legends: Orange (Medium Priority), Red (Critical/High Priority), Blue (Info), Green (Good work!)

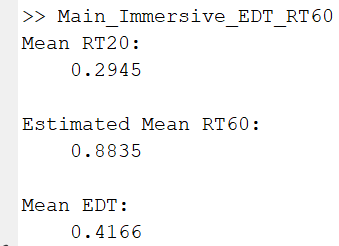
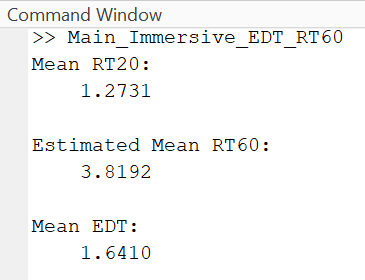
Done, Important

# Monday, 29 July 2024

## **FOCUSED TODO**

1. Use RT30 for RT60 estimation via extrapolation, and generate all graph/value to compare with Kim21, especially for RT60 on each bands as seen on picture above.
2. Try maximise/increase global steam audio parameters to increase accuracy (hopefully?) to take advantage of using baked instead of realtime.
3. Figure out what causing early reflections (high EDT) in ST scene.

Continue with trying to get RT30 working properly with provided matlab code because for some reason simply changing y\_fit to fit only 30 instead of 60 doesn’t work as expected.

1 good news (not really) and bad news. Good news is that the codebase itself it most likely working as expected (y\_fit), atleast it works the same as it is for RT60, the problem is that even on RT30, the best fit line freaks out when its not a good energy decay curve ie one without weird flat noise level. This is proven as using RT20 instead to get estimated RT60 gives very similar value to expected RT60 for KT. The bad news is that this doesn’t work at all for ST… Which means we need to go back to drawing board and find out what is difference between Mona vs mine and see where this weird noise floor level coming from. Ideally there should be no changes to codebase, or as little as possible and still use RT60 instead of extrapolation.

So for now, lets get back to drawing board and mess with Unity/Steam Audio to get rid of the noise level thing.

## **UPDATED FOCUSED TODO**

1. Troubleshoot the RT60 problem in Unity (reduce noise floor level)
2. ~~Use RT30 for RT60 estimation via extrapolation, and~~ back to normal RT60 calc. generate all graph/value to compare with Kim21, especially for RT60 on each bands as seen on picture above.
3. Try maximise/increase global steam audio parameters to increase accuracy (hopefully?) to take advantage of using baked instead of realtime.
4. Figure out what causing early reflections (high EDT) in ST scene.

Hypothesis 1: Due to the fact that my scene mesh are not voxel, maybe that’s why noise level higher?  
Let’s try it on S3A and see how it performs, next lets use 0.5 volume setting on ST.

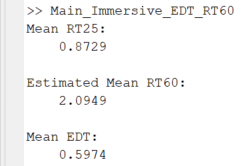
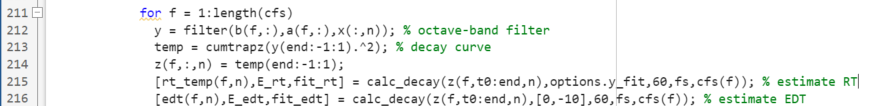
Maybe combination of 0.5 volume + ST30 extrapolation can work? If doesn’t, lets also add on 0.5 direct mix level instead of 1.

Ok both did not improve it by much.. Maybe lets turn off HRTF.. Nope, not doing much either.. Lets try 0.25 volume. Still too high.. ok EDT still soo bad, this needs proper experimentation and testing and troubleshooting instead of shooting into the dark. Lets list out all related component and settings etc. idk my brain fried for today. Lets figure it out tomorrow man.

# Tuesday, 30 July 2024

WFH, went back to drawing board. Studied the matlab code and research about related topic to further improve understanding. Brainstormed some ideas to troubleshoot problems.

# Wednesday, 31 July 2024

After reading the code line by line to figure out whats wrong, found the culprit! It was 2 lines in irStats that presumably forgot to account for y\_fit value changes, they hardcoded 60 instead of using y\_fit value for EDT and RT60 best line fit and calculation… That’s why the graph best fit line don’t really change even after changing y\_fit. THIS IS BIG FIX! Finally fix the bug halting progress for more than 4 days.

On line 215,216, the value (60) should not be hardcoded but dynamically change depending on value of y\_fit. After testing values on ST using 0.25 volume sound, it seems RT30 is too high and RT20 is too low, thus RT25 is a good middle ground to settle at getting 🡨 following result.

TODO tomorrow is to change the value before fs to dynamically change depending on y\_fit argument given, so the calculation works as expected. After that lets reupdate focused TODO. Done for today.