

# Using Heun's Method

What if the acceleration changes during each timestep?

Recall that Euler's method provides a simple means for calculating new values of some physical quantities, based upon their current values. In the case of a man falling through the air (without air resistance), the variables are:

$t_i$  = current time

$t_{i+1}$  = next time

$v_i$  = current velocity

$v_{i+1}$  = velocity at next time

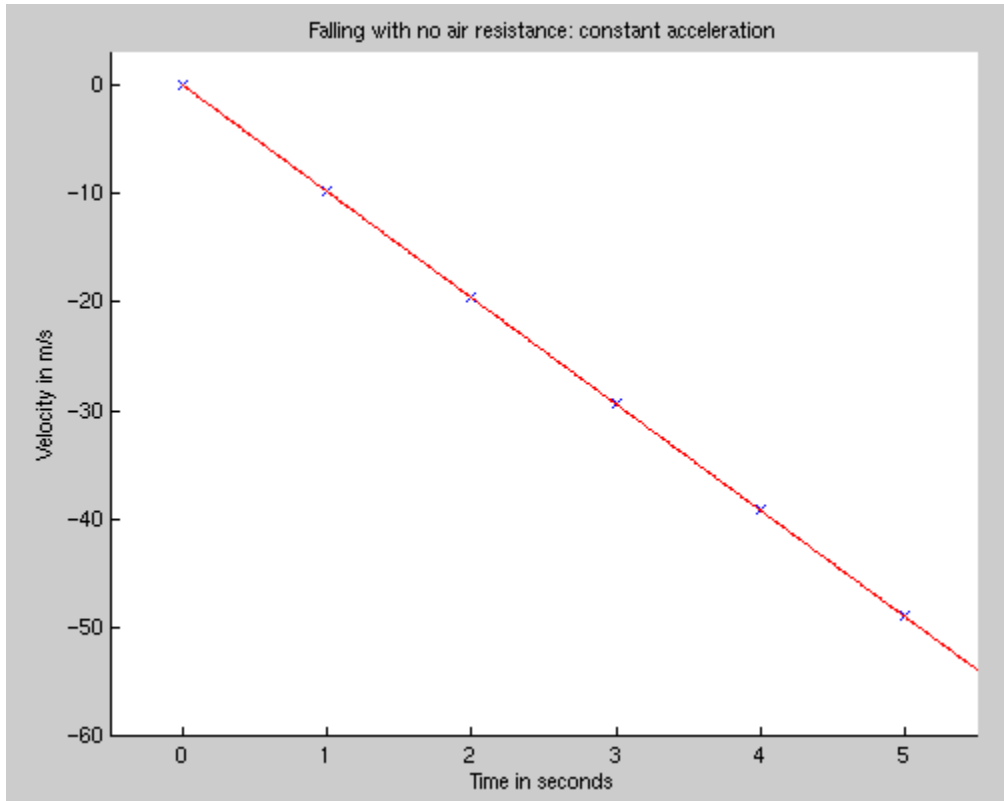
$a_i$  = current acceleration

Euler's method provides this blueprint for calculating the future values, based on the current ones:

$$v_{i+1} = v_i + a_i * (t_{i+1} - t_i)$$

$$y_{i+1} = y_i + v_i * (t_{i+1} - t_i)$$

Which produces the following values of velocity versus time, near the start of the fall (blue crosses are results of Euler's method, red line is actual, exact solution):



So, Euler's method uses the current values of acceleration and velocity, and **assumes** that those quantities don't change over the coming timestep. In this case -- no air resistance -- it works well. And it should, because the change in velocity **really is a constant** in each timestep: the velocity changes by exactly  $-9.8$  meters per second each second.

In real life, velocity might not change in such a simple fashion. In that case, is there any way to improve the estimate? Sure! If we **know** that the acceleration and velocity are probably going to change -- why not try to take that into account?

It will help a bit in describing how to improve the estimates of acceleration and velocity if I add a little complication to the system. Let's now allow include air resistance in the problem. The force of air resistance depends on the cross-section area of the jumper, the air density, and some other stuff, and it also depends on the velocity with which he falls. Let's describe it as

