1(a)

```
x0=[1;-1];

N=50;

A=[0.77,-0.35;0.49,0.91];

B=[0.04;0.15];

Q=[500,0;0,100];

R=1;

P=[1500,0;0,100];

[K,P] = lqrBatch(A,B,Q,R,P,N);

U0opt=K*x0

J0opt=x0'*P*x0
```

```
U0opt =


    4.0818
   -3.8039
   -3.0017
   -1.6477
   -0.8106
   -0.3813
   -0.1757
   -0.0801
   -0.0364
```

-0.0165

-0.0075

-0.0034

-0.0015

-0.0007

-0.0003

-0.0001

-0.0001

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

-0.0000

```
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
```

```
J0opt =

   1.8723e+03
```

1(b)

```
x0=[1;-1];
N=50;
A=[0.77,-0.35;0.49,0.91];
B=[0.04;0.15];
Q=[500,0;0,100];
R=1;
P=[1500,0;0,100];
nx = size(A,1);
nu = size(B,2);
```

```
Sx = zeros(nx*(N+1),nx);

Su = zeros(nx*(N+1),nu*N);

Sx(1:nx,:) = eye(nx);


for i=1:N

    Sx(nx*i+1:nx*(i+1),:) = A*Sx(nx*(i-1)+1:nx*i,:);

    Su(nx*i+1:nx*(i+1),1:i*nu) = [A*Su(nx*(i-1)+1:nx*i,1:(i-1)*nu) B];

end

Qbar  = blkdiag(kron(eye(N),Q),P);

Rbar = kron(eye(N),R);

H=Su'*Qbar*Su+Rbar;

F=Sx'*Qbar*Su;

u=sdpvar(N,1);

x=sdpvar(2,1);

C=x==x0;

obj=u'*H*u+2*x'*F*u+x'*Sx'*Qbar*Sx*x;

Options = sdpsettings('solver','quadprog');

out=optimize(C,obj,Options);

double(u)

double(obj)
```

```
Minimum found that satisfies the constraints.


Optimization completed because the objective function is non-decreasing in

feasible directions, to within the default value of the optimality tolerance,

and constraints are satisfied to within the selected value of the constraint tolerance.
```

```
ans =

    4.0818
   -3.8039
   -3.0017
   -1.6477
   -0.8106
   -0.3813
   -0.1757
   -0.0801
   -0.0364
   -0.0165
   -0.0075
   -0.0034
   -0.0015
   -0.0007
   -0.0003
   -0.0001
   -0.0001
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
```

```
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
   -0.0000
    0.0000
   -0.0000
    0.0000

ans =

   1.8723e+03
```

1(c)

```
x0=[1;-1];

N=50;

A=[0.77,-0.35;0.49,0.91];

B=[0.04;0.15];

Q=[500,0;0,100];

R=1;

nx = size(A,1);

nu = size(B,2);

P = zeros(nx,nx,N+1);

PN=[1500,0;0,100];

P(:,:,N+1) = PN;

F=zeros(1,2,N);

for i=N:-1:1

    F(:,:,i)= -inv(R+B'*P(:,:,i+1)*B)*B'*P(:,:,i+1)*A;


    P(:,:,i)            =            Q            +            A'*P(:,:,i+1)*A            -

A'*P(:,:,i+1)*B*inv(R+B'*P(:,:,i+1)*B)*B'*P(:,:,i+1)*A;


end

Jopt=x0'*P(:,:,1)*x0;
```

1(d)

```
%%Batch

D=[0.1;0.1];

N=50;

w=(10^0.5)*randn(1,51);
```

```
w(1)=0;

D=[0;0];

for i=1:N

    D=[D;0.1*w(i+1);0.1*w(i+1)];

end



x0=[1;-1];


A=[0.77,-0.35;0.49,0.91];

B=[0.04;0.15];

Q=[500,0;0,100];

R=1;

P=[1500,0;0,100];

nx = size(A,1);

nu = size(B,2);

Sx = zeros(nx*(N+1),nx);

Su = zeros(nx*(N+1),nu*N);

Sx(1:nx,:) = eye(nx);


for i=1:N

    Sx(nx*i+1:nx*(i+1),:) = A*Sx(nx*(i-1)+1:nx*i,:);

    Su(nx*i+1:nx*(i+1),1:i*nu) = [A*Su(nx*(i-1)+1:nx*i,1:(i-1)*nu) B];

end

Qbar  = blkdiag(kron(eye(N),Q),P);

Rbar = kron(eye(N),R);

H=Su'*Qbar*Su+Rbar;

F=Sx'*Qbar*Su;

u=sdpvar(N,1);
```

```
x=sdpvar(2,1);

C=x==x0;

obj=u'*H*u+2*(x'*F+D'*Qbar*Su)*u+x'*Sx'*Qbar*Sx*x;

Options = sdpsettings('solver','quadprog');

out=optimize(C,obj,Options);

double(u)


%%recursive

x0=[1;-1];

N=50;

A=[0.77,-0.35;0.49,0.91];

B=[0.04;0.15];

Q=[500,0;0,100];

R=1;

P = zeros(nx,nx,N+1);

PN=[1500,0;0,100];

P(:,:,N+1) = PN;

F=zeros(1,2,N);

for i=N:-1:1

    F(:,:,i)= -inv(R+B'*P(:,:,i+1)*B)*(B'*P(:,:,i+1)*A);


    P(:,:,i)            =            Q            +            A'*P(:,:,i+1)*A            -

A'*P(:,:,i+1)*B*inv(R+B'*P(:,:,i+1)*B)*B'*P(:,:,i+1)*A;


end

U=[F(:,:,1)*x0];

for i=2:N

    x=A*x0+B*U(i-1)+D(2*i-1:2*i);

    U=[U;F(:,:,1)*x+D(2*i+1:2*i+2)'*P(:,:,i+1)*B];
```

```
end

double(U)
```

```
Minimum found that satisfies the constraints.


Optimization completed because the objective function is non-decreasing in

feasible directions, to within the default value of the optimality tolerance,

and constraints are satisfied to within the selected value of the constraint tolerance.
```

```
ans =


   -0.1520

    4.1791

   -8.6397

   -1.4028

    4.1382

    1.0537

    0.6093

   -6.3415

    0.8636

    9.6288

   -8.1180

    3.1717

    2.1916

   -1.1402

    2.6654
```

