A Reinforcement Learning Approach to Solving Incomplete Market Models with Aggregate Uncertainty

Andrei Jirnyi[*], Vadym Lepetyuk[†]

August, 2014

Abstract

We develop a method of solving heterogeneous agent models in which individual decisions depend on the entire cross-sectional distribution of individual state variables, such as incomplete market models with liquidity constraints. Our method is based on the principle of reinforcement learning, and does not require parametric assumptions on either the agents' information set, or on the functional form of the aggregate dynamics. It uses stochastic simulation with a kernel regression estimator both to approximate continuation values and to prioritize the updating states.

Keywords: heterogeneous agents; dynamic programming; reinforcement learning
JEL classification: C63, C68, E20

1 Introduction

In this paper we propose a simulation-based nonparametric method of solving heterogeneous agent models with aggregate uncertainty. Finding equilibria of such models is often a challenge, since their state space includes cross-sectional distributions, which are typically objects of very high dimension. With most of the existing methods requiring computational time that is growing exponentially in the number of state variables, this leads to the so called "curse of dimensionality". Our method consists of a combination of a nonparametric k-nearest-neighbor (k-nn) regression with stochastic simulations. On each iteration of the algorithm, we simulate a realization of aggregate uncertainty, and estimate the expected continuation value using k closest historical simulated realizations of the cross-sectional distribution in the functional space. Equipped with the estimated continuation value, we solve the optimization problem for each agent and construct a cross-sectional distribution, which is added to the set of observations and can be used on the next iterations.

[*] Kellogg School of Management, Northwestern University, 2001 Sheridan Road, Evanston, IL 60208, USA, email: a-jirnyi@northwestern.edu, tel: +1 847 869 1958.

[†] Bank of Canada, 234 Laurier Ave, Ottawa, ON K1A 0G9, Canada, email: vlepetyuk@bankofcanada.ca, tel: +1 613 782 8420.

In our method, agents make their decisions based on a fully-specified high-dimensional cross-sectional distribution, and do not adopt any restrictive assumptions on its transition law, such as separability or a specific parametric form. This flexibility differentiates our method from others proposed in the literature, that has largely been following the approach of Krusell and Smith (1998). Their approach is based on restricting the agents' decision-making to a small number (often one) of aggregate statistics from the entire cross-sectional distribution, and on adopting additional assumptions (e.g. linearity) on their transition law. Our algorithm is comparable in simplicity of implementation and computational time to that of Krusell and Smith (1998).

Our proposed approach is motivated by the substantial recent advances that have been made in the fields of machine learning and operations research, leading to a development of a class of "reinforcement learning," or "approximate dynamic programming" algorithms, which have been used to compute approximate solutions to previously intractable large dynamic optimization problems with thousands, and sometimes hundreds of thousands, of state variables (Powell, 2011). The problems that have been addressed by these methods range from large-scale industrial logistics (Simão et al., 2009) to optimal investment (Nascimento and Powell, 2010) to the game of backgammon (Tesauro, 1994), but the main underlying common feature of the methods in this class is that they rely on stochastic simulations in order to both approximate the expectation of the objective in different future states of the world, and to prioritize the most important states for further analysis. While reinforcement learning has been employed in behavioral economics (Erev and Roth, 1998) and industrial organization (Pakes and McGuire, 2001), to the best of our knowledge we are the first to apply the method to compute numerically a dynamic equilibrium in a macroeconomic model.

We illustrate the method for the classical Krusell and Smith (1998) economy. In the model, the agents face both idiosyncratic employment shocks and aggregate productivity shocks, and can choose to save only into capital, subject to a no-borrowing constraint. The recursive formulation of the agent decision problem includes the cross-sectional distribution of capital stock, which in our implementation is described by 1,000 continuously-valued state variables.

The rest of this article is organized as follows. In Section 2, the Krusell and Smith (1998) model is discusses. In Section 3, we describe our reinforcement learning method for solving the model. In Section 4, we evaluate the accuracy of the method. Section 5 concludes.

2 A heterogeneous agent model

In this section, we describe the environment of Krusell and Smith (1998) with the unemployment insurance as in den Haan et al. (2010), and define the recursive competitive equilibrium.

We select this environment as our example because it is a classic, well-studied, and well-understood problem.[1] One limitation of this particular setup is that, as Krusell and Smith (2006) argue, the agents decisions here *are*, in fact, primarily driven by the economy-wide mean of wealth, whose dynamics is close to linear, and thus the methods that explicitly make such an assumption are applicable and show good performance.[2] Since in this paper we

[1] See den Haan et al. (2010) for a detailed comparison of several solution methods.

[2] This property does not generalize to other classes of models: for example, in the overlapping generations

solve the model without making these assumptions, our results can serve as an independent confirmation of the validity of the Krusell and Smith's conjecture.

2.1 The model

The baseline model is a modified version of Krusell and Smith (1998), as described by den Haan et al. (2010), whose notation we largely adopt.

There is a measure-one continuum of ex-ante identical agents indexed by $i \in [0,1]$ with the preferences over consumption good given by the following expected utility function

$$\mathrm{E} \sum_{t=0}^{\infty} \beta^t u(c_{it}), \tag{1}$$

where the period utility function exhibits constant relative risk aversion $u(c) = (c^{1-\gamma} - 1)/(1-\gamma)$. Agents face two sources of uncertainty: aggregate shock to productivity a_t and individual shock to employment e_{it}, where $e_{it} = 1$ if the agent is employed and zero otherwise. Employed agents inelastically supply l units of labor on which they earn a wage w_t. The employed agents pay a proportional labor tax at rate τ_t, while unemployed agents receive a lump-sum subsidy μw_t. The agents can save nonnegative amounts k_{it} by investing in capital, earning the net return $r_t - \delta$, where r_t is the rental price of capital and δ is the rate of capital depreciation. The agents cannot completely insure themselves against the employment risk by trading contingent bonds, i.e., the markets are incomplete.

The consumption good is produced by competitive firms having Cobb-Douglas production function. The aggregate output is therefore equal to the following

$$Y_t = a_t K_t^{\alpha} (l L_t)^{1-\alpha}, \tag{2}$$

where K_t is the aggregate capital, L_t is the employment rate, and lL_t is the aggregate employment. The government pays unemployment benefits, paid for by taxing the employed. It balances its budget every period, implying a tax rate $\tau_t = \mu(1-L_t)/(lL_t)$.

