Working on Haptics for Project Esther

Introduction

In the latest rendition of CES, Razer showcased Project Esther : The world's first HD Haptics Gaming Cushion. Interhaptics is a software company specializing in haptics software that provides a general-purpose SDK and design tool for haptics compatible with multiple peripherals. Interhaptics was acquired by Razer in 2022, powering Esther with its platform Razer Sensa HD Haptics devices. HD haptics were initially pioneered by PlayStation with their DualSense hardware implementing both wideband and refined haptic sensations in the handles. This technology has since been progressively integrated into first-party titles such as God of War, Spiderman, or Horizon. These leaps and bounds in haptic technologies more easily immerse players into whatever game they are playing. Razer aims to 'one up' the competition by completely immersing the user of its devices with Project Esther.

The most important element when showcasing haptic hardware is to pair it with software which shows just how immersive the experience can be. The Interhaptics team decided to develop a custom-made Mod of MechWarrior 5 to showcase the capabilities of Eshter. Modding is the process of using an existing game already on the market and creating new content. This definition stretches from adding a feature to creating a level to changing the original design of the game entirely. In our case we created a level and added features that allowed for haptics to be used.

For the CES showcase experience, Project Esther was accompanied by a pair of Razer headphones called Hypersense V3 (aka Kraken) and a custom controller, both equipped for the occasion with Razer Sensa HD Haptic capabilities.

My team and I developed the custom MechWarrior MOD, showcasing Esther's potential.

We produced a unique level, especially thought up for the CES showcase demonstrating how Esther can be used to accentuate gameplay experience. Haptics was our main focus throughout the process of designing the mod. The following article will more thoroughly explain this process : the choices we made to focus on user-centric haptics, how we designed the haptic effects, and the particularities of working with Project Esther.

https://youtu.be/SbuLiEGr7Tw

Focusing on User-Centric Haptics

During this project, my role was to produce and integrate all the Haptic effects for the Mod. When starting off, we decided to mod MechWarrior 5 for two main reasons: Firstly, they had a well developed editor that was based on Unreal Engine 4.26. Usually, modding tools use unique studio engines that limit what external tools that can be integrated into the project. Mechwarrior 5's tool allowed us to use a very well known game engine that has a lot of support online which made it easier for us to integrate Interhaptics' SDK even though they support many different platforms and engines. Second, the in-game avatar is controlling a powerful robot or mech with strong destructive abilities, making it a great selling point for what type of haptic effects could be created.

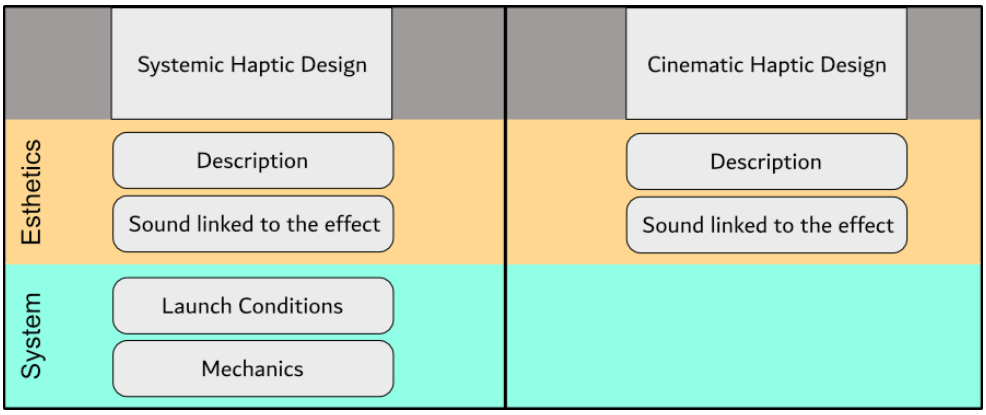
MechWarrior 5: Mercenaries is a BattleTech mecha game where the player accomplishes missions using different mechs to build-up a previously demolished mercenary group. The gameplay is similar to controlling a tank with a big emphasis on mech management with ammunition and weapon customisation. Our objective was to make a mission that fits the main narrative of the game, whilst making it accessible so that beginner players can easily play and discover project Esther.