 1.5392

-2.4292

-0.1380

-0.4499

 1.5359

 5.2295

-2.4661

-2.0813

 2.2037

-1.1616

-0.1299

 1.6485

-1.5569

 1.2879

-4.6596

 2.5015

-0.4816

-0.4444

 5.5654

-8.1759

 1.0148

 2.3036

-4.7355

 5.8580

-2.2144

 0.0568

-1.3435

-0.3430

 2.8192

-0.6572

```
    0.9679
   -1.1530
   -1.6619
   -1.0824
    4.3621


ans =


    4.0818
  -38.4260
   55.9916
  -53.7784
   35.9924
  -38.1029
   44.0772
    2.6121
   25.0541
  -49.8135
   93.7842
  -91.8465
   89.5133
  -79.7046
   75.5092
  -76.4168
   96.8048
  -81.8325
   97.3039
  -91.9014
   74.5682
```

-61.7530

 81.8103

-79.3583

 92.1456

-85.0163

 79.4924

-74.6196

 63.8036

-49.7994

 32.6617

-43.5879

 35.9687

-72.5803

 99.5751

-98.8109

 88.3667

-68.0962

 42.2646

-38.9543

 36.3465

-31.2547

 34.8825

-46.8869

 49.0549

-50.9292

 60.0175

-47.1638

 56.8344

-79.9610

2(a)

```
tic

x0=[-1;-1];

N=3;

A=[1,1;0,1];

B=[0;1];

Q=eye(2);

R=0.1;

P=eye(2);

nx = size(A,1);

nu = size(B,2);

Sx = zeros(nx*(N+1),nx);

Su = zeros(nx*(N+1),nu*N);

Sx(1:nx,:) = eye(nx);


u=sdpvar(N,1);

x=sdpvar(2,N+1);

C=[x(:,1)==x0,abs(u)<=1];

for i=1:N

    Sx(nx*i+1:nx*(i+1),:) = A*Sx(nx*(i-1)+1:nx*i,:);

    Su(nx*i+1:nx*(i+1),1:i*nu) = [A*Su(nx*(i-1)+1:nx*i,1:(i-1)*nu) B];

    C=[C,abs(x(:,i+1))<=15];

end

Qbar  = blkdiag(kron(eye(N),Q),P);

Rbar = kron(eye(N),R);

H=Su'*Qbar*Su+Rbar;

F=Sx'*Qbar*Su;
```

```
obj=u'*H*u+2*x(:,1)'*F*u+x(:,1)'*Sx'*Qbar*Sx*x(:,1);
Options = sdpsettings('solver','quadprog');
out=optimize(C,obj,Options);
double(u)
double(obj)
toc
```

```
Minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the default value of the optimality tolerance,
and constraints are satisfied to within the selected value of the constraint tolerance.



ans =

    1.0000
    0.9129
   -0.8299


ans =

   12.2743

Elapsed time is 1.469170 seconds.
```

2(b)

```
tic

x0=[-1;-1];

N=3;

A=[1,1;0,1];

B=[0;1];

Q=eye(2);

R=0.1;

P=eye(2);

nx = size(A,1);

nu = size(B,2);

Sx = zeros(nx*(N+1),nx);

Su = zeros(nx*(N+1),nu*N);

Sx(1:nx,:) = eye(nx);


u=sdpvar(N,1);

x=sdpvar(2,N+1);

C=[x(:,1)==x0,abs(u)<=1];

for i=1:N

    Sx(nx*i+1:nx*(i+1),:) = A*Sx(nx*(i-1)+1:nx*i,:);

    Su(nx*i+1:nx*(i+1),1:i*nu) = [A*Su(nx*(i-1)+1:nx*i,1:(i-1)*nu) B];

    C=[C,x(:,i+1)==A*x(:,i)+B*u(i),abs(x(:,i+1))<=15];

end

Qbar  = blkdiag(kron(eye(N),Q),P);

Rbar = kron(eye(N),R);

H=Su'*Qbar*Su+Rbar;

F=Sx'*Qbar*Su;
```

```
obj=u'*H*u+2*x(:,1)'*F*u+x(:,1)'*Sx'*Qbar*Sx*x(:,1);
Options = sdpsettings('solver','quadprog');
out=optimize(C,obj,Options);
double(u)
double(obj)
toc
```

```
Minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the default value of the optimality tolerance,
and constraints are satisfied to within the selected value of the constraint tolerance.



ans =

    1.0000
    0.9129
   -0.8299


ans =

   12.2743

Elapsed time is 0.570721 seconds.
```

3

```
For Batch solution:
U optimal:
ans =

    1.0000
    0.9129
   -0.8299


When N = 50, Feedback method:

U optimal:

ans =

    0.9887

    0.8219

    -0.7581

When N = 100, Feedback method:

U optimal:

ans =

    0.9987
    0.8862
    -0.8206
```

Summarize: When N increases, the Feedback solution becomes closer to the Batch Solution.

Published with MATLAB® R2016b