Queen's Economics Department Working Paper No. 1118

Bayesian Estimation of Dynamic Discrete Choice Models

Susumu Imai
Queen's University

Neelam Jain
Northern Illinois University

Andrew Ching
University of Toronto

Department of Economics
Queen's University
94 University Avenue
Kingston, Ontario, Canada
K7L 3N6

12-2006

Bayesian Estimation of Dynamic Discrete Choice Models*

Susumu Imai
Queen's University

Neelam Jain
Northern Illinois University

and
Andrew Ching
University of Toronto

December, 2006

Abstract

We propose a new methodology for structural estimation of dynamic discrete choice models. We combine the Dynamic Programming (DP) solution algorithm with the Bayesian Markov Chain Monte Carlo algorithm into a single algorithm that solves the DP problem and estimates the parameters simultaneously. As a result, the computational burden of estimating a dynamic model becomes comparable to that of a static model. Another feature of our algorithm is that even though per solution-estimation iteration, the number of grid points on the state variable is small, the number of effective grid points increases with the number of estimation iterations. This is how we help ease the "Curse of Dimensionality". We simulate and estimate several versions of a simple model of entry and exit to illustrate our methodology. We also prove that under standard conditions, the parameters converge in probability to the true posterior distribution, regardless of the starting values.

*Please direct all correspondence to Susumu Imai, Department of Economics, Queen's University, 233 Dunning Hall, 94 University Avenue, Kingston ON. K7L 5M2, Canada, e-mail: imais@econ.queensu.ca, phone: 613-533-2258, or Neelam Jain,Department of Economics, Northern Illinois University, 508 Zulauf Hall, DeKalb, IL. 60115, e-mail: njain@niu.edu, phone: (815)-753-6964

1 Introduction

Structural estimation of Dynamic Discrete Choice (DDC) models has become increasingly popular in empirical economics. Examples include Keane and Wolpin (1997) in labor economics, Erdem and Keane (1995) in marketing, Imai and Krishna (2004) on crime and Rust (1987) in empirical industrial organization. Structural estimation is appealing for at least two reasons. First, it captures the dynamic forward-looking behavior of individuals, which is very important in understanding agents' behavior in various settings. For example, in labor market, individuals carefully consider future prospects when they switch occupations. Secondly, since the estimation is based on explicit solution of a structural model, it avoids the Lucas Critique. Hence, after the estimation, policy experiments can be relatively straightforwardly conducted by simply changing the estimated value of "policy" parameters and simulating the model to assess the change. However, one major obstacle in adopting the structural estimation method has been its computational burden. This is mainly due to the following two reasons.

First, in dynamic structural estimation, the likelihood or the moment conditions are based on the explicit solution of the dynamic model. In order to solve a dynamic model, we need to compute the Bellman equation repeatedly until the calculated expected value function converges. Second, in solving the Dynamic Programming (DP) Problem, the Bellman equation has to be solved at each possible point in the state space. The possible number of points in the state space increases exponentially with the increase in the dimensionality of the state space. This is commonly referred to as the "Curse of Dimensionality", and makes the estimation of the dynamic model infeasible even in a relatively simple setting.

In this paper, we propose an estimator that helps overcome the two computational difficulties of structural estimation. Our estimator is based on the Bayesian Markov Chain Monte Carlo (MCMC) estimation algorithm, where we simulate the posterior distribution by repeatedly drawing parameters from a pseudo-Markov Chain until convergence. In contrast to the conventional MCMC estimation approach, we combine the Bellman equation step and the MCMC algorithm step into a single hybrid solution-estimation step, which we iterate until convergence. The key innovation in our algorithm is that for a given state space point, we need to solve the Bellman equation only once between each estimation step. Since evaluating a Bellman equation once is as computationally demanding as computing a static model, the computational burden of estimating a DP model is in order of magnitude comparable to that of estimating a static model[1].

Furthermore, since we move the parameters according to the pseudo-MCMC algorithm after each Bellman step, we are "estimating" the model and solving for the DP problem at the same time. This is in contrast to conventional estimation methods that "estimate" the model only after solving the DP problem. In that sense, our estimation method is related to the algorithm advocated by Aguirreagabiria and Mira

[1] Ferrall (2005) also considers optimal mix of model solution and estimation algorithms.

(2001), Kasahara and Shimotsu (2005), which are an extension of the method developed by Hotz and Miller (1993), and Hotz, Miller, Sanders and Smith (1994). They propose to iterate the Bellman equation only once before constructing the likelihood. The estimation strategy, which is not based on the full solution of the model, has had difficulties dealing with unobserved heterogeneity. This is because this estimation method essentially recovers the value function from the observed choices of people at each point of the state space. But if there are unobserved heterogeneities, the state space points are unobservable in the data. In contrast to the above estimation algorithm, our estimation algorithm is based on the full solution of the dynamic programming problem, hence can account for a variety of unobservable heterogeneities. But we only need to solve the Bellman equation once between each estimation step.[2]

Specifically, we start with an initial guess of the expected value function (emax function). We then evaluate the Bellman equation for each state space point, if the number of state space points is finite, or for a subset of the state space grid points if the state space is continuous. We then use Bayesian MCMC to update the parameter vector. We update the emax function for a state space point by averaging with those past iterations in which the parameter vector is "close" to the current parameter vector and the state variables are either exactly the same as the current state variables (if the state space is finite) or close to the current state variables (if the state space is continuous). This method of updating the emax function is similar to Pakes and McGuire (2001) except in the important respect that we also include the parameter vector in determining the set of iterations over which averaging occurs.

Our algorithm also helps in the 'the Curse of Dimensionality' situation where the dimension of the state space is high. In most DP solution exercises involving a continuous state variable, the state space grid points, once determined, are fixed over the entire algorithm, as in Rust (1997). In our Bayesian DP algorithm, the state space grid points do not have to be the same for each solution-estimation iteration. In fact, by varying the state space grid points at each solution-estimation iteration, our algorithm allows for an arbitrarily large number of state space grid points by increasing the number of iterations. This is how our estimation method reduces the computational burden in high dimensional cases.

The main reason behind the computational advantage of our estimation algorithm is the use of information obtained from past iterations. In the conventional solution-estimation algorithm, at iteration t, most of the information gained in all past estimation iterations remains unused, except for the iteration $t-1$ likelihood and its Jacobian and Hessian in Classical ML estimation, and MCMC transition function in Bayesian MCMC estimation. In contrast, we extensively use the vast amount

[2] In contrast to Ackerberg (2004), where the entire DP problem needs to be solved for each parameter simulation, in our algorithm, the Bellman equation needs to be computed only once for each parameter value. Furthermore, there is an additional computational gain because our pseudo-MCMC algorithm guarantees that except for the initial burn-in simulations, most of the parameter draws are from a distribution close to the true posterior distribution. In Ackerberg's case, the initial parameter simulation and therefore the DP solution would be inefficient because at the initial stage, true parameter distribution is not known.

of computational results obtained in past iterations, especially those that are helpful in solving the DP problem. However, notice that if we use information on past iterations to update parameters, then the probability function that determines the next period parameter values is not a Markov transition function any more. We prove that under mild conditions, the probability function converges to the true MCMC transition function as we keep iterating the Bayesian MCMC algorithm. Hence, as the number of iterations increase, then our algorithm will become closer to the standard MCMC algorithm.

We demonstrate the performance of our algorithm by estimating a dynamic model of firm entry and exit choice with observed and unobserved heterogeneities. The unobserved random effects coefficients are assumed to have a continuous distribution function, and the observed characterisitcs are assumed to be continuous as well. It is well known that for a conventional Dynamic Programming Simulated Maximum Likelihood estimation strategy, this setup imposes a severe computational burden. The computational burden is due to the fact that during each estimation step, the DP problem has to be solved for each firm hundreds of times. Because of the observed heterogeneity, each firm has a different parameter value, and furthermore, because the random effects term has to be integrated out numerically via Monte-Carlo integration, for each firm, one has to simulate the random effects parameter hundreds of times, and for each simulation, solve for the DP problem. This is why most practitioners of structural estimation follow Heckman and Singer (1984) and assume discrete distributions for random effects and only allow for discrete types as observed characteristics.

We show that using our algorithm, the above estimation exercise becomes one that is computationally quite similar in difficulty to the Bayesian estimation of a static discrete choice model with random effects (see McCullogh and Rossi (1994) for details), and thus is feasible. Indeed, though simulation/estimation exercises we show that the computing time for our estimation exercise is around 8 times as fast and significantly more accurate than the conventional Random Effects Simulated Maximum Likelihood estimation algorithm. In addition to the experiments, we formally prove that under very mild conditions, the distribution of parameter estimates simulated from our solution-estimation algorithm converges to the true posterior distribution in probability as we increase the number of iterations.

Our algorithm shows that the Bayesian methods of estimation, suitably modified, can be used effectively to conduct full solution based estimation of structural dynamic discrete choice models. Thus far, application of Bayesian methods to estimate such models has been particularly difficult. The main reason is that the solution of the DP problem, i.e. the repeated calculation of the Bellman equation is computationally so demanding that the MCMC, which typically involves far more iterations than the standard Maximum Likelihood routine, quickly becomes infeasible with a relatively small increase in model complexity. One of the few examples of Bayesian estimation is Lancaster (1997). He successfully estimates the equilibrium search model where the Bellman equation can be transformed into an equation where all the information on optimal choice of the individual can be summarized in the reservation wage, and hence, there is no need for solving the value function. Another example is Geweke and Keane (1995) who estimate the DDC model without solving the DP problem. In contrast, our paper accomplishes Bayesian estimation based on full solution of the DP problem by simultaneously solving for the DP problem and iterating on the pseudo-MCMC algorithm. The difference turns out to be important because the estimation algorithms that are not based on the full solution of the model can only accomodate limited specification of unobserved heterogeneities.

Our estimation method not only makes Bayesian application to DDC models computationally feasible, but possibly even superior to the existing (non-Bayesian) methods, by reducing the computational burden of estimating a dynamic model to that of estimating a static one. Furthermore, the usually cited advantages of Bayesian estimation over classical estimation methods apply here as well. That is, first, the conditions for the convergence of the pseudo-MCMC algorithm are in general weaker than the conditions for the global maximum of the Maximum Likelihood (ML) estimator, as we show in this paper. Second, in MCMC, standard errors can be derived straightforwardly as a byproduct of the estimation routine, whereas in ML estimation, standard errors have to be computed usually either by inverting the numerically calculated Information Matrix, which is valid only in a large sample world, or by repeatedly bootstrapping and reestimating the model, which is computationally demanding.[3]

The organization of the paper is as follows. In Section 2, we present a general version of the DDC model and discuss conventional estimation methods as well as our Bayesian DP algorithm. In Section 3, we prove convergence of our algorithm under some mild conditions. In Section 4, we present a simple model of entry and exit. In Section 5, we present the simulation and estimation results of several experiments applied to the model of entry and exit. Finally, in Section 6, we conclude and briefly discuss future direction of this research. The Appendix contains all proofs.

2 The Framework

Let θ be the J dimensional parameter vector. Let S be the set of state space points and let s be an element of S. We assume that S is finite. Let A be the set of all possible actions and let a be an element of A. We assume A to be finite to study discrete choice models.

The value of choice a at parameter θ and state vector s is,

$$\mathcal{V}(s, a, \epsilon_a, \theta) = U(s, a, \epsilon_a, \theta) + \beta E_{\epsilon'} \left[V(s', \epsilon', \theta) \right] \tag{1}$$

[3] Osborne (2006) has applied the Bayesian DP algorithm to the estimation of the dynamic discrete choice model with random effects, and estimated the dynamic consumer brand choice model. Norets (2006) applied it to the DDC model with serially correlated state variables. We follow them in adopting the modified Metropolis-Hastings algorithm for the MCMC sampling instead of Gibbs sampler used in the earlier draft.

where s' is the next period's state variable, U is the current return function. ϵ is a vector whose a th element ϵ_a is a random shock to current returns to choice a. Finally, β is the discount factor. We assume that ϵ follows a multivariate distribution $F_\epsilon(\epsilon|\theta)$, which is independent over time. The expectation is taken with respect to the next period's shock ϵ'. We assume that the next period's state variable s' is a deterministic function of the current state variable s, current action a, and parameter θ [4]. That is,

$$s' = s'(s, a, \theta).$$

The value function is defined to be as follows.

$$V(s, \epsilon, \theta) = \max_{a \in A} \mathcal{V}(s, a, \epsilon_a, \theta)$$

We assume that the dataset for estimation includes variables which correspond to state vector s and choice a in our model but the choice shock ϵ is not observed. That is, the observed data is $Y_{N,T} \equiv \left\{s^d_{i,\tau}, a^d_{i,\tau}, F^d_{i,\tau}\right\}_{i=1,\tau=1}^{N,T}$ [5], where N is the number of firms and T is the number of time periods. Furthermore,

$$a^d_{i,\tau} = \arg\max_{a \in A} \mathcal{V}(s^d_{i,\tau}, a, \epsilon_a, \theta)$$

$$\begin{array}{rclcl} F^d_{i,\tau} & = & U(s^d_{i,\tau}, a^d_{i,\tau}, \epsilon_{a^d_{i,\tau}}, \theta) & \text{if} & \left(s^d_{i,\tau}, a^d_{i,\tau}\right) \in \Psi \\ & & 0 & & \text{otherwise.} \end{array}$$

The current period return is observable in the data only when the pair of state and choice variables belongs to the set Ψ. In the entry/exit problem of firms that we discuss later, profit of a firm is only observed when the incumbent firm stays in. In this case, Ψ is a set whose state variable is being an incumbent (and the capital stock) and the choice variable is staying in.

Let $\pi()$ be the prior distribution of θ. Furthermore, let $L(Y_{N,T}|\theta)$ be the likelihood of the model, given the parameter θ and the value function $V(.,.,\theta)$, which is the solution of the DP problem. Then, we have the following posterior distribution function of θ.

$$P(\theta|Y_{N,T}) \propto \pi(\theta) L(Y_{N,T}|\theta). \tag{2}$$

Let $\boldsymbol{\epsilon} \equiv \{\epsilon_{i,\tau}\}_{i=1,\tau=1}^{N,T}$. Because $\boldsymbol{\epsilon}$ is unobserved to the econometrician, the likelihood is an integral over it. That is, if we define $L(Y_{N,T}|\boldsymbol{\epsilon}, \theta)$ to be the likelihood conditional on $(\boldsymbol{\epsilon}, \theta)$, then,

$$L(Y_{N,T}|\theta) = \int L(Y_{N,T}|\boldsymbol{\epsilon}, \theta) dF_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon}|\theta).$$

The value function enters in the likelihood through choice probability, which is a component of the likelihood. That is,

$$P\left[a = a^d_{i,\tau} | s^d_{i,\tau}, V, \theta\right] = \Pr\left[\widehat{\epsilon}_{a,i,\tau} : a^d_{i,\tau} = \arg\max_{a \in A} \mathcal{V}(s^d_{i,\tau}, a, \widehat{\epsilon}_{a,i,\tau}, \theta)\right]$$

[4] This is a simplifying assumption for now. Later in the paper, we study random dynamics as well.

[5] We denote any variables with d superscript to be the data.

Below we briefly describe the conventional estimation approaches and then, the Bayesian dynamic programming algorithm we propose.

2.1 The Maximum Likelihood Estimation

The conventional ML estimation procedure of the dynamic programming problem consists of two main steps. First is the solution of the dynamic programming problem and the subsequent construction of the likelihood, which is called "the inner loop" and second is the estimation of the parameter vector, which is called "the outer loop".

1. **Dynamic Programming Step (Inner Loop)**: Given parameter vector θ, we solve the Bellman equation, given by equation 1. This typically involves several steps.

   (a) First, the random choice shock, ϵ is drawn a fixed number of times, say, M, generating $\epsilon^{(m)}, m = 1, ..., M$. At iteration 0, set initial guess of the value function to be, for example, zero. That is, $V^{(0)}(s, \epsilon^{(m)}, \theta) = 0$ for every $s \in S$, $\epsilon^{(m)}$. We also let the expected value function (Emax function) to be zero, i.e., $E_{\epsilon'}\left[V^{(0)}(s, \epsilon', \theta)\right] = 0$ for every $s \in S$.

   (b) Assume we are at iteration t of the DP algorithm. Given $s \in S$ and $\epsilon^{(m)}$, the value of every choice $a \in A$ is calculated. For the Emax function, we use the approximated expected value function $\widehat{E}_{\epsilon'}\left[V^{(t-1)}(s', \epsilon', \theta)\right]$ computed at the previous iteration $t-1$ for every $s' \in S$. Hence, the iteration t value of choice a is,

   $$\mathcal{V}^{(t)}(s, a, \epsilon_a^{(m)}, \theta) = U(s, a, \epsilon_a^{(m)}, \theta) + \beta \widehat{E}_{\epsilon'}\left[V^{(t-1)}(s', \epsilon', \theta)\right].$$

   Then, we compute the value function,

   $$V^{(t)}(s, \epsilon^{(m)}, \theta) = \max_{a \in A} \mathcal{V}^{(t)}(s, a, \epsilon_a^{(m)}, \theta). \tag{3}$$

   The above calculation is done for every $s \in S$ and $\epsilon^{(m)}$, $m = 1, ..., M$.

   c. The approximation for the expected value function is computed by taking the average of value functions over simulated choice shocks as follows.

   $$\widehat{E}_{\epsilon'}\left[V^{(t)}(s', \epsilon', \theta)\right] \equiv \frac{1}{M} \sum_{m=1}^{M} V^{(t)}(s', \epsilon^{(m)}, \theta) \tag{4}$$

   Steps b) and c) have to be done repeatedly for every state space point $s \in S$. Furthermore, all three steps have to be repeated until the value function converges. That is, for a small $\delta > 0$,

   $$\left|V^{(t)}(s, \epsilon^{(m)}, \theta) - V^{(t-1)}(s, \epsilon^{(m)}, \theta)\right| < \delta$$

   for all $s \in S$ and $m = 1, .., M$.

2. **Likelihood Construction**

   The important increment of the likelihood is the conditional choice probability $P\left[a = a^d_{i,\tau} | s^d_{i,\tau}, V, \theta\right]$ given the state $s^d_{i,\tau}$, value function V and the parameter θ. For example, suppose that the per period return function is specified as follows.

$$U(s, a, \epsilon_a^{(m)}, \theta) = \widetilde{U}(s, a, \theta) + \epsilon_a^{(m)},$$

   where $\widetilde{U}(s, a, \theta)$ is the deterministic component of the per period return function. Also, denote,

$$\widetilde{\mathcal{V}}(s, a, \theta) = \widetilde{U}(s, a, \theta) + \beta \widehat{E}_{\epsilon'}\left[V(s', \epsilon', \theta)\right]$$

   to be the deterministic component of the value of choosing action a. Then,

$$P\left[a^d_{i,\tau} | s^d_{i,\tau}, V, \theta\right] = P\left[\epsilon_a - \epsilon_{a^d_{i,\tau}} \leq \widetilde{\mathcal{V}}(s, a^d_{i,\tau}, \theta) - \widetilde{\mathcal{V}}(s, a, \theta); \forall a \neq a^d_{i,\tau} | s^d_{i,\tau}, V, \theta\right]$$

   which becomes a multinomial probit specification when the error term ϵ is assumed to follow a joint normal distribution.

