University of Calgary
CPSC 585 — Winter 2019 — Games Programming

Hover Wars

Final Product Design Document

Team A — Pressurized Studios
Austin Easton, Evan Quan, James Coté, Jianan Ding
April 12, 2019

Contents

1 Introduction

Over the course of the last couple of months, there have been a few design decision changes. This document serves to highlight and discuss them.

2 Gameplay

2.1 Game Modes

The central game mode we propose to implement is **free-for-all**. At a lower priority, other modes may be developed if enough time is available.

2.1.1 Free-for-all

Each game is composed of a single round that lasts for the duration of a timer. Each player is placed in an arena to control a hovercraft with various abilities. They must fight each other in a free-for-all battle with those abilities, amidst a number of AI-controlled hovercrafts roaming the arena to hunt down players.

The central goal of the game is to gain the highest score possible before the round is over. Players have the following means to gain score:

- Attacking other player hovercrafts, which can only be done in multiplayer. This can be through the use of abilities accessible to every player.
- Attacking AI-controller hovercrafts (bots), which can be done in both single and multiplayer. Bot hovercrafts do not have all the abilities of player hovercrafts and so reward less points.
- Picking up power-ups. These help the player gain points by improving their abilities, but also innately give points when they are picked up.

2.2 Arenas/Maps

Primary Environmental Features

These are basic features that are core to the map's design, and so play a high priority in their implementation.

- **Barriers/obstacles** — These can block movement and projectiles. These can take the form of various things: street lamps, walls, buildings, tunnels, or other structures.
- **Ramps** — Ramps lead to higher and lower platforms, or can be jumped off of over obstacles. This can provide better vision of certain parts of the map, give opportunities to jump between arenas.

Tentative Environmental Features

These features are open to be cut, either due to time constraints to implement or potential game design problems they introduce.

- **Speed pads** — Driving over these will give a momentary boost of speed. Designed for getting power-ups quicker, and chasing or escaping other players.

  Potential problem: While these make sense for a racing game, they may detract from the gameplay for a combat arena game. Players will typically want full control over their vehicle and driving over a speed bump in locations where they don't want it can be annoying. If a temporary speed boost mechanic is to be implemented, it may be better to have a speed boost power-up instead of speed pads.

- **Pits** — Falling into the pit will instantly destroy the player hovercraft. Avoid at all costs or try to bump enemies into it.

  Potential problem: Depending on the placement, size and frequency of the pit, players can be easily frustrated if they are constantly falling in it. Instant death mechanics such as this can also break the flow of gameplay.

- **Jump pads** — Driving over these, will give a vertical boost while maintaining horizontal momentum. Designed for jumping over obstacles or reaching areas of higher elevation.

  Potential problem: Similar to speed pads, players may not actually want to use them when traversing across the map. It can potentially be very disorienting to players to use jump pads, especially if they lose control of their movement while in the air. Ramps can serve the same purpose without making the player lose control. Anohter possible solution would be to allow players to influence their trajectory while they are in the air.

It is important to note that more features doesn't always mean a better map. Even if the idea of certain map features are cool or interesting on paper, if they detract from the overall gameplay experience in practice, it is better to leave them out. There is value in a good but simple map design.

2.3 Hit Points, Lives, and Damage

There are a few hit point and damage systems we have considered. We are currently committed to implement the **hit and continue** system, but will be open to change as we play-test.

2.3.1 Hit and Continue

Modelled after **Mario Kart**'s damage system, hovercrafts are blown away or spun around when they are hit by abilities, temporarily making the player or bot that was hit lose control of their vehicle. Shortly afterwards, control is regained and the hovercraft is given a few seconds of invincibility before continuing on playing. This minimizes the downtime during play since there is no respawning, while still punishing players that get hit. It also avoids the disorienting effect that respawning can have, especially if the map is not familiar to the player.

2.3.2 Lives with Hit Points

Players start the game alive with a small set amount of hit points, which will be explicitly displayed to the player. They can be damaged by the abilities of other hovercrafts, lowering their current hit points. The same temporary loss of control will occur as the hit and continue system. When all hit points are removed, the player's hovercraft is destroyed.

When a player's hovercraft is destroyed, they are momentarily out of the game before respawning randomly at one of the respawn points on the map. If another player destroyed said hovercraft, that player is awarded points for the kill.

Under this system, bots could have fewer hit points than players.

2.3.3 One Hit One Kill

Similar to the lives and hit points system, but players are instantly destroyed upon impact of an ability. This makes getting hit more punishing and further promotes players to dodge abilities.

2.4 Players

2.4.1 Movement

- **Acceleration/braking** — This was deemed unnecessary as we have tuned the driving model so players can very easily stop and change directions at will.
- **Dashing** — Dash charges were implemented so that players could consecuatively dash to their target to use spikes. Not only did this feel better than without the charge system, but it also improved the spike ability's usefulness.