There are two types of shocks in the model, aggregate and individual. The exogenous shock to aggregate productivity a_t is a two-state Markov process, $a_t \in \{a^b, a^g\} \equiv \{1 - \Delta_a, 1 + \Delta_a\}$, with transition probability $\mathrm{P}\{a_{t+1} = a' | a_t = a\} \equiv \pi(a'|a)$. The individual shock to employment status is also a Markov process, conditional on the realization of the aggregate shock, with transition probabilities given by $\mathrm{P}\{e_{i,t+1} = e' | e_{it} = e, a_{t+1} = a'\} \equiv \pi(e'|e,a')$. The joint transition probability $\mathrm{P}\{e_{i,t+1} = e', a_{t+1} = a' | e_{it} = e, a_t = a\}$ is denoted as $\pi(e', a'|e, a)$, and is chosen so that the aggregate employment is only a function of the aggregate state of the economy.[3]

model of Krueger and Kubler (2004) an approximation of cross-sectional wealth distribution only by its mean can result in a very inaccurate solution to the individual optimization problem.

[3] While this assumption serves to simplify application of other solution methods, it is not necessary for our approach. We retain it, as well as the rest of the calibration in den Haan et al. (2010), for comparison purposes.

2.2 Recursive competitive equilibrium

We consider a recursive competitive equilibrium, which includes a stochastic law of motion of the aggregate state of the economy. Denoting the density of cross-sectional distribution over capital and employment as λ, the aggregate state of the economy is (a, λ). The individual state (k, e, a, λ) consists of the individual holdings of capital, the employment status of the agent, and the aggregate state.

The individual maximization problem in the recursive form is

$$V(k, e, a, \lambda) = \max_{c,k'} \{u(c) + \beta \mathrm{E}\{V(k', e', a', \lambda')|e, a, \lambda\}\} \tag{3}$$

subject to the budget constraint

$$c + k' = r(K, L, a)k + [(1 - \tau(L))le + \mu(1 - e)]\, w(K, L, a) + (1 - \delta)k, \tag{4}$$

the nonnegativity constraint on capital holdings $k' \geq 0$, and the transition law of λ

$$\lambda' = H(\lambda, a, a'). \tag{5}$$

We denote the policy function for the next period capital as $k' = f(k, e, a, \lambda)$.[4]

Wages and capital rental prices in this economy are competitive and given by

$$w(K, L, a) = (1 - \alpha)a \left(\frac{K}{lL}\right)^{\alpha}, \qquad r(K, L, a) = \alpha a \left(\frac{K}{lL}\right)^{\alpha - 1}, \tag{6}$$

where the market clearing conditions require $K = \int_0^1 k_i \lambda_i di$ and $L = \int_0^1 e_i \lambda_i di$.

A recursive competitive equilibrium is the aggregate law of motion H, a pair of individual functions V and f, and the pricing system $(w,\ r)$ such that (i) (V, f) solve the consumer problem, (ii) w and r are competitive, and (iii) the aggregate law of motion H is consistent with the individual policy function f.

3 Solving the model

Finding the equilibrium of this model is complicated, since a distribution function (generally, an infinite-dimensional object) enters the decision problem as a state, leading to the so called "curse of dimensionality". There are two primary aspects to this "curse". First, accurate representation of λ itself requires a large number of state variables. Second, the transition function governing the evolution of the distribution over time $\lambda' = H(\lambda, a, a')$ is an unknown, potentially nonlinear and nonseparable, high-dimensional function of a high-dimensional argument, modeling which presents a significant challenge.

An innovative algorithm has been suggested for such problems by Krusell and Smith (1998). It relies upon making two assumptions. First, Krusell and Smith assume limited rationality on part of the agents. The agents only consider a small number (commonly one)

[4] As the next-period capital is independent of the next-period aggregate productivity, the dependence of the transition law for the aggregate state (5) on the next-period productivity, a' trivially follows from the dynamics of exogenous shocks.

of summary statistics of the distribution λ in their decisions. Second, the aggregate law of motion for these statistics, as perceived by the agents, is restricted to a simple parametric (e.g. linear) functional form. Among the functions of this form, the authors use stochastic simulation to find a "self-confirming" rule, i.e. such a rule that, when taken by agents as given, results in simulated dynamics for the statistics of interest that are close to those implied by the rule within a given tolerance.

In addition to the classic algorithm of Krusell and Smith (1998), a number of alternative techniques have also been proposed in the literature. For instance, "projection methods" still rely on parametrization, but avoid the simulation step by embedding the aggregation of the individual policy functions into the individual problem explicitly. The cross-sectional distribution is either parametrized independently from the individual decision rules (Algan et al., 2010, Reiter, 2010), or follows from the parametrization of the individual policy functions (den Haan and Rendahl, 2010) or from the parametrization of expectations (den Haan, 1996). On the other hand, "perturbation methods" (Kim et al., 2010) are based on approximation of the individual policy functions and the aggregate law of motion around the steady state. These methods, however, are most suitable in environments where individual policies can be easily approximated by a few terms in a functional expansion, and thus face a difficulty with models with occasionally binding borrowing constraints, since the policy functions in such models are not differentiable.

There are two related caveats with respect to the algorithm of Krusell and Smith (1998). First, since it is by construction a limited-rationality solution, where agents are constrained in both their information set and decision-making, it does not easily allow to check how restrictive these constraints are, and how much of an impact they have on the solution. While it is possible to partially address this issue by including additional aggregate state variables, such as higher-order cross-sectional moments, such an expansion is limited due to the second problem: it would also require additional parameters in the transition function, and estimating these parameters in a robust manner can be quite difficult, especially when non-linear interactions between the states are allowed. Moreover, due to the "curse of dimensionality", the computational cost of solving the individual problem grows exponentially with each additional state variable. These issues become even more challenging when the aggregate distribution is multidimensional.

In contrast, our proposed method does not require these assumptions (although, it still allows to make use of them, if warranted by the theory). It is based on the "reinforcement learning" approach[5] to solving high-dimensional dynamic optimization problems. As noted by Powell (2011), there are two critical components that any stochastic algorithm facing the curse of dimensionality requires in order to be effective: (i) a way to infer approximate objective values in the states (e.g. realizations of the cross-sectional distribution) that have not yet been investigated from those that are already known, and (ii) a way to focus attention on the more likely states of the world (the so-called "ergodic set"). Importance of focusing the procedure on the ergodic set has been recently highlighted by Maliar et al. (2011). In our example, we combine both approaches to find solution to a model with a distribution that is at all times fully described by a 1000-dimensional state vector.

[5] Sometimes also referred to as "approximate", "asynchronous", or "adaptive" dynamic programming; see Sutton and Barto (1998) for an excellent introduction.