3. **Likelihood Maximization routine (Outer Loop)**

   Supppose we have K parameters to estimate. In a typical Maximum Likelihood estimation routine, where one uses Newton hill climbing algorithm, at iteration t, likelihhood is derived under the original parameter vector $\theta^{(t)}$ and under the perturbed parameter vector $\theta^{(t)} + \Delta\theta_j$, $j = 1, ..., K$. The perturbed likelihood is used together with the original likelihood to derive the new direction of the hill climbing algorithm. This is done to derive the parameters for the iteration $t + 1$, $\theta^{(t+1)}$. That is, during a single ML estimation routine, the DP problem needs to be solved in full $K + 1$ times. Furthermore, often the ML estimation routine has to be repeated many times until convergence is achieved. During a single iteration of the maximization routine, the inner loop algorithm needs to be executed at least as many times as the number of parameters plus one. Since the estimation requires many iterations of the maximization routine, the entire algorithm is usually computationally extremely burdensome.

2.2 The conventional Bayesian MCMC estimation

A major computational issue in Bayesian estimation method is that the posterior distribution, given by equation 2, is a high-dimensional and complex function of the parameters. Instead of directly simulating the posterior, we adopt the Markov Chain Monte Carlo (MCMC) strategy and construct a transition density from current parameter θ to the next iteration parameter θ', $f(\theta, \theta')$, which satisfies, among other more technical conditions, the following equality.

$$P(\theta | Y_{N,T}) = \int f(\theta, \theta') P(\theta' | Y_{N,T}) d\theta'$$

We simulate from the transition density the sequence of parameters $\left\{\theta^{(s)}\right\}_{s=1}^{t}$, which is known to converge to the correct posterior.

The conventional Bayesian estimation method applied to the DDC model proceeds in the following three main steps.

Metropolis-Hastings (M-H) Step: The Metropolis-Hastings algorithm is a Markov Chain simulation algorithm used to draw from a complex target distribution. See Robert and Casella (2004) for more details on the M-H algorithm. In our case, the traget density is proportional to $\pi(\theta)L(Y_{N,T}|\theta)$. Given $\theta^{(t)}$, the parameter vector at iteration t, draw the new parameter vector $\theta^{(t+1)}$ as follows: First, draw the candidate parameter vector $\theta^{*(t)}$ from a candidate generating density (or proposal density) $q\left(\theta^{(t)}, \theta^{*(t)}\right)$. Then, accept $\theta^{*(t)}$, i.e. set $\theta^{(t+1)} = \theta^{*(t)}$ with probability

$$\lambda\left(\theta^{(t)}, \theta^{*(t)}\right) = \min\left\{\frac{\pi\left(\theta^{*(t)}\right) L\left(Y_{N,T}|\theta^{*(t)}\right) q\left(\theta^{*(t)}, \theta^{(t)}\right)}{\pi\left(\theta^{(t)}\right) L\left(Y_{N,T}|\theta^{(t)}\right) q\left(\theta^{(t)}, \theta^{*(t)}\right)}, 1\right\}$$

otherwise, reject $\theta^{*(t)}$, i.e. set $\theta^{(t+1)} = \theta^{(t)}$.

Notice that since the likelihood is a function of the value function, during each M-H step, in order to compute the proposal density, for each $\theta^{*(t)}$ the DP problem needs to be solved and value function derived. Hence, the MCMC algorithm is the "Outer Loop" of the estimation algorithm, and we need the following Dynamic Programming step within the Hastings-Metropolis Step as the "Inner Loop".

Dynamic Programming Step: The Bellman equation, given by equation 1, is iterated until convergence for the given parameter vector $\theta^{(t)}$ and the candidate vector $\theta^{*(t)}$. This solution algorithm for the DP Step is similar to the Maximum Likelihood algorithm discussed above.

We now present our algorithm for estimating the parameter vector θ. We call it the Bayesian Dynamic Programming Algorithm. The key innovation of our algorithm is that we solve the dynamic programming problem and estimate the parameters simultaneously, rather than sequentially.

2.3 The Bayesian Dynamic Programming Estimation

Our method is similar to the conventional Bayesian algorithm in that based on the value function we compute at each estimation step, we construct an algorithm that is a modified version of the Metropolis-Hastings algorithm described above to generate a sequence of parameter simulations. The main difference between the Bayesian DP algorithm and the conventional algorithm is that during each estimation step, we do not solve the DP problem in full. In fact, during each modified Metropolis-Hastings step, we iterate the DP algorithm only once.

A key issue in solving the DP problem is the way the expected value function (or the Emax function) is approximated. In conventional methods, this approximation is given by equation 4. In contrast, we approximate the emax function by averaging over a subset of past iterations. Let $\Omega^{(t)} \equiv \left\{\epsilon^{(s)}, \theta^{*(s)}, V^{(s)}\right\}_{s=1}^{t}$ be the history of shocks, parameters and value functions up to the current iteration t[6]. Let $\mathcal{V}^{(t)}(s, a, \epsilon_a^{(t)}, \theta^{*(t)}, \Omega^{(t-1)})$ be the value of choice a and let $V^{(t)}(s, \epsilon^{(t)}, \theta^{*(t)}, \Omega^{(t-1)})$ be the value function derived at iteration t of our solution/estimation algorithm. Then, the value function and the approximation $\hat{E}{}_{\epsilon'}^{(t)}\left[V(s', \epsilon', \theta)|\Omega^{(t-1)}\right]$ for the expected value function $E_{\epsilon'}\left[V(s', \epsilon', \theta)\right]$ at iteration t are defined recursively as follows.

$$\hat{E}{}_{\epsilon'}^{(t)}\left[V(s', \epsilon', \theta)|\Omega^{(t-1)}\right] \equiv \sum_{n=1}^{N(t)} V^{(t-n)}(s', \epsilon^{(t-n)}, \theta^{*(t-n)}, \Omega^{(t-n-1)}) \frac{K_h(\theta - \theta^{*(t-n)})}{\sum_{k=1}^{N(t)} K_h(\theta - \theta^{*(t-k)})}, \tag{5}$$

and,

$$\begin{aligned} \mathcal{V}^{(t-n)}(s, a, \epsilon_a^{(t-n)}, \theta^{*(t-n)}, \Omega^{(t-n-1)}) \quad = \quad & U(s, a, \epsilon_a^{(t-n)}, \theta^{*(t-n)}) \\ & + \beta \hat{E}{}_{\epsilon'}^{(t-n)}\left[V(s', \epsilon', \theta^{*(t-n)})|\Omega^{(t-n-1)}\right], \end{aligned}$$

$$V^{(t-n)}(s, \epsilon^{(t-n)}, \theta^{*(t-n)}, \Omega^{(t-n-1)}) = Max_{a \in A} \mathcal{V}^{(t-n)}(s, a, \epsilon_a^{(t-n)}, \theta^{*(t-n)}, \Omega^{(t-n-1)})$$

where $K_h()$ is a kernel with bandwidth $h > 0$. That is,

$$K_h(z) = \frac{1}{h^J} K(\frac{z}{h}).$$

K is a nonnegative, continuous, bounded real function which is symmetric around zero and integrates to one. i.e. $\int K(z)dz = 1$. Furthermore, we assume that $\int zK(z)dz < \infty$.

The approximated expected value function given by equation 5 is the weighted average of value functions of $N(t)$ most recent iterations. The sample size of the average, $N(t)$, increases with t. Futhermore, we let $t - N(t) \to \infty$ as $t \to \infty$. The weights are high for the value functions at iterations with parameters close to the current parameter vector $\theta^{(t)}$. This is similar to the idea of Pakes and McGuire (2002), where the expected value function is the average of the past N iterations. In their algorithm, averages are taken only over the value functions that have the same state variable as the current state variable s. In our case, averages are taken over the value functions that have the same state variable as the current state variable s as well as parameters that are close to the current parameter $\theta^{*(t)}$.

We now describe the complete Bayesian Dynamic Programming algorithm at iteration t. Suppose that $\left\{\epsilon^{(l)}\right\}_{l=1}^{t}$, $\left\{\theta^{*(l)}\right\}_{l=1}^{t}$ are given and for all discrete $s \in S$, $\left\{V^{(l)}(s, \epsilon^{(l)}, \theta^{*(l)})\right\}_{l=1}^{t}$ is also given. Then, we update the value function and the parameters as follows.

[6] The simulated shocks $\epsilon^{(s)}$ are those used for calculating the value function.

1. **Modified Metropolis-Hastings Step**[7]: Draw the new parameters $\theta^{(t+1)}$ as follows: First, draw the candidate parameter $\theta^{*(t)}$ from the proposal density $q\left(\theta^{(t)}, \theta^{*(t)}\right)$. then, accept $\theta^{*(t)}$, i.e. set $\theta^{(t+1)} = \theta^{*(t)}$ with probability

$$
\begin{aligned}
&\lambda\left(\theta^{(t)}, \theta^{*(t)} | \Omega^{(t-1)}\right) \\
= \; & \min\left\{\frac{\pi\left(\theta^{*(t)}\right) L\left(Y_{N,T} | \theta^{p(t)}, \widehat{E}_{\epsilon'}^{(t)}\left[V\left(., \theta^{*(t)}\right) | \Omega^{(t-1)}\right]\right) q\left(\theta^{*(t)}, \theta^{(t)}\right)}{\pi\left(\theta^{(t)}\right) L\left(Y_{N,T} | \theta^{(t)}, \widehat{E}_{\epsilon'}^{(t)}\left[V\left(., \theta^{(t)}\right) | \Omega^{(t-1)}\right]\right) q\left(\theta^{(t)}, \theta^{*(t)}\right)}, 1\right\} \quad (6)
\end{aligned}
$$

otherwise, reject $\theta^{*(t)}$, i.e. set $\theta^{(t+1)} = \theta^{(t)}$.

2 **Bellman Equation Step**: During each Metropolis-Hastings step, we need to solve for the expected value function $\widehat{E}_{\epsilon'}^{(t)}\left[V(.,)|\Omega^{(t-1)}\right]$ for parameters $\theta^{(t)}$ and $\theta^{*(t)}$. To do so for all $s \in S$, we follow equation 5. For use in future iterations, we simulate the value function by drawing $\epsilon^{(t)}$ to derive,

$$
\mathcal{V}^{(t)}(s, a, \epsilon_a^{(t)}, \theta^{*(t)}, \Omega^{(t-1)}) = U(s, a, \epsilon_a^{(t)}, \theta^{*(t)}) + \beta \hat{E}{}_{\epsilon'}^{(t)}\left[V(s', \epsilon', \theta^{*(t)}) \mid \Omega^{(t-1)}\right],
$$

$$
V^{(t)}(s, \epsilon^{(t)}, \theta^{*(t)}, \Omega^{(t-1)}) = \max_{a \in A} \mathcal{V}^{(t)}(s, a, \epsilon_a^{(t)}, \theta^{*(t)}, \Omega^{(t-1)}).
$$

We repeat Steps 1 to 2 until the sequence of the parameter simulations converges to a stationary distribution. In our algorithm, in addition to the Dynamic Programming and Bayesian methods, nonparametric kernel techniques are also used to approximate the value function. Notice that the convergence of kernel based approximation is not based on the large sample size of the data, but based on the number of Bayesian DP iterations. The Bellman equation step (Step 2) is only done once during a single estimation iteration. Hence, the Bayesian DP algorithm avoids the computational burden of solving for the DP problem during each estimation step, which involves repeated evaluation of the Bellman equation.

It is important to notice that the modified Metropolis-Hastings algorithm is not a Markov Chain[8]. This is because it involves value functions calculated in past iterations. Hence, convergence of our algorithm is by no means trivial. In the next section, we prove that under some mild assumptions the distribution of the parameters generated by our algorithm converges to the true posterior in probability.

Both Osborne (2006) and Norets (2006) approximate the expected value function using the value functions computed in the past iterations evaluated at the past parameter draws $\theta^{(t-n)}$. Here, we use the value functions evaluated at the past proposal

[7] We are grateful to Andriy Norets for pointing out the flaw in the Gibbs Sampling scheme adopted in the earlier draft.

[8] We are grateful to Peter Rossi for emphasizing it.

parameter draws $\theta^{*(t-n)}$. We chose to do so because given $\theta^{(t)}$ it is easier to control the random movement of $\theta^{*(t)}$ than the random movement of $\theta^{(t+1)}$, since $\theta^{*(t)}$ is drawn from a known distribution function which we can easily change, whereas $\theta^{(t+1)}$ comes from a complex distribution which involves the solution of the dynamic model. For example, if in modified Metropolis-Hastings algorithm the parameter $\theta^{(t)}$ is "stuck" at a value for many iterations, then the value functions are only evaluated at that parameter value. But even then, $\theta^{*(t)}$ moves around so that we can compute the value functions at the parameter values around $\theta^{(t)}$, which becomes useful in computing the expected value function when the parameter $\theta^{(t)}$ finally moves to a different value. Furthermore, by setting the support of the proposal density to be the entire parameter set Θ, which we assume to be compact, we can insure that at each point θ in Θ, the proposal density draw $\theta^{*(t)}$ will visit the open neighborhood of θ arbitrarily many times as we increase the iteration to infinity, which turns out to be the main reason why the expected value function approximation of the Bayesian DP algorithm converges to the true ones. By keeping the conditional variance of the proposal density given $\theta^{(t)}$ small, we can guarantee that the invariant distribution of θ^* is not very different from that of θ.

3 Theoretical Results

In this section, we prove the convergence of the Bayesian DP algorithm. To facilitate the proof, we modify the Bellman equation step slightly. That is, we simulate the value function by drawing $\epsilon^{(t)}$ to derive,

$$\mathcal{V}^{(t)}(s,a,\epsilon_a^{(t)},\theta^{*(t)},\Omega^{(t-1)}) = \widetilde{U}(s,a,\epsilon_a^{(t)},\theta^{*(t)}) + \beta \hat{E}{}_{\epsilon'}^{(t)}\left[V(s',\epsilon',\theta^{*(t)}) \mid \Omega^{(t-1)}\right],$$

$$V^{(t)}(s,\epsilon^{(t)},\theta^{*(t)},\Omega^{(t-1)}) = \max_{a\in A}\mathcal{V}^{(t)}(s,a,\epsilon_a^{(t)},\theta^{*(t)},\Omega^{(t-1)}).$$

where

$$\widetilde{U}(s,a,\epsilon_a^{(t)},\theta^{*(t)}) = Min\left\{Max\left\{\widetilde{U}(s,a,\epsilon_a^{(t)},\theta^{*(t)}),-M_U\right\},M_U\right\}$$

for a large positive M_U. This makes the utility function used in the Bellman equation uniformly bounded, which simplifies the proof. This modification does not make any difference in practice because M_U can be set arbitrarily large. We also denote V to the the value function of the following Bellman equation.

$$V(s,\epsilon_a^{(m)},\theta) = Max\left\{\widetilde{U}(s,a,\epsilon_a,\theta) + \beta E_{\epsilon',s'}\left[V(s',\epsilon',\theta)\right]\right\}.$$

Next we show that under some mild assumptions, our algorithm generates a sequence of parameters $\theta^{(1)},\theta^{(2)},...$ which converges in probability to the correct posterior distribution.

Assumption 1: Parameter space $\Theta \subseteq R^J$ is compact, i.e. closed and bounded in the Euclidean space R^J. We set the proposal density $q(\theta,.)$ to be continuously differentiable, strictly positive and uniformly bounded in the parameter space given any $\theta \in \Theta$.

Compactness of the parameter space is a standard assumption used in proving the convergence of MCMC algorithm. See, for example, McCullogh and Rossi (1994). It is often not necessary but simplifies the proofs. An example of the proposal density that satisfies Assumption 1 is the multivariate normal density, truncated to only cover the compact parameter space.

Assumption 2: For any $s \in S$, $a \in A$, and ϵ, $\theta \in \Theta$, $\left|\widetilde{U}(s, a, \epsilon_a, \theta)\right| < M_U$ for some $M_U > 0$. Also, $\widetilde{U}(s, a, ., \theta)$ is a nondecreasing function in ϵ and $\widetilde{U}(s, a, ., .)$ satisfies the Lipschitz condition in terms of ϵ and θ. Also, the distribution of ϵ, has a density function $dF(\epsilon, \theta)$ which is continuous in θ.

Assumption 3: β is known and $\beta < 1$.

Assumption 4: For any $s \in S$, ϵ and $\theta \in \Theta$, $V^{(0)}(s, \epsilon, \theta) < M_I$ for some $M_I > 0$. Furthermore, $V^0(s, ., .)$ also satisfies the Lipschitz condition in terms of ϵ and θ.

Assumptions 2 and 3, and 4 jointly make $V^{(t)}(s, \epsilon, \theta)$ and hence $\hat{E}{}^{(t)}_{\epsilon'}[V(s', \epsilon', \theta)]$, $t = 1, ...$ uniformly bounded, measurable, continuous and satisfying the Lipschitz condition as well.

Assumption 5: $\pi(\theta)$ is positive and bounded for any $\theta \in \Theta$. Similarly, for any $\theta \in \Theta$ and V uniformly bounded, $L(Y_{NT}|\theta, V) > 0$ and bounded.

Assumption 6: The bandwidth h is a function of N and as $N \to \infty$, $h(N) \to 0$ and $Nh(N)^{2J} \to \infty$. The kernel K has an absolutely integrable Fourier transform.

Assumption 7: $N(t)$ is nondecreasing in t, increases at most by one for a unit increase in t, and $N(t) \to \infty$. Furthermore, $t - N(t) \to \infty$. Define the sequence $t(l)$, $\widetilde{N}(l)$ as follows. For some $t > 0$, define $t(1) = t$, and $\widetilde{N}(1) = N(t)$. Let $t(2)$ be such that $t(2) - N(t(2)) = t(1)$. Such $t(2)$ exists from the assumption on $N(t)$. Also, let $\widetilde{N}(2) = N(t(2))$. Similarly, for any $l > 2$, let $t(l+1)$ be such that $t(l+1) - N(t(l+1)) = t(l)$, and let $\widetilde{N}(l+1) = N(t(l+1))$. Assume that there exists a finite constant $A > 0$ such that $\widetilde{N}(l+1) < A\widetilde{N}(l)$.

An example of a sequence that satisfies Assumption 8 is:

$$t(l) \equiv \frac{l(l+1)}{2}, \widetilde{N}(l) = l$$

and,

$$N(t) = l \text{ for } t(l) \leq t < t(l+1).$$

The following two Lemmas establish some properties that are used in the later proofs.

Lemma 1: Let $h(.)$, ε_0 and $g(.)$ be defined as follows.

$$h(\theta^*) \equiv \inf_{\theta \in \Theta} q(\theta, \theta^*)$$

$$\varepsilon_0 = \int h\left(\tilde{\theta}\right) d\tilde{\theta}$$

$$g(\theta^*) \equiv \frac{h(\theta^*)}{\int h\left(\tilde{\theta}\right) d\tilde{\theta}}$$

Then,

$$0 < \varepsilon_0 \leq 1$$

and for any θ, $\theta^* \in \Theta$,

$$\varepsilon_0 g(\theta^*) \leq q(\theta, \theta^*).$$

Proof:

By Assumption 1 (Compactness of parameter space), for any $\theta^* \in \Theta$,

$$h(\theta^*) \equiv \inf_{\theta \in \Theta} q(\theta, \theta^*)$$

exists, is strictly positive and uniformly bounded below. Notice that $h(.)$ is Lebesgue integrable. Furthermore, for any $\theta \in \Theta$

$$\varepsilon_0 g(\theta^*) = h(\theta^*) \leq q(\theta, \theta^*).$$

Next, since q satisfies Assumption 1, $g(.)$ is strictly positive, bounded and $\int g(\theta)\, d\theta = 1$. Hence, $g(.)$ as a function is a density function. Also, by construction, ε_0 is a strictly positive constant. Finally, since both $g(.)$ and $q(\theta, .)$ are densities and integrate to 1, $0 < \varepsilon_0 \leq 1$.

