Tower of Babylon Research & Analysis

Hammerfight (PC)

First, we will start with an indie game: Hammerfight, which ties the movements of the hero's weapon to the movements of the player's mouse. As player wave the mouse, the hero swings his weapon, which is basically a metal chain with a ball or an axe attached to the end. In my game, Tower of Babylon, the rotation of the hero's hammer is tied to the rotation of the joystick. Differences between these two mechanisms comes from: 1. Different input devices; 2. Movement control of the hero himself; 3. Whether the weapon is a rigid body and how it interacts with other physical layers.

Circularly waving the mouse is definitely harder than rotating the joystick, but the input data from mouse is more precise than joystick. Most of the negative reviews of Hammerfight complains that it can makes your wrists really tired, and the movement control is confusing when you haven't get used to it. However, this also means players can control the movement more precisely, and get more sense of achievements from mastering it. In my game, no matter how quick the player rotates the joystick, as long as they reach the required degree in certain time, the hammer will start rotating in the same speed. This makes the player worries less about weapon movement, and focus more on the platformer part. In contrast, Hammerfight has a realistic physics simulation system that enables the weapon to react to every movement of your mouse. Its map is mostly empty, the amount of enemies is less than platformer games, and the AI of the enemies is more complicated. Therefore the main concern of the player is how to move the weapon to defeat the enemies.

In Hammerfight, movements of the hero is controlled by the mouse movement (left/right/up/down) as well, which provides a continuous experience of pure mouse control, but requires a large space on your desktop when combining with the weapon movements. In my game, left/right is controlled by a different joystick, but jumping is controlled by the weapon (the hammer is designed to have jet power) and falling is controlled by the gravity. Separating up/down with left/right is more platformer styled, adding difficulties to the movements of the hero.

The weapons in Hammerfight contains chains instead of rods, which brings more randomness, continue adding difficulties to the weapon movement control. This also slows down the speed for the weapons to react to the mouse movement, giving more time for player to adjust their mouse movement and bring fault tolerance. In my game, it is better that the weapon reacts to the input quickly and do exactly what the player wants it to do, since it is crucial to dodge and attack enemies at the right moment.

Nebulus (aka Tower Toppler) (C64)

At first glance, the square-shaped tower and the four-sided camera movement (which I have not implement) of my game may looks more like Fez. However, its mechanism is closer to Nebulus since the camera direction does not affect whether the hero can move to the next block.

Both Nebulus and my game creates a 3D world around a tower and only allows a 2D movement, but there are several differences. First, the regular platforms are longer in my game, aiming to reduce the number of jumps, since the jump control is much harder than the horizontal movement. Tower climbing requires the player to move carefully, since failing to land on the desired platform will waste a lot of previous effort. Nebulus force the player to be careful through setting a lot of short and connected platforms. My game will have some extremely long platforms that cover the entire side of the tower. These platforms will be placed in a spiral style, hence as long as the player moves to the next side, they will have a higher starting position and never falls below it. Second, the enemy movements are simple in Nebulus because its core is puzzle solving. My game will focus more on actions like dodging, attacking, and precise movements, results in placing ‘smarter’ enemies and less tricky switches and doors. Third, Nebulus has a round-shaped tower, providing an impressive visual effects and a mysterious atmosphere. My tower is square-shaped, aiming to imitate the world created in the story *Tower of Babylon* and provide an atmosphere of science fiction.

Metroid (NES)

As a classic platformer game, Metroid did well on informing the player and reasoning its design choice through the appearance of the hero. The green part on Samus' arm tells that the ways of attacking is shooting. Samus' magical armor gives reason to its air control design to some extent. In a realistic physical world, it is not reasonable that one can control its horizontal motion when it is not on the ground. However, enabling air control did make the player less frustrating, and help complete more flexible movement for the tricky platform puzzles. The cool armor of Samus create an atmosphere of science fiction, boosting the player's imagination: it may have some propelling device to help air control.

In my game, the hammer of the hero indicates his attacking rate: it will not be as quick as a gun. This indicates that the player needs to control the horizontal movement more precisely, and hit the enemy at right moment. Combing with the tower-shaped map, we can also infer that there will be less enemy, but more movement control requirements. To reason the air control, I am going to add animation of hammer blowing air from its left/right when player makes horizontal movement in the air. This also enables a more flexible movement control for run jump comparing to Metroid. In my game, run jump will be the same as static jump, which means player can stop the horizontal movement. I feel the design of run jump in Metroid adds difficulties when Samus is in the vertical-shaped map. Since my game requires more vertical movement and the jump operation is more complicated, I decide to give the player less constraint on the movement control.