Our algorithm belongs to the class of learning-based methods which generally come without a formal proof of convergence. Powell (2011) concludes that for value functions described by lookup tables it is not possible to obtain a convergence proof unless all states are visited infinitely often. Following Judd (1992), we thus check the solution accuracy by evaluating residuals in the equilibrium conditions such as Euler equations, which were not used directly by the method. The nonparametric flexibility of the method also comes at a cost of a low rate of convergence. For the standard one-agent neoclassic growth model, our algorithm boils down to a variant of value function iterations that exhibits linear convergence at the rate of time discount factor.

Both, our learning-based method and Krusell and Smith (1998) method hinge on limited ability of agents in predicting the future, albeit in different ways. We assume that agents form expectations based on the past. The resulting solution is self-confirming in the sense that agents's actions are optimal given their expectations and the expectations are consistent with the chosen actions. However, if we keep choosing actions that we think are the best we may end up not learning about true values of those states that we have not explored. Krusell and Smith (1998) assume that the agents are limited in their abilities of keeping track of the aggregate state and its evolution. However, even if only a limited number of characteristics of the aggregate state matter in the future, predicting these characteristics today could require the knowledge of the entire today's state.

The basic intuition for the proposed method is that it splits the value function the decision-maker maximizes into the current utility and the conditional expectation of the continuation value, and uses stochastic simulation with k-nn regression estimator to approximate the latter.

3.1 Continuation values and their estimation

We can rewrite the individual maximization problem (3) as

$$V(k,e,a,\lambda) = \max_{c,k'} \{u(c) + \beta\psi(k',e,a,\lambda)\} \tag{7}$$

subject to the conditions (4)-(6), where ψ is the continuation value of picking capital k', conditional on current shocks (e,a) and the current cross-sectional distribution λ:

$$\psi(k',e,a,\lambda) = \mathrm{E}_{e',a'|e,a} \{V(k',e',a',H(\lambda,a,a'))\,|e,a\}\,. \tag{8}$$

Here ψ is a scalar-valued, non-stochastic function that encompasses both the transitional dynamics $H(\lambda)$ and the dependency between λ and V, and maximization (7) no longer involves computing an explicit expectation. Nevertheless, it requires finding ψ, which is a function of λ and therefore a high-dimensional object.

If there were no aggregate uncertainty in the model, ψ could be found by value function iterations: first, given a value of $V^{(j)}$ in an iteration j, find $\psi^{(j)}$ by evaluating the expectation in (8). Next, compute the value $V^{(j+1)}$ at the next iteration by maximizing (7), and repeat the procedure until convergence. In this way, the classic value function iteration algorithm can be viewed as proceeding backward through time, with iteration-$(j+1)$ value function computed as the previous-period expectation of the iteration-(j) value.

Aggregate uncertainty and distribution-dependency greatly complicate things. First, the expectation in (8) can no longer be evaluated directly, since the exact form of H is not generally available. Second, even if it were available, evaluating (7) and (8) would require to cover all possible values of the distribution λ, leading to the "curse of dimensionality" issue.

Instead of solving for the continuation value ψ explicitly, we propose using stochastic simulation to approximate ψ with a sequence of easy to compute random functions $\hat{\psi}_t$ that converge to ψ.

Imagine that at time t we know the true continuation value $\psi(\cdot, \cdot, a, \lambda)$ in N points $\left\{(\tilde{a}_\tau, \tilde{\lambda}_\tau)\right\}_{\tau=1}^{N}$, i.e. we have a set of triplets $\Psi = \left\{\tilde{a}_\tau, \tilde{\lambda}_\tau, \psi_\tau(\cdot, \cdot, \tilde{a}_\tau, \tilde{\lambda}_\tau)\right\}_{\tau=1}^{N}$, and observe a_t and λ_t. Finding an approximation $\hat{\psi}_t(\cdot, \cdot, a_t, \lambda_t)$ can then be interpreted as a problem of statistical estimation, which can be addressed nonparametrically.

One popular method of such estimation is a k-nearest-neighbor regression, which dates back to Fix and Hodges (1951). It involves first finding M nearest realizations, i.e. such $(\tau_1, ..., \tau_M)$ that $\tilde{a}_{\tau_1} = ... = \tilde{a}_{\tau_M} = a_t$ and $d(\lambda_t, \tilde{\lambda}_{\tau_1}) \leq ... \leq d(\lambda_t, \tilde{\lambda}_{\tau_M}) \leq d(\lambda_t, \tilde{\lambda}_\tau)\ \forall \tau \notin \{\tau_1, ..., \tau_M\}$, where $d(\lambda_t, \tilde{\lambda}_\tau)$ is some distance metric between two distributions λ_t and $\tilde{\lambda}_\tau$. Then, the k-nn estimator of the continuation value ψ is computed as a simple average:

$$\hat{\psi}_t(k', e, a_t, \lambda_t) = \frac{1}{M} \sum_{j=1}^{M} \psi_{\tau_j}(k', e, \tilde{a}_{\tau_j}, \tilde{\lambda}_{\tau_j})\ \forall (k', e). \tag{9}$$

Since the k-nn regression estimator in a separable metric space is asymptotically consistent (Cover and Hart, 1967), $\hat{\psi}_t$ converges to ψ with probability one as the sample size N increases and thus allows to approximate functions of arbitrary complexity.

Consider now a sample realization of this economy, driven by a shock sequence $\{\tilde{a}_\tau\}$, with a corresponding sample path of cross-sectional distributions $\{\tilde{\lambda}_\tau\}$, observed up to time $(t-1)$. Given this history, at time t there can be only two possible combinations of (a_t, λ_t), $\left(a^g, H(\tilde{\lambda}_{t-1}, \tilde{a}_{t-1}, a^g)\right)$ and $\left(a^b, H(\tilde{\lambda}_{t-1}, \tilde{a}_{t-1}, a^b)\right)$, respectively corresponding to the "good" and "bad" realizations of the aggregate state, with known probabilities $\pi(a^g|\tilde{a}_{t-1})$ and $\pi(a^b|\tilde{a}_{t-1})$, and with the two values of transition function H that are implied by the individual policy function at time $t-1$. Therefore, if we knew the value function for these two values of (a_t, λ_t) and for all possible (k_t, e_t), then computing the period-$(t-1)$ continuation value $\psi(\cdot, \cdot, \tilde{a}_{t-1}, \tilde{\lambda}_{t-1})$ in (8) could be done for all k_t and e_{t-1} by simply applying the Markov transition matrix for (e, a):

$$\psi(k_t, e_{t-1}, \tilde{a}_{t-1}, \tilde{\lambda}_{t-1}) = \mathrm{E}_{e_t, a_t} \left\{ V\left(k_t, e_t, a_t, H(\tilde{\lambda}_{t-1}, \tilde{a}_{t-1}, a_t)\right) | e_{t-1}, \tilde{a}_{t-1} \right\}. \tag{10}$$

Thus, observing a time evolution of such an economy up to time $t-1$ provides a set of values $\{\psi(\cdot, \cdot, a_\tau, \lambda_\tau)\}_{\tau=1}^{t-1}$.