Lemma 1 implies that the proposal density of the modified Metropolis-Hastings algorithm has an important property: regardless of the current parameter values or the number of iterations, every parameter value in the compact parameter space is visited with a strictly positive probability.

Lemma 2: Let $\widetilde{h}(.)$, be a continuously differentiable function which satisfies the following inequality.

$$\widetilde{h}(\theta^*) \geq \sup_{\theta \in \Theta} q(\theta, \theta^*).$$

Let ε_1 and $\widetilde{g}()$ be defined as follows.

$$\varepsilon_1 \equiv \int \widetilde{h}\left(\tilde{\theta}\right) d\tilde{\theta}$$

$$\widetilde{g}(\theta^*) \equiv \frac{\widetilde{h}(\theta^*)}{\int \widetilde{h}\left(\tilde{\theta}\right) d\tilde{\theta}}$$

Then,

$$1 \leq \varepsilon_1 < \infty$$

and for any θ, $\theta^* \in \Theta$

$$q(\theta, \theta^*) \leq \varepsilon_1 \widetilde{g}(\theta^*)$$

Proof: Using similar logic as in Lemma 1, one can show that for any $\theta^* \in \Theta$,

$$\sup_{\theta \in \Theta} q(\theta, \theta^*)$$

exists and is bounded. Then, $\widetilde{g}(\theta)$ and ε_1 satisfy the conditions of the Lemma.

Lemma 2 implies that the proposal density is bounded above, the bound being independent of the current parameter value or the number of iterations.

Theorem 1: Suppose Assumptions 1 to 7 are satisfied for $V^{(t)}$, π, L, $\boldsymbol{\epsilon}$ and θ. Then, the sequence of approximated value functions $V^{(t)}(s,\epsilon,\theta)$ converges in probability uniformly over s, ϵ and $\theta \in \Theta$ to $V(s,\epsilon,\theta)$ as $t \to \infty$. Also, $\hat{E}{}^{(t)}_{\epsilon'}\left[V(s',\epsilon',\theta)\right]$ converges to $E_{\epsilon'}\left[V(s',\epsilon',\theta)\right]$ in probability uniformly over $s' \in S$, $\theta \in \Theta$.

Proof: See the Appendix.

Corollary 1: Suppose Assumptions 1 to 7 are satisfied. Then Theorem 1 implies that $\lambda\left(\theta^{(t)},\theta^{*(t)}|\Omega^{(t-1)}\right)$ converges to $\lambda\left(\theta^{(t)},\theta^{*(t)}\right)$ in probability uniformly.

Proof: Recall Equations 5 and 6. Since $\widehat{E}^{(t)}_{\epsilon'}[V] \to EV$ in probability uniformly in s, $\theta \in \Theta$, by compactness of Θ, the result follows.

Theorem 2: Suppose Assumptions 1 to 6 are satisfied for $V^{(t)}$, $t = 1,...$, π, L, ϵ and θ. Suppose $\theta^{(t)}, t = 1,...$ is generated by a modified Metropolis-Hastings Algorithm described earlier, where $\lambda\left(\theta^{(t)},\theta^{*(t)}|\Omega^{(t-1)}\right)$ converges to $\lambda\left(\theta^{(t)},\theta^{*(t)}\right)$ in probability uniformly. Then, $\theta^{(t)}$ converges to $\widetilde{\theta}^{(t)}$ in probability, where $\widetilde{\theta}^{(t)}$ is a Markov chain generated by the Metropolis-Hastings Algorithm with proposal density $q(\theta,\theta^{(*)})$ and acceptance probability function $\lambda\left(\theta,\theta^{(*)}\right)$.

Proof: See the Appendix.

Corollary 2: The sequence of parameter simulations generated by the Metropolis-Hastings algorithm with proposal density $q(\theta,\theta^*)$ and the acceptance probability $\lambda(\theta,\theta^*)$ converge to the true posterior.

Proof of Corollary 2: Here, we use the Corollary 7.7 of Robert and Casella (2004), which states that if the Metropolis-Hastings Markov Chain has invariant probability density f and if there exist positive ϵ and δ such that $q(x,y) > \epsilon$ if $|x-y| < \delta$, then for any $h \in L^1(f)$

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T} h\left(\theta^{(t)}\right) = \int h(\theta) f(\theta)\, d\theta$$

and

$$\lim_{n\to\infty}\left\|\int K^n(\theta,.)\,\mu(d\theta) - f\right\|_{TV} = 0$$

for arbitrary initial distribution μ, where $K^n(\theta,.)$ is the transition kernel for n iterations and the norm is the total variation norm. By construction the Metropolis-Hastings Markov Chain has an invariant probability density, which is proportional to $\pi(\theta)\,L(Y_{N,T}|\theta)$, which is assumed to be bounded and positive on Θ. Since the proposal density is strictly positive over the parameter space, the condition for the proposal density is also satisfied.

By Corollary 2, we can conclude that the distribution of the sequence of parameters $\theta^{(t)}$ generated by the Bayesian DP algorithm converges in probability to the true posterior distribution in probability.

To understand the basic logic of the proof of Theorem 1, suppose that the parameter $\theta^{(t)}$ stays fixed at a value θ^* for all iterations t. Then, equation (5) reduces to,

$$\widehat{E}_{\epsilon'}\left[V(s',\epsilon',\theta^*)\right] = \frac{1}{N(t)}\sum_{n=1}^{N(t)} V^{(t-n)}(s',\epsilon^{(t-n)},\theta^*).$$

Then, our algorithm boils down to a simple version of the machine learning algorithm discussed in Pakes and McGuire (2001) and Bertsekas and Tsitsiklis (1996). They approximate the expected value function by taking the average over all past value function iterations whose state space point is the same as the state space point $s\prime$. Bertsekas and Tsitsiklis (1996) discuss the convergence issues and show that under some assumptions the sequence of the value functions from the machine learning algorithm converges to the true value function almost surely. The difficulty of the proofs lies in extending the logic of the convergence of the machine learning algorithm to the framework of estimation, that is, the case where the parameter vector moves around as well. Our answer to this issue is simple: for a parameter value $\theta^* \in \Theta$ at an iteration t, we look at the past iteration and use value function of the parameters $\theta^{(t-n)}$ that are very close to θ^*. Then, the convergence is very similar to that where the parameter vector is fixed, as long as the number of the past value functions that can be used can be made arbitrarily large. We know from Lemma 1 that every parameter vector in the compact parameter space Θ has strictly positive probability of being drawn. Then, by increasing the number of iterations, we can make the number of draws for every finite open cover in the parameter space Θ as large as we want and still the probability of it can be made arbitrarily close to 1. It is important to note that for the convergence of the value function, the estimation algorithm does not have to be Markov. The only requirement is that during the iteration each parameter in Θ has a strictly positive probability of being drawn.

3.1 Random Effects

Consider a model where for a subset of the parameters each firm has different value. The parameter of the model is $\left(\theta_{(1)},\theta_{(2)}\right)$ where $\theta_{(1)}$ is the parameter vector for the distribution of the random coefficients and $\theta_{(2)}$ is the vector of other parameters. The parameter vector of the firm i is $\left(\widetilde{\theta}_i,\theta_{(2)}\right)$. That is, $\widetilde{\theta}_i$ is the set of random effects parameters which take different values for each firm whose distribution is defined to be $f\left(\widetilde{\theta}_i|\theta_{(1)}\right)$. Instead of explicitly integrating the likelihood over $\widetilde{\theta}_i$, we follow the commonly adopted and computationally effective procedure of treating each $\widetilde{\theta}_i$ as parameters and drawing it from its density. It is known (see McCullogh and Rossi (1994), Chib and Greenberg (1996)) that instead of drawing the entire parameter vector $\left(\{\theta_i\}_{i=1}^N,\theta_{(1)},\theta_{(2)}\right)$ at once, it is often simpler to partition the parameter vector into several blocks and draw the parameters of each block separately given the other ones. Here, we propose to draw them in the following 3 blocks. At iteration t the blocks are:

$$\begin{array}{ll} \text{Block 1: draw} & \left\{\widetilde{\theta}_i^{(t+1)}\right\}_{i=1}^{N} \text{ given } \theta_{(1)}^{(t)}, \theta_{(2)}^{(t)} \\ \text{Block 2: draw} & \theta_{(1)}^{(t+1)} \text{ given } \left\{\widetilde{\theta}_i^{(t+1)}\right\}_{i=1}^{N}, \theta_{(2)}^{(t)} \\ \text{Block 3: draw} & \theta_{(2)}^{(t+1)} \text{ given } \left\{\widetilde{\theta}_i^{(t+1)}\right\}_{i=1}^{N}, \theta_{(1)}^{(t+1)} \end{array}$$

Below we describe in detail the algorithm at each block[9].

Block 1 **Modified Metropolis-Hastings Step for drawing the Random Effects** $\widetilde{\theta}_i$: For firm i, we draw the new random effects parameters $\widetilde{\theta}_i^{(t+1)}$ as follows: We set the proposal density as the distribution function of $\widetilde{\theta}_i$, that is, $f\left(\widetilde{\theta}_i|\theta_{(1)}\right)$. Notice that the prior is a function of $\theta_{(1)}$ and $\theta_{(2)}$, and not of $\widetilde{\theta}_i$. Hence for drawing $\widetilde{\theta}_i$ given $\theta_{(1)}$ and $\theta_{(2)}$, the prior is irrelevant. Similarly given $\theta_{(1)}$ the likelihood increment of firms other than i is also irrelevant in drawing $\widetilde{\theta}_i$. Therefore, we draw $\widetilde{\theta}_i$ from the likelihood increment of firm i, which can be written as follows:

$$L_i\left(Y_{i,T}|\left(\widetilde{\theta}_i, \theta_{(2)}\right)\right) f\left(\widetilde{\theta}_i|\theta_{(1)}\right)$$

where we denote $L_i\left(Y_{i,T}|\left(\widetilde{\theta}_i, \theta_{(2)}\right)\right)$ to be

$$L_i\left(Y_{i,T}|\left(\widetilde{\theta}_i, \theta_{(2)}\right)\right) \equiv L\left(Y_{i,T}|\left(\widetilde{\theta}_i, \theta_{(2)}\right), \widehat{E}_{\epsilon'}^{(t)}\left[V\left(., \widetilde{\theta}_i, \theta_{(2)}\right)|\Omega^{(t-1)}\right]\right)$$

Now, we draw the candidate parameter $\widetilde{\theta}_i^{*(t)}$ from the proposal density $f\left(\widetilde{\theta}_i^{*(t)}|\theta_{(1)}\right)$. Then, accept $\widetilde{\theta}_i^{*(t)}$, i.e. set $\widetilde{\theta}_i^{(t+1)} = \widetilde{\theta}_i^{*(t)}$ with probability

$$\begin{aligned} & \lambda\left(\theta^{(t)}, \widetilde{\theta}_i^{*(t)}|\Omega^{(t-1)}\right) \\ = \; & \min\left\{\frac{L_i\left(Y_{i,T}|\left(\widetilde{\theta}_i^{*(t)}, \theta_{(2)}^{(t)}\right)\right) f\left(\widetilde{\theta}_i^{*(t)}|\theta_{(1)}^{(t)}\right) f\left(\widetilde{\theta}_i^{(t)}|\theta_{(1)}^{(t)}\right)}{L_i\left(Y_{i,T}|\left(\widetilde{\theta}_i^{(t)}, \theta_{(2)}^{(t)}\right)\right) f\left(\widetilde{\theta}_i^{(t)}|\theta_{(1)}^{(t)}\right) f\left(\widetilde{\theta}_i^{*(t)}|\theta_{(1)}^{(t)}\right)}, 1\right\} \\ = \; & \min\left\{\frac{L_i\left(Y_{i,T}|\left(\widetilde{\theta}_i^{*(t)}, \theta_{(2)}^{(t)}\right)\right)}{L_i\left(Y_{i,T}|\left(\widetilde{\theta}_i^{(t)}, \theta_{(2)}^{(t)}\right)\right)}, 1\right\} \end{aligned}$$

otherwise, reject $\widetilde{\theta}_i^{*(t)}$, i.e. set $\widetilde{\theta}_i^{(t+1)} = \widetilde{\theta}_i^{(t)}$.

[9] The procedure described below is similar to that of Osborne (2006)

Block 2 **Drawing** $\theta_{(1)}^{(t+1)}$: Conditional on $\left\{\widetilde{\theta}_i^{(t+1)}\right\}_{i=1}^{N}$, the density of $\theta_{(1)}^{(t+1)}$ is proportional to

$$\prod_{i=1}^{N} f\left(\widetilde{\theta}_i^{(t+1)} | \theta_{(1)}\right).$$

Drawing from the above density is straightforward as it does not involve the solution of the dynamic programming problem.

Block 3 **Modified Metroplis-Hastings Algorithm for drawing** $\theta_{(2)}$: We draw the new parameters $\theta_{(2)}^{(t+1)}$ as follows: First, we draw the candidate parameter $\theta_{(2)}^{*(t)}$ from the proposal density $q\left(\theta_{(2)}^{(t)}, \theta_{(2)}^{*(t)}\right)$. Then, accept $\theta_{(2)}^{*(t)}$, i.e. set $\theta_{(2)}^{(t+1)} = \theta_{(2)}^{*(t)}$ with probability

$$\begin{aligned} & \lambda\left(\theta_{(1)}^{(t+1)}, \theta_{(2)}^{*(t)} | \Omega^{(t-1)}\right) \\ = \; & \min\left\{ \frac{\pi\left(\theta_{(1)}^{(t+1)}, \theta_{(2)}^{*(t)}\right)\left[\prod_{i=1}^{N} L_i\left(Y_{i,T} | \widetilde{\theta}_i^{(t+1)}, \theta_{(2)}^{*(t)}\right)\right] q\left(\theta_{(2)}^{*(t)}, \theta_{(2)}^{(t)}\right)}{\pi\left(\theta_{(1)}^{(t+1)}, \theta_{(2)}^{(t)}\right)\left[\prod_{i=1}^{N} L_i\left(Y_{i,T} | \widetilde{\theta}_i^{(t+1)}, \theta_{(2)}^{(t)}\right)\right] q\left(\theta_{(2)}^{(t)}, \theta_{(2)}^{*(t)}\right)}, 1 \right\} \end{aligned}$$

otherwise, reject $\theta_{(2)}^{*(t)}$, i.e. set $\theta_{(2)}^{(t+1)} = \theta_{(2)}^{(t)}$.

Bellman Equation Step: During each Metropolis-Hastings step, for each firm i we solve for the expected value function $\widehat{E}_{\epsilon'}^{(t)}\left[V\left(., \widetilde{\theta}_i, \theta_{(2)}\right) | \Omega^{(t-1)}\right]$. To do so for all $s \in S$, as before we follow equation 5. For use in future iterations, we simulate the value function by drawing $\epsilon^{(t)}$ to derive,

$$\mathcal{V}^{(t)}(s, a, \epsilon_a^{(t)}, \widetilde{\theta}_i, \theta_{(2)}, \Omega^{(t-1)}) = U(s, a, \epsilon_a^{(t)}, \widetilde{\theta}_i, \theta_{(2)}) + \beta \hat{E}{}_{\epsilon'}^{(t)}\left[V(s', \epsilon', \widetilde{\theta}_i, \theta_{(2)}) \mid \Omega^{(t-1)}\right],$$

$$V^{(t)}(s, \epsilon^{(t)}, \widetilde{\theta}_i, \theta_{(2)}, \Omega^{(t-1)}) = \max_{a \in A} \mathcal{V}^{(t)}(s, a, \epsilon_a^{(t)}, \widetilde{\theta}_i, \theta_{(2)}, \Omega^{(t-1)}).$$

The additional computational burden necessary to estimate the random coefficient model is the computation of the value function which has to be done separately for each firm i, because each firm has different random effects parameter vector. That is why in this case the adoption Bayesian DP algorithm results in large reduction in computational cost.

3.2 Continuous State Space

So far, we assumed a finite state space evolving stochastically. However, the Bayesian DP algorithm can also be applied in a straightforward manner to other settings of dynamic discrete choice models, with minor modifications. One example is the Random grid approximation of Rust (1997). There, given continuous state variable s, action a and parameter θ, the transition function from state vector s to the next period state vector $s\prime$ is defined to be $f(s'|a,s,\theta)$. Then, to estimate the model, the Dynamic Programming part of our algorithm can be modified as follows.

At iteration t, the value of choice a at parameter θ, state vector s, shock ϵ is defined to be as,

$$\mathcal{V}^{(t)}(s,a,\epsilon_a,\theta) = U(s,a,\epsilon_a,\theta) + \beta \hat{E}_{s',\epsilon'}\left[V(s',\epsilon',\theta)\right],$$

where s' is the next period state varible. $\hat{E}_{s',\epsilon'}\left[V(s',\epsilon',\theta)\right]$ is defined to be the approximation for the expected value function. The value function is defined to be as follows.

$$V^{(t)}(s,\epsilon,\theta) = \max_{a\in A} \mathcal{V}^{(t)}(s,a,\epsilon_a,\theta)$$

Conventionally, randomly generated state vector grid points are fixed throughout the solution/estimation algorithm. If we follow this procedure, and let s_m, $m = 1, ..., M$ be the random grids that are generated before the start of the solution/estimation algorithm, then, given parameter θ, the expected value function approximation at iteration t of the DP solution algorithm using the Rust random grids method would be,

$$\sum_{m=1}^{M} E_\epsilon V^{(t)}(s_m,\epsilon,\theta)\frac{f\left(s_m|a,s,\theta\right)}{\sum_{l=1}^{M} f\left(s_l|a,s,\theta\right)}.$$

Hence, if we were to apply the Rust method in our solution/estimation algorithm, the Emax function (i.e., the expected value function) $\hat{E}_{s',\epsilon'}\left[V(s',\epsilon',\theta)\right]$ would be approximated as follows:

$$\begin{aligned}
&\hat{E}_{s',\epsilon'}\left[V(s',\epsilon',\theta)\right] \\
\equiv\ &\sum_{m=1}^{M}\left[\sum_{n=1}^{N(t)} V^{(t-n)}(s_m,\epsilon^{(t-n)},\theta^{(t-n)})\frac{K_h(\theta-\theta^{(t-n)})}{\sum_{k=1}^{N(t)} K_h(\theta-\theta^{(t-k)})}\right]\frac{f\left(s_m|a,s,\theta\right)}{\sum_{l=1}^{M} f\left(s_l|a,s,\theta\right)}.
\end{aligned}$$

Notice that in this definition of Emax approximation, the grid points remain fixed over all iterations. In contrast, in our Bayesian DP algorithm, random grids can be changed at each solution/estimation iteration. Let $s^{(t)}$ be the random grid point generated at iteration t. Here, $s^{(\tau)}$, $\tau = 1, 2, ...$ are drawn independently from a distribution. Furthermore, let $K_h(.)$ be the kernel function with bandwidth h. Then, the expected value function can be approximated as follows.

$$\begin{aligned}
&\hat{E}_{s',\epsilon'}\left[V(s',\epsilon',\theta)\right] \\
\equiv\ &\sum_{n=1}^{N(t)} V^{(t-n)}(s^{(t-n)},\epsilon^{(t-n)},\theta^{(t-n)})\frac{K_h(\theta-\theta^{(t-n)})f\left(s^{(t-n)}|a,s,\theta\right)}{\sum_{k=1}^{N(t)} K_h(\theta-\theta^{(t-k)})f\left(s^{(t-k)}|a,s,\theta\right)}
\end{aligned}$$

Notice that unlike Rust (1997), we do not need to fix the random grid points of the state vector throughout the entire estimation exercise. In fact, we could draw a different state vector for each solution/estimation iteration.

