ECON 714A - Problem Set 1

Alex von Hafften*

2/1/2020

Consider a neoclassical growth model with preferences $\sum_{t=0}^{\infty} \beta^t U(C_t)$, production technology $F(K_t)$, and the initial capital endowment K_0. Both $U(\cdot)$ and $F(\cdot)$ are strictly increasing, strictly concave and satisfy standard Inada conditions. The capital law of motion is $K_{t+1} = (1-\delta)K_t + I_t - D_t$ where D_t is a natural disaster shock that destroys a fixed amount of the accumulated capital.

1. Write down the social planner's problem and derive the inter-temporal optimality condition (the Euler equation).

The social planner's problem is to maximize the welfare of a representative agent subject to the resource constraint:

$$\max_{\{C_t, K_{t+1}\}} \sum_{t=0}^{\infty} \beta^t U(C_t)$$
$$\text{s.t. } C_t + D_t + K_{t+1} = F(K_t) + (1-\delta)K_t$$

Denote the lagrange multipler with $\beta^t \lambda_t$:

$$\sum_{t=0}^{\infty} \beta^t [U(C_t) + \lambda_t(F(K_t) + (1-\delta)K_t - C_t - D_t - K_{t+1})]$$

FOC $[C_t]$:

$$0 = \beta^t [U'(C_t) - \lambda_t] \implies U'(C_t) = \lambda_t$$

FOC $[K_{t+1}]$:

$$0 = -\beta^t \lambda_t + \beta^{t+1} \lambda_{t+1}(F'(K_{t+1}) + (1-\delta)) \implies \lambda_t = \beta \lambda_{t+1}(F'(K_{t+1}) + (1-\delta))$$

Thus, the Euler equation is:

$$U'(C_t) = \beta U'(C_{t+1})(F'(K_{t+1}) + (1-\delta))$$

*I worked on this problem set with a study group of Michael Nattinger, Andrew Smith, and Ryan Mather. I also discussed problems with Emily Case, Sarah Bass, and Danny Edgel.

2. Given the steady-state value of $D \geq 0$, write down the system of equations that determines the values of capital $\bar{K}(D)$ and consumption $\bar{C}(D)$ in the steady state. Draw a phase diagram with capital in the horizontal axis and consumption in the vertical axis, show the steady states, draw the arrows representing the direction of change, and the saddle path.

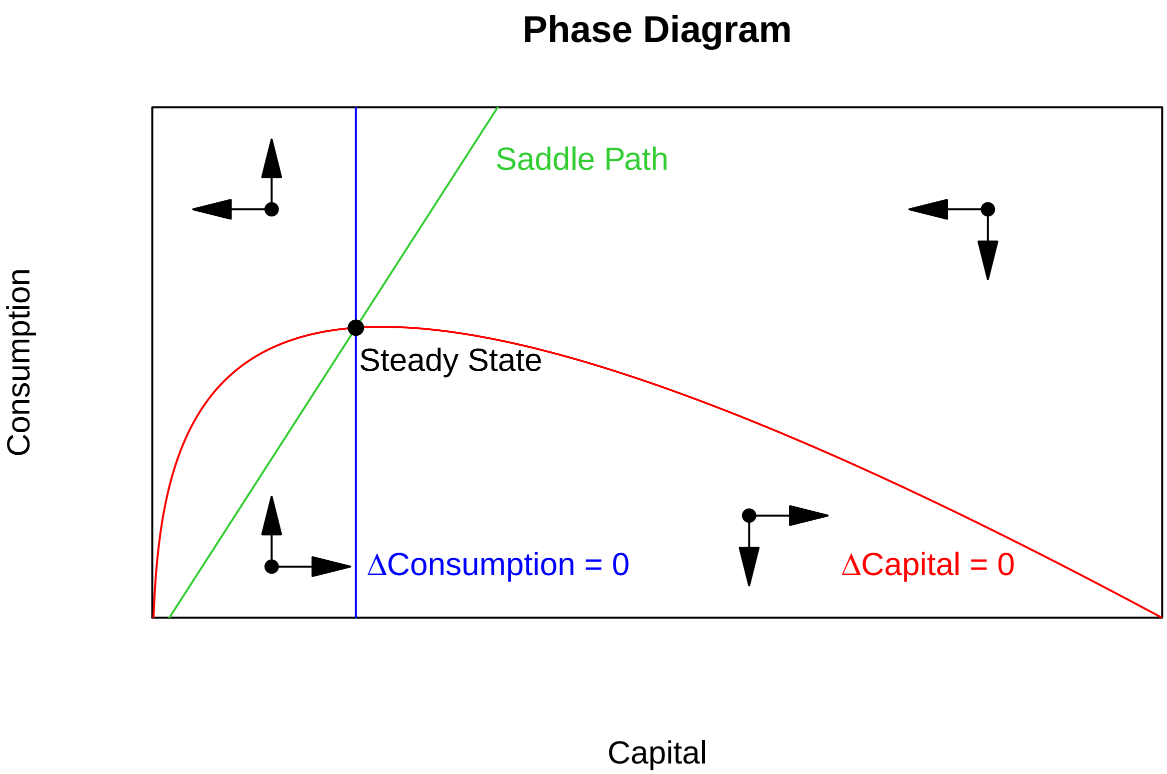
The system of equations can be derived from the Euler equation and the law of motion of capital. In a steady state, $C_t = C_{t+1} = \bar{C}$ and $K_t = K_{t+1} = \bar{K}$. In the steady state, the Euler equation implies

