ARC CENTRE OF EXCELLENCE IN
POPULATION AGEING RESEARCH

ARC Centre of Excellence in Population Ageing Research

Working Paper 2023/04

A fast upper envelope scan method for discrete-continuous dynamic programming

Loretti I. Dobrescu and Akshay Shanker

A fast upper envelope scan method for discrete-continuous dynamic programming[*]

Loretti I. Dobrescu[†] and Akshay Shanker [‡]

January 31, 2023

Abstract

We introduce a fast upper envelope scan (FUES) method to solve and estimate dynamic programming problems with discrete and continuous choices. FUES builds on the standard endogenous grid method (EGM). EGM applied to problems with continuous and discrete choices, however, does not by itself generate the optimal solution, as the first order conditions used to retrieve the endogenous grid are necessary but not sufficient. FUES sequentially checks EGM candidate solution points and eliminates those not on the upper envelope of the value correspondence by only allowing discontinuities in the policy function at non-concave regions of the value correspondence. Unlike previous methods used to perform EGM in discrete-continuous dynamic models, FUES does not require the monotonicity of the policy functions. It is also computationally efficient, straightforward to implement, and for sufficiently large EGM grid sizes, guaranteed to recover the optimal solution.

Key Words: discrete and continuous choices, non-convex optimization, Euler equation, computational methods, dynamic programming

JEL Classification: C13, C63, D91

[*] We thank Hazel Bateman, Chris Carroll, Mike Keane, Ben Newell, John Rust, Fedor Ishkakov, Greg Kaplan, Bertel Schjerning, Susan Thorp, Anant Mathur, Mateo Velasquez-Giraldo and conference and seminar participants at several institutions for valuable feedback, discussions and comments. Research support from the Australian Research Council (ARC LP150100608) and ARC Centre of Excellence in Population Ageing Research (CE17010005) is gratefully acknowledged. This research was undertaken with the assistance of resources and services from the National Computational Infrastructure (NCI), which is supported by the Australian Government. The software implementing FUES in our applications, as well as the one providing the function that implements FUES in its general form can be found at here.

[†] School of Economics, University of New South Wales, Australia. Email: dobrescu@unsw.edu.au.

[‡] Corresponding author. School of Finance, University of Sydney, Australia. Email: akshay.shanker@sydney.edu.au.

1 Introduction

Stochastic dynamic programming has become one of the primary tools used by quantitative researchers in fields such as economics, finance, decision theory and artificial intelligence to characterize optimal behavior across time. In such problems, today's decisions affect the payoffs today but also the payoffs and constraints faced by the agent in the future (Bertsekas, 2022; Stachurski, 2022). To solve these models, dynamic programming theory provides us with a general computational procedure (i.e., value function iteration - VFI) that does not require assumptions on the convexity and smoothness of the payoffs and constraints.[1] A drawback of VFI, however, is that it relies on numerical optimization or root-finding methods to iterate on the Bellman equation. In richer, more realistic models that demand additional state spaces, the grid size on which VFI iterates grows exponentially with each added state due to the curse of dimensionality (Rust, 1996).[2] When the grid size of a problem becomes very large, it is either inefficient or practically impossible to use a VFI that relies on numerical methods alone to solve the problem.

In practice, computational efficiency can, however, be gained by using first order conditions (FOCs) such as Euler equations. Euler equations can improve computational speed and accuracy by combining Coleman-Reffett iteration (Coleman, 1990; Reffett, 1996) with the endogenous grid method - EGM (Carroll, 2006).[3] In contrast to VFI, EGM relies on the analytical inversion of the Euler equation and may completely remove costly numerical root-finding or optimization steps (Iskhakov, 2015). The existing dynamic programming theory behind deriving Euler equations requires, however, (i) concavity of the payoffs and transition functions, and (ii) convexity of the feasibility constraints – both conditions jointly referred to as

[1] Under certain boundedness restrictions - see Stachurski (2022).

[2] The high-dimensional economics literature is as large as it is varied, but it ultimately spans two categories of problems. First, high dimensionality arises when the agent chooses a number of quantities - for instance, where the decision space is related to the household portfolio allocation across assets (Kaplan et al., 2020) or to multi-product decisions (Midrigan, 2011). Second, it also arises when a number of different agents are modeled, possibly interacting via equilibrium prices such as, for instance, in Krusell and Smith (1998).

[3] Other methods to use FOCs include the envelope condition method (Maliar and Maliar, 2013; Arellano et al., 2016), and deep learning methods that minimize the Euler equation error (Maliar et al., 2021). While here we present FUES in the context of EGM, FUES and our theoretical results carry over to any solution method that uses FOCs to solve for optimal solutions in a discrete-continuous optimization problem.

the *convexity of the problem*. Unfortunately, real world applications are increasingly posing problems that violate convexity, such as, for instance, in models used to understand frictions (Skiba, 1978; Rust, 1987; Khan and Thomas, 2008; Kaplan and Violante, 2014; Dobrescu et al., 2016; Attanasio et al., 2018; Kaplan et al., 2020), the dynamics of housing stock adjustments (Yogo, 2016; Fagereng et al., 2019), asset pricing in the presence of frictions (Cooper, 2006) or the effect of mortgage refinancing on life-cycle asset allocation (Laibson et al., 2021). Without convexity, EGM generates in such cases a value correspondence with possibly multiple sub-optimal solutions, all satisfying the necessary conditions.

Our contribution is to present a general, efficient, and user-friendly scan method to compute the upper envelope of the value correspondence generated by EGM, and obtain the optimal policy function for dynamic programming problems with discrete and continuous choices. The fast upper envelope scan (FUES) works by noting that the upper envelope of the value correspondence is the supremum of choice-specific concave value functions, with each value function corresponding to a history-dependent future sequence of discrete choices. The convex regions of the upper envelope occur where different choice-specific value functions cross, with the optimal policy function experiencing discontinuous jumps only in these regions. As a result, FUES sequentially checks if the inclusion of a potential optimal point forms a concave or convex region of the upper envelope. If it forms a concave region, the point is eliminated as sub-optimal if it induces a jump in the policy function. If, however, it forms a convex region of the upper envelope, the candidate point is considered optimal and retained.

We start by illustrating FUES in practice via three applications. The first application implements FUES in the context of a one-dimensional problem using the classic example of a model with discrete and continuous choices - i.e., a life-cycle model with discrete retirement and continuous consumption decisions. The second application implements FUES in a portfolio allocation problem where previous EGM methods that calculate upper envelopes in discrete-continuous models (Iskhakov et al., 2017) fail due to the non-monotonicity of the policy function. Featuring both liquid and illiquid savings, this example also shows how our method can be applied to multidimensional problems that do not satisfy the conditions for ‘pure EGM’ (Iskhakov, 2015), by breaking up such problems into a one-dimensional EGM step and a root-finding step. The third application implements FUES in the context of an infinite horizon model with discrete housing and continuous financial wealth that, unlike previously, requires finding a value function that is a fixed point to the Bellman operator.

Next, we provide the theoretical foundation for FUES in its general form and show that if (i) the size of the jumps between policy functions is bounded below (as is the case when the policy functions only contain finitely many jumps), (ii) the grid size is large enough relative to the size of the policy function jumps, and (iii) the endogenous grid is well-behaved around the crossing points between choice-specific value functions, then FUES recovers the optimal value function without error. The first two conditions above ensure FUES can differentiate a jump in the policy function when it induces a shift in the future discrete choices from a smooth change in the policy along a given sequence of choices. The third condition ensures that the optimal and sub-optimal points where value functions intersect are correctly identified. Despite (i)-(iii) being quite abstract to verify in practice, FUES continues to perform well even at lower grid sizes in all the applications we investigated.

FUES advances considerably the latest methods to solve discrete-continuous dynamic models proposed in both Fella (2014) and Iskhakov et al. (2017). These methods can successfully compute the upper envelope of the choice-specific value functions[4] but, as mentioned, they rely on the monotonicity of the optimal policy functions. While Iskhakov et al. (2017) does not suggest any way to circumvent non-monotonicity, Fella (2014) identifies optimal points by using the numerical solution of the Bellman equation in non-concave regions of the value correspondence. Doing so, however, can be computationally costly and thus impractical to use in problems with multiple states and large grids. In this respect, our method advances the literature in three ways. First, FUES does not require the monotonicity of any of the policy functions. Second, we prove the efficacy of FUES in the general case, which implies that FUES can be applied as a black-box method to identify upper envelopes. Third, FUES is considerably easier to implement and does not require knowing the shape of the policy functions, which is very useful in large, high-dimensional models.

Turning to the broader literature, the scan method we present here is informed by

[4] See also Druedahl and Jørgensen (2017) that uses triangulation to construct the upper envelope.

the methods used to calculate convex hulls of points (Graham, 1972). However, in the context of the existing literature and up to our best knowledge, we are the first to propose a scanning algorithm to identify upper envelopes. Additionally, dynamic programming problems with discrete and continuous choices are a special case of mixed integer non-linear programming problems, where discrete and continuous choices optimize an arbitrary function. Relatively recent studies have characterized both sufficient and necessary first and second order conditions for classes of such problems (Jeyakumar et al., 2007; Jeyakumar and Srisatkunarajah, 2009), but so far, a condition that is both sufficient and necessary in the general setting remains elusive. Rather than derive sufficient first or second order conditions, our method recovers the optimal solution by computationally approximating the upper envelope of those points that satisfy the necessary conditions.

The paper proceeds as follows. In Section 2, we start by solving and discussing our main results in the context of three well-known applications. Section 2.1 introduces FUES informally using the simple retirement choice model in Iskhakov et al. (2017) where agents choose their savings and labor force participation; here we also compare the performance of FUES to that of the DC-EGM method proposed by Iskhakov et al. (2017). Section 2.2 further demonstrates FUES in an application where DC-EGM fails, using a model in which agents choose whether or not to adjust their illiquid housing stock. Finally, Section 2.3 showcases FUES in the discrete choice model proposed by Fella (2014) where agents decide to hold liquid and illiquid assets via discrete choices. In Section 3, we follow our study of these applications by formally stating FUES in its general form and providing the proofs on how FUES accurately obtains the upper envelope without error. Section 4 concludes.

2 Illustrative applications

This section illustrates FUES using well-known dynamic optimization applications. We will briefly introduce each problem, discuss how their discrete choices result in non-convexity, and detail how FUES can be implemented to retrieve the optimal solution.

2.1 Application 1: Finite horizon retirement choice model

Let us start from the finite horizon retirement and savings choice model in Iskhakov et al. (2017).

2.1.1 Model environment

Consider an agent that consumes, works (if they so choose) and saves from time $t = 0$ until time $t = T$. At the beginning of each period, the agent starts as a worker or retiree, with the state variable denoting their beginning-of-period work status given by the discrete variable d_t. If the agent works, they earn a per-period wage y. Every period, the agent can choose to continue working during the next period by setting $d_{t+1} = 1$, or to permanently exit the workforce by setting $d_{t+1} = 0$. If the agent chooses to work the next period, they will incur a utility cost δ at time t. We assume all agents start as workers so $d_0 = 1$. Agents can also consume c_t and save in capital a_t, with $a_t \in S$ and $S\colon = [0, \bar{a}] \subset \mathbb{R}_+$. The intertemporal budget constraint is:

$$a_{t+1} = (1+r)a_t + d_t y - c_t \tag{1}$$

Per-period utility is given by $\log(c_t) - \delta d_t$. Letting the function u be defined by $u(c) = \log(c)$, the agent's maximization problem becomes:

$$V_0^{d_0}(a_0) = \max_{(c_t, d_{t+1})_{t=0}^{T}} \left\{ \sum_{t=0}^{T} \beta^t u(c_t) - \delta d_{t+1} \right\} \tag{2}$$

subject to Equation (1), $a_t \in S$ for each t, and the fact that the agent cannot return to work after retiring (i.e., $d_{t+1} = 0$ if $d_t = 0$). Let $V_t^{d_t}$ denote the beginning of period value function. If the agent enters the period as a worker, the agent's time t value function will be characterised by the Bellman equation:

$$V_t^1(a) = \max_{c, d' \in \{0,1\}} \left\{ u(c) - d'\delta + \beta V_{t+1}^{d'}(a') \right\} \tag{3}$$

where $a' = (1+r)a + y - c$ and $a' \in S$. If the agent enters the period as a retiree, the agent's value function becomes:

$$V_t^0(a) = \max_c \left\{ u(c) + \beta V_{t+1}^0(a') \right\} \tag{4}$$

with $a' = (1+r)a - c$. The optimization problem for the retiree is a standard concave problem. For the worker, however, the optimization problem is not concave since she optimizes jointly a discrete choice and a continuous choice. Moreover, even conditional on $d' = 1$, the next period value function V_{t+1}^1 will not be concave since the value function represents the supremum over *all future feasible combinations of discrete choices*. The non-concavity of V_{t+1}^1 produces the 'secondary kinks' described in Iskhakov et al. (2017).

To see how the choice at time t implicitly controls the future sequence of discrete choices and produces the secondary kinks, write the time t worker's value function as:

$$V_t^1(a) = \max_c \max_{\mathbf{d} \in \mathbb{D}} \left\{ u(c) - d'\delta + \beta Q_{t+1}^{\mathbf{d}}(a') \right\} \tag{5}$$

where $Q_{t+1}^{\mathbf{d}}$ is the $t+1$ value function conditional on a given sequence of future discrete choices $\mathbf{d}$, with $\mathbf{d} = \{d', d'', \dots\}$. In particular, we have:

$$Q_{t+1}^{\mathbf{d}}(a) = \max_{(c_k)_{k=t+1}^T} \left\{ \sum_{k=t+1}^{T} \beta^t u(c_k) \right\} \tag{6}$$

subject to Equation (1), $a_t \in S$ for each t, and holding the sequence $\mathbf{d}$ fixed. The set $\mathbb{D}$ contains all feasible sequences of discrete choices that can be made from t to T. By writing the Bellman equation as above at equation (5), we are able to see how the maximand on the RHS of Equation (5), for a given sequence of discrete choices $\mathbf{d}$, will be concave. However, the max operator over the discrete choices does not preserve concavity, and so V_t^1 will not be concave. Rather, V_t^1 will be the upper envelope of overlapping concave functions, with each concave function corresponding to a different sequence of future discrete choices. Figure 1 characterises such a situation, where the upper envelope of concave functions is not concave.