In Rust (1997), if the total number of random grids is M, then the number of computations required for each Dynamic Programming iteration is M. Hence, at iteration τ, the number of Dynamic Programming computations that is required is $M\tau$. If a single DP solution step requires τ DP iterations, and each Newton ML step requires K DP solution steps, then, to iterate Newton ML algorithm once, we need to compute a single DP iteration $M\tau K$ times.

In contrast, in our Bayesian DP algorithm, at iteration t we only need to draw one state vector $s^{(t)}$ (so that $M = 1$) and only compute the Bellman equation on that state vector. Further, we solve the DP problem only once (so that $\tau = 1$ and $K = 1$). Still, at iteration t, the number of random grid points is $N(t)$, which can be made arbitrarily large when we increase the number of iterations. In other words, in contrast to the Rust method, the accuracy of the Dynamic Programming computation in our algorithm automatically increases with iterations.

Another issue that arises in application of the Rust random grid method is that Rust (1997) assumes that the transition density function $f(s'|a, s, \theta)$ is not degenerate. That is, we cannot use the random grid algorithm if the transition from s to $s\prime$, given a, θ is deterministic. It is also well known that the random grid algorithm becomes inaccurate if the transition density has a small variance. In these cases, several versions of polynomial based expected value function (emax function) approximation have been used. Keane and Wolpin (1994) approximate the emax function using polynomials of deterministic part of the value functions for each choice and state space point. Imai and Keane (2004) use Chebychev polynomials of state variables. It is known that in some cases, global approximation using polynomials can be numerically unstable and exhibit "wiggling". Here, we propose a kernel based local interpolation approach to Emax function approximation. The main problem behind the local approximation has been the computational burden of having a large number of grid points. As pointed our earlier, in our solution/estimation algorithm, we can make the number of grid points arbitrarily large by increasing the total number of iterations, even though the number of grid points per iteration is one.

The next period state variable, s' is assumed to be a deterministic function of s, a, and θ. That is,

$$s' = s'(s, a, \theta).$$

Let $K_{h_s}(.)$ be the kernel function with bandwidth h_s for the state variable and $K_{h_\theta}(.)$ for the parameter vector θ. Then, $\hat{E}_{s',\epsilon'}[V(s', \epsilon', \theta)]$ is defined to be as follows.

$$\begin{aligned}&\hat{E}_{\epsilon'}[V(s', \epsilon', \theta)]\\ \equiv\ &\sum_{n=1}^{N(t)} V^{(t-n)}(s^{(t-n)}, \epsilon^{(t-n)}, \theta^{(t-n)}) \frac{K_{h_s}\left(s' - s^{(t-n)}\right) K_{h_\theta}(\theta - \theta^{(t-n)})}{\sum_{k=1}^{N(t)} K_{h_s}\left(s' - s^{(t-k)}\right) K_{h_\theta}(\theta - \theta^{(t-k)})}.\end{aligned}$$

4 Examples

We estimate a simple dynamic discrete choice model of entry and exit, with firms in competitive environment.[10] The firm is either an incumbent (I) or a potential entrant (O). If the incumbent firm chooses to stay, its per period return is,

$$R_{I,IN}(K_t, \epsilon_t, \theta) = \alpha K_t + \epsilon_{1t},$$

where K_t is the capital of the firm, $\epsilon_t = (\epsilon_{1t}, \epsilon_{2t})$ is a vector of random shocks, and θ is the vector of parameter values. If it chooses to exit, its per period return is,

$$R_{I,OUT}(K_t, \epsilon_t, \theta) = \epsilon_{2t}.$$

Similarly, if the potential entrant chooses to enter, its per period return is,

$$R_{O,IN}(K_t, \epsilon_t, \theta) = -\delta + \epsilon_{1t},$$

and if it decides to stay out, its per period return is,

$$R_{O,OUT}(K_t, \epsilon_t, \theta) = \epsilon_{2t}.$$

We assume the random component of the current period returns to be distributed i.i.d normal as follows.

$$\epsilon_{lt} \sim N(0, \sigma_l),\ \ l = 1, 2$$

The level of capital K_t evolves as follows. If the incumbent firm stays in, then,

$$\ln K_{t+1} = b_1 + b_2 \ln K_t + u_{t+1},$$

where,

$$u_{t+1} \sim N(0, \sigma_u),$$

and if the potential entrant enters,

$$\ln K_{t+1} = b_e + u_{t+1}.$$

Now, consider a firm who is an incumbent at the beginning of period t. Let $V_I(K_t, \epsilon_t, \theta)$ be the value function of the incumbent with capital stock K_t, and $V_O(K_t, \epsilon_t, \theta)$ be the value function of the potential entrant, who has capital stock 0. The Bellman equation for the optimal choice of the incumbent is:

$$V_I(K_t, \epsilon_t, \theta) = Max\{V_{I,IN}(K_t, \epsilon_t, \theta), V_{I,OUT}(K_t, \epsilon_t, \theta)\}.$$

where,

$$V_{I,IN}(K_t, \epsilon_t, \theta) = R_{I,IN}(K_t, \epsilon_{1t}, \theta) + \beta E_{t+1} V_I(K_{t+1}(K_t, u_{t+1}, \theta), \epsilon_{t+1}, \theta)$$

[10] For an estimation exercise based on the model, see Roberts and Tybout (1997).

is the value of staying in during period t. Similarly,

$$V_{I,OUT}(K_t, \epsilon_t, \theta) = R_{I,OUT}(K_t, \epsilon_{2t}, \theta) + \beta E_{t+1} V_O(0, \epsilon_{t+1}, \theta)$$

is the value of exiting during period t . The Bellman equation for the optimal choice of the potential entrant is:

$$V_O(0, \epsilon_t, \theta) = Max\{V_{O,IN}(0, \epsilon_t, \theta), V_{O,OUT}(0, \epsilon_t, \theta)\}.$$

where,

$$V_{O,IN}(0, \epsilon_t, \theta) = R_{O,IN}(0, \epsilon_{1t}, \theta) + \beta E_{t+1} V_I(K_{t+1}(0, u_{t+1}, \theta), \epsilon_{t+1}, \theta),$$

is the value of entering during period t and,

$$V_{O,OUT}(0, \epsilon_t, \theta) = R_{O,OUT}(0, \epsilon_{2t}, \theta) + \beta E_{t+1} V_O(0, \epsilon_{t+1}, \theta),$$

is the value of staying out during period t. Notice that the capital stock of a potential entrant is always 0.

The parameter vector θ of the model is $(\delta, \alpha, \beta, \sigma_1, \sigma_2, \sigma_u, b_1, b_2, b_e)$.The state variables are the capital stock K, and the status of the firm, $\Gamma \in \{I, O\}$, that is, whether the firm is an incumbent or a potential entrant. Notice that capital stock is a continuous state variable with random transition, in contrast to the theoretical framework where the state space was assumed to be finite and the transition function deterministic.

We assume that for each firm, we only observe the capital stock, profit of the firm that stays in and the entry/exit status over T periods. That is, we know,

$$\{K^d_{i,t}, \pi^d_{i,t}, \Gamma^d_{i,t}\}_{i=1,N}^{t=1,T}$$

where,

$$\pi^d_{i,t} = \alpha K^d_{i,t} + \varepsilon_{1t},$$

if the firm stays in.

We assume the prior distribution of all parameters to be uninformative. That is, we set $\pi(\theta) = 1$. Below, we explain the estimation steps in detail.

Bellman Equation Step

In this step, we derive the value function, i.e., $V_\Gamma^{(s)}(K, \epsilon^{(s)}, \theta^{(s)})$ for iteration s.

1) Suppose we have already calculated the approximation for the expected value function, where the expectation is over the choice shock ϵ, which is,

$$\widehat{E}_\epsilon^{(s)} V_\Gamma(K'(K, u^{(s)}, \theta^{(s)}), \epsilon, \theta^{(s)}).$$

To further integrate the value function over the capital shock u, we can either use the random grid integration method of Rust (1997) which uses a fixed grid or let the grid size increase over the iterations. Here, we use the Rust method. That is, given that we have drawn M i.i.d. capital stock grids K_m, $m = 1, .., M$ from a given distribution, we take the weighted average as follows,

$$\widehat{E}^{(s)}\left[V_\Gamma(K'(K,u,\theta^{(s)}),\epsilon,\theta^{(s)})\right] = \sum_{m=1}^{M} \widehat{E}_\epsilon^{(s)}\left[V_\Gamma^{(s)}(K_m,\epsilon,\theta^{(s)})\right] \frac{f(K_m|K,\theta^{(s)})}{\sum_{m=1}^{M} f(K_m|K,\theta^{(s)})}.$$

where $f(K_m|K,\theta^{(s)})$ is the capital transition function from K to K_m. In this example, the random grids remain fixed throughout the estimation. Note that if the firm exits or stays out, $K' = 0$. Hence, the expected value function becomes $\widehat{E}_\epsilon^{(s)}\left[V_O(0,\epsilon,\theta^{(s)})\right]$.

2) We draw $\epsilon^{(s)} = (\epsilon_1^{(s)}, \epsilon_2^{(s)})$.

3) Given $\epsilon^{(s)}$ and $\widehat{E}^{(s)}V_\Gamma(K,\epsilon,\theta^{(s)})$, we solve the Bellman equation, that is, we solve the decision of the incumbent (whether to stay or exit) or of the entrant (whether to enter or stay out) and derive the value function corresponding to the optimal decisions:

$$\begin{aligned} V_\Gamma^{(s)}(K,\epsilon^{(s)},\theta^{(s)}) &= Max\{R_{\Gamma,IN}(K,\epsilon_1^{(s)},\theta^{(s)}) + \beta\widehat{E}^{(s)}\left[V_I(K'(K,u,\theta^{(s)}),\epsilon,\theta^{(s)})\right], \\ &\quad R_{\Gamma,OUT}(K,\epsilon_2^{(s)},\theta^{(s)}) + \beta\widehat{E}^{(s)}\left[V_O(K'(K,u,\theta^{(s)}),\epsilon,\theta^{(s)})\right]\} \end{aligned}$$

Modified Metropolis-Hastings Step

We draw the new parameter vector $\theta^{(s+1)}$ from the posterior distribution. We denote the vector $\mathbf{I}_i$ as follows:

$$\mathbf{I}_i = [I_{i,1}^d(IN), ..., \ I_{i,t}^d(IN), ..., I_{i,T}^d(IN)]$$

where $I_{i,t}^d(IN) = 1$ if the firm either enters or decides to stay in, and 0 otherwise. Similarly, we denote $\mathbf{K}_i$, $\boldsymbol{\pi}_i$ to be the vector of $K_{i,t}^d$ and $\pi_{i,t}^d$. The likelihood increment for firm i at time t is

$$\begin{aligned}
& L_i\left(\mathbf{I}_i, \mathbf{K}_i, \boldsymbol{\pi}_i | \theta\right) \\
= \; & \Pr\left[\epsilon_{2t} \leq \pi^d_{it} + \beta\left\{\widehat{E}^{(s)}\left[V_{IN}(K'(K,u,\theta^{(s)}),\epsilon,\theta^{(s)})\right] - \widehat{E}^{(s)}\left[V_O(0,\epsilon,\theta^{(s)})\right]\right\}\right] \\
& \phi\left(\frac{\pi^d_{it} - \alpha K^d_{it}}{\sigma_{\epsilon_1}}\right)\frac{1}{K^d_{it+1}}\phi\left(\frac{\ln K^d_{it+1} - b_1 - b_2 \ln K^d_{it}}{\sigma_u}\right) I^d_{i,t}(IN) I^d_{i,t+1}(IN) \\
& + \Pr\left[\epsilon_{2t} - \epsilon_{1t} > \alpha K^d_{it} + \beta\left\{\widehat{E}^{(s)}\left[V_{IN}(K'(K,u,\theta^{(s)}),\epsilon,\theta^{(s)})\right] - \widehat{E}^{(s)}\left[V_O(0,\epsilon,\theta^{(s)})\right]\right\}\right] \\
& I^d_{i,t}(IN)\left(1 - I^d_{i,t+1}(IN)\right) \\
& + \Pr\left[\epsilon_{2t} - \epsilon_{1t} \leq -\delta + \beta\left\{\widehat{E}^{(s)}\left[V_{IN}(K'(0,u,\theta^{(s)}),\epsilon,\theta^{(s)})\right] - \widehat{E}^{(s)}\left[V_O(0,\epsilon,\theta^{(s)})\right]\right\}\right] \\
& \frac{1}{K^d_{it+1}}\phi\left(\frac{\ln K^d_{it+1} - b_e}{\sigma_u}\right)\left(1 - I^d_{i,t}(IN)\right) I^d_{i,t+1}(IN) \\
& + \Pr\left[\epsilon_{2t} - \epsilon_{1t} > -\delta + \beta\left\{\widehat{E}^{(s)}\left[V_{IN}(K'(0,u,\theta^{(s)}),\epsilon,\theta^{(s)})\right] - \widehat{E}^{(s)}\left[V_O(0,\epsilon,\theta^{(s)})\right]\right\}\right] \\
& \left(1 - I^d_{i,t}(IN)\right)\left(1 - I^d_{i,t+1}(IN)\right)
\end{aligned}$$

We employ the modified Metropolis-Hastings algorithm, where at iteration s the proposal density $q\left(\theta^{(s)}, \theta^*\right)$ is

$$\begin{aligned}
\delta^* &\sim N\left(\delta^{(s)}, \sigma^2_\delta\right) \\
\alpha^* &\sim N\left(\alpha^{(s)}, \sigma^2_\alpha\right) \\
\ln \sigma^*_{\epsilon_1} &\sim N\left(\ln \sigma^{(s)}_{\epsilon_1}, \sigma^2_{\ln \sigma_{\epsilon_1}}\right) \\
\ln \sigma^*_{\epsilon_2} &\sim N\left(\ln \sigma^{(s)}_{\epsilon_2}, \sigma^2_{\ln \sigma_{\epsilon_2}}\right) \\
b^*_1 &\sim N\left(b^{(s)}_1, \sigma^2_{b_1}\right) \\
b^*_2 &\sim N\left(b^{(s)}_2, \sigma^2_{b_2}\right) \\
b^*_e &\sim N\left(b^{(s)}_e, \sigma^2_{b_e}\right) \\
\ln \sigma^*_u &\sim N\left(\ln \sigma^{(s)}_u, \sigma^2_{\ln \sigma_u}\right)
\end{aligned}$$

That is, we adopt the modified random walk Metropolis Hastings algorithm. The algorithm sets $\theta^{(s+1)}_1 = \theta^*_1$ with probability

$$\lambda\left(\theta^{(s)}, \theta^*\right) = \min\left\{\frac{\prod_{i=1}^{N} L_i\left(\mathbf{I}_i, \mathbf{K}_i, \boldsymbol{\pi}_i | \theta^*\right)}{\prod_{i=1}^{N} L_i\left(\mathbf{I}_i, \mathbf{K}_i, \boldsymbol{\pi}_i | \theta^{(s)}\right)}, 1\right\}$$

Expected Value Function Iteration Step

Next, we update the expected value function for iteration $s+1$. First, we derive $E_{\epsilon}^{(s+1)} V_{\Gamma}(K, \epsilon, \theta^{(s+1)})$.

$$\begin{aligned} & E_{\epsilon}^{(s+1)}\left[V_{\Gamma}(K, \epsilon, \theta^{(s+1)})\right] \\ = & \frac{\sum_{l=Max\{s-N(s),1\}}^{s}\left[\frac{1}{M}\sum_{m=1}^{M} V_{\Gamma}^{(l)}(K, \epsilon_m^{(l)}, \theta^{(l)})\right] K_h(\theta^{(s+1)}-\theta^{(l)}) I\left[K_h(\theta^{(s+1)}-\theta^{(l)}) > \overline{K}_L\right]}{\sum_{l=Max\{s-N(s),1\}}^{s} K_h(\theta^{(s+1)}-\theta^{(l)}) I\left[K_h(\theta^{(s+1)}-\theta^{(l)}) > \overline{K}_L\right]}, \end{aligned}$$

where $K()$ is the kernel function. We adopt the following Gaussian kernel:

$$K_h(\theta^{(s)}-\theta^{(l)}) = (2\pi)^{-L/2}\prod_{j=1}^{J} h_j^{-1}\exp[-\frac{1}{2}(\frac{\theta_j^{(s)}-\theta_j^{(l)}}{h_j})^2].$$

$\overline{K}_L$ is the L^{th} largest value of the $N(s)$ kernel values $\left\{K_h(\theta^{(s)}-\theta^{(l)})\right\}_{l=Max\{s-N(s)\}}^{s}$. The expected value function is updated by taking the weighted average over the L value functions of past $N(s)$ iterations where the parameter vector $\theta^{(l)}$ was closest to $\theta^{(s+1)}$.

Then, if the firm enters or stays in, the expected value function is as follows.

$$\begin{aligned} & \widehat{E}^{(s+1)}\left[V_I(K'(K, u, \theta^{(s+1)}), \epsilon, \theta^{(s+1)})\right] \\ = & \widehat{E}_{\epsilon,K}^{(s+1)}\left[V_I(K'(K, u, \theta^{(s+1)}), \epsilon, \theta^{(s+1)})\right] \\ = & \sum_{m=1}^{M}\widehat{E}_{\epsilon}^{(s+1)}\left[V_I(K_m, \epsilon, \theta^{(s+1)})\right]\frac{f(K_m|K, \theta^{(s+1)})}{\sum_{m=1}^{M} f(K_m|K, \theta^{(s+1)})}. \end{aligned}$$

As discussed before, in principle, only one simulation of ϵ is needed during each solution/estimation iteration. But that requires the number of past iterations for averaging, i.e. $N(s)$ to be large, which adds to computational burden. Instead, in our example, we draw ϵ ten times and take an average. Hence, when we derive the expected value function, instead of averaging past value functions, we average over past average value functions, i.e., $\frac{1}{M}\sum_{m=1}^{M} V_{\Gamma}(K_m, \epsilon_m^{(j)}, \theta^{(j)})$, where $M=10$. This obviously increases the accuracy per iteration, and reduces the need to have a large $N(s)$. That is partly why in the examples below, to have $N(s)$ increase up to 2000 turns out to be sufficient for good estimation performance. L, the number of nearest parameter values is set to be 1000. Notice that if the firm stays out or exits, then its future capital stock is zero. Therefore, no averaging over capital grid points is required to derive the expected value function, i.e., the emax function is simply $E_{\epsilon}^{(s+1)}\left[V_O(0, \epsilon, \theta^{(s+1)})\right]$.

In the next section, we present the results of several Monte Carlo studies we conducted using our Bayesian DP algorithm. The first experiment is the basic model using the Rust random grid method. The second experiment incorporates observed and unobserved heterogeneity, and finally, we conduct an experiment in which capital stock evolves deterministically.

5 Simulation and Estimation

Denote the true values of θ by θ^0, i.e. $\theta^0 = (\delta^0, \sigma^0_{\epsilon_1}, \sigma^0_{\epsilon_2}, \sigma^0_u, \alpha^0, b^0_1, b^0_2, b^0_e, \beta^0)$. We set them as follows: $\delta^0 = 0.4$, $\sigma^0_{\epsilon_1} = 0.3$, $\sigma^0_{\epsilon_2} = 0.3$, $\sigma^0_u = 0.4$, $\alpha^0 = 0.1$, $b^0_1 = 0.0$, $b^0_2 = 0.4$, $b^0_e = 5.0$, $\beta^0 = 0.98$.

