
```

% Parameters
Y_bar = 10;
epsilon_min = -5;
epsilon_max = 5;

% Different critical values of epsilon
e_h = 2.07106; % Critical value e_high
e_l = -1.7556; % Critical value e_low
e = -0.286;    % Critical value e_mid

% Different eta values for each case
eta_h = 0.2;    % eta for e_high
eta_l = 0.05;   % eta for e_low
eta = 0.1;      % eta for e_mid

% Define epsilon range
epsilon = linspace(epsilon_min, epsilon_max, 100);

% Define P0 for each case based on eta and the critical value
P0_h = eta_h * Y_bar - (1 - eta_h) * e_h;
P0_l = eta_l * Y_bar - (1 - eta_l) * e_l;
P0 = eta * Y_bar - (1 - eta) * e;

% Payment under the optimal contract for e_h and eta_h
P_h = zeros(size(epsilon));
for i = 1:length(epsilon)
    if epsilon(i) <= e_h
        P_h(i) = P0_h + epsilon(i); % For epsilon <= e_h
    else
        P_h(i) = eta_h * (Y_bar + epsilon(i)); % For epsilon > e_h
    end
end

% Payment under the optimal contract for e_l and eta_l
P_l = zeros(size(epsilon));
for i = 1:length(epsilon)
    if epsilon(i) <= e_l
        P_l(i) = P0_l + epsilon(i); % For epsilon <= e_l
    else
        P_l(i) = eta_l * (Y_bar + epsilon(i)); % For epsilon > e_l
    end
end

% Payment under the optimal contract for e_mid and eta
P = zeros(size(epsilon));
for i = 1:length(epsilon)
    if epsilon(i) <= e
        P(i) = P0 + epsilon(i); % For epsilon <= e
    else
        P(i) = eta * (Y_bar + epsilon(i)); % For epsilon > e
    end
end

```

```

% Plot the payment schedules
figure;
plot(epsilon, P_h, 'LineWidth', 2, 'DisplayName', 'P(\epsilon) for e_h, \eta_h = 0.2');
hold on;
plot(epsilon, P_l, 'LineWidth', 2, 'DisplayName', 'P(\epsilon) for e_l, \eta_l = 0.05');
plot(epsilon, P, 'LineWidth', 2, 'DisplayName', 'P(\epsilon) for e, \eta = 0.1');
xlabel('Shock (\epsilon)');
ylabel('Payment (P)');
title('Payment Schedule as a Function of \epsilon for Different Critical Values and \eta');
grid on;

xline(0, 'Color', [0.4 0.4 0.4], 'LineWidth', 2);

legend show;
hold off;

```



