

Rollback Netcode, Implmentation and Adoption

Edward Boulderstone

School of Computing Science, Newcastle University, UK

Abstract

Rollback netcode is a peer to peer networking soultion. It had potential to improve online experiences by minimizing the effects of latency over traditional delay based . However the implementation of rollback netcode in the fighting game industry has been slow and difficult. This project aims to understand understand how the industry has had difficulty implementing rollback, and find any optimizations that can be made to existing public rollback netcode understanding.

Keywords: rollback, netcode, peer to peer, fighting games, networking, industry.

1 Introduction

The goal of networking in video games is to allow people from all over the world to play with each other. However fans of fighting games have historically (Before the pandemic) gone out of their way to organise local tournaments with most major tournaments before the pandemic taking place off-line[2]. This trend started because of the when fighting games were played in arcades in the mid 1990's,[3] and continued until the start of the pandemic, in part because fighting games rely on consistent timing and a low latency environment [7], [8]. The existing networking solutions at the time simply did not provide this environment for competitive play[4]. However the spread of the corona virus in the mid 2020's fighting games found themselves having to fall back onto these "lower quality" online platforms. Games which did not have well optimized netcode found themselves on the back foot, with reduced attention [5], and games with well written netcode found their success [6].

1.1 Delay Based Netcode

The first solution to online play for fighting games was a peer to peer system known as delay based netcode. Peer to peer netcode is important for fighting games because of the aforementioned need for low latency. Because most fighting games are

¹ Email: E.Boulderstone@ncl.ac.uk

between two players [10], introducing a server will increase the ping between the two players, as the information has to travel to a separate location, before being sent to the opposing player. Delay based netcode works by keeping two players games identical, by waiting for the remote player's input, before running the game update

1.2 Rollback Netcode

Rollback netcode was developed in 2006 as a solution to the problems with delay based netcode. [9]. It works on top of existing delay based netcode by adding rollback frames, predicting the remote users inputs. When the remote user's input arrives if it matches with the predicted input, nothing happens, however if there's a discrepancy the game will "rollback", to the point of the discrepancy and re-simulate the game state back to the real time frame as shown in figure 1.

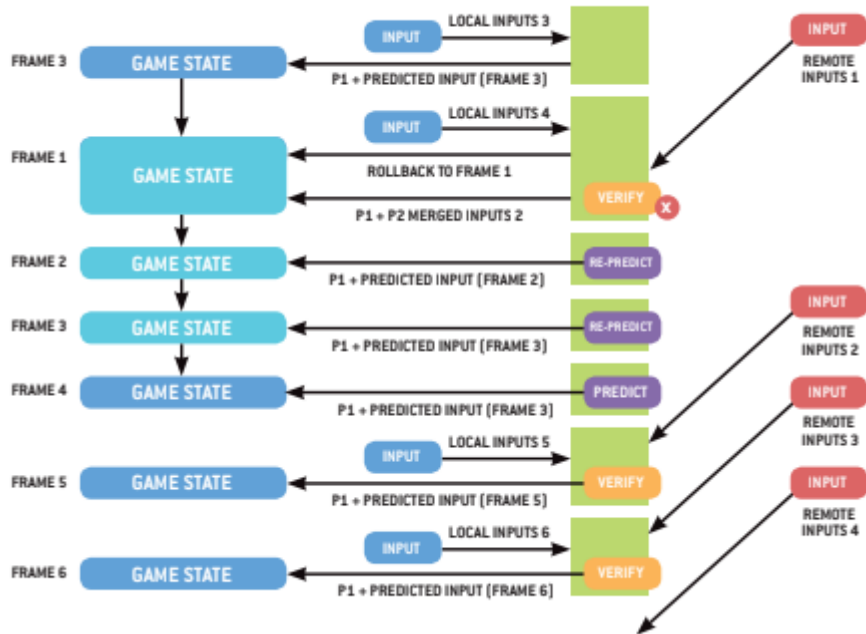


Fig. 1. Rollback netcode [1]

1.3 *Difficulties in industry*

In today’s fighting game market, many games have rollback [13]. However there are still notable exceptions such as:

- Super Smash Bros. Ultimate[14]
- Granblue Fantasy Versus[15]
- Under Night In-Birth[16]
- Samurai Shodown[17]
- Soulcalibur VI[18]
- Dead or Alive 6[19]
- EA Sports UFC 4[20]
- Dragon Ball FighterZ[21]

Other fighting games have had difficulty in implementing rollback, such as Street fighter V and Mortal Kombat X. These difficulties with implementing and developing rollback are the basis of the motivation of this paper.