We first solve the DP problem numerically using conventional numerical methods. Next, we generate artificial data based on this DP solution. All estimation exercises are done on a 2.8 GHz Pentium 4 Linux workstation. Below, we briefly explain how we solved the DP problem to generate the data for the basic model. For the other two experiments, the data generation step is basically similar with only minor changes. Notice that for data generation, we only need to solve the DP problem once, that is, for a fixed set of parameters. Hence, we took our time and made sure that the DP solution is accurate.

We first set the M_K capital grid points to be equally spaced between 0 and $\overline{K}$, which we set to be 5.0. Assume that we already know the expected value function of the s^{th} DP iteration for all capital grid points.

$$E^{(s)}_{\epsilon} V_{\Gamma}(K_m, \epsilon, \theta^0),\ \Gamma \in \{I, O\},\ m = 1, 2, ..., M_K.$$

Here, K_m $(m = 1, ..., M_K)$ are grid points.

The following steps are taken to generate the expected value function for the $(s+1)^{th}$ iteration.

Step 1 Given capital stock K, we derive,

$$E^{(s)} V_{\Gamma}(K'(K, u, \theta^0), \epsilon^{(s)}, \theta^*) = \sum_{m=1}^{M_K} E^{(s)}_{\epsilon} V_{\Gamma}(K_m, \epsilon^{(s)}, \theta^0) \frac{f(K_m|K, \theta^0)}{\sum_{l=1}^{M_K} f(K_l|K, \theta^0)}.$$

Here $f(K_m|K, \theta^0)$ is the transition probability from K to K_m.

Step 2 We draw the random shocks ϵ_l. Then, for a given capital stock K, calculate

$$\begin{aligned} V^{(s+1)}_{\Gamma}(K, \epsilon_l, \theta^0) &= Max\left\{R_{\Gamma,IN}(K, \epsilon_{1l}, \theta^0) + \beta E^{(s)} V_I(K', \epsilon, \theta^0),\right. \\ &\quad \left. R_{\Gamma,OUT}(K, \epsilon_{2l}, \theta^0) + \beta E^{(s)} V_O(0, \epsilon, \theta^0)\right\} \end{aligned}$$

Step 3 We repeat Step 2, M_ϵ times and take an average to derive the approximated expected value function for the next iteration.

$$E^{(s+1)}_{\epsilon} V_{\Gamma}(K, \epsilon, \theta^0) = \frac{1}{M_\epsilon} \sum_{l=1}^{M_\epsilon} V^{(s+1)}_{\Gamma}(K, \epsilon_l, \theta^0).$$

The above steps are taken for all possible capital grid points, $K = K_1, .., K_{M_K}$. In our simulation exercise, we set the simulation size M_ϵ to be 1000. The total number of capital grid points is set to be $M_K = 200$.

Step 4 Repeat Step 1 to Step 3 until the Emax function converges. That is, for a small δ (in our case, $\delta = 0.00001$),

$$Max_{m=1,..,M_K}\{E_\epsilon^{(s+1)}V_\Gamma(K_m, \epsilon, \theta^0), E_\epsilon^{(s)}V_\Gamma(K_m, \epsilon, \theta^0)\} < \delta.$$

We simulate artificial data of capital stock, profit and entry/exit choice sequences $\{K_{i,t}^d, \pi_{i,t}^d, I_{i,t}^d\}_{i=1,t=1}^{N,T}$ using the expected value functions derived above. We then estimate the model using the simulated data with our Bayesian DP routine. We do not estimate the discount factor β. Instead, we set it at the true value $\beta^0 = 0.98$. We simulated the sample size plus $2,000$ artificial data, where the first $2,000$ simulations were discarded.

5.1 Experiment 1: Basic Model

All the priors are set to be uninformative. We set the initial guesses of the parameters to be the true parameter values given by θ^0, and the initial guess of the expected value function to be 0. We used the same 200 grid points in each iteration as used in generating the data. The pseudo-MCMC sampler was generated $10,000$ times. The posterior mean and standard errors from the $(5,001)^{th}$ iteration up to $(10,000)^{th}$ iteration are shown in Panel 1 of Table 1. In Panel 2 we present the estimation result of the conventional Bayesian MCMC estimation, where during each estimation step the Dynamic Programming model is solved in full. The posterior means and posterior standard deviations presented are the sample average of 10 simulation/estimation exercises, where for each exercise a different seed was chosen. We also report the sample standard errors of the posterior means and posterior standard deviations of the 10 simulation/estimation exercises. As we can see, the sample averages of both the Bayesian DP posterior means and those of the Bayesian estimation with full solution are very close to the true parameter values. Furthermore, as we can see from the sample averages and the sample standard errors of the 10 simulation/estimation exercises, both the posterior means and the standard errors of the Bayesian DP estimation are very close to those of the conventional Bayesian MCMC estimates. The sample average of the posterior mean of the entry cost parameter estimate by the Bayesian DP algorithm for the sample size of 2000 seems to be relatively farther away from the true value (0.4), compared to the ones of other estimation exercises. This could reflect the approximation error of the expected value function in the Bayesian DP algorithm. That is, with smaller sample size and larger variance of the parameters, more past iterations may be required to accurately approximate the expected value functions. Notice also that for the Bayesian DP estimation, as the sample size decreases from $10,000$ to $2,000$, the CPU time decreases from 18 minutes 21. seconds to 4 minutes 55 seconds, a 3.5 to 1 decrease. On the other hand, for the full solution based Bayesian estimation the CPU time decreases from 29 minutes 44 seconds to 15 minutes 44 seconds, only a 50% decrease. That is, as the sample size decreases, relatively more CPU time is spent on the solution of the model than on computing the likelihood. Hence, the computational advantage of the Bayesian DP algorithm becomes more apparent.

In Panel 3 we also report the simulation/estimation exercises of the full solution based ML estimation[11]. The standard errors are based on the inversion of the information matrix. To compute the Information matrix, we adopt the BHHH algorithm, i,e. we approximate it by the inner product of the gradient vector of the likelihood increments. The parameter estimates are again very close to the true values and close to those of the Bayesian posterior means. However, the standard errors, are quite different from the standard deviations of the Bayesian estimates. For example, the standard error for the ML estimate for the entry cost is 0.0185 if the sample size is $10,000$, 0.0255 and 0.0417 if the sample size is $5,000$, and $2,000$, respectively. On the other hand, the corresponding standard deviations of the Bayesian DP estimates for the entry cost is 0.0136, 0.0195, and 0.0295, which are close to those of the full solution based Bayesian estimates. This reflects the inaccuracies of the computation of the Information matrix, which is based on a numerical first derivative of the likelihood increments. The CPU time required for the ML estimation is much smaller than the Bayesian estimates. For example, the CPU time for the ML estimation with sample size of $10,000$ is 17 seconds, whereas the Bayesian DP estimation requires about 18 minutes. That is, for the estimation of a simple dynamic structural model, the standard ML estimation is computationally superior to the Bayesian estimation. The computational time could become comparable if the standard errors were to be derived by bootstrap.

To check robustness of the Bayesian DP algorithm, we also ran a simulation/estimation exercise where the starting parameter value was set to be half of the true values. As we can see from the results reported in Panel 3, the posterior means and the standard deviations are almost the same as those of Panel 1 where the initial parameter values were set to be the true ones. These results confirm the theorems on convergence in Section 1 stating that the estimation algorithm is not sensitive to the initial values.

Table 1: Posterior Means and Standard Errors (standard errors are in parenthesis)

[11] The initial parameter values were set to be the true values.

Part 1. Bayesian dynamic programming estimation.				
Sample mean of posterior means and std. deviations of				
10 simulation/estimation exercises				
Parameter	estimate	estimate	estimate	true
δ	0.3874 (0.0295)	0.3958 (0.0195)	0.3983 (0.0136)	0.4
α	0.09828 (0.00629)	0.09936 (0.00404)	0.09938 (0.00291)	0.1
σ_{ϵ_1}	0.2989 (0.00734)	0.3018 (0.00482)	0.3012 (0.00354)	0.3
σ_{ϵ_2}	0.2963 (0.0224)	0.2933 (0.0160)	0.2950 (0.0120)	0.3
b_1	0.000925 (0.0117)	-0.000318 (0.00751)	-0.00179 (0.00537)	0.0
b_2	0.4027 (0.0223)	0.4037 (0.0151)	0.4039 (0.0106)	0.4
b_e	0.5019 (0.0246)	0.5121 (0.0156)	0.5073 (0.0111)	0.5
σ_u	0.3939 (0.00715)	0.3971 (0.00466)	0.3986 (0.00327)	0.4
sample	$2,000$	$5,000$	$10,000$	
CPU time	4 min. 55 sec.	9 min.47 sec.	18 min. 21 sec.	
sample std. errors of posterior means and posterior std. dev. of 10 sim./est. exercises				
δ	0.0274 (0.00480)	0.0226 (0.00259)	0.0162 (0.00124)	
α	0.00505 (0.000404)	0.00422 (0.000250)	0.00245 (0.000174)	
σ_{ϵ_1}	0.00998 (0.000933)	0.00557 (0.000465)	0.00490 (0.000276)	
σ_{ϵ_2}	0.0353 (0.0104)	0.0257 (0.00544)	0.0200 (0.00324)	
b_1	0.0134 (0.000613)	0.00919 (0.000347)	0.00562 (0.000229)	
b_2	0.0255 (0.00175)	0.0123 (0.00100)	0.00554 (0.000928)	
b_e	0.0146 (0.00446)	0.0165 (0.00137)	0.0123 (0.000457)	
σ_u	0.00878 (0.000599)	0.00496 (0.000298)	0.00378 ($8.7E-5$)	

Part 2. Bayesian estimation based on the full solution of the model.				
sample mean of posterior means and std. dev. of 10 simulation/estimation exercises				
parameter	estimate	estimate	estimate	true
δ	0.4020 (0.0278)	0.3992 (0.0184)	0.4025 (0.0135)	0.4
α	0.09939 (0.00571)	0.09951 (0.00378)	0.09967 (0.00270)	0.1
σ_{ϵ_1}	0.2981 (0.00731)	0.3015 (0.00492)	0.3013 (0.00341)	0.3
σ_{ϵ_2}	0.3069 (0.0239)	0.2968 (0.0177)	0.2973 (0.0133)	0.3
b_1	0.000538 (0.0118)	-0.000585 (0.00780)	-0.00214 (0.00544)	0.0
b_2	0.4040 (0.0234)	0.4040 (0.0151)	0.4047 (0.0105)	0.4
b_e	0.5006 (0.0241)	0.5121 (0.0157)	0.5075 (0.0109)	0.5
σ_u	0.3931 (0.00736)	0.3969 (0.00458)	0.3985 (0.00333)	0.4
sample size	$2,000$	$5,000$	$10,000$	
CPU time	15 min. 44 sec.	21 min.31 sec.	29 min. 44 sec.	
sample std. errors of posterior means and posterior std. dev. of 10 sim./est. exercises				
δ	0.0280 (0.00604)	0.0247 (0.00276)	0.0132 (0.00230)	
α	0.00457 (0.000516)	0.00453 (0.000299)	0.00199 (0.000228)	
σ_{ϵ_1}	0.00969 (0.000722)	0.00460 (0.000349)	0.00494 (0.000371)	
σ_{ϵ_2}	0.0325 (0.0113)	0.0306 (0.00655)	0.0176 (0.00459)	
b_1	0.0124 (0.000709)	0.00914 (0.000597)	0.00563 (0.000211)	
b_2	0.0287 (0.00247)	0.0140 (0.00108)	0.00616 (0.00109)	
b_e	0.0154 (0.00203)	0.0177 (0.00106)	0.0126 (0.000473)	
σ_u	0.00863 (0.000630)	0.00537 (0.000411)	0.00399 (0.000233)	

Panel 3. ML estimation based on the full solution of the model.				
sample mean of posterior means and std. dev. of 10 simulation/estimation exercises				
parameter	estimate	estimate	estimate	true
δ	0.3872 (0.0417)	0.3967 (0.0255)	0.4002 (0.0185)	0.4
α	0.09801 (0.00690)	0.09944 (0.00438)	0.09978 (0.00314)	0.1
σ_{ϵ_1}	0.2993 (0.00912)	0.3016 (0.00561)	0.3014 (0.00399)	0.3
σ_{ϵ_2}	0.2882 (0.0448)	0.2929 (0.0264)	0.2929 (0.0185)	0.3
b_1	0.00108 (0.00889)	0.0001493 (0.00613)	−0.001721 (0.00383)	0.0
b_2	0.4039 (0.0242)	0.4039 (0.0152)	0.4044 (0.0107)	0.4
b_e	0.5006 (0.0246)	0.5119 (0.0159)	0.5076 (0.0113)	0.5
σ_u	0.3929 (0.00736)	0.3968 (0.00469)	0.3983 (0.00332)	0.4
sample size	$2,000$	$5,000$	$10,000$	
CPU time	4 sec.	6 sec.	17 sec.	
sample std. errors of posterior means and posterior std. dev. of 10 sim./est. exercises				
δ	0.0220 (0.00840)	0.0159 (0.00291)	0.0135 (0.00207)	
α	0.00522 (0.000301)	0.00392 (0.000118)	0.00201 (0.0000653)	
σ_{ϵ_1}	0.00814 (0.000535)	0.00482 (0.000221)	0.00465 (0.0000962)	
σ_{ϵ_2}	0.0309 (0.00829)	0.0206 (0.00294)	0.0147 (0.00196)	
b_1	0.0125 (0.00341)	0.00868 (0.00115)	0.00532 (0.00145)	
b_2	0.0264 (0.000695)	0.0132 (0.000347)	0.00562 (0.000231)	
b_e	0.0119 (0.00116)	0.0166 (0.000401)	0.0120 (0.000286)	
σ_u	0.00909 (0.000224)	0.00538 (0.000125)	0.00387 (0.0000639)	

Panel 4: Bayesian DP with starting value: $0.5\theta^*$		
sample mean of posterior means and std. errors		
parameter	estimate	true
δ	0.3972 (0.0132)	0.4
α	0.09949 (0.00286)	0.1
σ_{ϵ_1}	0.3011 (0.00348)	0.3
σ_{ϵ_2}	0.2949 (0.0118)	0.3
b_1	-0.00168 (0.00537)	0.0
b_2	0.4039 (0.0105)	0.4
b_e	0.5077 (0.0114)	0.5
σ_u	0.3986 (0.00325)	0.4
sample size	10,000	
CPU time	18 min. 20sec.	
sample std. errors of posterior means and posterior std. errors		
δ	0.0165 (0.00139)	
α	0.00239 (0.000152)	
σ_{ϵ_1}	0.00524 (0.000288)	
σ_{ϵ_2}	0.0206 (0.00306)	
b_1	0.00573 (0.000231)	
b_2	0.00522 (0.000888)	
b_e	0.0126 (0.000157)	
σ_u	0.00380	

5.2 Experiment 2: Random Effects

We now report estimation results of a model that includes observed and unobserved heterogeneities. We assume that the profit coefficient for each firm i, α_i is distributed normally with mean $\mu_\alpha = 0.2$ and standard error $\sigma_\alpha = 0.1$. The transition equation for capital is,

$$\ln K_{i,t+1} = b_1 X_i^d + b_2 \ln K_{i,t} + u_{i,t+1},$$

where X_i^d is a firm characteristics observable to the econometrician. In our simulation sample, we simulate X_i^d from $N(0.0, 1.0)$.

Notice that if we use the conventional simulated ML method to estimate the model, for each firm i we need to draw α_i many times, say M_α times, and for each draw, we need to solve the dynamic programming problem with the constant coefficient for capital transition equation being $b_1 X_i^d$. If the number of firms in the data is N_d, then for a single simulated likelihood evaluation, we need to solve the DP problem $N_d M_\alpha$ times. This process is computationally demanding and most researchers use only a finite number of types, typically less than 10, as an approximation of the observed heterogeneity and the random effect. The only exceptions are economists who have access to supercomputers or large PC clusters. Since in our Bayesian DP estimation exercise, the computational burden of estimating the dynamic model is similar to that of a static model, we can easily accomodate random effects estimation.

As we discussed earlier, in contrast to the solution/estimation algorithn of the basic model, we solve the one step Bellman equation for each firm i separately. Let $\theta_{-\alpha}$ be the parameter vector except for the random effects term α_i. Then, for given K, $\widehat{E}_{\epsilon}^{(s)} V_{\Gamma}(K, \epsilon, \theta_{-\alpha}^{(s)}, \alpha_i^{(s)})$ is derived as follows.

$$
\begin{aligned}
& \widehat{E}_{\epsilon}^{(s)} V_{\Gamma}(K, \epsilon, \theta_{-\alpha}^{(s)}, \alpha_i^{(s)}) \\
= & \frac{\sum_{j=Max\{s-1-N(s-1),1\}}^{s-1} \left[\frac{1}{M_{\epsilon}} \sum_{l=1}^{M_{\epsilon}} V_{\Gamma}^{(j)}(K, \epsilon_l^{(j)}, \theta^{(j)}) \right] K_h(\theta_{-\alpha}^{(s)} - \theta_{-\alpha}^{(j)}) K_h(\alpha_i^{(s)} - \alpha_i^{(j)})}{\sum_{j=Max\{s-1-N(s-1),1\}}^{s-1} K_h(\theta_{-\alpha}^{(s)} - \theta_{-\alpha}^{(j)}) K_h(\alpha_i^{(s)} - \alpha_i^{(j)})}.
\end{aligned}
$$

The remaining step to derive the expected value function $\widehat{E}^{(s)} \left[V_{\Gamma}(K'(K, u, \theta^{(s)}), \epsilon, \theta^{(s)}) \right]$ is the same as in Experiment 1.

As pointed out by Heckman (1981) and others, the missing initial state vector (that is, the initial status of the firm and initial capital) is likely to be correlated with the unobserved heterogeneity α_i, which would result in bias of the parameter estimates. To deal with this problem, for each firm i, given parameters $(\theta_{-\alpha}, \alpha_i)$, we simulate the model for 100 initial periods to derive the initial capital and the initial status of the firm. Then, we proceed to construct the likelihood increment for firm i.

We set $N(s)$ to go up to 1000 iterations. The one-step Bellman equation is the part where we have an increase in computational burden. But it turns out that the additional burden is far lighter than that of computing the DP problem for each firm i M_{α} times to integrate out the random effects α_i, as would be done in the Simulated ML estimation strategy.

We set the sample size to be 100 firms for 100 periods, and the Bayesian DP iteration was conducted $10,000$ times. Column 2 of Table 2 reports the posterior mean and standard deviations from the $5,001^{th}$ iteration up to $10,000^{th}$ iteration. We also report in column 3 the result of the simulation/estimation exercise of the Bayesian MCMC algorithm where during each estimation iteration the DP problem is solved in full. When we solve for the DP problem, we set M_{ϵ}, the number of simulations for ϵ to be 100, instead of $1,000$. In column 4, we show the parameter estimates of the Simulated Maximum Likelihood estimates, which is based on the full solution of the model. To construct the simulated likelihood, for each firm, we simulated α_i one hundred times (i.e. $M_{\alpha} = 100$). We solved the DP problem using Monte-Carlo integration to integrate over the choice shock ϵ. We set the simulation size for ϵ to be 100[12]. If we were to set the simulation size of ϵ to be $1,000$ as before, then the CPU time required for a single likelihood calculation would take us about 23 minutes and 40 seconds. Since we take numerical derivatives over 9 parameters to derive the gradient of the likelihood, a single Newton iteration would take about

[12] For the ML algorithm, we used the Newton-Raphson routine. Since we took numerical derivatives, in addition to the likelihood evaluation under the original parameter θ, we calculated the likelihood for the 9 parameter perturbations $\theta + \Delta\theta_i$, $i = 1, ..., 9$. We stopped running the program when either the absolute values of all the gradiants were less than 0.01 or the step size became less than $1.0D - 5$.

