CS201 – Introductory Computational Physics

Assignment 2

Rajdeep Pinge

ID: 201401103

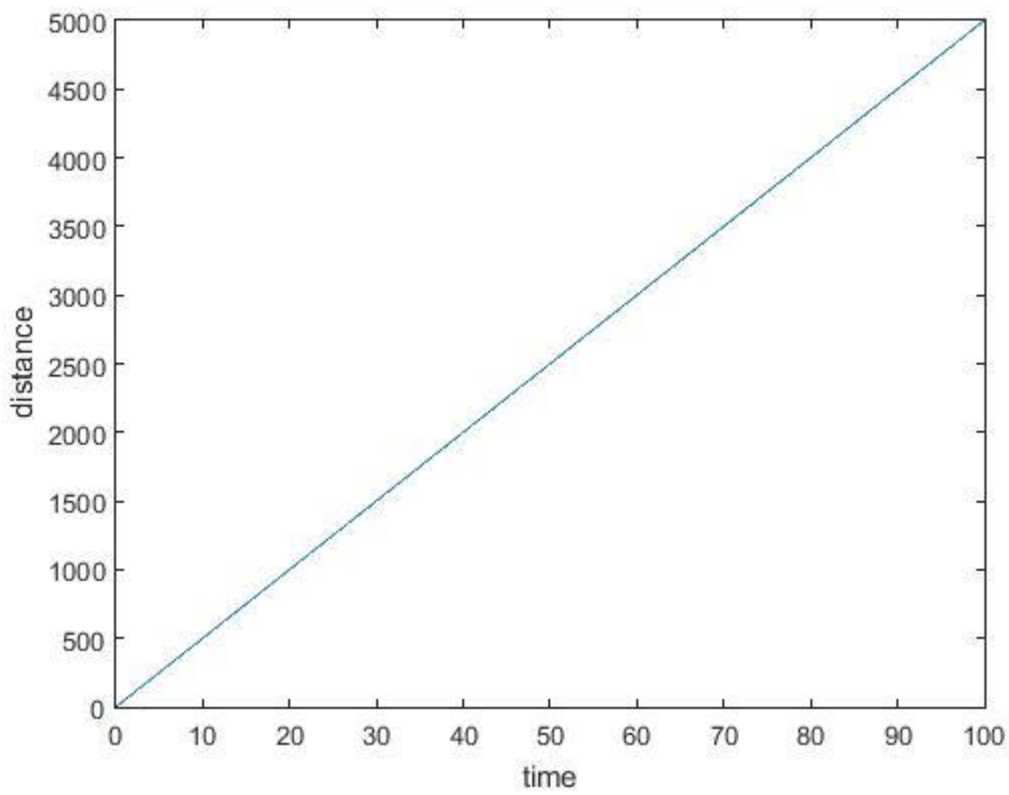
Q1. Write down the equation for position of an object moving horizontally with a constant velocity "v". Assume v=50 m/s, use the Euler method (finite difference) to solve the equation as a function of time. • Compare your computational result with the exact solution. • Compare the result for different values of the time-step.

MATLAB Code

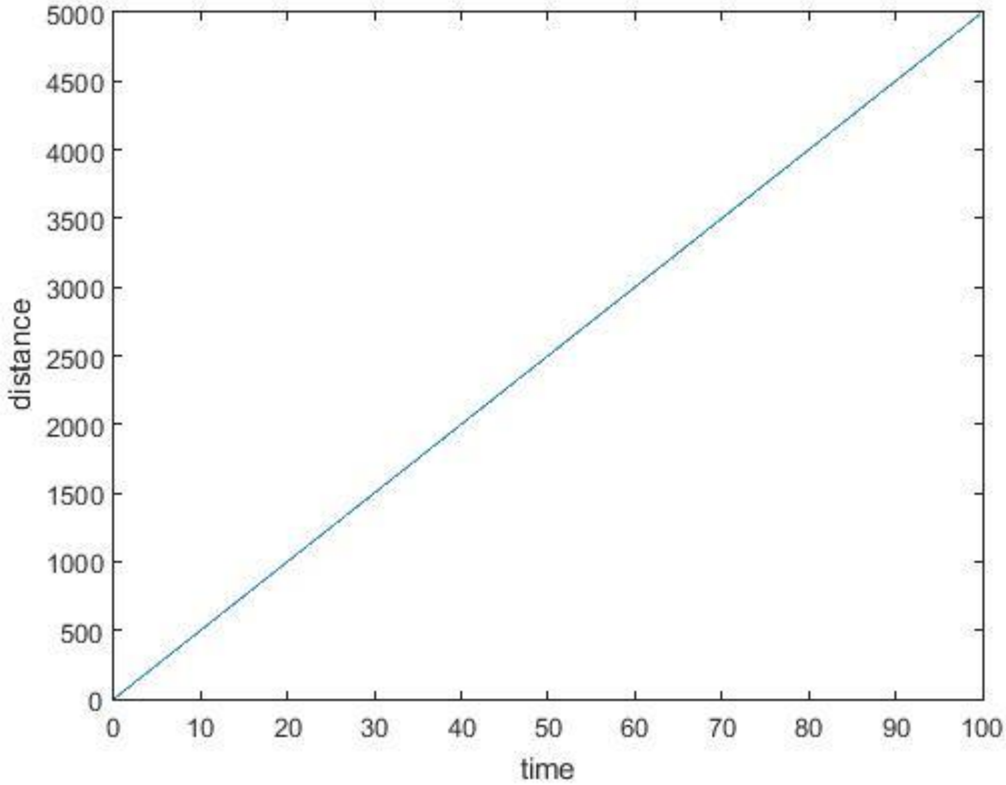
```
clear;
close all;

% define total time and time step
dt = 0.1;
total_time = 100;

% Exact Solution
time = 0:dt:total_time; %time array
velocity = 50; %m/s

% Equation for exact solution
dist = velocity * time;

% plotting the graph
plot(time, dist)
xlabel('time')
ylabel('distance')



% Finite difference method
num_ite = total_time / dt; %1000

% creating arrays
position = zeros(num_ite,1);
etime = zeros(num_ite,1);

%initial conditions
position(1) = 0;
etime(1) = 0;

% simulate using for loop
for step = 1:num_ite-1
    position(step+1) = position(step) + velocity * dt;
    etime(step+1) = etime(step) + dt;
end
```

```
% plot the graph
figure
plot(etime,position)
xlabel('time')
ylabel('distance')

% combined graph of both the solutions comparing the two
figure
plot(time, dist, '+r', etime, position)
xlabel('time')
ylabel('distance')
```

Exact solution graph:

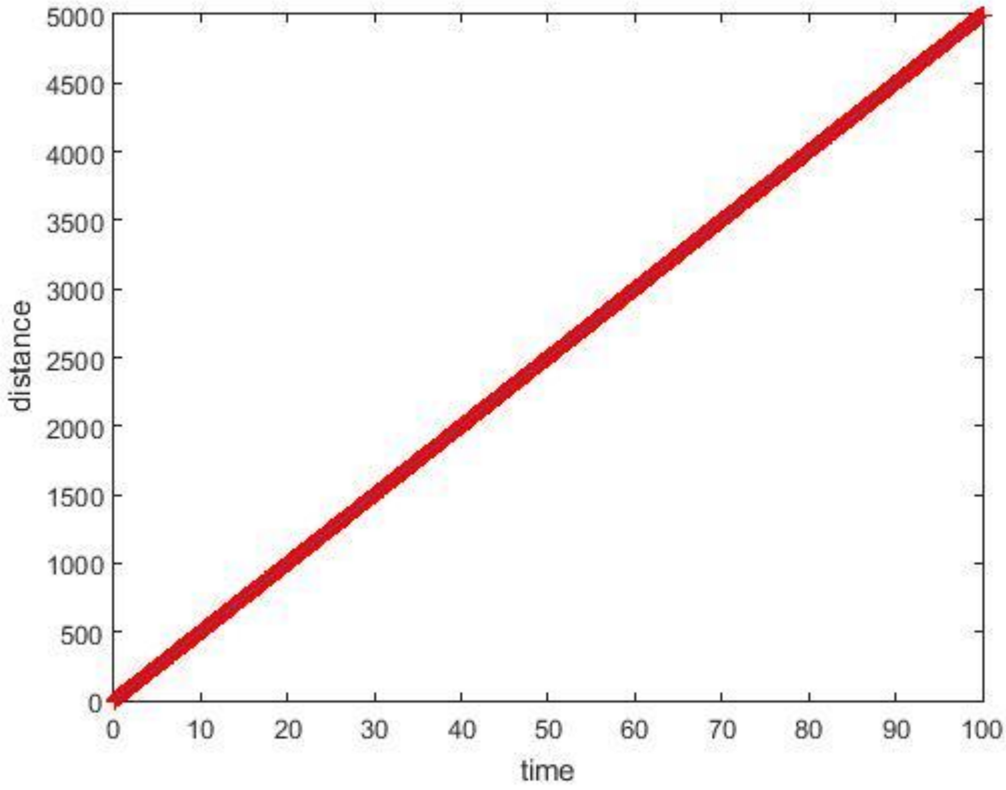


Euler – Finite Difference method solution graph:



Q. Compare your computational result with the exact solution.

Ans. Graph comparing exact solution with computational result:



Note : Here red line indicates exact solution while the thick black line indicates computational result.

Q. Compare the result for different values of the time-step.

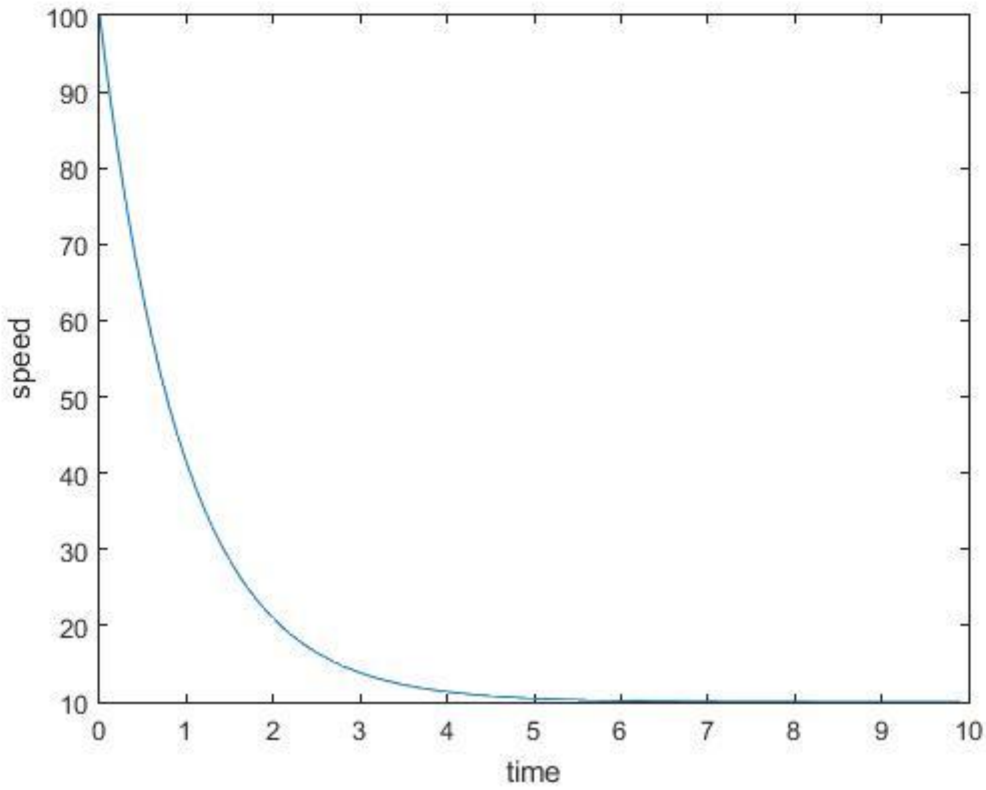
Ans. Different values of time-step give same result. This is because the graph is linear and, whichever time step we take, the points that are plotted follow the same line equation (position = 50 * time, where 50 m/s is the constant velocity) hence the line drawn is always same.

Q2. Parachute problem: frictional force on the object increases as the objects moves faster (as we learned today in the class). Role of parachute is to produce the frictional force in the form of air drag. Consider the most simple form, so the equation for velocity : $dv/dt = a - bv$ where a (from applied force), b (from friction) are constants. Use Euler's method to solve for "v" as a function of time. Choose $a=10$ and $b=1$. What is the terminal velocity in this case.

MATLAB Code

```
clear;
close all;

% declaring initial values
dt = 0.1;
total_time = 10;

init_velocity = 100; %m/s
a = 10;      %force constant (maybe gravity)
b = 1;       %friction constant because of air drag

% Finite difference method
num_ite = total_time / dt; %100

% initializing arrays
velocity = zeros(num_ite, 1);
time = zeros(num_ite, 1); %arrays made

%initial conditions
velocity(1) = init_velocity;
time(1) = 0;

% simulating step-wise
for step = 1:num_ite-1
    velocity(step+1) = velocity(step) + ( a - b * velocity(step) ) * dt;
    time(step+1) = time(step) + dt;
end

%plotting the graph
figure
plot(time, velocity)
xlabel('time')
ylabel('speed')
```

Graph of velocity w. r. t. time :



Note : Here we have taken initial velocity as 100 (m/s). This is because parachute is only opened when a free-falling person tries to reduce velocity for a safer touch-down. Here, the velocity of the person reduces to 10 (m/s).

Q. What is the terminal velocity in this case?

Ans. We find terminal velocity by solving equation, $dv/dt = 0$ i.e. $a-bv = 0 \Rightarrow v = a/b$. Here $a = 10$ and $b = 1$ Therefore, $v = 10$ (m/s), a constant value. The terminal velocity will always be the same whatever may be the starting velocity as long as the constants, a and b, are same.

Q3.Population growth problem can be modeled using a rate equation : dN/dt=aN-bN2N

number of individuals which varies with time.

First term (aN) birth of new members

Second term (bN2) corresponds to death; proportional to N2 because food will become harder to find when population becomes very large.

Use the Euler method to solve the equation as discussed in the class for the decay problem. Take a=10 and b=0; then take a=10, b=3.

Compare your numerical solution with exact solutions.

For different values of "a" and "b", give some explanations regarding your result.