1.4 *Aim*

To investigate rollback netcode, it’s usage in industry and any short comings of existing public rollback netcode infrastructure

1.5 *Objectives*

- Understand rollback netcode and the effects on the games it’s implemented in.
- Create a visualization for the differences between rollback and delay based netcode.
- Research the difficulties of implementing rollback in existing games.
- Explore optimizations for the existing public rollback structure.
- Investigate further uses of rollback netcode, in the wider video game industry.

2 Background

This section aims to provide information surrounding the different technologies and concepts which relate to the project objectives. There is a more in depth explanation on the structure of netcode, and some of the common problems it can encounter.

Peer to peer networks over the internet communicate data in packets which can run into the following issues.

- Packets take time to reach their destination (Network Latency).
- Packets can get lost on their way (Packet Loss).
- Packets can get there, but have their data corrupted (Corruption).
- Computers can run at different speeds.
- Computers can occasionally get hung up on doing things and skip a frame or two.

2.1 Delay-based netcode

2.1.1 Concept

Delay based netcode works by keeping both players games in lockstep, meaning that each player's simulation of the game waits for the input of the remote player, before simulating the frame[11]. This system works well when latency is not a major factor, for example, in a turn based game, where the inputs are spread apart by seconds, a pause of half a second may go unnoticed. Input latency frames are added to compensate for the Network Latency, and allow the local inputs to be sent before the frame they would be read on as shown in figure 2.

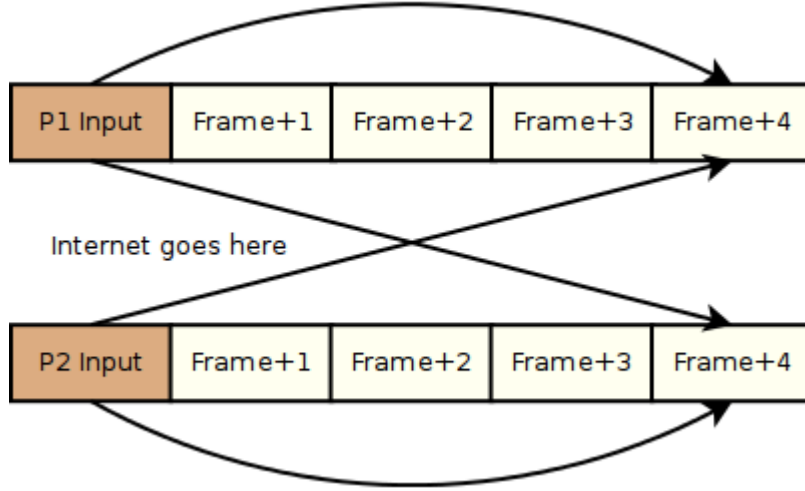


Fig. 2. 4 Frame Input Delay Example [12]

The Input delay to be used can be calculated as follows:

$$InputLatencyFrames = Ceiling((RoundTripLatency/2 + Constant)/FrameDuration)$$

Where the constant represents additional frames of delay to compensate for variance in Network Latency.

2.1.2 Improvements

However this model does not take into account potential for packet loss/ corrupted packets, which may lead the actual states to progress as shown in figure 3

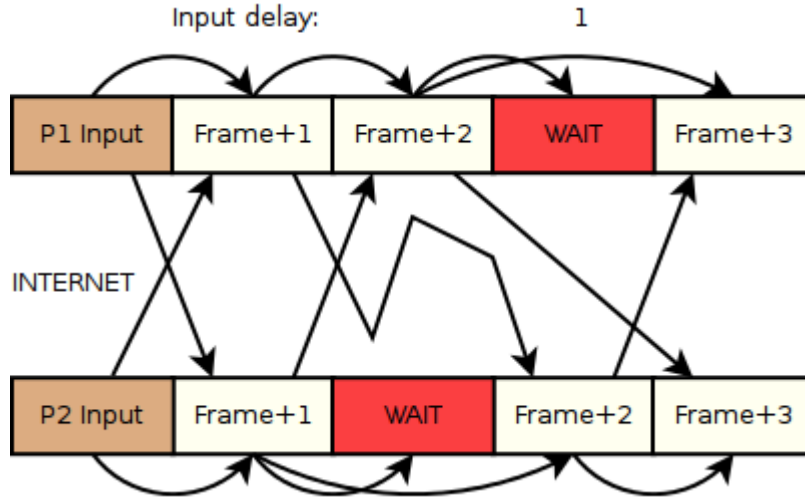


Fig. 3. Packet Loss's Affect on Game State progression [12]

