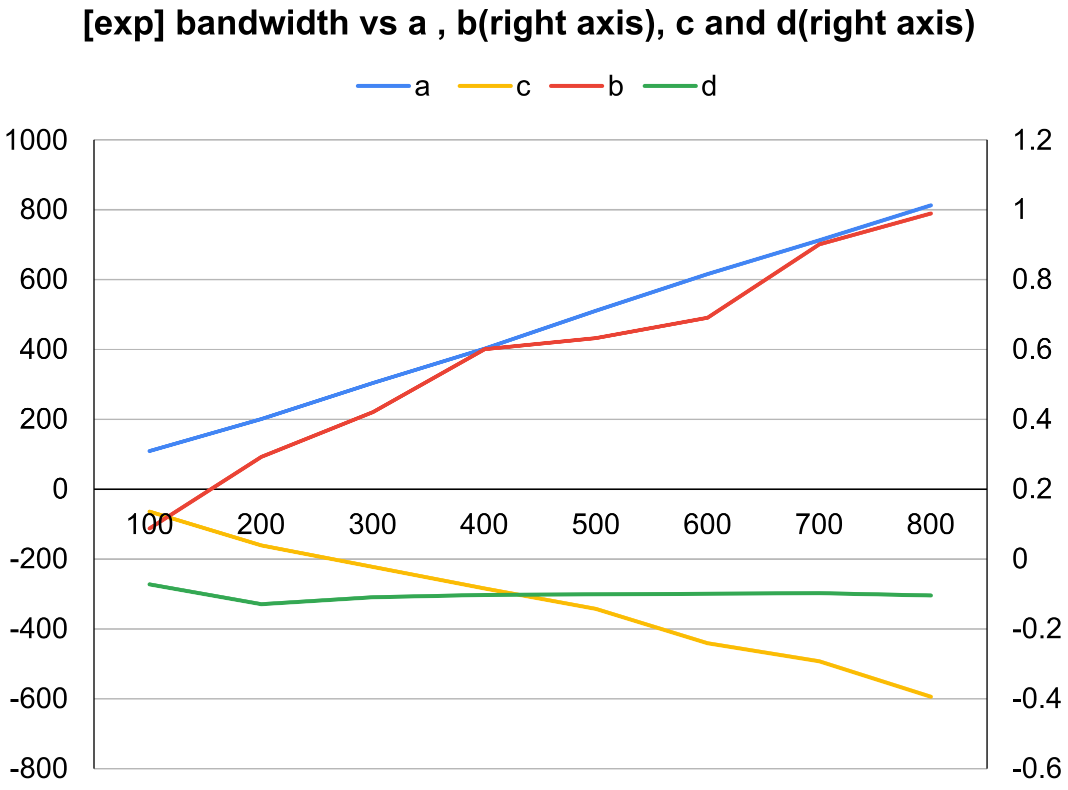

Figure 4.12 exponential model parameter vs bandwidth

$y = a + b * x1 + c * e^{d*x1}$				
bandwidth	a	b	c	d
100	109.0648606	0.08835448226	-64.38988199	-0.07211
200	200.7081413	0.2927630613	-160.0589813	-0.12959
300	303.5009306	0.4207557262	-222.3916178	-0.10891
400	402.6690784	0.6015144806	-284.1579811	-0.10255
500	510.6170993	0.6320828074	-343.0965147	-0.1007
600	615.3772486	0.6917477298	-441.3913047	-0.09909
700	712.5127397	0.9003161712	-492.0112486	-0.0979
800	813.4157107	0.9897947645	-593.8794707	-0.10452
x2	1.016*x2 + 1.355	0.001213*x2 + 0.03120	-0.7241*x2 + 0.6561	-0.1019

Table 4.3 exponential model parameter linear regression result

4.1.4. The Final Mathematical Model

The exponential model is selected because the R^2 score is larger and the trend of the curve never decrease.

$$\begin{aligned} y = \ & (1.016 * x2 + 1.355) + (0.001213 * x2 + 0.03120) * x1 \\ & + (-0.7241 * x2 + 0.6561) * e^{-0.1019 * x1} \end{aligned} \tag{4.19}$$

In this model, the maximum injection rate is measured in transactions per second, and the independent variables are the number of transactions per block (x1) and upload bandwidth measured in KB/S (x2). The diagram representing this model is depicted in Figure 4.13. The R^2 score for this model is 0.9993869670893204, compared to the original data displayed in Table 4.1.

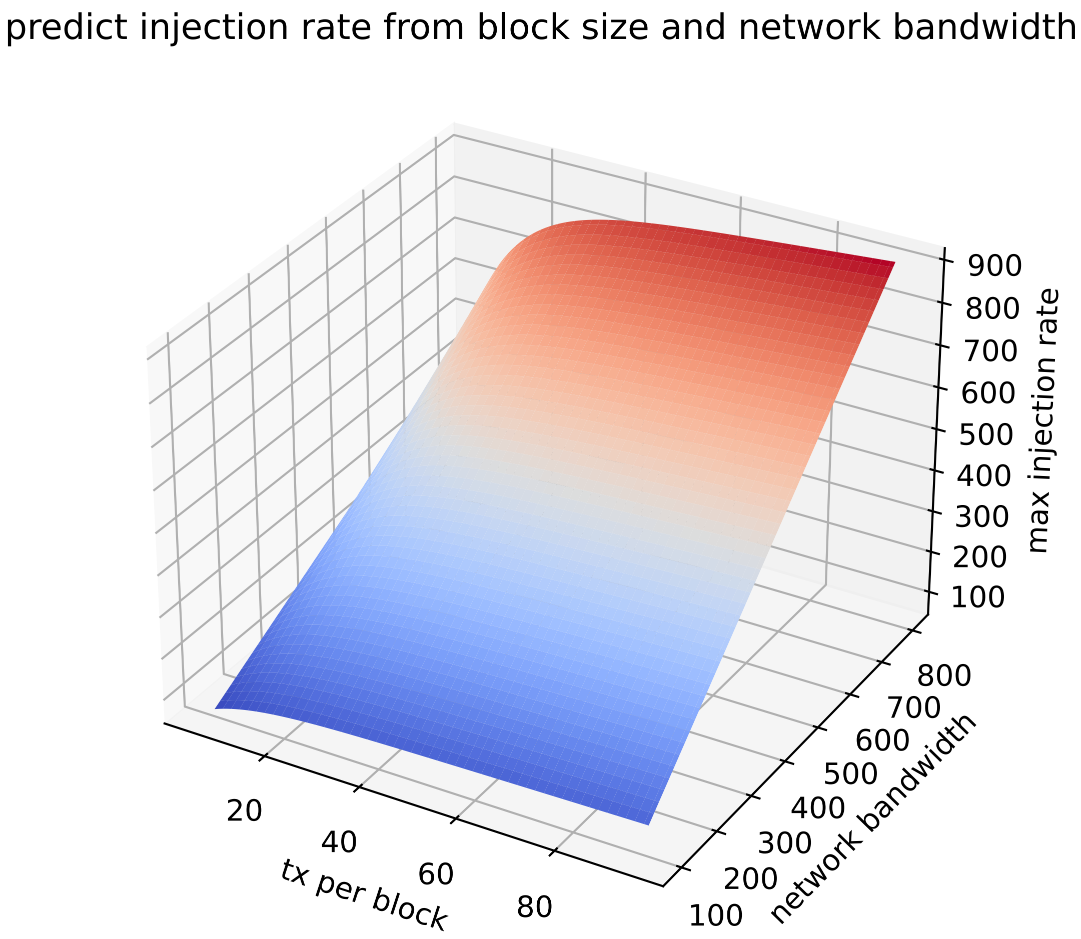

Figure 4.13 exponential model

4.2. Average Block Time vs Bandwidth and Block Size

4.2.1. Experiment Setting and Result

The impact of varying network bandwidths (measured in KB/s) on the average block time for different transaction per block configurations is presented in Figure 4.14, along with the relevant data outlined in Table 4.4. Each line on the graph exhibits a consistent linear trend. Different lines represent different bandwidths.