2

force of air resistance =  $K * v$

where  $K$  is some constant that depends upon the density of air, cross-section area, etc. This means that the net force on the jumper is

$$F(\text{net}) = -mg + Kv^2$$

and his acceleration is therefore

$$\frac{dv}{dt} = a = \frac{F(\text{net})}{m} = -g + \frac{K}{m}v^2$$

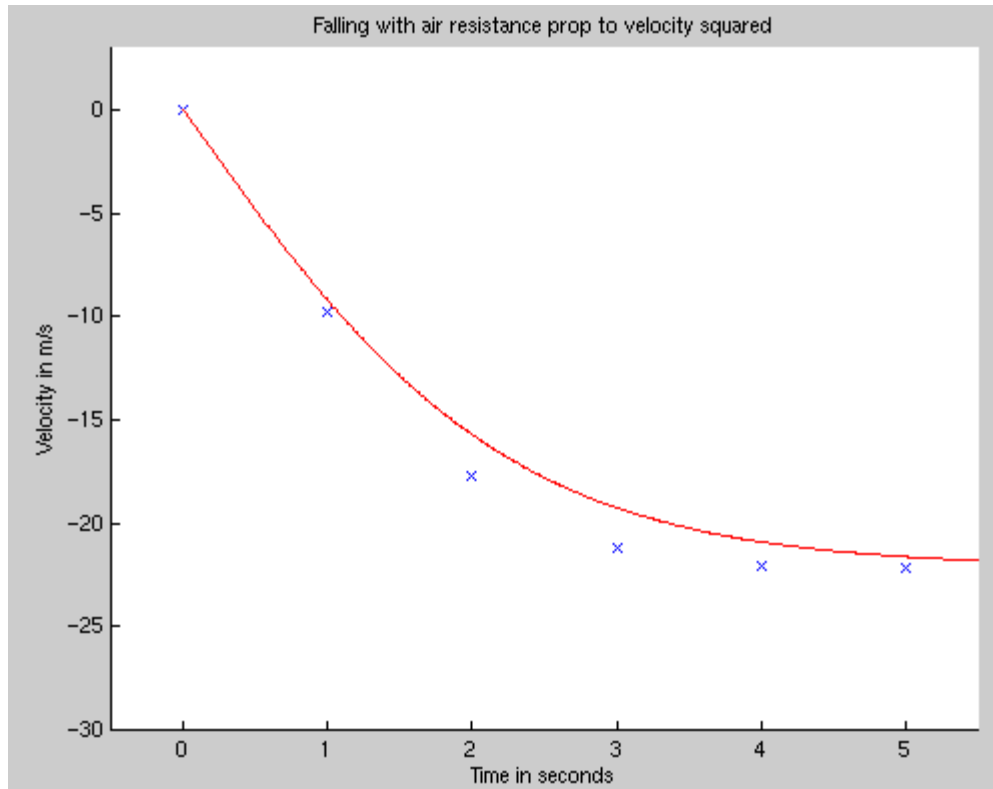
Now, the velocity doesn't change by a constant amount each second; its change depends in part upon the value of velocity, squared. This is a **differential equation**, which is not so easy to solve analytically.

Okay, let's write down Euler's method, applied to this problem:

$$v_{i+1} = v_i + a_i * (\text{timestep})$$

$$= v_i + \left[ -g + \frac{K}{m}v_i^2 \right] * (\text{timestep})$$

If we use Euler's method for this problem, we find that it doesn't work quite so well; again, in the plot below, blue crosses represent values derived by Euler's method, and the red line represents the exact solution:



In this case, Euler's method **overshoots** the true solution. At the start of the jump, the force of air resistance rises during the course of each second, as the jumper falls more and more rapidly. That means that the acceleration calculated with the current velocity will **overestimate** the actual acceleration during that next timestep.

## Heun's method

So, the physical quantities (velocity and position) are changing during each timestep. One way to improve our calculation is to **predict** how much they are going to change, and then use that information to **correct** our naive extrapolation of the current conditions. This idea forms the basis for **predictor-corrector** methods of integration.

There are many ways to estimate changes during the next timestep. The simplest is called **Heun's method**.

The basic idea is that, if we knew (for example)

- the current velocity of the jumper (from which we can calculate his current acceleration)

and

- the velocity of the jumper at the end of the timestep (from which we could calculate his future acceleration)

then we could take the average of the current and future accelerations, and use that to calculate his motion over the current timestep. In other words,

Let  $v_i$  = current velocity

$a_i$  = acceleration based on current velocity

$v_{i+1}$  = (poorly predicted) future velocity

$a_{i+1}$  = (poorly predicted) acceleration based on future velocity

Then a better guess at his acceleration over the next timestep is

$$a_{\text{avg}} = \frac{1}{2} (a_i + a_{i+1})$$

And then we can use this average acceleration to make an improved calculation of his motion.

