HW #3

Aly Khater

$$x_{k+1} = 0.99\,x_k + 0.1\,u_k$$
$$x_0 = 0$$

Desire Final state $r_N = 10$

a) Find optimal u^*_k and x^*_k.

- final state fixed. Initial state fixed.

$$J_0 = \frac{r}{2}\sum_{k=0}^{N-1} u_k^2$$

$$u^*_k = \frac{a^2-1}{b(1-a^{2N})}\left(a^N x_0 - r_N\right)a^{N-k-1}$$

$$x_k^* = a^k x_0 + \frac{1-a^{2k}}{1-a^{2N}}\left(a^N x_0 - R_N\right)^2 \sum_{k=0}^{N-1} a^{2(N-k-1)}$$

$a = 0.99$
$b = 0.1$

$$u^*_k = \frac{0.99^2-1}{0.1(1-0.99^{2N})}\left(0.99^N(0) - 10\right)0.99^{N-k-1}$$

$$u_k^* = \frac{(0.19701)^{N-k-1}}{(0.1 - 0.099^{2N})}$$

$$x_k^* = \cancel{0.99^k(0)}^{\,0} + \frac{1-0.99^{2k}}{1-0.99^{2N}}\left(0.99^N(0) - 10\right)^2 \sum_{k=0}^{N-1} 0.99^{2(N-k-1)}$$

$$x_k^* = \frac{1-0.99^{2k}}{1-0.99^{2N}}(100)\sum_{k=0}^{N-1} 0.99^{2(N-k-1)}$$

, fixed initial state ; free final state

$$J_0 = \frac{1}{2}(x_N - r_N)^2 + \frac{r}{2}\sum_{k=0}^{N-1} u_k^2$$

$a = 0.99$
$b = 0.1$

$$u_k^* = -\frac{b}{r(1+\beta)}(a^N x_0 - R_N)a^{N-k-1}; \quad \beta = \frac{b^2}{r}\frac{(1-a^{2N})}{(1-a^2)}$$

$$x_k^* = \frac{a^k x_0 + \beta r_N}{1+\beta}$$

$$\beta = \frac{(0.1)^2}{r}\frac{(1-0.99^{2N})}{(1-a^2)} = \frac{0.01(1-0.99^{2N})}{r(1-a^2)}$$

$$u_k^* = -\frac{0.1}{r(1+\beta)}(-10)\,0.99^{N-k-1} = \frac{1}{r(1+\beta)}\,0.99^{N-k-1}$$

$$u_k^* = \frac{0.99^{N-k-1}}{r\left(1+\frac{0.01(1-0.99^{2N})}{r(1-0.99^2)}\right)} = \frac{0.99^{N-k-1}}{r+\frac{0.01(1-0.99^{2N})}{0.0199}}$$

$$u_k^* = \frac{0.0199(0.99^{N-k-1})}{0.0199r + 0.01(1-0.99^{2N})}$$

$$x_t^* = \frac{0 + \frac{b^2}{r}\frac{(1-\alpha^{2N})}{(1-\alpha^2)}(-10)}{1 + \frac{b^2}{r}\frac{(1-\alpha^{2N})}{(1-\alpha^2)}}$$ Multiply by $\frac{r(1-\alpha^2)}{r(1-\alpha^2)}$

