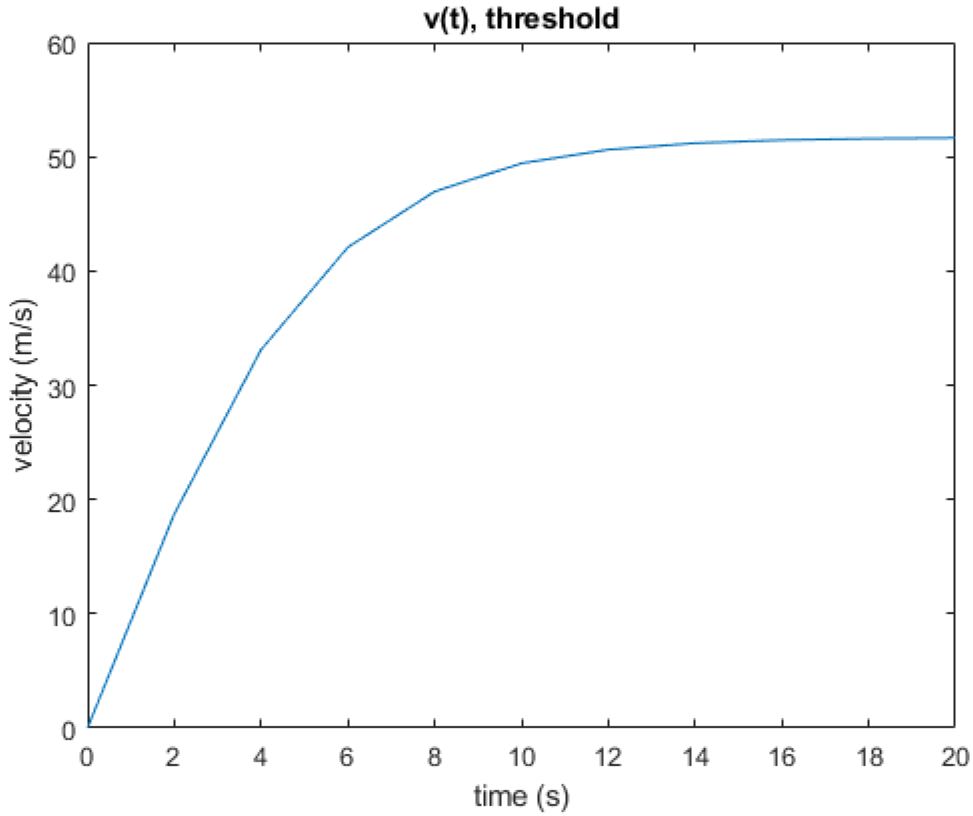
Analytical Solution to the Bungee Jumper Problem (extended)

```
% clear;

a_y_init = 9.81; % m/s^2
mass = 68.1; % kg
c_d = .25; % kg/m;
t_init = 0; % s
t_delta = 2; % s
t_current = t_init; % s
t_vector = []; % s

v_init = 0; % m/s
v_current = v_init; % m/s
v_previous = v_init; % m/s
v_threshold = .1; % m/s

v_time = []; % m/s
iteration = 1;
done = false;

v_time( iteration ) = v_init;
t_vector( iteration ) = t_init;

while (done == false)
    t_current = t_current + t_delta;
    v_current = sqrt( a_y_init * mass / c_d )* tanh( sqrt( (a_y_init *
 c_d) / mass ) * t_current );

    if ((v_current - v_previous) > v_threshold )
        % keep on calculating
    else
        done = true;
    end

    iteration = iteration + 1;
    t_vector( iteration ) = t_current;
    v_time( iteration ) = v_current;

    v_previous = v_current;
end

plot( t_vector, v_time );
title('v(t), threshold');
xlabel('time (s)');
ylabel('velocity (m/s)');
```



Published with MATLAB® R2018a