# MOVEMENT: CHALLENGE ANALYSIS

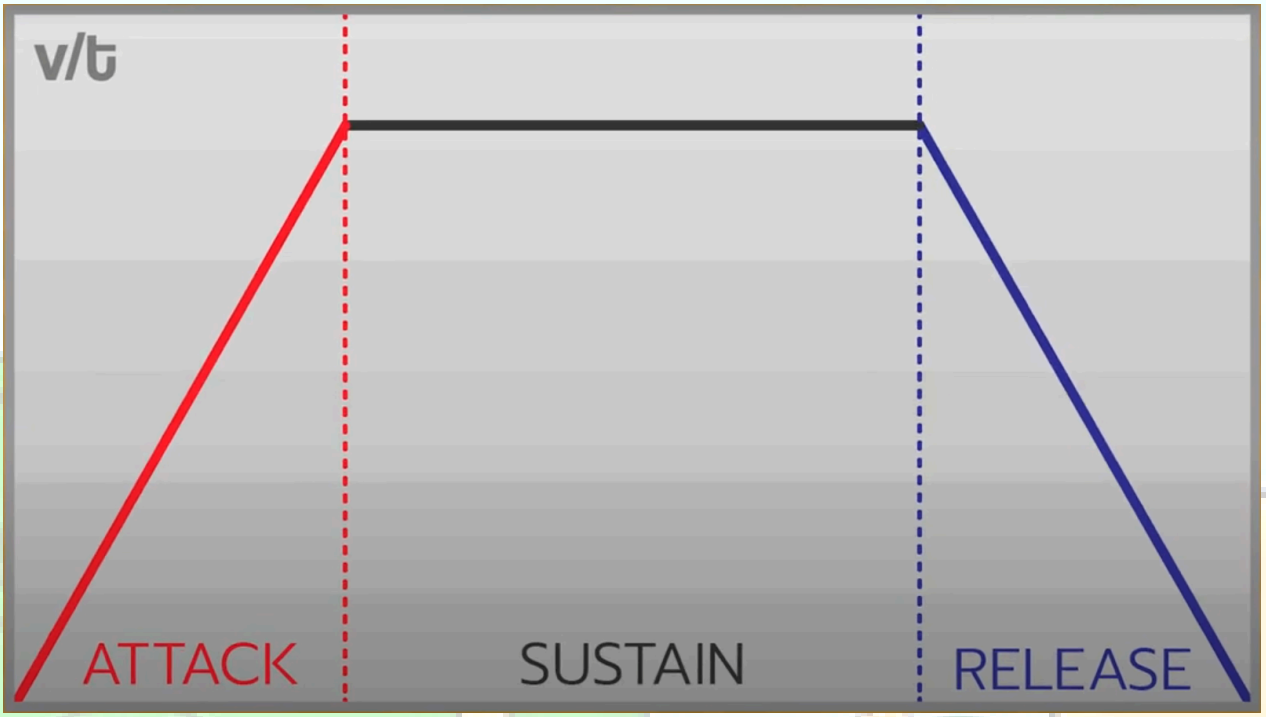
Movement in Super Mario Bros is a challenge that **the player faces for the entire duration of the gameplay.** It is used for **progression** in each level and allows the player to *advance, overcome obstacles and defeat enemies*. Together with the **jump** and the **dash** it represents the **totality of the player's agency regarding inputs and controls**.

## Challenge Evolution

The evolution of the movement challenge depends on the **input response**, the **inertia generated**, the **execution timing**, the rate of the atomic parameters, and the context imposed by the Level Design (obstacles, enemies, threats).

