Notes of a general household life-cycle problem

Tianhao Zhao (GitHub: Clpr) Jan 2019

Background

Assumptions

Denotation

1. lowercase letters, such as c, k, l, are used to denote the variables with economic meanings
2. capital letters, such as A, B, C, are used to denote general algebrae. These algebrae are usually input by users
3. specifically, s denotes time/moment/period; c denotes consumption; l denotes **labor** (not leisure!); k denotes wealth/capital; b denotes bequests (if applicable)

Demography

1. One life-cycle problem is solved for only a (group of homogeneous) agent(s). If there are multiple agent groups (e.g. income groups), each group's life-cycle problem needs to be separately solved.
2. Finite life expectancy, where the maximum limit of life is $S \geq 2$ (years).
3. Determined retirement age $1 \leq S_r < S$, i.e. an agent works for S_r years.
4. $s = 1, \ldots, S + 1$ is used to denote the moment of the **beginning** of a year; e.g. when we talk about "year t", it means we are talking about a period from the moment $s = t, s = 1, \ldots, S$ to moment $s = t + 1$. However, we define some special moments:
   1. An agent is born at the **beginning** of the 1st year; the moment of birth is marked as $s = 1$.
   2. An agent begins to work when born at the moment $s = 1$, then retires at the **end** of the last working year S_r; the moment of retirement is marked as $s = S_r + 1$. It means that the agent is still working in year S_r. He/she becomes retired at moment $S_r + 1$ (the beginning of age-year $S_r + 1$!), then he/she begins to receive pension benefits in year $S_r + 1$.
   3. An agent dies at the **end** of the very last year, where the moment is marked as $s = S + 1$. It means this agent spends his last year, makes all deals (e.g. consumptions) then dies at the end of year S.
5. Deaths, no matter accidental deaths before $s = S + 1$ or the fatal natural death at moment $s = S + 1$, always happen at the end of years, i.e. agents always finish other activities such as consumption in their dead years before the moment of death.
6. No agent dies at moment $s = 1$ (no die-young case considered)

Utility

1. Cross-sectional utility function $u(c, l)$ is defined as the utility obtained from one year's consumption and labor supply. $u(c, l)$ is consumption-labor separable, i.e. $u(c_s, l_s) = u_1(c_s) + u_2(l_s)$
2. Agents try to maximize U, the summation of the present values of every year's cross-sectional utilities. The optimization can be denoted with a Bellman equation.
3. Agents use utility discounting factors $\beta_s \geq 0, 1 \leq s \leq S$ to discount the cross-sectional utility $u(c_s, l_s)$ in year s to moment $s - 1$.

Consumption, Work, Retirement

1. Agents only care his/her private interests; no children's interests considered.
2. Perfect forward-looking.
3. Agents possess capital k_s, $s = 1, \ldots, S + 1$, where k_s is a moment amount. k_s denotes the capital at the **beginning** of year s.
4. Agent decide every year's consumption c_s, $s = 1, \ldots, S$ & labor $0 \leq l_s \leq \bar{l}_s$, $s = 1, \ldots, S_r$, where $\bar{l}_s$ is time endowment; by default, $\bar{l}_s \equiv 1$ for each s. And, be careful, both c_s & l_s is a period amount, i.e. they indicates the consumption/labor during the period from moment $s = s$ to moment $s = s + 1$.
5. For the convenient of computing, $l_s \equiv 0$ for $s = S_r, \ldots, S$.
6. No final bequest left, i.e. agents always try to spend all their capital before natural fatal death at $s = S + 1$.
7. No capital when born.
8. Capital is never less than zero ($k_s \geq 0$)

Budget Constraint

1. Agents' consumption & labor supply are constrained by **linear** inter-temporal budget constraints:
   1. Before retirement: $A_s k_{s+1} = B_s k_s + D_s l_s - E_s c_s + F_s, 1 \leq s \leq S_r$
   2. After retirement: $A_s k_{s+1} = B_s k_s - E_s c_s + F_s, S_r < s \leq S$
   3. where $A_s > 0$ is a multiplier; $B_s > 0$ is a multiplier; $D_s, E_s \in \mathbf{R}$ are multipliers; and F_s is the extra cash/capital flows not related to possessed capital, labor or consumption.
2. According to our assumptions in previous sections, we have some boundary conditions:
   1. $A_s k_2 = D_1 l_1 - E_1 c_1 + F_1$, which is equivalent to $k_1 \equiv 0$.
   2. $0 = B_S k_S + D_S l_S - E_S c_S + F_S$, which is equivalent to $k_{S+1} \equiv 0$.