The first step for our team was to find a mission structure that promoted a high level of accessibility while giving the chance to have interesting haptic effects. In MechWarrior 5, the

player has to move, shoot, and orient at the same time. The mech controls like a battle tank, which takes a while to get used to for the average player. This poses a problem for us designers. How can we make a hard-to-control game more accessible so that players of any skill level can just grab the controller, sit on Project Esther, and enjoy the experience? Our solution was to make the mission a Rail Shooter. This sub-genre of game is characterized by the fact it locks the avatar of the player on a predetermined path where they must focus on firing the incoming enemies on the screen. This would completely remove the need for the player to simultaneously move and shoot at enemies, all the while giving us total control over the time it would take to complete the level and the added benefit of adding easily controlled scripted events to the mission. This solution allowed us to really focus on what the user had to feel and how we could showcase the different haptic devices.

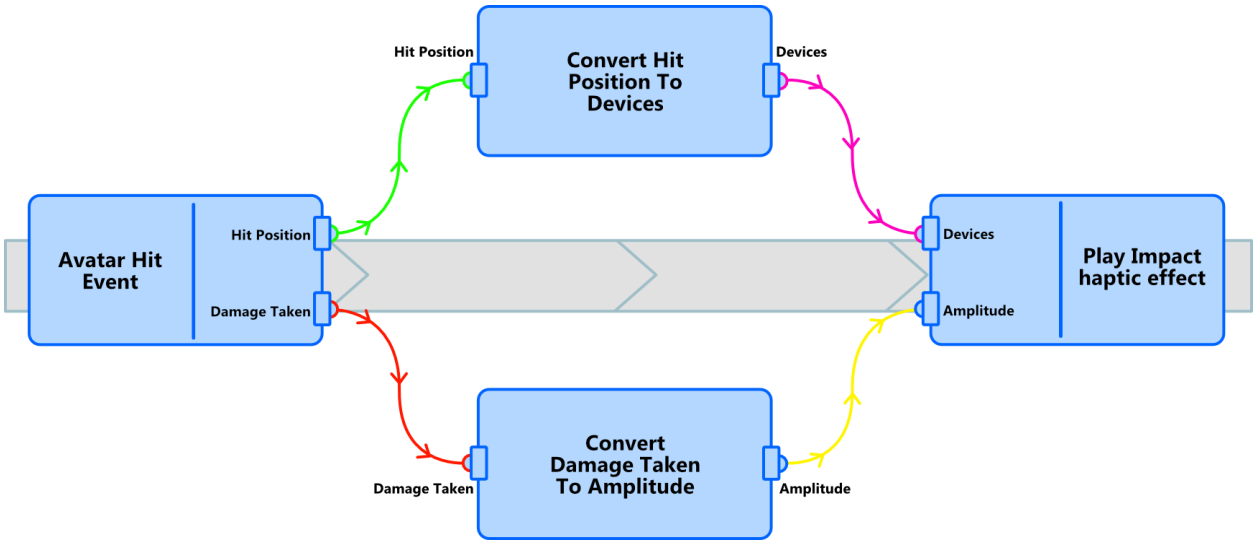
youtu.be/xy0mfibPWss

Once the fundamentals of the game were decided, our focus was on how to best showcase all the capabilities of Esther. The first step to this was linking a haptic effect to every possible interaction in-game. It could be a player-action (like firing weapons, or movement) or an interaction with the player's avatar (getting hit by enemy fire, or colliding with a building). This part can be considered the systemic haptic design part of this process, i.e. how certain events can launch certain haptic effects. The next step was deciding how we can create specific events to showcase different haptic ideas. Our team storyboarded the entire level we had created to achieve this. This document explained and illustrated what events would occur during the experience and what unique haptic effects would launch during them. For example, we detailed a scene where the mech would board the monorail. For this event, we described the interaction the player would make, the visual effects that would launch -in this case a cinematic-, the sound effects that would play and the haptic effects associated. This document allowed us to see if we fully utilised Project Esther's capabilities and gave us a road map on what was left to produce.