Written as an algorithm, Heun's method looks like this:

LOOP:

```
% first, we calculate the current acceleration
current_acceleration = function(current_velocity, etc.)

% we use this to do a poor job of estimating future velocity
future_velocity = current_velocity + new_acceleration * timestep;

% use (poorly predicted) future velocity to calc future acceleration
future_acceleration = function(future_velocity, etc.)
```

```
% find the average acceleration
acceleration = 0.5 * (current_acceleration + future_acceleration)

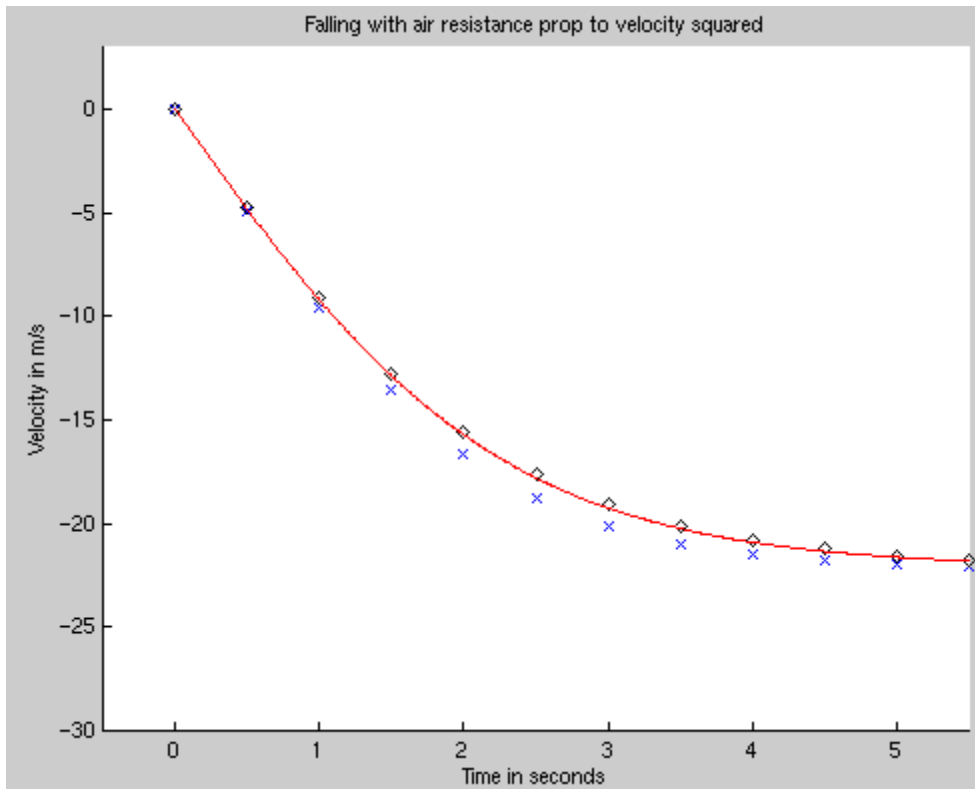
% use the average acceleration
new_velocity = current_velocity + acceleration * timestep;
```

(go back to top of loop)

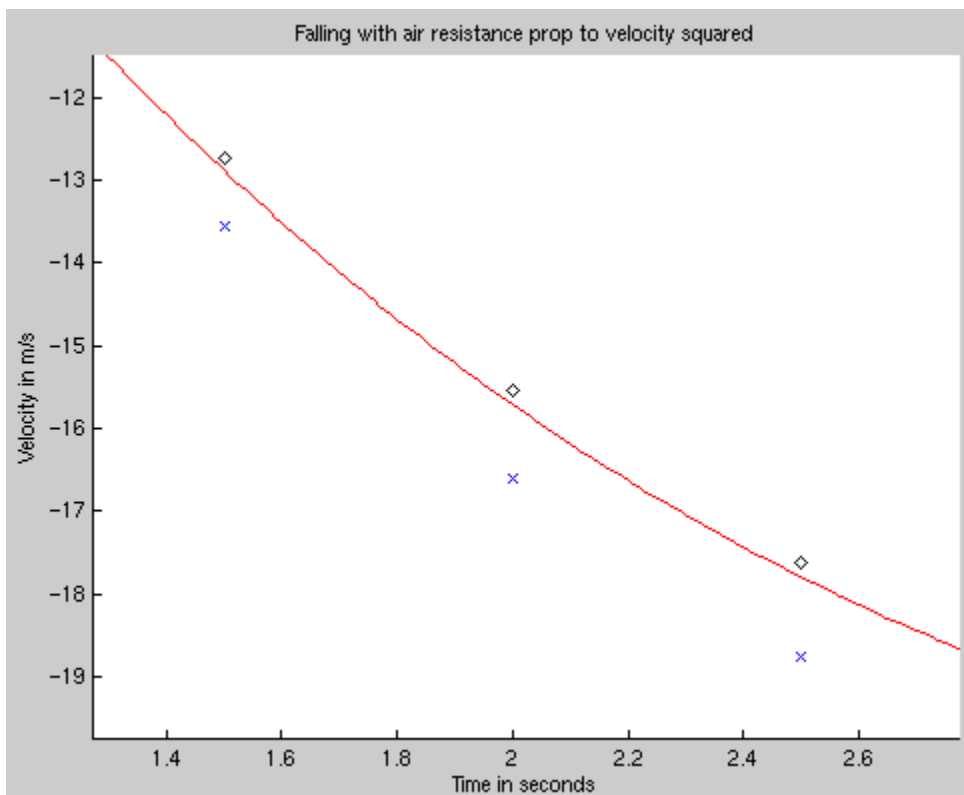
Does Heun's method do better? Yes, in two ways!