$$U'(\bar{C}) = \beta U'(\bar{C})(F'(\bar{K}) + (1-\delta)) \implies \bar{K}(D) = \bar{K} = (F')^{-1}\left(\frac{1}{\beta} - (1-\delta)\right)$$

In the steady state, the law of motion of capital implies:

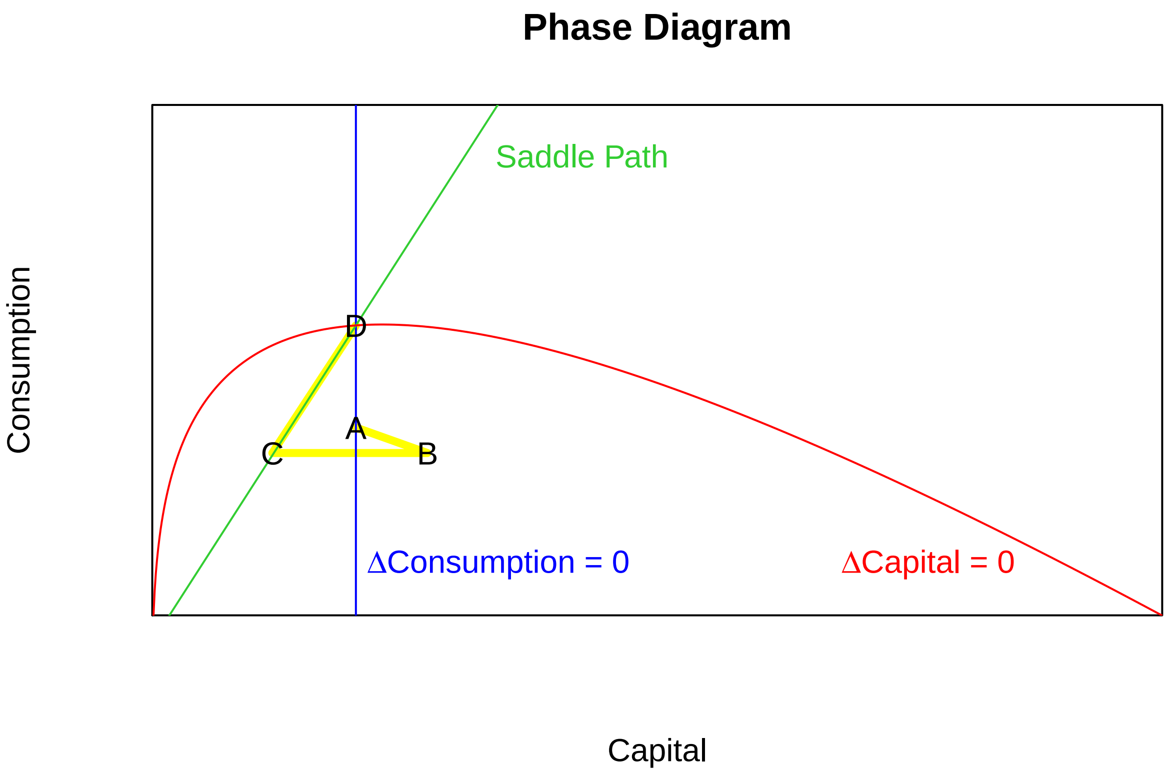
$$\bar{K} = (1-\delta)\bar{K} + F(\bar{K}) - \bar{C} - D \implies \bar{C}(D) = F(\bar{K}(D)) - D - \delta\bar{K}(D)$$