4 hours and 20 minutes, which, as we will see later, is about the same CPU time required for the entire Bayesian DP algorithm.

As we can see both the posterior means of the Bayesian DP estimates and those of the Full solution based Bayesian estimates are very close to the true values. Furthermore, the posterior means and the standard errors of the two estimators are very close to each other as well. On the other hand, we see a fairly large bias in the parameter estimates by Simulated ML. The entry cost parameter δ, the mean of profit coefficient μ_α and its standard error σ_α and the standard error of the choice shock σ_{ϵ_2} are all downwardly biased, and except for σ_α the magnitude of the bias is larger than the standard error. The downward bias seems to be especially large for μ_α, which leads us to conclude that the simulation size of $M_\alpha = 100$ is not enough to integrate out the unobserved heterogeneity sufficiently accurately. The CPU time required for the Bayesian DP algorithm is about 4 hours, whereas for the Full solution based Bayesian MCMC estimation we needed about 31 hours, and for the full solution based ML estimation, 21 hours. That is, the Bayesian DP is about 8 times as fast as the Full solution based Bayesian MCMC algorithm and about 5 times as fast as the Simulated ML algorithm. We also tried to reduce the computational time for the full solution based ML algorithm by reducing the number of draws for α_i from 100 to 20. Then, the CPU time reduces to 8 hours and 43 minutes, which is about twice as much time required for the Bayesian DP algorithm. However, the average of the 10 ML estimates of α is 0.145, which is even smaller than 0.170 , which is the result for the estimation with 100 α_i draws. The true value is 0.2. We can see that the downward bias is larger than before. The sample standard error of μ_α and σ_α over 10 simulation/estimation experiments are much larger as well. If we were to try to reduce the bias by increasing the simulation size of unobserved heterogeneity from $M_\alpha = 100$ to, say $M_\alpha = 1,000$, then the CPU time would be at least 200 hours, which would be more than a week of computation. We also report the ML estimation results where the simulation size for ϵ draws is reduced from 100 to 20. The parameter estimates and their standard errors are very similar to that of the 100 ϵ draws. Notice that the sample average of the parameter estimates over 10 simulation/estimation exercises is 0.3895, which is closer to the truth than that of 100 ϵ draws:0.3795. However, the total CPU time of the ML estimation with 20 ϵ draws is 18 hours and 15 minutes, hardly different from 20 hours and 47 minutes of the 100 ϵ draws. That is, even though the reduction in the number of ϵ simulations does not result in any noticeable decline in the accuracy of the posterior, the gain in CPU time is also small.

Table 2: Posterior Means and Standard Errors (standard errors are in parenthesis)

Sample mean of 10 simulation/estimation exercises				
	Bayesian DP	Full Solution Bayes	Full Solution ML 100 α_i draws	true value
δ	0.3954 (0.0161)	0.3981 (0.0182)	0.3795 (0.0171)	0.4
μ_α	0.1974 (0.0105)	0.1977 (0.0105)	0.1701 (0.0135)	0.2
σ_α	0.1010 (0.00743)	0.1008 (0.00729)	0.09326 (0.0140)	0.1
σ_{ϵ_1}	0.3017 (0.00284)	0.3017 (0.00302)	0.3025 (0.00317)	0.3
σ_{ϵ_2}	0.3002 (0.0109)	0.3022 (0.0149)	0.2805 (0.0176)	0.3
b_1	0.09972 (0.00484)	0.1000 (0.00487)	0.1004 (0.00530)	0.1
b_2	0.3970 (0.00960)	0.3971 (0.00978)	0.4003 (0.0101)	0.4
b_e	0.4982 (0.0128)	0.4965 (0.0137)	0.5054 (0.0145)	0.5
σ_u	0.4000 (0.00317)	0.4003 (0.00321)	0.3990 (0.00317)	0.4
sample	100×100	100×100	100×100	
CPU time	4 hrs. 0 min.	30 hrs. 59 min.	20 hrs. 47 min.	
Sample std. error of 10 simulation/estimation exercises.				
δ	0.0151 (0.00196)	0.0148 (0.00246)	0.0140 (0.00185)	
μ_α	0.0118 (0.000491)	0.00560 (0.000565)	0.00969 (0.000963)	
σ_α	0.00536 (0.000395)	0.00536 (0.000376)	0.00935 (0.00257)	
σ_{ϵ_1}	0.00258 (0.000187)	0.00249 (0.000184)	0.00225 (0.000289)	
σ_{ϵ_2}	0.0103 (0.00252)	0.0127 (0.00322)	0.0116 (0.00160)	
b_1	0.00483 (0.000340)	0.00444 (0.000348)	0.00439 (0.000662)	
b_2	0.00597 (0.000509)	0.00590 (0.000650)	0.00583 (0.000806)	
b_e	0.0133 (0.00164)	0.0135 (0.00101)	0.0133 (0.00159)	
σ_u	0.00373 (0.000110)	0.00376 (0.000181)	0.00407 (0.000277)	

Sample mean of 10 simulation/estimation exercises, Full solution ML			
parameter	20 α_i draws	20 ϵ draws	true value
δ	0.3795 (0.0173)	0.3895 (0.0192)	0.4
μ_α	0.1450 (0.0123)	0.1764 (0.0157)	0.2
σ_α	0.1076 (0.0203)	0.09527 (0.0126)	0.1
σ_{ϵ_1}	0.3030 (0.00315)	0.3028 (0.00315)	0.3
σ_{ϵ_2}	0.2790 (0.0177)	0.2810 (0.0181)	0.3
b_1	0.1003 (0.00526)	0.09977 (0.00524)	0.1
b_2	0.3999 (0.0100)	0.4000 (0.00996)	0.4
b_e	0.5030 (0.0146)	0.5048 (0.0145)	0.5
σ_u	0.3988 (0.00318)	0.3988 (0.00317)	0.4
sample size	100×100	100×100	
CPU time	8 hrs. 43 min.	18 hrs. 15 min.	
Sample std. error of 10 simulation/estimation exercises.			
δ	0.0138 (0.00272)	0.0140 (0.00265)	
μ_α	0.0273 (0.00177)	0.0106 (0.00143)	
σ_α	0.0316 (0.00783)	0.0110 (0.00146)	
σ_{ϵ_1}	0.00234 (0.000311)	0.00235 (0.000303)	
σ_{ϵ_2}	0.0136 (0.00220)	0.0123 (0.00185)	
b_1	0.00488 (0.000590)	0.00485 (0.000574)	
b_2	0.00595 (0.000977)	0.00594 (0.000925)	
b_e	0.0138 (0.00152)	0.0143 (0.00159)	
σ_u	0.00392 (0.000280)	0.00391 (0.000276)	

Another estimation strategy for the simulated ML could be to expand the state variables of the DP problem to include both X and α. Then, we have to assign grid points for the three-dimensional state space points (K, X, α). If we assign 100 grid points per dimension, then we end up having $10,000$ times more grid points than before. Hence, the overall computational burden would be quite similar to the orginal simulated ML estimation strategy.

5.3 Experiment 3: Continuous State Space with Deterministic Transition

The framework is similar to the basic model in Experiment 1 except for the capital transition of the incumbent, which now is deterministic. Assume that if the incumbent decides to stay in, the next period capital is,

$$K_{t+1} = K_t.$$

If the firm decides to either exit or stay out, then the next period capital is 0, and if it enters, then the next period capital is,

$$\ln (K_{t+1}) = b_1 + u_{t+1},$$

where,

$$u_{t+1} \sim N(0, \sigma_u).$$

Since the state space is continuous, we use $K_1^{(t)}, ..., K_{M_K}^{(t)}$ as grid points. As in the previous experiment, we set $M_K = 10$ but let the grid points grow over iterations. Now, the formula for the expected value function for the incumbent who stays in is as follows.

$$\begin{aligned}
&\hat{E}\left[V_I(K, \epsilon', \theta)\right] \\
\equiv\ & \sum_{n=1}^{N(t)} \sum_{m=1}^{M_K} \left[\frac{1}{M_\epsilon} \sum_{j=1}^{M_\epsilon} V_I^{(t-n)}(K_m^{(t-n)}, \epsilon_j^{(t-n)}, \theta^{(t-n)}) \right] \\
& \frac{K_{h_K}\left(K - K_m^{(t-n)}\right) K_{h_\theta}(\theta - \theta^{(t-n)})}{\sum_{k=1}^{N(t)} \sum_{m=1}^{M_K} K_{h_K}\left(K - K_m^{(t-k)}\right) K_{h_\theta}(\theta - \theta^{(t-k)})},
\end{aligned}$$

where K_{h_K} is the kernel for the capital stock with bandwidth h_K. The expected value function for the entrant is different now because unlike the incumbent who stays in, the entrant faces uncertain future capital. Thus, the entrant's expected value function is,

$$\begin{aligned}
&\hat{E}_{K', \epsilon'}\left[V_I(K'(u), \epsilon, \theta)\right] \\
\equiv\ & \sum_{n=1}^{N(t)} \sum_{m=1}^{M_K} \left[\frac{1}{M_\epsilon} \sum_{j=1}^{M_\epsilon} V_I^{(t-n)}(K_m^{(t-n)}, \epsilon_j^{(t-n)}, \theta^{(t-n)}) \right] \\
& \times \frac{f\left(K_m^{(t-n)} | \theta^{(t-n)}\right) K_h(\theta - \theta^{(t-n)})}{\sum_{k=1}^{N(t)} \sum_{m=1}^{M_K} f\left(K_m^{(t-k)} | \theta^{(t-k)}\right) K_h(\theta - \theta^{(t-k)})}.
\end{aligned}$$

The formula for the expected value function for either the firm who stays out or the firm who exits is the same as in the infinite random grids case:

$$\begin{aligned}
&\hat{E}_{\epsilon'}\left[V_O(0, \epsilon, \theta)\right] \\
\equiv\ & \sum_{n=1}^{N(t)} \left[\frac{1}{M_\epsilon} \sum_{j=1}^{M_\epsilon} V_O^{(t-n)}(0, \epsilon_j^{(t-n)}, \theta^{(t-n)}) \right] \frac{K_h(\theta - \theta^{(t-n)})}{\sum_{k=1}^{N(t)} K_h(\theta - \theta^{(t-k)})}
\end{aligned}$$

We let the number of grid points increase up to $20,000$ over the iterations.

Table 3 shows the estimation results. We can see that the estimates parameters are close to the truth. The entire exercise took about 47 minutes.

Table 3: Posterior Means and Standard Deviations

(Standard deviations are in parenthesis)

parameter	estimate		true value
δ	0.1891	(0.0123)	0.2
α	0.1044	(0.00478)	0.1
σ_{ϵ_1}	0.3956	(0.00511)	0.4
σ_{ϵ_2}	0.3993	(0.0135)	0.4
b_1	0.1996	(0.00474)	0.2
σ_u	0.2017	(0.00301)	0.2
sample size	$10,000$		
CPU time	47 min 30 sec		

6 Conclusion

In conventional estimation methods of Dynamic Discrete Choice models, such as GMM, Maximum Likelihood or Markov Chain Monte Carlo, at each iteration step, given a new set of parameter values, the researcher first solves the Bellman equation to derive the expected value function, and then uses it to construct the likelihood or moments. That is, during the DP iteration, the researcher fixes the parameter values and does not "estimate". We propose a Bayesian estimation algorithm where the DP problem is solved and parameters estimated at the same time. In other words, we move parameters during the DP solution. This dramatically increases the speed of estimation. We have demonstrated the effectiveness of our approach by estimating a simple dynamic model of discrete entry-exit choice. Even though we are estimating a dynamic model, the required computational time is in line with the time required for Bayesian estimation of static models. The reason for the speed is clear. The computational burden of estimating dynamic models has been high because the researcher has to repeatedly evaluate the Bellman equation during a single estimation routine, keeping the parameter values fixed. We move parameters, i.e. 'estimate' the model after each Bellman equation evaluation. Since a single Bellman equation evaluation is computationally no different from computing a static model, the speed of our estimation exercise, too, is quite similar to that of a static model. The additional computational cost of our algorithm is the cost of using information obtained in past iterations. The more complex the model becomes, it becomes smaller relative to the cost of computing the full solution, which is what we have seen in the simulation/estimation examples.

Another computational obstacle in the estimation of a Dynamic Discrete Choice model is the Curse of Dimensionality. That is, the computational burden increases exponentially with the increase in the dimension of the state space. In our algorithm, even though at each iteration, the number of state space points on which we calculate the expected value function is small, the total number of 'effective' state space points over the entire solution/estimation iteration grows with the number of Bayesian DP iterations. This number can be made arbitrarily large without much additional computational cost. And it is the total number of 'effective' state space points that determines accuracy. Hence, our algorithm moves one step further in overcoming the

Curse of Dimensionality. This also explains why our nonparametric approximation of the expected value function works well under the assumption of continuous state space with deterministic transition function of the state variable. In this case, as is discussed in the main body of the paper, Rust (1997) random grid method may face computational difficulties.

It is worth mentioning that since we are locally approximating the expected value function nonparametrically, as we increase the number of parameters, we may face the "Curse of Dimensionality" in terms of the number of parameters to be estimated. So far, in our examples, this issue does not seem to have made a difference. The reason is that most dynamic models specify per period return function and transition functions to be smooth and well-behaved. Hence, we know in advance that the value functions we need to approximate are smooth, hence well suited for nonparametric approximation. Furthermore, the simulation exercises in the above examples show that with a reasonably large sample size, the MCMC simulations are tightly centered around the posterior mean. Hence, the actual multidimensional area where we need to apply nonparametric approximation is small. But in empirical exercises that involve many more parameters, one probably needs to adopt an iterative MCMC strategy where only up to 4 or 5 parameters are moved at once, which is also commonly done in conventional ML estimation.

7 References

Aguirreagabiria, Victor and Pedro Mira (2002) "Swapping the Nested Fixed Point Algorithm: A Class of Estimators for Discrete Markov Decision Models. " *Econometrica*, Vol. 70, pp. 1519 − 1543

Ackerberg, Daniel (2004) "A New Use of Importance Sampling to Reduce Computational Burden in Simulation Estimation." mimeo. University of Arizona.

Albert, James. and Siddharta Chib (1993) "Bayesian Analysis of binary and polychotomous data." *Journal of the American Statistical Association* 88, pp. 669-679.

Arcidiacono, Peter and John B. Jones (2003) "Finite Mixture Distributions, Sequential Likelihood and the EM Algorithm." *Econometrica*, Vol. 71. No. 3, pp. 933 − 946

Bierens, Herman J. (1994) "Topics in Advanced Econometrics." Cambridge University Press.

Chib, Siddhartha and Edward Greenberg (1996) "Markov Chain Monte Carlo Simulation Methods in Econometrics," *Econometric Theory,* vol. 12, pp. 409 − 431

Erdem, Tulin and Michael P. Keane (1996) "Decision Making under Uncertainty: Capturing Dynamic Brand Choice Processes in Turbulent Consumer Goods Markets." *Marketing Science*, Vol. $15:1$, pp. $1-20$.

Ferrall, Christopher (2005) "Solving Finite Mixture Models: Efficient Computation in Economics Under Serial and Parallel Execution." *Computational Economics*, Vol. 25, pp. $343-379$

Geweke, John and Michael P. Keane (1995) "Bayesian Inference for Dynamic Discrete Choice Models without the Need for Dynamic Programming." in Mariano, Scheuermann and Weeks (eds.), *Simulation Based Inference and Econometrics: Methods and Applications*, Cambridge University Press.

Geweke, John, Daniel Houser and Michael P. Keane (1998) "Simulation Based Inference for Dynamic Multinomial Choice Models." working paper, University of Arizona.

Haerdle, Wolfgang (1989) "Applied Nonparametric Regression." Cambridge University Press.

Heckman, James, J. (1981) "The Incidental Parameters Problem and the Problem of Initial Conditions in Estimating a Discrete Time-Discrete Data Stochastic Process." in *Structural Analysis of Discrete Data with Econometric Applications*, ed. C. F. Manski and D. McFadden. Cambridge, MA. MIT Press, $pp.179-195$

Hotz, Joseph V. and Robert Miller (1993) "Conditional Choice Probabilities and the the Estimation of Dynamic Models." *Review of Economic Studies*, Vol. 60, No. 3, pp. $497-529$.

Hotz, Joseph V., Robert A. Miller, Seth Sanders and Jeffrey Smith (1994) "A Simulation Estimator for Dynamic Models of Discrete Choice." *Review of Economic Studies*, Vol. 61, No. 2, pp. $265-289$.

Houser, Daniel (2003) "Bayesian Analysis of Dynamic Stochastic Model of Labor Supply and Savings." *Journal of Econometrics*, Vo. $113:2$, pp. $289-335$.

Imai, Susumu and Michael P. Keane (2004) "Intertemporal Labor Supply and Human Capital Accumulation." *International Economic Review*, Vol. 45, No. 2, pp. $601-641$.

Imai, Susumu and Kala Krishna (2004) "Employment, Deterrence and Crime in a Dynamic Model." *International Economic Review*, Vol. 45, No. 3, pp. $845-872$

Kasahara Hiroyuki and Katsumi Shimotsu (2006) "Nested Pseudo-likelihood Estimation and Bootstrap-based Inference for Structural Discrete Markov Decision Models." mimeo. University of Western Ontario.

Keane, Michael P. and Kenneth I. Wolpin (1994) "The Solution and Estimation of Discrete Choice Dynamic Programming Models by Simulation and Interpolation: Monte Carlo Evidence." *The Review of Economics and Statistics*, Vol. 74 (4), pp. 648 – 72

Keane, Michael P. and Kenneth I. Wolpin (1997) "The Career Decisions of Young Men." *Journal of Political Economy*, Vol. 105, pp. 473 – 521.

Lancaster, Tony (1997) "Exact Structural Inference in Optimal Job Search Models." *Journal of Business and Economic Statistics*, Vol. 15 (2), pp. 165 – 179.

McCullogh, Robert, and Peter Rossi (1994) "An Exact Likelihood Analysis of the Multinomial Probit Model." *Journal of Econometrics,* Vol. 64 (1), pp. 207 – 240.

Norets, Andriy (2006) "Inference in Dynamic Discrete Choice Models with Serially Correlated Unobserved State Varibles." mimeo, University of Iowa.

Osborne, Matthew J. (2006) "Consumer Learning, Habit Formation, and Heterogeneity: A Structural Estimation." mimeo. Stanford University.

Pakes, Ariel and Paul McGuire (2001) "Stochastic Algorithms, Symmetric Markov Perfect Equilibrium, and the 'Curse' of Dimensionality." *Econometrica,* Vol. 69, No. 5, pp. 1261 – 1281

Robert, Christian P. and George Casella (2004) "Monte Carlo Statistical Methods", *Springer Texts in Statistics.*

Roberts, Mark and James Tybout (1997) "The Decision to Export in Columbia: An Empirical Model of Entry with Sunk Costs." *American Economic Review*, Vol. 87, No. 4, pp. 545 – 564

Rosenthal, Jeffrey, S. (1995) "Minorization Conditions and Convergence Rates for Markov Chain Monte Carlo." *Journal of the American Statistical Association*, Vol. 90, No. 430, pp. 558 – 566

Rust, John (1997) "Using Randomization to Break the Curse of Dimensionality." *Econometrica*, Vol. 55, No. 5, pp. 999 – 1033

Tanner, Martin A. and Wing H. Wong (1987) "The Calculation of Posterior Distributions by Data Augmentation." *Journal of the American Statistical Association*, vol. 82, pp. 528 – 549.