The main idea of our method is to simulate a sequence of shocks; at each time t to approximate $\psi(\cdot, \cdot, a_t, \lambda_t)$ by the k-nn regression estimator $\hat{\psi}_t$ (9) using the simulated values Ψ_{t-1}; and substitute this approximation into the time t optimization problem (7) in order to find the *approximate* value function $\tilde{V}_t$:

$$\tilde{V}_t(k_t, e_t, a_t, \lambda_t) = \max_{c, k'} \left\{ u(c) + \beta \hat{\psi}_t(k', e_t, a_t, \lambda_t) \right\} \tag{11}$$

subject to (4)-(6). This newly-obtained estimate $\tilde{V}_t$ is then used to compute an approximate continuation value at time $(t-1)$:

$$\tilde{\psi}_{t-1}(k_t, e_{t-1}, \tilde{a}_{t-1}, \tilde{\lambda}_{t-1}) = \mathrm{E}_{e_t, a_t} \left\{ \tilde{V}_t \left(k_t, e_t, a_t, H(\tilde{\lambda}_{t-1}, \tilde{a}_{t-1}, a_t) \right) | e_{t-1}, \tilde{a}_{t-1} \right\}, \tag{12}$$

which, in its turn, is then added to the set of observations:

$$\Psi_t = \Psi_{t-1} \cup \left(\tilde{a}_{t-1}, \tilde{\lambda}_{t-1}, \tilde{\psi}_{t-1}(\cdot, \cdot, \tilde{a}_{t-1}, \tilde{\lambda}_{t-1}) \right). \tag{13}$$

This way, the problem is solved iteratively *forward* in time, with each new iteration corresponding to the next simulated time period, as opposed to the backward direction (with each new iteration corresponding to the previous time period) in the standard value function iteration algorithm.

One complication that arises in this approach is that it results in a data-generating process which is not stationary due to the ongoing learning by the agents. For example, as the simulation progresses and more observations become available, the agents learn their continuation values better, and their value function approximations and policy decisions improve. In order to mitigate the effect of non-stationarity, we only use the most recent $m(t)$ realizations: $\Psi_t = \left\{ a_j, \lambda_j, \tilde{\psi}_j(\cdot, \cdot, a_j, \lambda_j) \right\}_{j=t-m(t)-1}^{t-1}$, where $m(t)$ is an unbounded, monotonically increasing function defined on natural numbers such that $2 \leq m(t) \leq t-1$. For example, if $m(t) = \max(2, \min(t-1, \nu t))$ with $\nu = 0.05$, the look-back period only includes the most recent 5% of the sample.

As a practical convergence criteria of reinforcement learning algorithm, we check the Bellman equation error over the last T_0 iterations, and terminate the learning when the maximum error is below the tolerance level. The Bellman equation error is calculated as the absolute difference between the k-nn estimator of the continuation value ψ and the computed continuation value $\hat{\psi}$. Convergence of Algorithm 1 ensures that the resulting solution is optimal in the limited-rationality sense, that is, the agents are content with their decision rule to the extent that value forecasts they are making are off by no more than $\bar{\varepsilon}$. The complete procedure is summarized in Algorithm 1.

3.2 Distance measurement

The algorithm described in the previous section requires a distance metric between two probability distributions. In our implementation, the distribution is defined on a grid of values of $k \in \mathcal{K}, e \in \{0, 1\}$. The possible metrics thus include those induced by L_1, L_2, or L_∞ norms in the space of empirical distribution functions.

Choice of the distance metric is important for convergence. Since different metrics emphasize different divergent features of distributions, metrics that focus on features of low relevance do poorly at signal extraction and matching neighbors, resulting in noisier k-nn estimates $\hat{\psi}$, leading to noisier policy functions, and poor convergence.

Recently, Sriperumbudur et al. (2010) have proposed a group of kernel-based distance metrics, and shown that they have strong theoretical foundation for learning applications.

For a given kernel function $\kappa(x, y)$, an induced distance between two conditional empirical densities, $\lambda_{k|e}(\cdot|e)$ and $\lambda'_{k|e}(\cdot|e)$ takes the form

$$d_\kappa(\lambda, \lambda') = \sum_{k \in \mathcal{K}} \sum_{k' \in \mathcal{K}} \kappa(k, k') \, [\lambda(k) - \lambda'(k)] \, [\lambda(k') - \lambda'(k')]. \tag{14}$$

The two conditional distributions are then weighted with the probabilities of each employment status e:

$$d(\lambda, \lambda') = \sum_{e \in \{0,1\}} d_\kappa(\lambda(k|e), \lambda'(k|e))\pi(e). \tag{15}$$

To compare the conditional distributions across capital $k \in \mathcal{K}$, we use the Gaussian kernel, $\kappa(x, y) = e^{-\|x-y\|^2/\sigma^2}$, which is the most common kernel form. Sriperumbudur et al. (2010) show that the Gaussian kernel is integrally strictly positive definite and all kernels of this class induce a proper metric.

It is necessary to note, however, that even though, for demonstration purposes, we explicitly compute the distance between the distribution functions at their highest level of disaggregation, in some cases there may be prior economic reasons why a smaller number of specific aggregate statistics would be expected to matter. Krusell and Smith (2006) argue that the model considered in this paper is in fact such a case, and the agents' decisions in it are primarily governed by the mean capital stock. In order to take such information into account, one could, for example, measure distances between distributions by distances between the corresponding aggregates, thus reducing the estimation noise and improving the continuation value estimates, while still retaining full flexibility in terms of their transitional dynamics (i.e. avoiding the linearity assumption).

3.3 Improving convergence

In practice, convergence of reinforcement learning algorithms can be slow, especially when the time discount factor β is close to unity (see Figure 1, Panel B). We propose a refinement based on the principle of "temporal difference" (TD) learning (Sutton, 1988).