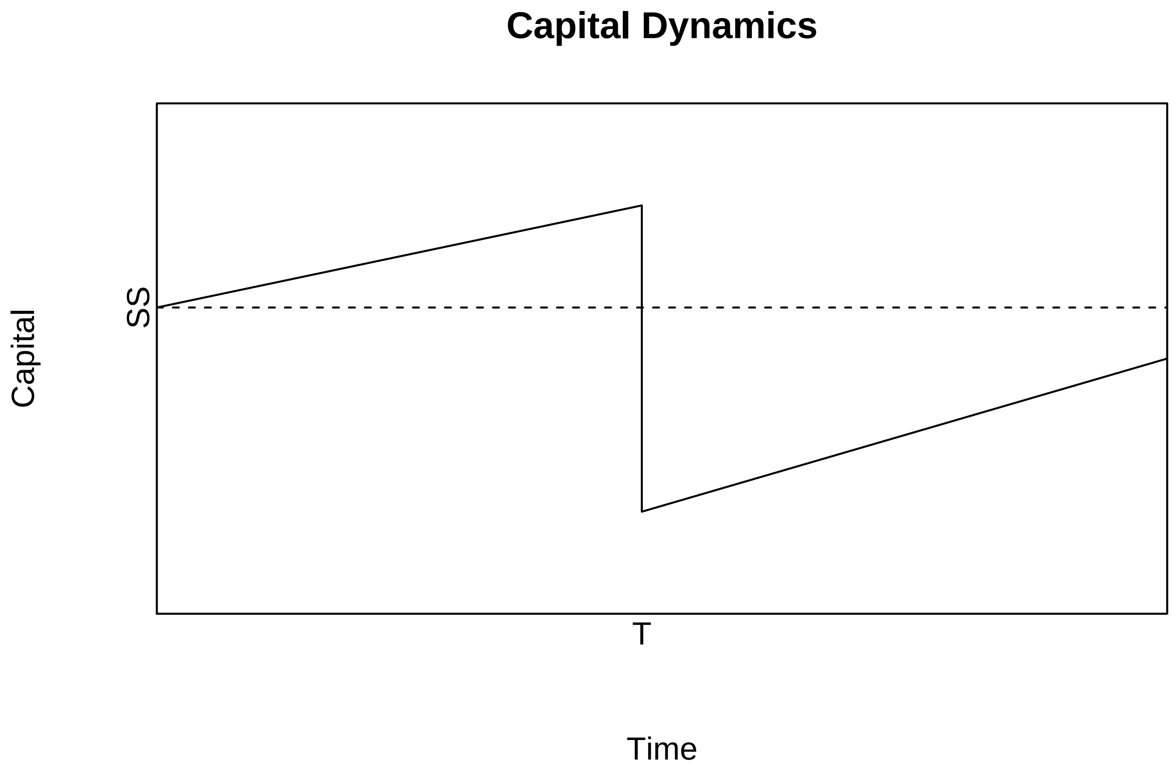
The phase diagram below shows the line where consumption does not change (blue), the line where capital does not change (red), the steady state (green), draw the arrows representing the direction of change, and the saddle path (black).

