

Answers to Problem Set 3

Group name: Ferienspass

Sebastian Kühnl: 5642348
 Alexander Dück (as: reebyte): 5504077
 Patrick Blank (as: paddyblank): 6729110
 Christian Wierschem: 6729288

1 Question 1

2 Question 2

2.1 Subquestion 1

From the slides, we get the following conditions for the variables on the balanced growth path

$$s_k f(k^*, h^*) - (\delta_k + g + n + ng) k^* = 0 \quad (1)$$

$$s_h f(k^*, h^*) - (\delta_h + g + n + ng) h^* = 0. \quad (2)$$

solving for k^* and h^* , respectively, gives

$$k^* = \frac{s_k f(k^*, h^*)}{(\delta_k + g + n + ng)} \quad (3)$$

$$h^* = \frac{s_h f(k^*, h^*)}{(\delta_h + g + n + ng)}. \quad (4)$$

However, since $f(k^*, h^*)$ depends on the variables that we try to solve for, this is not the final form yet.

Since we are assuming $F(K_t, H_t, A_t L_t) = K_t^\alpha H_t^\beta (A_t L_t)^{1-\alpha-\beta}$, we can further solve the expression

$$f(k^*, h^*) = \frac{F(K_t, H_t, A_t L_t)}{A_t L_t} = \frac{K_t^\alpha H_t^\beta (A_t L_t)^{1-\alpha-\beta}}{A_t L_t} = \left(\frac{K_t}{A_t L_t} \right)^\alpha \left(\frac{H_t}{A_t L_t} \right)^\beta. \quad (5)$$

By definition of k^* and h^* , this gives

$$f(k^*, h^*) = k_t^\alpha h_t^\beta. \quad (6)$$

Now, inserting this into 3 and 4

$$k^* = \frac{s_k k_t^\alpha h_t^\beta}{(\delta_k + g + n + ng)} \Leftrightarrow k^* = \left(\frac{s_k h_t^\beta}{(\delta_k + g + n + ng)} \right)^{\frac{1}{1-\alpha}} \quad (7)$$

$$h^* = \frac{s_h k_t^\alpha h_t^\beta}{(\delta_h + g + n + ng)} \Leftrightarrow h^* = \left(\frac{s_h k_t^\alpha}{(\delta_h + g + n + ng)} \right)^{\frac{1}{1-\beta}}. \quad (8)$$

It becomes evident, that k^* and h^* are a function of each other. However, we can simply substitute and then solve for the expression depending on parameters only

$$k^* = \left(\frac{s_k \left(\left(\frac{s_h k_t^\alpha}{(\delta_h + g + n + ng)} \right)^{\frac{1}{1-\beta}} \right)^\beta}{(\delta_k + g + n + ng)} \right)^{\frac{1}{1-\alpha}} \quad (9)$$

$$h^* = \left(\frac{s_h \left(\left(\frac{s_k h_t^\beta}{(\delta_k + g + n + ng)} \right)^{\frac{1}{1-\alpha}} \right)^\alpha}{(\delta_h + g + n + ng)} \right)^{\frac{1}{1-\beta}}. \quad (10)$$

which can now be solved for the respective variable. For k^* :

$$\begin{aligned} k^* &= \left(\frac{s_k (s_h k_t^\alpha)^{\frac{\beta}{1-\beta}}}{(\delta_h + g + n + ng)^{\frac{\beta}{1-\beta}} (\delta_k + g + n + ng)} \right)^{\frac{1}{1-\alpha}} \\ &= \left(\frac{s_k s_h^{\frac{\beta}{1-\beta}}}{(\delta_h + g + n + ng)^{\frac{\beta}{1-\beta}} (\delta_k + g + n + ng)} \right)^{\frac{1}{1-\alpha}} k_t^{\frac{\alpha\beta}{1-\beta-\alpha+\alpha\beta}} \\ &= \left(\frac{s_k s_h^{\frac{\beta}{1-\beta}}}{(\delta_h + g + n + ng)^{\frac{\beta}{1-\beta}} (\delta_k + g + n + ng)} \right)^{\frac{1-\beta-\alpha+\alpha\beta}{(1-\alpha)^2+\alpha\beta-\beta}} \end{aligned}$$

For h^* :

$$\begin{aligned} h^* &= \left(\frac{s_h (s_k h_t^\beta)^{\frac{\alpha}{1-\alpha}}}{(\delta_k + g + n + ng)^{\frac{\alpha}{1-\alpha}} (\delta_h + g + n + ng)} \right)^{\frac{1}{1-\beta}} \\ &= \left(\frac{s_h s_k^{\frac{\alpha}{1-\alpha}}}{(\delta_k + g + n + ng)^{\frac{\alpha}{1-\alpha}} (\delta_h + g + n + ng)} \right)^{\frac{1}{1-\beta}} h_t^{\frac{\alpha\beta}{1-\beta-\alpha+\alpha\beta}} \\ &= \left(\frac{s_h s_k^{\frac{\alpha}{1-\alpha}}}{(\delta_k + g + n + ng)^{\frac{\alpha}{1-\alpha}} (\delta_h + g + n + ng)} \right)^{\frac{1-\beta-\alpha+\alpha\beta}{(1-\beta)^2+\alpha\beta-\alpha}}. \end{aligned}$$