To sum up, value function non-concavity, even holding the choice d' fixed, is brought on by the implicit changes in the entire future sequence of discrete choices

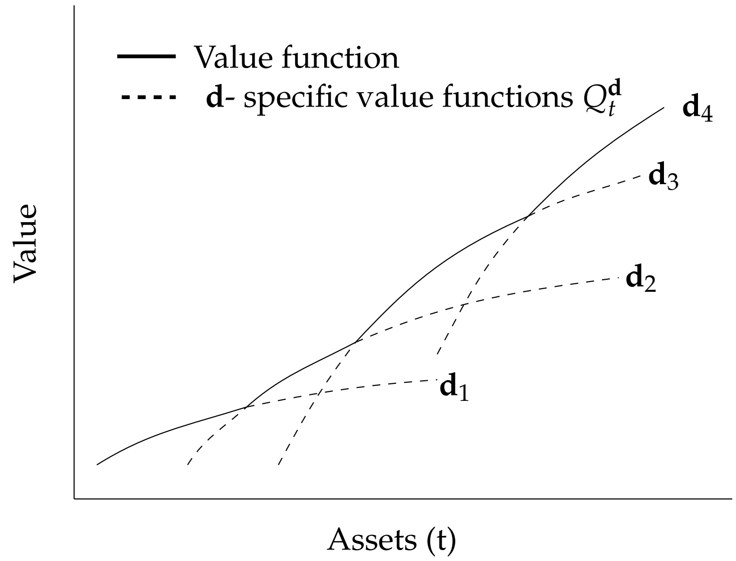

Figure 1: The time t worker value function V_t^1 is the upper envelope of concave functions, where each concave function is a value function conditional on a *sequence* of future discrete choices. The subscript 'i' on $\mathbf{d}_i$ indicate distinct sequences of discrete choices.

as one controls the choice variable c in Equation (5). In this case, the Bellman equation, Equation (3), still holds and one can numerically implement VFI to compute a solution. The challenge arises when, in high-dimensional models, solving the Bellman equation using numerical methods becomes burdensome computationally. A more computationally efficient strategy involves recovering the policy function by solving for points that satisfy the FOCs (i.e., the Euler equations) of the Bellman equation. However, since the upper envelope is not concave, the points satisfying the FOCs could be associated with any future sequence of discrete choice in Figure 1, and may not be on the upper envelope.

2.1.2 The Euler equation

We now discuss the FOCs and then proceed to our contribution (i.e., the FUES) as a way to use necessary first order information to efficiently compute V_t. If the agent chooses $d_{t+1} = 1$ (i.e., they continue as a worker in $t+1$), we can write the time t worker Euler equation as:

$$u'(c_t^1) \geq \beta(1+r)u'(c_{t+1})$$

where c_t^1 is the time t consumption policy conditional on $d_{t+1} = 1$, while c_{t+1} is the unconditional time $t+1$ consumption policy. On the other hand, if the agent chooses $d_{t+1} = 0$ (i.e.,they retire), then the Euler equation is:

$$u'(c_t^0) \geq \beta(1+r)u'(c_{t+1}^0)$$

Functional Euler equations. It will now be helpful to write the Euler equation in its functional form. Let $\sigma_t^d \colon S \times \{0,1\} \to \mathbb{R}_+$ be the conditional asset policy function for the worker at time t if $d = 1$ and for the retiree if $d = 0$. We call σ_t^d the conditional policy because it will depend, through its second argument, on the discrete choice - to work or not to work in $t+1$ - made by the worker at time t. The time t and time $t+1$ policy functions will satisfy the functional Euler equation:

$$u'((1+r)a + dy - \sigma_t^d(a,d')) \geq \beta(1+r)u'((1+r)\sigma_t^d(a,d') + d'y - \sigma_{t+1}^{d'}(a',d'')) \quad (7)$$

where $a' = \sigma_t^d(a,d')$. On the choice of whether to work or not, the time t worker will chose $d_{t+1} = 1$ if and only if:

$$\begin{aligned} &u((1+r)a + y - \sigma_t^1(a,1)) - \delta + \beta V_{t+1}^1(\sigma_t^1(a,1)) \\ &\qquad > u((1+r)a + y - \sigma_t^1(a,0)) + \beta V_{t+1}^0(\sigma_t^1(a,0)) \end{aligned} \quad (8)$$

Since the discrete choice is itself a function of the state, we can also define a discrete choice policy function $\mathcal{I}_t \colon S \times \{0,1\} \to \{0,1\}$. As such, we will have $d' = \mathcal{I}_t(a,d)$ and $d'' = \mathcal{I}_{t+1}(a',d')$, where $\mathcal{I}_t$ is evaluated to satisfy (8) each period conditional on the $t+1$ value function.

2.1.3 Computation using EGM and FUES

We now turn to how FUES can identify the upper envelope from a set of points that satisfy the Euler equations. Fix a time t and suppose the value function V_{t+1}^d, the discrete choice function $\mathcal{I}_{t+1}$ and the optimal policy function σ^{d}_{t+1} for $d = 0$ and $d = 1$ are known. Let $\hat{\mathbb{X}}_t$, $\hat{\mathbb{V}}_t$ and $\hat{\mathbb{X}}'_t$ be sequences of points satisfying the Euler equation for workers:

$$u'((1+r)\hat{x}_i + dy - \hat{x}'_i) = \beta(1+r)u'((1+r)\hat{x}'_i + yd' - \sigma^{d'}_{t+1}(\hat{x}'_i, d'')) \tag{9}$$

$$\hat{v}_i = u((1+r)\hat{x}_i + dy - \hat{x}'_i) - d\delta + V^{d}_{t+1}(\hat{x}_i) \tag{10}$$

where $d'' = \mathcal{I}_{t+1}(\hat{x}'_i, d')$, $\hat{x}_i \in \hat{\mathbb{X}}_t$, $\hat{v}_i \in \hat{\mathbb{V}}_t$ and $\hat{x}'_i \in \hat{\mathbb{X}}_t$. Such a sequence of points can be generated analytically using EGM. In particular:

$$(1+r)\hat{x}_i + dy - \hat{x}'_i = u'^{,-1}\left[\beta(1+r)u'((1+r)\hat{x}'_i + yd' - \sigma^{d'}_{t+1}(\hat{x}'_i, d''))\right] \tag{11}$$

In the case of the EGM, $\hat{\mathbb{X}}_t$ is the endogenous grid of points, $\hat{\mathbb{X}}'_t$ is the exogenous one, and $\hat{\mathbb{V}}_t$ is the value correspondence.

Next, order the points in $\hat{\mathbb{X}}_t$, $\hat{\mathbb{V}}_t$ and $\hat{\mathbb{X}}'_t$ in ascending order of the *endogenous grid points* $\hat{\mathbb{X}}_t$. Consider the left panel of Figure 2 as a stylised plot of the endogenous grid points $\hat{\mathbb{X}}_t$ and associated continuation payoffs $\hat{\mathbb{V}}_t$ generated by EGM. Assume for the purpose of illustration that the EGM points are associated with two overlapping future choice-specific value functions. The right panel of Figure 2 displays the policy functions (i.e., next period assets) associated with each future choice-specific value function. Pick a point $\hat{x}_i$, with $\hat{x}_i \in \hat{\mathbb{X}}_t$ such that $\hat{x}'_i$ is optimal given $\hat{x}_i$ and it lies on the upper envelope. Note if the points $\hat{x}_{i+1}$ and $\hat{x}'_{i+1}$ imply a different future sequence of discrete choices to $\hat{x}_i$ and $\hat{x}'_i$, then $\hat{x}'_{i+1}$ will experience a 'discontinuous jump' from $\hat{x}'_i$. However, for $\hat{x}'_{i+1}$ to be on the upper envelope, it must be that $\hat{x}'_{i+1}$ can only jump if it occurs after the crossing point between two value functions (for instance, the point $\hat{x}_6$). That is, $(\hat{x}'_{i+1}, \hat{v}_{i+1})$ can only jump if it makes a *convex* 'left turn' from the line joining $(\hat{x}'_{i-1}, \hat{v}_{i-1})$ and $(\hat{x}'_i, \hat{v}_i)$. On the other hand, if $(\hat{x}'_{i+1}, \hat{v}_{i+1})$ makes a *concave* 'right turn', it cannot jump for it to be on the upper envelope. The reason is that if $(\hat{x}'_{i+1}, \hat{v}_{i+1})$ has made a right turn, for $(\hat{x}'_{i+1}, \hat{v}_{i+1})$ to be on the upper envelope, it must be on the *concave value function* yielding the same future sequence of discrete choices $\mathbf{d}$ as implied by $(\hat{x}'_i, \hat{v}_i)$. If a right turn is associated with a jump (e.g., point $\hat{x}_7$), then it must be on a value function associated with a sub-optimal set of future discrete choices. We formally prove this argument in Section 3. Informally, we can use the intuition from Figure 2 to implement the FUES method as follows (see Section 3 for a formal pseudo-code):[5]

Box 1: FUES method

1. Compute $\hat{\mathbb{X}}_t$, $\hat{\mathbb{V}}_t$ and $\hat{\mathbb{X}}'_t$ using standard EGM.
2. Set a pre-determined 'jump detection' threshold $\bar{M}$.
3. Sort all sequences in order of the *endogenous* grid $\hat{\mathbb{X}}_t$.
4. Start from point $i = 2$. Compute $g_i = \frac{\hat{v}_i - \hat{v}_{i-1}}{\hat{x}_i - \hat{x}_{i-1}}$ and $g_{i+1} = \frac{\hat{v}_{i+1} - \hat{v}_i}{\hat{x}_{i+1} - \hat{x}_i}$.
5. If $|\frac{\hat{x}'_{i+1} - x'_i}{\hat{x}_{i+1} - \hat{x}_i}| > \bar{M}$ and a right turn is made ($g_{i+1} < g_i$), then remove point $i+1$ from grids $\hat{\mathbb{X}}_t$, $\hat{\mathbb{V}}_t$ and $\hat{\mathbb{X}}'_t$. Otherwise, set $i = i+1$.
6. If $i + 1 \leq |\hat{\mathbb{X}}_t|$, then repeat from step 5.

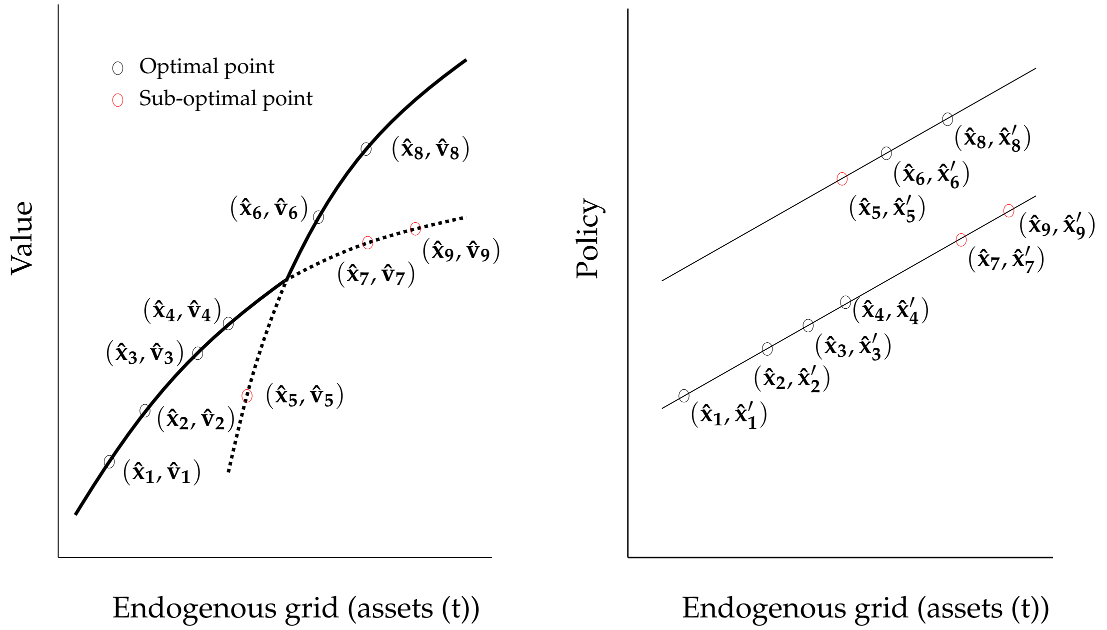

Figure 2: FUES eliminates points that cause a concave 'right turn' (left panel) from an optimal point and also cause a discontinuous jump in policy (right panel).

[5] Note that it may be possible to analytically derive an approximate jump detection threshold value ($\bar{M}$) and a grid size value based on the theoretical curvature of the policy function at each point using the value of the marginal utility and the future policy function. In practice, we find that selecting a reasonable value for $\bar{M}$ based on experimentation and verifying that the upper envelope is recovered works just as well.

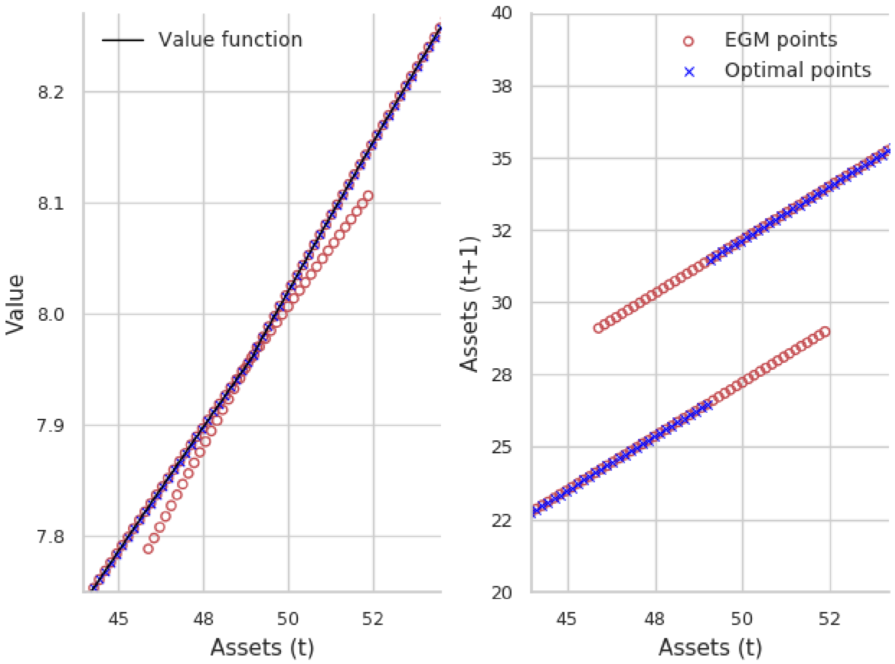

Figure 3: Value correspondence and optimal points for $t = 17$. Parameters from Iskhakov et al. (2017), Figure 3.

The method will yield a set of refined grids $\mathbb{X}_t$, $\mathbb{V}_t$ and $\mathbb{X}'_t$. To address the occasionally binding lower bound asset constraint, we follow the approach by Iskhakov et al. (2017). The policy and value functions can then be interpolated over these grids to yield the time t approximated solution for the worker who chooses to continue working. The retiree value and policy functions can be calculated using standard EGM since the retiree problem is concave at each time t. Once the retiree problem is solved for time t, the discrete choice at time t of whether or not to work at time $t+1$ can be evaluated. The procedure can then be repeated at $t-1$ as per standard backward policy iteration.