The idea of the refinement is to account for possible bias in the continuation value estimator $\hat{\psi}$. Note that if the estimator $\hat{\psi}$ is unbiased, at time $t-1$ the expected one-step discrepancy is equal to zero:[6]

$$\mathrm{E}_{t-1}\varepsilon_{t-1} = \mathrm{E}_{t-1}(\hat{\psi}_{t-1} - \tilde{\psi}_{t-1}) = \mathrm{E}_{t-1}\left(\tilde{\psi}_{t-1} - \frac{1}{M}\sum_{j=1}^{M}\tilde{\psi}_{\tau_j}\right) = \frac{1}{M}\mathrm{E}_{t-1}\sum_{j}\left(\tilde{\psi}_{t-1} - \tilde{\psi}_{\tau_j}\right) = 0 \tag{16}$$

In case there is a bias, the expectation in (16) is no longer zero, and the sample realization of ε_{t-1} (observed at time t) is an estimate of this bias. Since the predicted continuation value $\hat{\psi}_{t-1}$ was computed as a mean of past continuation values, a bias in $\hat{\psi}_{t-1}$ implies that, on average, there is also a bias in $\{\tilde{\psi}_{\tau_j}\}$. Adjusting $\hat{\psi}_{t-1}$ by a fraction α_{TD} of the bias, where $0 \le \alpha_{\mathrm{TD}} \le 1$, we have

$$\hat{\hat{\psi}}_{t-1} = \hat{\psi}_{t-1} - \alpha_{\mathrm{TD}}(\hat{\psi}_{t-1} - \tilde{\psi}_{t-1}) = \frac{1}{M}\sum_{j=1}^{M}\left[(1-\alpha_{\mathrm{TD}})\tilde{\psi}_{\tau_j} + \alpha_{\mathrm{TD}}\tilde{\psi}_{t-1}\right] \tag{17}$$

[6] Note that $\tilde{\psi}_{t-1}$ here is an object from the time t information set.

i.e., such an adjustment can be done at time t by "nudging" each of the neighbors in step $t-1$ towards $\tilde{\psi}_{t-1}$, thus affecting future estimates that would depend on these neighbors. The entire modified procedure is summarized as Algorithm 2.

4 Results

We apply our nonparametric reinforcement learning (NPRL) algorithm to solve the KS model for the same parameter values as employed in the computational suite project that compares properties of numerical algorithms for solving incomplete markets models with aggregate uncertainty (den Haan et al., 2010).

We employ the following solution procedure. First, a sequence of $T = 350{,}000$ random aggregate shocks is generated, and Algorithm 2 is applied to find a solution, which in this case is represented by a lookup table Ψ_T. This lookup table is subsequently used to simulate an economy that is subjected to a predefined sequence of 10,000 aggregate shocks to productivity, and a single agent within this economy, who has her starting value of individual capital equal to 43, and is subjected to another predefined sequence of 10,000 individual shocks to employment. Both sequences are as specified in den Haan et al. (2010).

As a benchmark to compare our algorithm against, we select the implementation of the KS algorithm by Maliar et al. (2010) (henceforth KS-sim), subject to the same test shock sequence.

The baseline parameters of the NPRL algorithm are listed in Table 1. The remaining parameters of the model correspond to den Haan et al. (2010) and are specified in Table A.1.

Table 1: Parameters of the NPRL algorithm

Parameter	Value
Window width coefficient, ν	0.05
Kernel bandwidth, σ^2	30,000
Grid for capital: uniformly spaced points over interval	[0,100]
Grid for capital: number of points, N	501
"Temporal difference" adjustment factor, α_{TD}	0.5
Maximum number of neighbors, $\overline{M}$	4

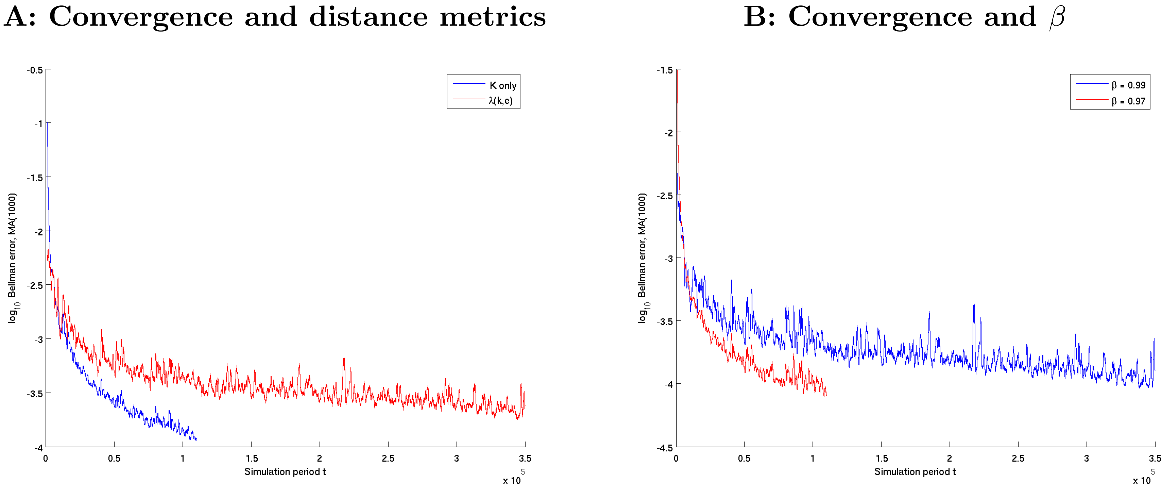
4.1 Convergence

The convergence of our algorithm can be characterized by the maximum error in the Bellman equation evaluated in each period across all agents. In Figure 1 we illustrate the convergence of our original nonparametric reinforcement learning algorithm without the temporal difference adjustment. After a rapid decline of the error in the first 100,000 simulation periods, further learning occurs at a slower rate. As in the standard value function iterations, the rate of convergence is equal to the discount factor β (see Figure 1, Panel B).

In Krusel-Smith model, the intertemporal choice of unconstrained agent depends on interest rates that in turn depend on aggregate capital. One way to improve convergence is to rely on this information and measure the distance between distributions by the difference in means. The Bellman equation errors for this metric are illustrated in Figure 1, Panel A.

Figure 1: Convergence of the NPRL algorithm

$\log_{10}$ of the mean absolute Bellman equation error $\varepsilon_t = |\hat{\psi}_t - \tilde{\psi}_t|$, 1000-period moving average. Left: Convergence under different metrics: mean-k only (blue line), fully disaggregated, d_κ (red line); $\nu = 0.05$. Right: Convergence for different values of discount factor β; $\nu = 0.10$; fully disaggregated distance metric. All other parameters are listed in Tables 1 and A.1.



A: Convergence and distance metrics

B: Convergence and β

4.2 Aggregate law of motion

Table 2 presents summary statistics of the capital stock per capita, according to the solution by the NPRL and KS-sim algorithms.[7] Figure A.1 shows part of the sample path of the aggregate law of motion of capital.Clearly, the resulting aggregate dynamics are very similar, both across employment status and across productivity states, indicating that the KS algorithm is indeed well-suited for its namesake application, and that its assumptions are not overly restrictive in this case.

