Liquid and Illiquid Assets
with Fixed Adjustment Costs

Greg Kaplan, Peter Maxted and Benjamin Moll

This note describes a continuous-time version of the model in Kaplan and Violante (2014) in which a household optimally splits his wealth between liquid and illiquid assets. Because adjustments are subject to a fixed cost, the model has a stopping-time element. We here describe how such model can be solved numerically using a finite difference method. Based on ongoing research (Kaplan and Moll, 201?). Note that the problem is different from that in Kaplan, Moll and Violante (2018). The problem presented here features a non-convex adjustment cost whereas the problem in Kaplan, Moll and Violante (2018) features a kinked but strictly convex adjustment cost. Both types of cost functions give rise to an inaction region. However, the mathematical structure is fundamentally different: a non-convex adjustment cost results in a stopping-time or impulse-control problem whereas a strictly convex cost function does not.

These impulse control problems can be formulated as so-called Hamilton-Jacobi-Bellman Variational Inequalities (HJBVIs) or Hamilton-Jacobi-Bellman Quasi-Variational Inequalities (HJBQVIs). See Bensoussan and Lions (1982, 1984) and Bardi and Capuzzo-Dolcetta (1997). More recently, Bertucci (2017, 2018) analyzes Mean Field Games with stopping and impulse control, with the prototypical problem featuring a coupled system of an HJBVI or HJBQVI for agents’ problems and the corresponding variant of a Kolmogorov Forward equation for the evolution of the distribution of agents.

1 Model Setup

Households can invest in two assets: an illiquid asset $a$, and a liquid asset $b$. The liquid asset pays a real return $r_b$ and can be freely traded subject to a borrowing limit. The illiquid asset pays a return $r_a > r_b$. Deposits and withdrawals can be made into and out of the illiquid asset only upon payment of a transaction cost $\kappa$. The households receive an income flow $z$ where $z$ can take a finite number of values. Productivity shocks arrive according to a stochastic process which is either Poisson or a diffusion process.

---

1We thank Alberto Polo for help with cleaning up this document and improving the corresponding code.
1.1 Households’ Problem

Households solve the following problem

\[
v_k(a, b) = \max_{\{c_t\}, \tau} \mathbb{E}_0 \int_0^\tau e^{-\rho t} u(c_t) dt + e^{-\rho \tau} \mathbb{E}_0 v_k^*(a_\tau + b_\tau)
\]

\[
\dot{a}_t = ra_t, \quad \dot{b}_t = rb_t + wz_t - c_t
\]

\[
a_t \geq 0, \quad b_t \geq 0, \quad (a_0, b_0, z_0) = (a, b, z_k)
\]

where \(z_t \in \{z_1, z_2\}\) is a two-state Poisson process with intensities \(\lambda_1, \lambda_2\), and where

\[
v_k^*(a + b) = \max_{a', b'} v_k(a', b') \quad \text{s.t.} \quad a' + b' = a + b - \kappa, \quad a' \geq 0, \quad b' \geq 0.
\]

(2)

For future reference, denote the optimal adjustment decisions conditional on adjustment by \(a_k^*(a, b)\) and \(b_k^*(a, b)\). Note that these only depend on the total amount of assets \(a + b\) (rather than \(a\) and \(b\) separately). The HJB equation is

\[
\rho v_k(a, b) = \max_c u(c) + \partial_a v_k(a, b)r_a a + \partial_b v_k(a, b)(wz_k + rb_b - c) + \lambda_k(v_{-k}(a, b) - v_k(a, b))
\]

(3)

for \(k = 1, 2\), and with state-constraint boundary condition

\[
\partial_b v_k(a, 0) \geq u'(wz_k)
\]

(4)

and a constraint that

\[
v_k(a, b) \geq v_k^*(a + b) \quad \text{all} \ a, b.
\]

(5)

This can also be written compactly as follows

\[
\min \{\rho v_k(a, b) - \max_c u(c) - \partial_a v_k(a, b)r_a a - \partial_b v_k(a, b)(wz_k + rb_b - c) - \lambda_k(v_{-k}(a, b) - v_k(a, b)), v_k(a, b) - v_k^*(a + b)\} = 0
\]

(6)

In mathematics, (6) is called an “HJB variational inequality” (HJBVI for short), or more precisely an “HJB quasi-variational inequality” (HJBQVI). See e.g. Bensoussan and Lions (1982, 1984), Barles, Daher and Romano (1995), Bardi and Capuzzo-Dolcetta (1997) and Tourin (2013).