*Input Response*

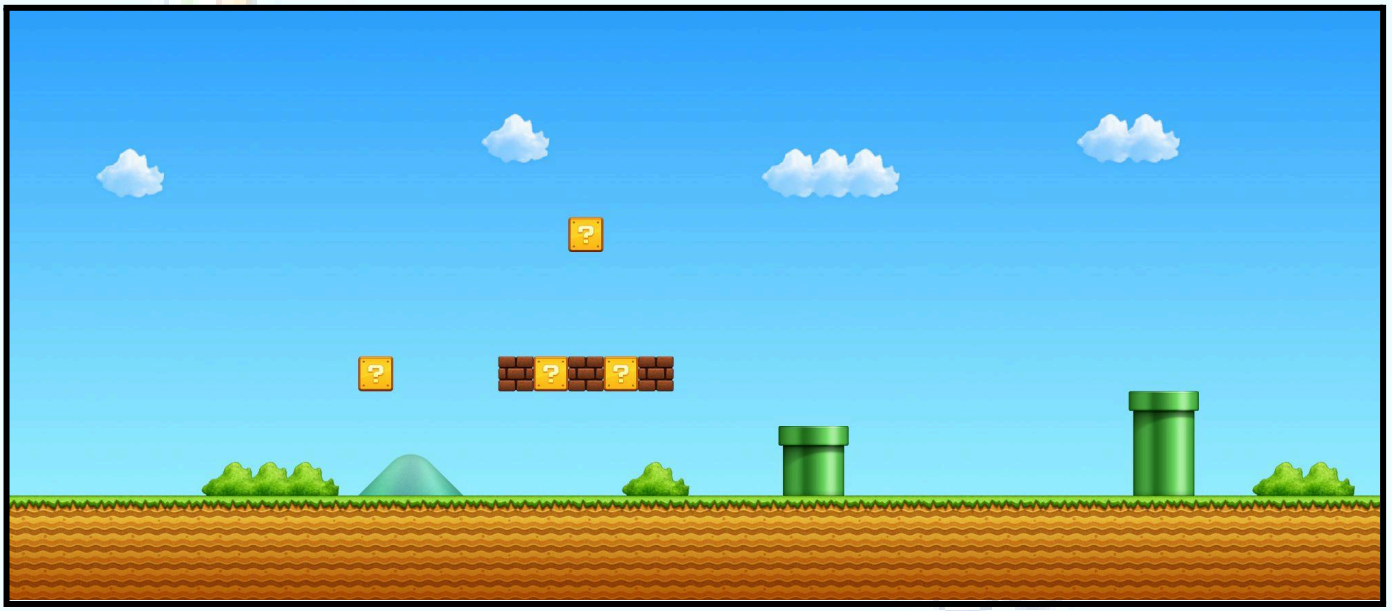
One of the main points of interest in the movement of Super Mario Bros is the **input response**, or the **character's reaction when the action input is started**. In platform games, the action of movement and the response to inputs is structured following a function that determines the **speed of the character depending on the time in which the button is pressed** (*consequence of the definition of a linear or **non-linear increment***), the **maximum speed** that the character can reach and the **time** in which the speed **drops to zero** after the input is released (*a **linear or non-linear** decrement must be defined*).




*Fonte: Game feel*

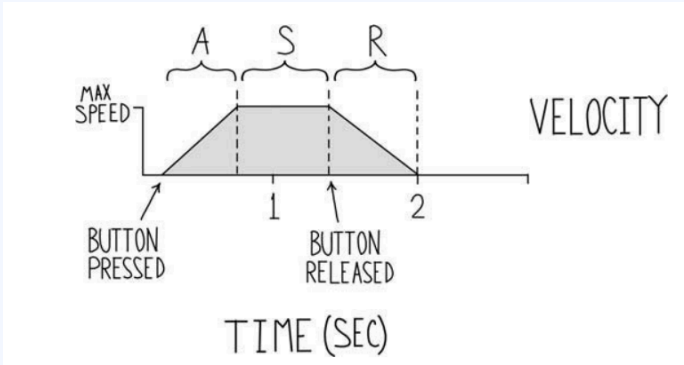
The function representing the **input response** is the **ratio between the speed of the character in relation to the time the key is pressed.** The **time** and the **increase/decrease in speed** can have different variations in the *Attack phase* (**increase in speed**) and *Release* (**decrease**), but in the *Sustain phase* (**maximum speed reached**) the derivative of the function must remain **0**, consequently **no speed variations** must occur.