Systemic and cinematic design have a similar esthetic root to them. Each haptic effect is thought of through a description and a sound effect that resembles the description we give so as to be the most clear as to what the effect will aesthetically look and feel like. The main difference

is that when designing systemic haptic effects, we take into account when the effect will launch in the game, and the mechanics of the effect, for example “the effect will launch when the avatar collides with an object with the tag building”. The mechanics of the effect can come in various forms. A simple definition of this is how the haptic effect will change depending on variables or conditions in-game. For example, when the avatar is hit by an enemy weapon, the amplitude of the effect will be stronger the greater the damage the player receives from the attack. This method is commonly used when doing sound design and later on in the article I’ll go into detail explaining how we integrated certain mechanics into the mod.



We also had to rationalize how we would separate the haptic functions of each device: Esther, Kraken and the custom controller. Each device is positioned at a different part of the player’s body and, as a consequence, needs a different feedback role. Esther is a cushion where the actuators are connected to the player’s upper legs and back, Kraken’s actuators are next to the ears and the controller’s actuators impact the hands. Since the controller, the tool that allows control of any interaction in-game, is next to the hands, we decided that any “Player-Action” haptic feedback would be linked to it. This means that whenever the player would click a button, a haptic effect would trigger. Next we decided that Esther and Kraken would take on the role of “Reactor”, meaning anytime the avatar would experience an interaction (impacts, collisions, state changes) a haptic feedback would launch in reaction to those events on said devices. Most of these reaction haptic effects are linked to Esther and would only launch on Kraken if the reaction is critical for the player (very strong impacts, destruction of a part of the mech, hit on the cockpit).

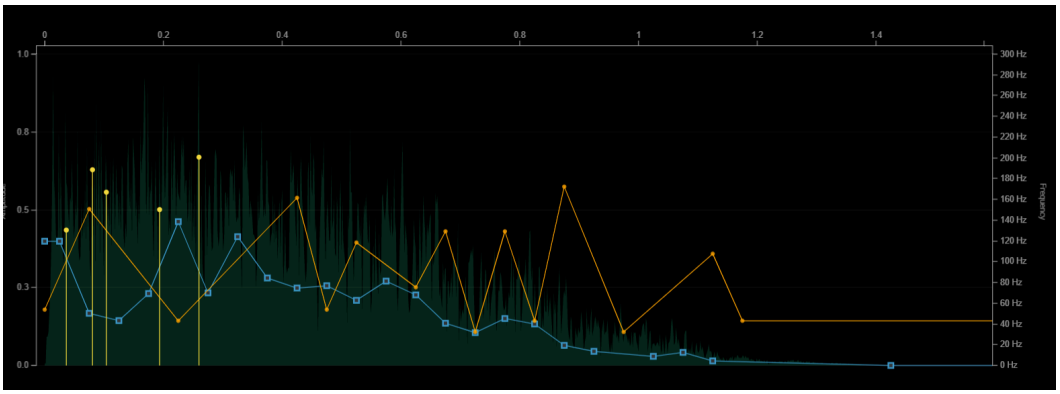
With the concept of the demo done, we proceeded to actually design the haptic effects that would appear in the demo.

