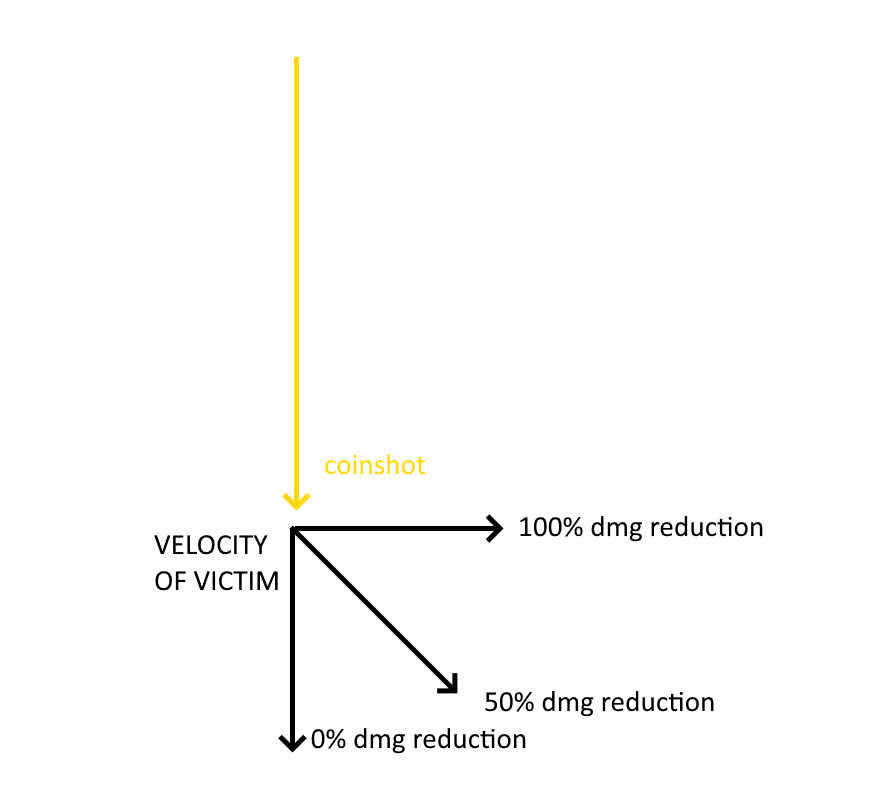
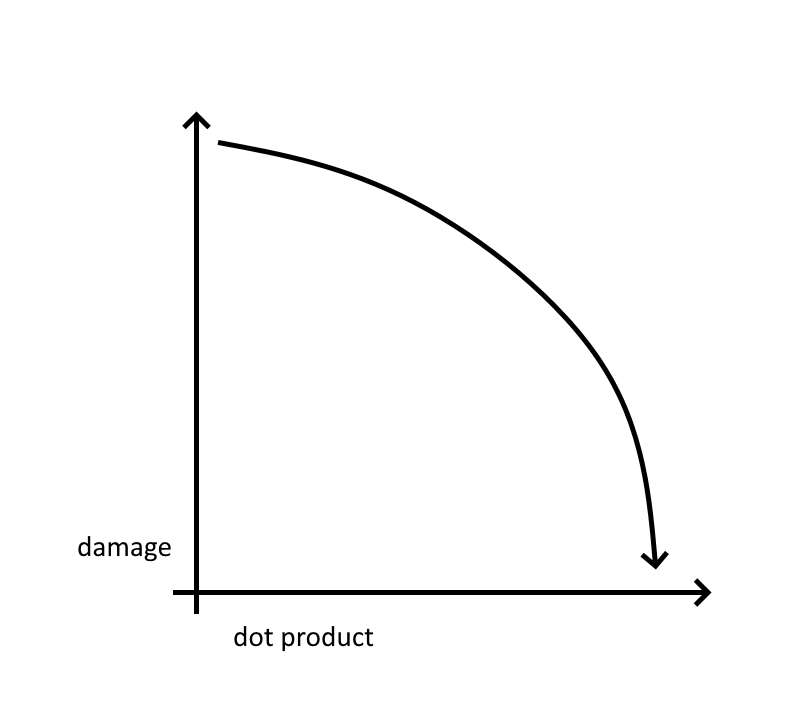
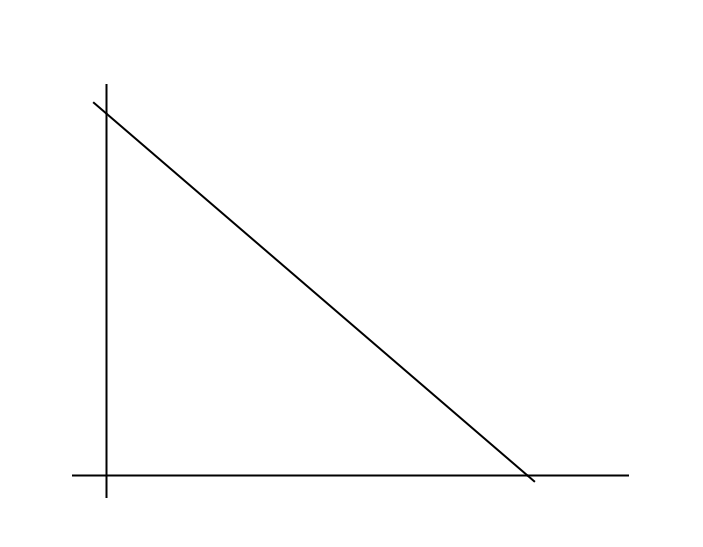
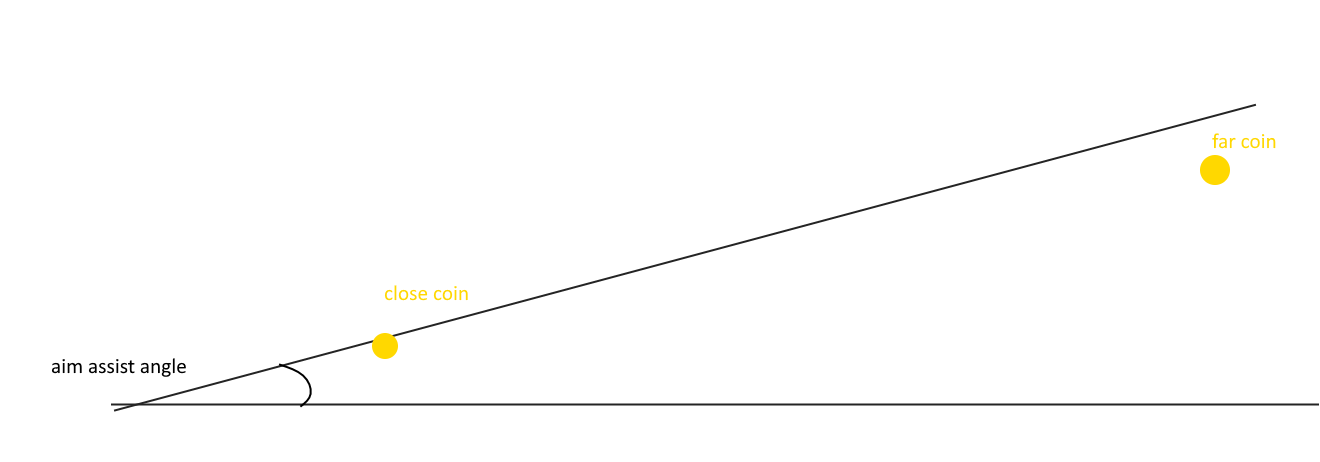
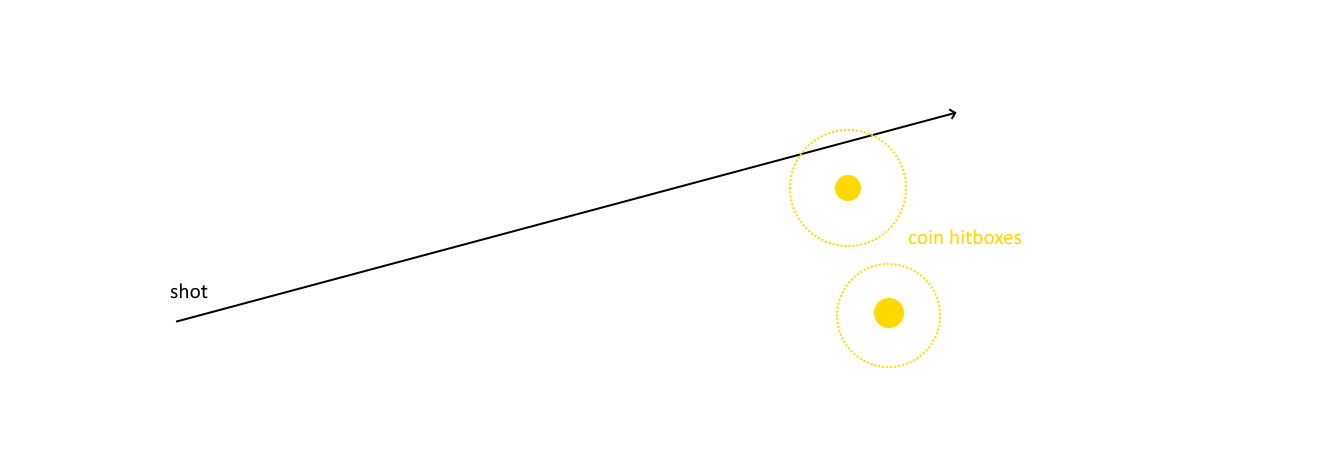
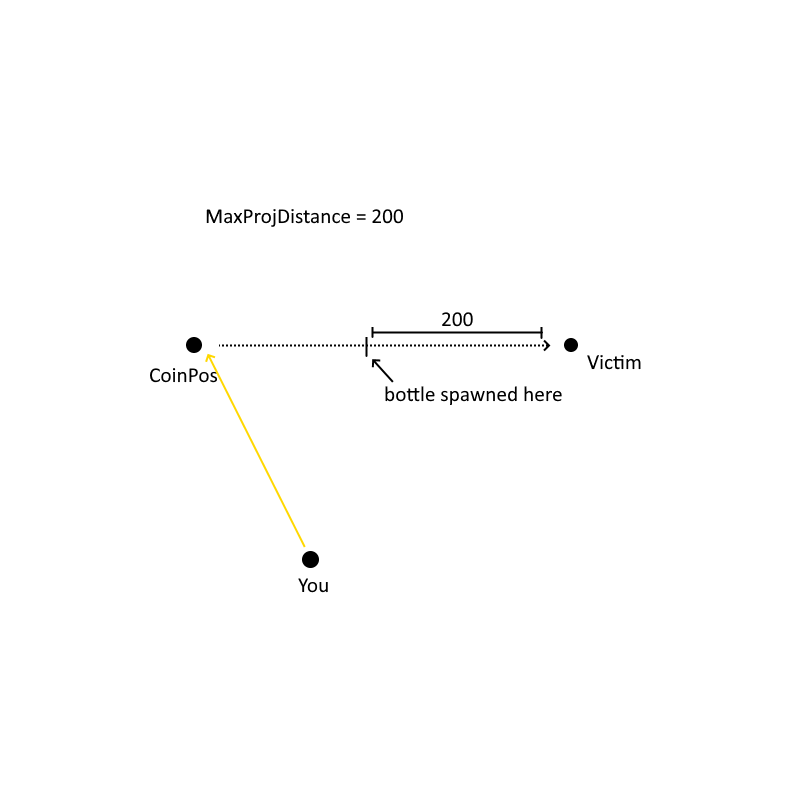
Grazing  
 The Grazing function was created since the non-bottle damaging method of shooting the coin made it hitscan. How do we make hitscans dodgeable? You realistically cant. BUT Katsu devised a way for others to take less damage from this hitscan attack depending on 2 factors.  
  
