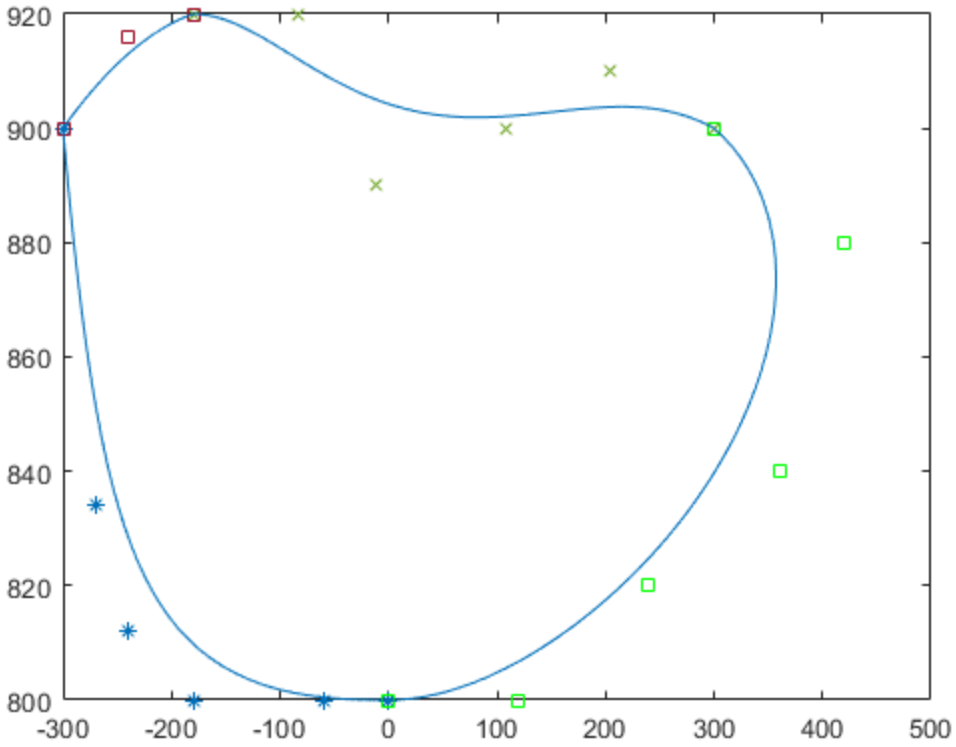
```
function bezier_plot()

    origin_x = 0.0;
    origin_y = 900.0;
    height = 100.0;
    length_step = 600.0;
    lift_height = 20;
    control_PTS_x = [origin_x - 0.5*length_step
                     origin_x - 0.45*length_step
                     origin_x - 0.4*length_step
                     origin_x - 0.3*length_step
                     origin_x - 0.1*length_step
                     origin_x
                     ];
    control_PTS_y = [origin_y
                     origin_y - 0.66*height
                     origin_y - 0.88*height
                     origin_y - 1.0*height
                     origin_y - 1.0*height
                     origin_y - 1.0*height
                     ];
    div = 1000;
    x_buff = zeros(1,div);
    y_buff = zeros(1,div);
    k = 1;
    plot(origin_x, origin_y, 'h');


    %##
    plot(control_PTS_x, control_PTS_y, '*');
    hold on;
    for t = 0.0:1/div:1.0
        [xd, yd] = bezier_generate(control_PTS_x, control_PTS_y, t);
        x_buff(k) = xd;
        y_buff(k) = yd;
        k = k + 1;
    end
    plot(x_buff, y_buff); hold on;


    %##
    control_PTS_x2 = [origin_x
                      origin_x + 0.2*length_step
                      origin_x + 0.4*length_step
                      origin_x + 0.6*length_step
                      origin_x + 0.7*length_step
                      origin_x + 0.5*length_step
                      ];
    control_PTS_y2 = [origin_y - 1.0*height
                      origin_y - 1.0*height
                      origin_y - 0.8*height
                      origin_y - 0.6*height
```

```
                origin_y - 0.2*height
                origin_y
               ];
plot(control_PTS_x2, control_PTS_y2, 'gs');
for t = 0.0:1/div:1.0
    [xd, yd] = bezier_generate(control_PTS_x2, control_PTS_y2, t);
    x_buff(k) = xd;
    y_buff(k) = yd;
    k = k + 1;
end
plot(x_buff, y_buff); hold on;

%##
control_PTS_x3 = [origin_x + 0.5*length_step
                origin_x + 0.34*length_step
                origin_x + 0.18*length_step
                origin_x - 0.02*length_step
                origin_x - 0.14*length_step
                origin_x - 0.3*length_step
                ];
control_PTS_y3 = [origin_y
                origin_y + lift_height*0.5
                origin_y + 0
                origin_y - lift_height*0.5
                origin_y + lift_height
                origin_y + lift_height
               ];
plot(control_PTS_x3, control_PTS_y3, 'x');
for t = 0.0:1/div:1.0
    [xd, yd] = bezier_generate(control_PTS_x3, control_PTS_y3, t);
    x_buff(k) = xd;
    y_buff(k) = yd;
    k = k + 1;
end
plot(x_buff, y_buff); hold on;


%##
 control_PTS_x4 = [origin_x - 0.3*length_step
                  origin_x - 0.4*length_step
                  origin_x - 0.5*length_step

               ];
control_PTS_y4 = [origin_y + lift_height
                origin_y + lift_height*0.8
                origin_y
               ];
plot(control_PTS_x4, control_PTS_y4, 's');
for t = 0.0:1/div:1.0
    [xd, yd] = bezier_generate(control_PTS_x4, control_PTS_y4, t);
    x_buff(k) = xd;
    y_buff(k) = yd;
    k = k + 1;
end
```

```
    plot(x_buff, y_buff); hold on;


end

function [ret] = dot_c(A, B)
    buff = 0;
    len = length(A);
    for i = 1:len
        buff = buff + A(i)*B(i);
    end
    ret = buff;
end

function [ret] = factorial_c(n_)
    buff = 1;
    for i = 1:n_
        buff = buff * i;
    end
    ret = buff;
end

function [ret] = comb(n, i)
    ret = factorial_c(n)/(factorial_c(n-i) * factorial_c(i));
end

function [poly_B_u]  = ploy_B_u(n, i, u)
    %poly_B_u = nchoosek(n, i) * u^i * (1 - u)^(n-i);
    %poly_B_u = factorial_c(n)/(factorial_c(n-i) * factorial_c(i))*
 u^i * (1 - u)^(n-i);
    poly_B_u = comb(n, i) * u^i * (1 - u)^(n-i);
end

function [x, y] = bezier_generate(control_pts_x, control_pts_y, u)
    n = length(control_pts_x) - 1; % the order of Bezier
    B_n = zeros(1, n + 1);
    for i = 0:n
        B_n(i + 1) = ploy_B_u(n, i, u);
    end

    x = dot_c(B_n,control_pts_x);
    y = dot_c(B_n,control_pts_y);
end
```



Published with MATLAB® R2019b