4.3 Accuracy evaluation

We measure the accuracy of solution by Euler equation errors (den Haan, 2010b, Judd, 1992). For a given agent, given the period-t expectation of period-$(t+1)$ consumption and the interest rate, we calculate period-t consumption that is implied by the Euler equation:

$$\mathrm{E}_t\left[\beta \frac{u'(c_{t+1})}{u'(c_t)}(1 + r_{t+1} - \delta)\right] = 1 \tag{18}$$

We measure the Euler equation errors as the percentage difference between the consumption implied by the Euler equation in period t and the computed period-t consumption. We exclude the periods in which the agent is bound by the liquidity constraint, since the Euler equation does not hold in those periods.

[7] Since the initial state of the economy at the beginning of the simulation would be generally different between the two methods (in case of the NPRL model, determined by the realizations of the random shocks at the end of the training sample), we drop first 1,000 periods of the test sequence, and compute the statistics over the remaining 9,000 periods.

Table 2: Aggregate capital stock

Time-series means and standard deviations of capital stock per capita. Last 9,000 periods of the sample sequence in den Haan et al. (2010). Recessions and expansions are defined as periods of low and high aggregate productivity, respectively. NPRL is the solution produced by the nonparametric reinforcement-learning algorithm, KS-sim is the solution in Maliar et al. (2010).

	Full sample		Recessions		Expansions	
	NPRL	KS-sim	NPRL	KS-sim	NPRL	KS-sim
Means:						
Total	39.321	39.333	39.027	39.040	39.635	39.645
Employed	37.684	37.697	36.959	36.974	38.456	38.467
Unemployed	39.463	39.475	39.257	39.269	39.684	39.694
Standard Deviations:						
Total	1.031	1.026	0.992	0.988	0.977	0.971
Employed	1.460	1.456	1.278	1.274	1.228	1.223
Unemployed	0.994	0.989	0.972	0.968	0.970	0.964

Euler equation errors primarily reflect the accuracy of the solution of the individual problem. In this paper, we used a piecewise-linear approximation of the value function on the grid of capital $\mathcal{K}$. In addition to a uniform equally-spaced grid, we evaluated a polynomial grid suggested in Maliar et al. (2010), with grid points defined as

$$k_j = \bar{k}\,(j/N)^{\theta}\,, \tag{19}$$

where N is the grid size and $\bar{k} = 100$ is the upper bound for capital; we consider a number of values for the power parameter θ. In Table 3 we report mean and maximum errors for different grid sizes and types. Clearly, an irregular grid is superior to the uniformly-spaced one for solving the individual problem. However, grid choice had very little effect on the dynamics of the *aggregate* capital: for example, while individual Euler equation errors were as high as 9% for some agents in case of a uniformly-spaced 501-point grid, the largest difference between the two resulting series of aggregate capital was only 0.07%. The reason for this is that the largest Euler equation errors are observed for agents with very low level of capital, whose decisions have a very small effect on the aggregate investment in this model.

Accuracy of the KS method is often measured by the R^2 statistic. While, as den Haan (2010a) notes, this measure does not reflect the accuracy of the solution sufficiently well, it is nevertheless an indicator that reflects how well the actual aggregate law of motion corresponds to the one perceived by the agents, and serves to evaluate whether the "limited-rationality" assumption of a linear aggregate law of motion is realistic. Since our method relies on approximation of the continuation values for each agent, rather than of the aggregate law of motion for capital, a similar statistic is the R^2 of the one-step-ahead k-nn predictor $\tilde{\psi}$ (9), which can be computed for any (k, e) as follows:

$$R^2 = 1 - \sum_{t=1}^{T} \frac{(\tilde{\psi}_t - \hat{\psi}_t)^2}{(\tilde{\psi}_t - \overline{\psi})^2}, \tag{20}$$

Table 3: Euler equation errors

Euler equation errors (18). 350,000 simulations; kernel distance with $\sigma^2 = 30,000$. Column 1: number of grid points for capital on $[0, 100]$, for each level of individual shock. $p(\theta)$ denotes a polynomially-spaced grid (19) with power factor θ. Euler equation errors are in percent of current consumption. Presented are time-series mean and maximum of the Euler equation errors for a single simulated agent, using the individual and aggregate shock sequences as per den Haan et al. (2010).

Grid points	Spacing	EE error, %: mean	EE error, %: max
1001	Uniform	0.0937	8.5426
	p(2)	0.0923	0.4302
	p(4)	0.0933	0.2981
	p(7)	0.0968	0.3499
501	Uniform	0.0976	9.2257
	p(2)	0.0953	0.8661
	p(4)	0.1005	0.6313
	p(7)	0.1125	0.4063
KS-sim		0.0930	0.4360

where $\overline{\psi}$ is the sample mean of $\tilde{\psi}_t$. Similarly to the R^2 statistic in the KS method, a high R^2 implies that an agent finds her estimates of the continuation value to be sufficiently precise.

Figure 2 shows R^2 of the nearest-neighbor estimator for different values of k and e, Panel A corresponding to a simulation with distance measured between the full λ distributions, and Panel B to one with distance between mean capital only. Both methods result in high values of the R^2 statistic for all agents, although slightly higher in the capital-only case (equally-weighted mean $R^2 = 0.999988$) than in the full-distribution case (mean $R^2 = 0.999948$).

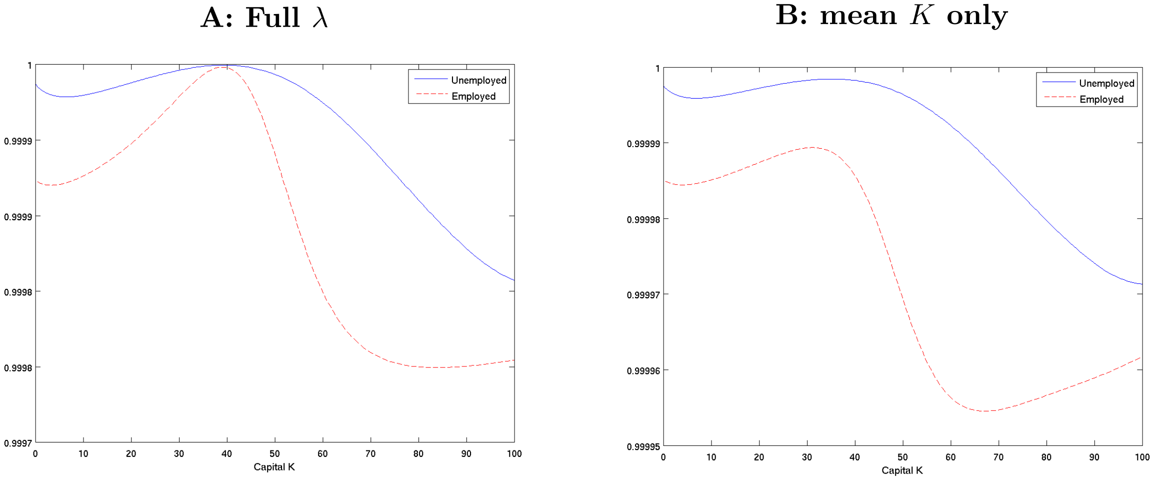
5 Conclusion

In this paper, we have develop a method of solving heterogeneous agent models in which individual decisions depend on the entire cross-sectional distribution of individual state variables, that does not require parametric assumptions on either the agents' information set, or on the functional form of the aggregate dynamics. As an illustration, we apply the method to the classic Krusell and Smith economy, as described in den Haan et al. (2010). Our unconstrained solution of this model is very close to the limited-rationality solution of the original Krusell and Smith algorithm.