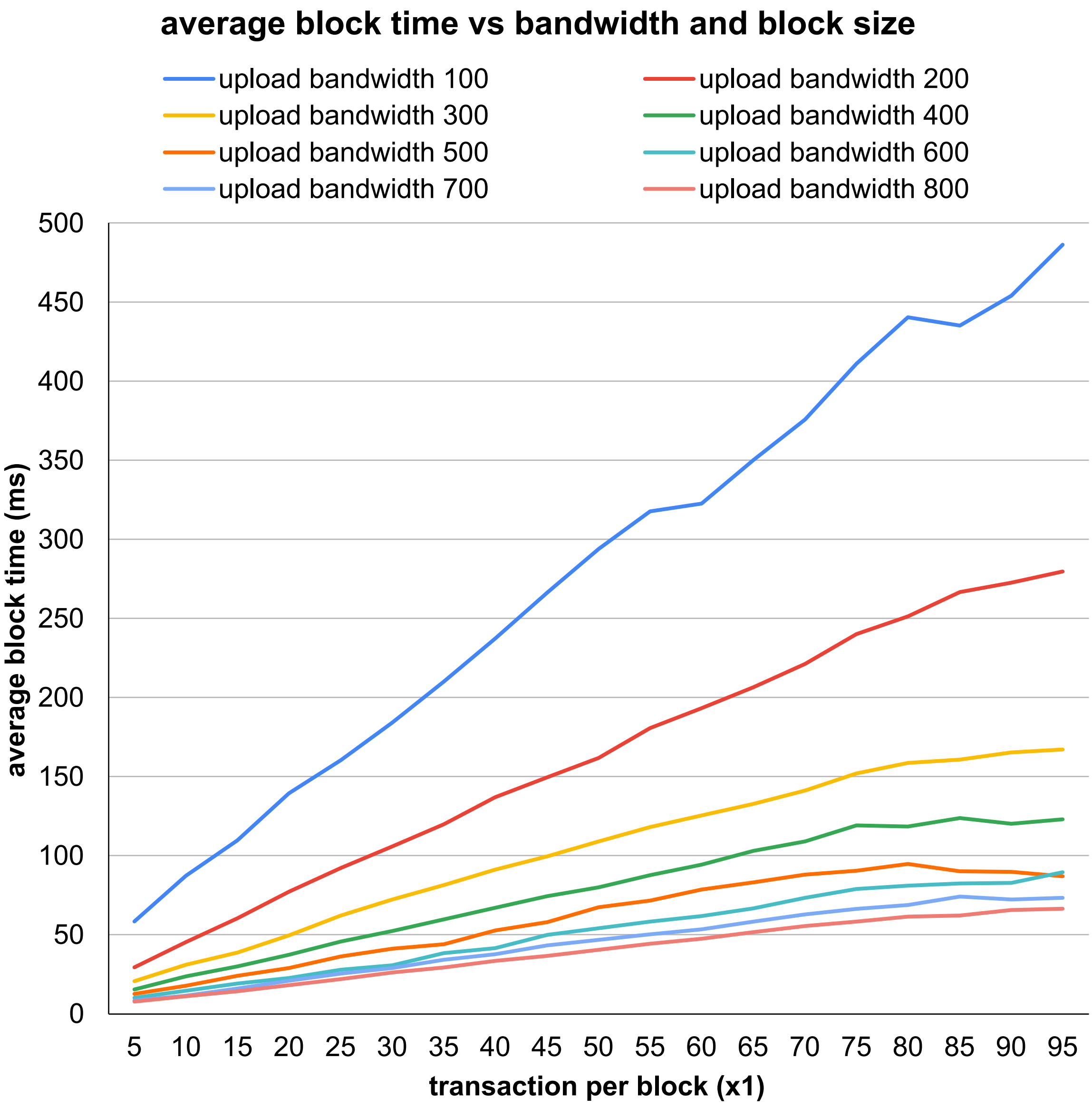

Figure 4.14 average block time vs bandwidth and block size

avr block time(ms)		upload bandwidth(x2)							
		100	200	300	400	500	600	700	800
tx per block(x1)	5	58.36	29.5	20.57	15.6	12.55	10.19	8.34	7.83
	10	87.31	45.39	31.1	23.86	17.93	14.58	11.43	11.25
	15	109.74	60.6	38.87	30.25	24.2	19.2	15.93	14.32
	20	139.47	77.34	49.6	37.42	28.98	22.66	20.85	18.25
	25	160.36	92.45	62.26	45.71	36.37	27.82	25.53	22.09
	30	184.2	105.86	72.31	52.31	41.36	30.65	28.96	26.21
	35	210.1	119.98	81.31	59.76	44.04	38.56	34.43	29.45
	40	237.21	136.99	91.33	67.13	52.63	41.48	37.88	33.59
	45	266.2	149.48	99.72	74.49	58.08	50.16	43.24	36.69
	50	293.85	161.9	109.16	80.19	67.37	54.29	46.74	40.47
	55	317.82	180.67	118.11	87.9	71.79	58.39	50.4	44.38
	60	322.45	193.22	125.34	94.48	78.51	61.93	53.47	47.41
	65	350.1	206.52	132.95	102.98	83.12	66.72	58.27	51.68
	70	375.62	221.36	141.31	109.08	88	73.53	62.94	55.71
	75	411.12	240.01	151.98	119.07	90.7	79.18	66.54	58.44
	80	440.31	251.2	158.63	118.65	94.68	81.06	69.04	61.53
	85	435.31	266.8	160.68	123.67	90.25	82.47	74.1	62.19
	90	454.18	272.74	165.37	120.23	89.77	82.72	72.43	65.67
	95	486.35	279.85	167.07	123.19	87.03	89.67	73.4	66.46

Table 4.4 data table of average block time vs bandwidth and block size

4.2.2. Linear Regression for Each Bandwidth

The linear regression is used to find the relation between average block time and block size when bandwidth is fixed. The linear regression result is shown in Table 4.5. The linear model is

$$y = a * x1 + b \tag{4.20}$$

where y is average block time and x1 is number of transactions per block.

model: y = a * x + b	y: block time	x: tx per block			
bandwidth	a	b	R^2	p-value	f-value
100	4.748	43.64	0.9953	<0.0001	3630
200	2.864	19.54	0.9977	<0.0001	7246
300	1.707	18.73	0.9872	<0.0001	1307
400	1.289	13.78	0.9804	<0.0001	850.4
500	0.9623	12.8	0.9426	<0.0001	278.9
600	0.9178	5.967	0.9904	<0.0001	1757
700	0.7792	5.984	0.9887	<0.0001	1489
800	0.6876	5.286	0.9938	<0.0001	2725

Table 4.5 linear regression result between average block time and block size

4.2.3. Extend the Model to Different Bandwidth with Curve Fitting

The coefficient a and b vs bandwidth and curve fitting result is shown in Figure 4.15. Both a and b use a power curve fitting model

$$y0 = c * x2^{d} \tag{4.21}$$

where y0 is the coefficient a or b, the x2 is bandwidth. The fitting result are

$$a = 413.2 * x2^{-0.96} \tag{4.22}$$

$$b = 4687.7 * x2^{-1.002} \tag{4.23}$$

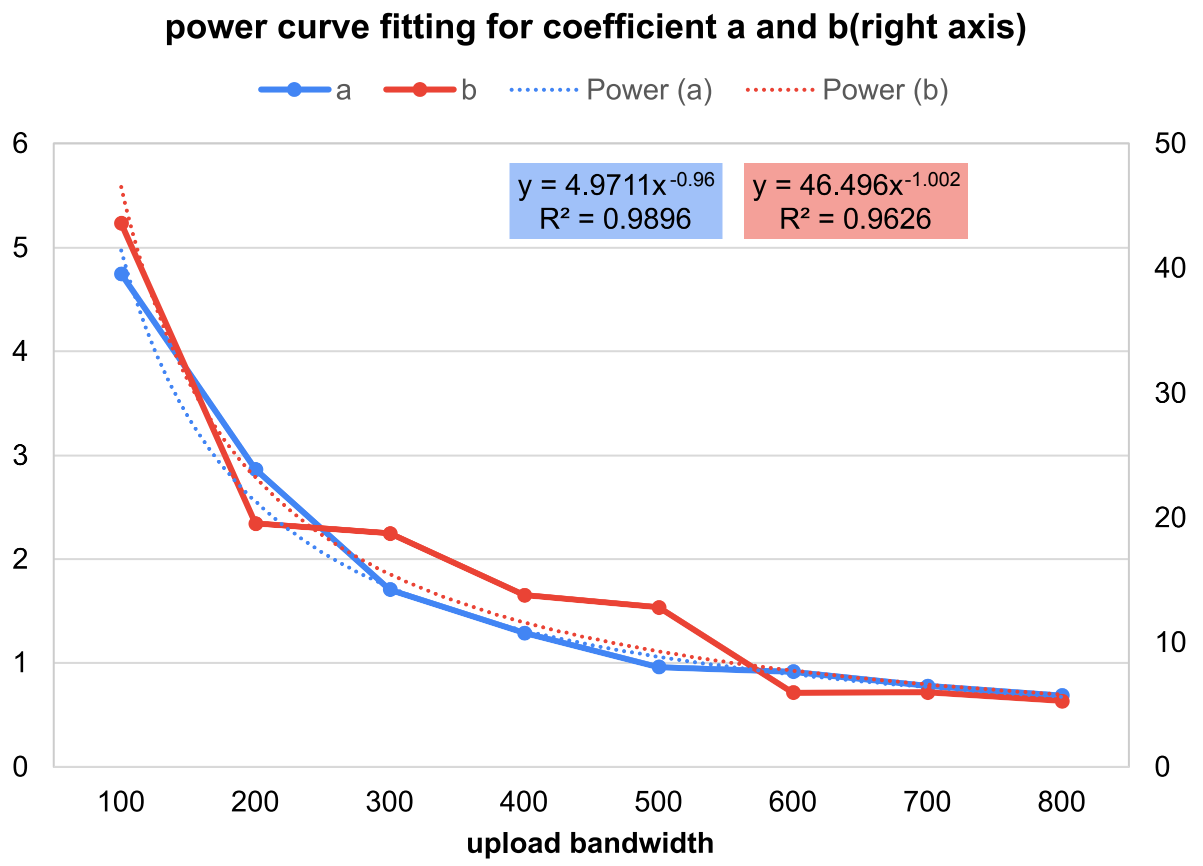

Figure 4.15 power curve fitting for coefficient a and b

4.2.4. The Final Mathematical Model

After put coefficient a and b into linear model. The final model is

$$y = (413.2 * x2^{-0.96}) * x1 + 4687.7 * x2^{-1.002} \tag{4.24}$$

where y is predicted average block time, x1 is number of transaction per block, x2 is upload bandwidth. The R2 score is 0.9914034275323211. The Visual of this model shows in Figure 4.16.