Tierney, Luke (1994) "Markov Chains for Exploring Posterior Distributions." *The Annals of Statistics*, Vol. 22, No. 4, pp. 1701 – 176

Appendix

Proof of Theorem 1

For notational convenience, in the subsequent proofs we omit Ω. We need to show that for any $s \in S$, ϵ, $\theta \in \Theta$,

$$V^{(t)}\left(s, \epsilon, \theta\right) \xrightarrow{p} V\left(s, \epsilon, \theta\right) \text{ uniformly, as } t \to \infty$$

But since,

$$V^{(t)}(s, \epsilon, \theta) = \max_{a \in A} \mathcal{V}^{(t)}(s, a, \epsilon, \theta), \quad V(s, \epsilon, \theta) = \max_{a \in A} \mathcal{V}(s, a, \epsilon, \theta),$$

it suffices to show that for any $s \in S$, $a \in A$, ϵ, $\theta \in \Theta$,

$$\mathcal{V}^{(t)}\left(s, a, \epsilon, \theta\right) \xrightarrow{p} \mathcal{V}\left(s, a, \epsilon, \theta\right) \text{ as } t \to \infty.$$

Define

$$W_{N(t),h}(\theta, \theta^{(t-n)}) \equiv \frac{K_h(\theta - \theta^{(t-n)})}{\sum_{k=1}^{N(t)} K_h(\theta - \theta^{(t-k)})}.$$

Then, the difference between the true value function of action a and that obtained by the Bayesian Dynamic Programming iteration can be decomposed into 3 parts as follows.

$$\mathcal{V}\left(s, a, \epsilon, \theta\right) - \mathcal{V}^{(t)}\left(s, a, \epsilon, \theta\right)$$

$$= \beta \left[\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - \sum_{n=1}^{N(t)} V^{(t-n)}(s', \epsilon^{(t-n)}, \theta^{*(t-n)}) W_{N(t),h}(\theta, \theta^{*(t-n)}) \right]$$

$$= \beta \left[\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - \sum_{n=1}^{N(t)} V(s', \epsilon^{(t-n)}, \theta^{*(t-n)}) W_{N(t),h}(\theta, \theta^{*(t-n)}) \right]$$

$$+ \beta \left[\sum_{n=1}^{N(t)} \left[V(s', \epsilon^{(t-n)}, \theta^{*(t-n)}) - V^{(t-n)}(s', \epsilon^{(t-n)}, \theta^{*(t-n)}) \right] W_{N(t),h}(\theta, \theta^{*(t-n)}) \right]$$

$$\equiv A_1^{(t)}\left(\theta\right) + A_2^{(t)}\left(\theta\right)$$

The kernel smoothing part is difficult to handle because the underlying distribution of $\theta^{*(s)}$ has a density function conditional on $\theta^{(s-1)}$. Therefore, instead of deriving the asymptotic value of $\frac{1}{N(t)} \sum_{k=1}^{N(t)} K_h(\theta - \theta^{*(t-k)})$, as is done in standard nonparametric kernel asymptotics, we sometimes derive and use its asymptotic lower bound and upper bound. Lemma 1 in the main text is used for the derivation of the asymptotic lower bound. Lemma 2 is used for the derivation of the asymptotic upper bound. Using the results of Lemma 1 and 2, in Lemma 3 we prove that $A_1^{(t)}(\theta) \to 0$ uniformly in $\theta \in \Theta$.

Lemma 3: $\left|A_1^{(t)}(\theta)\right| \xrightarrow{P} 0$ uniformly in Θ as $t \to \infty$.

Proof: Recall that,

$$\left|\frac{A_1^{(t)}(\theta)}{\beta}\right| = \left|\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - \sum_{n=1}^{N(t)} V(s', \epsilon^{(t-n)}, \theta^{*(t-n)}) W_{N(t),h}(\theta, \theta^{*(t-n)})\right|.$$

Rewrite it as,

$$\left|\frac{A_1^{(t)}(\theta)}{\beta}\right| = \left|\frac{\frac{1}{N(t)}\sum_{n=1}^{N(t)}\left(\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - V(s', \epsilon^{(t-n)}, \theta^{*(t-n)})\right) K_h(\theta - \theta^{*(t-n)})}{\frac{1}{N(t)}\sum_{k=1}^{N(t)} K_h(\theta - \theta^{*(t-k)})}\right|.$$

We show that the numerator goes to zero in probabilty uniformly in Θ and the denominator is bounded below by a positive number uniformly in Θ with probability arbitrarily close to one as $t \to \infty$.

Let

$$\begin{aligned}
\widehat{X}_{N(t)}(\theta) &\equiv \frac{1}{N(t)}\sum_{n=1}^{N(t)}\left[\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - V(s', \epsilon^{(t-n)}, \theta^{*(t-n)})\right] \\
&\quad K_h(\theta - \theta^{*(t-n)}) \\
\widehat{X}_{N(t),t-n}(\theta) &\equiv \left[\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - V(s', \epsilon^{(t-n)}, \theta^{*(t-n)})\right] K_h(\theta - \theta^{*(t-n)})
\end{aligned}$$

Then, because $\epsilon^{(t-n)}$'s are i.i.d. and $\epsilon^{(t-n)} \sim F_{\epsilon'}(\epsilon', \theta)$,

$$\begin{aligned}
&E\left[X_{N(t),t-n}(\theta)|\theta^{(t-n)}\right] \\
&= E\left[\left[\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - \int V(s', \epsilon', \theta^{*(t-n)}) dF_{\epsilon'}(\epsilon', \theta^{*(t-n)})\right]\right. \\
&\quad \left. K_h(\theta - \theta^{*(t-n)})|\theta^{(t-n)}\right] \\
&\to \left[\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - \int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta)\right] q\left(\theta^{(t-n)}, \theta\right) = 0 \\
\text{as } h &\to 0 \qquad \text{(A1)}
\end{aligned}$$

Also, because V is uniformly bounded, $\left|E\left[X_{N(t),t-n}(\theta)|\theta^{(t-n)}\right]\right|$ is uniformly bounded. Therefore, from Dominated Convergence Theorem,

$$E\left[X_{N(t),t-n}(\theta)\right] \to 0 \qquad \text{(A2)}$$

We can also show that the above convergence is uniform. For some $M > 0$

$$\begin{aligned}
& \sup_{\theta\in\Theta}\left|E\left[X_{N(t),t-n}(\theta)|\theta^{(t-n)}\right]\right| \\
= & \sup_{\theta\in\Theta}\Big|E\Big[\Big[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-\int V(s',\epsilon',\theta^{*(t-n)})dF_{\epsilon'}(\epsilon',\theta^{*(t-n)})\Big] \\
& K_h(\theta-\theta^{*(t-n)})I\left(\left|\theta-\theta^{*(t-n)}\right|\le M\sqrt{h}\right)|\theta^{(t-n)}\Big]\Big| \\
& +\sup_{\theta\in\Theta}\Big|E\Big[\Big[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-\int V(s',\epsilon',\theta^{*(t-n)})dF_{\epsilon'}(\epsilon',\theta^{*(t-n)})\Big] \\
& K_h(\theta-\theta^{*(t-n)})I\left(\left|\theta-\theta^{*(t-n)}\right|> M\sqrt{h}\right)|\theta^{(t-n)}\Big]\Big| \\
\le & \sup_{\theta,\theta'\in\Theta,|\theta-\theta'|\le M\sqrt{h}}\left|\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-\int V(s',\epsilon',\theta')dF_{\epsilon'}(\epsilon',\theta')\right| \\
& +2\sup_{\theta}\left|\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)\right|\varepsilon_1\int_{|z|>M/\sqrt{h}}K(z)\widetilde{g}\,(\theta-hz)\,dz \qquad \text{(A3)}
\end{aligned}$$

The second inequality comes from Lemma 2, where $q\,(\theta,\theta')\le\varepsilon_1 g\,(\theta')$ for any θ, $\theta'\in\Theta$. Because V satisfies the Lipschitz condition and $dF_{\epsilon'}(\epsilon',\theta)$ is uniformly continuous in $\theta\in\Theta$, the first term of the equation $A3$ converges to zero as $h\to 0$. The second term also converges to zero as $h\to 0$. Therefore, we have shown that the convergence in $A2$ is uniform.

Furthermore,

$$E\left[X_{N(t),t-n}(\theta)X_{N(t),t-m}(\theta)\right]=E\left[E\left[X_{N(t)n}(\theta)|\theta^{(t-n-1)}\right]X_{N(t)m}(\theta)\right]\to 0 \qquad \text{(A4)}$$

for $t-n>t-m$ as $h\to 0$. Hence, there exists $\eta\,(h)>0$ such that $\eta\,(h)\to 0$ as $h\to 0$ and

$$\begin{aligned}
& E\left[X_{N(t)}(\theta)^2\right] \\
\le & \frac{1}{N\,(t)^2}\sum_{n=1}^{N(t)}E\left[X_{N(t)n}(\theta)^2\right]+\eta\,(h) \\
\le & \sup_{s',\theta}\frac{1}{N\,(t)^2}\sum_{n=1}^{N(t)}E\Big[\Big[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-V(s',\epsilon^{(t-n)},\theta^{*(t-n)})\Big] \\
& K_h(\theta-\theta^{*(t-n)})\Big]^2+\eta\,(h) \\
\le & \sup_{s',\theta}\frac{1}{N\,(t)\,h}\int K(z)^2 \\
& E\left[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-V(s',\epsilon^{(t-n)},\theta-zh)\right]^2\varepsilon_1\widetilde{g}\,(\theta-zh)\,dz+\eta\,(h) \\
\to & 0.
\end{aligned}$$

The last inequality comes from Lemma 2. That is, we have shown that there exists $\widetilde{g}$, $\varepsilon_1 \geq 1$ such that $q(\theta, \theta') \leq \varepsilon_1 \widetilde{g}(\theta')$ for any θ, $\theta' \in \Theta$. Hence, from Chebychev Inequality, for any $\gamma > 0$, $\delta > 0$ there exists $\bar{t}_\gamma$ such that for any $t > \bar{t}_\gamma$, i.e., $N(t) \geq N(\bar{t}_\gamma)$,

$$\Pr\left\{\left\|\frac{1}{N(t)}\sum_{n=1}^{N(t)} X_{N(t),t-n}\right| - 0\right| \geq \delta\right\} \leq \frac{\gamma}{\delta^2} \tag{A4}$$

Since γ/δ^2 can be made arbitrarily small, this shows that the numerator in $A_1^{(t)}(\theta)/\beta$ converges to zero in probability. Next, we show that it converges to zero uniformly in Θ. Here we follow Section 10.3 of Bierens (1994). Denote

$$R_{N(t)}(\theta) \equiv \frac{1}{N(t)}\sum_{n=1}^{N(t)} V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) K_h(\theta - \theta^{*(t-n)})$$

By using the Fourier transform, we can express the kernel as follows.

$$K(x) = \left(\frac{1}{2\pi}\right)^J \int \exp(-iz'x)\,\psi(z)\,dz$$

where

$$\psi(z) = \int \exp(iz'x)\,K(x)\,dx.$$

Because of Assumption 6,

$$\int |\psi(z)|\,dz < \infty.$$

Then, by Fourier inversion

$$\begin{aligned} & R_{N(t)}(\theta) \\ = & \left[\frac{1}{2\pi}\right]^J \frac{1}{N(t)h^J}\sum_{n=1}^{N(t)} V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right)\int \exp\left(\frac{-iz'\left(\theta - \theta^{*(t-n)}\right)}{h}\right)\psi(z)\,dz \\ = & \left[\frac{1}{2\pi}\right]^J \frac{1}{N(t)}\int\left[\sum_{n=1}^{N(t)} V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right)\exp\left(iz'\theta^{*(t-n)}\right)\right]\exp(-iz'\theta)\,\psi(hz)\,dz. \end{aligned} \tag{A5}$$

Hence,

$$\begin{aligned} & E\left[\sup_{\theta\in\Theta}\left|R_{N(t)}(\theta) - E\left[R_{N(t)}(\theta)\right]\right|\right] \\ \leq & \left[\frac{1}{2\pi}\right]^J \int E\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}\left\{V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right)\exp\left(iz'\theta^{*(t-n)}\right)\right.\right. \\ & \left.\left. - E\left[V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right)\exp\left(iz'\theta^{*(t-n)}\right)\right]\right\}\right|\,|\psi(hz)|\,dz \end{aligned} \tag{A6}$$

Using the Liapunov's Inequality, and

$$\exp(ia) = \cos(a) + i \sin(a),$$

we get

$$\begin{aligned} & E\left| \frac{1}{N(t)} \sum_{n=1}^{N(t)} \left\{ V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \exp\left(iz'\theta^{*(t-n)}\right) \right. \right. \\ & \left. \left. -E\left[V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \exp\left(iz'\theta^{*(t-n)}\right)\right] \right\} \right| \\ \leq \; & \left\{ Var\left[\frac{1}{N(t)} \sum_{n=1}^{N(t)} V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \cos\left(z'\theta^{*(t-n)}\right)\right] \right. \\ & \left. +Var\left[\frac{1}{N(t)} \sum_{n=1}^{N(t)} V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \sin\left(z'\theta^{*(t-n)}\right)\right] \right\}^{1/2} \end{aligned}$$

Now, because $\epsilon^{(t-n)}, \epsilon^{(t-m)}$ $n \neq m$ are i.i.d,

$$\begin{aligned} & Var\left[\frac{1}{N(t)} \sum_{n=1}^{N(t)} V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \cos\left(t'\theta^{*(t-n)}\right)\right] \\ = \; & \frac{1}{N(t)^2} \sum_{n=1}^{N(t)} \sum_{m=1}^{N(t)} Cov\left[V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \cos\left(z'\theta^{*(t-n)}\right)\right. \\ & \left., V\left(s', \epsilon^{(t-m)}, \theta^{*(t-m)}\right) \cos\left(z'\theta^{*(t-m)}\right)\right] \\ = \; & \frac{1}{N(t)^2} \sum_{n=1}^{N(t)} Var\left[V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \cos\left(z'\theta^{*(t-n)}\right)\right] \end{aligned} \tag{A7}$$

Similarly,

$$\begin{aligned} & Var\left[\frac{1}{N(t)} \sum_{n=1}^{N(t)} V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \sin\left(z'\theta^{*(t-n)}\right)\right] \\ = \; & \frac{1}{N(t)^2} \sum_{n=1}^{N(t)} \sum_{m=1}^{N(t)} Cov\left[V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \sin\left(z'\theta^{*(t-n)}\right)\right. \\ & \left., V\left(s', \epsilon^{(t-m)}, \theta^{*(t-m)}\right) \sin\left(z'\theta^{*(t-m)}\right)\right] \\ = \; & \frac{1}{N(t)^2} \sum_{n=1}^{N(t)} Var\left[V\left(s', \epsilon^{(t-n)}, \theta^{*(t-n)}\right) \sin\left(z'\theta^{*(t-n)}\right)\right] \end{aligned}$$

Together, we derive that

$$
\begin{aligned}
& E\left[\sup_{\theta\in\Theta}\left|R_{N(t)}(\theta)-E\left[R_{N(t)}(\theta)\right]\right|\right] \\
\leq\; & \left[\frac{1}{2\pi}\right]^{J}\int\left\{\frac{1}{N(t)^2}\sum_{n=1}^{N(t)}\left\{Var\left[V\left(s',\epsilon^{(t-n)},\theta^{*(t-n)}\right)\cos\left(z'\theta^{*(t-n)}\right)\right]\right.\right. \\
& \left.\left.+Var\left[V\left(s',\epsilon^{(t-n)},\theta^{*(t-n)}\right)\sin\left(z'\theta^{*(t-n)}\right)\right]\right\}\right\}^{1/2}\left|\psi(hz)\right|dz \\
\leq\; & \left[\frac{1}{2\pi}\right]^{J}\left\{\frac{1}{N(t)^2}\sum_{n=1}^{N(t)}\sup_{\epsilon,\theta\in\Theta}\left|V(s',\epsilon,\theta)\right|^2\right\}^{1/2}\int\left|\psi(hz)\right|dz \\
=\; & \left[\frac{1}{2\pi}\right]^{J}\left\{\frac{1}{N(t)h^{2J}}\sup_{\epsilon,\theta\in\Theta}\left|V(s',\epsilon,\theta)\right|^2\right\}^{1/2}\int\left|\psi(z)\right|dz\to 0 \text{ as } N(t)\to\infty
\end{aligned}
\tag{A8}
$$

Therefore, from Chebychev Inequality, for any $\delta_1>0$, $\eta_1>0$ there exists $t(\delta_1,\eta_1)$ such that for any $t>t(\delta_1,\eta_1)$

$$
\Pr\left[\sup_{\theta\in\Theta}\left|R_{N(t)}(\theta)-E\left[R_{N(t)}(\theta)\right]\right|<\delta_1\right]>1-\eta_1 \tag{A9}
$$

Furthermore, from we know from the uniform convergence of $E\left[X_{N(t)n}(\theta)\right]$ to zero in Θ that

$$
\begin{aligned}
E\left[R_{N(t)}(\theta)\right] &\to E\left[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)K_h(\theta-\theta^{*(t-n)})\right] \\
&= \int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta).
\end{aligned}
$$

uniformly in Θ. Together with $A9$, we have shown that the numerator of $\left|A_1(\theta)\right|$ converges to zero uniformly in Θ. We next show that the denominator is uniformly bounded below with probability arbitrarily close to one as t goes to infinity. Let

$$
R^{(t-n)}\equiv\varepsilon_0\frac{g\left(\theta^{*(t-n)}\right)}{q\left(\theta^{(t-n)},\theta^{*(t-n)}\right)}.
$$

Then, from Lemma 1, $0\leq R^{(t-n)}\leq 1$ and $0<\varepsilon_0\leq 1$. Also, define a random variable $Y^{(t-n)}(\theta)$ as follows.

$$
Y^{(t-n)}(\theta)=\begin{cases} K_h\left(\theta-\theta^{*(t-n)}(q)\right) & \text{with probability } R^{(t-n)} \\ 0 & \text{with probability } 1-R^{(t-n)}\end{cases}.
$$

Then, $Y^{(t-n)}$ is a mixture of 0 and $K_h\left(\theta-\theta^{*(t-n)}(g)\right)$, with the mixing probability being $1-\varepsilon_0$ and ε_0. That is,

$$
Y^{(t-n)}(\theta)=\begin{cases} K_h\left(\theta-\theta^{*(t-n)}(g)\right) & \text{with probability } \varepsilon_0 \\ 0 & \text{with probability } 1-\varepsilon_0\end{cases}
$$

or, equivalently,

$$Y^{(t-n)}(\theta) = K_h\left(\theta - \theta^{*(t-n)}(g)\right) I^{(t-n)}$$

where

$$I^{(t-n)} = \begin{cases} 1 & \text{with probability } \varepsilon_0 \\ 0 & \text{with probability } 1-\varepsilon_0 \end{cases}$$

