Twitch plays TwiPi: An analysis and comparison with Twitch plays Pokémon

Kedar More
Department of Mechanical Engineering
University of Colorado, Boulder
Boulder, USA
kedar.more@colorado.edu

***Abstract*— Social experiments on crowd-controlled gaming have been done on video games which are devoid of any physical obstructions/limitations. The best example is *Twitch Plays Pokémon* where players from all over the world played Pokémon Red by inputting commands (A, B, up, down, left, right, start, select) on chat while watching the live stream. This paper shows a detailed analysis of varying number of players (1, 2, 4, …) on physical wheeled robot TwiPi (Twitch Raspberry Pi Bot) and its comparison with video game (Pokémon Red). The infinite monkey theorem states that a monkey hitting keys at random on a typewriter keyboard for an infinite amount of time will almost surely type any given text, such as the complete works of William Shakespeare. This situation is quite like the theorem. The only and the most important difference is that instead of random hitting of keys, multiple intelligent entities are trying to solve a problem with a specific goal.**

Keywords—Twich, Pokémon

I. Introduction

Twich is a gamer's live streaming platform where this crowd-controlled game experiment was first tested on a Nintendo GameBoy Colour Pokémon Red. This game was started on an emulator and live streamed on twitch.tv so that players from all over the world were able to see the character.

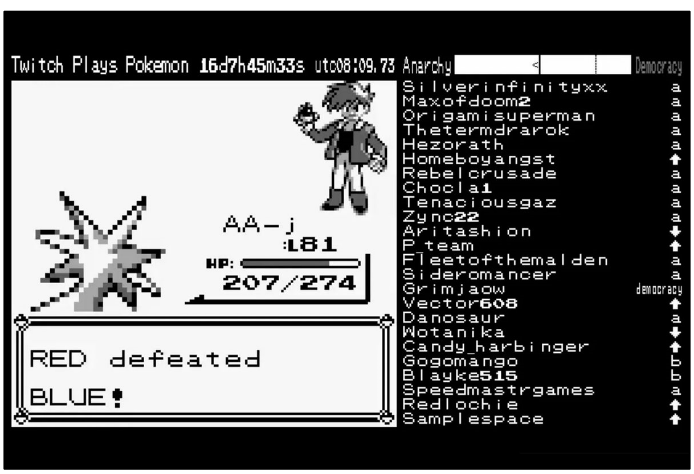


Fig. 1. Players inputting commands and those being excecuted

In the game the character must travel across the map in search of Pokémons and collect badges from in-game bosses. After collecting 8 badges the character can compete Elite 4 and become the Pokémon champion by defeating them. Among these tasks there are a few errands that the character runs to open new places on the map. There are a few tasks where the player has to decide between two routes e.g. you can reach Fuschia City (green) from Safron City (red) by going south of Lavender Town (blue) or by the bike path (yellow) near Celadon City. Each player has different approach to control the character and to choose between alternatives. This is where the conflict among the many players kick in.

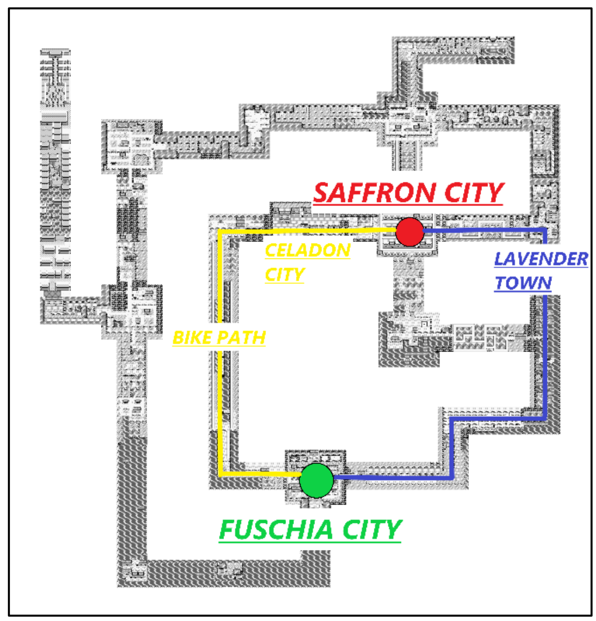


Fig. 2. Map of Pokémon Red with two route options

In TwiPi there are some physical hindrances such as ground friction and motor backlash. This change is addressed in this paper by using a physical robot as a character. Players completing the task remain the same as they are playing real-time. Here the players have to complete tasks with varying difficulty so that analysis of varying amount of time and number of commands inputs used to complete different tasks could be made. Also, in Twitch plays Pokémon the commands were executed 30 to 40 seconds after they were input but, in this experiment, there will be significantly short difference between input and execution.

II. LITERATURE REVIEW

A. A Crude Analysis of Twitch Plays Pokémon

Twitch Plays Pokémon Red has two types of input methods namely Anarchy and Democracy. Anarchy executes all the commands which are input by all the players while democracy take a poll of 20 secs and executes the most voted command. This paper predicts the time taken to complete the game and the probability of winning for N players. This analysis is done with an assumption that there are no non-command texts from the players. The other assumption is that there are only two types of commands namely good and bad. In reality, there can be difference of opinion among the players where both approaches are correct.