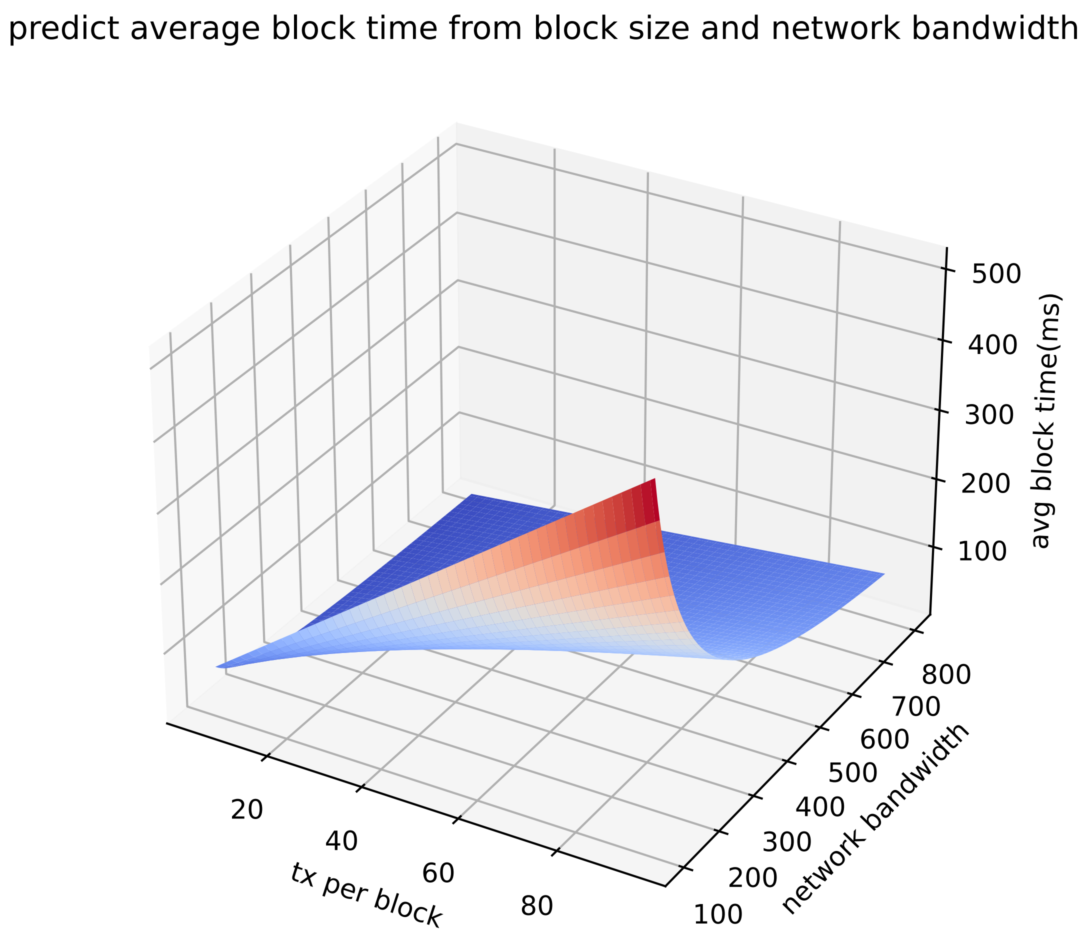

Figure 4.16 predicted average block time vs bandwidth and block size

4.3. Example Usage

All example will assume the blockchain p2p network setting and validators system performance were same as the simulation system.

4.3.1. Item Racing Game

Suppose we have a racing game where players can purchase power-up items and collect coins while racing. Each race consists of 8 to 10 players, and the game server can accommodate a total of 10,000 players online. On average, each game lasts for 5 minutes. Players typically wait approximately 10 seconds in the player queue before joining a new game. Within a single game, players have the option to purchase up to 3 items, and any remaining coins are transferred to their game account at the end of the game.

$$\begin{aligned} injection\_rate &= \frac{transaction\_number * num\_of\_players}{game\_time + queue\_time} \\ &= \frac{4 * 10000}{300 + 10} \approx 129.032 \end{aligned} \tag{4.25}$$

Under these conditions, when the server is at maximum player capacity and all players are actively engaged in the game, the average transaction injection rate is calculated by Equation 4.25 to be 129.032 transactions per second. By setting 129.032 as the upper limit for the injection rate in the injection rate model, we can establish the relationship between the number of transactions per block (x1) and the network bandwidth (x2) using the following equation:

$$\begin{aligned} 129.032 = \ & (1.016 * x2 + 1.355) + (0.001213 * x2 + 0.0312) * x1 \\ & + (-0.7241 * x2 + 0.6561) * e^{-0.1019 * x1} \end{aligned} \tag{4.26}$$

Rearranging the equation by moving x2 to the left side, we obtain:

$$x2 = \frac{129.032 - 1.355 - 0.0312 * x1 - 0.6561 * e^{-0.1019 * x1}}{1.016 + 0.001213 * x1 - 0.7241 * e^{-0.1019 * x1}} \tag{4.27}$$

If the internal delay for game trading needs to be less than 100 ms, we can use a block time model by substituting 100 as the lower bound for the block time, resulting in the relationship between the number of transactions per block (x1) and the network bandwidth (x2):

$$100 = (413.2 * x2^{-0.96}) * x1 + 4687.7 * x2^{-1.002} \tag{4.28}$$

By rearranging the equation and moving x1 to the left side, we get:

$$x1 = \frac{100 - 4687.7 * x2^{-1.002}}{413.2 * x2^{-0.96}} \tag{4.29}$$

The resulting diagram, shown in Figure 4.17, displays the relationship between the number of transactions per block (horizontal axis) and the network bandwidth (vertical axis). The blue area represents combinations of x1 and x2 that provide a block time of less than 100 ms. The red area represents combinations that yield a transaction handling rate greater than 129. The purple area represents the intersection of the red and blue areas, satisfying both the injection rate and block time limits.

To determine the minimum required bandwidth, we can calculate the values of x1 and x2 by substituting x2 into Equation 4.29 and using the bisection method. The resulting values are $x1 \approx 18.10$ and $x2 \approx 137.50$. This means that a minimum bandwidth of 138 KB/s and 18 transactions per block are required to meet the criteria of a 129 transaction injection rate and a 100 ms block time.

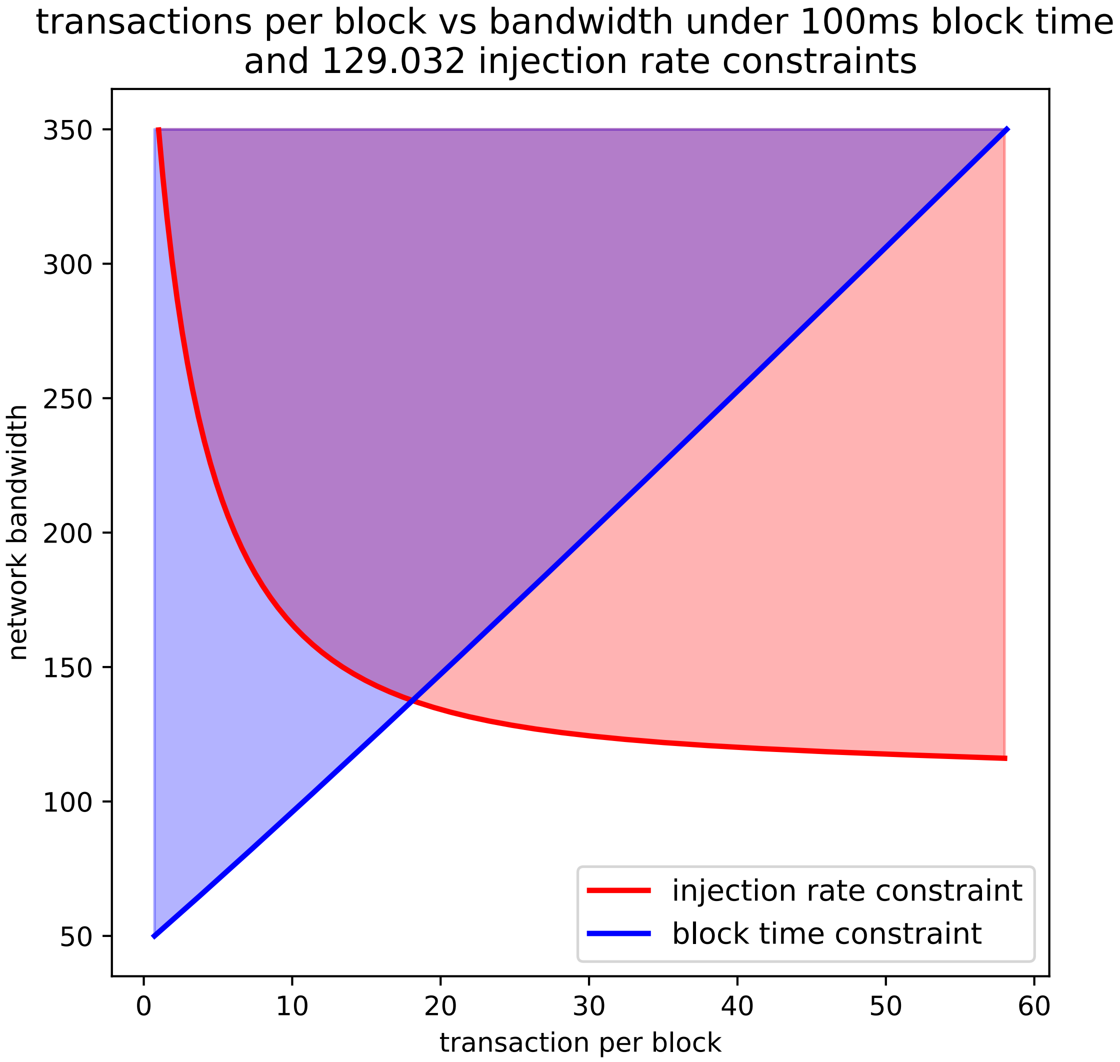

Figure 4.17 transaction per block vs bandwidth under 100ms block time and 129 injection rate constraints