Numerical example. We apply FUES to solve the model studied in Iskhakov et al. (2017), with Figure 3 showing how FUES removes sub-optimal points in the value function (left panel) and selects the optimal policy function (right panel). Figure 4 plots the policy functions for workers at different ages, as a direct comparison to

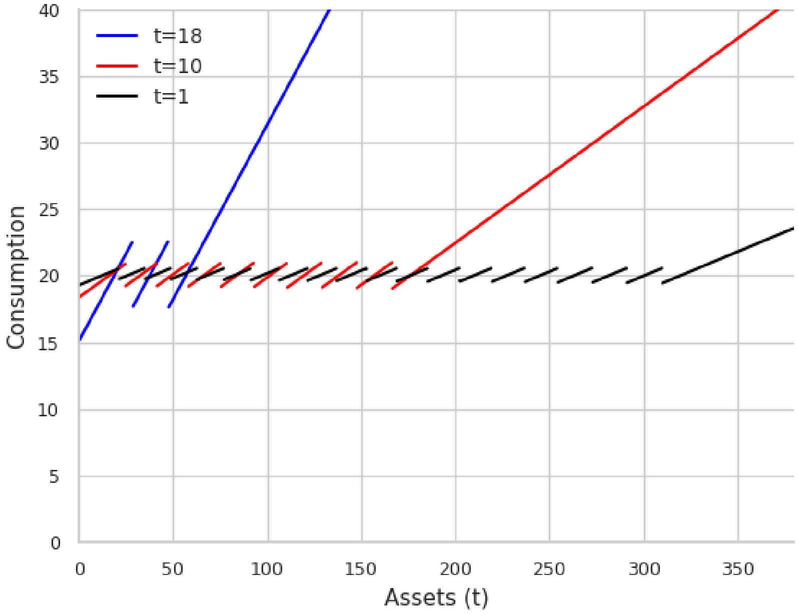

Figure 4: Optimal consumption functions for workers. Parameters from Iskhakov et al. (2017), Figure 3.

Figure 3 in Iskhakov et al. (2017).[6] Despite a modest grid size of 2,000 points and performing only naive linear interpolation, we find that FUES is able to accurately pick up the upper envelope of the future choice-specific value functions, and also accurately replicate the shape of the consumption functions. Moreover, the added computational time involved in running FUES is only a small proportion of the overall EGM time. This means that FUES has inherited EGM's computational efficiency, which has been shown to be orders of magnitudes higher than that of VFI (Iskhakov et al., 2017).[7]

For comparison, we also compute the upper envelope using DC-EGM as in Iskhakov et al. (2017) using the code libraries provided by Carroll et al. (2018). For a 100^3 grid of parameter values for β, r, and y and holding the jump detection threshold $\bar{M}$ fixed, we find that FUES recovers an identical solution to DC-EGM for each

[6] In doing this comparison, note that we retain liquid assets as a state variable (since we do not need to prove monotonicity), while plot Figure 3 in Iskhakov et al. (2017) uses total wealth - i.e., the sum of liquid assets, asset returns and wages - as a state variable.

[7] On a single core Intel Xeon 'Cascade Lake' CPU, the overall EGM time to compute the policy functions for 20 years was 1.2 seconds, while FUES only took 0.3 seconds.

parameter value.[8]

2.1.4 Forward and backward scans

So far, we assumed that if a point $(\hat{x}_{i+1}, \hat{v}_{i+1})$ makes a left turn from point $(\hat{x}_i, \hat{v}_i)$, then it is a sufficient and *necessary* condition for the point $(\hat{x}_{i+1}, \hat{v}_{i+1})$ to lie after a cross-point of value functions between itself and $(\hat{x}_i, \hat{v}_i)$. However, a point $(\hat{x}_{i+1}, \hat{v}_{i+1})$ may lie after a crossing point and yet not generate a left turn with respect to point x_i. This situation is presented in the left panel of Figure 5. We also assumed that the first point after a crossing point $(\hat{x}_{i+1}, \hat{v}_{i+1})$ must be on the optimal choice-specific value function. However, a selected point may be the first point after a crossing between choice-specific value functions and not be optimal. This situation is presented in the right panel of Figure 5, where the point $(\hat{x}_i, \hat{v}_i)$ is sub-optimal but will not be removed by the basic FUES method since it is on the same discrete choice as the optimal point right before the crossing point.

To rectify the issue seen in the left panel of Figure 5, we can implement a forward scan before a point is eliminated. A forward scan picks a point $\hat{x}_q$ to the right of $\hat{x}_{i+1}$ that is on the same value function as $\hat{x}_i$. The point $\hat{x}_q$ can be found by checking points after $i+1$ on the endogenous grid and finding the first point whose policy function does not jump from the point $\hat{x}_i$. We can then check to see if the point $\hat{x}_{i+1}$ dominates the line segment joining $(\hat{x}_i, \hat{v}_i)$ to $(\hat{x}_q, \hat{v}_q)$, drawn as g_w in the left panel of Figure 5. If $\hat{x}_{i+1}$ dominates the line segment, it means $\hat{x}_{i+1}$ lies after a crossing and must be included as an optimal point.

[8] To independently check FUES's sensitivity to various parameter values, use the code available here.

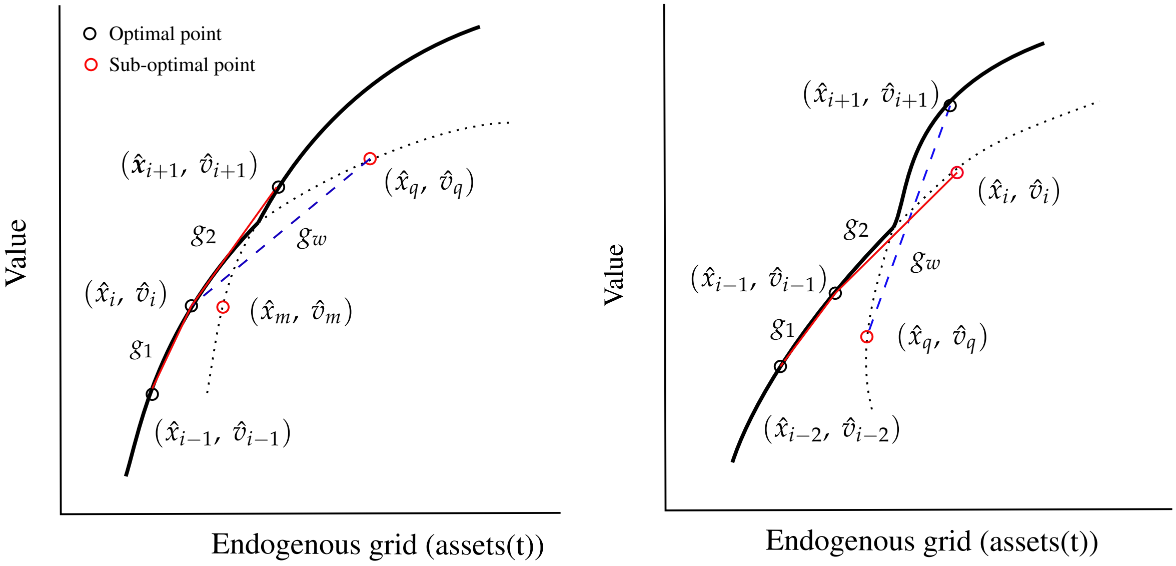
Optimal point
Sub-optimal point
Value
$(\hat{x}_{i+1},\ \hat{v}_{i+1})$
$(\hat{x}_q,\ \hat{v}_q)$
g_2
g_w
$(\hat{x}_i,\ \hat{v}_i)$
$(\hat{x}_m,\ \hat{v}_m)$
g_1
$(\hat{x}_{i-1},\ \hat{v}_{i-1})$
Endogenous grid (assets(t))
Value
$(\hat{x}_{i+1},\ \hat{v}_{i+1})$
$(\hat{x}_i,\ \hat{v}_i)$
g_2
g_w
$(\hat{x}_{i-1},\ \hat{v}_{i-1})$
g_1
$(\hat{x}_q,\ \hat{v}_q)$
$(\hat{x}_{i-2},\ \hat{v}_{i-2})$
Endogenous grid (assets(t))

Figure 5: Forward and backward scans to improve accuracy of FUES around crossing points.

To rectify the issue seen in the right panel of Figure 5, we can implement a backward scan after making a left turn. A backward scan picks a point $\hat{x}_q$ to the left of $\hat{x}_i$ that is on the same value function as $\hat{x}_{i+1}$. The point $\hat{x}_q$ can be found by checking points before i on the endogenous grid and finding the first point whose policy function does not jump from the point $\hat{x}_{i+1}$. We can then check to see if the point $\hat{x}_i$ is dominated by the line segment joining $(\hat{x}_{i+1}, \hat{v}_{i+1})$ to $(\hat{x}_q, \hat{v}_q)$, drawn as g_w in the right panel of Figure 5. If $\hat{x}_i$ is dominated by the line segment, then it means $\hat{x}_i$ lies after a crossing and must not be included as an optimal point.

Finally, note that the forward and backward scan procedures can be used to attach approximations of the choice-specific value functions crossings to the endogenous grid. For instance, in the right panel of Figure 5, a crossing point can be attached by taking the intersection of the line segment g_w with the line segment g_2.

2.1.5 Requirements for FUES accuracy and performance on lower grid sizes

Let us now briefly discuss the conditions required for FUES to guarantee it deletes all sub-optimal points, but only these ones. We present a formal proof and state conditions under which FUES recovers the upper envelope in Section 3.

First, note that to guarantee a zero approximation error, the formal FUES theory requires that the pathologies discussed in Section 2.1.4 do not occur. In practice, even if this does not hold, we still see the forward and backward scan performing well in approximating the optimal points around choice-specific value function crossings.

Second, the formal treatment of FUES also requires the jumps between two choice-specific policy functions to be detectable. Note that every time we move between the two segments of the policy function in Figure 3, a large jump in x' occurs. For FUES to guarantee that an optimal point is picked up, one condition we require is that the jump size between all policies generated by different discrete choice combinations is sufficiently large compared to the distance between the endogenous grid points. This ensures that when a jump to a sub-optimal point occurs, it is registered by FUES. Moreover, policy functions are all required to be smooth and have a common Lipshitz constant, which ensures that changes along a policy function, given a sequence of discrete choices, are not so large as to incorrectly delete optimal points. The Lipshitz condition on the problem here is straightforward to satisfy, since there are only finitely many policy functions at any given time t, and each one is smooth given a future sequence of discrete choices.

Despite the above requirement of a sufficiently dense endogenous grid, we find FUES performs well on lower grid sizes too. Figures 10 - 13 compare the upper envelopes of FUES with DC-EGM, with grid sizes as small as 200, for $t = 17$ and using the baseline Iskhakov et al. (2017) parameters. Even though lower grid sizes result in a poorer approximation of the discrete jump point, as can be expected with any EGM method that uses a uniform exogenous grid, we find that FUES performs as well as DC-EGM. Indeed, for each grid size we explore, the FUES upper envelope matches DC-EGM.

One limitation of the proofs we present in Section 3 is that when there are infinitely (uncountably) many possible discrete choices, it may not be possible to show that jump sizes are bounded below. With the addition of taste shocks that generate a smoothed version of the model presented here by incorporating uncountably many future discrete choices, we see, however, that jump sizes are sufficiently large. In such cases, FUES is then able to successfully pick the optimal grid points (see Figures 8 - 9 in the Appendix).

2.2 Application 2: Continuous housing stock with adjustment frictions

Having introduced FUES via a simple application, we now turn to demonstrate it in a housing frictions example where the DC-EGM method outlined by Iskhakov et al. (2017) cannot be applied due to the non-monotonicity of the policy function. Interestingly, the example in this section also features two dimensions of savings, one for liquid assets and one for illiquid (housing) assets. Despite being a veritable workhorse in the housing frictions literature (Kaplan and Violante, 2014; Yogo, 2016; Dobrescu et al., 2022), the conditions that Iskhakov (2015) requires for multidimensional 'pure EGM' do not hold here, with the set of Euler equations not analytically invertible. As such, this example also shows how one-dimensional EGM and FUES can be practically applied to a multidimensional setting where the Iskhakov (2015) conditions fail. The solution strategy will involve numerically obtaining all the roots of the liquid assets Euler equation along a single dimension, and then evaluate the optimal non-monotonic housing policy, and implicitly the optimal liquid asset policy, using EGM and FUES.[9]

2.2.1 Model environment

To start, let a non-negative liquid asset (i.e., financial wealth) be denoted by a_t and a non-negative housing asset be denoted by H_t. The liquid asset earns a rate of return r and for simplicity we assume housing earns no returns. Investments can be made in and out of the stock of a_t without friction. However, adjusting the housing stock to a value H_{t+1} requires a payment of τH_{t+1}, with $\tau > 0$. In each period t, with $t = 0, 1 \ldots, T$, the agent consumes non-housing goods c_t, and invests a total of H_{t+1} in the housing stock and a_{t+1} in the liquid stock. The agent also makes a discrete choice d_t, where investment in and out of the housing stock can only be made if $d_t = 1$; otherwise if $d_t = 0$ then $H_{t+1} = H_t$. Finally, in each period, the agent earns a stochastic wage y_t and we assume $(y_t)_{t=0}^{T}$ is a finite,

[9] We point out that non-monotonicity and multidimensionality are in general separate issues. The inapplicability of DC-EGM (Iskhakov et al., 2017) arises not because of the failure of the multidimensional conditions in Iskhakov (2015). Moreover multiple dimensions and failure of Iskhakov (2015) are not necessary conditions for non-monotonicity to arise in a policy function.

Markov process.

The following budget constraints will hold for each t. First, total investments and consumption cannot exceed total available wealth each period:

$$y_t + (1+r)a_t + d_t H_t \geq c_t + a_{t+1} + (1+\tau)H_{t+1} \tag{12}$$

Second, in terms of payoffs, the agent lives up to time T, after which they die and value the bequest they leave behind according to a function $\theta\colon \mathbb{R}_+ \to \mathbb{R} \cup \{-\infty\}$. Per-period utility is given by a real-valued function $\varphi_t^u\colon \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R} \cup \{-\infty\}$ defined as:

$$\varphi_t^u(c_t, H_{t+1}) = \mathbb{1}_{t \leq T} u(c_t, H_{t+1}) + \mathbb{1}_{t=T+1}\theta(c_{T+1}) \tag{13}$$

where u is a concave, jointly differentiable, increasing function. In the per-period utility function, the agent earns payoffs from non-housing consumption c_t and the housing stock available at the end of the period, H_{t+1}.[10] Formally, the agent's dynamic optimization problem becomes:

$$V_0(a_0, H_0, y_0) = \max_{(a_t, H_t, d_t)_{t=0}^{T+1}} \sum_{t=0}^{T+1} \beta^t \mathbb{E}\varphi_t^u(c_t, H_{t+1}) \tag{14}$$

such that (12) and (13) hold, $H_t \geq 0$ and $a_t \geq 0$ for each t and a_0, H_0 and y_0 are given. The expectation above is taken over the wage process $(y_t)_{t=0}^T$. The sequential problem implies the following recursive Bellman equation:

$$V_t(a, H, y) = \max_{a', H', d} \left\{ \varphi_t^u(c, H') + \beta \mathbb{E}_y V_{t+1}(a', H', y') \right\} \tag{15}$$

where the prime notation indicates next period state values satisfying the budget constraint, and $c = y_t + (1+r)a + dH - a' - d(1+\tau)H'$. In the Bellman equation, expectations are now conditional on time t realization of the wage process y.

One final remark. In contrast to Application 1, the application here features uncertainty. The solutions to the sequential problem in Equation (14) provide the paths of the discrete choices. Within a path, each period's discrete choice depends on the history of shocks up to that time. However, we assume the exogenous shock

[10] See Yogo (2016) for a similar formulation.

takes on finitely many values each period. There are thus finitely many possible future sequences of history-dependent discrete choices that can be collected in a finite set $\mathbb{D}$, similar to Application 1 (see discussion of Equation (5)). As a result, the intuition related to what causes the secondary kinks we discussed via Figure 1 carries over to the context of this application too.