MATLAB Code

```
clear;
close all;

% declaring initial values
dt = 1e-4;
total_time = 1;

etime = 0:dt:total_time; %time array

init_population = 1;
a = 10;  %increase factor
b = 3; %decrease factor

num_ite = total_time/dt; % number of iterations = 1000



% Exact solution

%let us see if formula is correct for exact solution
const = log(init_population/(a-(b*init_population)))/a;
epopulation = zeros(num_ite+1,1);

for j = 1:(num_ite+1)
    epopulation(j) = a*( exp( a*(etime(j)+const) ) )/( 1+b*exp( a*(
etime(j)+const ) ) );
end

%plotting the graph for exact solution
plot(etime,epopulation)
xlabel('time')
ylabel('population')


% Finite difference method

%initializing arrays
population = zeros(num_ite,1);
time = zeros(num_ite,1); %arrays made
```

```
%initial conditions
population(1) = init_population;
time(1) = 0;

for step=1:num_ite-1
    population(step+1) = population(step) + ( a * population(step) - b * (
population(step) * population(step) ) ) * dt;
    time(step+1) = time(step) + dt;
end

% plotting the graph for finite difference method
figure
plot(time, population)
xlabel('time')
ylabel('popualtion')

% combined graph comparing the two solutions
figure
plot(etime, epopulation, '+r', time, population)
xlabel('time')
ylabel('population')
```

Derivation of exact solutions:

$$\frac{dN}{dt} = aN - bN^2 \Rightarrow \int_{N_0}^{N} \frac{dN}{aN - bN^2} = t$$

$$\int_{N_0}^{N} \frac{dN}{N(a-bN)} = t$$

$$\frac{1}{\cancel{N(a-bN)}} = A(a-bN) + BN$$

$$1 = Aa - (Ab - B)N$$

$$Aa = 1$$

$$\boxed{A = 1/a} \qquad Ab - B = 0 \qquad \boxed{\frac{b}{a} = B}$$

$$\therefore \int_{N_0}^{N} \frac{1}{aN} + \frac{b}{a(a-bN)}\, dN = t$$

$$\int_{N_0}^{N} \frac{dN}{aN} + \frac{b}{a}\int_{N_0}^{N} \frac{dN}{a-bN} = t$$

$$\frac{1}{a}\left[\ln N\right]_{N_0}^{N} + \frac{\cancel{b}}{a}\left[\frac{\ln a-bN}{-\cancel{b}}\right]_{N_0}^{N} = t$$

$$\frac{1}{a}\left(\ln N - \ln N_0\right) - \frac{1}{a}\left(\ln a-bN - \ln a-bN_0\right) = t$$

$$\therefore \frac{1}{a}\left(\ln \frac{N}{N_0} - \ln \frac{a-bN}{a-bN_0}\right) = t$$

$$\Rightarrow \cancel{\frac{1}{a}} \ln \frac{N(a-bN_0)}{N_0(a-bN)} = at$$

$$\frac{N(a-bN_0)}{N_0(a-bN)} = e^{at}$$

$$\frac{N}{a-bN} = \frac{N_0 e^{at}}{(a-bN_0)}$$

$$Na - bNN_0 = (aN_0 - bNN_0)e^{at}$$

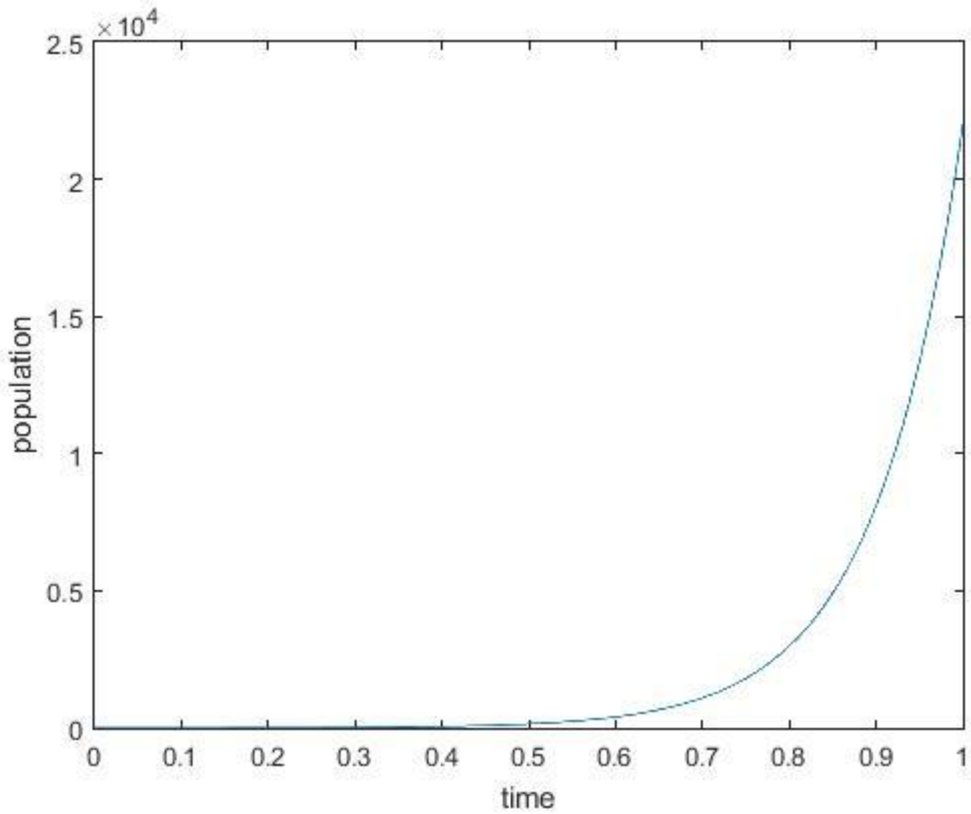
$$Na - bNN_0 = aN_0e^{at} - bNN_0e^{at}$$

$$Na - bNN_0 + bNN_0e^{at} = aN_0e^{at}$$

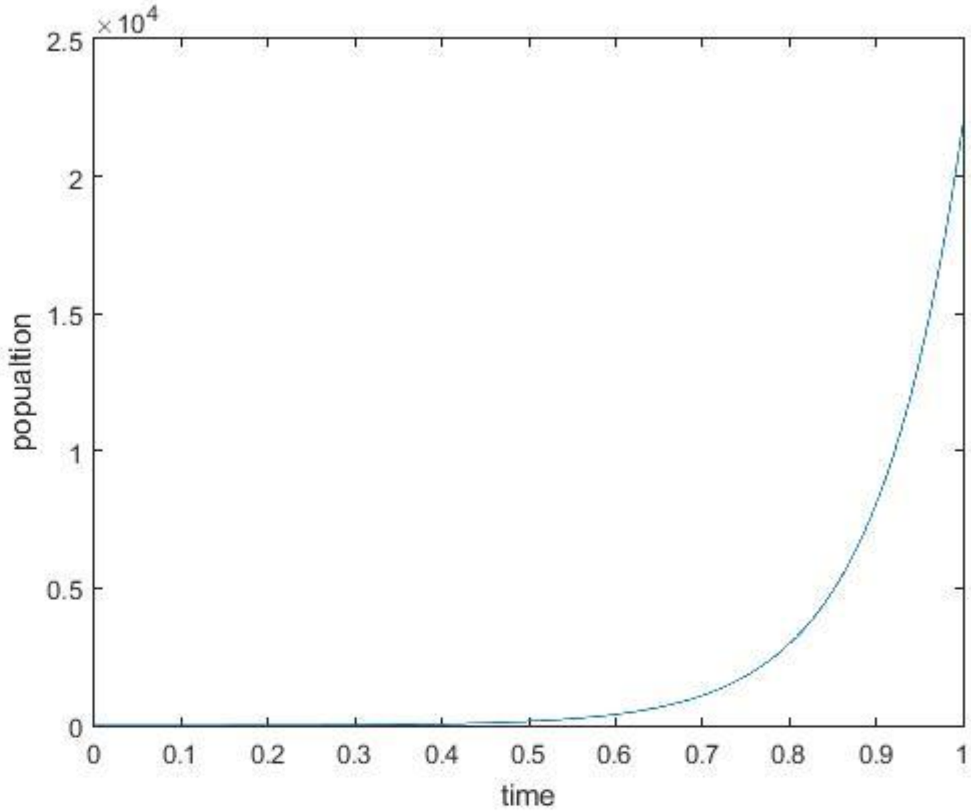
$$N(a - bN_0 + bN_0e^{at}) = aN_0e^{at}$$