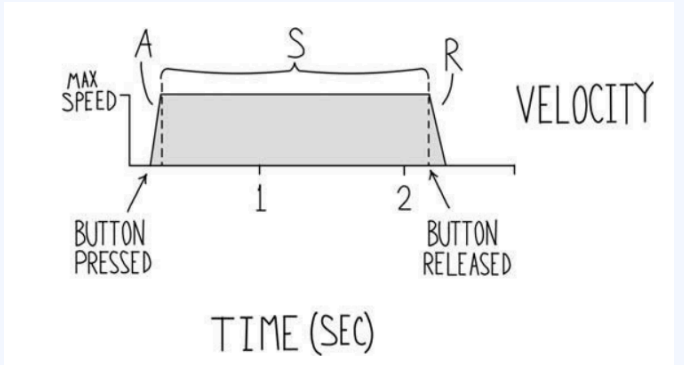
Super Mario Bros has **identical and linear Attack and Release Rates** (*as shown in the figure*). As a result, the derivative of the function in both phases remains the same with **respect to the magnitude** (*with signs reversed*). This feature allows the player to have greater control over the character by **decreasing the inertia suffered by the movement when stopping or changing direction.**



*Inertia Generated*

A movement system produced following the described function aims to **generate a sense of inertia** with the **aim of simulating a realistic movement but which still maintains a fluidity of movement suitable for being easy for the player to learn.**
The movement is **not interrupted** as soon as the player releases the button, this allows him to carry out actions while keeping the movement active for a specific period of time (***10 frames for Super Mario Bros***).



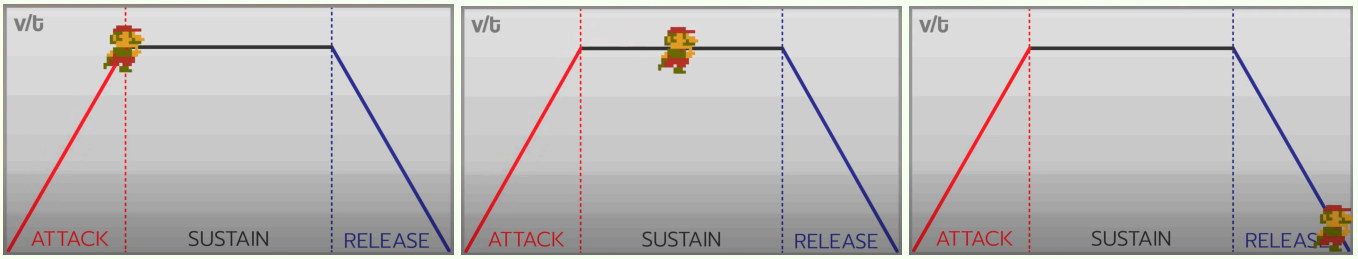

Based on the **context** of the level being played, the **values** corresponding to the **Attack and Release phases are varied in subsequent levels**, influencing the evolution of the challenge during the gameplay.

*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 Jump induced in Mario by pressing the button **increases as Mario's current speed increases**, creating, for the purposes of an analysis based on Rational Game Design, a fundamental ingredient **(Movement +Jump)** for establishing the development patterns.