2.2.2 The Euler equations

The problem will feature two Euler equations, one for each state. For periods prior to the terminal one, the Euler equation for the liquid assets will be:

$$u_1(c_t, H_{t+1}) \geq \beta(1+r)\mathbb{E}_t u_1(c_{t+1}, H_{t+2}) \tag{16}$$

where we have used the subscript '1' to refer to the first partial derivative of u and will use the subscript '2' to refer to the second partial derivative of u. If $d_t = 1$, the Euler equation for the housing stock will be:

$$(1+\tau)u_1(c_t, H_{t+1}) \geq \overbrace{\mathbb{E}_t \sum_{k=t}^{\iota-1} \beta^{t-\iota} u_2(c_k, H_{k+1})}^{\text{Marginal value of housing services stream}} + \underbrace{\mathbb{E}_t \beta^{t-\iota} \left(u_1(c_\iota, H_{\iota+1})\right)}_{\text{Marginal value of liquidating housing at time } \iota} \tag{17}$$

The intuition of the Euler equation (16) for the liquid stock is standard. The Euler equation for the housing stock (17), however, features a stochastic time subscript ι, defined as the next time period when $d_\iota = 1$. Since the next time the stock is adjusted will be stochastic, ι becomes a random stopping time. The Euler equation for the housing stock then tells us that the shadow value (price) of investment (or withdrawal) from the housing stock is given by the discounted expected value of the stock *when the stock is next liquidated*,[11] along with the stream of housing services provided up to the time of liquidation.

[11] See Section 4.1 in Kaplan and Violante (2014) for a similar intuition, and Appendix C.3 in Dobrescu et al. (2022) for a formal treatment.

Functional Euler equations. Since the solution sequence for the problem will be recursive, there exists measurable functions σ_t^a, σ_t^H and $\mathcal{I}_t$ such that $H_{t+1} = \sigma_t^H(a_t, H_t, y_t)$, $a_{t+1} = \sigma_t^a(a_t, H_t, y_t)$ and $d_t = \mathcal{I}_t(a_t, H_t, y_t)$ for each t. Also let $\sigma_t^{a,d}$ and $\sigma_t^{H,d}$ denote the choice-specific policy functions conditional on the time t discrete choice $d \in \{0, 1\}$. Inserting the policy functions back into (16) and (17) yields the functional Euler equation for the housing stock:

$$\begin{aligned} u_1(c, H')(1+\tau) \geq \beta \mathbb{E}_y \mathcal{I}_{t+1}(a', H', y') u_1(c', H'') + u_2(c, H') \\ + (1 - \mathcal{I}_{t+1}(a', H', y'))\beta \left[\mathbb{E}_y \Theta_{t+1}(a', H', y') + u_2(c', H')\right] \end{aligned} \tag{18}$$

where:

$$\begin{aligned} c' = (1+r)\sigma_t^{a,1}(a, H, y) + \sigma_t^{H,1}(a, H, y) + y' \\ - \sigma_{t+1}^{a,1}(a', H', y') - \sigma_{t+1}^{H,1}(a', H', y')(1+\tau) \end{aligned} \tag{19}$$

$$c = (1+r)a + H + y - \sigma_t^{a,1}(a, H, y) - \sigma_t^{H,1}(a, H, y)(1+\tau) \tag{20}$$

$$H'' = \sigma_{t+1}^{H,1}(a', H', y) \tag{21}$$

and we have $a' = \sigma_t^{a,1}(a, H, y)$ and $H' = \sigma_t^{H,1}(a, H, y)$. The additional term Θ_{t+1} is a multiplier denoting the continuation marginal value of the housing stock if the time $t+1$ stock is not adjusted. For a set of recursive $t+1$ recursive policy functions, we can compute Θ_t as a function of the states as follows:

$$\begin{aligned} \Theta_t(a, H, y) = \mathbb{E}_y \mathcal{I}_{t+1}(a', H', y') + u_2(c, H') \\ + \beta \mathbb{E}_y (1 - \mathcal{I}_{t+1}(a', H', y')) \left[\Theta_{t+1}(a', H', y') + u_2(c', H')\right] \end{aligned} \tag{22}$$

The functional Euler equation for the liquid assets stock becomes:

$$u_1(c, H') \geq \beta(1+r)\mathbb{E}_y u_1(c', H'') \tag{23}$$

with c, c' H' defined analogously to (19) and (20), and where if the time t discrete choice is not to adjust, then $d = 0$.

2.2.3 Computation using EGM and FUES

Suppose we know V_{t+1}, $\{\sigma^{a,d}_{t+1}, \sigma^{H,d}_{t+1}\}_{d\in\{0,1\}}$, $\mathcal{I}_{t+1}$ and Θ_{t+1}. We can first apply standard EGM with FUES to evaluate $\sigma^{a,0}_{t}$ for non-adjusters since only one Euler equation - i.e., equation (23) - will hold. For each possible time t housing state H_t in the housing grid and exogenous state y_t, we can approximate $\sigma^{a,0}_{t}(\cdot, H_t, y_t)$ by first setting an exogenous grid of a' values (holding $H' = H_t$ fixed since no adjustment takes place) and then creating an endogenous grid of t period liquid assets using Equation (23) and the budget constraint (12). FUES can then be applied as described in Section 2.1.3 to eliminate the sub-optimal points, allowing us to obtain an approximation of the policy function $\sigma^{a,0}_{t}$.[12]

Let us now calculate the housing policy functions for adjusters. First, note we can evaluate consumption today c, for a given value of H' and a' as follows:

$$c = u_1^{-1}((1+r)\beta\mathbb{E}_y u_1(c', H''), H') \tag{24}$$

where u_1^{-1} is the analytical inverse of u_1 in its first argument. The procedure to evaluate the policy functions for adjusters is as follows:

[12] Once again, to address occasionally binding constraints, we follow the approach Iskhakov et al. (2017).

Box 2: FUES and EGM for adjuster policy function

1. Fix y, fix an exogenous grid over H', $\tilde{\mathbb{H}}'$ and initialise empty arrays for the housing policy ($\hat{\mathbb{H}}'$), current period value ($\hat{\mathbb{V}}$), liquid assets policy function ($\hat{\mathbb{A}}'$) and endogenous wealth grid ($\hat{\mathbb{X}}$).
2. For each $\hat{h}_i \in \tilde{\mathbb{H}}'$:
   - (i) Evaluate the P multiple roots to Equation (18) in terms of a', with c evaluated by Equation (24) and H' fixed as $\hat{h}_i$; collect the roots in a tuple $(\hat{a}'^{,0}, \ldots, \hat{a}'^{,p}, \ldots \hat{a}'^{,P})$.
   - (ii) For each root in $(\hat{a}'^{,0}, \ldots, \hat{a}'^{,p}, \ldots, \hat{a}'^{,P})$, evaluate the endogenous grid points of wealth $(\hat{x}^0, \ldots, \hat{x}^p, \ldots, \hat{x}^P)$ using the budget constraint:[a]

     $$\hat{x}^p = \hat{a}'^{,p} + (1+\tau)\hat{h}_i + c$$

     and evaluate the current period values $(\hat{v}'^{,0}, \ldots, \hat{v}'^{,p}, \ldots \hat{v}'^{,P})$ as:

     $$\hat{v}'^{,p} = u(c, \hat{h}_i) + \mathbb{E}_y V_{t+1}(\hat{a}'^{,p}, \hat{h}_i)$$

   - (iii) Append the P multiple roots $(\hat{a}'^{,0}, \ldots, \hat{a}'^{,p}, \ldots \hat{a}'^{,P})$ to $\hat{\mathbb{A}}'$, the time t values to $\hat{\mathbb{V}}$, the endogenous grid points to $\hat{\mathbb{X}}$, and P copies of $\hat{h}_i$ to $\hat{\mathbb{H}}'$.
3. Apply FUES (Box 1, Section 2.1.3) to the grids $\hat{\mathbb{V}}$, $\hat{\mathbb{X}}$ and $\hat{\mathbb{H}}$ to recover the refined endogenous grid points $\mathbb{X}$, value grid $\mathbb{V}$, and policy grids $\mathbb{A}'$ and $\mathbb{H}'$.

[a] To save computation time, the endogenous grid for adjusters is defined in terms of cash-on-hand x, where $x = (1+r)a + H + y$ since once housing is liquidated, only total wealth will affect the agent's decision.

We can apply the above steps to each y in the exogenous shock grid. With $\sigma_t^{H,1}$ and $\sigma_t^{a,1}$ approximated on a uniform grid, we can then construct σ_t^H and σ_t^a in the standard way by comparing the value function for adjusters and non-adjusters at each point on the uniform grid of current period states.

Before we turn to the numerical example, let us discuss how the non-monotonicity of the endogenous grid points arises for the housing-adjusting policy functions. The EGM above generates an exogenous grid $\hat{\mathbb{H}}'$, and an endogenous grid $\hat{\mathbb{X}}$ that satisfy:

$$u_1(\hat{c}, \hat{H}') = \beta \mathbb{E}_y \mathcal{I}_{t+1}(\hat{\psi}_t(\hat{x}, \hat{H}'), \hat{H}', y') u_1(c', H'') + u_2(c, H') \\ + (1 - \mathcal{I}_{t+1}(\hat{\psi}_t(\hat{x}, \hat{H}'), \hat{H}', y'))\beta \mathbb{E}_y \left[\Theta_{t+1}(\hat{\psi}_t(\hat{x}, \hat{H}'), \hat{H}', y') + u_2(c', H')\right] \quad (25)$$

where:

$$\hat{c} = \hat{x} - \hat{\psi}_t(\hat{x}, \hat{H}') - (1 + \tau)\hat{H}' \quad (26)$$

and the terms c' and H'' are defined as in equations (19) - (21). The function $\hat{\psi}_t$ gives the value of a' that satisfies the liquid asset Euler equation (23), for a given value of $\hat{x}$ and $\hat{H}'$. To see the source of the non-monotoncity of H' in terms of the $\hat{x}$ above, first note how holding $\hat{\psi}_t$ fixed, the term on the LHS of Equation (25) above falls as wealth $\hat{x}$ increases, resulting in a higher level of housing investment. However, allowing $\hat{\psi}_t$ to also adjust alongside $\hat{x}$ implies that $\hat{\psi}_t$ may experience a discontinuous jump up, causing a discontinuous rise in the marginal utility of consumption today and a discontinuous *fall* in housing investment. Such a discontinuous jump in $\hat{\psi}_t$ occurs because $\hat{\psi}_t$ is not evaluated for a fixed future path of discrete choices. If in a future period the agent reaches a threshold level of wealth and upgrades their housing stock, they will discontinuously decrease their non-housing consumption, causing a jump in the RHS of Equation (23).[13]

Numerical example. To parametrise the model, we use a separable specification for non-housing consumption and housing as follows:

$$u(c, H) = \frac{c^{\gamma-1} - 1}{\gamma - 1} + \alpha \log(H) \quad (27)$$

and define the bequest function as $\theta(c) = \bar{\theta} u(c, 0)$. To parametrize the model, we set $\gamma = 3$, $\alpha = 0.66$, $\beta = .93$, $\tau = .18$, $r = 0.01$, $\bar{\theta} = 1.34$ and $T = 60$. For simplicity, we assume i.i.d income shocks with a random variable $\tilde{z}$ taking values in $[0.1, 1]$ with equal probability, and assume $y_t = \tilde{y}_t(\tilde{z})$ where $\tilde{y}_t$ is the wage function for

[13] In terms of the monotone comparative static arguments given by Iskhakov et al. (2017) in Theorem 4., allowing a' to adjust implicitly as a function H' implies that the continuation value no longer enjoys the single crossing property since now the per-period payoff u also exhibits a jump and is not concave.

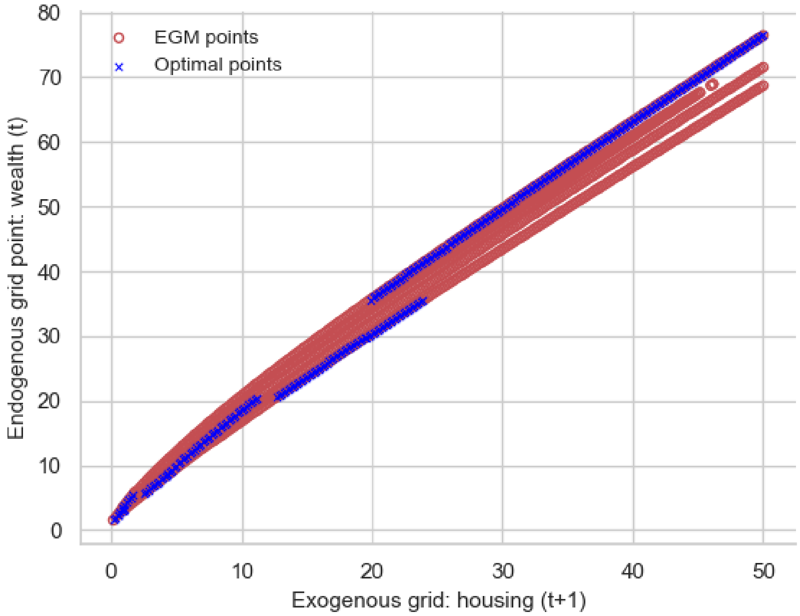

Figure 6: EGM wealth points plotted on a uniform exogenous grid in the housing model with frictions.

females in Dobrescu et al. (2016).

We implement VFI with a combination of Nelder-Mead and Brent's method to perform numerical optimization at each point on the state space for adjusters and non-adjusters. Discretizing the model on a 300 × 300 grid, despite the intermediate root-finding step, still shows EGM with FUES outperforming VFI. The map of endogenous grid points to the exogenous grid $\hat{\mathbb{H}}'$ for a 59 year old is shown in Figure 6. Additionally, Figure 14 in the Appendix compares the policy function computed via VFI with the one computed via FUES and EGM.[14] First note that there are multiple endogenous grid points for each value of the exogenous grid. Also, the endogenous grid has no discernible monotone segments that can be identified by the DC-EGM procedure suggested by Iskhakov et al. (2017). Moreover, the optimal endogenous grid 'jumps down', implying the policy function is not monotone. We can see, however, how FUES recovers the optimal points by zooming in on the top right-hand section of the exogenous grid in Figure 7. Once again, we see our method successfully recovering the upper envelope by eliminat-

[14] On a single core CPU, VFI used approx. 260 secs/iteration; EGM + FUES used approx. 7 secs.