B. Twitch Plays Pokémon: A Case Study in Big G Games Dennis

As per the Infinite Monkey Theorem a goal can be achieved given enough time and effort. This paper analyses a similar phenomenon encountered in a social experiment called Twitch Plays Pokémon. Instead of monkeys there were actual people playing the game at the same time. This paper differentiates the players into categories namely explorers, achievers, socializers and killers or griefers and analyses different effects caused due to this socio-psychological experiment. It concludes that randomness caused by many intelligent entities not only achieves a goal but also delivers insightful and rich meaning behind this collective effort. This gave rise to many lore pertaining the Helix and Dome fossils (in-game items).

C. The dynamics of collective social behavior in a crowd controlled game

Despite many efforts, the behavior of a crowd is not fully understood. The advent of modern communication means has made it an even more challenging problem, as crowd dynamics could be driven by both human-to-human and human-technology interactions. Here, we study the dynamics of a crowd controlled game (Twitch Plays Pokémon), in which nearly a million players participated during more than two weeks. Unlike other online games, in this event all the players controlled exactly the same character and thus it represents an exceptional example of a collective mind working to achieve a certain goal. We dissect the temporal evolution of the system dynamics along the two distinct phases that characterized the game. We find that having a fraction of players who do not follow the crowd's average behavior is key to succeed in the game. The latter finding can be well explained by an nth order Markov model that reproduces the observed behavior. Secondly, we analyze a phase of the game in which players were able to decide between two different modes of playing, mimicking a voting system. We show that the introduction of this system clearly polarized the community, splitting it in two. Finally, we discuss one of the peculiarities of these groups in the light of the social identity theory, which appears to describe well some of the observed dynamics.

III. DESCRIPTION OF TWIPI

Designing of the robot for the experiment includes 4 parts:

1) *Physical body of the TwiPi*
2) *Processor to take inputs from the players and give commands to body*
3) *Server to connect the players to the robot*
4) *Integration of hardware and software*

A. Physical Body of TwiPi

The chassis used is Yahboom Raspberry Pi 4WD Robotic Car Kit. The dimensions of the final assembly are 25.4cm*17.78cm*20.16cm. TwiPi is a mid-sized robot with 4 wheels. Material of the chassis is aluminum with holes for attachments of motors, expansion board and Raspberry Pi and the camera. The camera used is an HD Camera USB module with resolution 640x480@30 ('height'x'width'@'framerate') which is mounted on the rear side of TwiPi. The camera won't move with respect to the TwiPi and give a First-Person Perspective.

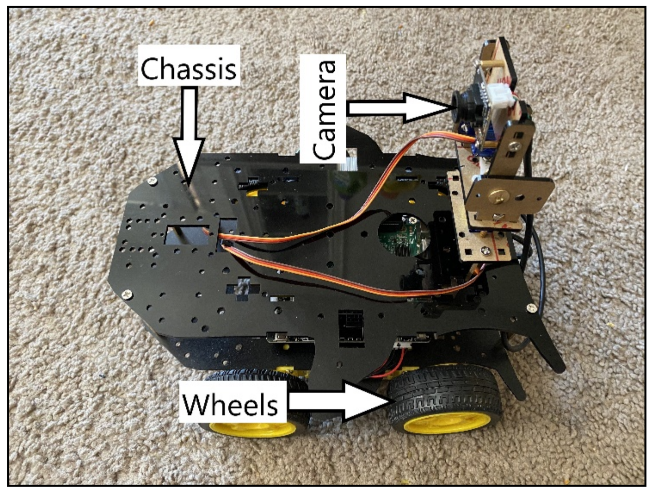


Fig. 3. Parts of Robot

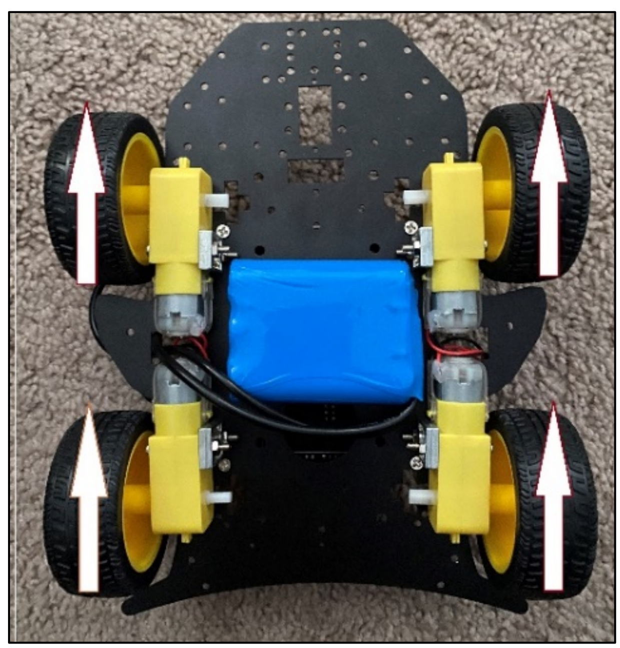


Fig. 4. Example of a Motion (Forward)

B. Processor

The Raspberry Pi is like a connection between the body and server. The server runs on the Pi and listens to the messages on the chat continuously.