$$N = \frac{aN_0 e^{at}}{a - bN_0(1 - e^{at})}$$

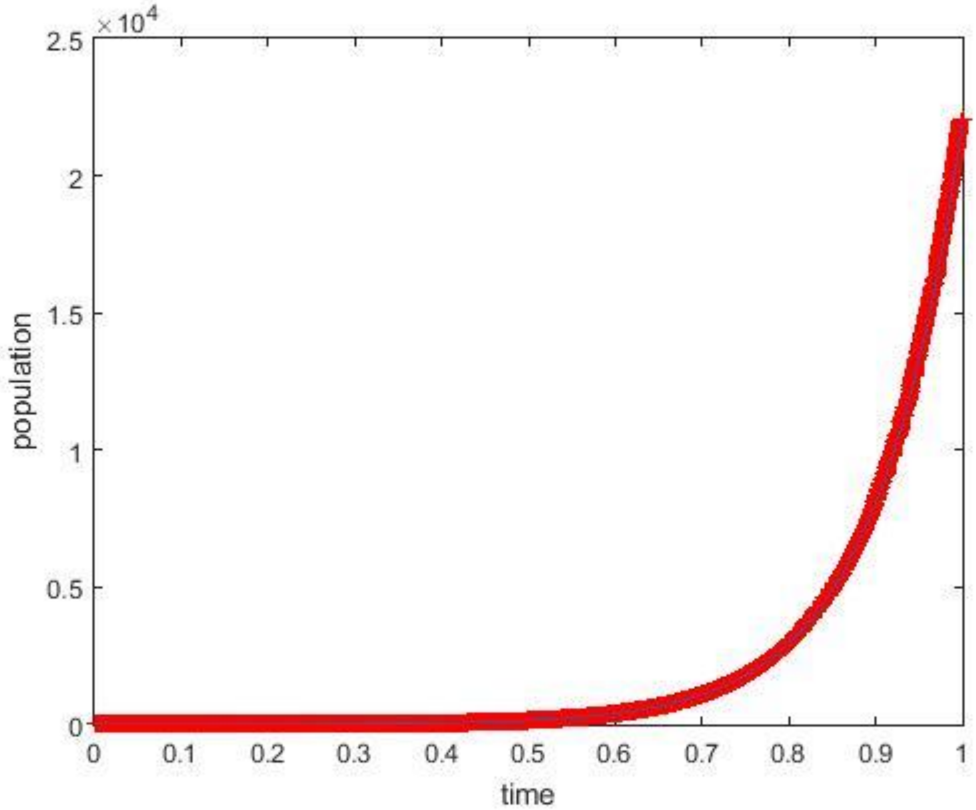
Graph of exact solution when a = 10, b = 0:



Graph of Euler's method when a = 10, b = 0:

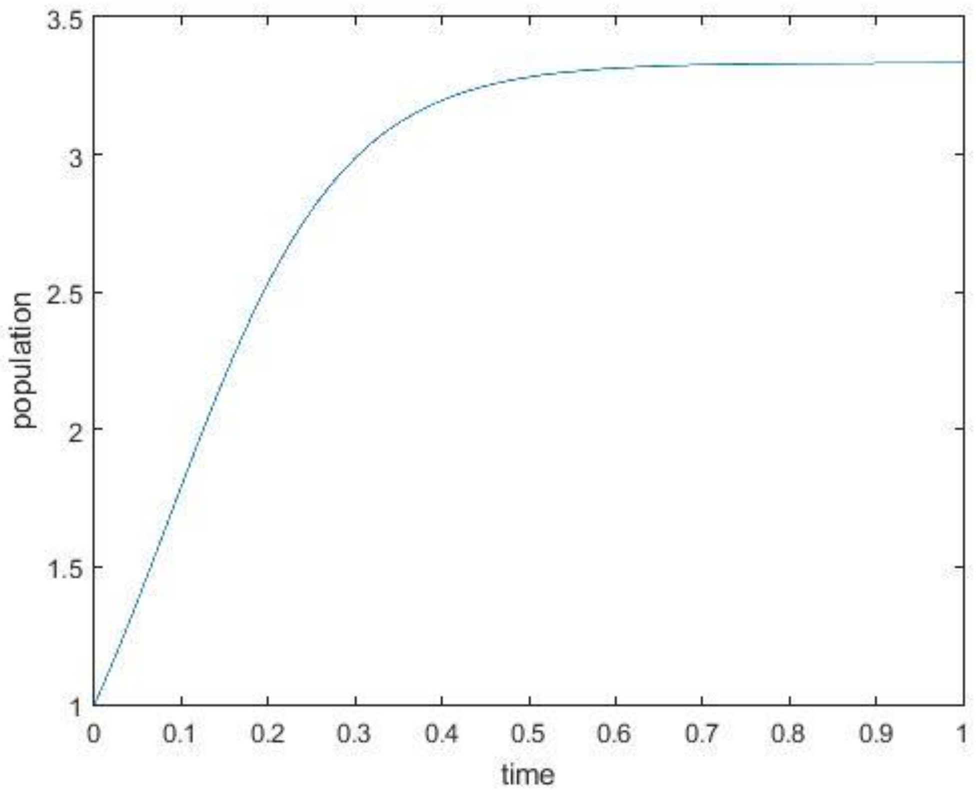


Combined graph comparing the two solutions when a = 10, b = 0:

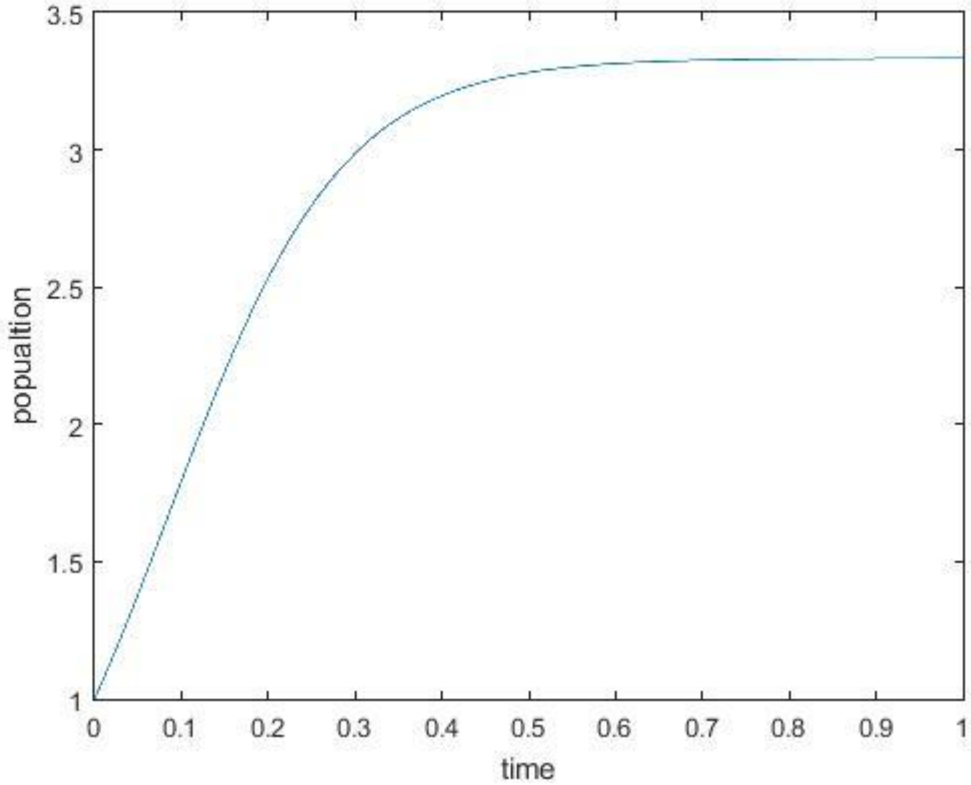


Note: Red line indicated exact solution while black thin line indicated Euler method solution.