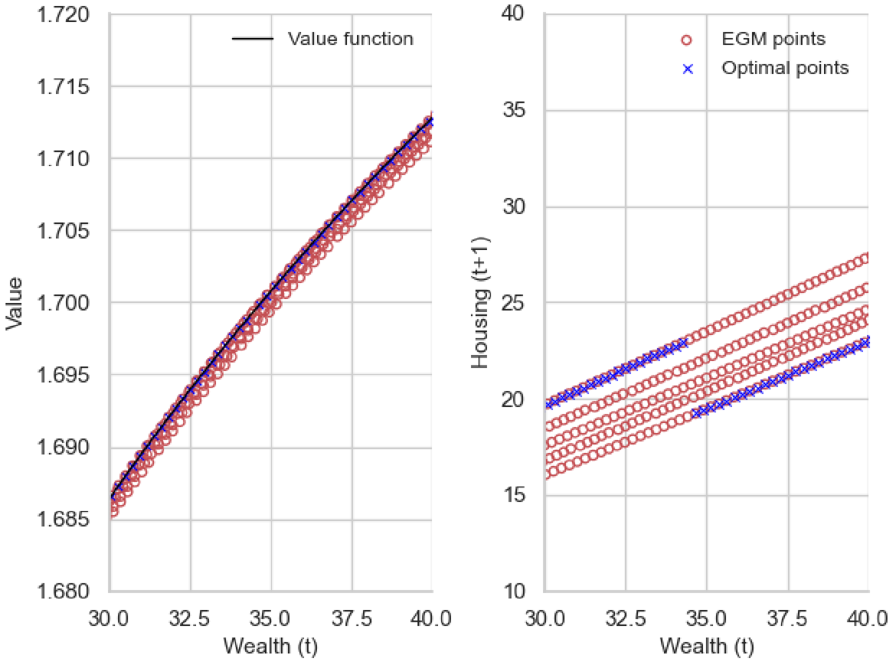

Figure 7: Value function and optimal housing policy function for the top right-hand section of the endogenous grid in the housing model with frictions.

ing all jumps in the policy function that do not result in a left turn on the value function.

2.3 Application 3: Infinite horizon housing choice model

In our final example, we turn to the discrete choice model of housing decisions posed by Fella (2014). Unlike the previous example, the housing choice grid itself is discrete. Because of this, we no longer have to use root-finding procedures in addition to the EGM.

2.3.1 Model environment

Consider an agent who draws an infinite sequence of bounded, stationary, Markov labor income shocks $(y_t)_{t=0}^{\infty}$. Each period the agent decides how much to consume of durable goods c_t, how much to save in liquid assets a_{t+1}, and how much to save in housing assets H_{t+1}. Similar to Application 2, housing assets serve a dual role, namely they are a form of investment good but also provide consumption services. Moreover, housing can only be purchased in discrete amounts on a finite grid $\mathbb{H}$, and adjusting the housing stock each period incurs a fixed transaction cost τH_{t+1}, with $\tau \in [0, 1)$. We assume no borrowing and so, $a_t \geq 0$ must hold.

Formally, the budget constraint for each household can be stated as:

$$a_{t+1} = (1+r)a_t + y_t - (H_{t+1} - H_t) - \tau \mathbb{1}_{H_{t+1} \neq H_t} H_{t+1} - c_t \tag{28}$$

with the per-period utility function defined as:

$$u(c, H) = \alpha \log(c) + (1-\alpha) \log(H) \tag{29}$$

The agent's maximization problem yields the following value function:

$$V_0(a, y, H) = \max_{(a_t, H_t)_{t=0}^{\infty}} \mathbb{E} \sum_{t=0}^{\infty} \beta^t u(c_t, H_{t+1}) \tag{30}$$

such that (28) and (29) hold, $H_t \in \mathbb{H}$ and $a_t \geq 0$ for each t, and $a_0, y_0, h_0 = a, y, H$. We can write the Bellman equation as:

$$V(a, y, H) = \max_{c, H'} \left\{ u(c, H') + \mathbb{E}_y V'(a', y', H') \right\} \tag{31}$$

such that a' satisfies the feasibility condition. Since this is an infinite horizon problem, recall the fixed point to the Bellman equation yields the value function, and the maximizing correspondence yields the policy functions.

2.3.2 The Euler equation

Let us write down the necessary Euler equation. Consider time $t+1$ policy functions σ^a_{t+1} and σ^H_{t+1}, which map the time $t+1$ states (i.e., liquid assets a_{t+1} and housing H_{t+1}) to time $t+2$ states. We then have $a_{t+2} = \sigma^a_{t+1}(a_{t+1}, H_{t+1})$ and $H_{t+2} = \sigma^H_{t+1}(a_{t+1}, H_{t+1})$. A necessary condition is for a_{t+1} to solve the Euler equation in terms of a' as follows:

$$
\begin{aligned}
&u'((1+r)a_t + y_t - \Phi(H_t, H_{t+1}) - a') \\
&\qquad \geq \mathbb{E}_{y_t}(1+r)\beta u'((1+r)a' + y_{t+1} - \Phi(H_{t+1}, H_{t+2}) - \sigma^a_{t+1}(a', H_{t+1}))
\end{aligned} \tag{32}
$$

where we let $\Phi(H_t, H_{t+1}) = (H_{t+1} - H_t) - \tau \mathbb{1}_{H_{t+1} \neq H_t} H_{t+1}$ to ease the notation.

Once again, to state the source of the non-concavity here, recall that the standard approach in concave models is to solve a', then approximate σ_t, σ_{t-1}, σ_{t-2} and so on until σ_t converges. However, in the case of discrete choices, even if the choices H_{t+1} and H_{t+2} are held fixed or calculated to maximize the value function, the difficulty arises when we try to compute the policy functions recursively. Given the time $t+1$ asset policy function σ^a_{t+1}, c_{t+1} may not be monotone and there may be multiple roots to the above equation in a' because σ^a_{t+1} has implicitly inherited discrete choices from future periods. As a result, we may implicitly select a sub-optimal discrete choice for periods after time $t+2$ and hence select a sub-optimal local turning point a'.

2.3.3 Computation using EGM and FUES

In light of our discussion in Section 2.1, the use of FUES to the application here is straightforward. To proceed with computation using EGM and FUES, we can start with an initial guess for the value function and policy functions, V_T, σ^a_T and σ^H_T. Since there is only one Euler equation, we can use EGM without any root-finding steps and calculate the endogenous grid for each H, H' choice fixed. We then follow the procedure detailed in Section 2.1.3 to remove sub-optimal grid points and evaluate the discrete choice policy function. Once approximations of V_{T-1}, σ^a_{T-1} and σ^H_{T-1} are obtained, we continue to iterate until $\|V_{T-i} - V_{T-i-1}\|_\infty < \epsilon$ for some pre-determined $\epsilon > 0$. We use the same parameters as Fella (2014), except we set the lower bound on assets to zero, we use 2-state i.i.d income shocks, and work with a housing grid size of 4 points. Figure 15 in the Appendix shows the asset policy functions for different beginning-of-period housing levels (but allowing the end-of-period housing choice H' to be endogenous). Once again, we see that FUES is able to recover the optimal policy function. Note that while the infinite horizon policy function implicitly defines infinitely many future discrete housing choices, the policy functions only contain finitely many jumps.[15]

3 Theoretical foundation

In this section, we formally define a general discrete-continuous optimization problem, apply the EGM and state the FUES method as a formal pseudo-code. Once we have stated the formal version of FUES, we provide the proof that FUES recovers the optimal endogenous grid points.

3.1 The general discrete-continuous optimization problem

Let the set $\mathbb{D}$ be a finite family of discrete choices and let $\{G^d\}_{d\in\mathbb{D}}$ be a family of continuously differentiable, increasing and strictly concave functions indexed by the discrete choices. Assume that $G^d \colon S \times Z \to \mathbb{R} \cup \{-\infty\}$ for each d, with $S \subset \mathbb{R}$ and $Z \subset \mathbb{R}$, and that S and Z are compact. In the context of a dynamic programming problem, each element in the choice set $\mathbb{D}$ will be a future sequence of history-dependent choices (and not a single discrete choice at a given time), or a set of maps from future sequences of shocks to discrete choices. However, for simplicity, going forward we refer to each d as a discrete choice. The function G^d will be the maximand on the right hand side of the Bellman operator, holding the discrete choice fixed (see discussion in context of applications below). Now, consider the following problem of maximizing the upper envelope, U, of the choice

[15] Nonetheless, in a general infinite horizon model, we have not been able to define conditions that guarantee only finitely many jumps.

specific functions with respect to a continuous choice variable z:

$$V(x) = \max_z \quad U(x, z) \tag{33}$$

where:

$$U(x, z) = \max_d \quad G^d(x, z) \tag{34}$$

such that x is fixed. Next, define $\mathcal{I}(x, z) = \arg\max_d G^d(x, z)$ as the optimal discrete choice holding the arguments of each G^d fixed. Also, define the optimal policy function as:

$$\sigma(x) = \arg\max_z \quad U(x, z) \tag{35}$$

and the choice-specific policy function and choice-specific value function as:

$$\sigma^d(x) = \arg\max_z \quad G^d(x, z), \qquad Q^d(x) = \max_z \quad G^d(x, z) \tag{36}$$

Finally, define a twice differentiable and invertible transition function $f \colon S \times Z \to S$ that maps the current period choice and state to next period state in a dynamic programming problem. We set x' to denote the next period state, where x' will obey $x' = f(x, z)$.

Note that the mapping between our discrete-continuous optimization problem as stated above and the three applications we tackle is as follows:

Application 1, Section 2.1. Fix t and consider workers making a decision to continue work in time $t + 1$. Each d, with $d \in \mathbb{D}$, corresponds to a future path of feasible discrete choices made from $t + 1$ to T. Formally, we can write $\mathbb{D} = \{0, 1\}^{T-t}$. The choice-specific payoff $G^d(x, z)$ corresponds to $u((1 + r)x + y - z) + V^d_{t+1}(z)$, where x is the liquid asset, z is the choice of next period assets, and V^d_{t+1} is the value function for those who work in $t + 1$ conditional on the future path of discrete choices made from $t + 1$ to T.

Application 2, Section 2.2. Fix t, recall our discussion at Section 2.2.3 and consider those making a decision to adjust their housing stock at time t. Each d, with $d \in \mathbb{D}$, corresponds to a future history-dependent path of housing stock adjustment choices from $t+1$ to T. The choice-specific payoff G^d corresponds to:

$$G^d(x,z) = u(x - \hat{\psi}_t^d(x,z) - (1+\tau)z, z) + \mathbb{E}_{y_t} V_{t+1}^d(\hat{\psi}_t^d(x,z), z, y')$$

where x is the total cash-on-hand, z represents the $t+1$ housing choice, and V_{t+1}^d is the $t+1$ choice-specific value function. For a fixed discrete choice, the function $\hat{\psi}_t^d$ maps the wealth and housing choice to the optimal value of next period liquid assets defined by the Euler equation (18).

Application 3, Section 2.3. Fix t and consider those making a decision to adjust their housing stock at time t. Each d, with $d \in \mathbb{D}$, corresponds to an infinite history-dependent path of future housing choices. The choice specific payoff G^d corresponds to the continuation payoff:

$$G^d(x,z) = u((1+r)x + y_t + (1-\delta)H - z - (1+\tau)H', H') + \mathbb{E}_{y'} V^{\prime,d}(z, y') \quad (37)$$

where z represents the next period liquid assets choice, $V^{d,\prime}$ the next period choice-specific value function, and H and H' are the fixed beginning- and end-of-period housing assets.

Since we have stated the above examples as reduced form dynamic programming problems, we have $x' = f(x,z) = z$.

3.1.1 The EGM and formal FUES pseudo-code

Let $\hat{\mathbb{X}}'$ be the exogenous grid of values x', with $|\hat{\mathbb{X}}'| = N$, where N is the exogenous grid size. We assume grids are sequences, that is $\hat{\mathbb{X}}' = \{\hat{x}_0', \hat{x}_1' \ldots \hat{x}_i' \ldots\}$. Recall that a necessary FOC for an interior solution to the problem is that:

$$G_2^{\mathcal{I}(x,z)}(x,z) = 0 \quad (38)$$

where the subscript '2' denotes the partial derivative with respect to the second argument of G^d. Recall that if a continuous choice satisfies the FOC, it will be optimal conditional on the discrete choice (although, the discrete choice may not optimal). Formally:

Remark 1 *For $x \in S$, consider $\tilde{z}$ that satisfies $G_2^{\mathcal{I}(x,\tilde{z})}(x,\tilde{z}) = 0$. If $d = \mathcal{I}(x,\tilde{z})$ is fixed, then $\tilde{z}$ will satisfy:*

$$\tilde{z} = \arg\max_{z} \quad G^d(x,z) \tag{39}$$

We now define the unrefined endogenous grid of values $\hat{x}$ as:

$$\hat{\mathbb{X}} \colon = \left\{ \hat{x} \in S \,|\, \hat{x}' = f(\hat{x},z), G_2^{\mathcal{I}(\hat{x},z)}(\hat{x},z) = 0, \hat{x}' \in \mathbb{X}', z \in Z \right\} \tag{40}$$

and we can also calculate unrefined grids containing candidate policies and values as:

$$\hat{\mathbb{Z}} \colon = \left\{ \hat{z} \in Z \,|\, \hat{x}' = f(x,\hat{z}), G_2^{\mathcal{I}(x,\hat{z})}(x,\hat{z}) = 0, \hat{x}' \in \mathbb{X}', x \in S \right\} \tag{41}$$

$$\hat{\mathbb{V}} \colon = G^{\mathcal{I}(\hat{\mathbb{X}},\hat{\mathbb{Z}})}(\hat{\mathbb{X}}') \tag{42}$$

Note the definition of the endogenous grid values above does not necessarily imply the values of $\hat{\mathbb{X}}$ can be calculated analytically. Rather, we leave open the possibility that the endogenous grid is a general grid (computed analytically or numerically) of points that satisfy the necessary FOCs of a discrete-continuous optimization problem.

The endogenous grid will be initially ordered in ascending order according to the order of $\hat{\mathbb{X}}'$. We now order the grids by the values in $\hat{\mathbb{X}}$ and, through a slight abuse of notation, continue to denote the reordered grids as $\hat{\mathbb{X}}'$, $\hat{\mathbb{Z}}$, $\hat{\mathbb{V}}$ and $\hat{\mathbb{X}}$. For each of these grids, the FUES method will generate a subsequence of refined grids $\mathbb{X}'$, $\mathbb{Z}$, $\mathbb{V}$ and $\mathbb{X}$ indexed by i_k, for instance, $\mathbb{X} = \left\{ \hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{i_k}, \ldots \right\}$.

In the Algorithm below, we let $N = |\hat{\mathbb{X}}|$. Moreover, let $\bar{M}$ be chosen by the researcher. The constant $\bar{M}$ represents the threshold at which FUES registers a jump between policy functions. Finally, $\mathcal{C}_{l+1}$ denotes a Boolean value that is true if point a x_{l+1} is optimal and $\mathcal{D}_l$ denotes a Boolean value that is true if x_k is not optimal and should be deleted. With this notation, we are ready to formally state the FUES method.

