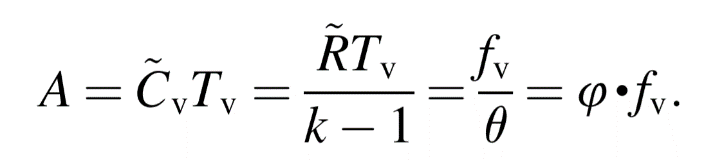
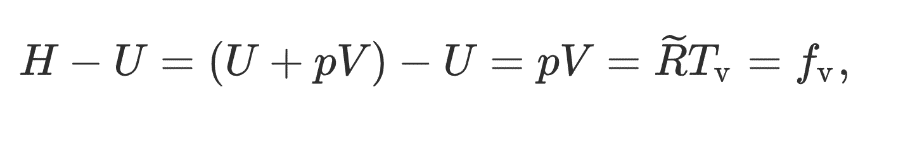
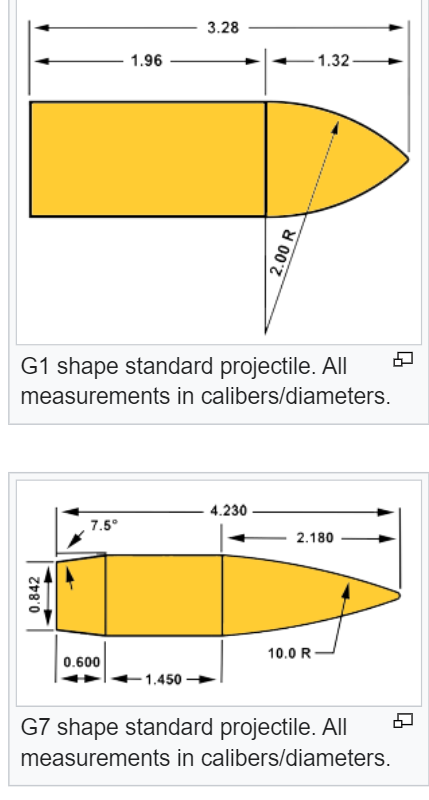
Gun Script

 My calculations and data for the bullet velocity calculations come from googling the exit velocity of a 5.56mm round coming out of a 13 inch barrel at 910 m/s at 1atm and 273.15 K. The model I developed for it is, as the gas expands from behind the bullet, it applies a force at a distance equivalent to the barrel. Giving us a work equation. Calculating the Force behind the bullet or F of propellent is Fp = RT. R being the gas constant and T being the adiabatic flame temperature. However solving for T and R is more complicated than it sounds, ie:

Now the great thing about this is, we know each of these variables under the same pressure and temperature will have the same output depending on the enthalpy of the gunpowder reaction. Enthalpy is H = E \* PV, however these equations solve for p and v so we will discard them, Our formula for Fv will be F = E \* gamma, where gamma is as constant and we classify gamma as being the same for all states, as pressure and volume remain equivalent for bullet types, and the rest will remain the same somewhat similar and having small variations, like air temperature and air pressure. Gamma is really hard to calculate, however we are able to back track the calculation for Fv, and we are able to backtrack the calculation for H, the rest of the equation we will group into gamma as a constant. Gamma is calculated to be, , and the equation for H is, Gunpowder mass \* heat capacity of Gunpowder, From this we can develop and equation for exit velocity,

Vf = ((2\*d\*H\*m\*gamma)/ m)^(1/2)

Update: This formula works fine, the bottom mass is the weight of the projectile and the upper is the grain of gunpowder (I confused the two), also enthalpy is a constant for a single mass of gunpowder, and then multiplied by weight to get energy released. Also my unknown coefficient gives similar velocity outputs for other guns. If I wanted to be exact, I can calculate the

unknown coefficient for all guns and bullets and then take an average of

them. unknown coefficient = gamma

For the Bullet drag,

Bullets are classified depending on there Coefficient of,

or the more descriptive mathematical mode, the Ballistics Coefficient.

A set of rankings include the G1-G7 types, the most popular kind. The shapes

are designed to create laminar flow behind the bullet so it reduces the drag,

Usually bigger and faster bullets will be shaped like the G7 but slower

Rounds like the 9mm pistol will be shaped like the G1 as the elongated back

has minimal to effect to its laminar flow at its max speed. The coefficient of

drag can be calculated by classifying the bullets in the game into there

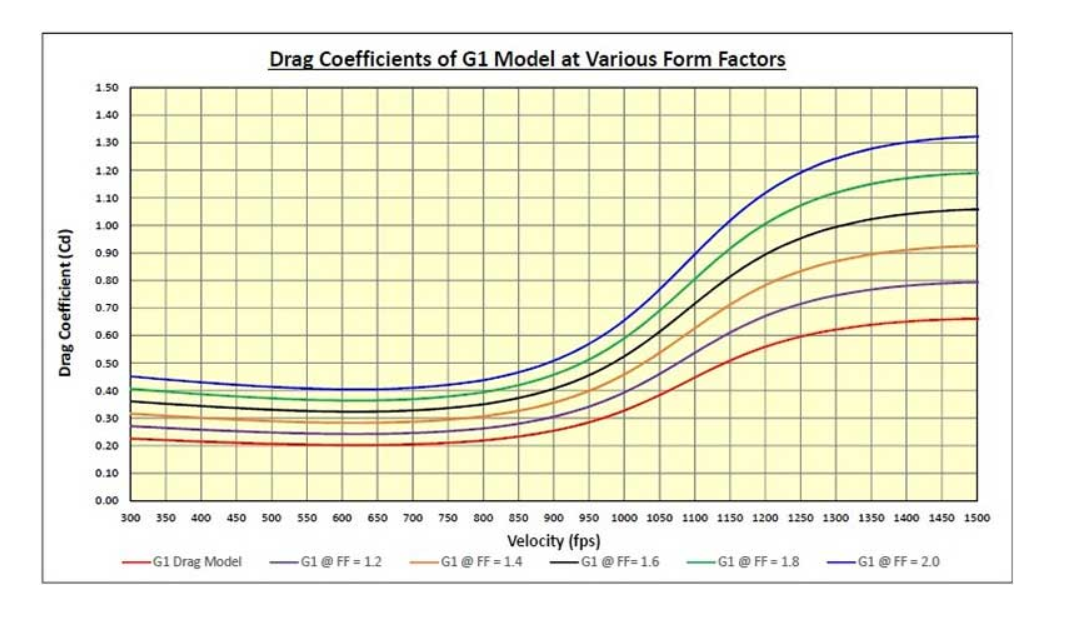
equivalent categories with there common ballistics coeffienct, from there