Even though in this paper we focus on a heterogeneous-agent setting with aggregate uncertainty, we believe that related approximate optimization methods could prove useful in other large economic models, such as multi-country growth models, as well.

Figure 2: R^2 of the k-nn estimator

One-step-ahead R^2 of the k-nn continuation-value estimator (20) for each value of individual capital and employment status. 350,000 simulations; 501-point polynomial grid (19) with $\theta = 4$. Panel A: kernel distance metric with $\sigma^2 = 30{,}000$. Panel B: distance between means of capital.



A: Full λ

B: mean K only

References

Algan, Y., O. Allais, and W. J. den Haan (2010): "Solving the incomplete markets model with aggregate uncertainty using parameterized cross-sectional distributions," *Journal of Economic Dynamics and Control*, 34, 59–68.

Cover, T. and P. Hart (1967): "Nearest neighbor pattern classification," *Information Theory, IEEE Transactions on Information Theory*, 13, 21–27.

den Haan, W. J. (1996): "Heterogeneity, aggregate uncertainty, and the short-term interest rate," *Journal of Business & Economic Statistics*, 14, 399–411.

——— (2010a): "Assessing the accuracy of the aggregate law of motion in models with heterogeneous agents," *Journal of Economic Dynamics and Control*, 34, 79–99.

——— (2010b): "Comparison of solutions to the incomplete markets model with aggregate uncertainty," *Journal of Economic Dynamics and Control*, 34, 4–27.

den Haan, W. J., K. Judd, and M. Juillard (2010): "Computational suite of models with heterogeneous agents: Incomplete markets and aggregate uncertainty," *Journal of Economic Dynamics and Control*, 34, 1–3.

den Haan, W. J. and P. Rendahl (2010): "Solving the incomplete markets model with aggregate uncertainty using explicit aggregation," *Journal of Economic Dynamics and Control*, 34, 69–78.

Erev, I. and A. E. Roth (1998): "Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria," *American Economic Review*, 88, 848–881.

Fix, E. and J. Hodges (1951): "Discriminatory analysis. Nonparametric discrimination: Consistency properties," Technical report 4, project number 21-49-004, USAF School of Aviation Medicine, Randolph Field, Texas.

Judd, K. L. (1992): "Projection methods for solving aggregate growth models," *Journal of Economic Theory*, 58, 410–452.

Kim, S. H., R. Kollmann, and J. Kim (2010): "Solving the incomplete markets model with aggregate uncertainty using a perturbation method," *Journal of Economic Dynamics and Control*, 34, 50–58.

Krueger, D. and F. Kubler (2004): "Computing equilibrium in OLG models with stochastic production," *Journal of Economic Dynamics and Control*, 28, 1411–1436.

Krusell, P. and A. A. Smith, Jr. (1998): "Income and Wealth Heterogeneity in the Macroeconomy," *Journal of Political Economy*, 106, 867–896.

——— (2006): "Quantitative macroeconomic models with heterogeneous agents," in *Advances in Economics and Econometrics: Theory and Applications, Ninth World Congress*, vol. 1, 298–340.

Maliar, L., S. Maliar, and F. Valli (2010): "Solving the incomplete markets model with aggregate uncertainty using the Krusell-Smith algorithm," *Journal of Economic Dynamics and Control*, 34, 42–49.

Maliar, S., L. Maliar, and K. Judd (2011): "Solving the multi-country real business cycle model using ergodic set methods," *Journal of Economic Dynamics and Control*, 35, 207–228.

Nascimento, J. and W. Powell (2010): "Dynamic programming models and algorithms for the mutual fund cash balance problem," *Management Science*, 56, 801–815.

Pakes, A. and P. McGuire (2001): "Stochastic algorithms, symmetric Markov perfect equilibrium, and the 'curse' of dimensionality," *Econometrica*, 69, 1261–1281.

Powell, W. B. (2011): *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, Wiley-Interscience, 2nd ed.

Reiter, M. (2010): "Solving the incomplete markets model with aggregate uncertainty by backward induction," *Journal of Economic Dynamics and Control*, 34, 28–35.

Simão, H. P., J. Day, A. P. George, T. Gifford, J. Nienow, and W. B. Powell (2009): "An approximate dynamic programming algorithm for large-scale fleet management: A case application," *Transportation Science*, 43, 178–197.

Sriperumbudur, B., A. Gretton, K. Fukumizu, B. Schölkopf, and G. Lanckriet (2010): "Hilbert space embeddings and metrics on probability measures," *Journal of Machine Learning Research*, 11, 1517–1561.

Sutton, R. S. (1988): "Learning to predict by the methods of temporal differences," *Machine Learning*, 3, 9–44.

Sutton, R. S. and A. G. Barto (1998): *Reinforcement learning*, MIT Press.

Tesauro, G. (1994): "TD-Gammon, a self-teaching backgammon program, achieves master-level play," *Neural Computation*, 6, 215–219.