Algorithm 1: Fast upper envelope scan

```
Data: $N > 0, \bar{M} > 0, \hat{\mathbb{X}}', \hat{\mathbb{Z}}, \hat{\mathbb{V}}, \hat{\mathbb{X}}$
Result: $\mathbb{X}', \mathbb{Z}, \mathbb{X}$ and $\mathbb{V}$
$k \leftarrow 1, l \leftarrow 1$;
$\hat{x}'_{i_q} \leftarrow \hat{x}'_q, \hat{v}_{i_q} \leftarrow \hat{v}_q, \hat{x}_{i_q} \leftarrow \hat{x}_q \quad q \leq 1$ ;      /* Init. first two points. */
while $l < N$ do
    $g^v_k \leftarrow \frac{\hat{v}_{i_k}-\hat{v}_{i_{k-1}}}{\hat{x}_{i_k}-\hat{x}_{i_{k-1}}}, g^v_{l+1} \leftarrow \frac{\hat{v}_{l+1}-\hat{v}_{i_k}}{\hat{x}_{l+1}-\hat{x}_{i_k}}, g^x_{l+1} \leftarrow \frac{\hat{x}'_{l+1}-\hat{x}'_{i_k}}{\hat{x}_{l+1}-\hat{x}_{i_k}}$;
    if If $g^v_{l+1} \leq g^v_k$ and $|g^x_{l+1}| > \bar{M}$ ;      /* Right turn, jump */
    then
        $\mathcal{C}_{l+1} \leftarrow$ False;
        $\mathcal{C}_{l+1} \leftarrow$ Forward scan ;      /* Optional fwd. scan (Fig. 5) */
    else if $g^v_{l+1} > g^v_k$ ;      /* Left turn */
    then
        $\mathcal{C}_{l+1} \leftarrow$ True ;
        $\mathcal{D}_l \leftarrow$ Backward scan ;      /* Optional bwd. scan (Fig. 5) */
    else
        $\mathcal{C}_{l+1} \leftarrow$ True ;      /* Right turn, no jump */
    end
    if $\mathcal{D}_l$ then
        $\hat{x}'_{i_k} \leftarrow \hat{x}'_{l+1}, \hat{v}_{i_k} \leftarrow \hat{v}_{l+1}$;
        $\hat{x}_{i_k} \leftarrow \hat{x}_{l+1}, \hat{z}_{i_k} \leftarrow \hat{z}_{l+1}$ ;      /* Replace $i_k$ sub-opt. point. */
    else if $\mathcal{C}_{l+1}$ then
        $\hat{x}'_{i_{k+1}} \leftarrow \hat{x}'_{l+1}, \hat{v}_{i_{k+1}} \leftarrow \hat{v}_{l+1}$;
        $\hat{x}_{i_{k+1}} \leftarrow \hat{x}_{l+1}, \hat{z}_{i_{k+1}} \leftarrow \hat{z}_{l+1}$;      /* Add point; keep $i_k$ opt. point. */
        $k = k + 1$
    else
        Pass
    end
    $l \leftarrow l + 1$;
end
```

3.2 Proof FUES recovers optimal points

Under certain conditions, FUES can be guaranteed to recover the optimal points on the endogenous grid. To prove this result, let us start with some definitions and also state the required assumptions on the problem structure and on the endogenous grid.

3.2.1 Preliminaries and assumptions

The optimal subset of points, $\mathbb{X}^\star$, of the endogenous grid can be defined as:

$$\mathbb{X}^\star \colon = \left\{ \hat{x}_i \in \hat{\mathbb{X}} \;\middle|\; \hat{v}_i = V(\hat{x}_i) \right\} \tag{43}$$

The following formalises the definition of the points on the space S where choice-specific value functions intersect.

Definition 1 *Let $T_P \subset S$ be the set of 'crossing points' between the choice specific value functions. That is, the set of $x \in S$ such that for some $m, q \in \mathbb{D}$, $Q^m(x) = Q^q(x)$ and for some ϵ with $\epsilon > 0$, $V(x+\epsilon) = Q^q(x+\epsilon) > Q^m(x+\epsilon)$ and $V(x-\epsilon) = Q^m(x-\epsilon) > Q^q(x-\epsilon)$.*

Turning now to the required assumptions, we first assume that the distance between choice-specific policy functions is bounded below.

Assumption 1 *The term $|f(x, \sigma^d(x)) - f(x, \sigma^s(x))|$ is bounded below by a constant D for all $d, s \in \mathbb{D}$, $d \neq s$ and $x \in S$.*

Second, we assume that the rate of change of the next period states mapped by the transition and policy functions is uniformly bounded above by a common constant.

Assumption 2 *The family of functions $x \mapsto f(x, \sigma^d(x))$ for $d \in \mathbb{D}$ have a common Lipshitz constant M.*

Third and fourth, we also assume the distance between the endogenous grid points is small enough and the jump detection threshold is chosen such that FUES is able to use the previous assumptions to differentiate between a jump and a movement along a choice-specific policy function.

Assumption 3 *There exists $\delta > 0$ such that for all $x^\star_{j+1}, x^\star_j \in \mathbb{X}^\star$ we have $|x^\star_{j+1} - x^\star_j| \leq \delta$ and $\frac{D}{\delta} > 2M$.*

Assumption 4 *The jump detection threshold satisfies $\bar{M} = \frac{D}{\delta} - M$.*

Finally, we place some assumptions on the grid points around crossing points. The assumption below uses the notation from Definition 1 and says that the crossing points are included in the endogenous grid, and that the next period policy associated with a crossing point is associated with the choice specific value function that 'crosses' the optimal value function from below.

Assumption 5 *We have $T_P \subset \mathbb{X}^\star$ and for each $\hat{x}_i \in \mathbb{X}^\star$ such that $\hat{x}_i \in T_P$, $\hat{x}'_i = f(\hat{x}_i, \sigma^m(\hat{x}_i))$.*

Next, we place conditions on the crossing points between choice-specific value functions. Item 1. states that the first point after a crossing is an optimal point and item 2. states that an optimal point after a crossing is sufficiently close to the previous grid point to make a large enough left turn in the value function.

Assumption 6 *Fix $\tilde{x} \in T_P$ and let $x^\star_k$ be the largest element in $\mathbb{X}^\star$ such that $x^\star_k \leq \tilde{x}$ and $x^\star_{k+1}$ be the smallest element in $\mathbb{X}^\star$ such that $x^\star_{k+1} > \tilde{x}$. The following hold:*

1. *We have $\hat{v}_{k+1} = Q^q(\hat{x}_{k+1})$ where Q^q is identified by Definition 1 as the value function crossing at point $\tilde{x}$ from below.*
2. *We have:*

$$\frac{Q^q(x^\star_{k+1}) - Q^q(x^\star_k)}{\delta} > \frac{Q^m(x^\star_{k+1}) - Q^q(x^\star_k)}{x^\star_{k+1} - x^\star_k} + \frac{Q^m(x^\star_k) - Q^m(x^\star_{k-1})}{x^\star_k - x^\star_{k-1}} \tag{44}$$

3.2.2 Main result

Proposition 1 *Let Assumptions 1-6 hold and let $(\mathbb{X}, \mathbb{X}', \mathbb{Z}, \mathbb{V})$ be the tuple of outputs of Algorithm 1 without the forward and backward scans. If $\hat{v}_i = V(\hat{x}_i)$ for $i = 1, 2$, then $\mathbb{X} = \mathbb{X}^\star$.*

Proof. We can prove the proposition above by proving the following claim holds:

$$\hat{x}_i \in \mathbb{X} \iff \hat{x}_i \in \mathbb{X}^\star, \qquad\qquad i \leq l,\ \hat{x}_i \in \mathbb{X} \tag{45}$$

for $l = |\mathbb{X}|$. We will show that if the claim holds for all i with $i \leq l$ for some l, then it will hold for all i with $i \leq l + 1$. Thus, by the principle of induction, the claim will hold for all i with $i \leq l$ for any l. In particular, the claim will hold for $l = |\mathbb{X}|$. As such, begin the proof by making the inductive assumption that the claim at (45) holds for all i with $i \leq l$ for some $l \leq |\mathbb{X}|$. Moreover, let $\{\hat{x}_{i_0}, \ldots, \hat{x}_{i_j}, \ldots, \hat{x}_{i_k}\}$ denote the first k points of the sub-sequence $\mathbb{X}$ such that i_k satisfies $i_k \leq l$.

Part 1: Proof that $\hat{x}_i \in \mathbb{X} \implies \hat{x}_i \in \mathbb{X}^\star$ holds for all i with $i \leq l+1$.

If $i_{k+1} > l+1$, then the proof of this part is complete since $\hat{x}_i \in \mathbb{X} \implies \hat{x}_i \in \mathbb{X}^\star$ continues to hold for all $i < l+1$ by the inductive assumption. On the other hand, suppose $i_{k+1} = l+1$ and the point $\hat{x}_{i_{k+1}}$ satisfies $\hat{x}_{i_{k+1}} \in \mathbb{X}$. There are two cases. The first case arises if a right turn is made on the value correspondence, $\frac{\hat{v}_{i_{k+1}}-\hat{v}_{i_k}}{\hat{x}_{i_{k+1}}-\hat{x}_{i_k}} \leq \frac{\hat{v}_{i_k}-\hat{v}_{i_{k-1}}}{x_{i_k}-\hat{x}_{i_{k-1}}}$. The second case arises if a left turn is made on the value correspondence, $\frac{\hat{v}_{i_{k+1}}-\hat{v}_{i_k}}{\hat{x}_{i_{k+1}}-\hat{x}_{i_k}} > \frac{\hat{v}_{i_k}-\hat{v}_{i_{k-1}}}{\hat{x}_{i_k}-\hat{x}_{i_{k-1}}}$.

[Part 1, Case I: Right turn]

Let $\frac{\hat{v}_{i_{k+1}}-\hat{v}_{i_k}}{\hat{x}_{i_{k+1}}-\hat{x}_{i_k}} \leq \frac{\hat{v}_{i_k}-\hat{v}_{i_{k-1}}}{x_{i_k}-\hat{x}_{i_{k-1}}}$ and let $\tilde{x}$ be the smallest value in T_p such that $\tilde{x} > \hat{x}_{i_k}$. Note that by Claim 2 in the Appendix, we will have $\hat{x}_{i_k} \notin T_P$. There will be two sub-cases. First, that $\hat{x}_{i_{k+1}} \leq \tilde{x}$ and second, that $\hat{x}_{i_{k+1}} > \tilde{x}$.

[Part 1, Case I.A: Right turn and candidate weakly less than next T.P.]

Consider the sub-case where $\hat{x}_{i_{k+1}} \leq \tilde{x}$. We will show that $V(\hat{x}_{i_{k+1}}) = U(\hat{x}_{i_{k+1}}, \hat{z}_{i_{k+1}})$. Assume by contradiction $V(\hat{x}_{i_{k+1}}) \neq U(\hat{x}_{i_{k+1}}, \hat{z}_{i_{k+1}})$ and let $m \in \mathbb{D}$ be such that $V(\hat{x}_{i_{k+1}}) = G^m(\hat{x}_{i_{k+1}}, \sigma^m(\hat{x}_{i_{k+1}}))$. Since $\tilde{x} \geq \hat{x}_{i_{k+1}}$, we have:

$$V(\hat{x}_{i_k}) = G^m(\hat{x}_{i_k}, \hat{z}_{i_k})$$

$$\hat{x}'_{i_{k+1}} = f(\hat{x}_{i_{k+1}}, \sigma^q(\hat{x}_{i_{k+1}}))$$

for some q such that $m \neq q$. Now, by Assumption 1, Assumption 2 and using the reverse triangle inequality, we have:

$$\begin{aligned}
|\hat{x}'_{i_{k+1}} - \hat{x}'_{i_k}| &\geq \left| |\hat{x}'_{i_{k+1}} - f(\sigma^m(\hat{x}_{i_{k+1}}), \hat{x}_{i_{k+1}})| - |\hat{x}'_{i_k} - f(\sigma^m(\hat{x}_{i_{k+1}}), \hat{x}_{i_{k+1}})| \right| \\
&\geq |D - M(\hat{x}_{i_{k+1}} - \hat{x}_{i_k})|
\end{aligned}$$

Dividing through, and noting Assumption 3, we get:

$$\frac{|\hat{x}'_{i_{k+1}} - \hat{x}'_{i_k}|}{\hat{x}_{i_{k+1}} - \hat{x}_{i_k}} \geq |\frac{D}{\hat{x}_{i_{k+1}} - \hat{x}_{i_k}} - M| > \frac{D}{\delta} - M$$

However, this yields a contradiction to $\hat{x}_{i_{k+1}} \in \mathbb{X}$ by line 14 of Algorithm 1, implying $U(\hat{x}_{i_{k+1}}, \hat{z}_{i_{k+1}}) = V(\hat{x}_{i_{k+1}})$.

[Part 1, Case I.B: Right turn and candidate strictly greater than next T.P.]

Now consider the sub-case where $\hat{x}_{i_{k+1}} > \tilde{x}$. If $\hat{x}_{i_{k+1}}$ is the first point in $\hat{\mathbb{X}}$ such that $\hat{x}_{i_{k+1}} > \tilde{x}$, then $\hat{x}_{i_{k+1}}$ will be optimal and $\hat{x}_{i_{k+1}} \in \mathbb{X}^\star$ by Assumption 6- item 1. On the other hand, suppose there exists x_s such that $\hat{x}_{i_{k+1}} > x_s > \tilde{x}$ and x_s is the smallest point in $\hat{\mathbb{X}}$ strictly greater than $\tilde{x}$. By Assumption 6 - item 1., $\hat{x}_s > \tilde{x}$ is optimal. Moreover, by Assumption 6 - item 2., we have:

$$\frac{\hat{v}_s - \hat{v}_{i_k}}{\hat{x}_s - \hat{x}_{i_k}} \geq \frac{Q^q(x_s) - Q^m(\hat{x}_{i_k})}{\delta} = \frac{\hat{v}_s - \hat{v}_{i_k}}{\delta} > \frac{\hat{v}_{x_{i_k}} - \hat{v}_{x_{i_{k-1}}}}{\hat{x}_{i_k} - \hat{x}_{i_{k-1}}} \tag{46}$$

where the first equality follows from Assumption 3. However, the above inequality implies that $x_s \in \mathbb{X}$, which is a contradiction to the assumption of this case that $\hat{x}_{i_{k+1}} > \tilde{x}$.

[Part 1, Case II: Left turn]

Let $\frac{\hat{v}_{i_{k+1}} - \hat{v}_{i_k}}{\hat{x}_{i_{k+1}} - \hat{x}_{i_k}} > \frac{\hat{v}_{i_k} - \hat{v}_{i_{k-1}}}{\hat{x}_{i_k} - \hat{x}_{i_{k-1}}}$. Then by Claim 1 in the Appendix, we must have that $U(\hat{x}_{i_{k+1}}, \hat{z}_{i_{k+1}}) = V(\hat{x}_{i_{k+1}})$, which implies $\hat{x}_{i_{k+1}} \in \mathbb{X}^\star$.

Part 2: Proof that $\hat{x}_i \in \mathbb{X} \Longleftarrow \hat{x}_i \in \mathbb{X}^\star$ holds for i with $i \leq l + 1$.

Now we show that if a candidate point $\hat{x}_{l+1}$, with $\hat{x}_{l+1} \in \mathbb{X}^\star$ and $V(\hat{x}_{l+1}) = Q(\hat{x}_{l+1}, \hat{z}_{l+1})$, then $\hat{x}_{l+1} \in \mathbb{X}$. Consider the first case of a right turn.