we can calculate the area of the bullet depending on its size, or form factor,

and provide a look up table for each bullets drag and set it accordingly in the

game.

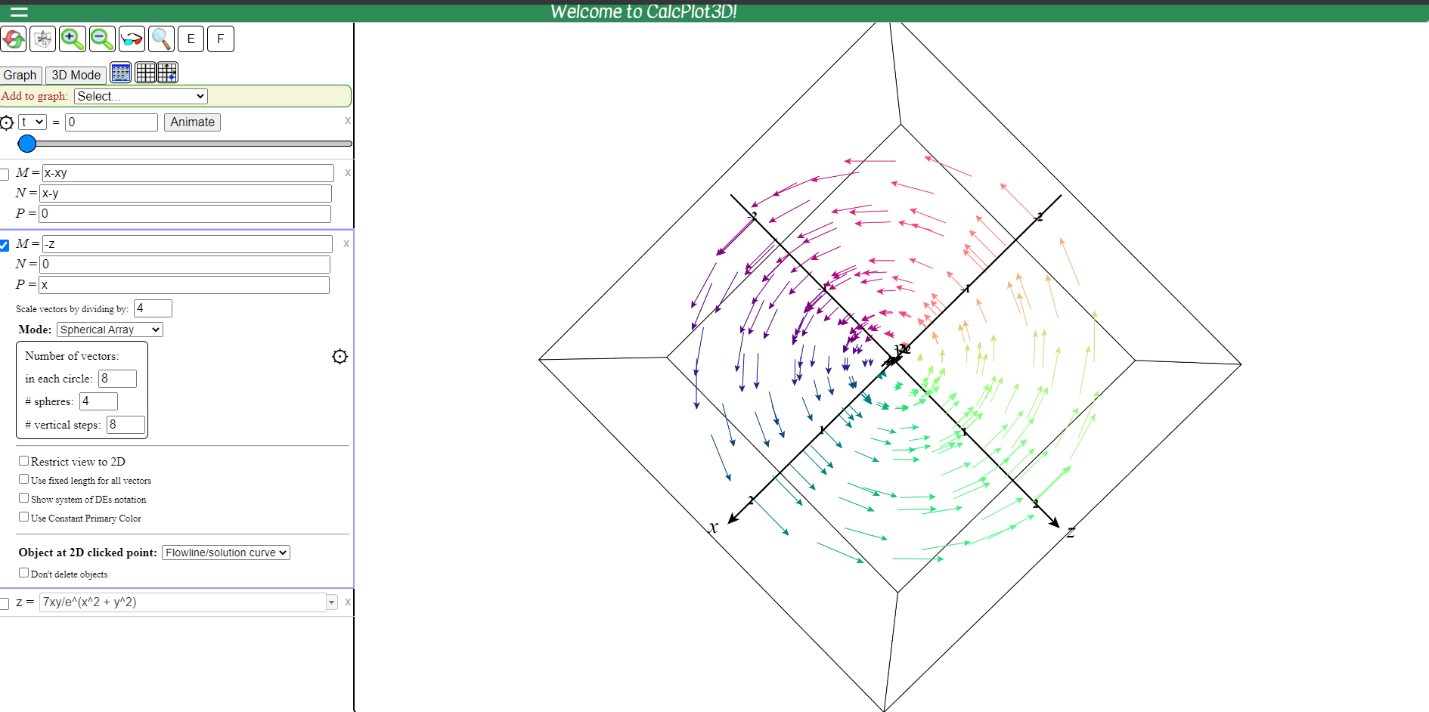
Here’s a table for the G1 model with different form factors, for my test game I’ll use the coefficient of drag assigned to the generic FF at the standard muzzle velocity of the m240, at 0.4.



For the wind factor

A nice addition to the quality of bullet physics would be the addition of wind effect on the bullets, simply splitting the bullets sides into 6 groups, front back, and pointing to the left from the back, and pointing the left from the front, and same for the right. Assigning an area for each quadrant and then depending on which area the wind vector falls in we can determine a force of wind with a generic air pressure and density value. This would be best modeled in a vector field with the equation,

F = Pi + Qj + Rk, and then a simple line function F(x,y,z) for the bullet, computing the line integral of these two equations will give us an equation for the force of the wind and which direction its point to on the bullet during its path through the wind field.



Screen shot of my vector field for the hurricane class, I can add more and make different types of wind effects with the vector field. The line integral is unnecessary.