4.3.2. Action Shooting Game

In this scenario, we have a dynamic multiplayer action shooter game where players can buy, sell, trade, or retrieve weapons while a battle is underway. These weapons are swapped for coins, the game's native currency. These activities construct the in-game economic structure, producing a multitude of transactions. A single battlefield supports 16 to 32 players at the same time, while the overarching game server can accommodate up to 10,000 players concurrently. Each discrete game session spans about 30 minutes. The game is crafted such that players experience an average wait time of 30 seconds in the queue before being able to engage in a new game session. Upon entering the game, each player must bring a collection of coins and weapons. This means that every player initiates the game with a certain amount of coins and one or multiple weapons for their initial survival and progress. Within a game, trading between players takes place, on average, 20 times. These trades encompass the exchange of weapons and/or coins. At the game's end, any remaining coins and weapons in a player's inventory are transferred to their game account. This transfer represents an additional form of transaction in the game, besides those created by the direct player-to-player trades.

Considering conditions where every possible slot in a server is occupied and each player actively participates in the game, the average rate of transaction injection into the system is computed to be 120.219 transactions per second. This rate is delineated and justified in Equation 4.30.

This rate, 120.219 transactions per second, is then utilized as the upper limit for the injection rate model. By inserting this value into the model, we can discern the relationship between the number of transactions per block (notated as x1) and the network bandwidth (signified by x2). This correlation is further detailed and visually displayed in Equation 4.31.

$$\begin{aligned} injection\_rate &= \frac{transaction\_number * num\_of\_players}{game\_time + queue\_time} \\ &= \frac{22 * 10000}{1800 + 30} \approx 120.219 \end{aligned} \tag{4.30}$$

$$\begin{aligned} 120.219 = \ & (1.016 * x2 + 1.355) + (0.001213 * x2 + 0.03120) * x1 \\ & + (-0.7241 * x2 + 0.6561) * e^{-0.1019 * x1} \end{aligned} \tag{4.31}$$

Rearranging the equation by moving x2 to the left side, we obtain:

$$x2 = \frac{120.219 - 1.355 - 0.0312 * x1 - 0.6561 * e^{-0.1019 * x1}}{1.016 + 0.001213 * x1 - 0.7241 * e^{-0.1019 * x1}} \tag{4.32}$$

If the internal delay for game trading needs to be less than 50 ms, we can use a block time model by substituting 50 as the lower bound for the block time, resulting in the relationship between the number of transactions per block (x1) and the network bandwidth (x2):

$$50 = (413.2 * x2^{-0.96}) * x1 + 4687.7 * x2^{-1.002} \tag{4.33}$$

By rearranging the equation and moving x1 to the left side, we get:

$$x1 = \frac{50 - 4687.7 * x2^{-1.002}}{413.2 * x2^{-0.96}} \tag{4.34}$$

The correlation between the count of transactions per block (x1) and the network bandwidth (x2) is visually represented in Figure 4.18. This chart uses the horizontal axis to display the transactions per block and the vertical axis to depict the network bandwidth.

The chart is divided into three separate regions, each denoted by a distinct colour. The blue region represents the combinations of x1 and x2 that result in a block time under 50 milliseconds. This range guarantees speedy transaction processing, a crucial factor for a smooth and reactive gaming experience. Conversely, the red region represents combinations that result in a transaction handling rate higher than 120. This rate is significant as it surpasses the determined upper boundary of the transaction injection rate, thus illustrating combinations that can handle high transactional volumes. The purple region is the intersection of the blue and red regions, indicating combinations that can satisfy both the block time and transaction injection rate limits. Essentially, the purple region exhibits combinations where both the rapid block time and high transaction handling rate are sustained.

To determine the minimum network bandwidth needed to fulfil these criteria, we must find the optimal values for x1 and x2. This is accomplished by substituting x2 into Equation 4.34 and employing the bisection method for the computation. The outcome of this computation suggests that $x1 \approx 7.66$ and $x2 \approx 170.58$. This signifies that to meet the criteria of an injection rate of 120 transactions per second and a block time of less than 50 milliseconds, a minimum bandwidth of 171 KB/s and 8 transactions per block are necessary.

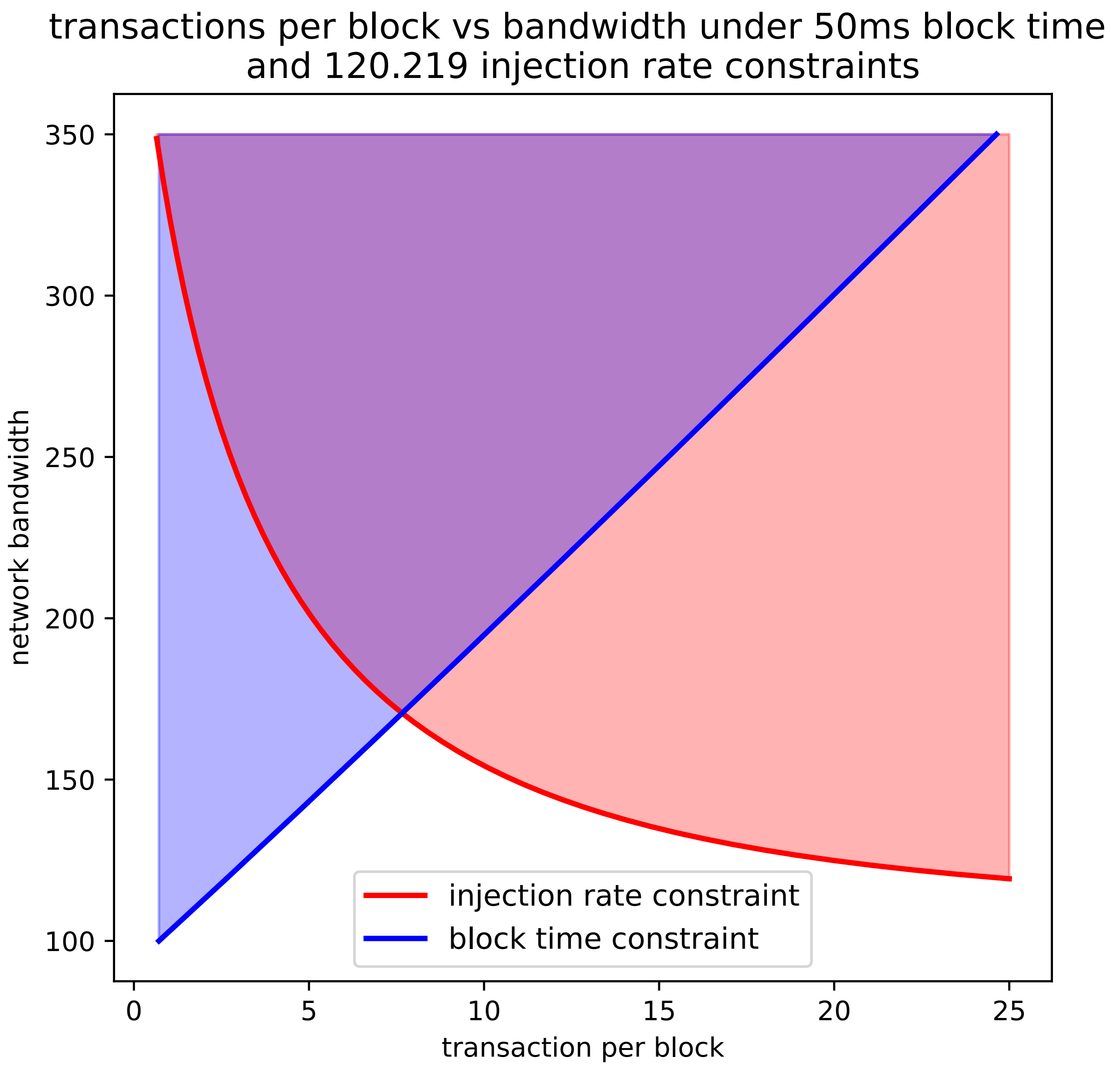

Figure 4.18 transaction per block vs bandwidth under 50ms block time and 120 injection rate constraints

4.3.3. MMORPG

In this scenario, we're examining a Massively Multiplayer Online Role-Playing Game (MMORPG), which includes a marketplace where players are able to trade in-game resources. The server setup for each game has the capacity to support up to 100,000 players interacting online concurrently.

The trading activity among players typically unfolds after combat encounters and averages around 25 transactions per hour. During peak gaming hours, which persist for a duration of 6 hours every day, the server operates at full capacity, accommodating the maximum limit of

100,000 players online. However, during the less crowded off-peak hours, server utilization dips to 20% of its capacity, which translates to roughly 30,000 players being online.