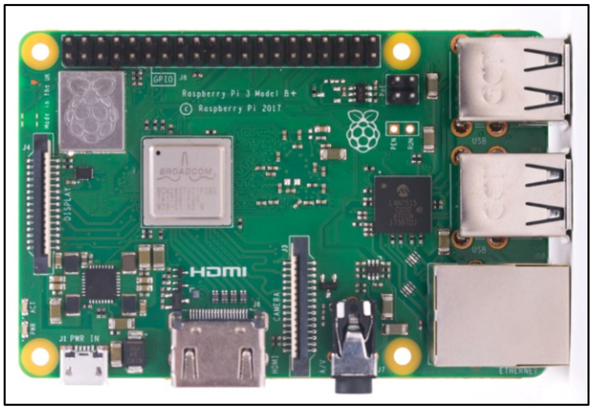


Fig. 5. Raspberry Pi

The expansion board in the kit has in-built motor drivers which take power input from external source and logic (HIGH or LOW) from Raspberry Pi to run the motors.

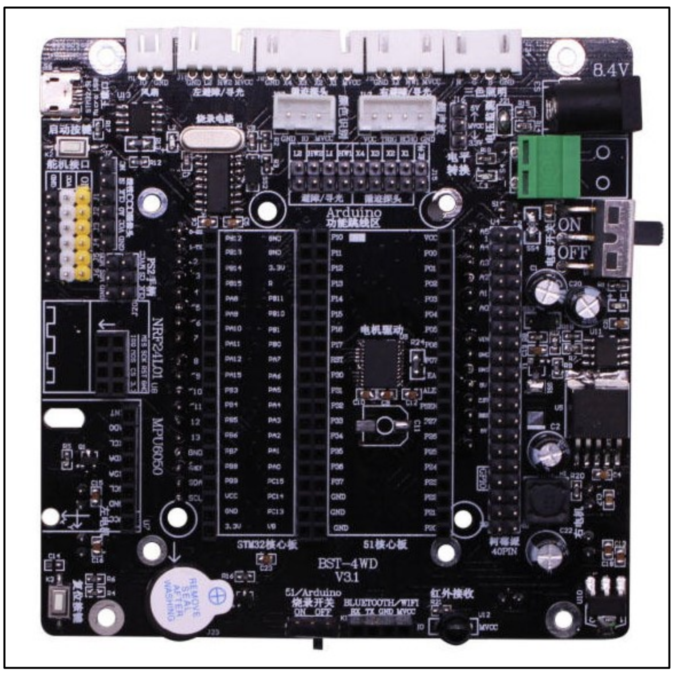


Fig. 6. Expansion Board

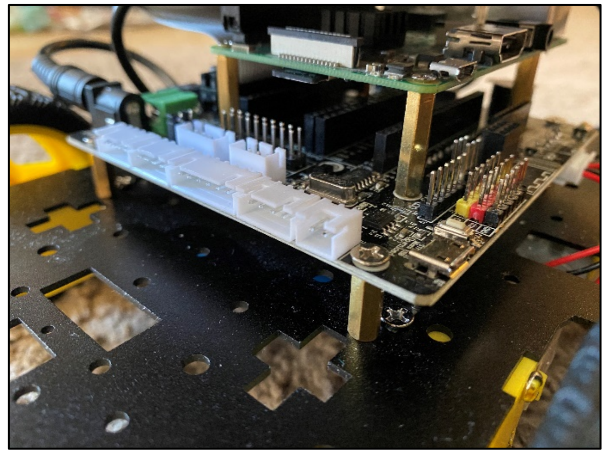


Fig. 7. Connection of Expansion Board to Processor (Raspberry Pi)