Mathematics: Problem Statements (DP)

This life-cycle problem can be written as a standard DP problem:

$$v(s) = \max_{c_s, l_s} [u(c_s, l_s) + \beta_s v(s + 1)], s = 1, \ldots, S$$

$$s.t. \begin{cases} A_s k_{s+1} = B_s k_s + D_s l_s - E_s c_s + F_s, 1 \le s \le S \\ l_s = 0, s = S_r + 1, \ldots, S \\ k_1 = k_{S+1} = 0 \\ v(S+1) = 0 \\ -k_s \le 0, s = 1, \ldots, S+1 \\ -c_s \le 0, s = 1, \ldots, S \\ l_s \le \bar{l}_s, s = 1, \ldots, S_r \\ -l_s \le 0, s = 1, \ldots, S_r \end{cases} \tag{1}$$

where $A_s > 0, B_s > 0, D_s \in \boldsymbol{R}, E_s \in \boldsymbol{R}, F_s \in \boldsymbol{R}, \beta_s \ge 0$ are **exogeneous** parameters which should be manually input/set by users/economists. Meanwhile, users need to explicitly define the structure of the cross-sectional utility function $u(c, l)$ according to our assumptions.

Based on Eq (1), users may use standard DP algorithms to solve this problem. However, solving this DP problem for many times (esp. in a large model system) may bring un-acceptable time cost. Therefore, I introduce a half-analytical approximated solution instead. This half-analytical algorithm uses Lagrange functions without the inequalities of endowment constraints (c_s, l_s, k_s etc.). It solves such a partial-constrained (only equality constraints!) problem, then uses a simple root-searching algorithm to find the consumption & labor & capital paths which meet those inequality endowment constraints). Of course, quite different from DP which does not require the structure of $u(\bullet)$, the proposed half-analytical algorithm uses the **analytical** FOCs of $u(\bullet)$ to accelerate computations. Therefore, the proposed algorithm requires the explicit definition of $u(\bullet)$, which leads to so many strong assumptions above. In this document, I will state one of the most simplest case: consumption-leisure separable $u(\bullet)$.

A Recursive Series

Before discussing the proposed half-analytical algorithm, let's look at the recurrence formula of a series:

$$x_{t+1} = a_t x_t + b_t, t \ge 1 \tag{2}$$

where $\{x_t | t \ge 1\}$ is the real-number series, $a_t > 0$ and $b_t \in \boldsymbol{R}$ are exogeneous parameters, and x_1 is a known constant. Readers may feel familiar with this series because it is an abstract form of our inter-temporal budget constraint. However, this series cannot be simply summarized because a_t is variant with t. Now, our task is clear: how to use k_1, the known constant, to denote any k_s for $s > 1$?

Let's use mathematical induction:

> ## Proof 1: $x_s(x_1), s > 1$
>
> 1. Substitute $x_2 = a_1 x_1 + b_1$ to $x3 = a_2 x_2 + b_2$, then get: $x_3 = a_2(a_1 x_1 + b_1) + b_2 = a_2 a_1 x_1 + a_2 b_1 + b_2$;
> 2. Substitute x_3 to x_4, then get: $x_4 = x_1 \prod_{i=3}^{4} a_i + (a_4 a_3 a_2 b_1 + a_4 a_3 b_2 + a_4 b_3 + b_4)$;
> 3. Define: $\prod_{i=n}^{m} a_i \equiv 1$ for $\forall n > m$;
> 4. Therefore, guess: $x_{s+1} = x_1 \prod_{i=1}^{s} a_i + \sum_{j=1}^{s} (b_j \prod_{i=j+1}^{s} a_i), s \ge 1$;