Given these operational circumstances, the computed average transaction injection rate is found to be 120.219 transactions per second during the bustling peak hours and falls to 120 transactions per second during the quieter off-peak hours. These rates are extracted from Equation 4.35 for peak periods and Equation 4.36 for off-peak periods.

These calculated injection rates are then set as the maximum thresholds in our injection rate model, which allows us to discern two separate relationships between the number of transactions per block (notated as x1) and the network bandwidth (expressed as x2). These relationships are characterized in Equation 4.37 for the peak time injection rate and Equation 4.38 for the off-peak time injection rate.

$$injection\_rate_{peak} = \frac{transaction\_number * num\_of\_players}{time(onehour)} = \frac{20 * 100000}{3600} \approx 833.334 \tag{4.35}$$

$$injection\_rate_{off\_peak} = \frac{transaction\_number * num\_of\_players}{game\_time + queue\_time} = \frac{20 * 20000}{3600} = 166.667 \tag{4.36}$$

$$\begin{aligned} 833.334 = \; & (1.016 * x2_{peak} + 1.355) + (0.001213 * x2_{peak} + 0.03120) * x1_{peak} \\ & + (-0.7241 * x2_{peak} + 0.6561) * e^{-0.1019 * x1_{peak}} \end{aligned} \tag{4.37}$$

$$\begin{aligned} 166.667 = \; & (1.016 * x2_{off\_peak} + 1.355) + (0.001213 * x2_{off\_peak} + 0.03120) * x1_{off\_peak} \\ & + (-0.7241 * x2_{off\_peak} + 0.6561) * e^{-0.1019 * x1_{off\_peak}} \end{aligned} \tag{4.38}$$

Rearranging the equations by moving both x2 to the left side, we obtain:

$$x2_{peak} = \frac{833.334 - 1.355 - 0.0312 * x1_{peak} - 0.6561 * e^{-0.1019 * x1_{peak}}}{1.016 + 0.001213 * x1_{peak} - 0.7241 * e^{-0.1019 * x1_{peak}}} \tag{4.39}$$

$$x2_{off\_peak} = \frac{166.667 - 1.355 - 0.0312 * x1_{off\_peak} - 0.6561 * e^{-0.1019 * x1_{off\_peak}}}{1.016 + 0.001213 * x1_{off\_peak} - 0.7241 * e^{-0.1019 * x1_{off\_peak}}} \tag{4.40}$$

Given that the majority of trading activity occurs outside of combat time, players will have more patience for a slightly extended transaction processing period. If the internal delay for in-game trading needs to be kept under 500 milliseconds (0.5 second), this value can be set as the limit for the block time in our model.

By placing this 500 ms threshold into the block time model, we can establish the relationship between the number of transactions per block (denoted as x1) and the network bandwidth (represented by x2). This relationship can be determined by the following equation for both peak and off-peak times:

$$500 = (413.2 * x2^{-0.96}) * x1 + 4687.7 * x2^{-1.002} \tag{4.41}$$

By rearranging the equation and moving x1 to the left side, we get:

$$x1 = \frac{500 - 4687.7 * x2^{-1.002}}{413.2 * x2^{-0.96}} \tag{4.42}$$

The results for the peak time are presented in Figure 4.19, where the horizontal axis represents the transactions per block (tx per block or x1), and the vertical axis denotes network bandwidth (x2). In the graph, the blue area corresponds to combinations of x1 and x2 that result in a block time of less than 500ms. This duration is significant as it ensures quick transaction processing, which is crucial for maintaining an immersive and responsive gaming experience. The red line represents the combinations of x1 and x2 that can accommodate a transaction handling rate of over 833. This rate is essential to meet the high transaction injection rate during peak times. The purple area represents the intersection of the blue area and the red line. Combinations within this area satisfy both the block time and transaction injection rate requirements, indicating a situation where rapid transaction processing and high transaction handling rates are achieved. Therefore, the lowest bandwidth requirement that satisfies both conditions can be found at the intersection point of two curves. At this point, x1 is approximately 475.09, and x2 is approximately 513.20.

In conclusion, to meet the demands of an 833 transaction injection rate and a block time of 500ms, the system needs a minimum bandwidth of 513 KB/S and must be capable of handling 475 transactions per block. The prediction outcome for this example fell outside the range of experimental data, which can lead to inaccuracies in the prediction result.

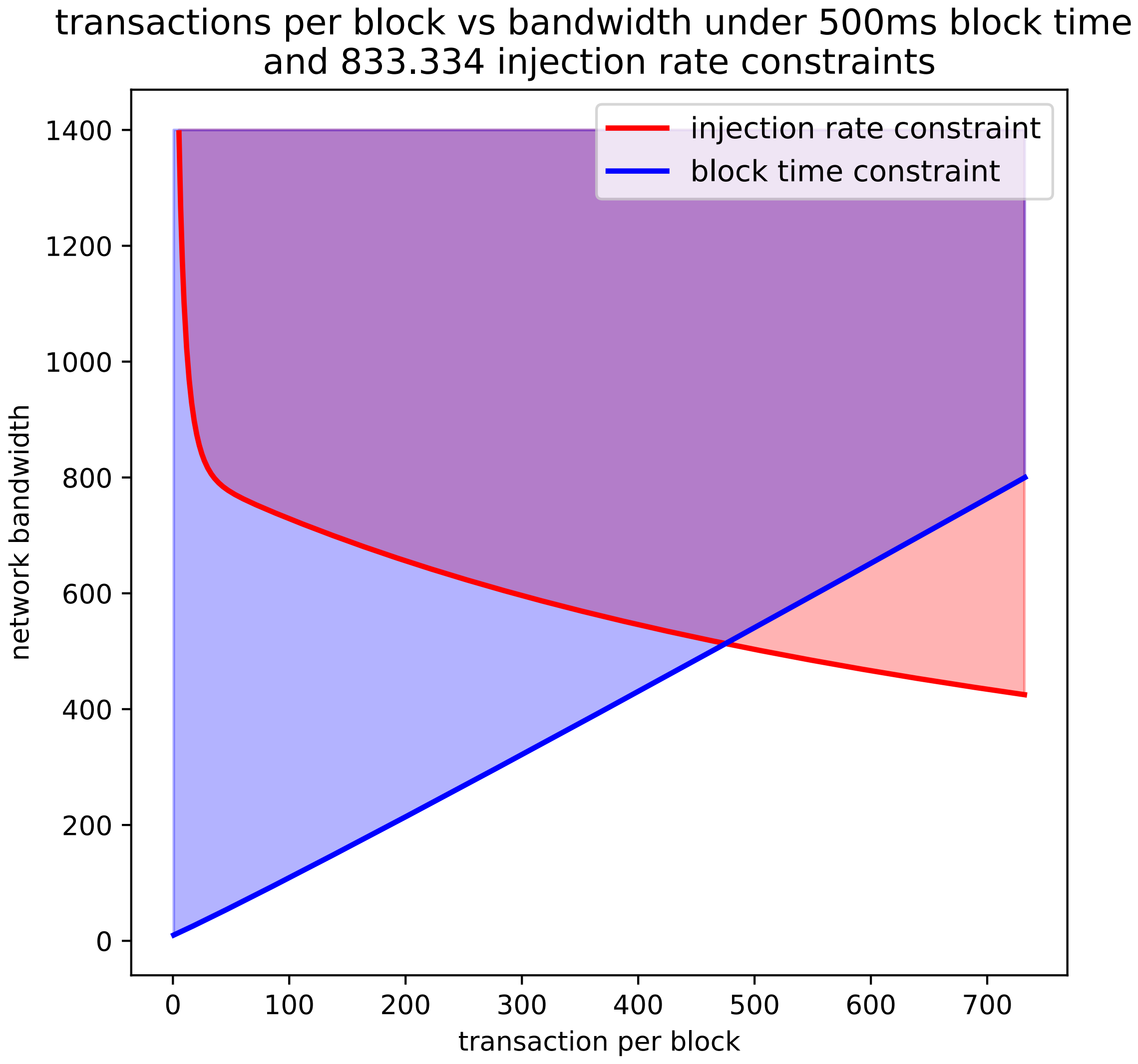

Figure 4.19 transaction per block vs bandwidth under 500ms block time and 833 injection rate constraints

The results for off-peak hours are visually displayed in Figure 4.20, with the horizontal axis representing the number of transactions per block (x1), while the vertical axis illustrates the network bandwidth (x2). The diagram is sectioned into specific areas. The blue region encapsulates combinations of x1 and x2 that yield a block time below 500 milliseconds, enabling rapid transaction processing critical for a seamless gaming experience. Contrastingly, the red area encapsulates combinations of x1 and x2 that can maintain a transaction handling rate above 167tps, vital for managing the transaction injection rate during off-peak times.

The overlap of these areas, indicated by the purple region, houses combinations that meet both the injection rate and block time stipulations. These are combinations facilitating both efficient transaction processing and sufficient transaction handling rates. The intersection point of the two curves, calculated via the bisection method as explained in previous examples, is approximately where $x1 \approx 127.69$ and $x2 \approx 137.78$. Therefore, in this context, the minimum bandwidth requirement stands at 138KB/S, accommodating 127 transactions per block to meet a transaction injection rate of 167tps and a block time of 500ms. The prediction outcome for this example fell outside the range of experimental data, which can lead to inaccuracies in the prediction result.