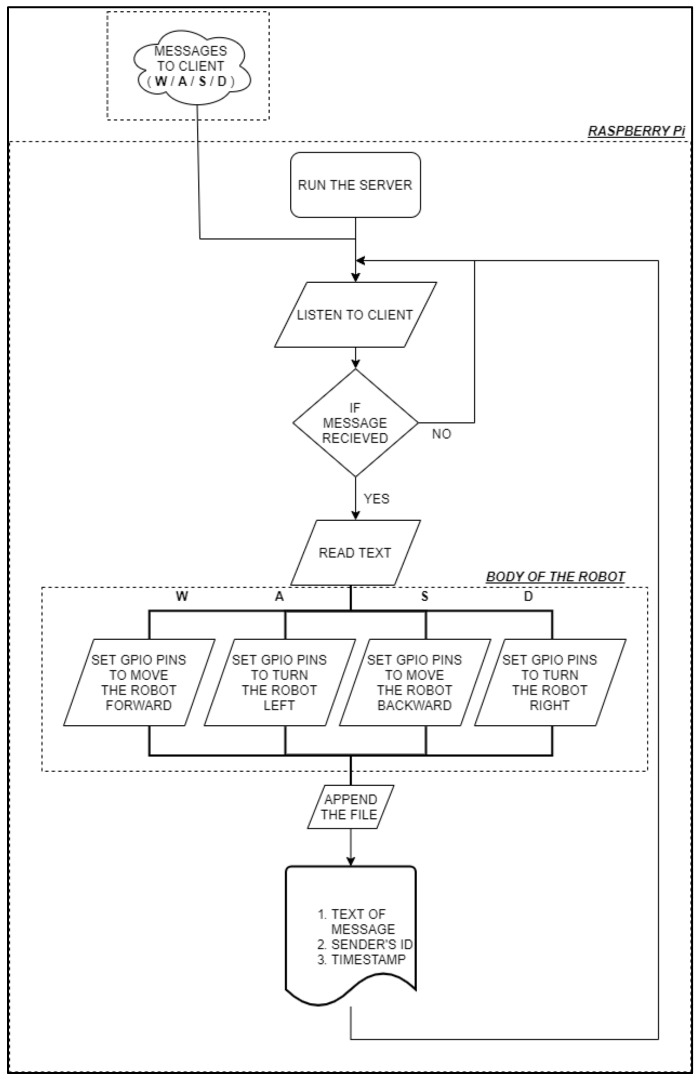


Fig. 8. Working flowchart

C. Server

To take commands from the players over the internet, YouTube Live Chat is used with a python library ytcaht. The players message the client on the chat window (right) with one of the WASD commands for forward, left, back, right respectively. This is done while watching the live stream (left) from the camera on TwiPi. The same and every command will be executed on TwiPi in the order of its input.

A log file (.csv) is generated containing the raw data of command texts, name of player and timestamp.

e.g.

w	Player1	10.7753743
A	Player2	11.4783468
s	Player1	11.7789763
stop	Player4	11.9429476
d	Player3	12.1232432
A	Player2	12.3454563

There is a 5 seconds delay between the input of the command and its execution. In the original Twitch Plays Pokemon Red there was a delay of 30 to 40 seconds in execution of commands. Even now when Twitch conducts the live game stream there is a 10 second delay.

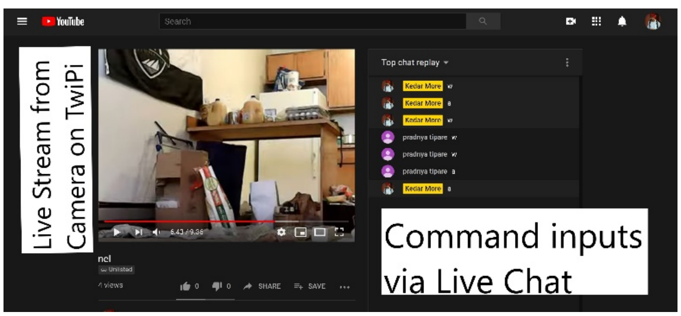


Fig. 9. Screenshot of Messenger chat service with command inputs for TwiPi

D. Integration

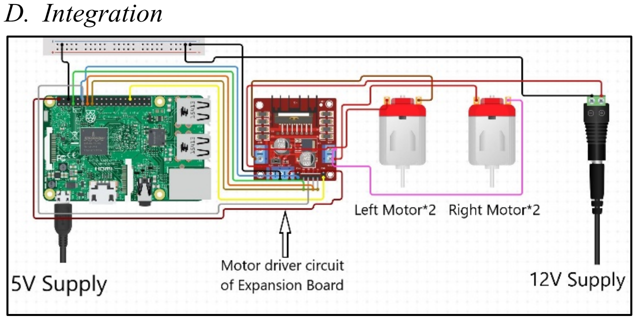


Fig. 10. Basic Circuit Representation

IV. EXPERIMENTS

A. Increasing difficulty (levels) with varying number of players (1, 2, 4).

The red circle indicates the starting position of TwiPi whereas the green circle indicates the targets it must reach. The red rectangle is levels 3, 4 and 5 is a barrier which has to be dodged by TwiPi.

The players can decide (individually) which target to reach first and which route to take for consecutives targets.