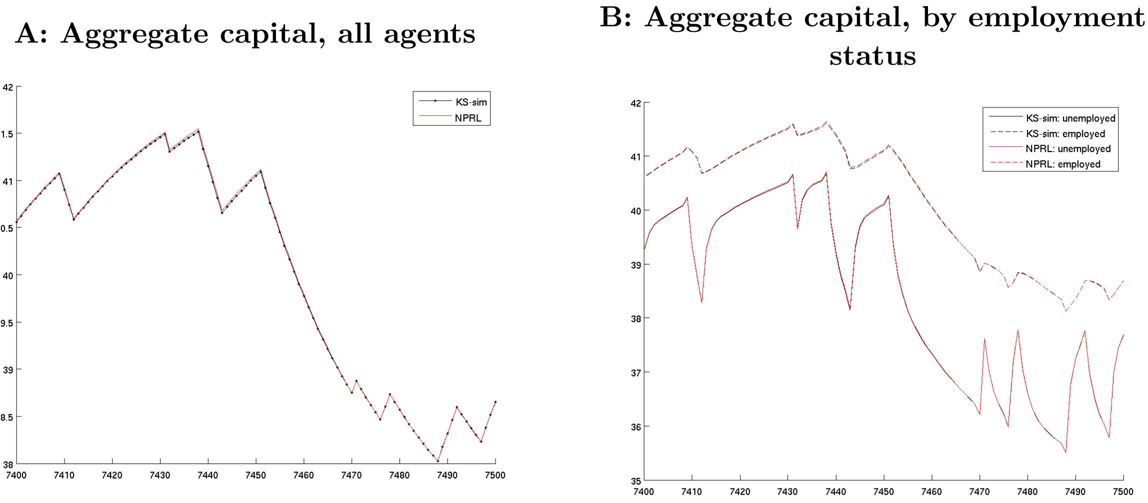
A Appendix

Table A.1: Parameters of the model

Parameter	Value
Time-discount factor (quarterly), β	0.99
Coefficient of relative risk aversion, γ	1
Capital share of total output, α	0.36
Capital depreciation rate, δ	0.025
Labor endowment, l	1/0.9
Unemployment benefit, μ	0.15
Standard deviation of aggregate productivity shocks, Δ_a	0.01

Figure A.1: Dynamics of aggregate capital

Aggregate capital dynamics in two solutions: Maliar et al. (2010) (KS-sim, black lines) and non-parametric reinforcement learning (NPRL, red lines, 350,000 periods simulated, TD with $\alpha = 0.5$, kernel distance with $\sigma^2 = 30,000$, $\nu = 0.1$.) Time is from the beginning of the aggregate shock sequence in den Haan et al. (2010). Panel A: aggregate capital; Panel B: aggregate capital by employment status.



A: Aggregate capital, all agents

B: Aggregate capital, by employment status

Algorithm 1 Stochastic simulation

```
 1: Define a grid $\mathcal{K}$ over capital, $\mathcal{K} = (k_1, ..., k_N)$
 2: Pick initial realization of the aggregate shock $\tilde{a}_0$, initial approximation $\hat{\psi}_0(k', e, a)$, and
    initial distribution $\lambda_1(k, e|a) \in \mathbb{R}^{|\mathcal{K}| \times 2}$
 3: Pick tolerance $\overline{\varepsilon}$, maximum number of neighbors $\overline{M}$, and lookback window $m(t)$, e.g.,
    $m(t) = 0.1t$
 4: Initialize $t \leftarrow 1$, $\Psi_0 \leftarrow \emptyset$
 5: repeat
 6:   for each value of the aggregate state $a \in \{a^b, a^g\}$ and corresponding capital distribu-
      tion $\lambda_t(\cdot, \cdot|a)$ do
 7:     Compute aggregate capital $K \leftarrow \sum_{k \in \mathcal{K}} [\lambda_t(k, 0|a) + \lambda_t(k, 1|a)]k$
 8:     Compute state-dependent wage $w(K, a)$ and interest rate $r(K, a)$
 9:     Search $\Psi_{t-1}$ for $M$ nearest realizations $\{t_1, ..., t_M\}$, i.e. find the largest $M \leq \overline{M}$ and
        $\{t_j\}_{j=1}^{M} \subset (t - m(t), ..., t - 2)$, such that $\forall j = 1, ..., M$, $a_{t_j} = a$, and $d(\lambda_t, \lambda_{t_j}) \leq$
        $d(\lambda_t, \lambda_\tau) \; \forall \tau \notin \{t_1, ..., t_M\}$
10:     for each value of the individual state $e \in \{0, 1\}$ and capital $k' \in \mathcal{K}$ do
11:       Compute $\hat{\psi}_t(k', e, a) \leftarrow \frac{1}{M} \sum_{j=1}^{M} \tilde{\psi}_{t_j}(k', e, \tilde{a}_{t_j}, \tilde{\lambda}_{t_j})$ if $M > 0$, or otherwise set $\hat{\psi}_t \leftarrow$
          $\hat{\psi}_0$
12:       Solve the optimization problem (7) using $\hat{\psi}_t$ in place of $\psi$, and determine $k'_t(k, e, a)$
          and $V_t(k, e, a)$
13:     end for
14:   end for
15:   Compute $\tilde{\psi}_{t-1}(k_t, e_{t-1}, \tilde{a}_{t-1}) \leftarrow \mathrm{E}\{V_t(k_t, e_t, a_t)|e_{t-1}, \tilde{a}_{t-1}\}$, using the Markov transition
      matrix $\pi(e', a'|e, a)$
16:   Compute discrepancy $\varepsilon_{t-1} \leftarrow \max_{k', e} |\hat{\psi}_{t-1}(k', e, \tilde{a}_{t-1}) - \tilde{\psi}_{t-1}(k', e, \tilde{a}_{t-1})|$
17:   Add an observation $(\lambda_{t-1}, \tilde{a}_{t-1}, \tilde{\psi}_{t-1})$ to the lookup table: $\Psi_t \leftarrow \Psi_{t-1} \cup (\lambda_{t-1}, \tilde{a}_{t-1}, \tilde{\psi}_{t-1})$

18:   Generate the period-$t$ realization of the aggregate shock $\tilde{a}_t$ according to $\pi(a_t|\tilde{a}_{t-1})$
19:   Using the policy function $k'_t(k, e, a)$ found in step 12, compute the next period capital
      distribution $\lambda_{t+1}(k_{t+1}, e_{t+1}|a_{t+1})$ for all $(e_{t+1}, a_{t+1})$ and $k_{t+1} \in \mathcal{K}$
20:   advance $t \leftarrow t + 1$
21: until $\max_{t - T_0 \leq \tau \leq t-1} \varepsilon_\tau < \overline{\varepsilon}$
```

Algorithm 2 Stochastic simulation with temporal-difference adjustment

```
1: Pick the temporal difference update factor $\alpha_{\mathrm{TD}} \in [0, 1]$
2: Initialize $M' \leftarrow 0$
3: Proceed with steps 1 - 17 of Algorithm 1
4: If $t > 1$ and $M' > 0$, for each of the nearest neighbors of period $(t - 1)$, $\tau_1, ..., \tau_{M'}$,
   perform an adjustment of $\Psi_t$ : $\tilde{\psi}_{\tau_j} \leftarrow (1 - \alpha_{\mathrm{TD}})\tilde{\psi}_{\tau_j} + \alpha_{\mathrm{TD}}\tilde{\psi}_{t-1}(k, e, \tilde{a}_{t-1})$
5: Update $M' \leftarrow M$ and $\{\tau_j\} \leftarrow \{t_j\}$, $j = 1, ..., M$
6: Proceed with the remaining steps of Algorithm 1 until completion
```