5. If our guess is true, we have: $x_{s+2} = x_1 \prod_{i=1}^{s+1} a_i + \sum_{j=1}^{s+1}(b_j \prod_{i=j+1}^{s+1} a_i), s \geq 1$;

6. Substitute x_{s+1} to $x_{s+2} = a_{s+1}x_{s+1} + b_{s+1}$, then get: $x_{s+2} = x_1 \prod_{i=1}^{s+1} a_i + \sum_{j=1}^{s+1}(b_j \prod_{i=j+1}^{s+1} a_i), s \geq 1$;

7. Therefore, our guess is true.

So far, we have proved:

$$x_{s+1} = x_1 \prod_{i=1}^{s} a_i + \sum_{j=1}^{s}(b_j \prod_{i=j+1}^{s} a_i), s \geq 1 \tag{3}$$

Now, let's go back to our inter-temporal budget constraint:

$$A_s k_{s+1} = B_s k_s + D_s l_s - E_s c_s + F_s, 1 \leq s \leq S \tag{4}$$

Because we have assumed $A_s > 0, B_s > 0$, this recursive formula of budget constraint can be written as:

$$k_{s+1} = \frac{B_s}{A_s} k_s + (\frac{D_s}{A_s} l_s - \frac{E_s}{A_s} c_s + \frac{F_s}{A_s}), 1 \leq s \leq S \tag{5}$$

Eq (5) looks familiar? Right! Denote $x_s = k_s, a_i = \frac{B_i}{A_i}$ and $b_j = \frac{D_j}{A_j} l_j - \frac{E_j}{A_j} c_j + \frac{F_j}{A_j}$, then substitute them to Eq (3), we have:

$$k_{s+1} = k_1 \prod_{i=1}^{s} \frac{B_i}{A_i} + \sum_{j=1}^{s}[(\frac{D_j}{A_j} l_j - \frac{E_j}{A_j} c_j + \frac{F_j}{A_j}) \prod_{i=j+1}^{s} \frac{B_i}{A_i}], s \geq 1 \tag{6}$$

or equivalently:

$$k_{s+1} = k_1 \prod_{i=1}^{s} \frac{B_i}{A_i} + \sum_{j=1}^{s}(\frac{D_j}{A_j} l_j \prod_{i=j+1}^{s} \frac{B_i}{A_i}) - \sum_{j=1}^{s}(\frac{E_j}{A_j} c_j \prod_{i=j+1}^{s} \frac{B_i}{A_i}) + \sum_{j=1}^{s}(\frac{F_j}{A_j} \prod_{i=j+1}^{s} \frac{B_i}{A_i}), s \geq 1 \tag{7}$$

Now, let $s = S$, then k_{S+1} becomes bequests, the capital when agents die. According to our assumptions, $k_{S+1} = 0$, thus, define:

$$G(c_s, l_s | s = 1, \ldots, S) = k_1 \prod_{i=1}^{S} \frac{B_i}{A_i} + \sum_{j=1}^{S}[(\frac{D_j}{A_j} l_j - \frac{E_j}{A_j} c_j + \frac{F_j}{A_j}) \prod_{i=j+1}^{S} \frac{B_i}{A_i}], s \geq 1 \tag{8}$$

Therefore, all inter-temporal budget constraints have been compressed to a single equation:

$$G(c_s, l_s | s = 1, \ldots, S) = 0 \tag{9}$$

Mathematics: Problem Statements (Lagrange)

The proposed half-analytical algorithm requires the following partial-constrained optimization problem denoted as an standard maximization problem:

$$\max_{c_s, s=1,\ldots,S; l_s, s=1,\ldots,S_r} U = \sum_{s=1}^{S} \beta_s u(c_s, l_s)$$
$$\text{s.t. } G(c_s, l_s | s = 1, \ldots, S) = 0 \tag{10}$$

In Eq (10), we temporarily neglect all endowment constraints because they are inequalities, and use standard Lagrange multiplier to solve Eq (10). The Lagrange function is:

$$\max_{c_s, s=1,\ldots,S; l_s, s=1,\ldots,S_r} L = U(c_s, l_s | s = 1, \ldots, S) - \lambda G(c_s, l_s | s = 1, \ldots, S) \tag{11}$$

Now, we are going to solve this problem with FOCs & Euler equation.