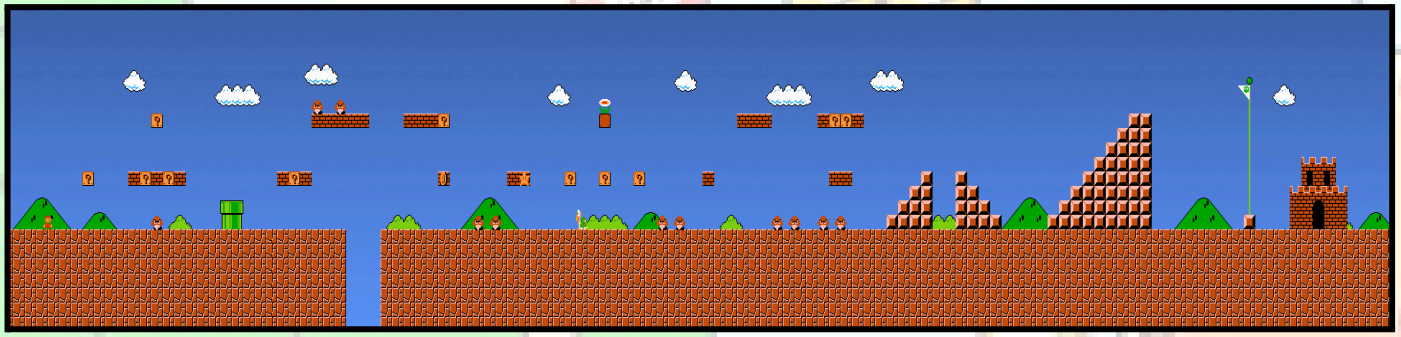
*Execution Timing*

The **timing** of the use of movement is made a skill to be mastered by the **presence of obstacles and dangers** in the Super Mario Bros. levels. The player must **calculate the moment in which to move to avoid being hit or falling into a trap.**




A fundamental element for deciding the **timing of movement** is **Mario's speed variation** during the course of the challenge. Once the movement enters the **Sustain phase**, the character's control decreases due to **reaching maximum speed**. This dynamic increases the difficulty of **calculation** and **prediction** regarding the **distance traveled once the movement is interrupted.**

The input release time is influenced by a **delay of 10 frames** which is inserted to allow the player to **maintain maximum speed while performing a jump** (*or shooting fireballs in motion*) and then resume movement from the Sustain phase.

*Level Design Context*

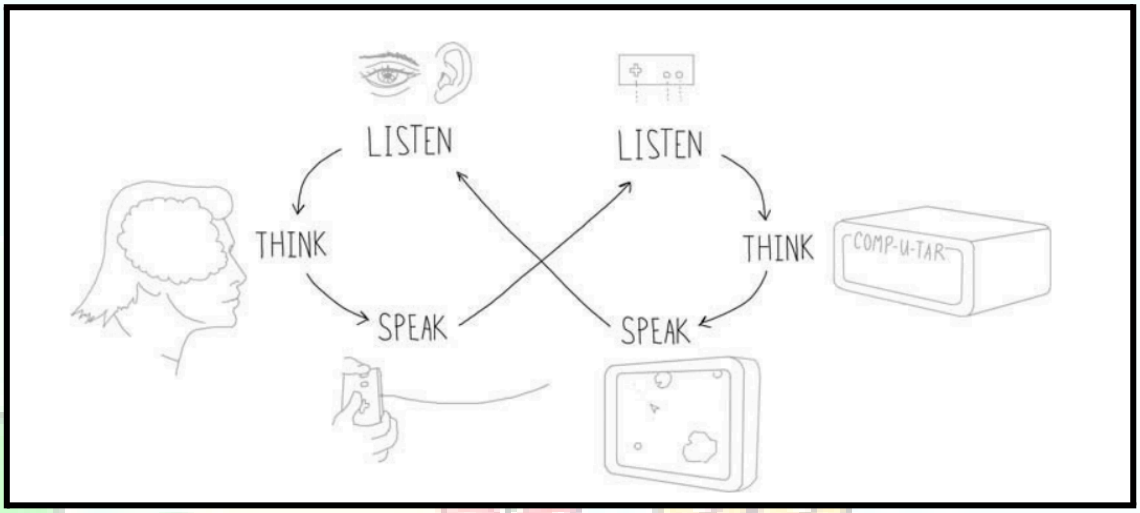
The context inserts a variable into the **calculation of the actual difficulty of the movement challenge**. The **times and Rest Spaces** (*spaces without a pattern to follow, analyzed in* ***Patterns***) can make carrying out the movement more difficult for the player (**requiring greater mastery**).
The position and type of obstacles and dangers present in the level can completely change the mental calculations that are made by the player **regarding the timing of execution of the movement**. Even the **landing spaces after a jump** affect the player's reaction time regarding movement, this consideration is due to the **presence of the inertia of the movement which continues to make Mario continue even after the release of the input.**

## Win & Fail Conditions

The **Win and Fail conditions** for movement in Super Mario Bros are defined by **two Boolean variables**, i.e. whether after carrying out the movement **the character is unharmed or not** and whether the movement actually **caused the character to move.**

The obstacles that make up the platforming of Super Mario Bros are elements that aim to offer the player **challenges to overcome in order to progress through the level**. Enemies hit Mario **inflicting a life** of damage when they come into contact with him, bottomless holes and pools of water/lava remove a life from the player when he falls into them and traps inflict one damage to the protagonist if he is above/under them at the time of their activation.