Note that we can also write the adjustment value \(v_k^*\) as \(v_k^* = M v_k\) where \(M\) is known as the “intervention operator” (see e.g. Oksendal and Sulem, 2002; Azimzadeh, Bayraktar and
Labahn, 2018) so that (suppressing dependence on \((a, b)\)) the HJBQVI equation becomes

\[
\min \{ \rho v_k - \max_c u(c) - r_a a \partial_a v_k - (w z_k + r_b b - c) \partial_b v_k - \lambda_k (v_{-k} - v_k), v_k - \mathcal{M} v_k \} = 0
\]

## 2 Numerical Solution

See [http://www.princeton.edu/~moll/HACTproject/liquid_illiquid_LCP.m](http://www.princeton.edu/~moll/HACTproject/liquid_illiquid_LCP.m).

### 2.1 Household’s Problem: Linear Complementarity Problem + Finite Differences

We use an analogous approach to the simple model of exercising an option as in these notes [http://www.princeton.edu/~moll/HACTproject/option_simple.pdf](http://www.princeton.edu/~moll/HACTproject/option_simple.pdf). Once discretized the HJBVI (6) becomes

\[
\min \{ \rho v - u(v) - A(v)v, v - v^*(v) \} = 0 \tag{7}
\]

The main difference to the “exercising an option” problem laid out above is that the HJB equation without adjustment (the left branch of (7)) is non-linear in \(v\) and hence some iteration is necessary. Related also the value of having a reoptimized portfolio \(v^*\) depends on the value function as can be seen from (2). We proceed as follow:

1. as an initial guess \(v^0\) use the solution to

\[
\rho v - u(v) - A(v)v = 0 \tag{8}
\]

i.e. the problem without adjustment, i.e. in which the fixed cost \(\kappa\) is infinite.

2. Given \(v^n\), find \(v^{n+1}\) by solving

\[
\min \left\{ \frac{v^{n+1} - v^n}{\Delta} + \rho v^{n+1} - u(v^n) - A(v^n)v^{n+1}, v^{n+1} - v^*(v^n) \right\} = 0
\]

Exactly as in [http://www.princeton.edu/~moll/HACTproject/option_simple.pdf](http://www.princeton.edu/~moll/HACTproject/option_simple.pdf), this problem can be written as a linear complementarity problem (LCP).

3. Stop when \(v^{n+1}\) is sufficiently close to \(v^n\).
We have also tried another approach to solve the household's problem (6). This approach combines the finite-difference method with a so-called “operator splitting method” to take care of the constraint (5). However, this method seems to be inferior to the LCP-algorithm developed above, particularly in terms of speed.

Denote \( v_{i,j,k} = v_k(a_j, b_i) \). The algorithm works as follows. Start with an initial guess \( v_{i,j,k}^0 \) and for \( n = 1, 2, ... \) compute \( v_{i,j,k}^n \) as follows.

1. given \( v_{i,j,k}^n \), obtain \( v_{i,j,k}^{n+\frac{1}{2}} \) by solving a discretized HJB equation that is the same as if there were no adjustment decision:

\[
\frac{v_{i,j,k}^{n+\frac{1}{2}} - v_{i,j,k}^n}{\Delta} + \rho v_{i,j,k}^{n+\frac{1}{2}} = u(c_{i,j,k}^n) + \partial_a v_{i,j,k}^{n+\frac{1}{2}} r_a a_j + \partial_b v_{i,j,k}^{n+\frac{1}{2}} (wz_k + r b_i - c_{i,j,k}^n) + \lambda_k(v_{i,j,k}^n - v_{i,j,k}),
\]

\[c_{i,j,k}^n = (u')^{-1}(\partial_b v_{i,j,k}^n).\]