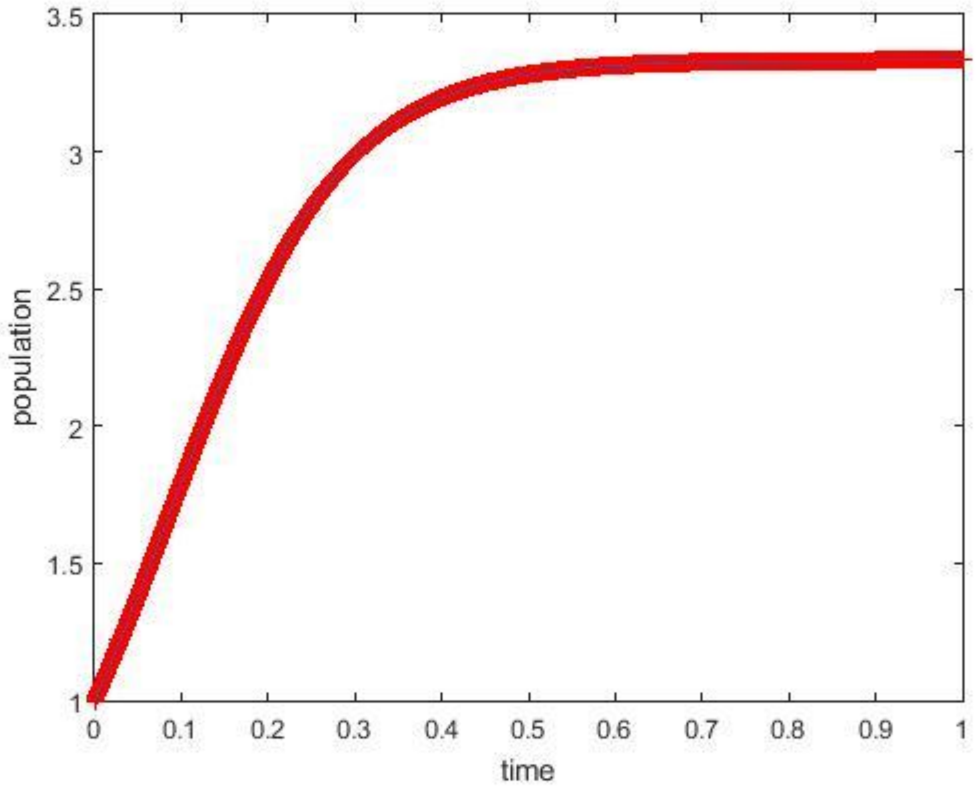
Graph of exact solution when a = 10, b = 3:



Graph of Euler’s method when a = 10, b = 3:



Combined graph comparing the two solutions when a = 10, b = 3:



Note: Red line indicates exact solution while black thin line indicated Euler method solution.

Note: Comparing the exact solution with Finite difference method, we get to know that for sufficiently small time step, both the graphs are nearly identical. But we can see the difference when the time scale, dt, becomes too large.

Q. For different values of “a” and “b”, give some explanations regarding your result.

Ans. For different values of a and b, we get different terminal velocities. This is because, to get terminal velocities, we take dN/dt = 0. i.e. a*N - b*N*N = 0 => N = a/b. Therefore, regardless of the initial population, the final population saturates to a single value given by a/b for particular a and b.

For a = 10 and b = 0, a/b -> infinity (oo) hence population constantly increases.

For finite a and b (b not equal to 0), we get a finite saturation population.

Q4. Bicycle problem: (a) Rewrite the bicycle problem/code as discussed in the class. Investigate the effect of rider's power, mass and frontal area on the ultimate velocity.

Generally for a rider in the middle of a group the effective frontal area is about 30% less than the rider at the front. How much less energy does a rider in the group expend than one at the front (assuming both moving at 12.5 m/s).

(b) Run your code (case (a) discussed during class) with initial v=0; observe the output and give possible explanation. Explain why it is important to give a non-zero initial velocity.

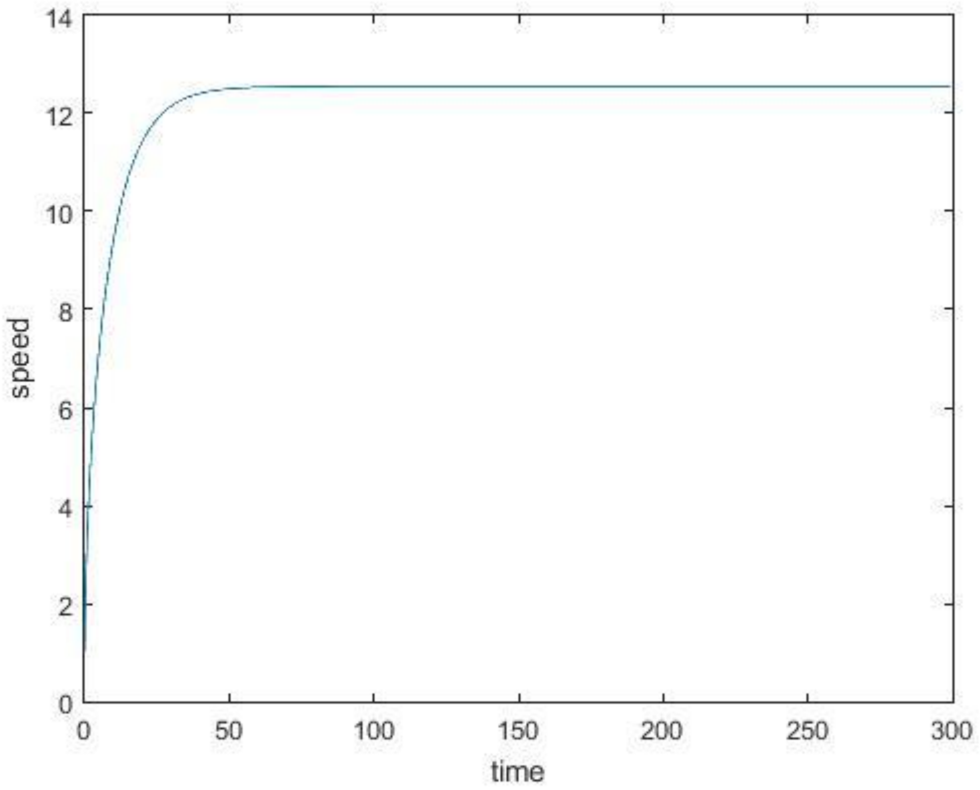
(c) As discussed in the class, we have assumed that the bicyclist maintains a constant power. What about the assumption when the bicycle has a very small velocity? (instantaneous power=product of force and velocity).