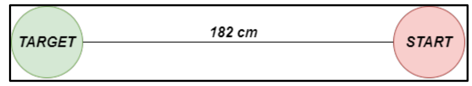


Fig. 11. Top view of Level 1 path

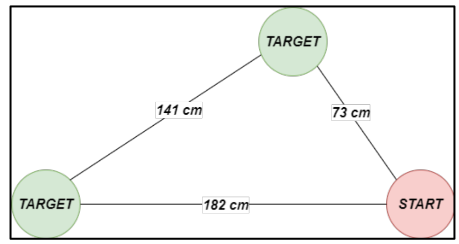


Fig. 12. Top view of Level 2 path

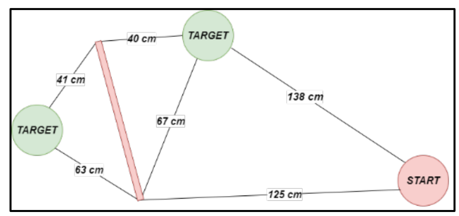


Fig. 13. Top view of Level 3 path

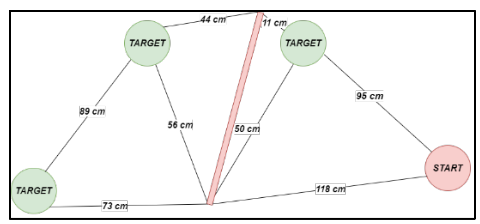


Fig. 14. Top view of Level 4 path

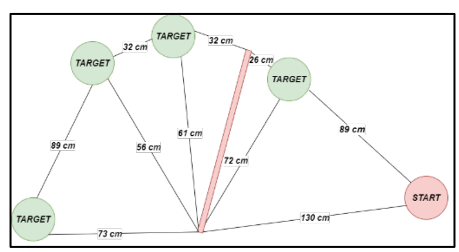


Fig. 15. Top view of Level 5 path

B. Longer path with maze and many players with non-continous participation

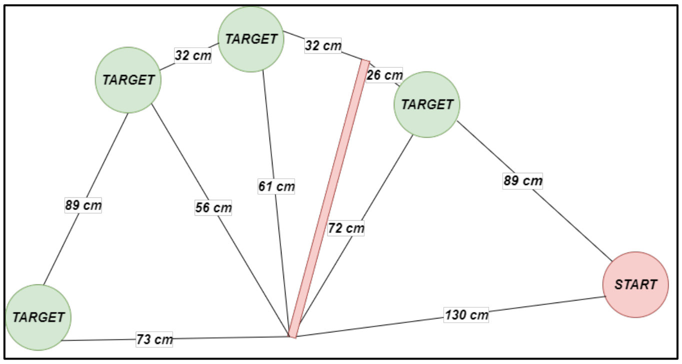
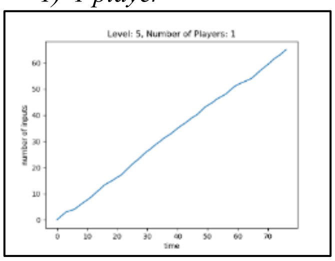


Fig. 16. Maze

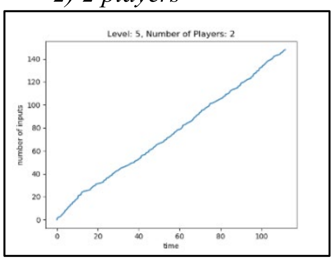
V. OBSERVATIONS ON LESS NUMBER OF PLAYERS (EXPERIMENT *A.*)

A. Number of Inputs over Time (Level 5 path)

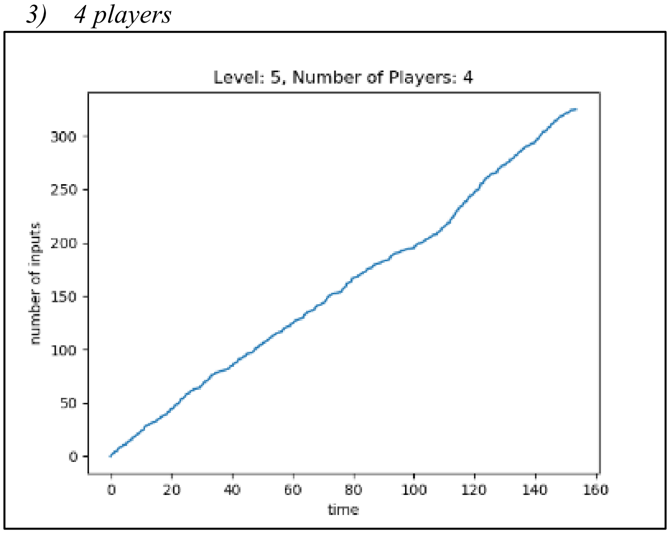
1) 1 player



2) 2 players



3) 4 players



As expected, the no of inputs increases in an irregular manner with time. The slope of this line can be used to find the change in value of inputs with time.

TABLE I. summaries the total time taken by varying no of players in the all the levels.

TABLE I. TOTAL TIME TAKEN TO COMPLETE THE LEVELS (SECONDS)

Levels	**Number of Players**		
	1	***2***	***4***
1	31.12	23.09	27.63
2	41.59	44.09	79.91
3	63.56	121.29	56.64
4	73.24	71.38	87.88
5	76.02	111.49	153.49

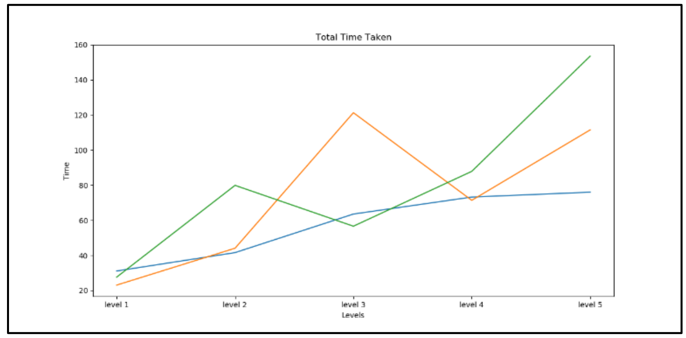


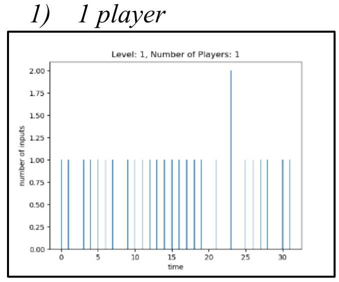
Fig. 17. Graphical Visualisation of Total Time

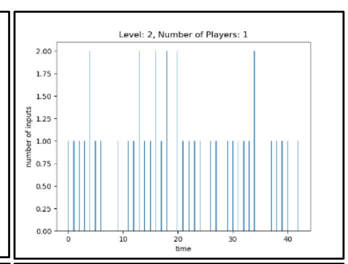
This trend of total time versus the number of players and complexity does not change linearly for lower number of players. But as the complexity will change significantly the total time will also increases almost linearly.

