SUPER MARIO BROS.

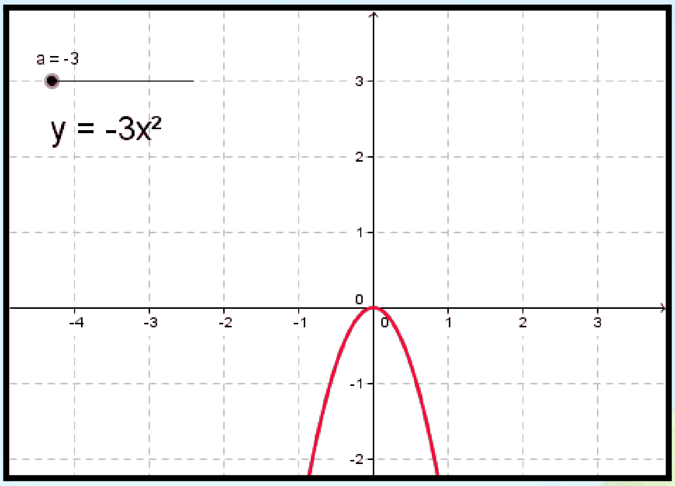
JUMP: CHALLENGE ANALYSIS

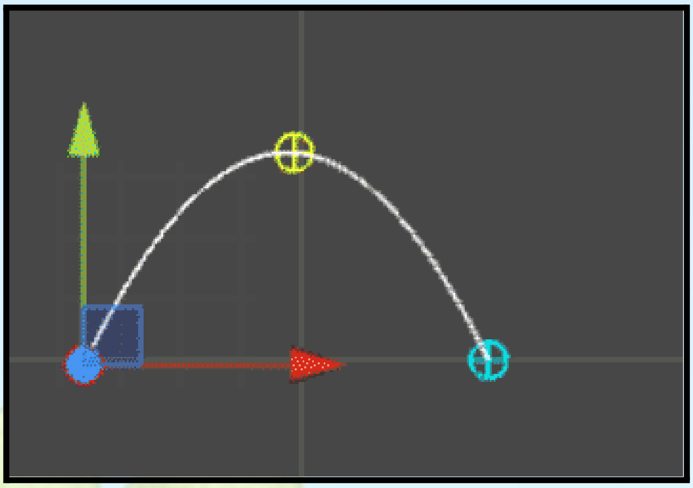
Jump in Super Mario Bros is a **versatile game verb** that the player uses to **progress through levels** by overcoming platforms and obstacles, kill enemies, obtain coins and power-ups from cubes placed in the air and **interact with the main game systems**.
The jump adds a **further dimension to Mario's movement** by exploiting **verticality** to generate additional challenges for the player.

Challenge Evolution

The **evolution of the jump challenge** depends on factors such as the **response** of the inputs (***time in which the input button is pressed***), the relationship between **verticality** obtained based on the **horizontal** speed, the **timing of execution** of the action and the **context** of level design. The last one mentioned deals with varying the other aspects by regulating the **frequency** of the action (*by obstacles/enemies number and position*) and the **precision** necessary to carry it out correctly.

