

Heart of Metal – Postmortem

05/22/2020

Members: Luyao Zhao, Price Shen, Yonghe Mu, Yanjiao Liu, Wenbo Wu

What went right

In order to make a multiplayer game, we came up with an idea that two people together control the same robot but a different part of it. The player one plays as the shooter part, on the top of the robot, can aim and shoot in a first-person view. The second player plays as the movement part, in a third-person view and can move around, jump, and use the jetpack to fly.

Because of the different views of the two players, they should communicate with each other and tell the other player what is happening in the other's view. That makes a connection to the player you are playing with.

We plan to make player skills, such as *Trigger Missile* that the robot can deal damage to the enemy on the ground. We also made some simple enemies AI system as a backup that may have it in use later. Traps, some interaction with the environment are also considered, together with the item system.

What went wrong

At the very beginning of the idea, we tend to make this game as a coop game, in which the two players cooperate with each other and solve puzzles by using their different views. But after the first game test, we realize that it will be pretty hard to design interesting and playable puzzle levels in that short time. So as the instructions of the testers, we changed it to a PVP game, where four players control two robots and fight in a simple map.

Another problem is that we made the game scale much too big when we started to make it. The internet part, since it's our first time to make a multiplayer game, caused some problems and wasted most of our time. So we have to give up some special mechanics like traps and items.

We also had some network issues. Since that's our first game built for the network, we didn't really think about how the network can affect the game. Our mind was operated as we were building a single player game that has connection features. The only worry we had were packaging issues since others had experienced that before. The game runs well on a single PC, but as we connected through the internet, some fatal errors appeared. We experienced serious lag and packet loss. During the painful process of looking for a solution to the problem, we realized we have to be careful each time we decide to send data through the internet. Since in our game players interact a lot by controlling the same robot, the game needs to be running smoothly in order to provide adequate experience for them. That requires us to do network profiling in order to find out how much data we are sending through the internet so we can shrink the size. We also realized that building a network game requires a lot of extra tests for the network part. Players always have different network conditions. We should have tested how the game performs by creating some lags or packet loss in order to simulate how the clients experience on their ends.