FOCs and Euler Equation

To solve Eq (11), we need the first-order conditions (FOCs) of Eq (11). We have assumed that the cross-sectional utility function $u(c, l)$ is explicitly defined and it is a consumption-labor separable $u(c, l) = u_1(c) + u_2(l)$. Therefore, we write the FOCs of Eq(11):

$$\begin{cases} \frac{\partial U}{\partial c_s} = \beta_s \frac{\partial u}{\partial c_s} - \lambda \frac{\partial G}{\partial c_s} = 0, s = 1, \ldots, S \\ \frac{\partial U}{\partial l_s} = \beta_s \frac{\partial u}{\partial l_s} - \lambda \frac{\partial G}{\partial l_s} = 0, s = 1, \ldots, S_r \\ \frac{\partial U}{\partial \lambda} = G = 0 \end{cases} \tag{12}$$

Because $\lambda > 0$, combining $\frac{\partial U}{\partial c_s}$ and $\frac{\partial U}{\partial c_{s+1}}$, we get our Euler equation:

$$\frac{\partial u / \partial c_s}{\partial u / \partial c_{s+1}} = \frac{\beta_{s+1}}{\beta_s} \frac{\partial G / \partial c_s}{\partial G / \partial c_{s+1}}, s = 1, \ldots, S - 1 \tag{13}$$

The Euler equation, when $\partial u / \partial c_s$ is explicitly defined, can be equivalently denoted as a recursive formula of c_s:

$$c_s = \varepsilon(s | c_1) \tag{14}$$

which we will use to compress the optimization about many c_s to a new one only about c_1. Meanwhile, combining $\frac{\partial U}{\partial l_s}$ and $\frac{\partial U}{\partial c_s}$, we can get the optimal conversion formula from l_s to c_s without endowment constraints:

$$\frac{\partial u / \partial c_s}{\partial u / \partial l_s} = \frac{\partial G / \partial c_s}{\partial G / \partial l_s}, s = 1, \ldots, S_r \tag{15}$$

which is equivalent to the following conversion function:

$$l_s = \tilde{\gamma}(c_s) = \tilde{\gamma}(\varepsilon(s | c_1)) = \gamma(s | c_1) \tag{16}$$

So far, we have used the information of the first two conditions of Eq(12). Now, we substitute Eq(14) and Eq(16) to $G = 0$ to solve the optimal c_1^* where no endowment constraints are considered:

$$G(c_s^*, l_s^* | s = 1, \ldots, S) = G(c_1^*) = 0 \tag{17}$$

Adjustment on l_s

Eq (17) has answered the optimal paths of c_s^* and l_s^* under the conditions without endowment constraints. However, endowment constraints, i.e. $c_s \geq 0$ and $0 \leq l_s \leq \bar{l}_s$, are essential which c_s^* and l_s^* cannot always meet. Therefore, we need to adjust c_s^* and l_s^* to make them meet these endowment constraints.

> **Algorithm: adjust l_s^* and c_s^***
>
> 1.Assumptions:
>
>> 1. The dynamics of consumptions (Euler equation) are retained, no matter what the value of c_1^* and/or what the values of l_s^*;
>> 2. However, the conversion formula $l_s^* = \tilde{\gamma}(c_s^*)$ is now not retained because we adjusted l_s^*;
>> 3. Budget constraint $G = 0$ must be satisfied.
>
> 2. Algorithm:
>
>> 1. Let $\tilde{l}_s^* = [[l_s^*, \bar{l}_s]^-, 0]^+$ be the bounded/adjusted labor supplies, where $[a, b]^+$ denotes the larger one of a, b, and $[a, b]^-$ denotes the smaller one of a, b;
>> 2. Solve the equation $G(\tilde{c}_s | \tilde{l}_s, s = 1, ..., S) = G(\tilde{c}_1 | \varepsilon(\cdot), \tilde{l}_s, s = 1, ..., S) = 0$ with numerical root-searching algorithms (e.g. bisection search).