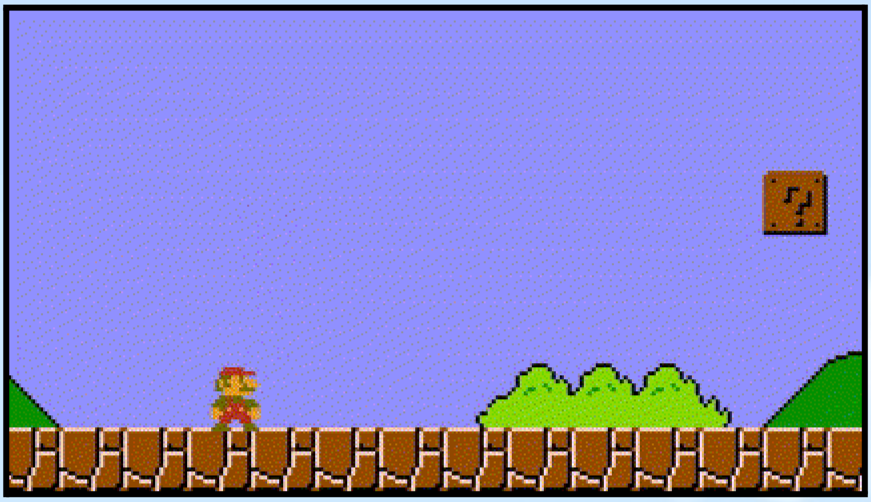
The resultant of Mario's jump is a $y = ax^2 + c$ type **parabola** with an amplitude coefficient ***a*** directly proportional to the **horizontal speed** of the player and a factor **c** which takes care of **translating** the focal point of the parabola forward with respect to the player orientation.