Thus, the solution vector becomes

$$\begin{bmatrix} k^* \\ h^* \end{bmatrix}' = \begin{bmatrix} \left(\frac{s_k s_h^{\frac{\beta}{1-\beta}}}{(\delta_h + g + n + ng)^{\frac{\beta}{1-\beta}} (\delta_k + g + n + ng)} \right)^{\frac{1-\beta-\alpha+\alpha\beta}{(1-\alpha)^2+\alpha\beta-\beta}} \\ \left(\frac{s_h s_k^{\frac{\alpha}{1-\alpha}}}{(\delta_k + g + n + ng)^{\frac{\alpha}{1-\alpha}} (\delta_h + g + n + ng)} \right)^{\frac{1-\beta-\alpha+\alpha\beta}{(1-\beta)^2+\alpha\beta-\alpha}} \end{bmatrix}'$$

2.6

All questions that lie between the first and this one are excluded from this document since they only involved programming exercises.

The path to be interpreted:

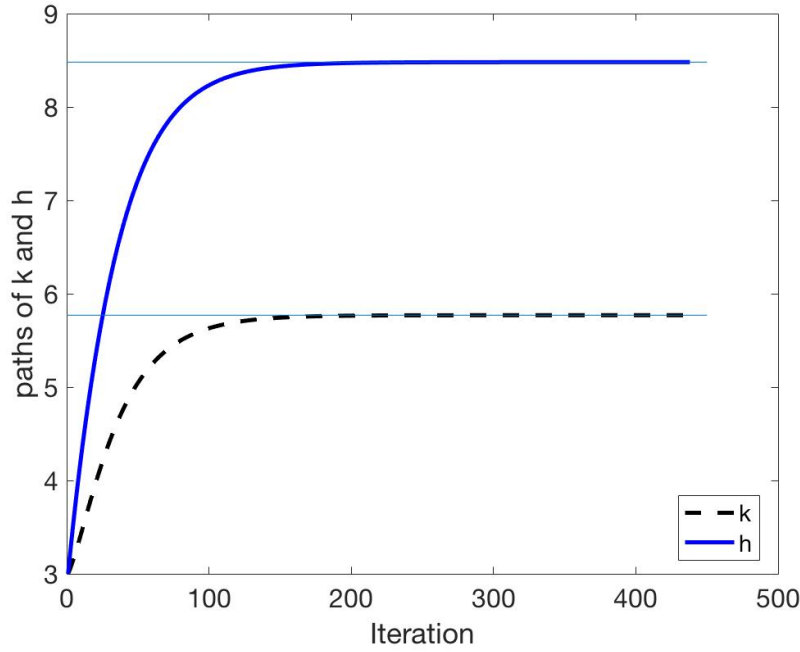


Figure 1: Paths of k^* and h^*

Both variables converge to the solution at the same time/speed. Optimal allocation human capital exceeds the optimal level of capital in all iterations.

A possible interpretation: That the balanced growth path level of human capital is greater than the balanced growth path level of physical capital could have two (simultaneous) reasons:

1. Human capital has a greater marginal product compared to physical capital

2. Human capital has a lower depreciation rate than physical capital

The reasoning behind this is that if an additional unit of human capital yields more than an additional unit of physical capital, and lasts longer than it as well, and investment in human capital is preferred over an investment in physical capital. This goes on this way until a point is reached, where an investment in human capital has the same marginal-product-to-depreciation-rate as physical capital. The investment in the two goods will increase such that their levels always give the same marginal-product-to-depreciation-rate (which implies a constant ratio of human to physical capital (which can be seen in the graph as well)) until the entire income is spent. At this point, the balanced growth path level is reached.