Here \( \partial_a v_{i,j,k} \) and \( \partial_b v_{i,j,k} \) denote the finite-difference approximation of the partial derivative of \( v \) with respect to \( a \) and \( b \) (either forward- or backward-difference approximations), and where \( \Delta \) is the size of the updating step. In particular, one can write the corresponding system of equations

\[
\frac{1}{\Delta} \left(v^{n+\frac{1}{2}} - v^n\right) + \rho v^n = u^n + A v^{n+\frac{1}{2}}
\]

where \( v^n \) is a vector of length \( L = I \times J \times 2 \) with the stacked value function as its entries, and \( A \) is the \( L \times L \) transition matrix corresponding to the discretized process summarizing the evolution of \((a_t, b_t, z_t)\). See http://www.princeton.edu/~moll/HACTproject/HACT_Numerical_Appendix.pdf for details in a similar model (but with one asset only).

2. Compute

\[
(v_{i,j,k}^{n+\frac{1}{2}}) = \max_{a',b'} v_{k}^{n+\frac{1}{2}}(a', b') \quad \text{s.t.} \quad a' + b' = a_j + b_i - \kappa, \quad a' \geq 0, \quad b' \geq 0.
\]

where \( v_{k}^{n+\frac{1}{2}}(a', b') \) is \( v_{i,j,k}^{n+\frac{1}{2}} \) interpolated at points \((a', b')\).

\footnote{Note that this splitting method has nothing to do with the “splitting the drift” trick in http://www.princeton.edu/~moll/HACTproject/two_asset_nonconvex.pdf.}
3. given \( v_{i,j,k}^{n+\frac{1}{2}} \) by setting
\[
v_{i,j,k}^{n+1} = \max\{v_{i,j,k}^{n+\frac{1}{2}}, (v_{i,j,k}^{*})^{n+\frac{1}{2}}\}.
\]

4. If \( v_{i,j,k}^{n+1} \) is “close to” \( v_{i,j,k}^{n} \), stop.

The algorithm is called a “splitting algorithm” because the “operator” on \( v \) defined by (6) is split into two steps: that of going from \( v^n \) to \( v^{n+\frac{1}{2}} \) and that of going from \( v^{n+\frac{1}{2}} \) to \( v^{n+1} \). It can be shown that the algorithm converges if the discretization of the HJB equation in step 1 satisfies the monotonicity, consistency and stability conditions of Barles and Souganidis (1991). See Achdou et al. (2017) for a discussion in the context of a model without a stopping-time decision. See e.g. Barles, Daher and Romano (1995) and Tourin (2013) for a discussion of convergence of numerical schemes for models with a stopping-time decision.

### 2.3 Kolmogorov Forward Equation

Without adjustment the KF equation is
\[
0 = -\partial_a(s^a_k(a,b)g_k) - \partial_b(s^b_k(a,b)g_k) - \lambda_k g_k + \lambda_{-k} g_{-k}
\]
for all \((a,b)\) and \(k = 1, 2\). Here \(s^a_k\) and \(s^b_k\) are the illiquid and liquid saving policy functions. With adjustment there are extra terms. The mathematical formulation of Kolmogorov Forward equations with stopping and/or impulse control is not straightforward. See Bertucci (2017, 2018) for a treatment. However, this is not an obstacle for the numerical solution. In particular, it turns out to be quite easy to work with the discretized process as captured by the matrix \(A\) constructed in section 2.1 (or 2.2). See section 2 here http://www.princeton.edu/~moll/HACTproject/HACT_numerical_appendix.pdf for a more detailed explanation of this logic in a model without an adjustment decision.

If there were no adjustment, things would be very simple. In particular, the vector of stacked finite difference of the distribution, \(g_\ell\) for \(\ell = 1, ..., L\) where \(L = I \times J \times 2\), would satisfy a linear system
\[
0 = A^T g
\]
where \(A^T\) is the transpose of the transition matrix \(A\) from the HJB equation (8).