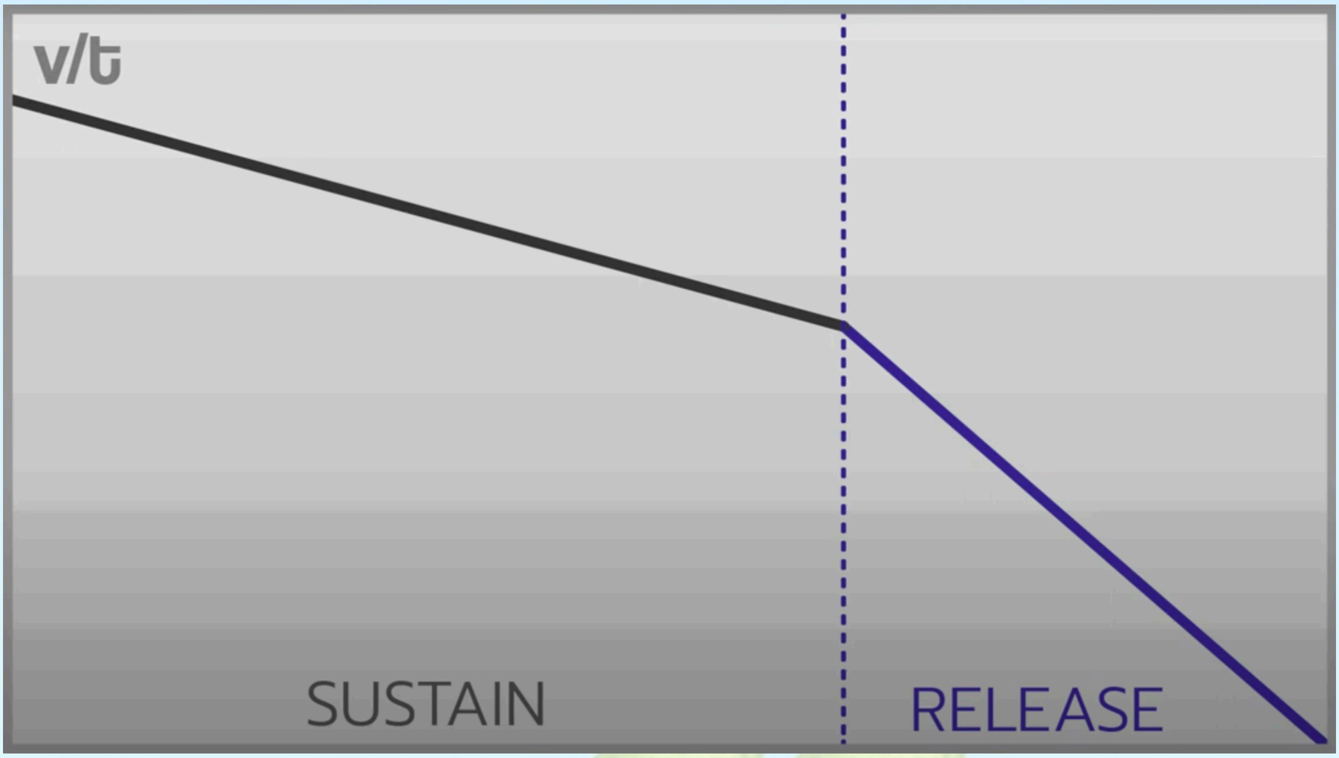


Input Response

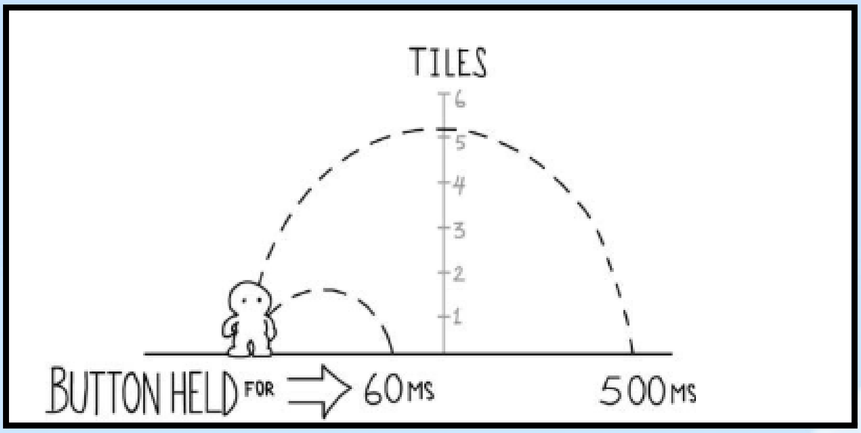
The **response** to the input pressed by the player is (*in addition to a change of sprite set for the entire duration of the jump*) a **vertical movement** that varies based on the **time in which the button was pressed** and the current **horizontal speed** of Mario.



Mario's jump **occurs immediately** and, unlike movement, **once maximum speed is reached** he will move on to the **Release phase** *(decrease in vertical speed due to the gravitational force imposed in the game*). Gravity is an aspect that is varied in the levels of Super Mario, varying the **challenge difficulty** imposed by the mechanics. The gravity is varied by the challenge itself, in fact during the **Sustain time** the force that is opposed to the movement is **less than the gravity itself**, increasing the **player's agency on the control of Mario's horizontal movement in the descending phase.**