(d) At low velocities it is more realistic to assume, that the rider is able to exert a constant force. That means for small "v" there is a constant force, which means eqn is dv/dt=F0/m

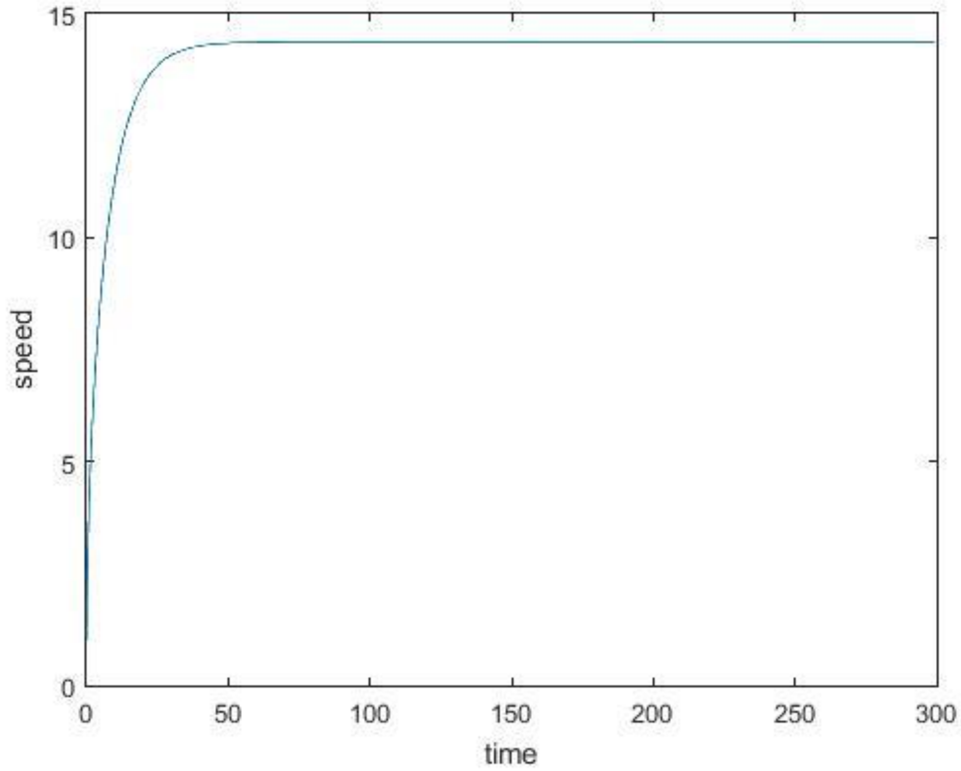
Modify your matlab code to include this term for small velocities, that means we have 2 regimes and 2 eqns one for small velocities and one for larger velocities. Make your code work automatically for both the regimes and crossover from small to large v occur when the power reaches P(=F0v) . Take F0=P/v where v=5m/s.

Change different parameters and report about important observations.

a) Original graph for initial code:

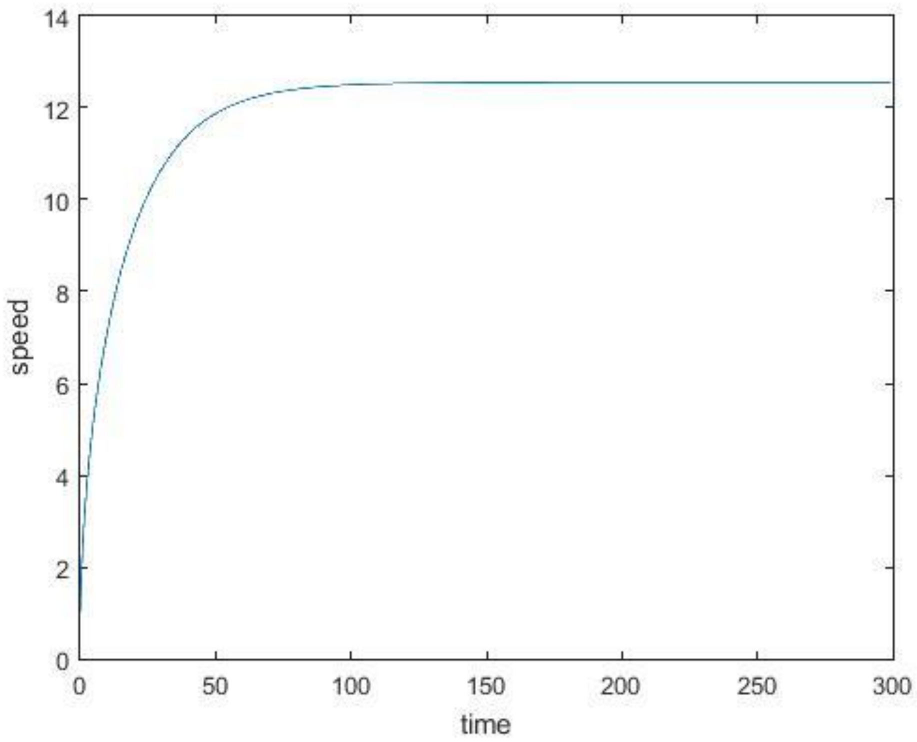


Effect of power:



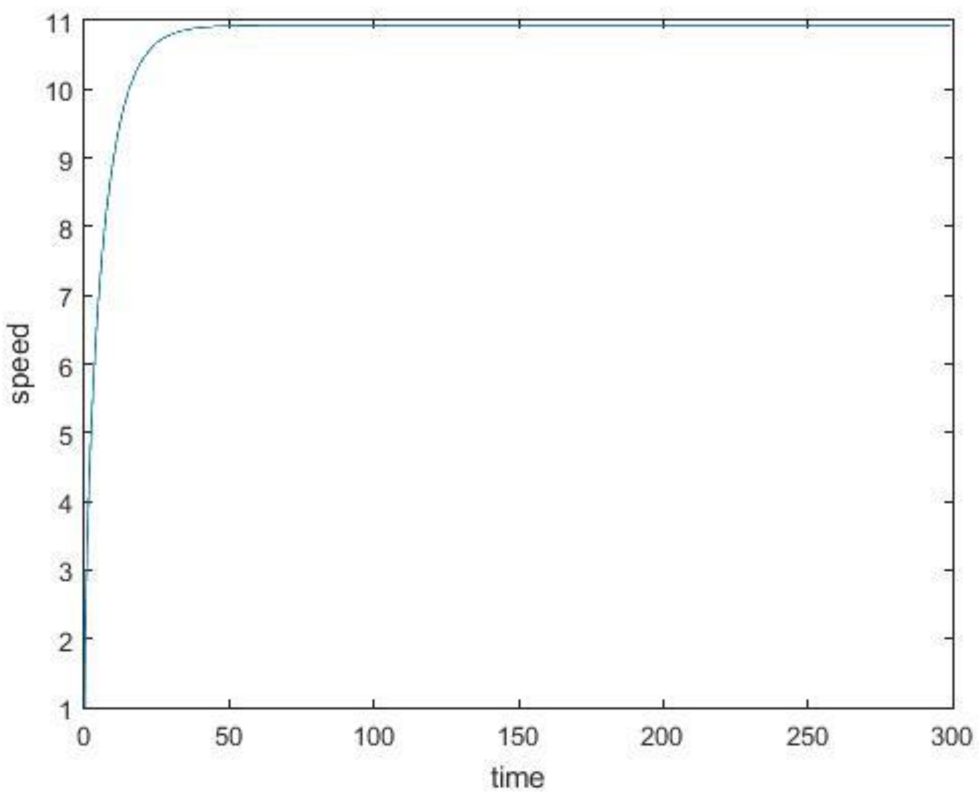
Note: Here the power is increased from 400 (W) to 600 (W). The terminal velocity has increased. Power determines at what velocity the drag will balance out rider's applied force.