Haptic Design

Haptic design is a uniquely multi-sensory design process. It is difficult to design an effect without taking into account the sound and visuals that are linked with it. When first

experimenting with haptic design, using Interhaptics' design software called the "haptic composer", I often found that the effect would be obtrusive to the experience. Sometimes the effects worked, but I didn't understand what made a good haptic effect and what didn't. It wasn't until I had a crash course with Interhaptics' CEO Eric Vezzoli that I understood how intertwined the haptics were with the sound. In the "haptic composer", it is possible to integrate a sound file and convert it straight away into a haptic effect. This feature exists because if the haptic effect and the sound do not match, the haptic effect will stick out to the user. The haptic composer allows you to edit your haptic effect while having the sound waves in the background to verify that the shape of the haptic effects matches the shape of the sound (see image below where the sound waves are in dark green). The fact that haptics and sound are very intertwined is a point that I will mention often in the creation and integration process of haptics. Having been a sound designer in many of my personal and professional projects (including this one), I can see the strong similarities between the two fields.

#add an image of an haptics effect with the background sound envelope



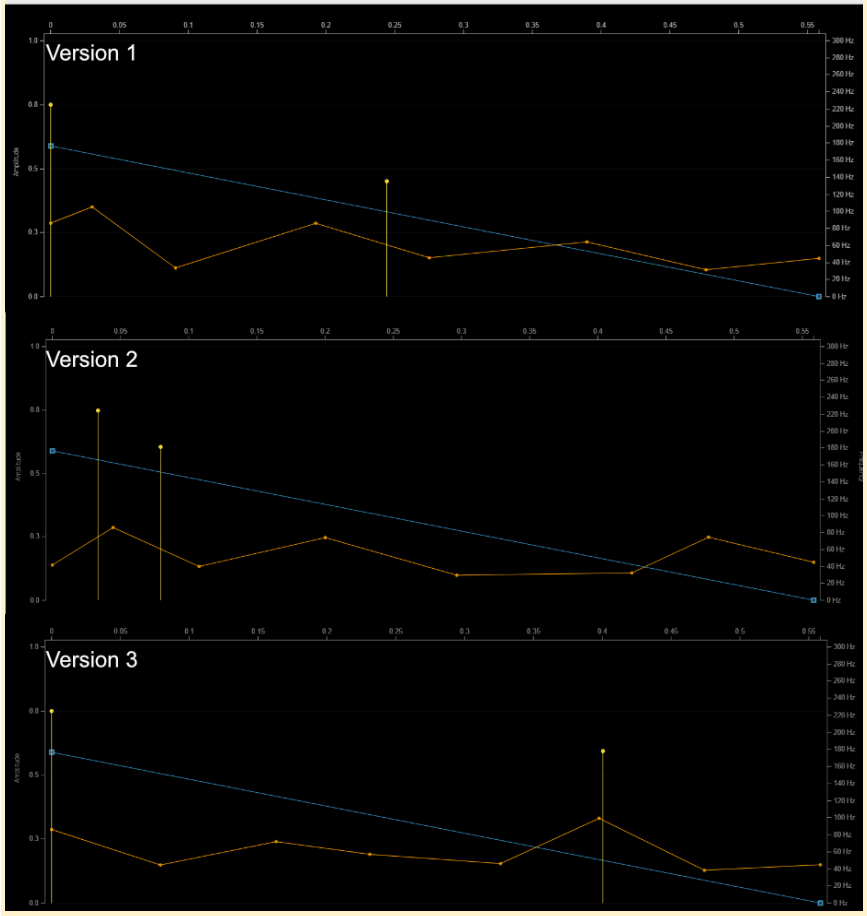
This means that, when starting the creation process of a haptic effect, there already exists the visual and sound effect that it will be linked to. The haptic effect would then have to be tested within the experience to see if it worked properly. In our project's case, we designed many events around what potential haptic effects we wanted and created events to manifest them. Before creating the effect, we would implement the event's systemic, visual and sound functions to then create the haptic effect linked to it. The way we organised ourselves was very top down, searching for the haptic effect and making the event around it.