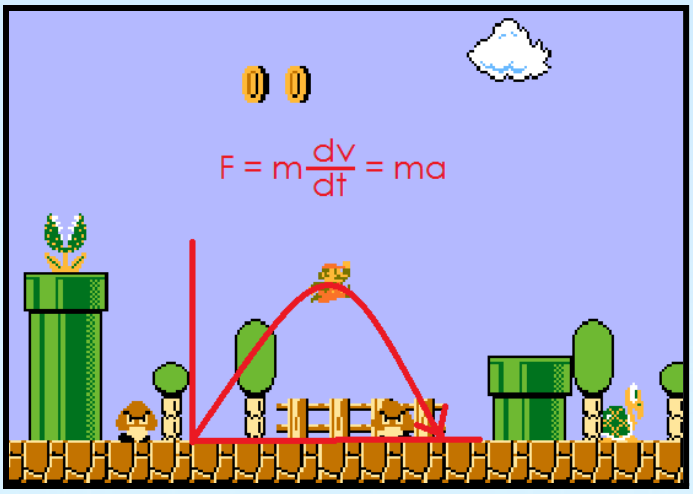


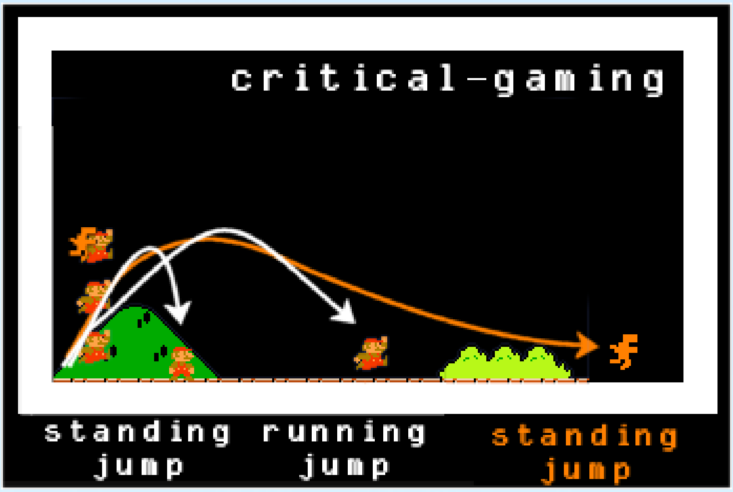
The addition of a **physical force** makes the jump an **impulsive vertical magnitude** only initially assigned to Mario (*thus considering **Mario as a Projectile** and assigning him a **variable range** via the character's horizontal movement*). The player can hold down the jump input to **increase the magnitude of the direction vector of the initial impulse**, in order to make Mario jump **higher** and land him later to *overcome obstacles* that develop horizontally or **reduce the time input** to make *shorter but more precise* jumps or to start landing in the release phase earlier.



Vertical and Horizontal velocity relation

One of the aspects that most **changes the level of mastery required of the player** as the gameplay progresses is the **direct relationship between horizontal speed and vertical speed for Mario's jump**. The magnitude of the vector for the vertical impulse induced in Mario by pressing the button i**ncreases as Mario's current speed increases,** creating, for the purposes of an analysis based on Rational Game Design, an **ingredient (Movement +Jump)** useful for establishing the development patterns of a level.

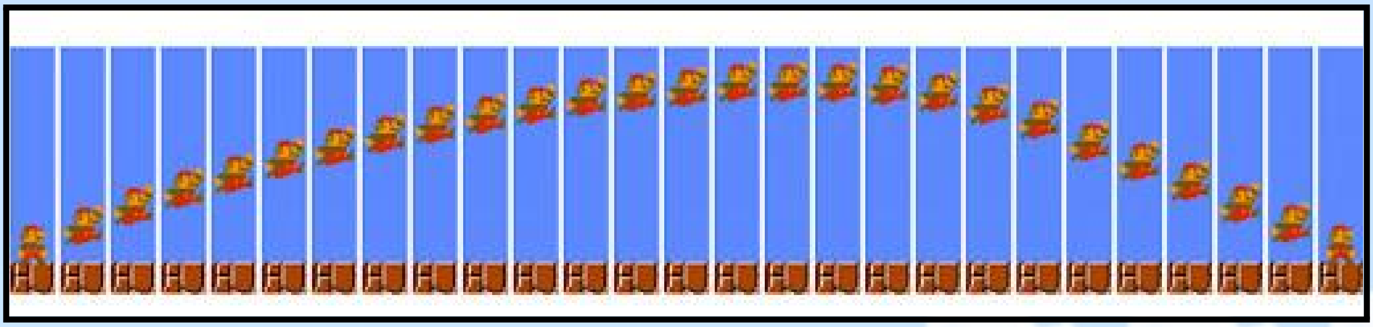




Execution Timing

The **execution timing** of a jump is a fundamental **Physical Skill** for the player in order to *overcome an obstacle, reach a platform or defeat an enemy.* The two fundamental aspects regarding the timing for the execution of the jump are:

- **Distance to travel**: varies the difficulty by placing the player in a condition of **reasoning** both on the **time to press** the jump button, on the **horizontal speed** to maintain for that jump, and on the **time** in which to **keep the button pressed** while it's in the air.
- **Landing Space:** The landing space also determines the **maximum distance** from which the player **can start the jump** to correctly overcome the obstacle. The more landing space there is, the **less precise** the player will have to be when releasing the button.
- **Movement of obstacle/enemy:** The movement of the obstacle/enemy to overcome forces the player to predict and jump at the right time.