Effect of mass:



Note: Here, the mass is changed from 75 kg to 150kg. It takes more time to achieve saturation velocity.

Effect of Frontal Area:



Note: As area increases, drag force increases. The applied force balances drag force. Now, when drag force increases, but the applied force is constant, therefore to keep drag force same, the velocity decreases. Time to reach terminal velocity decreases.

Energy expended by the rider in the middle of the group:

For calculating difference in energy, we would consider that the terminal velocity has been achieved $\therefore \frac{dv}{dt} = 0$ —(1)

By Newton's second law,

$$F_{net} = ma$$

$$F - F_{drag} = m\frac{dv}{dt}$$

From (1)

$$F - F_{drag} = 0$$

$$\therefore F = F_{drag}$$

$$\therefore \frac{P}{v} = -C\rho A v^2$$ [As done in class]

$$\therefore P = -C\rho A v^3$$

$$P = A(-C\rho v^3)$$

Here C, ρ, v are constants since terminal velocity is achieved and density of air is assumed constant

$$\therefore P = kA \qquad \text{where } k = -C\rho A v^3$$

Let frontal area for 1st rider $= A_0$

And Power is P_0

$$\therefore P_0 = kA_0$$

Now for the rider in the middle frontal area $A = A_0 - \frac{30}{100}A_0$

Page No.:
Date: / /

$\therefore \quad A = 0.7 A_0$

$\therefore \quad P = kA$

$P = 0.7 k A_0$

$\therefore \quad \boxed{P = 0.7 P_0}$ — (2)

Now since both the rider at the front and rider in the middle are moving in same group and hence move with same velocity ~~and~~ ~~take~~. Hence we have taken same equation.

Also since they take same amount of time,

— (3)

$E_0 = P_0 t$

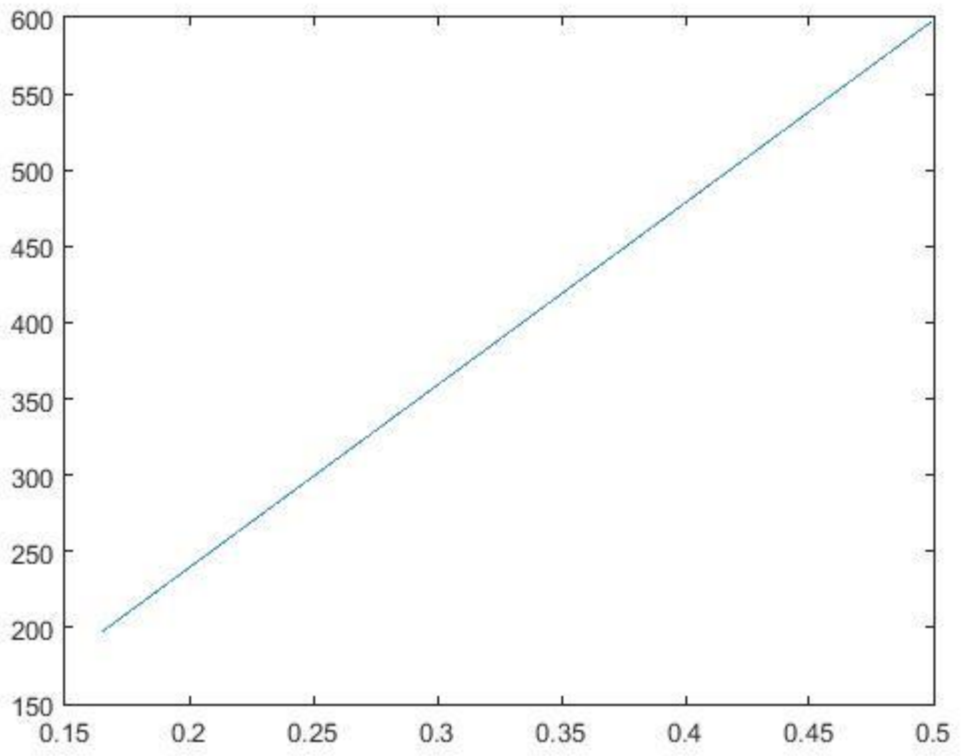
$E = P t$

$E = 0.7 P_0 t$

$\boxed{E = 0.7 E_0}$ [from (3)]

$\therefore$ Energy expended by the rider in the middle is 70% of the rider at the front that is, the energy expended is 30% less when the ~~ener~~ frontal area is 30% less.

Graph of power vs area:



Note: The graph is linear, which implies that as the area increases, the power to be applied to maintain same velocity increases. Hence riders sit at a particular position and move in groups to save power.

b) The output is blank. Since, if initial velocity is zero, the velocity at next instant cannot be determined since while calculating it, we get a ‘divide by 0’ error. It generally gives NaN(Not a Number) output.

c) P = Fv => F = P/v. Therefore if power P is constant and velocity v is very small, v -> 0, then F -> infinity (oo), which is impractical since a cyclist cannot apply infinite force. Hence Power must become very small, when v -> 0, to keep force applied within limits.

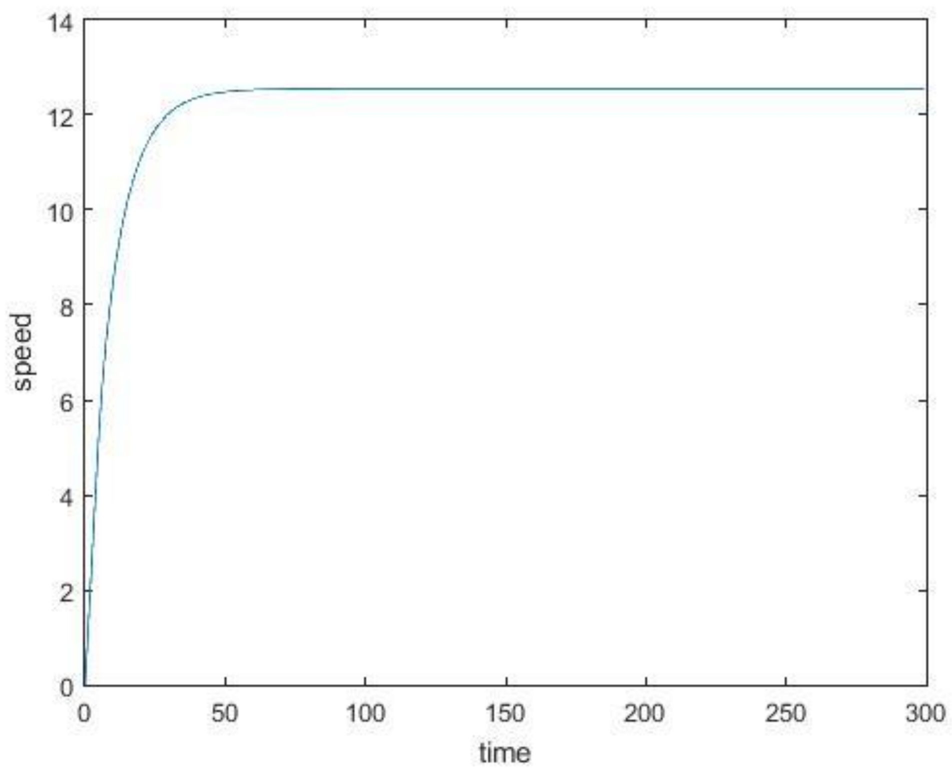
d) For small “v” there is a constant force, which means eqn is dv/dt=F0/m