3 Question 3

4 Question 4

4.1 Analytical Solution

From the Euler equation and the budget constraint we get that

$$c_t = (\beta(1+r))^{-\frac{1}{\theta}} c_{t+1} \quad (11)$$

$$\sum c_t \left(\frac{1}{1+r} \right)^t = \sum w_t \left(\frac{1}{1+r} \right)^t + a_0 \quad (12)$$

Since we can express all c_t as a function of c_0 , we can write

$$c_0 \sum \left(\beta^{-\frac{1}{\theta}} (1+r)^{\frac{\theta-1}{\theta}} \right)^t = \sum w_t \left(\frac{1}{1+r} \right)^t + a_0 \quad (13)$$

which allows to compute all c_t after solving for c_0

$$c_0 = \frac{\sum w_t \left(\frac{1}{1+r} \right)^t + a_0}{\sum \left(\beta^{-\frac{1}{\theta}} (1+r)^{\frac{\theta-1}{\theta}} \right)^t} \quad (14)$$

$$c_{t+1} = c_t (\beta(1+r))^{\frac{1}{\theta}}. \quad (15)$$

$$(16)$$

This can now be implemented into an algorithm by using a for-loop.

4.2 Algorithm

```
1 %PS3 Problem 4
2 %No T is given, so declare it at the beginning of the
  program, and for any
```

```

3 %theta
4 clear;
5 close all;
6 clc;
7 T=20;
8 theta=1;
9 %variable initialization (fixed)
10 time=linspace(0,T,T+1); %just for plot
11 w=zeros(T+1,1);
12 w(1)=10;
13 beta=0.99;
14 r=0.05;
15 %these variables will be determined
16 a=zeros(T+1,1);
17 c=zeros(T+1,1);
18 %computation (could also be written as a function)
19 num=zeros(T+1,1);
20 denom=zeros(T+1,1);
21 if theta==1
22     factor=beta*(1+r);
23 else
24     factor=(1+r)^((1-theta)/theta)*beta^(1/theta);
25 end
26 for i=0:T
27     num(i+1)=w(i+1)*(1+r)^(-i);
28     denom(i+1)=(factor)^i;
29 end
30 c(1)=sum(num)/sum(denom);
31 for i=1:T
32     a(i+1)=a(i)*(1+r)+w(i)-c(i);
33     c(i+1)=c(i)*(beta*(1+r))^(1/theta);
34 end
35 figure
36 plot(time,c,time,w,time,a);
37 legend('C','W','A')
38 acheck=a(end);
39 ccheck=c(end);
40 if (round(ccheck*1000)/1000==round(acheck*(1+r)*1000)/1000)
41     display('True --> a_{T+1}= 0')
42 elseif (round(ccheck*1000)/1000>round(acheck*(1+r)*1000)/1000)
43     display('a_{T+1}<0')
44     disc=a(end);
45     subst=zeros(T+1,1);
46     for i=0:T

```

```

47     subst(i+1)=(1+r)^(i*((beta*(1+r))^(i/theta))^(T-i);
48 end
49 d=sum(subst);
50 c(1)=c(1)-d;
51 for i=1:T
52     c(i+1)=c(i)*(beta*(1+r))^(i/theta);
53     a(i+1)=a(i)*(1+r)+w(i)-c(i);
54 end
55 else
56     display('error, a_{T+1}>0')
57 end

```

4.3 What to do if $\theta = 1$

in the case that $\theta = 1$, we get

$$\frac{c_t^{1-1}-1}{1-1} = \frac{1-1}{0} = \frac{0}{0}.$$

In this case, an application of L'Hôpital's rule has to be applied

$$\frac{f(x)}{g(x)} = \frac{\left(\frac{\partial f(x)}{\partial x}\right)}{\left(\frac{\partial g(x)}{\partial x}\right)}.$$

We now have to use

$$f(\theta) = c_t^{1-\theta} - 1 \quad (17)$$

$$g(\theta) = 1 - \theta \quad (18)$$

giving

$$\frac{\partial f(\theta)}{\partial \theta} = (-1) \ln(c_t) \exp((1-\theta) \ln(c_t)) \quad (19)$$

$$\frac{\partial g(\theta)}{\partial \theta} = -1 \quad (20)$$

$$\frac{\left(\frac{\partial f(x)}{\partial x}\right)}{\left(\frac{\partial g(x)}{\partial x}\right)} = \ln(c_t) \exp((1-\theta) \ln(c_t)) \quad (21)$$

in the limit

$$\lim_{\theta \rightarrow 1} \frac{\left(\frac{\partial f(x)}{\partial x}\right)}{\left(\frac{\partial g(x)}{\partial x}\right)} = \ln(c_t) \quad (22)$$

which can then be used in the maximization problem to obtain a new Euler equation.