Level Design Context

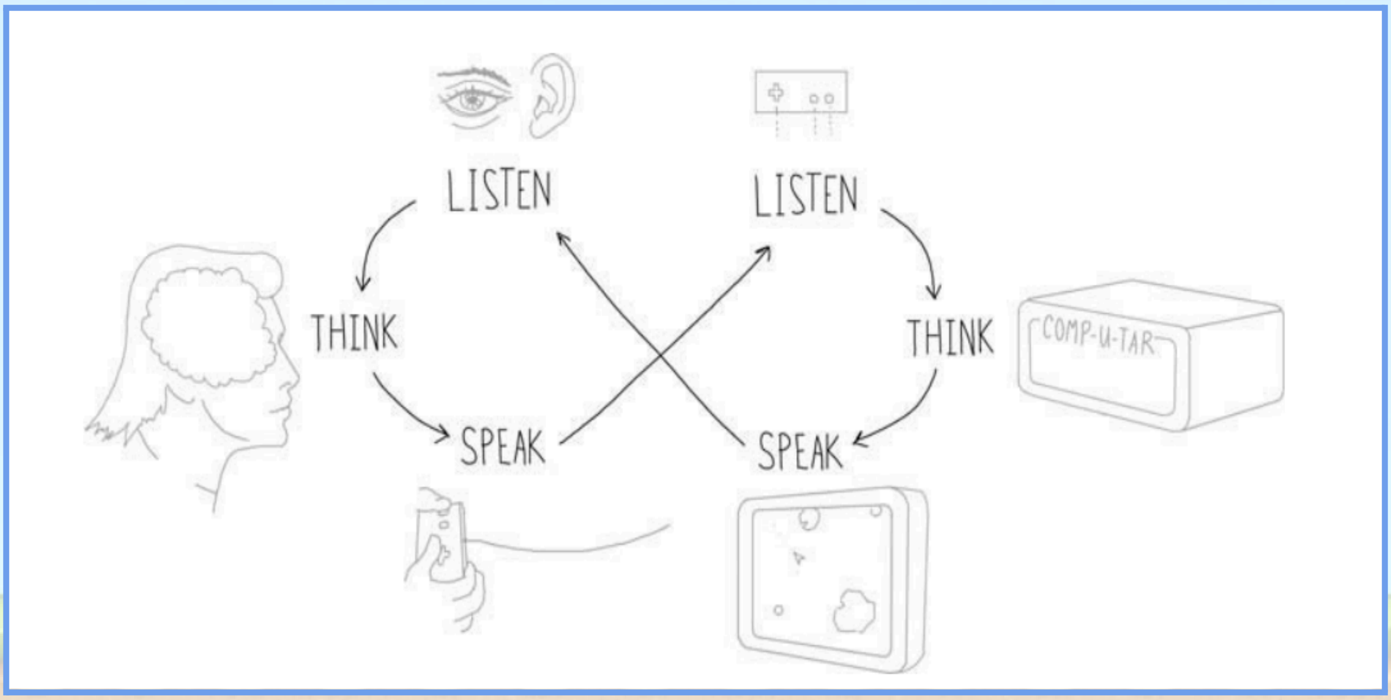
The single level has **challenges** for the player to face. The **duration, frequency and individual characteristics** of these challenges offer an evolution of the mechanics during the course of the gameplay by **varying atomic parameters and execution timing required**.

Specific levels vary the gravity (underwater levels), significantly increasing the character's descent time and slightly increasing the vertical impulse, providing another way to exploit the jump.

Win & Fail Conditions

The Win and Fail conditions for Jump in Super Mario Bros are defined by Boolean values that establish the outcome of the action carried out by the player:

- **Vertical speed**: if the player manages to move vertically, then the outcome of the jump is positive.
- **Player death**: if after completing the jump the player hits an enemy sideways or hits an obstacle that causes the loss of a life, the outcome is negative.
- **Reaching the platform**: if the final position of the character is higher than the starting position of the jump, then the outcome is positive.