We now explain how to deal with the fact that there is adjustment. We first introduce some additional notation. Denote by \(a_{i,j,k}^*(a_i, b_i)\) and \(b_{i,j,k}^*(a_i, b_i)\) the optimal “adjustment targets” conditional on adjustment. Switching notation to the stacked and discretized state space, \(\ell = 1, ..., L\), denote by \(k^*(\ell)\) the grid point \(k = 1, ..., L\) that is reached from point \(\ell = 1, ..., L\) upon adjustment. Finally, denote by \(I\) the set of grid points
in the inaction region. This set is defined by the requirement that \( v_\ell > v^*_\ell \) for all \( \ell \in I \), whereas \( v_\ell = v^*_\ell \) for all \( \ell \in I \) (i.e. \( v_\ell = v^*_\ell \) for points in the adjustment region).

The problem with using transition matrix \( A \) alone is that it does not capture adjustment. To introduce adjustment, we now define a binary matrix \( M \) which we term the “intervention matrix”. The elements of \( M \), denoted by \( M_{\ell,k} \) for \( \ell = 1, \ldots, L \) and \( k = 1, \ldots, L \) are given by

\[
M_{\ell,k} = \begin{cases} 
1, & \text{if } \ell \in I \text{ and } \ell = k \\
1, & \text{if } \ell \in I \text{ and } k^*(\ell) = k \\
0, & \text{otherwise}
\end{cases}
\]

This matrix moves points that are in the adjustment region to their corresponding adjustment targets. For instance, note that for points in the adjustment region the outside option \( v^*(v) \) in the discretized HJBVI equation (7) satisfies \( v^*(v) = Mv \). Therefore, the intervention matrix is the natural discretization of the intervention operator \( M \) discussed in Section 1.

To see how we use \( M \) to solve the Kolmogorov Forward equation with adjustment, consider a time-dependent KF equation but with fixed policy rules given by \( A \) and \( M \). Denoting \( g^n = g(t^n), n = 1, \ldots, N \), the goal is to find a mapping from \( g^n \) to \( g^{n+1} \). Motivated by the “operator splitting method” in section 2.2 and using the same notation as there, we split the step of finding \( g^{n+1} \) given \( g^n \) into two sub-steps:

1. Given \( g^n \) find \( g^{n+\frac{1}{2}} \) from

\[
g^{n+\frac{1}{2}} = M^T g^n
\]

2. Given \( g^{n+\frac{1}{2}} \) find \( g^{n+1} \) from

\[
\frac{g^{n+1} - g^{n+\frac{1}{2}}}{\Delta t} = (AM)^T g^{n+1}
\]

Adjustment introduces two related but distinct questions: (i) how should we treat the density at grid points in the adjustment region? (ii) how should we treat the density at grid points in the inaction region but from which the stochastic process for idiosyncratic state variables ends up in the adjustment region?

The two parts of the operator splitting scheme show how we answer these questions. Step 1 answers question (i) by simply moving any mass from the adjustment region to the inaction region. Step 2 answers question (ii). Instead of using matrix \( A \) as the transition matrix as in the case without adjustment, we now use matrix \( AM \) as the transition matrix.\(^4\) To

---

\(^3\)This is an implicit scheme. The analogous explicit scheme is \( \frac{g^{n+1} - g^{n+\frac{1}{2}}}{\Delta t} = (AM)^T g^{n+\frac{1}{2}} \).