When designing a haptic effect through the "Haptic Composer", the designer modifies how the actuator (the haptic motor) will behave. There are three variables available to create a haptic effect. The first is the "Amplitude" which modifies the intensity at which the actuator vibrates. When modifying it, it resembles a graph where the horizontal axis is time is seconds and the vertical axis is the amplitude going from 0 to 1. The second variable is called a "Transient". It is a point that can be placed on the time axis to create a feeling of impact. The vertical axis determines the transient's strength in the haptic effect. The final variable that can be tweaked is the "Frequency" which is linked to the speed at which the actuator vibrates its mass. This variable is most felt when the amplitude is fixed and the frequency changes, it changes the overall feeling from "rumbly" to "sharp" of the effect without modifying the intensity. The "Amplitude" and "Transient" variables are the ones that most follow the sound wave shape.

The example below shows one of our effects for a medium collision. This effect is played when the player's avatar collides with a building. We can see the "Amplitude" follow the general shape of the sound waves in light green and the "Transients" generally appear when a peak in sound is seen. I modified the frequency based on how it felt during tests.

Designing systemic haptic effects is similar to the process of Sound design. The most common mechanic we integrated into the demo was an effect randomizer so that the player can't easily perceive repetitions. The human brain is quite sensitive to patterns, and haptics are easily perceived by the player. So without variation the player will quickly feel the repetition of the effect and be detached from the experience. To solve this issue we had to create slight variations to the same effect. For example, we made 7 different haptic effects for when the player's mech collides with buildings. For the variations, we slightly modified the amplitude line (slightly so as to not make it too different from the sound wave shape) and changed the position of the "Transients" to focus on different peaks of the sound wave. After multiple tests, the displacement of "Transients" seemed to most impact the differentiation of the different effects.

3 haptic files linked to the same Foliage collision effect. Launches when the player's avatar collides with trees in the environment.



To furthermore increase variations we also randomized the amplitude of the effect, meaning that if the same effect happened to play twice in a row, it would be harder to perceive both instances as being the same effect.

Another mechanic we added to the demo is the looped haptic effects, like the jetpack. It is a much easier effect to create since we just have to make sure that the variables at the end of the horizontal axis are at the same height as the variables on the start of the horizontal axis. Interhaptics' software then handles the looping by itself. These different mechanics I presented to you above are commonly used in Sound Design when making mechanics for sound effects. The reason these practices are used are all to better player immersion by adding modifications on the effects based on the actions of the player or the environment.

Integrating Haptics into the project

The integration of haptics was only possible due to Interhaptics' work on the development tool and that MechWarrior 5's mod editor is based on Unreal Engine. The mod editor allowed the use of blueprints to work on functions which allowed us to create systems that would use Interhaptics' development tool to launch haptic effects on specific actuators located on Esther, Kraken, and the controller. The Esther used for CES possessed 16 actuators, Kraken possessed 2 actuators and the controller possessed 2 actuators. This control allowed us to integrate very unique mechanics for haptics to increase player immersion and player comprehension.

Project Esther allowed for the possibility of adding a mechanic that localizes haptic effects in the game space. In Sound Design, spatialization is playing sounds louder on the left or right side of headphones depending on the position of the avatar and the orientation of the camera, giving the player the impression they are really located inside the game space and bettering immersion. This was an effect that we believed could be translated into haptics. The term employed in haptics for this characteristic is Multidirectional Haptics.

Using Interhaptics' development tool, and thanks to the number of actuators on Esther, we were able to make a system of multidirectional haptics. I was able to program functions that activated only certain zones depending on the enemy position, impact positions, and avatar positions. For example, if the player's mech is hit on the right side, the haptic effect will be launched on the actuators located on the right side of Esther. When this mechanic was integrated, it completely changed the feeling of the game. Player immersion was heightened

since the orientation of the mech would directly impact how Esther would vibrate, giving the impression of truly being inside the mech, and player comprehension was increased since they could understand where they were hit. If multidirectional haptics did not exist and the player was hit simultaneously by two different enemies, he could not distinguish the different impacts.