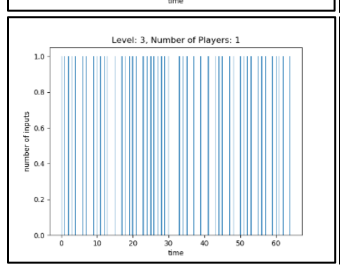
B. Number of Inputs per Second

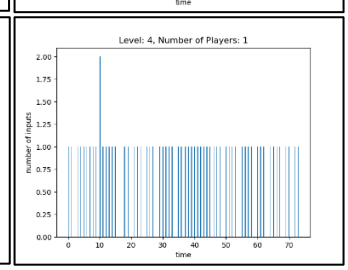
A more interesting trend to observe would be the number of inputs given in the interval of 1 second. These graphs can be seen as slope of the previous number of inputs vs time graphs. For simplicity and understanding this slope is represented as bars in a discreteway. We can conclude about the interest of players in the game with the density of lines on the x axis.

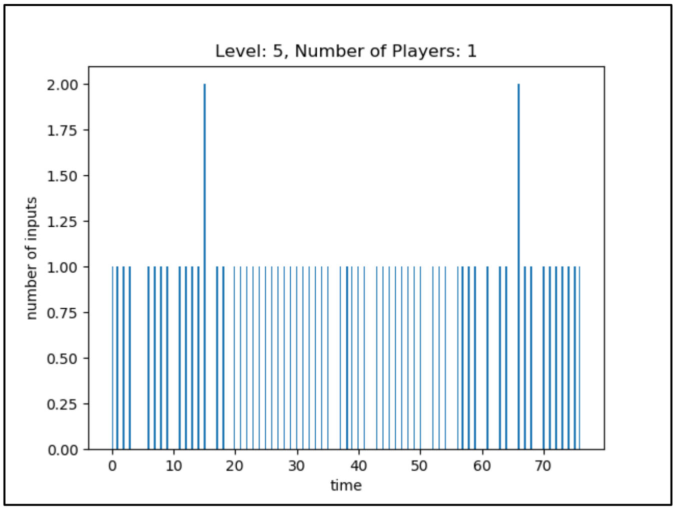
1) 1 player





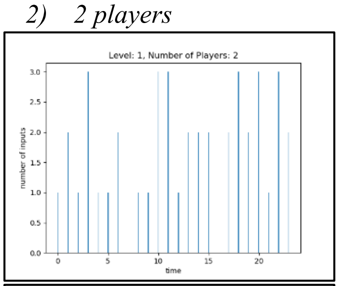


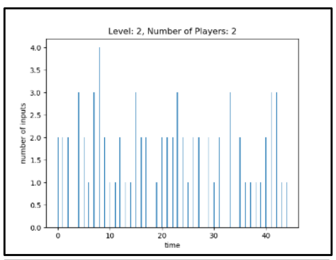


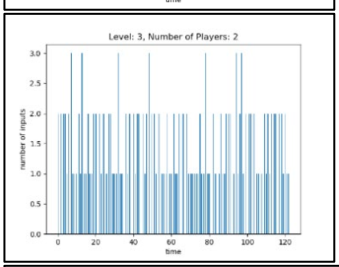


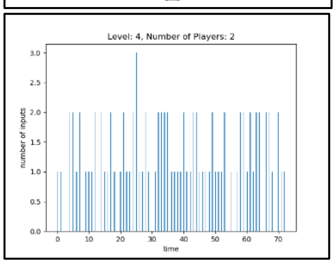
The increased density at the time 15, 39 and 65 seconds is due to the proximity of first target to the barrier, turning around the barrier and less space around the second target. The same trend is seen with 2 and 4 players. The only difference being the increased number of inputs per second.