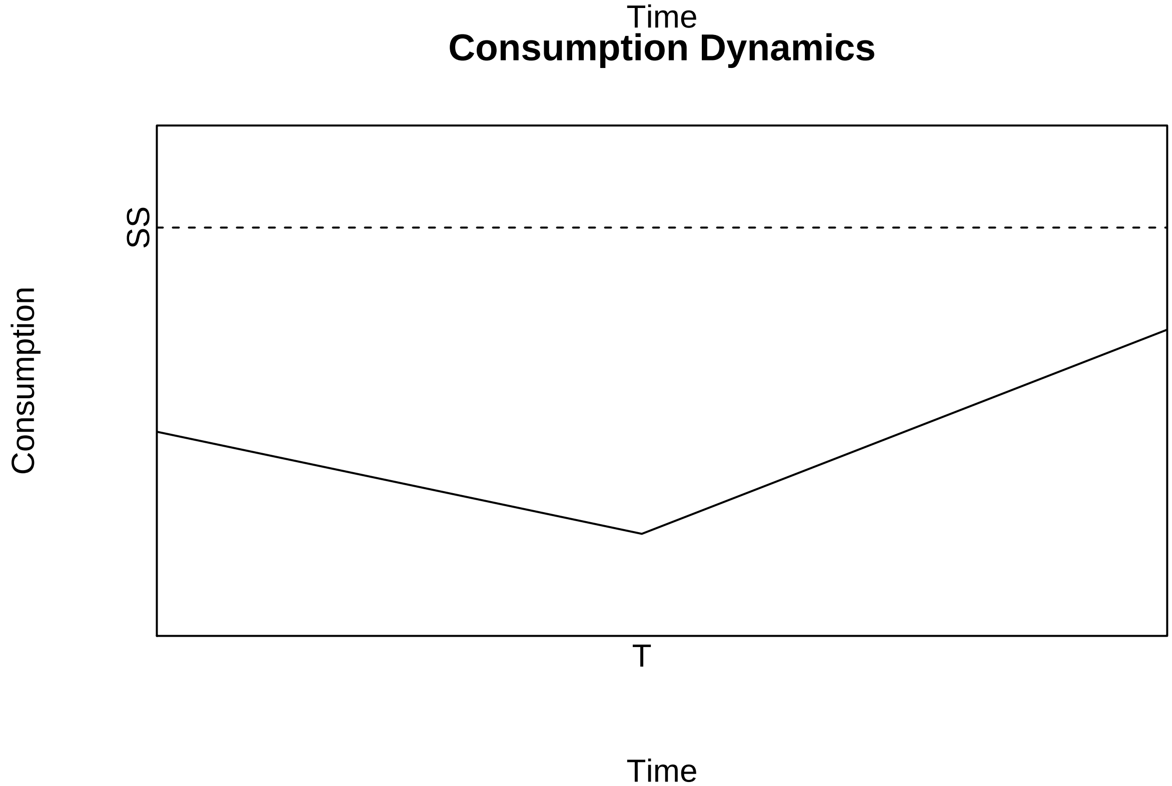


3. The scientists forecast an earthquake T periods from now that will destroy $D > 0$ units of capital. Assuming that economy starts from a steady state with $D = 0$, draw a phase diagram that shows the optimal transition path. Make two separate graphs showing the evolution of capital and consumption in time.

The phase diagram below shows the optimal transition path. At $t = 0$, news arrives about the earthquake and the shock to the capital shock. Consumption starts at point A in order to build up the capital shock. At $t = 1, ..., T - 1$, the economy follows a trajectory from point A to point B that further increases capital and decreases consumption. At $t = T$, the earthquake strikes and destroys D of the capital shock. The horizontal distance between point B and point C is D. At $t = T + 1, ...$, the economy follows the saddle path back to the steady state from point C to point D. The initial level of consumption (at point A) must therefore be such that point C is on the saddle path.