5 Question 5

5.1 Subquestion 3

5.1.1 Subquestion b

```
1 clear;
2 close all;
3
4 % Variables
5
6 phi=-3;
7 prob_low=0.1;
8 prob_high=1-prob_low;
9 alpha_upper=1;
10 alpha_lower=0;
11 w_init=10;
12 r_f=0.02;
13 r_high=0.12;
14 r_low=-0.08;
15
16 % Apply fminbnd
17 fun= @(alpha) (-utility(alpha,phi,prob_low,prob_high,
18     w_init,r_f,r_high,r_low));
19 alpha_fminbnd = fminbnd(fun,alpha_lower,alpha_upper);
20
21 % Apply fmincon
22 A=[];
23 b=[];
24 Aeq=[];
25 beq=[];
26 lb=alpha_lower;
27 ub=alpha_upper;
28 alpha_fmincon = fmincon(fun,0,A,b,Aeq,beq,lb,ub);
29
30 % Plot Utility and Marginal Utility
31 alpha_v=nan(100,1);
32 a=nan(100,1);
33 a_marg=nan(100,1);
34
35 for i=1:100
36     a(i,1)=utility(i/100,phi,prob_low,prob_high,w_init,
37         r_f,r_high,r_low);
38     a_marg(i,1)=marginal_utility(i/100,phi,prob_low,
39         prob_high,w_init,r_f,r_high,r_low);
40     alpha_v(i,1)=i/100;
```

```

38 end
39
40 figure
41 subplot(2,1,1); plot(alpha_v,a);
42 hline=refline((a(100,1)-a(1,1)),a(1,1));
43 hline.Color = 'r';
44 xlabel('alpha');
45 ylabel('utility');
46 legend('utility','Location','northwest');
47 subplot(2,1,2); plot(alpha_v,a_marg);
48 hline=refline((a_marg(100,1)-a_marg(1,1)),a_marg(1,1))
49 ;
49 hline.Color = 'r';
50 xlabel('alpha');
51 ylabel('marginal utility');
52 legend('marginal utility','Location','northeast');
53
54 disp(['fminbnd solution at alpha=', num2str(
55     alpha_fminbnd), '. fmincon solution at alpha=',
56     num2str(alpha_fmincon)]);
57
58 % Define functions for utility and marginal utility
59 function util=utility(alpha,phi,prob_low,prob_high,
60     w_init,r_f,r_high,r_low)
61 r=prob_high*r_high+prob_low*r_low;
62 util=(1/phi)*(w_init*(1+r_f+alpha*(r-r_f)))^(phi);
63 end
64
65 function mutil=marginal_utility(alpha,phi,prob_low,
66     prob_high,w_init,r_f,r_high,r_low)
67 r=prob_high*r_high+prob_low*r_low;
68 mutil=(w_init*(r-r_f))/((w_init*(1+r_f+alpha*(r-r_f)))
69     ^((1-phi)));
70 end

```

The fminbnd solution is at $\alpha=0.99993$. The first fmincon solution, starting from the upper boundary is at $\alpha=0.99828$. The second fmincon solution, starting from the lower boundary is at $\alpha=0.99767$.

The two methods can yield different results because

1. The way the constraints are defined differ. In fmincon, they can hold with equality, in fminbnd they have to hold with strict inequality.
2. The way the algorithm searches for a minimum. fmincon starts at some point x_0 and moves along the x -axis while evaluating the function. fminbnd searches for a local minimum on the entire interval.

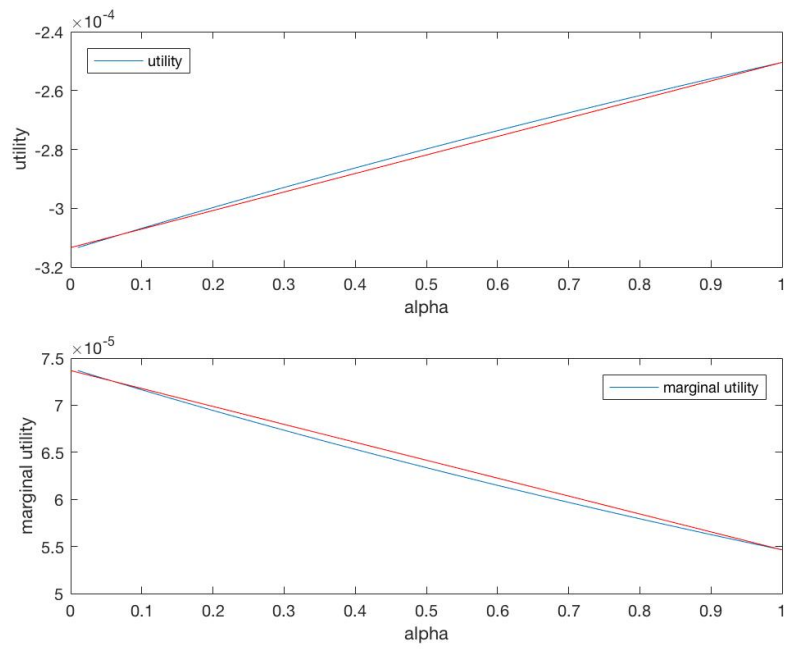


Figure 2: Illustration of the utility and marginal utility, depending on α .
(The red line is a reference line with linear slope)