The first factor of negating the coinshot attack is player velocity. The second is if the player is moving more perpendicular to the coin or more parallel.  
  
Using math shit I have no idea how to properly explain, the jist is that if you are moving toward or directly away from the coin you may as well just be standing still. You will take full damage if lined up with the shot.  
  
In an ideal circumstance, a perfect dodge and NO damage is taken is when you are on a path exactly 90 degrees of the coin’s ricoshot attack AND moving at a pace of 600+ source units per second. This is possible for advanced players through well-timed acceleration hops (forward/backward) and the most guarenteed consistent way you can dodge this attack is without using the Movement script.  
  
For normal players, damage is “Grazed” or partial dodged.  
At greater than 350 u/s (roughly sprint hopping speed), the average player can make somewhere around a 50% damage reduction when perfectly moving 90 degrees of the attack.  
If moving at less than 350 u/s AND moving <45 degrees from the shot coin, the player should expect to take around 75% or greater damage.  
  
 The Movement Script’s Dashes easily accelerate you beyond what a vanilla player is capable of and so is suitable for negating this attacks damage although the second factor is still in effect. You need to be moving as perpendicular to this attack as you possibly can. But once mastered, the Dash dodge will be the most ideal way to dodge an opponent’s coinshot damage.

Grazing’s Curve vs Slope

Initially when programming how grazing damage should be dealt, Katsu opted for a curve equation.   
  