One unexpected side effect of packet loss with single input packets is knock on delays, where by a delay in one game, makes the frame process later, so the inputs for that frame are postponed, delaying the origins game.

To combat packet loss and data corruption multiple inputs can be sent with one packet, extra frames of input delay can be added, and inputs can be sent multiple times, as shown in 4

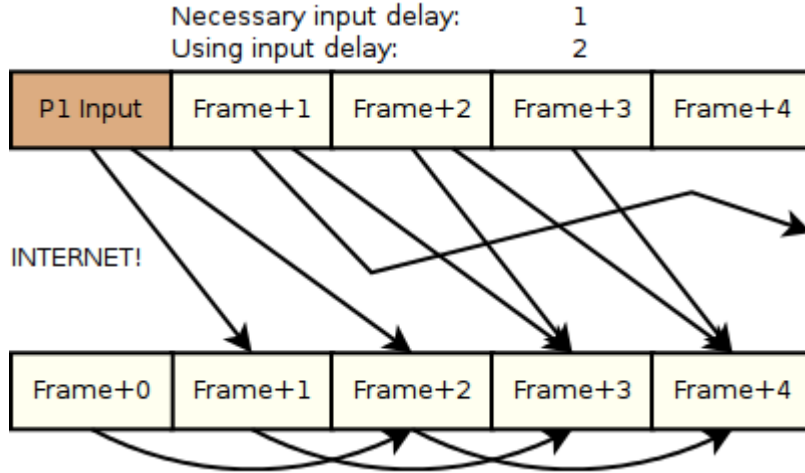


Fig. 4. Potential Solutions to Packet Loss in Delay Based Netcode [12]

Another improvement to made to delay based netcode is to account for Network Latency variance with a dynamic number of delay frames. way to account for Network Latency variance is to have a dynamic number of Delay Frames, varying with average Latency[7]. This can hide stutters, and remedy degrading network conditions causing consistent pauses. However, consistency for fighting games is the one of the highest determiners on network quality[22].

2.1.3 Evaluation

When there is minimal Network Latency variance, delay based netcode functions similarly to a non networked experience, with increased Input Latency. This is the ideal case for delay based networking. However, when the Network Latency variance is high, the game can freeze at seemingly random moments, removing agency from the player in the middle of a game. This breaks user immersion and significantly decreases the quality of the experience[7].

Ultimately Delay based netcode functions as network solution for high quality connections, but struggles when operating as connection quality degrades. [23].

2.2 Rollback netcode

Rollback netcode takes the existing framework of delay based netcode and builds on top of it. By not requiring the input of the remote user to simulate a frame, it can tolerate late and lost packets better than delay based [1]. When a packet is lost or late, the rollback frames act as a buffer preventing the from game freezing. In theory one could use the rollback buffer to reduce apparent network latency as well. This will be explored later.

max tolerable ping increases

2.2.1 Difficulties in Implementation

- separate gameplay from rendering - serializable game state - particle simulation - object lifetime - sound fx - animation system - UI - desync detection

2.3 Third party Rollback

- Slippi - Fightcade

3 Effects of Rollback on Game Design

3.1 8 Frames

3.2 Audio Design

3.3 Visual Design

4 Improvements for Rollback

I propose a theory for rollback frames

for dynamic: dynamically changing rollback = minimal effect, (Problem with maxing rollback?), however try to keep delay static for static: use upperbound of tolerable network connections (cite Mortal kombat)

4.1 *Prediction Quality*

4.2 *Packet Contents Optimization*

4.3 *Input Locking*

5 Rollback Visualization tool

5.1 *Design*

6 Wider use case of rollback netcode

Rollback ideas that only apply to fighting games? (Visual teleportation?, Expensive to implement?,) RTS (Including sports games)

Prediction quality: - Connection ‘quality’. This doesn’t mean bandwidth, but rather how much a connection’s response time fluctuates. This has improved in modern times, making rollback a much more viable option than it once was. - The type of game. Because rollback needs you to be able to revert the state of the game in some cases, trying to implement it in a game with hundreds of different units all acting at once is extremely challenging (not to mention very offputting for the player). As a result, it’s more suited to fighting games and other games with fewer player-controlled assets flying around at any given time.

