Simulation of baseball in 2D

Introduction

I want to simulate a baseball bat striking an incoming ball based on an starting point, contact point and acceleration for the bat. More specifically, variables for the mass, starting velocity, starting angle, starting height and gravitational pull. Based on this an original movement arch of the ball will be calculated. The contact point of the bat and the ball must be on this arch.

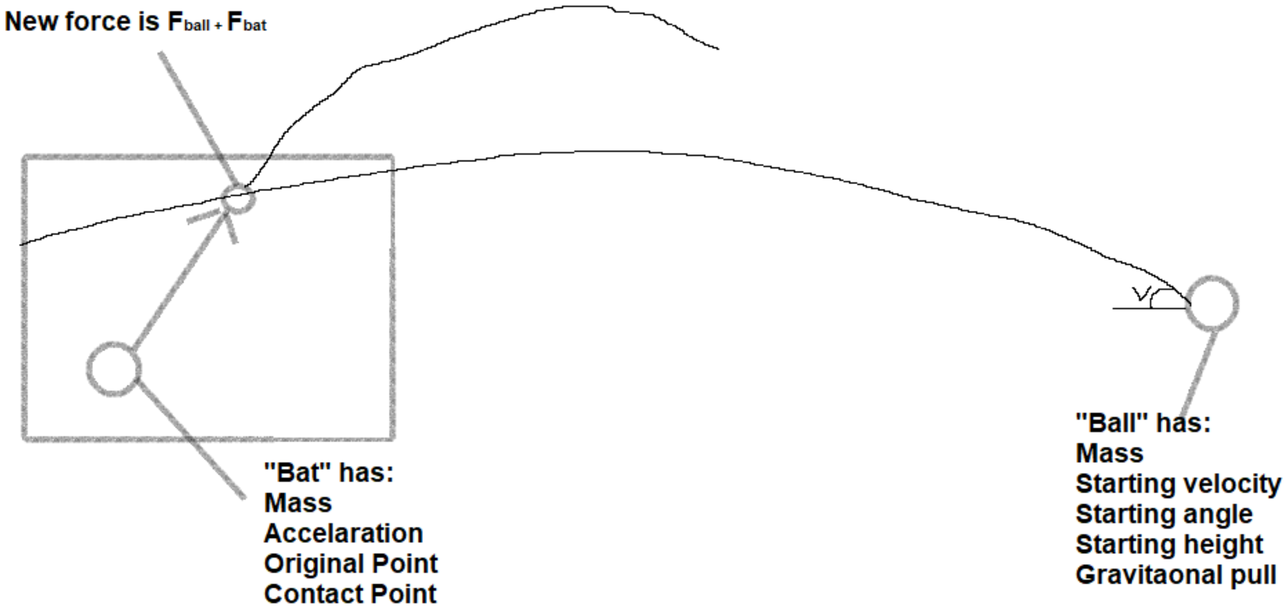
Steps in simulation

1. Calculate starting velocity vector for ball
2. Calculate velocity vector[i+1]
3. Calculate position vector[i+1] (Original arch)
4. Calculate starting angle for bat
5. Calculate contact-time for bat and ball
6. Calculate starting time for bat
7. Calculate balls new starting velocity after contact
8. Calculate new velocity vector[i+1] (Same model as step 2)
9. Calculate new arch (Same model as step 3)

I have ideas of how to implement each of these steps

Concept drawing

Please ignore misspellings



Possible extensions

- Multiple "bats"

- Let the program optimize collision-point
- Let the "bat" be affected by gravity
- Let the bat be a rotating body (instead of just another moving ball, which is the idea right now)
- Extend to 3D (This seems quite cumbersome though)
- Bouncing ball
- Air resistance

Visualization

Maybe implement in python and use pygame for visual representation. Unity otherwise

Feedback

I know this is quite a trivial simulation (basically only a simulation of collision in 2D space) with only 7 steps unique steps, but I also just want a pass. Is this advanced enough for a pass? If not, what extensions are possible to make it advanced enough for a E of higher (extending to 3D seems quite cumbersome)? Or is this just a dead end and should I just change concept? **I am only a one man show and as stated previously, I just want to pass**