Deriving Drag Force in Python Computable Terms

The goal is to convert the standard quadratic drag formula into terms that can be easily used in a python program, without having to separately update velocity

The drag equation is:

$$F_D = -\frac{1}{2}\rho A C v^2 \tag{1}$$

Where:

F_D is the drag force
ρ is the density of the fluid
A is the cross sectional area
C is the drag coefficient of the object
v is the velocity

Vectorizing:

$$\vec{F_D} = -\frac{1}{2}\rho A C |\vec{v}|^2 \hat{v} \tag{2}$$

Substitution:

Knowing that

$$v = \frac{P}{m} \tag{3}$$

Where:

P is the momentum
m is the mass
v is the velocity

$$\vec{F_D} = \frac{1}{2}\rho A C \frac{|\vec{P}|}{m}^2 \hat{v} \tag{4}$$

Because momentum and velocity are in the same direction ($\vec{P} = \mathrm{m}\vec{v}$)

$$\vec{F_D} = -\frac{1}{2}\rho A C \frac{|\vec{P}|}{m}^2 \hat{P} \tag{5}$$

We now have an equation for drag force that we can use without having to calculate velocity and momentum separately.