Modified MATLAB Code:

```
clear
close all;

% initializing
total_time = 300; % length of simulation
init_vel = 0.01;  % initial velocity
dt = 0.1;
niter = total_time / dt; % number of iterations

% declaring arrays
```

```
time = zeros(niter, 1);
speedr = zeros(niter, 1);

% initial values
time(1) = 0;
speedr(1) = init_vel;

mass = 75;
power = 400;
const_force = power / 5;
constant = .5;
density = 1.225;
area = .33;

% for loop to simulate using finite difference method
for step=1:niter-1
    if(const_force*speedr(step) >= power)
        speedr(step+1) = speedr(step)+power*dt/ (mass*speedr(step)) - (
dt*constant*density*area*speedr(step)*speedr(step) )/(mass);
    else
        speedr(step+1) = speedr(step)+const_force*dt/mass;
    end

    time(step+1) = time(step) + dt;
end

% plotting the graph
plot(time, speedr)
xlabel('time')
ylabel('speed')
```

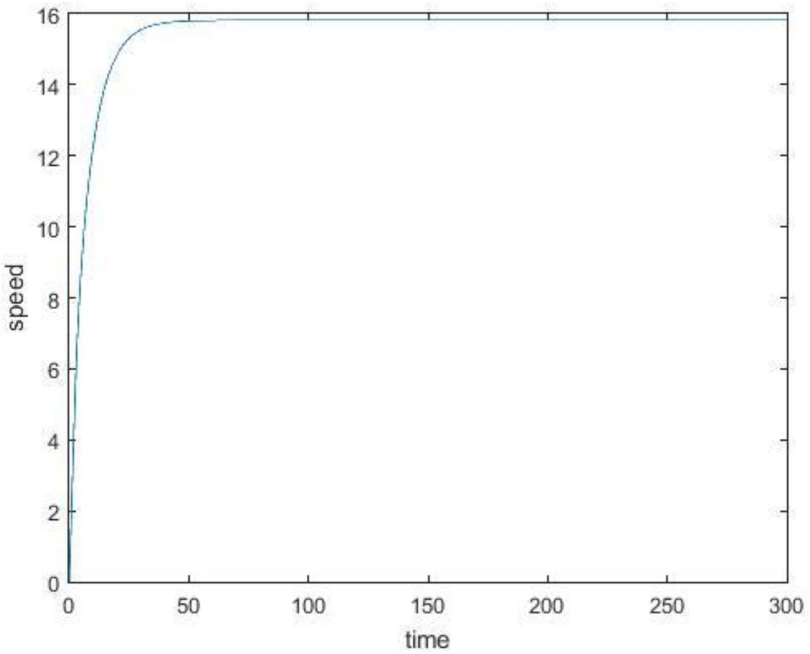
Graph for above code:



Note: while total power is less than F0 * speed(at that instant), the force remains constant and hence the graph is linear. Hence till v = 5 (m/s), graph is linear. After v = 5, the air drag becomes significant and hence the force changes. Eventually the drag force becomes equal to the applied force and the velocity becomes constant (terminal velocity).

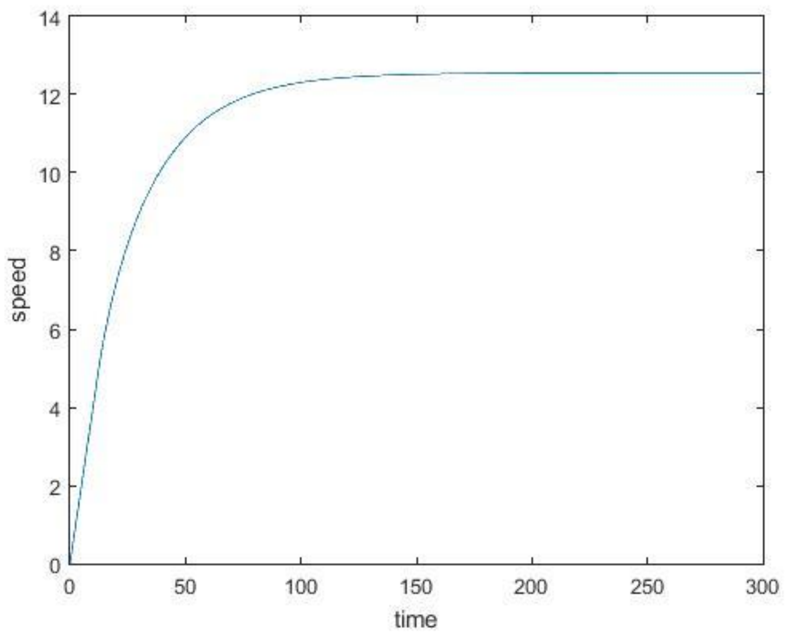
Changing different parameters

Changing power:



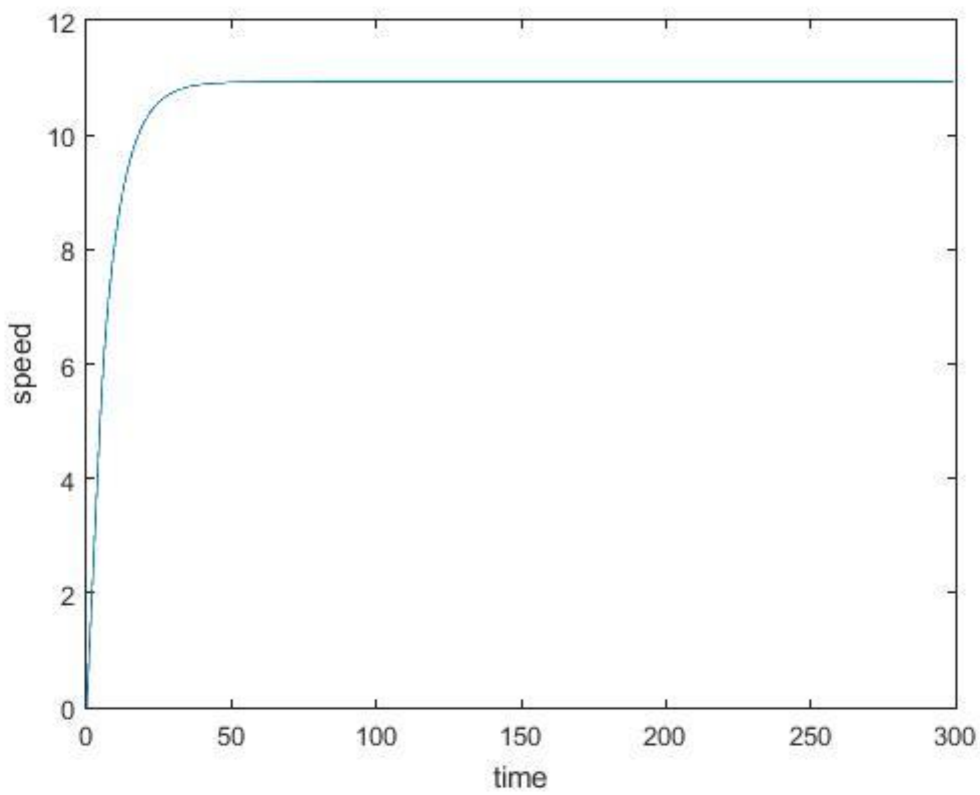
Note: Power changed from 400 to 800. Graph remains linear for small velocities but for large velocities, saturates to a higher value.

Effect of mass:



Note: The graph is linear for v <= 5 m/s. Later, when the mass is changed from 75 kg to 150kg. It takes more time to achieve saturation velocity.

Effect of changing area:



Note: As area increases, drag force increases. The applied force balances drag force. Now, when drag force increases, but the applied force is constant, therefore to keep drag force same, the velocity decreases.