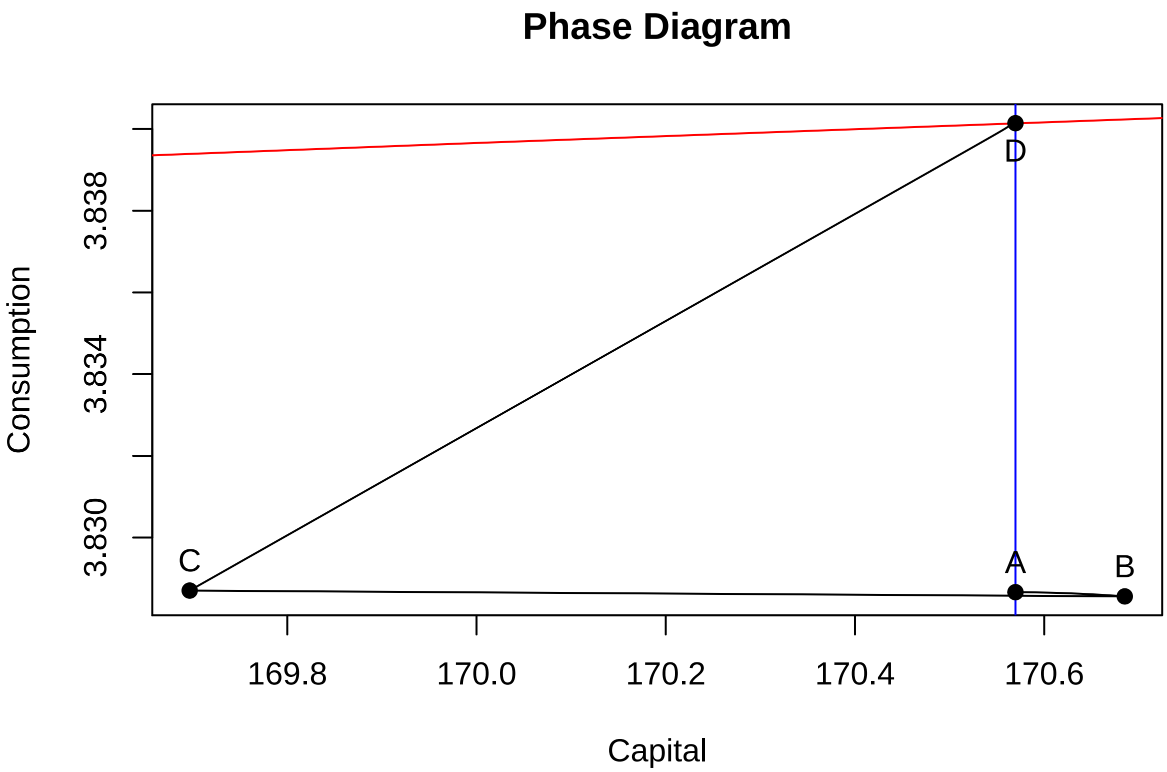

Capital Dynamics
Capital
SS
T
Time

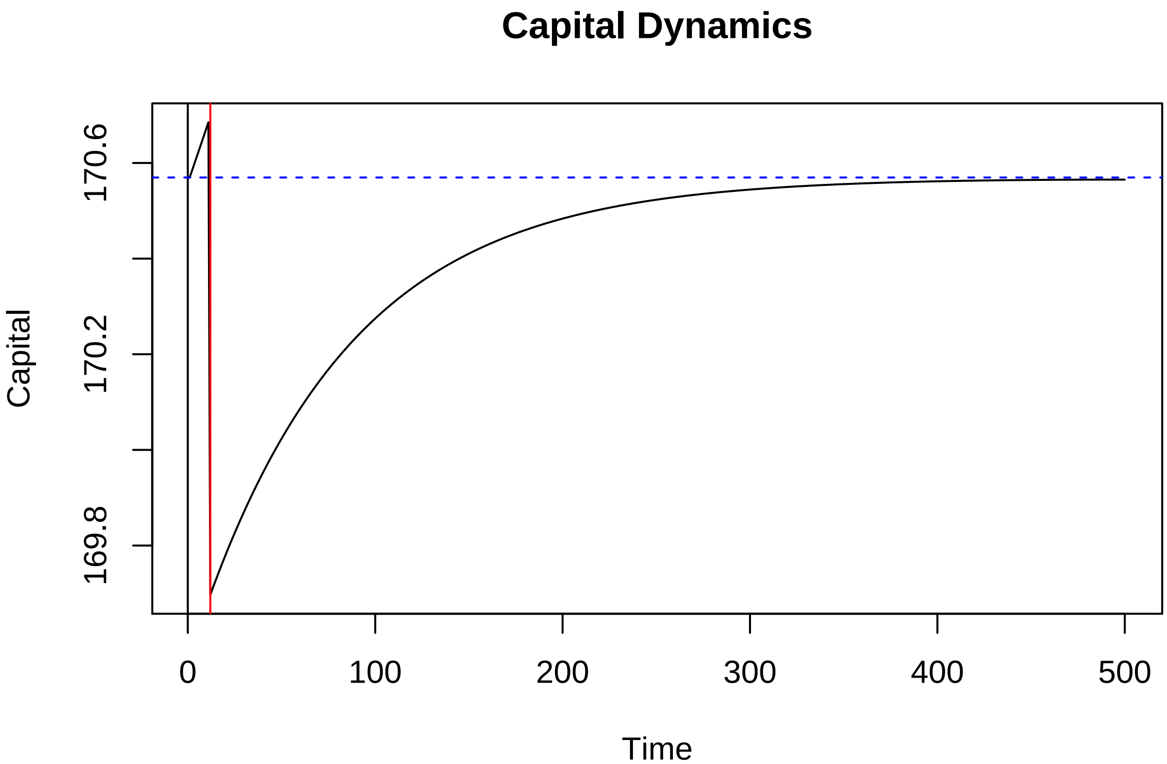

Consumption Dynamics
Consumption
SS
T
Time

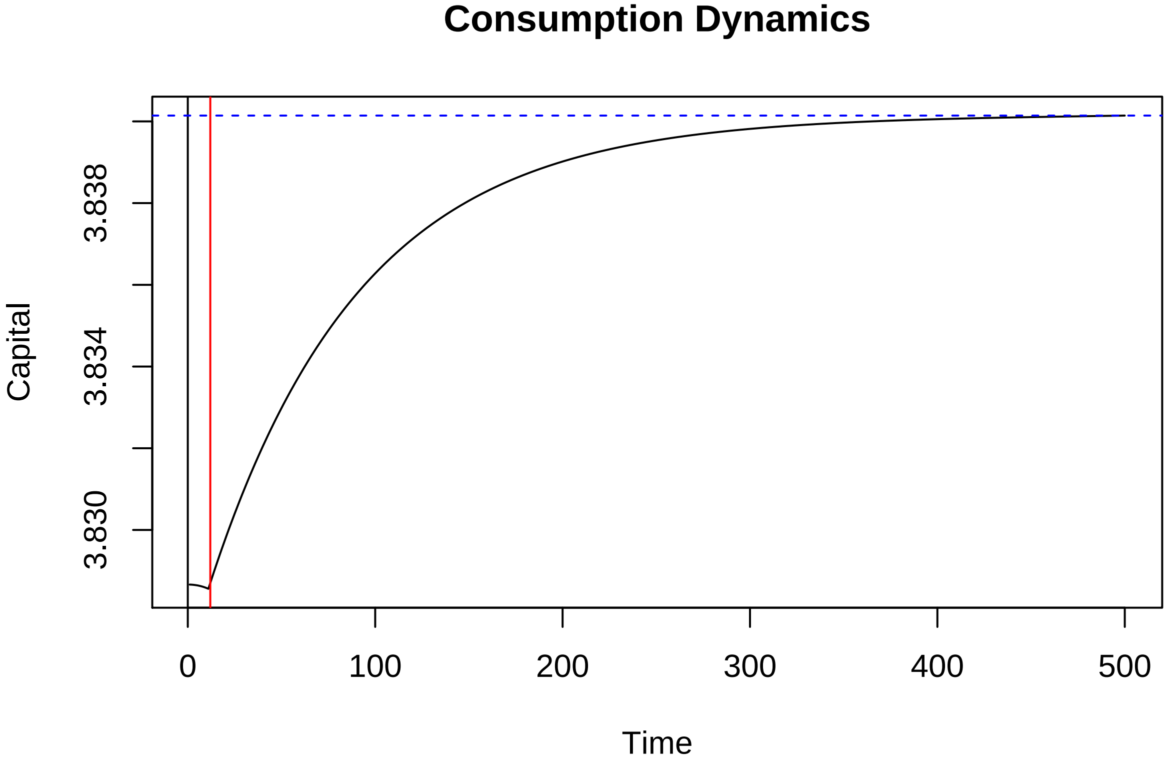
4. Assume that $U(C) = \frac{C^{1-\sigma}-1}{1-\sigma}$ and $F(K) = K^{\alpha}$ and the values of parameters are $\sigma = 1$, $\alpha = 1/3$, $\beta = 0.99^{1/12}$ (monthly model), $\delta = 0.01$, $T = 12$, $D = 1$. Using a shooting algorithm, solve numerically for the optimal transition path and plot dynamics of consumption and capital.