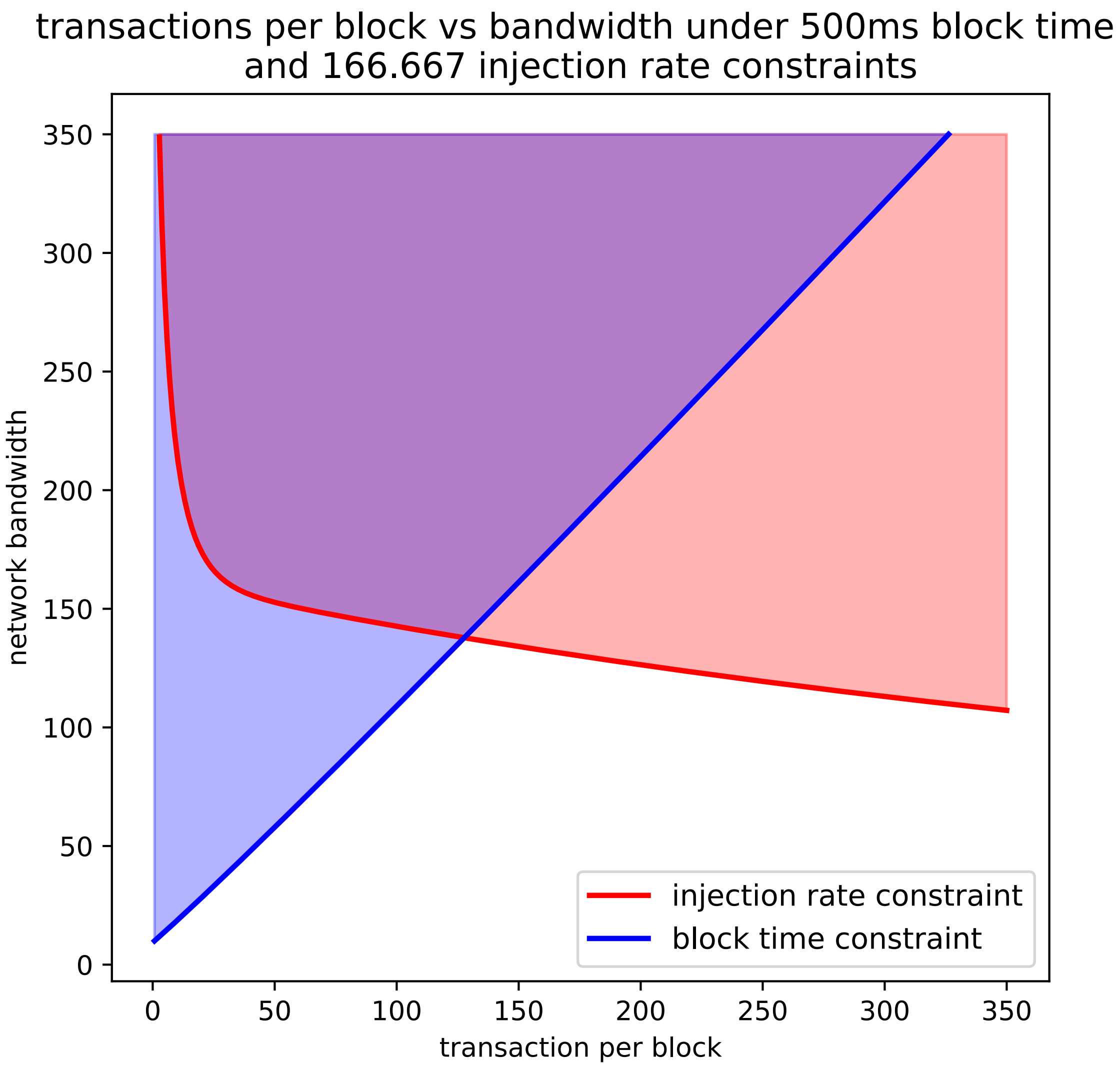

Figure 4.20 transaction per block vs bandwidth under 500ms block time and 167 injection rate constraints

Figure 4.21 provides a detailed representation of the transaction handling rate versus the transaction injection rate over the course of a typical day, broken down by hours, for this particular MMORPG. More specific data is available in Table 4.6.

As discussed in section 3.8, we can calculate the minimum length of the transaction pending queue by determining the area between the blue and red lines when the blue line is positioned above the red line. In this particular case, the minimum queue length that results is 186 transactions.

However, we must bear in mind a critical nuance. If the system sets the upper limit for pending time for any transaction during the transition from off-peak to peak times at 1000 milliseconds

(1 second), it suggests that when a system processes transactions at a rate of 167 transactions per second (tps), and the queued transactions surpass 167, any newly injected transaction could potentially experience an internal delay exceeding 1 second.

In practical terms, this means that the service provider needs to anticipate and increase the transaction handling rate as the number of transactions in the queue approaches 167. Such a proactive measure guarantees all transactions are processed within the acceptable limit, thereby ensuring a consistent and smooth gaming experience for players, even during peak transition periods.

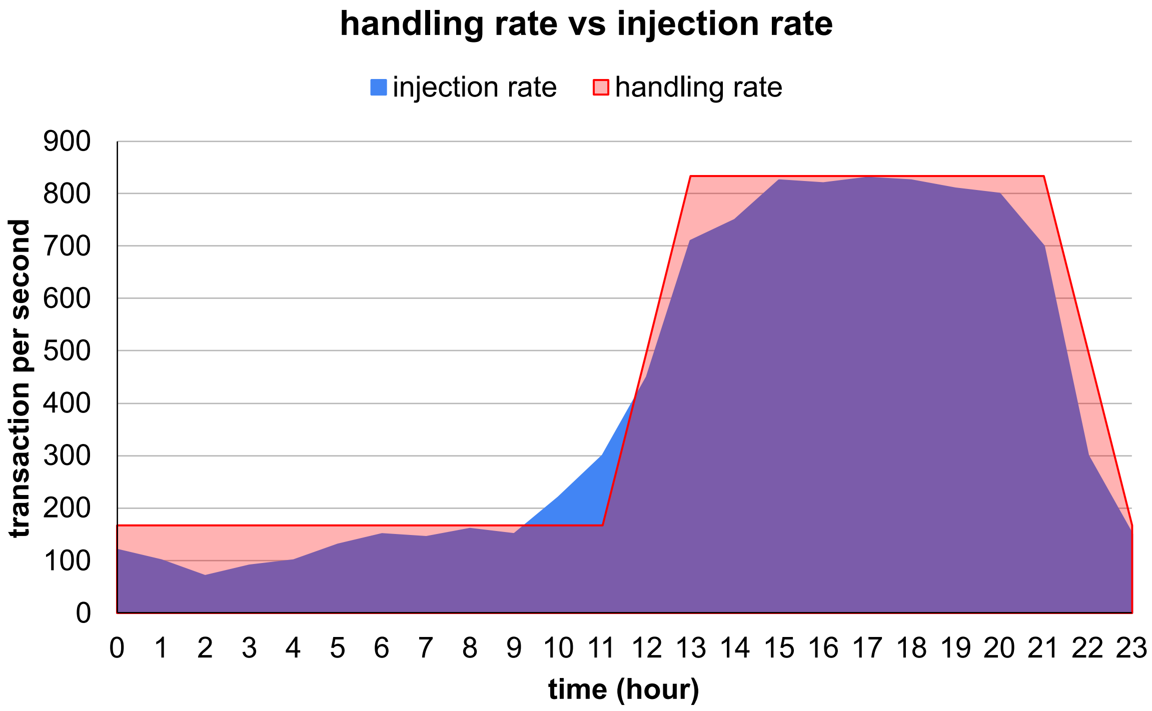

Figure 4.21 handling rate and injection rate in MMORPG example

time(hour)	injection rate	handling rate
0	120	167
1	100	167
2	70	167
3	90	167
4	100	167
5	130	167
6	150	167
7	145	167
8	160	167
9	150	167
10	220	167
11	300	167
12	450	500
13	710	833
14	750	833
15	825	833
16	820	833
17	830	833
18	825	833
19	810	833
20	800	833
21	700	833
22	300	500
23	150	167

Table 4.6 handling rate and injection rate data in MMORPG example

4.4. Summary

This chapter encapsulates the construction of a comprehensive mathematical model as a notable contribution of the thesis, focusing on the performance prediction of blockchain-based streamed gaming trading systems. This model has been built on mathematical modelling techniques to provide an in-depth understanding of how the system parameters, including block size, network bandwidth, and transaction pending queue size, correlate with their impact on the performance metrics such as transaction delay and transaction handling rate in the gaming environment.

The mathematical model was tested with three scenarios representing a diverse range of situations, each corresponding to a different combination of block size, network bandwidth, and transaction pending queue size. The results displayed significant variations in transaction handling rate and transaction delay, demonstrating the model's effectiveness in predicting system performance.

The model leverages the simulator for a nuanced analysis of different scenarios and configurations, paving the way for informed decision-making processes for blockchain performance optimization specific to the demands of streamed gaming. With its potential to significantly enhance system optimization and design processes, the mathematical model emerges as the primary contribution of this thesis. The chapter underscores the importance of understanding the correlation between system parameters and their performance impacts, thereby aiding in the efficient design and management of blockchain-based gaming systems.

