5 Appendix

Listing 1: projection_ode.m

```
function s = projection_ode(t, z, g, m, K)
  s = zeros(4, 1);
  s(1) = z(3);
  s(2) = z(4);
  s(3) = -K * sqrt(z(3)^2 + z(4)^2) * z(3) / m;
  s(4) = -g - K * sqrt(z(3)^2 + z(4)^2) * z(4) / m;
end
```

Listing 2: events_function.m

```
function [value, isTerminal, direction] = events_function(t, z, distance)
  value(1) = z(1) - distance;
  isTerminal(1) = 0;
  direction(1) = 1;

  value(2) = z(2);
  isTerminal(2) = 1;
  direction(2) = -1;
end
```

Listing 3: projection_solution.m

```
function [t, z, te, ze] = projection_solution(g, v, m, K, theta, distance)
  ode = @(t, z) projection_ode(t, z, g, m, K);
  tspan = [0, 2 * v / g];
  ic = [0, 0, v * cos(theta), v * sin(theta)];
  events_fun = @(t, z) events_function(t, z, distance);
  options = odeset('events', events_fun, 'reltol', 1e-8);
  [t, z, te, ze] = ode45(ode, tspan, ic, options);
end
```

Listing 4: distance_function.m

```
function d = distance_function(g, v, m, K, theta)
  [~, z] = projection_solution(g, v, m, K, theta, 0);
  d = z(end, 1);
end
```

Listing 5: maximum_distance.m

```
g = 9.8;       % Standard gravity (ms^-2)
v = 450;       % Initial velocity (ms^-1)
m = 6;         % Weight of ball (kg)
K = 2e-5;      % Drag constant (kgm^-1)

distance_fun = @(theta) -distance_function(g, v, m, K, theta);
[theta, distance] = fminbnd(distance_fun, eps, pi / 2);
fprintf('Firing at %.6f radians yields a maximum distance of %.2fm\n', ...
        theta, -distance);
```

Listing 6: delay_function.m

```
function d = delay_function(t, h)
  n = ceil(0.05 * (t - 0.01 * h));
  td = 0.01 * h + 20 * n - t;
  d = [max(td - 10, 0), td];
end
```

Listing 7: firing_times.m

```
g = 9.8;       % Standard gravity (ms^-2)
v = 450;       % Initial velocity (ms^-1)
m = 6;         % Weight of ball (kg)
K = 2e-5;      % Drag constant (kgm^-1)

target_distance = 15000;
distance_fun = @(t) distance_function(g, v, m, K, t);
[maxTheta, ~] = fminbnd(@(theta) -distance_fun(theta), 0, pi / 2);

distance_from_target = @(theta) distance_fun(theta) - target_distance;
thetaOne = fzero(distance_from_target, [eps, maxTheta]);
thetaTwo = fzero(distance_from_target, [maxTheta, pi / 2]);

ic_distance = 12000;
[t1, z1, te1, ze1] = projection_solution(g, v, m, K, thetaOne, ic_distance);
[t2, z2, te2, ze2] = projection_solution(g, v, m, K, thetaTwo, ic_distance);

d1 = delay_function(te1(1), ze1(1, 2));
d2 = delay_function(te2(1), ze2(1, 2));

fprintf('[theta = %.4f]: Min Delay = %.4f, Max Delay = %.4f\n', ...
        thetaOne, d1(1), d1(2));
fprintf('[theta = %.4f]: Min Delay = %.4f, Max Delay = %.4f\n', ...
        thetaTwo, d2(1), d2(2));
```

Plots

Listing 8: max_distance_plot.m

```
g = 9.8;          % Standard gravity (ms^-2)
v = 450;          % Initial velocity (ms^-1)
m = 6;            % Weight of ball (kg)
K = 2e-5;         % Drag constant (kgm^-1)

distance_fun = @(theta) -distance_function(g, v, m, K, theta);
[thetaMax, distance] = fminbnd(distance_fun, eps, pi / 2);
[t0, z0] = projection_solution(g, v, m, K, thetaMax, 0);
[t1, z1] = projection_solution(g, v, m, 0, pi / 4, 0);

plot(z0(:, 1), z0(:, 2), z1(:, 1), z1(:, 2));
xlabel('Horizontal Displacement (m)');
ylabel('Vertical Displacement (m)');

xlim([0, 1.05 * z1(end, 1)]);
ylim([0, 1.05 * max(max(z0(:, 2)), max(z1(:, 2)))]);
legend('K = 0.00002', 'K = 0');
```

Listing 9: theta_distance_plot.m

```
g = 9.8;          % Standard gravity (ms^-2)
v = 450;          % Initial velocity (ms^-1)
m = 6;            % Weight of ball (kg)
K = 2e-5;         % Drag constant (kgm^-1)

distance = @(theta) distance_function(g, v, m, K, theta);

angles = linspace(0, pi / 2, 100);
displacement = arrayfun(distance, angles);
displacement(1) = 0;

plot(angles, displacement);
title('Horizontal Displacement vs Firing Angle');
xlabel('Firing Angle (theta)');
ylabel('Horizontal Displacement (m)');
yline(15000);

% Remove exponential notation from axis tick labels
ax = gca;
ax.YAxis.Exponent = 0;
```

Listing 10: target_distance_plot.m

```
g = 9.8;       % Standard gravity (ms^-2)
v = 450;       % Initial velocity (ms^-1)
m = 6;         % Weight of ball (kg)
K = 2e-5;      % Drag constant (kgm^-1)

target_distance = 15000;
distance_fun = @(t) distance_function(g, v, m, K, t);
[maxTheta, ~] = fminbnd(@(theta) -distance_fun(theta), 0, pi / 2);

distance_from_target = @(theta) distance_fun(theta) - target_distance;
thetaOne = fzero(distance_from_target, [eps, maxTheta]);
thetaTwo = fzero(distance_from_target, [maxTheta, pi / 2]);

ic_distance = 12000;
[t1, z1, te1, ze1] = projection_solution(g, v, m, K, thetaOne, ic_distance);
[t2, z2, te2, ze2] = projection_solution(g, v, m, K, thetaTwo, ic_distance);

plot(z1(:, 1), z1(:, 2), z2(:, 1), z2(:, 2));
xlabel('Horizontal Displacement (m)');
ylabel('Vertical Displacement (m)');

xlim([0, target_distance]);
ylim([0, 1.05 * max(max(z1(:, 2)), max(z2(:, 2)))]);
legend('\theta_1', '\theta_2');
```