7 Evaluation

This section might alternatively be called “Experimental results”. It describes how you assessed your project and what you found out. Where you give results in tables or graphs, remember to highlight in the text the key points that the data shows and, if possible, try to explain why you got any unexpected results.

8 Conclusions and further work

In this section you summarise what you did and discovered and (importantly) what else you would have done (or done differently) if you had the chance. It is a chance to reflect on your success.

References

- [1] https://drive.google.com/file/d/1nRa3cRBQmKj0-SEyrT_1VNOkPOJWNhVI/view
or https://web.archive.org/web/20220101162600/https://drive.google.com/file/d/1nRa3cRBQmKj0-SEyrT_1VNOkPOJWNhVI/view Tony C., GGPO Game Developer Magazine’s article. 2012. (Visited 25/05/2022)
- [2] https://liquipedia.net/fighters/Tier_1_Tournaments
or https://web.archive.org/web/20210121113554/https://liquipedia.net/fighters/Tier_1_Tournaments
Liquidpedia list of major fighting game tournaments. (Visited 25/05/2022)
- [3] <https://www.usgamer.net/articles/the-oral-history-of-evo> or <https://web.archive.org/web/20220321000753/https://www.usgamer.net/articles/the-oral-history-of-evo> John L., The Oral History of EVO: The Story of the World’s Largest Fighting Game Tournament (Visited 25/05/2022)