Chapter 5. Conclusions

5.1. Thesis Summary

The thesis provides a comprehensive exploration of the integration of blockchain technology in the gaming industry, focusing on in-game virtual item trading. Through extensive background research, experimentation, and result analysis, it offers valuable insights into the potential of blockchain to enhance security, transparency, and efficiency in online gaming environments. The development of mathematical models and experimental validation further contributes to understanding the balance between blockchain performance and system resource usage. The findings of this research hold significant potential for the future development and implementation of blockchain-based solutions in the gaming industry.

The introduction chapter sets the stage for the exploration of blockchain technology, cryptocurrency transactions, and public ledger recording in streamed gaming. The project aims to facilitate content evolution and third-party game asset trading in online gaming environments. It also seeks to find a balance between blockchain performance and system resource usage through mathematical modelling. Key areas of focus include in-game virtual item trading and the application of blockchain technology to enhance security, trustworthiness, and transparency in the gaming industry.

Chapter 2 provides an extensive background on streamed gaming and distributed blockchain. It covers communication models, video streams, game genres, cryptographic hashing, digital signatures, transactions, blocks, and the blockchain's immutable and transparent nature. The chapter also discusses the challenges of streamed gaming in-game trading, such as ensuring security in transactions and maintaining real-time interactivity. It explores enhancing in-game trading through blockchain technology and compares different consensus algorithms in a streamed gaming environment.

Chapter 3 focuses on the experimental aspects of the thesis, detailing the requirements for in-game trading transactions, communication models, simulator design, and implementation. It also includes analysis tools, simulation details, and various experiments related to transaction delay, pending queue size, average injection rate, and peak injection rate. The chapter concludes with a summary of the pre-test results and findings.

Chapter 4 presents the results and analysis of the experiments conducted. It includes detailed mathematical modelling and analysis related to transaction injection rate limits, average block time, and example usage in different gaming scenarios such as item racing games, action shooting games, and MMORPGs. The chapter provides a comprehensive summary of the findings and their implications for the integration of blockchain technology in the gaming industry.

5.2. Contribution of Thesis

In reflecting on the journey of this research, the following key contributions emerge as significant milestones in the exploration of blockchain technology within the gaming industry:

5.2.1. Evaluating Consensus Algorithms for Streamed Gaming

The thesis has broken new ground by systematically evaluating various consensus algorithms, identifying those that best minimize latency and maximize throughput in real-time game trading. This evaluation has provided a roadmap for the gaming industry, guiding the selection and optimization of consensus algorithms.

5.2.2. Development of a Specialized Communication Model

A novel communication model specifically designed for blockchain-based streamed gaming has been introduced. This model has shed light on the dynamics of transaction delays, enhancing our understanding of real-time trading within gaming environments. It represents a significant step towards integrating blockchain technology with gaming.

5.2.3. Creation of a Predictive Mathematical Model

The thesis has made a significant contribution by developing a mathematical model designed to predict system performance and resource allocation. This innovative approach holds substantial potential for enhancing system optimization and design processes within the gaming industry.

5.3. Future Work

5.3.1. More Realistic Network Protocols

To improve the accuracy and realism of the simulation, future work should focus on extending the simulator to incorporate more realistic network communication protocols. For example, integrating TCP or UDP models would allow for the simulation of different network conditions and provide a more robust framework for analyzing the performance of blockchain-based gaming environments. This extension would make the simulator more applicable to real-world scenarios, where networking behavior plays a critical role in system performance.

5.3.2. Further Analysis for Fail Rate

Figure 5.1, Figure 5.2, and Figure 5.3 show the failure rate under varying transaction injection rates and transactions per block configurations. The y-axis represents the failure rate, while the x-axis signifies the injection rate, and distinct lines correspond to different transactions per block.

In all three diagrams, the fail rate limit increased in a step-like manner, remaining stable for some time, then increasing at a certain point and stabilizing again. This behaviour may be caused by the injection rate being on the threshold of whether the system needs to add a new block or not, requiring further investigation. The first rising edge of the line could be an indication that the injection rate is reaching a level that could potentially cause the system to fail. It is possible to collect the start and end points for each "first rising edge" in the fail rate limit for each transaction per block configuration. However, the method for selecting the start and end points requires further investigation.