Walls, brick blocks and block barriers, on the other hand, **prevent the player from continuing through the level with the sole use of the movement mechanics.**
If at least one of the variables is true (*or the character suffered damage during the movement action or was unable to carry out any movement* - **Fail Condition**) then the action was **successful** (**Win condition**).




## Physical / Mental / Social Skills

The **numerical definition of the difficulty of the overall skill of the movement** 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 establishing value scales with the relative meanings from which to extract the final data.

*What does the player have to do?*

The player must **move within the game levels**. He must overcome obstacles, move on moving and rotating platforms, avoid enemy attacks and try to reach the end of the level within the set time (game timer) without losing his available lives.

*Equation Definition*

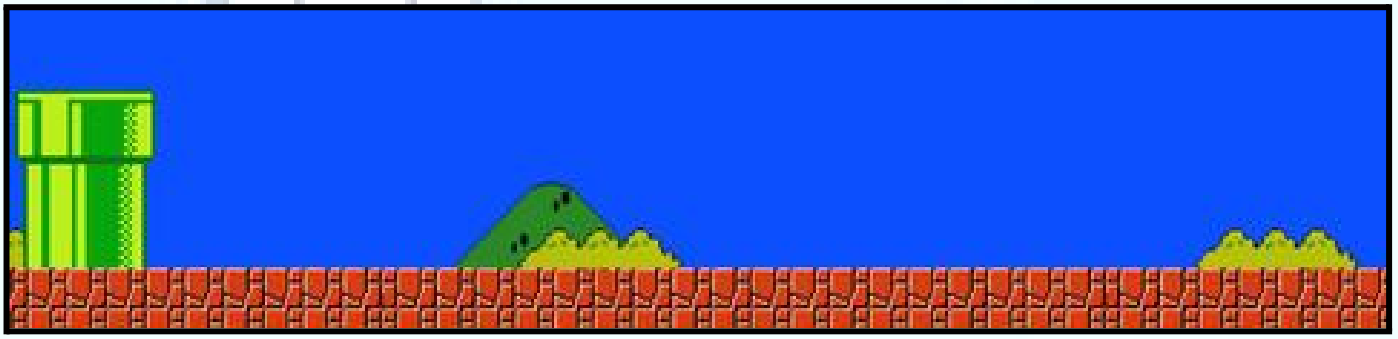
$$Difficulty = \frac{PW(P) + MW(M) + SW(S)}{3}$$

Where:

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

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 a **greater weight because they are those required to perform the movement 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 | **1** |
| **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 | **2** |
| **Reflex** | The player must have ready reflexes to respond to a danger/trap at the right time | **3** |

$$P = \frac{\sum_{1}^{n} (P\ Interaction\ Value)}{n} = \frac{1+3+2+3}{4} = 2.25$$

*Mental Skills*

Mental skills present **calculation, observation and reasoning challenges for the player**. Their values vary depending on the **type of pattern he is about to execute 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 | **5** |
| **Predict** | The player must predict enemies actions to overcome them | **6** |

$$M = \frac{\sum_{1}^{n} (M\ Interaction\ Value)}{n} = \frac{5+6}{2} = 5.5$$

*Social Skills*

There are **no social skills for movement mechanics**. It is the basic game interaction and is used for purely systemic purposes, allowing the player to progress within the game world.

## Movement: Final Skill Difficulty

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

| Skill Types Weights | | |
|---|---|---|
| **Physical Skills** | **Mental Skills** | **Social Skills** |
| 0.6 | 0.4 | 0 |

$$Difficulty = \frac{PW(P) + MW(M) + SW(S)}{3} = \frac{0.6(2.25) + 0.4(5.5) + 0(0)}{3} = 1.183 \simeq 1$$