- [4] <https://esportsinsider.com/2021/10/can-fighting-games-become-a-mainstream-esport/> or <https://web.archive.org/web/20220516060805/https://esportsinsider.com/2021/10/can-fighting-games-become-a-mainstream-esport/> Elizbar R., Can fighting games become a mainstream esport without abandoning grassroots?. (Visited 26/02/2022)
- [5] https://www.ssbwiki.com/COVID-19_pandemic_and_its_impact_on_competitive_Smash or https://web.archive.org/web/20220516020551/https://www.ssbwiki.com/COVID-19_pandemic_and_its_impact_on_competitive_Smash List of smash tournaments affected by the pandemic. (Visited 26/02/2022)
- [6] <https://techraptor.net/gaming/opinions/guilty-gear-strive-changed-fighting-games-rollback-netcode> or <https://web.archive.org/web/20210713144527/https://techraptor.net/gaming/opinions/guilty-gear-strive-changed-fighting-games-rollback-netcode> Davi B. Guilty Gear Strive Might Have Changed Fighting Games Forever. (Visited 26/02/2022)
- [7] <https://arstechnica.com/gaming/2019/10/explaining-how-fighting-games-use-delay-based-and-rollback-netcode/> or <https://web.archive.org/web/20220425211823/https://arstechnica.com/gaming/2019/10/explaining-how-fighting-games-use-delay-based-and-rollback-netcode/> Ricky P. Explaining how fighting games use delay-based and rollback netcode. (Visited 26/05/2022)
- [8] <https://www.polygon.com/2020/3/25/21192522/netcode-samurai-showdown-fighting-games-rollback-delay> or <https://web.archive.org/web/20220401231646/https://www.polygon.com/2020/3/25/21192522/netcode-samurai-showdown-fighting-games-rollback-delay> David C. Bad netcode is killing many of your favorite fighting games. (Visited 26/05/2022)
- [9] https://gamasutra.com/view/news/34050/Interview_How_A_Fighting_Game_Fan_Solved_Internet_Latency_Issues.php or https://web.archive.org/web/20220325062609/https://gamasutra.com/view/news/34050/Interview_How_A_Fighting_Game_Fan_Solved_Internet_Latency_Issues.php Kyle O. Interview: How A Fighting Game Fan Solved Internet Latency Issues. (Visited 26/05/2022)
- [10] <https://archive.org/details/nextgen-issue-015/page/n33/mode/2up> "The Next Generation 1996 Lexicon A to Z: Fighting Game". (Visited 26/05/2022)
- [11] https://www.gamasutra.com/view/feature/3094/1500_archers_on_a_288_network_.php or https://web.archive.org/web/20220506231117/https://www.gamasutra.com/view/feature/3094/1500_archers_on_a_288_network_.php Mark T. 1500 Archers on a 28.8: Network Programming in Age of Empires and Beyond. (Visited 26/05/2022)
- [12] <https://web.archive.org/web/20210228051849/https://mauve.mizuumi.net/2012/07/05/understanding-fighting-game-networking.html> Mauve M. Understanding Fighting Game Networking. (Visited 26/05/2022)
- [13] <https://attackofthefanboy.com/guides/rollback-netcode-games-list/> or <https://web.archive.org/web/20211122034207/https://attackofthefanboy.com/guides/rollback-netcode-games-list/> Andron S. Rollback Netcode Games List (March 2022). (Visited 26/05/2022)
- [14] https://en.wikipedia.org/wiki/Super_Smash_Bros._Ultimate or https://web.archive.org/web/20220519200835/https://en.wikipedia.org/wiki/Super_Smash_Bros._Ultimate Super Smash Bros. Ultimate. (Visited 26/05/2022)
- [15] https://en.wikipedia.org/wiki/Granblue_Fantasy_Versus or https://web.archive.org/web/20220512125807/https://en.wikipedia.org/wiki/Granblue_Fantasy_Versus Grandblue Fantasy Versus (Visited 26/05/2022)
- [16] https://en.wikipedia.org/wiki/Under_Night_In-Birth or https://web.archive.org/web/20220516020108/https://en.wikipedia.org/wiki/Under_Night_In-Birth Under Night In-birth (Visited 26/05/2022)
- [17] [https://en.wikipedia.org/wiki/Samurai_Shodown_\(2019_video_game\)](https://en.wikipedia.org/wiki/Samurai_Shodown_(2019_video_game)) or [https://web.archive.org/web/20220504020927/https://en.wikipedia.org/wiki/Samurai_Shodown_\(2019_video_game\)](https://web.archive.org/web/20220504020927/https://en.wikipedia.org/wiki/Samurai_Shodown_(2019_video_game)) Samurai Shodown (Visited 26/05/2022)
- [18] https://en.wikipedia.org/wiki/Soulcalibur_VI or https://web.archive.org/web/20220421074040/https://en.wikipedia.org/wiki/Soulcalibur_VI Soulcalibur VI (Visited 26/05/2022)
- [19] https://en.wikipedia.org/wiki/Dead_or_Alive_6 or https://web.archive.org/web/20220130144906/https://en.wikipedia.org/wiki/Dead_or_Alive_6 Dead or Alive 6 (Visited 26/05/2022)

- [20] https://en.wikipedia.org/wiki/EA_Sports_UFC_4
or https://web.archive.org/web/20211019192531/https://en.wikipedia.org/wiki/EA_Sports_UFC_4 EA Sports UFC 4 (Visited 26/05/2022)
- [21] https://en.wikipedia.org/wiki/Dragon_Ball_FighterZ
or https://web.archive.org/web/20220424200258/https://en.wikipedia.org/wiki/Dragon_Ball_FighterZ Dragon Ball FighterZ (Visited 26/05/2022)
- [22] <https://www.youtube.com/watch?v=0NLe4IpdS1w>
or <https://web.archive.org/web/20220529231214/https://www.youtube.com/watch?v=0NLe4IpdS1w> Core A Gaming. Analysis: Why Rollback Netcode Is Better. (Visted 31/05/2022)
- [23] <https://www.youtube.com/watch?v=1RI5scXYhK0>
or <https://web.archive.org/web/20220330062542/https://www.youtube.com/watch?v=1RI5scXYhK0> Hold Back To Block. Talking Netcode With Adam "Keits" Heart (Visited 31/05/2022)
- [24] <https://displaylag.com/video-game-input-lag-database/> or <https://web.archive.org/web/20220519183335/https://displaylag.com/video-game-input-lag-database/> Display Lag. Video Game Input Lag Database. (Visted 02/06/2022)