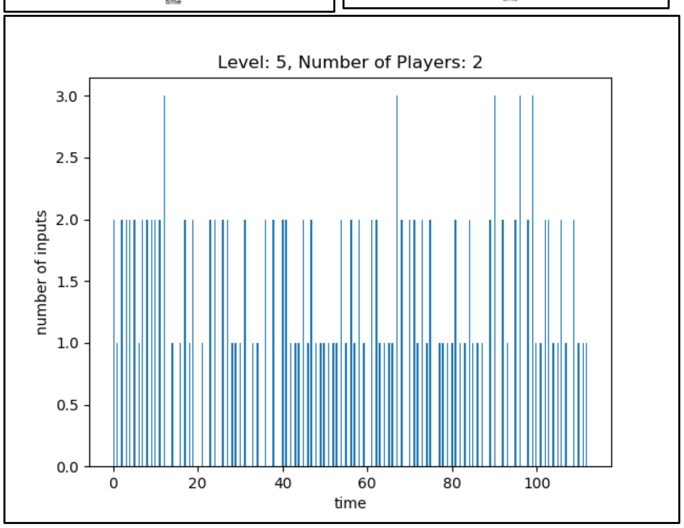
2) 2 players





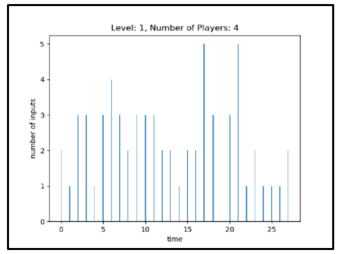


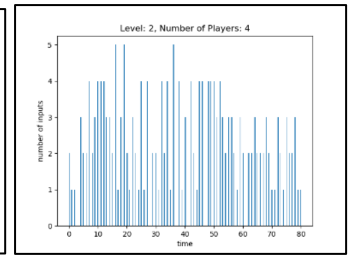


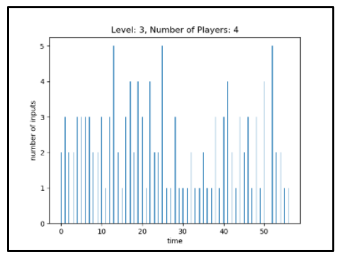


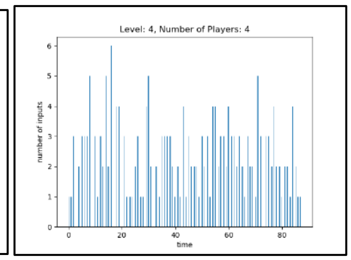
Here the increased density is more spread out due to the confusion created between the players about which route to take and the overlapping in the inputs. Overlapping of inputs is the major cause of anarchy in crowd-based gameplays. When more than one player thinks of a move and tries to implement it at the same time the character (TwiPi in this case) considers these to be multiple inputs and implements everything.

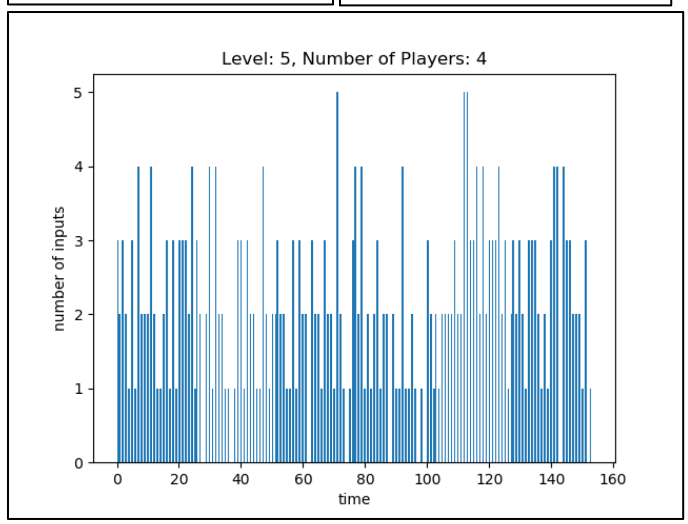
3) 4 players











We can make out the high-density patches at 15, 75 and 140 seconds. This is similar to the single player with mapped time interval. It is natural for the confusion to rise with many players and close spaces. There are a few patches where exact inputs are needed to complete the obstacle. Even if one player inputs a wrong command the obstacle is to be started all over again.

Such an obstacle was faced in Pokemon Red which is famous by the name "The Ledge".

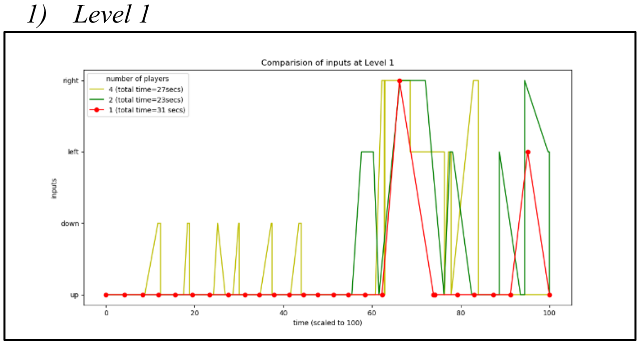
Fig. 18. Screenshot of Pokemon Red with the character jumping over "The Ledge"

Ledges are in-game obstacle from which we can jump down but can't climb up. Such obstacles need much cooperation and are also reasons of many inputs given at the same time.

C. Comarision with single player inputs

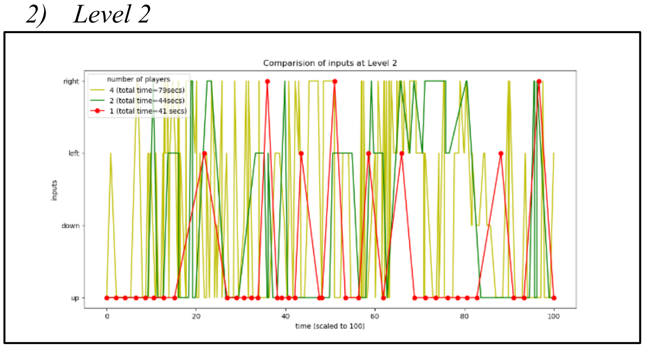
The Red dots indicate the correct inputs (single player) which should be given in order to complete the task. Time of all the inputs (1, 2 and 4 players) is scaled to 100 for comparison along the path paved by single player. This enables us to check the trend of inputs at the same position along the path (at different times as per number of players).

