Solution DO.1: Down payment of a loan

Static and Dynamic Optimization

Notice, together with this solution comes (on the course home page) a distribution (dist1.zip) of m-files. On a unix system the distribution can be unpacked by the command: unzip -a dist1.zip.

1 Optimization

Just follow the instructions in the exercise.

2 Solving a set of equations

Just follow the instructions in the exercise.

3 Dynamic Optimization

Question: 1 We identify quite easily that:

\[ N = 10, \quad x_0 = 50000, \quad \alpha = 0.05, \quad a = 1 + \alpha = 1.05, \quad b = -1 \]

and

\[ f = ax_i + bu_i, \quad \phi = \frac{1}{2} px_i^2, \quad L = \frac{1}{2} q x_i^2 + \frac{1}{2} r u_i^2 \]

\[ \square \]

Question: 2

\[ H_i = \frac{1}{2} q x_i^2 + \frac{1}{2} r u_i^2 + \lambda_{i+1} (a x_i + b u_i) \]

\[ \square \]

Question: 3

\[ \frac{\partial}{\partial x} f = a \quad \frac{\partial}{\partial x} L = q x_i \]

\[ \frac{\partial}{\partial u} f = b \quad \frac{\partial}{\partial u} L = r u_i \]

\[ \square \]
3 Dynamic Optimization

Question: 4 Solution given in the text.

Question: 5 The stationarity condition (last equation) is simply:

\[ u_i = -\frac{b}{r} \lambda_{i+1} \]

Question: 6 If we reverse the costate equation, then

\[ \lambda_{i+1} = \frac{1}{a} \left[ \lambda_i - qx_i \right] \]

Question: 7 Solution given in the text.

Question: 8 The following code (fejlf.m) solves the recursions in (1).

```matlab
function err=fejlf(la0,a,b,x0,p,r,q,N)
la=la0; x=x0;
for i=0:N-1,
    la=(la-q*x)/a;
    u=-b*la/r;
    x=a*x+b*u;
end
err=la-p*x;
```

The output is the error in the terminal boundary condition.

Question: 9 The script below (runex3) uses fsolve for finding the correct initial costate (\( \lambda(0) \) alias la0) such that the terminal boundary condition is fulfilled.

```matlab
% Constants etc.
alf=0.05;
a=1+alf; b=-1;
x0=50000;
N=10;
qu=q*alf^2; r=q; p=q;

%r=10*q;
%r=q/10;
%p=0;
%p=100*q;

% The search for la0
la0=10;
opt=optimset('fsolve');
opt=optimset(opt,'Display','off');
la0=fsolve('fejlf',la0,opt,a,b,x0,p,r,q,N)
```
% The simulation with the correct la0
ut=[];
la=la0; x=x0;
lat=la; xt=x;
for i=0:N-1,
    la=(la-q*x)/a;
    u=-b*la/r;
    x=a*x+b*u;
    xt=[xt;x]; lat=[lat;la]; ut=[ut;u];
end

subplot(211);
bar(ut); grid; title('Input sequence');
axis([0 15 0 50000]);
subplot(212);
bar(xt); grid; title('Balance');
axis([0 15 0 50000]);

Question: 10 Change the values in the script above (runex3.m) and run the script.