\(^4\)Without adjustment (i.e., \( I = \{1, \ldots, L\} \)), matrix \( M \) would be an identity matrix.
understand this, recall that \( A \) is a Poisson transition matrix with rows corresponding to the starting position of a Poisson process and columns corresponding to the finishing position. For each row \( \ell \), the purpose of matrix \( M \) is to take the entries of \( A_{\ell,k} \) that finish in the adjustment region (i.e. columns \( k \notin \mathcal{I} \)) and move them to columns \( k^*(\ell) \) corresponding to the adjustment target. So, whereas transition matrix \( A \) can switch a process into the adjustment region, our updated transition matrix \( AM \) instead switches the process immediately to its corresponding adjustment target. Note that \( AM \) is still a valid Poisson transition matrix for all rows \( \ell \in \mathcal{I} \). In particular, the rows sum to zero and diagonal elements are non-positive (capturing outflows) whereas off-diagonal elements are non-negative (capturing inflows).\(^5\)

Some readers may wonder why the first step is necessary? To see this, consider an initial distribution \( g^0 \) with mass in the adjustment region. Essentially, without step 1, the distribution at all future points in time \( g^n, n = 1, ..., N \) would always keep mass in the adjustment region. This is because matrix \( M \) is specifically constructed so that the columns of \( AM \) contain only zeros for \( k \notin \mathcal{I} \). Therefore the corresponding rows of its transpose \( (AM)^T \) will contain only zeros and hence \( \frac{g^{n+1} - g^n}{\Delta t} = (AM)^T g^{n+1} = 0 \) for all points in the adjustment region, meaning that any mass that starts there will stay there.

How can we find a stationary distribution \( g \)? The simplest strategy is to simply run (12) and (13) forward in time until convergence. This works well in practice. Alternatively, a stationary distribution satisfies

\[
(i) \quad g = M^T g, \quad \text{and} \\
(ii) \quad 0 = (AM)^T g
\]

or

\[
(i) \quad g_\ell = 0 \text{ if } \ell \notin \mathcal{I}, \quad \text{and} \\
(ii) \quad 0 = (AM)^T g
\]

Condition (i) simply ensures \( g \) has no mass in the adjustment region. This additional condition is needed because the rows of matrix \( (AM)^T \) corresponding to points in the adjustment region will contain only zeros.\(^6\)

\(^5\)For rows \( \ell \notin \mathcal{I} \), matrix \( AM \) may have negative off-diagonal elements. However, this will not affect calculations using \( AM \) so long as the density \( g \) has no mass at points \( \ell \notin \mathcal{I} \).

\(^6\)To implement this numerically, we define matrix \( D \) whose elements \( D_{\ell,k} \) are given by

\[
D_{\ell,k} = \begin{cases} 
1, & \text{if } \ell \notin \mathcal{I} \text{ and } \ell = k \\
0, & \text{otherwise}
\end{cases}
\]

Matrix \( D \) contains zeros everywhere except for the elements of the diagonal corresponding to points in the adjustment region, where it contains a 1 (any constant will work). Matrix \( D \) is used to ensure that condition
Finally, consider the case where the policy rules change over time, that is $A(t)M(t)$ are time-dependent. In this case, the time dependent KF equation can be solved by solving the fully time-dependent analogue of (12) and (13), namely

1. Given $g^n$ find $g^{n+\frac{1}{2}}$ from $g^{n+\frac{1}{2}} = (M^n)^T g^n$

2. Given $g^{n+\frac{1}{2}}$ find $g^{n+1}$ from $\frac{g^{n+1} - g^{n+\frac{1}{2}}}{\Delta t} = (A^n M^n)^T g^{n+1}$

where $A^n = A(t^n)$ and $M^n = M(t^n)$, $n = 1, ..., N$.

3 Results

Figure 1 plots the “adjustment region”, i.e. the region of the state space $(a, b)$ in which individuals adjust their portfolios. The adjustment region is in yellow and the non-adjustment region is in blue. Figures 2 and 3 plot the adjustment targets for liquid assets $b$ and illiquid assets $a$ conditional on adjusting. Finally Figure 4 plots the stationary distribution.

![Figure 1: Adjustment and Non-Adjustment Regions](image)

References


$g_\ell = 0$ if $\ell \notin I$, or equivalently $g = M^T g$, is met. Specifically, let $\tilde{A}^T = D + (AM)^T$. One can now solve $0 = \tilde{A}^T g$ exactly as one would solve $0 = A^T g$ when there is no adjustment.


Figure 4: Stationary Distributions, $g_k(a, b)$