[Part 2, Case I: Right turn]

Let $\frac{\hat{v}_{l+1} - \hat{v}_{i_k}}{\hat{x}_{l+1} - \hat{x}_{i_k}} \leq \frac{\hat{v}_{i_k} - \hat{v}_{i_{k-1}}}{\hat{x}_{i_k} - \hat{x}_{i_{k-1}}}$. By Claim 3 in the Appendix, we must have that $\hat{x}_{i_k} \notin T_P$. Suppose we have:

$$V(\hat{x}_{i_k}) = G^m(\hat{x}_{i_k}, \hat{z}_{i_k})$$

$$\hat{x}'_{l+1} = f(\hat{x}_{l+1}, \sigma_q(\hat{x}_{l+1}))$$

$$V(\hat{x}_{l+1}) = G^q(\hat{x}_{l+1}, \sigma_m(\hat{x}_{l+1}))$$

and $m = q$. By Assumption 2 and Assumption 3, we have:

$$\frac{|\hat{x}'_{l+1} - \hat{x}'_{i_k}|}{|\hat{x}_{l+1} - \hat{x}_{i_k}|} \leq M < \frac{D}{\delta} - M$$

Thus, by Algorithm 1, Line 14, we must have $\hat{x}_{l+1} \in \mathbb{X}$.

Alternatively, suppose $m \neq q$, then there exists $\tilde{x}$ such that $\hat{x}_{l+1} > \tilde{x} > \hat{x}_{i_k}$. By Assumption 6 - item 2., Equation (46) will hold. However, this contradicts the assumption of this case that $\frac{\hat{v}_{l+1}-\hat{v}_{i_k}}{\hat{x}_{l+1}-\hat{x}_{i_k}} \leq \frac{\hat{v}_{i_k}-\hat{v}_{i_{k-1}}}{\hat{x}_{i_k}-\hat{x}_{i_{k-1}}}$.

[Part 2, Case II: Left turn]

Let $\frac{\hat{v}_{l+1}-\hat{v}_{i_k}}{\hat{x}_{l+1}-\hat{x}_{i_k}} > \frac{\hat{v}_{i_k}-\hat{v}_{i_{k-1}}}{\hat{x}_{i_k}-\hat{x}_{i_{k-1}}}$. By line 16 of Algorithm 1, we immediately have $\hat{x}_{l+1} \in \mathbb{X}$.

To conclude the proof, we have shown that if the claim given by (45) holds for all i with $i \leq l$, then it will hold for all i with $i \leq l+1$. Finally, note that by the assumption stated by the Proposition, $V(\hat{x}_i) = Q(\hat{x}_i, \hat{z}_i)$ for $i = 1, 2$ and $\hat{x}_i \in \mathbb{X}$ for $i \in \{0, 1\}$, thus the claim is true for $l = 2$. By the principle of induction, the claim given by (45) holds for all l, completing the proof ■

3.2.3 Discussion of assumptions

While the proof above guarantees recovery of the optimal points using FUES, some of the required assumptions may be rather abstract to verify in practice. Assumptions 1 - 2 are straightforward to satisfy if there are finitely many periods and the problem is smooth – as is the case in the first two applications we have considered. Assumption 3 implies that grid sizes play an important role in allowing one to identify discontinuities (as is the case with any discrete-continuous EGM method), and large grid points may be required to pick up the correct location of discontinuities. Assumptions 5 - 6 are more abstract, although the forward and backward scans (see Figure 5 and subsequent discussion) implemented by FUES and the inclusion of approximate crossing points serve as good approximations of these assumptions as demonstrated in Section 2.1. In particular, the forward scan addresses the violation of Assumption 6 - item 2., while the backward scan addresses the violation of Assumption 6 - item 1. Finally, note that Assumption, item 2. will be satisfied for a fine enough grid size.[16]

[16] For $x^\star_{k+1}$ and $x^\star_k$ close enough,$\frac{Q^m(x^\star_{k+1})-Q^q(x^\star_k)}{x^\star_{k+1}-x^\star_k} < 0$. Taking the limit $\delta \to 0$ will yield the inequality in Equation (44), since $Q^{m,\prime}(x^\star_k) < Q^{q,\prime}(x^\star_k)$.

4 Conclusion

This paper provides a fast upper envelope scan (FUES) method to compute the optimal value function for a dynamic optimization problems with discrete and continuous choices. FUES uses the fact that the upper envelope of the value correspondence generated by EGM is only convex in regions where concave choice-specific value functions intersect. Accordingly, it removes all points that cause a jump in the policy function and do not form a convex region of the value correspondence.

FUES is a general, easy-to-code method that can be applied efficiently to dynamic programming problems without assuming policy function monotonicity - a key assumption in previous discrete continuous EGM methods (Fella, 2014; Iskhakov et al., 2017). We prove this method can accurately recover the optimal policy if the grid size is large enough relative to the smallest jump size between choice-specific value functions. This assumption is straightforward to satisfy in finite horizon models with finitely many choices. The infinite horizon models and the models with infinitely many choices due to tastes shocks we considered here also displayed finitely many jumps in the policy functions. Nonetheless, the formal conditions under which an infinite state space produces only finitely many discontinuities remain an area for further work.

Finally, we also identify a few other directions of future work. First, up to our best knowledge, the formal error bounds for EGM are not known, also limiting any formal error analysis for FUES. Moreover, without an analytical example as worked out by Iskhakov et al. (2017) and computed in Application 1 in this paper, it is difficult to verify whether or not FUES - or any EGM method - applied to a discrete-continuous problem recovers the optimal solution. And Euler equation errors are not suitable to evaluate accuracy since Euler equations are only a necessary condition along smooth sections of a value function (Görtz and Mirza, 2018). Additionally, similar to Iskhakov et al. (2017), we also detail FUES in a one-dimensional context. FUES can be, however, applied to multiple dimensions by deconstructing the problem into one-dimensional EGM steps and/or combining them with some root-finding on a reduced state space. Beyond this, there remains scope for a formal multidimensional FUES method that takes into account the geometric structure of the value correspondence, and uses the appropriate multidimensional interpolation method. Progress on such a multidimensional FUES method requires, however, further advances in the general understanding of EGM when the conditions set by Iskhakov (2015) fail. This is a very interesting area of research that we plan to tackle next.

References

Arellano, C., Maliar, L., Maliar, S., and Tsyrennikov, V. (2016). Envelope condition method with an application to default risk models. *Journal of Economic Dynamics and Control*, 69:436–459.

Attanasio, O., Levell, P., Low, H., and Sanchez-Marcos, V. (2018). Aggregating elasticities: Intensive and extensive margins of women's labor supply. *Econometrica*, 86(6):2049–2082.

Bertsekas, D. (2022). *Abstract dynamic programming*. Athena Scientific.

Carroll, C. D. (2006). The method of endogenous gridpoints for solving dynamic stochastic optimization problems. *Economics Letters*, 91(3):312–320.

Carroll, C. D., Kaufman, A. M., Kazil, J. L., Palmer, N. M., and White, M. N. (2018). The Econ-ARK and HARK: Open Source Tools for Computational Economics. In Fatih Akici, David Lippa, Dillon Niederhut, and Pacer, M., editors, *Proceedings of the 17th Python in Science Conference*.

Coleman, W. J. (1990). Solving the stochastic growth model by policy-function iteration. *Journal of Business and Economic Statistics*, 8(1):27–29.

Cooper, I. (2006). Asset pricing implications of nonconvex adjustment costs and irreversibility of investment. *Journal of Finance*, 61(1):139–170.

Dobrescu, L. I., Fan, X., Bateman, H., Newell, B. R., and Ortmann, A. (2016). Retirement savings: A tale of decisions and defaults. *Economic Journal*, 128(2016):1047–1094.

Dobrescu, L. I., Shanker, A., Bateman, H., Newell, B. R., and Thorp, S. (2022). Eggs and baskets: Lifecycle portfolio dynamics. *SSRN Working Paper No. 4069226*.

Druedahl, J. and Jørgensen, T. H. (2017). A general endogenous grid method for multi-dimensional models with non-convexities and constraints. *Journal of Economic Dynamics and Control*, 74:87–107.

Fagereng, A., Holm, M. B., Natvik, G., and Moll, B. (2019). Saving behavior across the wealth distribution : The importance of capital gains. *NBER Working Paper No. 26588*.

Fella, G. (2014). A generalized endogenous grid method for non-smooth and non-concave problems. *Review of Economic Dynamics*, 17:329–344.

Görtz, C. and Mirza, A. (2018). Solving models with jump discontinuities in policy functions. *Oxford Bulletin of Economics and Statistics*, 80(2):434–456.

Graham, R. L. (1972). An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters*, 1:132–133.

Iskhakov, F. (2015). Multidimensional endogenous gridpoint method: Solving triangular dynamic stochastic optimization problems without root-finding operations. *Economics Letters*, 135:72–76.

Iskhakov, F., Jørgensen, T. H., Rust, J., and Schjerning, B. (2017). The endogenous grid method for discrete-continuous dynamic choice models with (or without) taste shocks. *Quantitative Economics*, 8(2):317–365.

Jeyakumar, V., Rubinov, A. M., and Wu, Z. Y. (2007). Non-convex quadratic minimization problems with quadratic constraints: Global optimality conditions. *Mathematical Programming*, 110(3):521–541.

Jeyakumar, V. and Srisatkunarajah, S. (2009). Lagrange multiplier necessary conditions for global optimality for non-convex minimization over a quadratic constraint via S-lemma. *Optimization Letters*, 3(1):23–33.

Kaplan, G., Mitman, K., and Violante, G. L. (2020). The housing boom and bust: Model meets evidence. *Journal of Political Economy*, 128(9):3285–3345.

Kaplan, G. and Violante, G. L. (2014). A model of the consumption response to fiscal stimulus payments. *Econometrica*, 82(4):1199–1239.

Khan, A. and Thomas, J. K. (2008). Idiosyncratic shocks and the role of nonconvexities in plant and aggregate investment dynamics. *Econometrica*, 76(2):395–436.

Krusell, P. and Smith, A. A. (1998). Income and wealth heterogeneity in the macroeconomy. *Journal of Political Economy*, 106(5):867–896.

Laibson, D., Maxted, P., and Moll, B. (2021). Present bias amplifies the household balance-sheet channels of macroeconomic policy. *NBER Working Paper No. 29094*.

Maliar, L. and Maliar, S. (2013). Envelope condition method versus endogenous grid method for solving dynamic programming problems. *Economics Letters*, 120(2):262–266.

Maliar, L., Maliar, S., and Winant, P. (2021). Deep learning for solving dynamic economic models. *Journal of Monetary Economics*, 122:76–101.

Midrigan, V. (2011). Menu costs, multiproduct firms, and aggregate fluctuations. *Econometrica*, 79(4):1139–1180.

Reffett, K. L. (1996). Production-based asset pricing in monetary economies with transactions costs. *Economica*, 63(251):427–443.

Rust, J. (1987). Optimal replacement of GMC bus engines: An empirical model of Harold Zurcher. *Econometrica*, 55(5):999–1033.

Skiba, A. K. (1978). Optimal growth with a convex-concave production function. *Econometrica*, 46(3):527–539.

Stachurski, J. (2022). *Economic Dynamics, second edition: Theory and Computation*. MIT Press.

Yogo, M. (2016). Portfolio choice in retirement: Health risk and the demand for annuities, housing, and risky assets. *Journal of Monetary Economics*, 80:17–34.

Online Appendix

A Intermediate proofs

The first intermediate result says that if a point makes a left turn from an optimal point, then it is also optimal.

Claim 1 *Fix the triple $\hat{x}_i, \hat{x}_{i+1}, \hat{x}_{i+2}$ for some i and assume $\hat{x}_i, \hat{x}_{i+1} \in \mathbb{X}^\star$. If we have:*

$$\frac{\hat{v}_{i+1} - \hat{v}_i}{\hat{x}_{i+1} - \hat{x}_i} < \frac{\hat{v}_{i+2} - \hat{v}_{i+1}}{\hat{x}_{i+2} - \hat{x}_{i+1}} \tag{47}$$

then $\hat{x}_{i+2} \in \mathbb{X}^\star$.

Proof. Fix l, with $l \in \mathbb{D}$ and such that $V(\hat{x}_{i+1}) = G^l(\hat{x}_{i+1}, \sigma^l(\hat{x}_{i+1}))$. Suppose by contradiction that $\hat{x}_{i+2} \notin \mathbb{X}^\star$. Suppose first that $V(\hat{x}_i) = G^l(\hat{x}_i, \sigma^l(\hat{x}_i))$. Since $\hat{x}_{i+2} \notin \mathbb{X}^\star$, we must have $\hat{v}_{i+2} = G^m(\hat{x}_{i+2}, \sigma^m(\hat{x}_{i+2}))$ and $m \neq l$. By concavity and Equation (47), we must have $G^m(\hat{x}_{i+2}, \sigma^m(\hat{x}_{i+2})) > G^l(\hat{x}_{i+2}, \sigma^l(\hat{x}_{i+2}))$. This implies that $V(\hat{x}_{i+2}) = G^p(\hat{x}_{i+2}, \sigma^p(\hat{x}_{i+2}))$ for $p \neq l$ and $p \neq m$. Moreover, the functions G^p and G^l must cross at some point $\tilde{x}_i \in T_p$ with $\tilde{x}_i \in [\hat{x}_{i+1}, \hat{x}_{i+2})$. However, we have now violated Assumption 6 - item 1., since $\hat{x}_{i+2}$ is the first point after a crossing point and must be optimal.

Now suppose $V(\hat{x}_i) = G^b(\hat{x}_i, \sigma^b(\hat{x}_i))$ for some $b \neq l$. This implies a crossing point $\tilde{x}$ satisfies $\tilde{x} \in (\hat{x}_i, \hat{x}_{i+1})$. However, $\tilde{x} \in (\hat{x}_i, \hat{x}_{i+1})$ also yields a contradiction since by Assumption 5, we must have $\tilde{x} \in \mathbb{X}^\star$. ■

Claim 2 *Consider the setting of Part 1 of the proof of Proposition 1. Let $\tilde{x}$ be the smallest value in T_p such that $\tilde{x} > \hat{x}_{i_k}$. If $\frac{\hat{v}_{i_{k+1}} - \hat{v}_{i_k}}{\hat{x}_{i_{k+1}} - \hat{x}_{i_k}} \leq \frac{\hat{v}_{i_k} - \hat{v}_{i_{k-1}}}{x_{i_k} - \hat{x}_{i_{k-1}}}$, then $\hat{x}_{i_k} \notin T_P$.*

Proof.

Suppose by contradiction that $\hat{x}_{i_k} \in T_P$, by Assumption 6, item 2., we must have that $\hat{x}_{i_k+1} \in \mathbb{X}^\star$. Moreover, by Assumption 6, Item 2, we must have:

$$\frac{\hat{v}_{i_k+1} - \hat{v}_{i_k}}{\hat{x}_{i_k+1} - \hat{x}_{i_k}} \geq \frac{Q^q(x_{i_k+1}) - Q^m(x_{i_k})}{\delta} = \frac{\hat{v}_{i_k+1} - \hat{v}_{i_k}}{\delta} > \frac{\hat{v}_{x_{i_k}} - \hat{v}_{x_{i_{k-1}}}}{x_{i_k} - x_{i_{k-1}}} \tag{48}$$

where q and m are elements of $\mathbb{D}$ that satisfy the notation from Definition 1 with $\hat{x}_{i_k}$ as the turning point. To complete the proof, note that since the above equation implies $i_k + 1$ makes a left turn from the point i_k and i_{k-1} on the value correspondence, by Line 9 of Algorithm 1, $\hat{x}_{i_k+1} \in \mathbb{X}$ and $\hat{x}_{i_k+1} \in \mathbb{X}^\star$ and that $i_{k+1} = i_k$, however, this is a contradiction to the Assumption of the claim that x_{i_k} makes a right turn on the value correspondence. Thus, $\hat{x}_{i_k} \notin T_P$. ■

The proof for the following claim is analogous to the previous claim, replacing $i_k + 1$ with $l + 1$.