1) Level 1

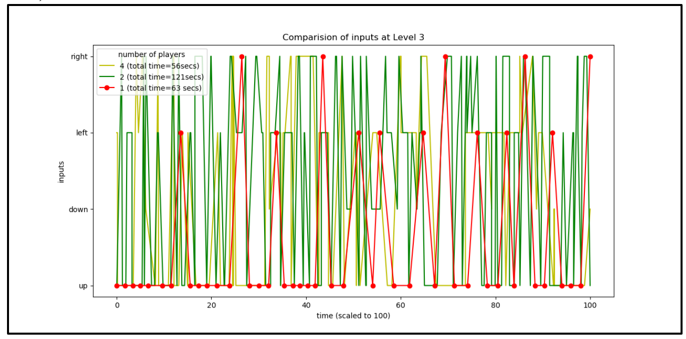


From this graph we can easily make out the one of the players was inputting 'down' instead of 'up'. This type of player in crowd-based gaming is known as a 'Rouge'. These players want to create more chaos and confusion within the game in turn increasing the total time. After 60 seconds the chaos was started to align TwiPi in the straight direction. This is where overlapping of inputs come in. Some commands are given to rectify this overcompensation.

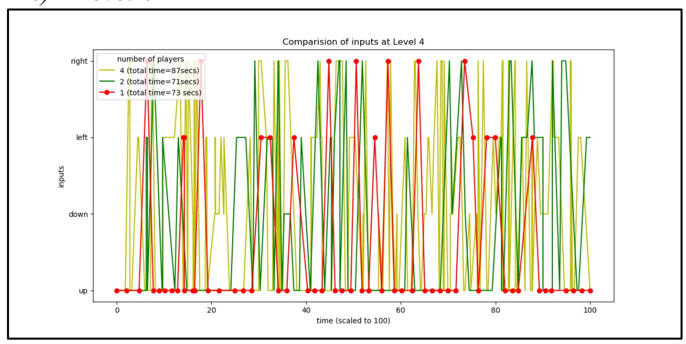
2) Level 2



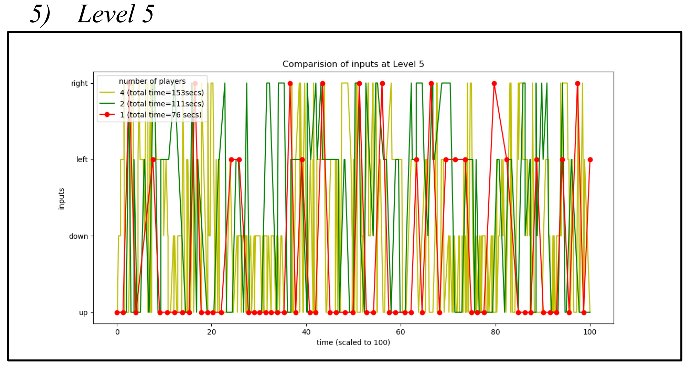
3) Level 3



4) Level 4

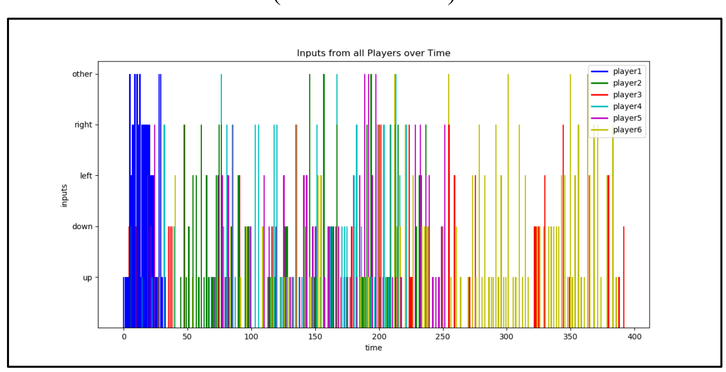


5) Level 5



All other graphs show total chaos by the players. We can see a similar trend in the inputs from single player and multiple players. The reason is that this is a simple task with limited possibilities for completion. When the complexity of the task increases there will be many possibilities with which all the players will be divided. The confusion created in that situation will be more in terms of randomness.

VI. Observation on Many Players (Experiment *B*.)



VII. Conclusion

References

[1] Chen, S. D. (2014). A Crude Analysis of Twitch Plays Pokémon, 1–6. Retrieved from http://arxiv.org/abs/1408.4925

[2] Aleta, A., & Moreno, Y. (2019). The dynamics of collective social behavior in a crowd controlled game. *EPJ Data Science*, *8*(1), 1–16. https://doi.org/10.1140/epjds/s13688-019-0200-1

[3] Ramirez, D., Saucerman, J., & Dietmeier, J. (2014). Twitch Plays Pokémon: A Case Study in Big G Games. *Proc. DiGRA*, 3--12.