The phase diagram based on the numerical simulation is below. The economy starts at A and over the first 11 periods it builds up its capital stock (from A to B). At $t = 12$, the earthquake destroys 1 unit of the capital stock (from B to C). Point C is on the saddle path, so over the next many periods, the economy converges on the steady state (point D).

The dynamics of capital and consumption are also shown below. In each figure, the black line represents $t = 0$, the red line represents $t = 12$, and the blue line represent the steady state value.




Capital Dynamics
Capital
170.6
170.2
169.8
0
100
200
300
400
500
Time


Consumption Dynamics
Capital
3.838
3.834
3.830
0
100
200
300
400
500
Time

Appendix

```
# model parameters
delta <- 0.01
alpha <- 1/3
sigma <- 1
beta <- .99^(1/12)
d_earthquake <- 1
t_earthquake <- 12

# Solve SS
k_ss <- ((1/beta - 1 + delta) / alpha)^(1 / (alpha - 1))
c_ss <- k_ss^alpha - delta*k_ss

# simulation parameters
n_periods <- 500
c0_min <- 0
c0_max <- c_ss
max_iter <- 10000
iter <- 1
tolerence <- 0.001

# initialize simulation variables
k_path <- NULL
c_path <- NULL

while (iter < max_iter) {
  c_path[1] <- (c0_max + c0_min)/2
  k_path[1] <- k_ss

  # iterate for n_periods
  for (t in 2:n_periods) {
    k_path[t] <- (1 - delta) * k_path[t-1] + k_path[t-1] ^ alpha - c_path[t - 1]
    if (t == t_earthquake) k_path[t] <- k_path[t] - d_earthquake
    c_path[t] <- c_path[t-1] * beta * (alpha * k_path[t]^(alpha - 1) + 1 - delta)
  }

  # c and k value for testing if in SS
  c_test <- c_path[n_periods]
  k_test <- k_path[n_periods]

  # replace infinite test values with large double.
  if (!is.finite(c_test)) c_test <- .Machine$double.xmax
  if (!is.finite(k_test)) k_test <- .Machine$double.xmax

  # Break if within tolerence
  if (abs(c_test - c_ss) + abs(k_test - k_ss) < tolerence)  break

  # Update min and max of guess for consumption initial value and iterate counter.
  if (c_test < c_ss)  c0_min <- c_path[1]
  if (c_test > c_ss)  c0_max <- c_path[1]
  iter <- iter + 1
}
```