2.4.2 Abilities

- **Rocket** — Initially we had a very short cooldown for the rocket.
- **Spikes** —
- **Trail** —

2.5 Bots

2.5.1 Movement

2.5.2 Abilities

More than promised, bots are able to use all abilities.

1. **Rockets** —
2. **Dashing** — Bots would be able to dash out of the way to dodge abilities.

2.6 Power-Ups

2.6.1 Spawning

2.7 Menu

3 Game Design

3.1 Aesthetic

We finally decided to follow an a 1980-theme retrofuturistic art and music style commonly referred to as Outrun.

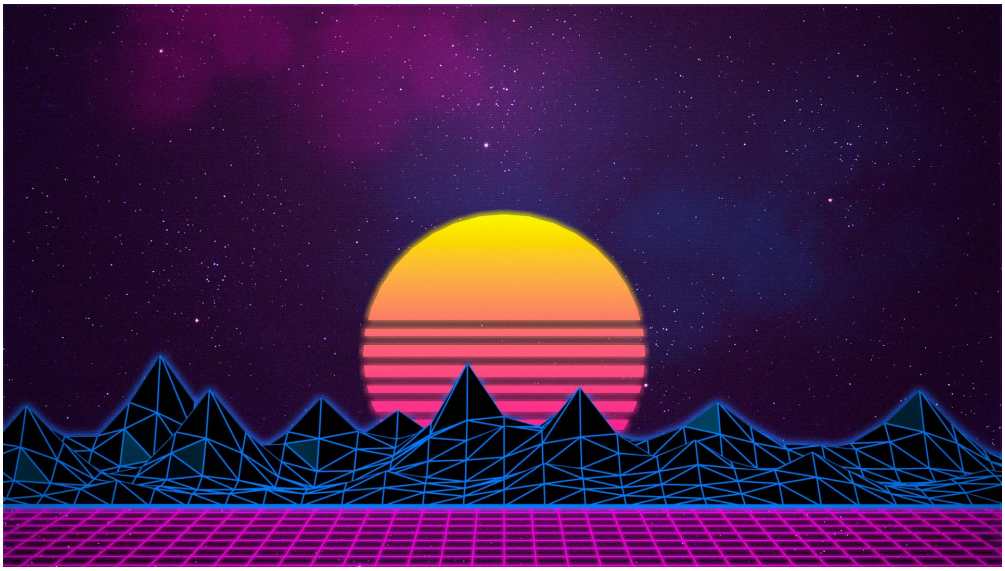

Figure 1: Wireframe appearance. Bright neon colours contrast heavy use of black.

3.2 Designer Insight/Goals

Looking back at our designer goals in our intial high-level design document, this is how we followed through:

Figure 2:

3.2.1 Role of AI

The introduction of AI-controlled hovercrafts (bots) adds an interesting element to the game, but also a few problems.

First, given our past experience and the time-frame creating the AI, we don't believe the bots will be equally competent to a skilled human player. If bots are given a hovercraft with equal capability to that of a player, it is unlikely they will be able to utilize their abilities and movement sufficiently to compete with players, or have sufficient game sense to outplay and counter different play styles. This poses a problem for single-player, as competing against a group of underperforming bots will not particularly fun or challenging.

To address the challenge issue, it is possible to give bots point bonuses when they score to improve their chances to beat players. However, this does not necessarily address the fun issue, as the player will still experience fighting against simple bots.

Instead, bots can be given an alternative role rather than replacing a player. By explicitly giving them less capabilities than the player, and having them exist in-game independent from the player count, they can add an extra depth to the gameplay without heavily relying on the depth of their capabilities. The benefit is that if the bots end up less capable than we initially planed, the combat can still be just as fun since bots will be in greater numbers and team up against the players. If they are more capable than we initially planned then all the better, since this will simply make the gameplay more engaging.

Overall, the focus of multiplayer will be on the interactions between players for the multiplayer experience, where the bots will provide a background or secondary element to the gameplay. For single player, the bots will be core to the gameplay, as they are the only enemy against the player.

3.2.2 Driving System

3.2.3 Getting Hit

Independent of which damage system we implement (hit and continue, lives with hit points, or one hit one kill), we want to ensure that it is in line with number of our design goals.

- **Emphasize the impact of getting hit —**
- **Ease user memory load —**
- **Easy to Interpret —**

3.2.4 Abilities

In designing our abilities, we have several goals in mind. For each one included, we want to make sure that they:

- **Feel distinct to use:**
- **Serve different purposes:**
- **Introduce some level of counterplay:**
- **Are deliberate in their use:**