Physical / Mental / Social Skills

*(Same as **Movement: Challenge Definition** **Skill paragraph**, I also reported it here to keep the reasoning developed for the development of weights and values always visible in all documents regarding the definitions of the challenges)*

The **numerical definition** of the difficulty of the overall skill of the jump consists of a **weighted mathematical average** of the values of the physical, mental and social skills required of the player to carry out the challenge. The weights and values for the analysis are **arbitrarily established** after determining **value scales** with the relative meanings from which to **extract the final data**.

What does the player have to do?

The player must use the **jump** to continue the level in order to **finish it before the timer expires**, to obtain a **better score** by defeating enemies and to **avoid** fixed or mobile **obstacles** during gameplay progression.

Equation Definition

$$Difficulty = \frac{PW(P) + MW(M) + SW(S)}{3}$$

Where:

$$P = \frac{\sum_{1}^{n} (P\ Interaction\ Value)}{n} \quad \Bigg| \quad M = \frac{\sum_{1}^{n} (M\ Interaction\ Value)}{n} \quad \Bigg| \quad S = \frac{\sum_{1}^{n} (S\ Interaction\ Value)}{n}$$

are the arithmetic averages of the difficulty values of each skill type (**P**: Physical, **M**: Mental, **S**: Social) The calculated average is weighted based on the weights assigned to each skill type (**PW, MW, SW**).

In order to have a **mathematically coherent analysis**, the objective is to give the categories **weights whose sum is equal to 1**. This premise allows the values to be **scaled correctly** and to have data that represent a percentage of impact of the skills in evaluating the difficulty of mechanics.The calculated average is **weighted based on the weights assigned to each skill type.**

Skill types weights Scale

To define the weights and values of the different types of Skills, the **number of interactions** that the player must perform which concern that category and their **frequency** are taken into consideration. The definition criteria are listed below and examples of assignment for the values are shown:

Weight Value	Number of interactions	Interactions Frequency
0	**0**	**Low/Zero**
0.25	**<2**	**Low**
0.5	**<4**	**Medium**
0.75	**<6**	**High**
1	**>=7**	**Very High**

Physical Skills

Single interaction difficulty value

The **difficulty values for individual physical interactions** are established based on the **movements** made by the player to carry out the challenge and the **duration** of the interaction. The difficulty is **directly proportional to the quantity of movements** and **inversely proportional to the duration**, the more movements the player makes the more the difficulty increases. On the other hand, **the slower the interaction (longer duration) the more the difficulty decreases.**

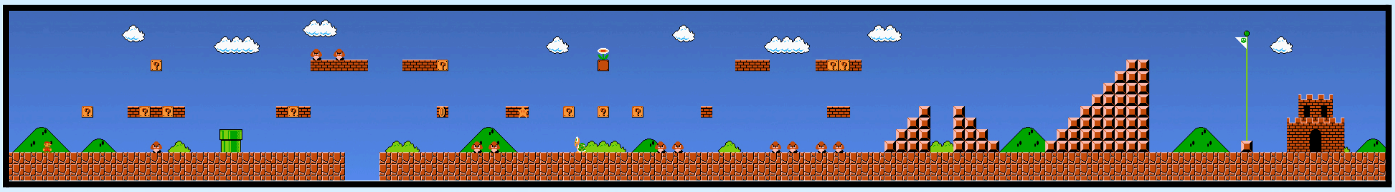
Difficulty Value	Interaction Movements	Interaction Duration Time
1	**0**	**High**
2	**1**	**Medium**
3	**>1**	**Low**

Skill Difficulty final value

Physical skills have equal weight to mental skills because they are those required to perform the jump through the pressure of the input. The interactions that are performed in this category are:

Skill	Description	Difficulty Value
Button Precision	The player must press the right/left button correctly. The jump is combined with the movement, (2 buttons are required)	**2**
Pressed and Release Time	The player must press the button for the correct duration in order to move Mario to the desired final position	**3**
Timing	The player must press the button at the right time to avoid losing lives, reach a platform or hit enemies	**2**
Reflex	The player must have ready reflexes to respond to a danger/trap at the right time	**3**

$$P = \frac{\sum_{1}^{n} (P\ Interaction\ Value)}{n} = \frac{2+3+2+3}{4} = 2.5$$

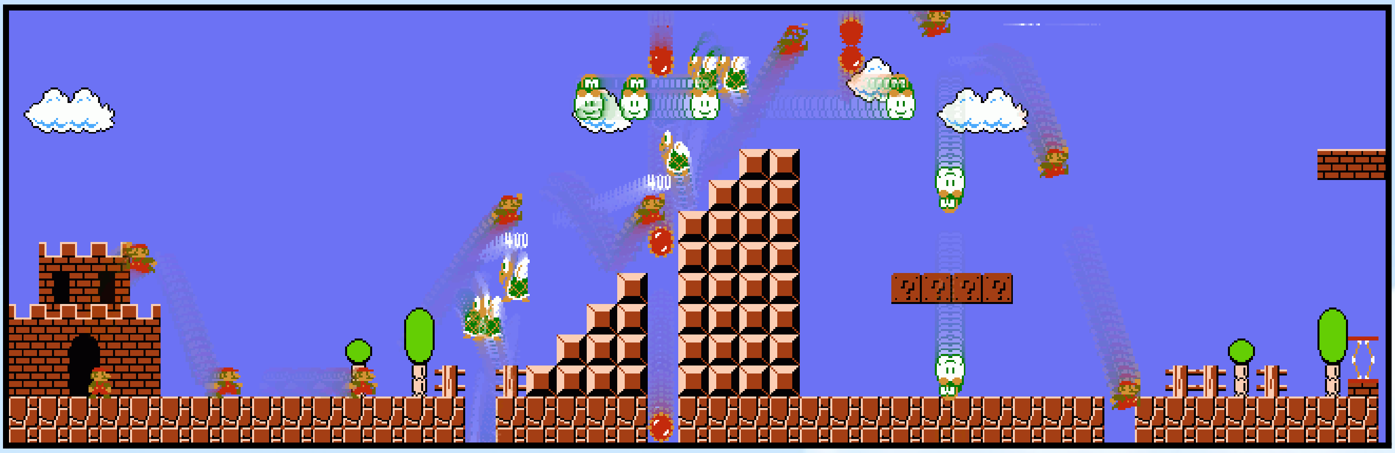


Mental Skills

Mental skills present calculation, observation and reasoning challenges for the player. **Their values vary depending on the type of pattern the player executes in order to overcome obstacles**. By removing this aspect from the calculation, the player's reference point becomes the **distance to calculate or the movement pattern of an enemy/trap to predict**.

Skill	Description	Difficulty Value
Calculation	The player must calculate the distance to travel to overcome the obstacle or to hit an enemy at the right time	**7**
Predict	The player must predict enemies actions or platform movement/position to overcome them	**6**
Global Vision	The player must observe everything that is shown on the screen to think about the sequence of jumps to make to progress through the level	**6**

$$M = \frac{\sum_{1}^{n} (M\ Interaction\ Value)}{n} = \frac{7+6+6}{3} = 6.33$$



Social Skills

There are **no social skills** for jump mechanics. It is a basic game interaction and is used for purely systemic purposes, allowing the player to progress within the game world or to reach a greater score.

Movement: Final Skill Difficulty

Before calculating the **final value of the Difficulty of the Super Mario Bros Jump system**, the weights must be assigned to the different types of Skills.

Skill Types Weights		
Physical Skills	**Mental Skills**	**Social Skills**
0.5	0.5	0

$$Difficulty = \frac{PW(P) + MW(M) + SW(S)}{3} = \frac{0.5(2.5) + 0.5(6.33) + 0(0)}{3} = 1.415 \simeq 1.5$$