Cat and Mouse.

Model description:

- ☐ Mouse moving toward to hole (to abscissa)
- ☐ Every dot on abscissa is hole : (n;0) where n is any value is hole
- ☐ Cat always moving toward to mouse
- ☐ Cat is catching a mouse when there's distance less than radius of attack

Other descriptions:

- ☐ Cats' movement vector equals to sum of reversed radius-vectors of cat and radius-vectors mouse
- ☐ After the changes, I get a new radius-vector of the cat, that equals to the sum of the previous radius vector and cats' movement vector

Counting the distance traveled by a cat:

```
vec_cat = UnitVector(-r_cat + r_mouse) * (v_cat * dt + a_cat * dt * dt / 2);
r_cat = r_cat + vec_cat; // new cat radius-vector
v_cat = v_cat + a_cat * dt; // new cat speed
```

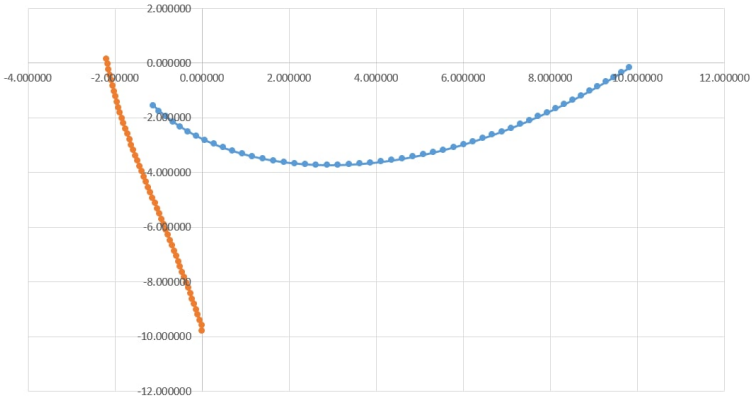
```
Cat movement vector equals to sum of vectors divided by vector length
and multiplied by uniformly accelerated movement
```

Condition for mouse acceleration, when cat is closer than critical distance:

```
if (Norm(r_mouse + r_cat) - distance < 0)
{
    vec_mouse = UnitVector(vec_cat + vec_h) * (v_mouse * dt + a_mouse * dt * dt / 2);
}
else vec_mouse = (-r_mouse) / Norm(r_mouse) * (v_mouse * dt + a_mouse * dt * dt / 2);
```

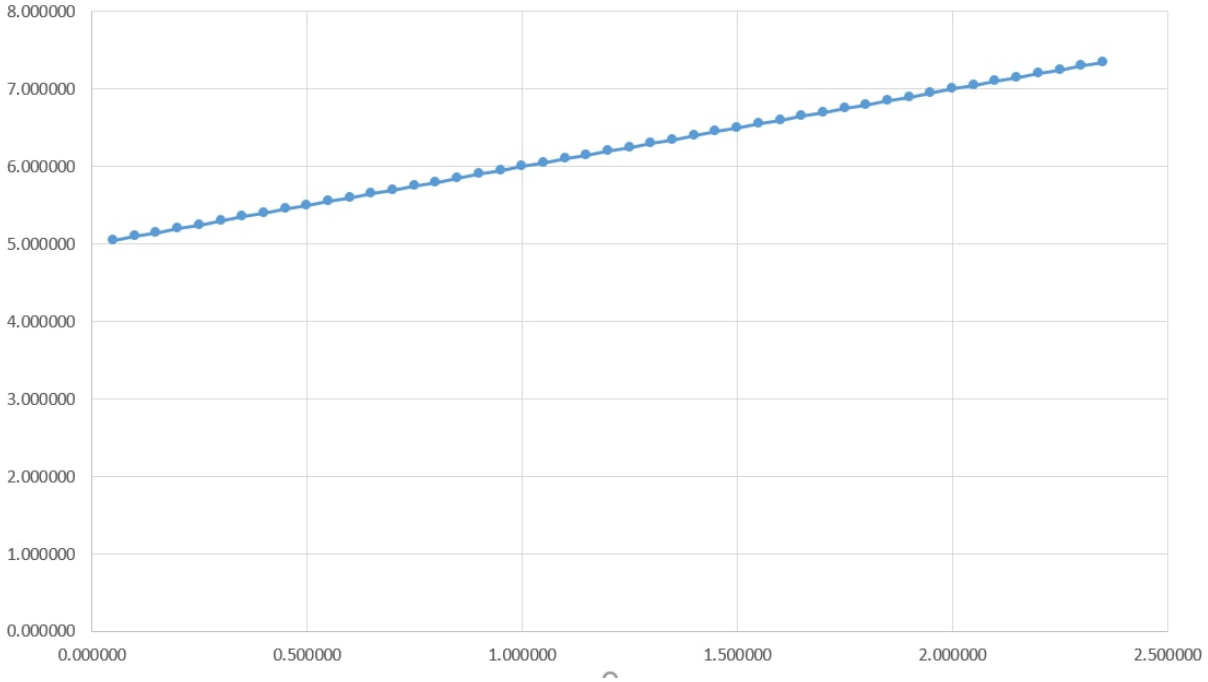
```
If cat is near to mouse, calculate new mouse vector, which is equal to
sum of mouse vector to abscissa and mouse vector of running from cat
multiplied by uniformly accelerated movement
```

The graph shows that the mouse deviates from the rectilinear motion:



After adding accelerations for both we get constant cat accelerated movement and uniformly accelerated movement of mouse:

cat speed vs time graph



mouse speed vs time graph