Claim 3 *Consider the setting of Part 1 of the proof of Proposition 1. Let $\hat{x}_{l+1}$, with $\hat{x}_{l+1} \in \mathbb{X}^\star$ and $V(\hat{x}_{l+1}) = G(\hat{x}_{l+1}, \hat{z}_{l+1})$. If $\tilde{x}$ is the smallest value in T_p such that $\tilde{x} > \hat{x}_{i_k}$, then $\hat{x}_{i_k} \notin T_P$. If $\frac{\hat{v}_{l+1} - \hat{v}_{i_k}}{x_{l+1} - \hat{x}_{i_k}} \leq \frac{\hat{v}_{i_k} - \hat{v}_{i_{k-1}}}{x_{i_k} - \hat{x}_{i_{k-1}}}$, then $\hat{x}_{i_{l+1}} \notin T_P$.*

B Additional figures

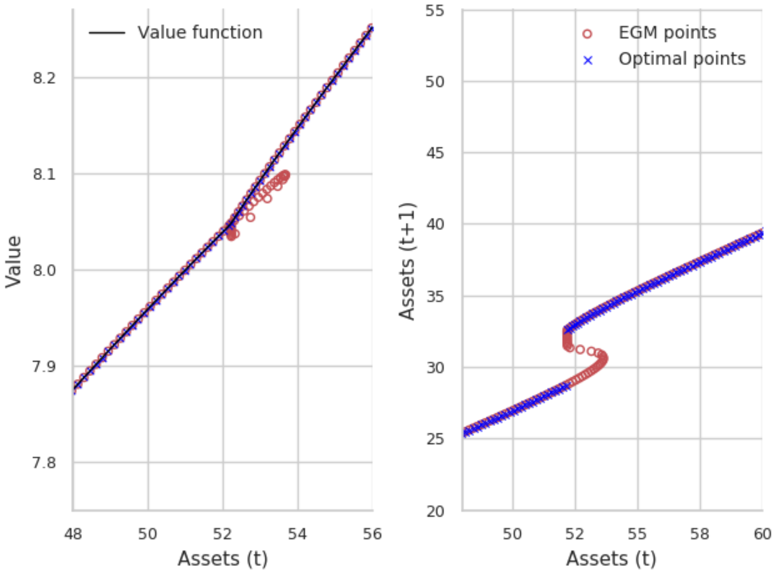

Figure 8: Value correspondence and optimal points for $t = 17$. Parameters from Iskhakov et al. (2017), Figure 4 with smoothing parameter $\sigma = 0.5$.

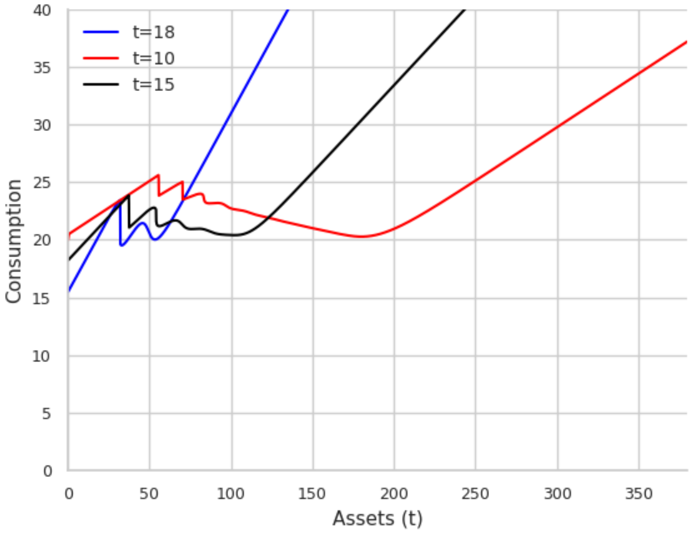

Figure 9: Optimal consumption functions with smoothing for workers.

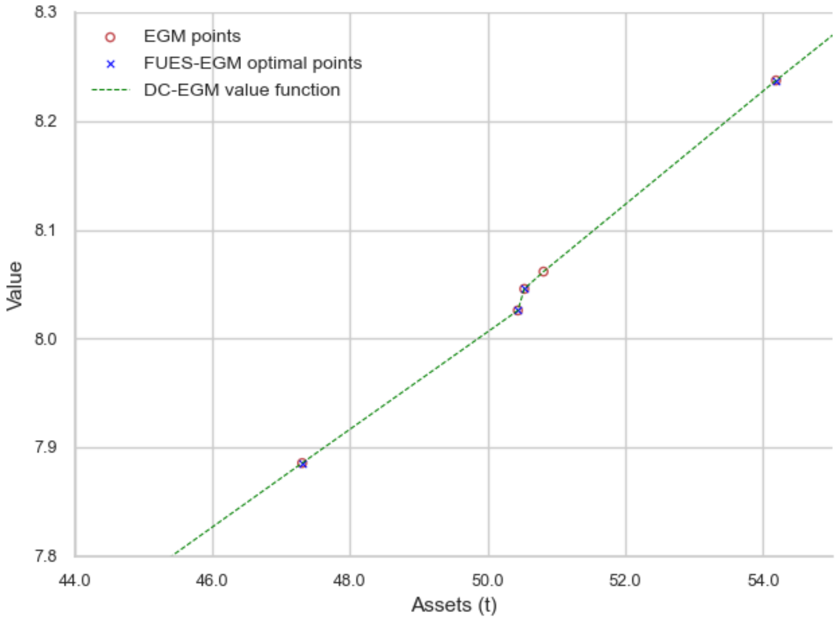

Figure 10: DC-EGM and FUES value correspondence and optimal points for $t = 17$ and 200 grid points. Parameters from Iskhakov et al. (2017).

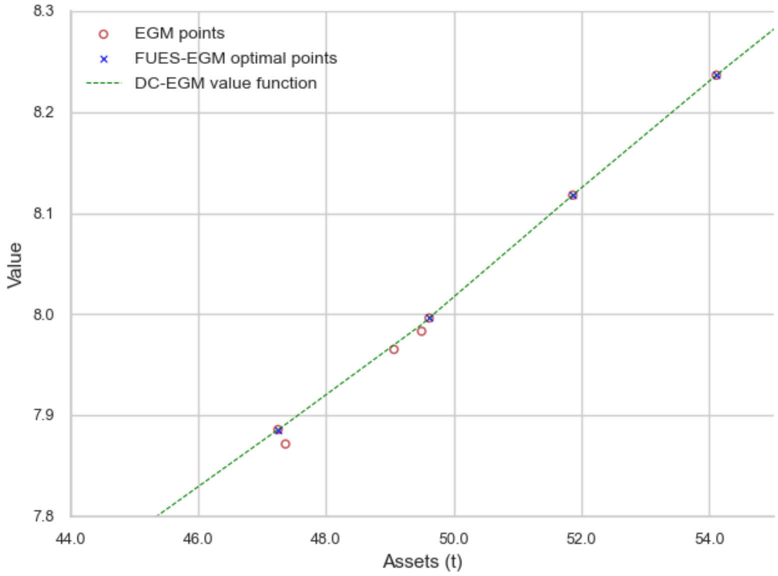

Figure 11: DC-EGM and FUES value correspondence and optimal points for $t = 17$ and 300 grid points. Parameters from Iskhakov et al. (2017).

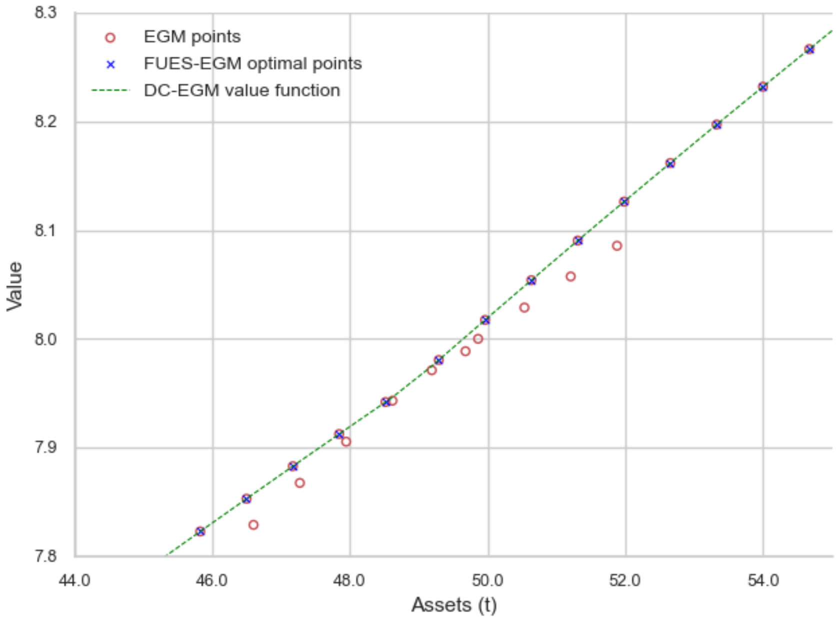

Figure 12: DC-EGM and FUES value correspondence and optimal points for $t = 17$ and 1000 grid points. Parameters from Iskhakov et al. (2017).

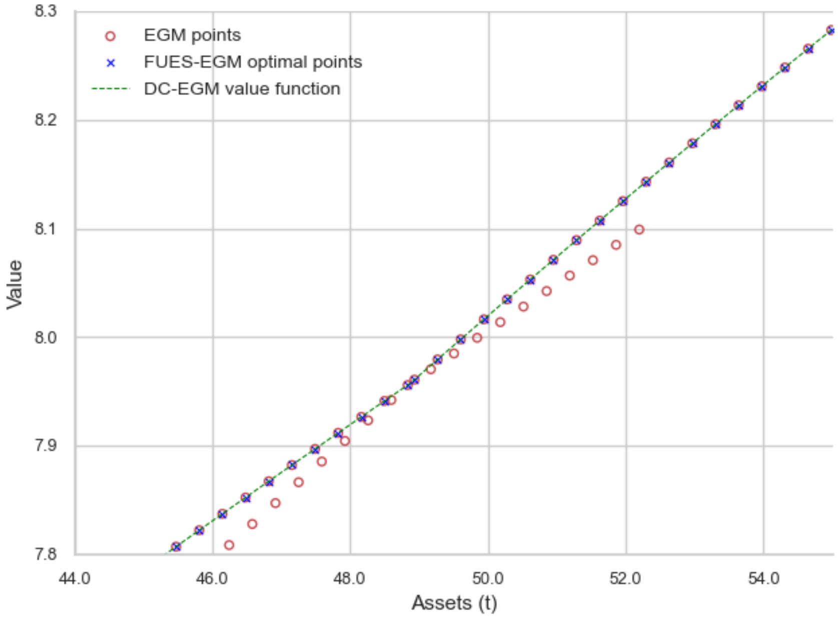

Figure 13: DC-EGM and FUES value correspondence and optimal points for $t = 17$ and 2,000 grid points. Parameters from Iskhakov et al. (2017).

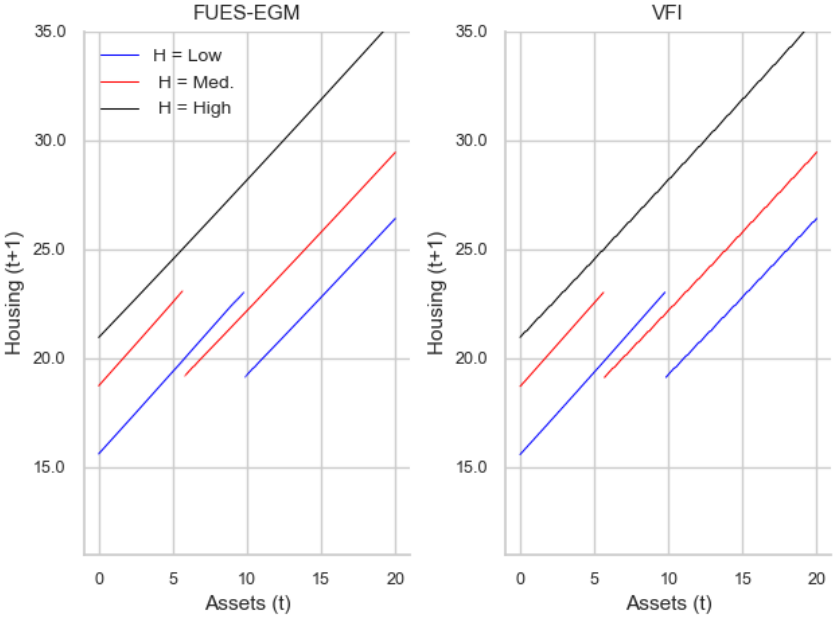

Figure 14: Housing policy function for age 59 in the housing investment friction model (Application 2) using FUES compared to VFI. The plot uses the lowest income shock and three housing capital stock levels at the start of period t.

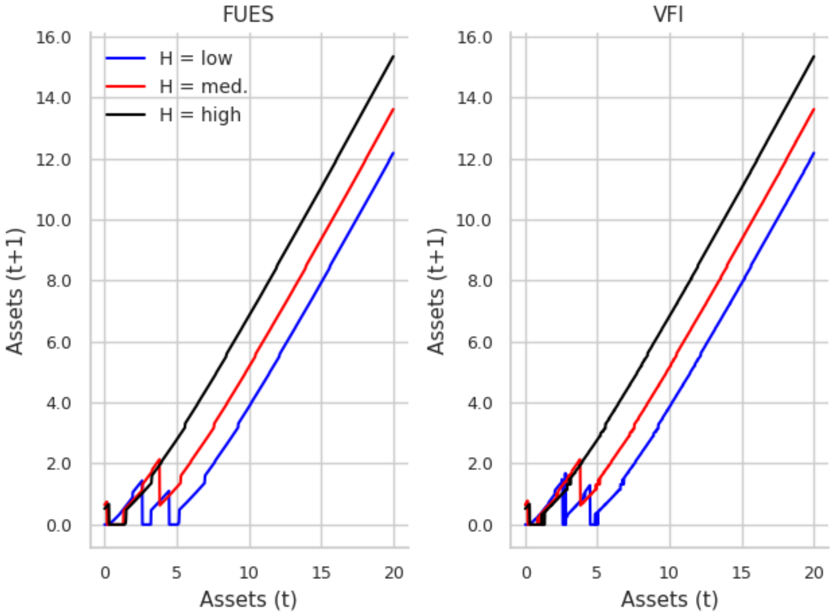

Figure 15: Liquid asset policy function in the infinite horizon housing choice model (Application 3) using FUES compared to VFI. The plot uses the lowest income shock and three housing capital stock levels at the start of period *t*.