150 is the damage of a .357 headshot in source according to our PVP and balancing expert, RadioJackal, and so was chosen as the maximum capable damage of an un-mitigated 4 coin ricochot attack.  
  
Katsu took this maximum value and turned it into a curve that would be manipulated by the grazing mechanic.  
  
150\*(x-0.1)^1/7  
and x=1  
  
  
  
  
  
Katsu came to realize this curve created a balancing issue.  
  
When using the curve, the coinshot would heavily favor doing MORE damage than not when strafing.  
  
Im paraphrasing, but Katsu basically went *“my code literally having pyronies disease is cringe ngl”* and straightened that shit as soon as he could.  
  
The NEW sloped version creates fairness in how moving out of the way, and moving quickly are equally valued variables in damage reduction as opposed to favoring one input over another.

Hit Detection  
In early production of the E2, the only way the coin coild be triggered was if shot directly and damaging its prop shell which…. Was a terracotta pot made for plants. As anyone could guess, how the hell are you supposed to hit that consistently? And across a spawnzone? Let alone 2 yards out.  
  
Very few people would be able to use this E2 if we kept it that way, and that CERTAINLY was not how hard it was to hit in Ultrakill, the very thing we are basing this off of.  
  
So Katsu built a primitive aim assistor into the script. This aim assist is a cone that basically says “if the coin is within this degree of your crosshair, you will hit it”  
  
The obvious problem with this is coins that are furthest away are piss easy to hit.  
And if Katsu was going to add a countering mechanic, this needed to be done away with immediately.  
  
Katsu’s gift for implementing mathematic fuckery prevails again. By changing the aim assist from a 2 degree cone to a per-coin sphere of 47.5 hammer unit radius, Katsu solves multiple problems and sets the foundation for balanced counterplay. This change also helps make the ricoshot chaining mechanic possible now that there is no way you can accidentally trigger multiple coins simultaneously. Far coins are no longer piss easy to hit and require extreme percision, and ricochot chaining. Near coins, conversely, are appropriately accurate to Ultrakill’s ease of hitting. Since coins have their own personal hitsphere, you cant cheese them through walls like the old version from 2020. The sphere method is slightly more taxing on the CPU cycles of the E2 chip though but worth every cycle.  
  
The maximum range of coin-to-victim targetting was reduced to 3000 hammer units as well. Sniping someone from across the map now requires punting the coin within 3000u of the target before iether shooting manually or setting off with ricoshot. This change also helped to make sure plebs chucking coins around on 1 corner of flatgrass don’t snipe you off of the spawn zone. They now need actual skill to accomplish that from that distance.

The Bottle Method (Depricated)  
The first Idea I had when commissioning this E2 to anyone was proposing that a bottle should be used to act as “The bullet” that ricochets off of a shot coin.  
I chose this since, bottle go break, no prop cleanup required, and with sufficient mass and speed can do plenty of damage to a target.  
  
And so Katsu made it happen. This meant players struck would instadie. There was no good way to make the bottle do consistent nonlethal damage since source was finnicky and always considered “crush” as a factor when applying prop damage.

Crushing damage means any thing applying compressive force down on a player, or squeezing the player against the map means that object now does critically obscene ammounts of damage and kills the player. So even if the bottle mass was 20, yielding roughly 30 damage a torso shot, some hits would instantly kill the player when hitting at certain angles.  
  
Before katsu EVER switched to using the new hitscan damager & graze functions, He implimented “ProjDist” (See figure above).  
  
The ProjDist function’s goal was to make the bottles as close to hitscan as possible, making them harder to dodge, but this also meant hits would do less damage when the bottle has had less distance to travel. ProjDist as you wouldve guessed stands for Projectile Distance. And more specifically. Max Projectile Distance.  
  
This distance is how many source units away the bottle would spawn from a player. Katsu opted to make it spawn 200 units away. Sufficient enough to have some velocity and travel in order to damage the player, and not enough time to really dodge it.  
  
MaxProjDistance used to be a configurable variable in the upper page of the E2 script. One any skid could access and alter. This has since been removed in the latest revision because honestly why the fuck would you be changing it.  
  
Katsu rightfully deems this hit method absolutely primitive and archaic. The E2 will ONLY use the bottle method when it cannot find the dependencies for hitscanning a player on a server.  
  
None the less, the ricochot damage stacking somewhat applies to the depricated method although not nearly as consistently as the true hitscan because of crushing damage.