Kraken, Razer's headphones, had also a few particularities that we had to take into account when integrating haptics into the game experience. Since the actuators are both located near the speakers, placed close to the user's ears, haptic effects can be audible to the player. Depending on the amplitude, the sound could be heard more than the sound playing through the headphones. My first instinct was to create specific haptic effects for Kraken and to keep the amplitude at a lower level so as to not be audible. However, after having integrated the explosion scene into the demo, I thought that actually being able to hear the vibration of the headphones could add to the grandiosity of the event. With a well-controlled effect, it is very powerful to be able to add a deep rumbly bass sound to certain effects. Though, I do not recommend using "Transients" for haptic effects on the Kraken, because it feels like and sounds like a jackhammer is hitting the side of your head.

Integrating haptic effects on the controller required a lot more precision. Our hands are more sensitive to haptics then our back and thus, the effect must precisely match the sound effect it is linked to. That is why we tested all haptic effects with the controller first to validate them and then tested on the required devices for final validation. The controller has two actuators, one located near each hand. We decided to localize most actions on the controller to simulate their position on the mech. For example, if a weapon is positioned on the left arm of the mech, the haptic effect will be played on the left side of the controller. To not detach entirely the left and right side of the controller, the effect is played more intensely on one side (90% intensity on the main side and 10% intensity on the other). The spatialisation of the effects on the controller helped to better understand which weapon was firing. If all effects played on both actuators, the effect itself would be the only factor to help distinguish them. For some weapons, notably the powerful shoulder weapons, we added the effects to Esther in addition to the

controller to increase the feeling of power. The combination of different devices was a big hurdle and required many testing and balancing sessions.

During integration, with three different haptic devices to work with, the biggest issue that had to be resolved through testing and precise tweaking was what is called “Haptic masking”. This phenomenon is when multiple haptic effects are played at different intensities and the player tends to perceive one of the effects. The body focuses on usually the more intense effect and completely removes the existence of the other. It's a phenomenon that Eric Vezzoli warned about and that we perceived during the testing of the mod. Since we noticed Haptic Masking when many haptic effects were played in quick succession during intense action sequences, we had to create a priority system. The priority system would stop haptic effects when a new haptic effect with higher priority is played. These systems are commonly used in Sound Design software like FMod and Wwise to have a more clear sound so that certain sound effects don’t play on top of each other, and we used this system for this same reason.

Once everything was integrated, we saw that we had 53 haptic effects implemented into the project. 40 haptic effects were systemic and 13 were cinematic. This means that 40 of the haptic effects change depending on various conditions that could happen during the gameplay experience. As explained previously, these changes can be in intensity, duration, location or form of the haptic effect, effectively meaning we have many more haptic effects then what we might think. Using Sound Design concepts allowed us to create all these different variations to make every interaction unique and help participate in the immersion of the player within the game experience.

Conclusion

I can not emphasize enough how much Esther improves the gameplay experience of our MechWarrior 5 demo. Having tested the demo many times with and without the haptic devices, I am convinced by the scale of how greatly it betters the game experience. The player simultaneously is more immersed in the experience and better understands what is going on within the game space through haptics.I had not previously worked on haptics before this project with Interhaptics and had no idea how greatly haptics impacted game experience. I ended up making 53 haptic effects for an 8 min Demo. After working with Esther, I now see a new world of potential gameplay experiences. Esther allows game studios to push the immersion of their games. Many newer releases are innovating in haptics, like Ratchet and Clank: A Rift appart’s trigger haptics creating great variations in the feeling of weapon fire using the PS5 controller’s resistant triggers. Avatar: Frontiers of Pandora has around 400 haptic effects designed for the game. Haptics are slowly becoming more of interest to many studios and I am hoping that Esther pushes them to search for new innovative ways of deploying haptics.