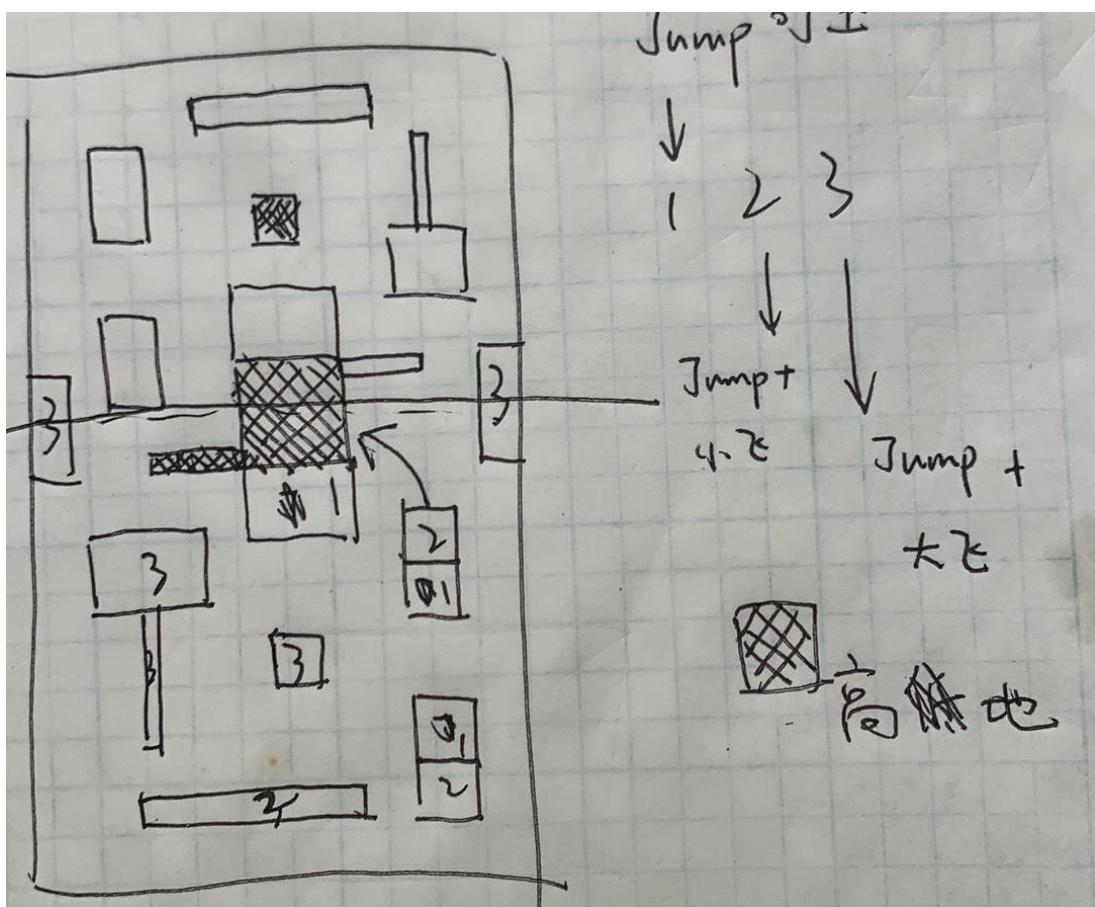
How we develop the game

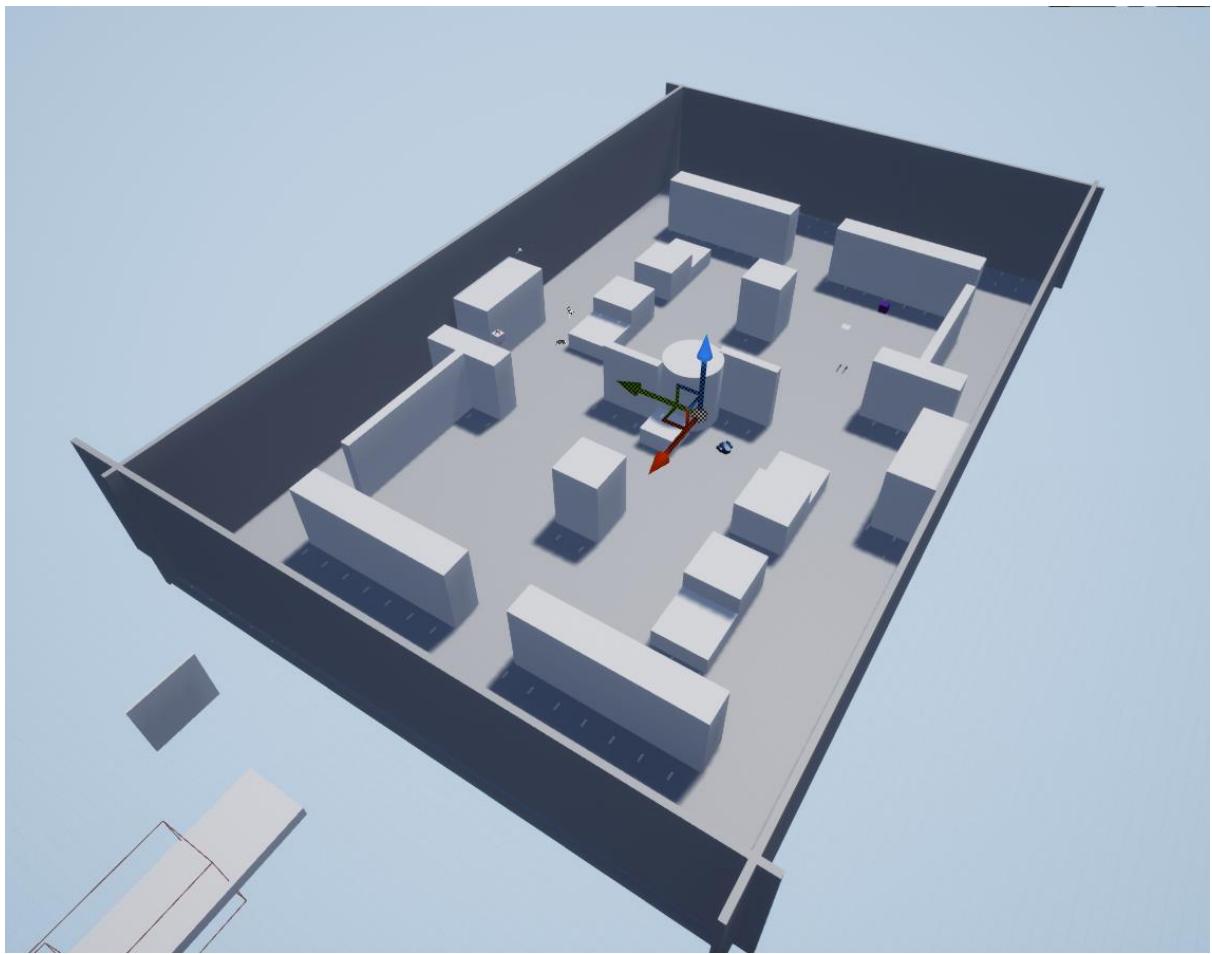
Since we are all working remotely, we use git to ensure all the members of our team can get the latest version. We separate our game contents into several parts in order to make the game can run smoothly:

- Basic gameplay. This part includes some basic mechanics of our game, including the internet sync, player movement, map collision.

- Polish game content. After we finish the first part, the game is ready to run. Then we continue to handle some polish issues such as player animation, texture, visual effects, UI, and menus.
- Extra game content. During the QA and playtest of our game, we found some disadvantages and then gave out some solutions.

As for the design part, we did what we learned before that to make a paper version of it then polish it in the engine. Considering the balanced issue, we make it symmetrical.





We also make another version of the map.



Game feature

- Shooter player: First-person view, can move the camera around. Press the left mouse to shoot and give out damage to the enemy.
- Movement Player: Third-person view, can move around, jump, and fly with the jetpack.
- Symmetrical map to balance the combat.
- If any robot was destroyed by the other, it will disassemble to two parts, and the movement part has to go find your shooter friend. The separate parts also have individual health points.
- The Player will die after being destroyed several times.

What we may do after the playtest

We are facing too many internet sync server issues that we failed to make it run smoothly and may take some time for our programmers to fix it. We'll try to figure out how the program works, and after that, we may polish the UI, animation, then work more on new features, such as what we planned before, to make it funnier as a PVP game.

Contribution

Programming: Luyao Zhao, Yanjiao Liu, Yonghe Mu

Model, rigging, art: Daoyuan Shen, Yanjiao Liu

Design: Yonghe Mu, Wenbo Wu