1. First, it does a better job because it anticipates changes in the velocity, and makes at least some correction for them.
2. Second, its performance increases more rapidly than Euler's method as the timestep is decreased. If we cut the timestep in half, Euler's method might yield results more accurate by a factor of two; but Heun's method might yield results more accurate by a factor of **two squared = four**. Another way to say this is, "Euler's method is correct to **first order** in the timestep, but Heun's method is correct to **second order**."

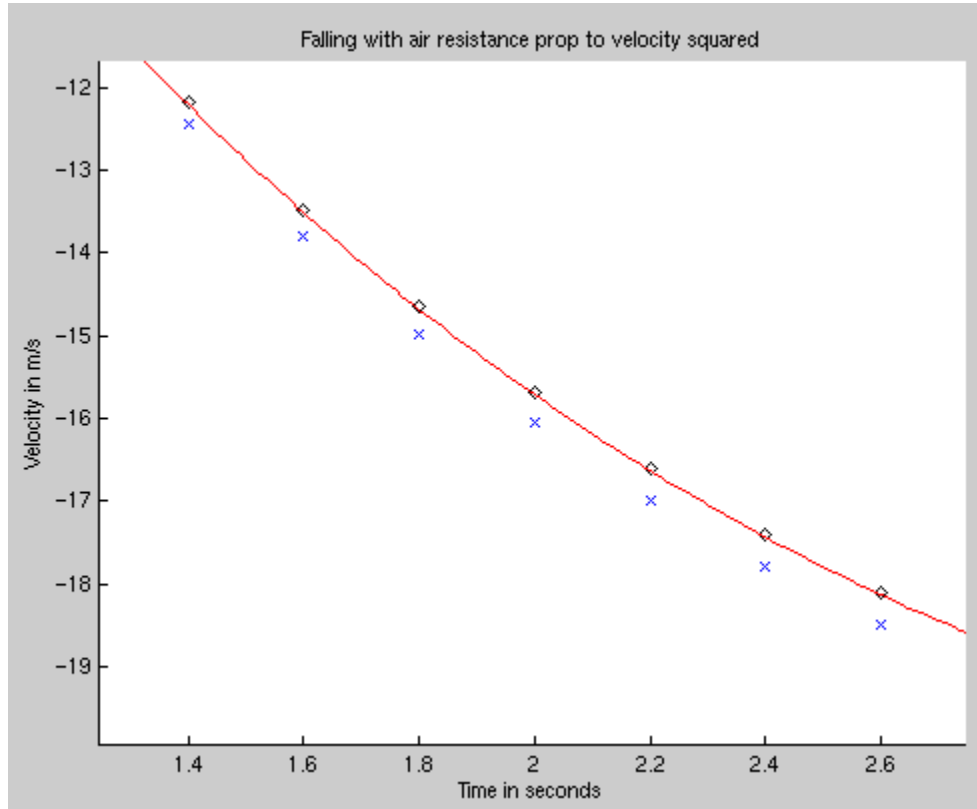
Let's see this in action. Here's a comparison of the velocities calculated via Euler's method (blue crosses) and Heun's method (black diamonds), using a timestep of 0.5 seconds.



Here's a closeup near  $t = 2$  seconds .



Now, if we decrease the timestep size from 0.5 to 0.2 seconds, both methods improve -- but by different amounts. Here's another closeup of the region around  $t = 2$  seconds; note how much closer Heun's method is to the actual velocity now.



### Heun::Euler as Trapezoid::Rectangle

The difference between Heun's method and Euler's method is analogous to the difference between the Trapezoid and Rectangle methods you used earlier.

Rectangle method assumes constant value for function within each interval      Euler's method assumes constant value for derivative within each interval

Trapezoid method assumes linear change in function within each interval      Heun's method assumes linear change for derivative within each interval



Another way to say this is to characterize the Rectangle and Euler methods as **zero-th order**, but the Trapezoid and Heun methods as **first order**. There are higher-order methods, of course: Simpson's method is second-order, and there are analogous second-order methods in the derivative domain (see Section 25.3 for an example of a second-order Runge-Kutta algorithm).

In general, higher-order methods require

- more information (higher derivatives, or values at different times)
- more complicated calculations

but, in return, they

- are more accurate, for a given timestep/interval
- converge more quickly as the timestep/interval is decreased

Source: <http://spiff.rit.edu/classes/phys317/lectures/heun/heun.html>