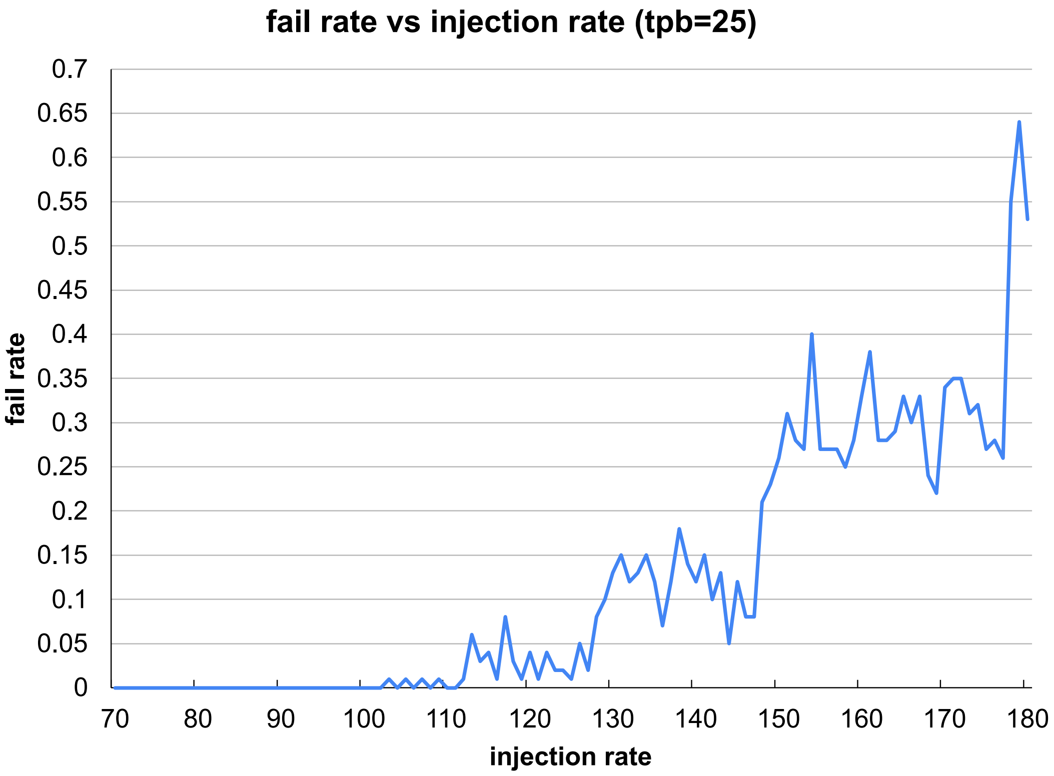

Figure 5.1 fail rate when transaction per block is 25

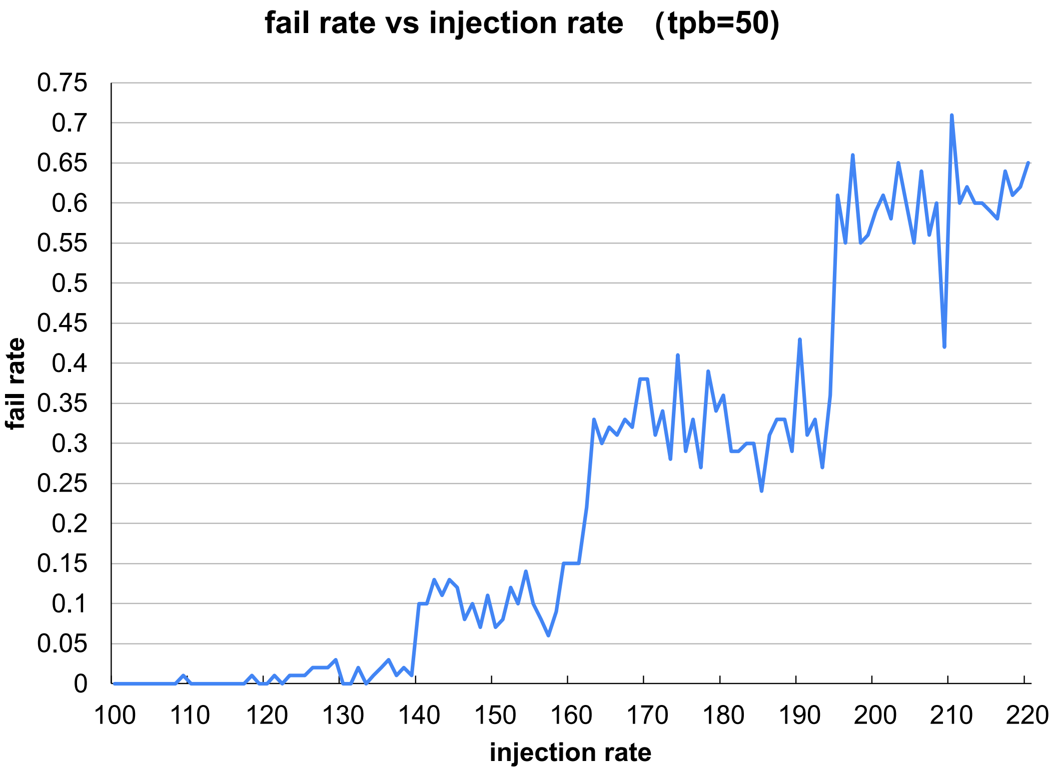

Figure 5.2 fail rate when transaction per block is 50

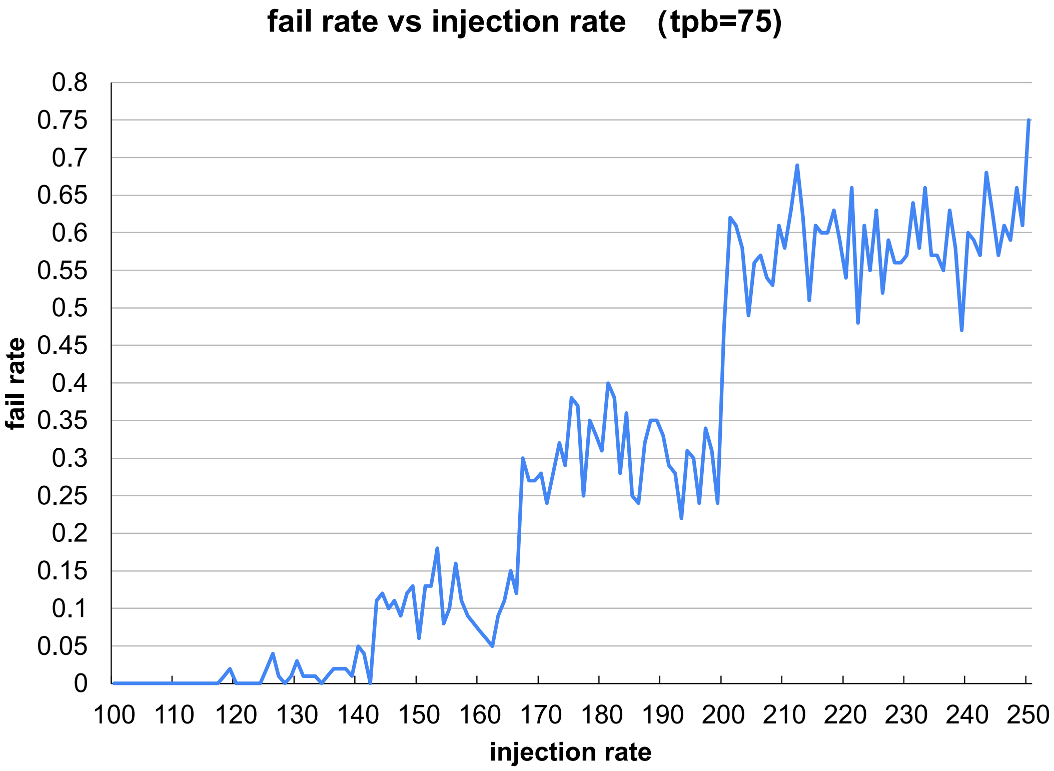

Figure 5.3 fail rate when transaction per block is 75

5.3.3. Lighting Network Simulation

The analysis in section 2.6.5 highlights the lighting network as a promising solution for cloud game trading. Though it offers the potential for rapid and secure transactions, the application of the lighting network within the gaming environment is still an emerging concept that warrants further exploration. Future work could focus on developing a simulation for cloud game trading using the lighting network. This research would entail a detailed examination of transaction speed, security, and scalability and could provide valuable insights into optimizing real-time in-game transactions. The creation and testing of this simulation could significantly advance both the theoretical understanding and practical application of the lighting network in the gaming industry.

References

Ai, Y., Peng, M., and Zhang, K. (2018). Edge computing technologies for Internet of Things: a primer. *Digital Communications and Networks*, 4(2):77–86.

Alharby, M. and van Moorsel, A. (2020). Blocksim: An extensible simulation tool for blockchain systems. *Frontiers in Blockchain*, 3.

all rights reserved copyright 2012, P. C. (2012). Comprehensive solutions and products for video compressionists. Accessed: 30/07/2021.

Arlinghaus, S. L. (c1994). *Practical handbook of curve fitting*. CRC, Boca Raton ; London. Includes index.

Baniata, H. and Kertesz, A. (2021). Fobsim: an extensible open-source simulation tool for integrated fog-blockchain systems. *PeerJ Computer Science*, 7:1–40.

Besancon, L., Silva, C. F. D., and Ghodous, P. (2019). Towards blockchain interoperability: Improving video games data exchange. *ICBC 2019 - IEEE International Conference on Blockchain and Cryptocurrency*, pages 81–85.

Bhojan, A., Ng, S. P., Ng, J., and Ooi, W. T. (2020). Cloudygame: Enabling cloud gaming on the edge with dynamic asset streaming and shared game instances. *Multimedia Tools and Applications*, 79(43):32503–32523.

Böhme, R., Christin, N., Edelman, B., and Moore, T. (2015). Bitcoin: Economics, technology, and governance. *Journal of Economic Perspectives*, 29(2):213–38.

Chen, H., Zhang, X., Xu, Y., Ren, J., Fan, J., Ma, Z., and Zhang, W. (2019). T-gaming: A cost-efficient cloud gaming system at scale. *IEEE Transactions on Parallel and Distributed Systems*, 30(12):2849–2865.

Chohan, U. W. (2021). Non-fungible tokens: Blockchains, scarcity, and value. *Economics of Innovation eJournal*.

Conoscenti, M., Vetrò, A., and Martin, J. C. D. (2021). Cloth: A lightning network simulator. *SoftwareX*, 15.

Di Domenico, A., Perna, G., Trevisan, M., Vassio, L., and Giordano, D. (2020). A network analysis on cloud gaming: Stadia, geforce now and psnow. *arXiv preprint arXiv:2012.06774*.

Fecheyr-Lippens, A. (2010). A review of http live streaming.

Fontes Rebello, G. A., Franco Camilo, G., Potop-Butucaru, M., Campista, M. E. M., Dias de Amorim, M., and Maciel Kosmalski Costa, L. H. (2022). PCNsim: A Flexible and Modular Simulator for Payment Channel Networks. In *2022 IEEE INFOCOM*, London, United Kingdom. IEEE.

Foytik, P., Shetty, S., Gochhayat, S. P., Herath, E., Tosh, D., and Njilla, L. (2020). A blockchain simulator for evaluating consensus algorithms in diverse networking environments. *Proceedings of the 2020 Spring Simulation Conference, SpringSim 2020*.

Gervais, A., Karame, G., Wüst, K., Glykantzis, V., Ritzdorf, H., and Capkun, S. (2016). On the security and performance of proof of work blockchains. In *Proceedings of the 23nd ACM SIGSAC Conference on Computer and Communication Security (CCS)*. ACM.

Harshini, V. M., Danai, S., Usha, H. R., and Kounte, M. R. (2019). Health record management through blockchain technology. *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, 2019-April(Icoei):1411–1415.

Huang, C.-Y., Chen, K.-T., Chen, D.-Y., Hsu, H.-J., and Hsu, C.-H. (2014). Gaminganywhere: The first open source cloud gaming system. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 10(1s):1–25.

Ito, M. (2005). Technologies of the childhood imagination: Yugioh, media mixes, and everyday cultural production. *Structures of participation in digital culture*, pages 88–111.

Malik, S., Dedeoglu, V., Kanhere, S. S., and Jurdak, R. (2019). TrustChain: Trust management in blockchain and iot supported supply chains. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, pages 184–193.

Montecchi, M., Plangger, K., and Etter, M. (2019). It's real, trust me! establishing supply chain provenance using blockchain. *Business Horizons*, 62:283–293.

Nakamoto, S. (2019). Bitcoin: A peer-to-peer electronic cash system. Technical report, Manubot.

Poon, J. and Dryja, T. (2016). The bitcoin lightning network: Scalable off-chain instant payments. *Draft version 0.5*, 9.

Rayna, T. and Striukova, L. (2014). 'few to many': Change of business model paradigm in the video game industry. *Microeconomics: Production*.

Regner, F., Urbach, N., and Schweizer, A. (2019). Nfts in practice - non-fungible tokens as core component of a blockchain-based event ticketing application. In *International Conference on Interaction Sciences*.

Satyanarayanan, M. (2017). Edge Computing. *Computer*, 50(10):36–38.

Schulzrinne, H., Rao, A., and Lanphier, R. (1998). Rfc2326: Real time streaming protocol (rtsp).

Seeling, P., Fitzek, F. H., Ertli, G., Pulipaka, A., and Reisslein, M. (2010). Video network traffic and quality comparison of vp8 and h. 264 svc. In *Proceedings of the 3rd workshop on Mobile video delivery*, pages 33–38.

Sigwart, M., Borkowski, M., Peise, M., Schulte, S., and Tai, S. (2019). Blockchain-based data provenance for the internet of things. In *Proceedings of the 9th International Conference on the Internet of Things*, IoT '19, New York, NY, USA. Association for Computing Machinery.

Sodagar, I. (2011). The mpeg-dash standard for multimedia streaming over the internet. *IEEE multimedia*, 18(4):62–67.

Wahab, A., Ahmad, N., and Schormans, J. (2020). Variation in qoe of passive gaming video streaming for different packet loss ratios. In *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–4. IEEE.

Zhao, J., Chi, Y., Wang, Z., Leung, V. C., and Cai, W. (2020). Cloudarcade: A blockchain empowered cloud gaming system. In *Proceedings of the 2nd ACM International Symposium on Blockchain and Secure Critical Infrastructure*, BSCI '20, page 31–40, New York, NY, USA. Association for Computing Machinery.