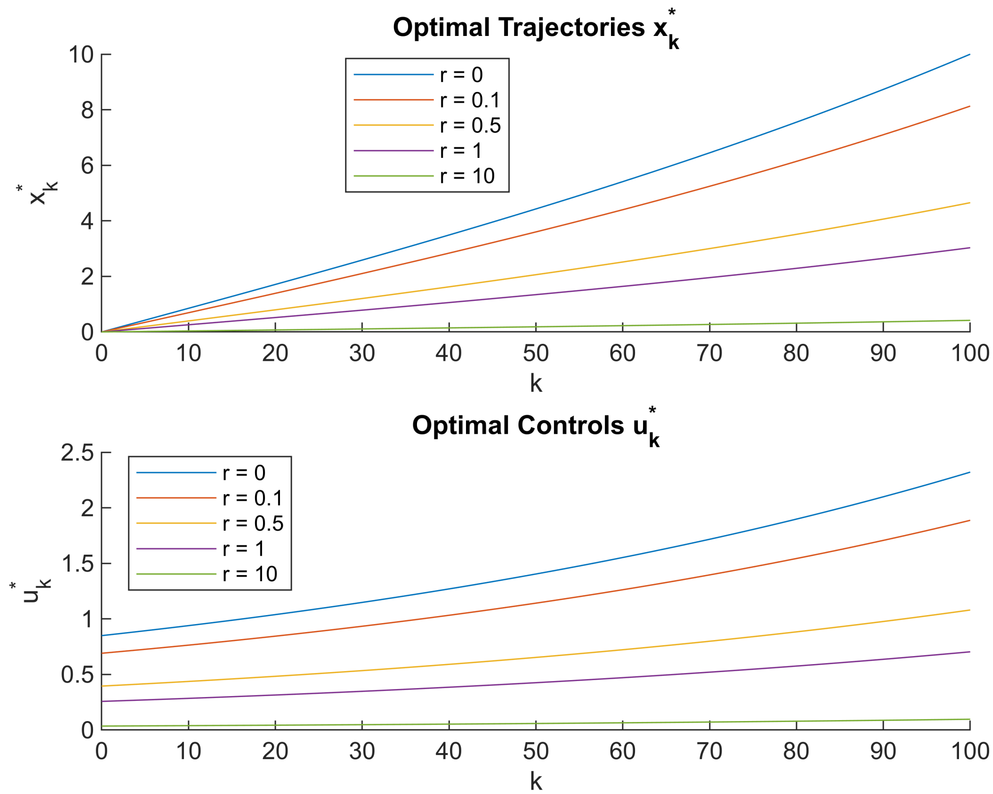
$$= \frac{-10b^2(1-\alpha^{2N})}{r(1-\alpha^2) + b^2(1-\alpha^{2N})}$$

$$= \frac{-0.1(1-0.99^{2N})}{r(1-0.99^2) + 0.1^2(1-0.99^{2N})} = \frac{-0.1(1-0.99^{2N})}{0.0199r + 0.01(1-0.99^{2N})}$$

$$x_k^* = -\frac{(1-0.99^{2N})}{0.199r + 0.1(1-0.99^{2N})}$$

```matlab
%Aly Khater
% Plot optimal control u*_k and trajectory x*_k for various r
clear;
clc;
close all;

% parameters
a   = 0.99;
b   = 0.1;
N   = 100;
x0  = 0;      % initial state
rN  = 10;     % desired final state
r_vals = [0.1, 0.5, 1, 10];

figure;
subplot(2,1,1);
hold on;
title('Optimal Trajectories x^*_k');
xlabel('k');
ylabel('x^*_k');

subplot(2,1,2);
hold on;
title('Optimal Controls u^*_k');
xlabel('k');
ylabel('u^*_k');

% Fixed initial state. r = 0
[x,u] = scoptco_fixed(a, b, 0, N, x0, rN);

k = 0:N;  % time

subplot(2,1,1);
plot(k, x, 'DisplayName',['r = ' '0']);

subplot(2,1,2);
plot(k, u, 'DisplayName',['r = ' '0']);

% Loop over r values for free state
for r = r_vals
    [x, u] = scoptco_free(a, b, r, N, x0, rN);


    subplot(2,1,1);
    plot(k, x, 'DisplayName',['r = ' num2str(r)]);

    subplot(2,1,2);
    plot(k, u, 'DisplayName',['r = ' num2str(r)]);
end
```

```
subplot(2,1,1);
legend('Location','best');
subplot(2,1,2);
legend('Location','best');
```




```
%Functions taken from the optimal control book
```

```
function [x, u] =scoptco_fixed (a, b, r, N, x0, rN)
% Simulation of Optimal Control for Scalar Systems
% Fixed Final State Case
x(1) =x0;
alam= (1-a ^ (2*N)) / (1-a ^ 2); alam=alam*b ^ 2;
u(1) =b*(rN-x(1)*a ^ N)*a ^ N/ (alam);
u(1) =u(1) / a;
for k=1:N
% Update the Plant State
x(k+1)=a*x(k) +b*u(k);
% Update the Optimal Control Input
u(k+1) =u(k) /a;
end
end
```

```
function [x, u]=scoptco_free (a, b, r, N, x0, rN)
% Simulation of Optimal Control for Scalar Systems
% Free Final State Case
x(1) =x0;
alam=(1-a ^ (2*N)) / (1-a ^ 2); alam=alam*b ^ 2/r;
u(1) =b*(rN-x(1)*a ^ N)*a ^ N/ (r*(alam+1));
u(1) =u(1) /a;
for k=1:N
% Update the Plant State
x(k+1) =a*x(k) +b*u(k);
% Update the Optimal Control Input
u(k+1) =u(k) /a;
end
end
```

c)

As r increases, you can see that uk* increases in value, requiring more energy to reach the steady state. In the plot of xk* vs k, we notice that Xn is slowly reaching our desired final state value of 10. These plots show that we can choose/control our r value to reach our desired final state for our control system.