Further, from the construction of $Y^{(t-n)}$,

$$Y^{(t-n)}(\theta) \leq K_h\left(\theta - \theta^{*(t-n)}(q)\right).$$

Now, because $\theta^{*(t-n)}(g)$, $n = 1, ..., N(t)$ are i.i.d., following Bierens (1994), section 10.1, and 10.3, we can derive uniform convergence. That is, by using the Fourier transform of the kernel,

$$\begin{aligned}
&\frac{1}{N(t)}\sum_{n=1}^{N(t)} Y^{(t-n)}(\theta) \\
=\ & \frac{1}{N(t)}\sum_{n=1}^{N(t)} I^{(t-n)}\left(\frac{1}{2\pi h}\right)^J \int \exp\left(\frac{-iz'\left(\theta - \theta^{*(t-n)}(g)\right)}{h}\right)\psi(z)\,dz \\
=\ & \left(\frac{1}{2\pi h}\right)^J \int \left[\frac{1}{N(t)}\sum_{n=1}^{N(t)} I^{(t-n)} \exp\left(iz'\theta^{*(t-n)}(g)\right)\right] \exp(-iz'\theta)\,\psi(hz)\,dz
\end{aligned}$$

Hence, using equations 2.3.4 and 2.3.5 in Bierens (1994), we get

$$\begin{aligned}
&E\left[\sup_\theta \left|\frac{1}{N(t)}\sum_{n=1}^{N(t)} Y^{(t-n)}(\theta) - E\left[\frac{1}{N(t)}\sum_{n=1}^{N(t)} Y^{(t-n)}(\theta)\right]\right|\right] \\
\leq\ & \left(\frac{1}{2\pi}\right)^J \int E\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}\left[I^{(t-n)} \exp\left(iz'\theta^{*(t-n)}(g)\right)\right.\right. \\
& \left.\left. - E\left[I^{(t-n)} \exp\left(iz'\theta^{*(t-n)}(g)\right)\right]\right]\right| |\psi(hz)|\,dz \\
\leq\ & \sqrt{\frac{E\left[I^{(t-n)2}\right]}{N(t)}}\left(\frac{1}{2\pi}\right)^J \int |\psi(ht)|\,dt = \sqrt{\frac{E\left[I^{(t-n)2}\right]}{N(t)\,h^J}}\left(\frac{1}{2\pi}\right)^J \int |\psi(t)|\,dt \\
\rightarrow\ & 0
\end{aligned} \tag{A11}$$

as $t \rightarrow \infty$. Therefore, using Chebychev's Inequality, we can show that

$$\sup_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)} Y^{(t-n)}(\theta) - \varepsilon_0 g(\theta)\right| \xrightarrow{P} 0 \tag{A12}$$

Therefore, for any $\kappa > 0$, $\eta > 0$, there exists $\bar{t} > 0$, $\overline{N} \equiv N(\bar{t})$ such that for any $t > \bar{t}$, i.e, $N(t) > \overline{N}$,

$$\Pr\left[\sup_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}Y^{(t-n)}(\theta) - \varepsilon_0 g(\theta)\right| \leq \kappa\right] > 1-\eta.$$

That is,

$$\Pr\left[\inf_{\theta\in\Theta}\frac{1}{N(t)}\sum_{n=1}^{N(t)}Y^{(t-n)}(\theta) + \kappa \geq \varepsilon_0 \inf_{\theta\in\Theta} g(\theta)\right] > 1-\eta$$

Now, choose $\kappa < \frac{1}{2}\inf_{\theta\in\Theta}\varepsilon_0 g(\theta)$. Then,

$$\Pr\left[\inf_{\theta\in\Theta}\frac{1}{N(t)}\sum_{n=1}^{N(t)}Y^{(t-n)}(\theta) > \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g(\theta)\right] > 1-\eta.$$

Since $\sum_{n=1}^{N} K_h\left(\theta - \theta^{*(t-n)}(q)\right) \geq \sum_{n=1}^{N} Y^{(t-n)}$, we conclude that for any $\eta > 0$, there exists $\bar{t}_\eta > 0$, $\overline{N} \equiv N(\bar{t}_\eta)$ such that for any $t > \bar{t}_\eta$, i.e, $N(t) > \overline{N}$,

$$\Pr\left[\inf_{\theta\in\Theta}\frac{1}{N(t)}\sum_{n=1}^{N(t)}K_h\left(\theta - \theta^{*(t-n)}(q)\right) > \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g(\theta)\right] > 1-\eta. \qquad \text{(A13)}$$

From the uniform convergence of the numerator to zero, and $A13$, we can see that for $\bar{t} = \max\{\bar{t}(\delta_1, \eta_1), \bar{t}_\eta\} > 0$, $\overline{N} \equiv N(\bar{t})$, the following holds: for any $t > \bar{t}$, i.e,

$N(t) > \overline{N}$

$$
\begin{aligned}
& \Pr\left[\sup_{\theta\in\Theta}\left|\frac{\frac{1}{N(t)}\sum_{n=1}^{N(t)}\left[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-V(s',\epsilon^{(t-n)},\theta^{*(t-n)})\right]K_h(\theta-\theta^{*(t-n)})}{\frac{1}{N(t)}\sum_{n=1}^{N(t)}K_h\left(\theta-\theta^{*(t-n)}\right)}\right|\right. \\
& \left.\leq\frac{\delta_1}{\frac{1}{2}\varepsilon_0\inf_{\theta\in\Theta}g(\theta)}\right] \\
\geq\ & \Pr\left[\frac{\sup_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}\left[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-V(s',\epsilon^{(t-n)},\theta^{*(t-n)})\right]K_h\right|}{\inf_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}K_h\left(\theta-\theta^{*(t-n)}\right)\right|}\right. \\
& \left.\leq\frac{\delta_1}{\frac{1}{2}\varepsilon_0\inf_{\theta\in\Theta}g(\theta)}\right] \\
\geq\ & \Pr\left\{\left[\sup_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}\left[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-V(s',\epsilon^{(t-n)},\theta^{*(t-n)})\right]K_h\right|\leq\delta_1\right]\right. \\
& \left.\bigcap\left[\inf_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}K_h\left(\theta-\theta^{*(t-n)}\right)\right|>\frac{1}{2}\varepsilon_0\inf_{\theta\in\Theta}g(\theta)\right]\right\} \\
\geq\ & 1-\Pr\left[\sup_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}\left[\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)-V(s',\epsilon^{(t-n)},\theta^{*(t-n)})\right]K_h\right|>\delta_1\right] \\
& -\Pr\left[\inf_{\theta\in\Theta}\left|\frac{1}{N(t)}\sum_{n=1}^{N(t)}K_h\left(\theta-\theta^{*(t-n)}\right)\right|\leq\frac{1}{2}\varepsilon_0\inf_{\theta\in\Theta}g(\theta)\right] \\
\geq\ & 1-\eta_1-\eta \qquad\qquad \text{(A14)}
\end{aligned}
$$

uniformly over Θ. Since $\delta_1/\left[\frac{1}{2}\inf_{\theta\in\Theta}\varepsilon_0 g(\theta)\right]$ can be made arbitrarily small by choosing δ_1 small enough, we have shown that

$$\sup_{\theta\in\Theta}\left|A_1^{(t)}(\theta)\right|\xrightarrow{P}0 \text{ as } N(t)\to\infty.$$

From Lemma 3, we know that,

$$\sup_{\theta\in\Theta}\left|A_1^{(t)}(\theta)\right|\xrightarrow{P}0, as\ t\to\infty,$$

Therefore,

$$\left|A_1^{(t)}(\theta^{(t)})\right|\xrightarrow{P}0\ as\ t\to\infty$$

Now,

$$\mathcal{V}(s,a,\epsilon,\theta) - \mathcal{V}^{(t)}(s,a,\epsilon,\theta) = A_1^{(t)}(\theta)$$
$$+\beta\left[\sum_{n=1}^{N(t)}\left[V(s',\epsilon^{(t-n)},\theta^{*(t-n)}) - V^{(t-n)}(s',\epsilon^{(t-n)},\theta^{*(t-n)})\right]W_{N(t),h}(\theta,\theta^{*(t-n)})\right] \quad \text{(A15)}$$

Notice that if $V(s,\epsilon,\theta) \geq V^{(t)}(s,\epsilon,\theta)$, then

$$\begin{aligned} 0 &\leq V(s,\epsilon,\theta) - V^{(t)}(s,\epsilon,\theta) = Max_{a\in A}\mathcal{V}(s,a,\epsilon,\theta) - Max_{a\in A}\mathcal{V}^{(t)}(s,a,\epsilon,\theta) \\ &\leq Max_{a\in A}\left[\mathcal{V}(s,a,\epsilon,\theta) - \mathcal{V}^{(t)}(s,a,\epsilon,\theta)\right] \leq Max_{a\in A}\left|\mathcal{V}(s,a,\epsilon,\theta) - \mathcal{V}^{(t)}(s,a,\epsilon,\theta)\right| \end{aligned}$$

Similarly, if $V(s,\epsilon,\theta) \leq V^{(t)}(s,\epsilon,\theta)$, then

$$\begin{aligned} 0 &\leq V^{(t)}(s,\epsilon,\theta) - V(s,\epsilon,\theta) = Max_{a\in A}\mathcal{V}^{(t)}(s,a,\epsilon,\theta) - Max_{a\in A}\mathcal{V}(s,a,\epsilon,\theta) \\ &\leq Max_{a\in A}\left[\mathcal{V}^{(t)}(s,a,\epsilon,\theta) - \mathcal{V}(s,a,\epsilon,\theta)\right] \leq Max_{a\in A}\left|\mathcal{V}(s,a,\epsilon,\theta) - \mathcal{V}^{(t)}(s,a,\epsilon,\theta)\right| \end{aligned}$$

Hence, taking supremum over $s\prime$ on the right hand side of $A15$ and then taking absolute values on both sides, we obtain:

$$\begin{aligned} &\left|V(s,\epsilon,\theta) - V^{(t)}(s,\epsilon,\theta)\right| \leq Max_{a\in A}\left|\mathcal{V}(s,a,\epsilon,\theta) - \mathcal{V}^{(t)}(s,a,\epsilon,\theta)\right| \\ \leq\ &\sup_{s'\in S}\left|A_1^{(t)}(\theta)\right| \\ &+\beta\left[\sum_{n=1}^{N(t)}\sup_{\widehat{s}\in S}\left|V(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)}) - V^{(t-n)}(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})\right|W_{N(t),h}(\theta,\theta^{*(t-n)})\right] \end{aligned} \quad \text{(A15')}$$

Now, $\left|V(s,\epsilon,\theta) - V^{(t)}(s,\epsilon,\theta)\right|$ appears on the LHS and $\left|V(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)}) - V^{(t-n)}(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})\right|$ appears on the RHS of equation $A15\prime$. Using this, we can recursively substitute away $\left|V(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)}) - V^{(t-n)}(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})\right|$. This logic is used in the following Lemma. Before we proceed with the Lemman and its proof, we introduce some additional notation. For $\tau < t$, let

$$\widetilde{W}(t,\tau) \equiv \beta W_{N(t),h}(\theta,\theta').$$

where θ is the parameter vector at iteration t and θ' the parameter vector at iteration τ. Now, for $\underline{N} \geq 1$ and for m such that $0 < m \leq \underline{N}+1$, define

$$\begin{aligned} &\Psi_m(t+\underline{N},t,\tau) \\ \equiv\ &\{J_m = (t_m,t_{m-1},...,t_1,t_0) : t_m = t+\underline{N} > t_{m-1} > .... > t_2 > t_1 \geq t, t_0 = \tau\}. \end{aligned}$$

That is, $\Psi_m(t+\underline{N},t,\tau)$ the resulting set of iterations where the largest is $t+N$ and the smallest τ, and the other $m-1$ iterations are greater than or equal to t.

Furthermore, let

$$\widehat{W}(t+\underline{N},t,\tau) \equiv \sum_{m=1}^{\underline{N}+1} \left\{ \sum_{\Psi_m(t+\underline{N},t,\tau)} \prod_{k=1}^{m} \widetilde{W}\left(t_k,t_{k-1}\right) \right\}.$$

Notice that $\widehat{W}(t,t,\tau) \equiv \widetilde{W}(t,\tau)$.

Lemma 4:

For any $\underline{N} \geq 1$, $t > 0$,

$$\begin{aligned} & \left|V\left(s,\epsilon,\theta\right) - V^{(t+\underline{N})}\left(s,\epsilon,\theta\right)\right| \\ \leq & \sup_{s' \in S}\left|A_1^{(t+\underline{N})}\left(\theta\right)\right| \\ & + \sum_{m=0}^{\underline{N}-1} \widehat{W}\left(t+\underline{N},t+\underline{N}-m,t+\underline{N}-m-1\right) \sup_{s' \in S}\left|A^{(t+\underline{N}-m-1)}\left(\theta^{*(t+\underline{N}-m-1)}\right)\right| \\ & + \sum_{n=1}^{N(t)} \sup_{\widehat{s} \in S}\left|V(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)}) - V^{(t-n)}(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})\right| \widehat{W}(t+\underline{N},t,t-n) \end{aligned} \quad (A16)$$

Furthermore,

$$\sum_{n=1}^{N(t)} \widehat{W}(t+\underline{N},t,t-n) \leq \beta \quad (A17)$$

Proof of Lemma 5.

First, we show that inequality $A16$ and $A17$ hold for $\underline{N}=1$. For iteration $t+1$, we get

$$\begin{aligned} & \left|V\left(s,\epsilon,\theta\right) - V^{(t+1)}\left(s,\epsilon,\theta\right)\right| \\ \leq & \sup_{s' \in S}\left|A_1^{(t+1)}\left(\theta\right)\right| \\ & + \sum_{n=1}^{N(t+1)} \sup_{s' \in S}\left|V(s',\epsilon^{(t+1-n)},\theta^{*(t+1-n)}) - V^{(t+1-n)}(s',\epsilon^{(t+1-n)},\theta^{*(t+1-n)})\right| \\ & \widetilde{W}(t+1,t+1-n) \\ \leq & \sup_{s' \in S}\left|A_1^{(t+1)}\left(\theta\right)\right| + \sup_{s' \in S}\left|V(s',\epsilon^{(t)},\theta^{*(t)}) - V^{(t)}(s',\epsilon^{(t)},\theta^{*(t)})\right| \widetilde{W}\left(t+1,t\right) \\ & + \sum_{n=1}^{N(t+1)-1} \sup_{s' \in S}\left|V(s',\epsilon^{(t-n)},\theta^{*(t-n)}) - V^{(t-n)}(s',\epsilon^{(t-n)},\theta^{*(t-n)})\right| \widetilde{W}(t+1,t-n) \end{aligned}$$

Now, we substitute away $\left|V(s',\epsilon^{(t)},\theta^{*(t)}) - V^{(t)}(s',\epsilon^{(t)},\theta^{*(t)})\right|$ by using $A15'$) and the fact that $N(t) \geq N(t+1)-1$,

$$
\begin{aligned}
&\left|V\left(s,\epsilon,\theta^{*(t+1)}\right)-V^{(t+1)}\left(s,\epsilon,\theta^{*(t+1)}\right)\right| \\
\leq\;& \sup_{s'\in S}\left|A_1^{(t+1)}(\theta^{*(t+1)})\right| + \sup_{s'\in S}\left|A^{(t)}(\theta^{*(t)})\right|\widetilde{W}(t+1,t) \\
&+\sum_{n=1}^{N(t)}\sup_{\widehat{s}\in S}\left|V(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})-V^{(t-n)}(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})\right| \\
&\{\widetilde{W}(t+1,t)\widetilde{W}(t,t-n)+\widetilde{W}(t+1,t-n)\} \\
=\;& \sup_{s'\in S}\left|A_1^{(t+1)}\left(\theta^{*(t+1)}\right)\right| + \sup_{s'\in S}\left|A_1^{(t)}\left(\theta^{*(t)}\right)\right|\widehat{W}(t+1,t+1,t) \\
&+\sum_{n=1}^{N(t)}\sup_{\widehat{s}\in S}\left|V(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})-V^{(t-n)}(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})\right|\widehat{W}\left(t+1,t,t-n\right)
\end{aligned}
$$

Hence, Inequality $A16$ holds for $\underline{N}=1$.

Furthermore, because $\sum_{n=1}^{N(t)}\widetilde{W}(t,t-n)/\beta=\sum_{n=1}^{N(t)}W_{N(t),h}(\theta^{*(t)},\theta^{*(t-n)})=1$,

$$
\begin{aligned}
&\sum_{n=1}^{N(t)}\widehat{W}\left(t+1,t,t-n\right)=\sum_{n=1}^{N(t)}\widetilde{W}(t+1,t)\widetilde{W}(t,t-n)+\sum_{n=1}^{N(t)}\widetilde{W}(t+1,t-n) \\
=\;& \widetilde{W}(t+1,t)\sum_{n=1}^{N(t)}\widetilde{W}(t,t-n)+\sum_{n=1}^{N(t)}\widetilde{W}(t+1,t-n) \\
=\;& \beta\widetilde{W}(t+1,t)+\sum_{n=1}^{N(t)}\widetilde{W}(t+1,t-n)\leq\sum_{n=1}^{N(t)+1}\widetilde{W}(t+1,t+1-n)
\end{aligned}
$$

Since $\widetilde{W}(t+1,t+1-n)=0$ for any $n>N(t+1)$,

$$
\begin{aligned}
\sum_{n=1}^{N(t)+1}\widetilde{W}\left(t+1,t+1-n\right) &= \sum_{n=1}^{N(t+1)}\widetilde{W}(t+1,t+1-n) \\
&= \beta\sum_{n=1}^{N(t+1)}W_{N(t+1),h}(\theta^{(t+1)},\theta^{*(t+1-n)})=\beta
\end{aligned}
$$

Thus,

$$
\sum_{n=1}^{N(t)}\widehat{W}\left(t+1,t,t-n\right)\leq\beta
$$

Hence, inequality $A17$ holds for $\underline{N}=1$.

Next, suppose that inequality $A16$ holds for $\underline{N}=M$. Then, using $t+1$ instead of t in inequality $A16$, we get

$$
\begin{aligned}
& \left|V(s,\epsilon,\theta) - V^{(t+1+M)}(s,\epsilon,\theta)\right| \\
\leq\ & \sup_{s' \in S}\left|A_1^{(t+1+M)}(\theta)\right| \\
& + \sum_{m=0}^{M-1} \widehat{W}(t+1+M, t+1+M-m, t+M-m) \sup_{s' \in S}\left|A^{(t+M-m)}\left(\theta^{*(t+M-m)}\right)\right| \\
& + \sup_{\widehat{s} \in S}\left|V(\widehat{s},\epsilon^{(t)},\theta^{*(t)}) - V^{(t)}(\widehat{s},\epsilon^{(t)},\theta^{*(t)})\right| \widehat{W}(t+1+M, t+1, t) \\
& + \sum_{n=2}^{N(t+1)} \sup_{\widehat{s} \in S}\left|V(\widehat{s},\epsilon^{(t+1-n)},\theta^{*(t+1-n)}) - V^{(t+1-n)}(\widehat{s},\epsilon^{(t+1-n)},\theta^{*(t+1-n)})\right| \\
& \widehat{W}(t+1+M, t+1, t+1-n).
\end{aligned}
$$

Now, using $A15\prime$ to substitute away $\sup_{\widehat{s} \in S}\left|V(\widehat{s},\epsilon^{(t)},\theta^{*(t)}) - V^{(t)}(\widehat{s},\epsilon^{(t)},\theta^{*(t)})\right|$, we get

$$
\begin{aligned}
& \left|V(s,\epsilon,\theta) - V^{(t+M+1)}(s,\epsilon,\theta)\right| \\
\leq\ & \sup_{s' \in S}\left|A_1^{(t+M+1)}(\theta)\right| \\
& + \sum_{m=0}^{M} \widehat{W}(t+M+1, t+M+1-m, t+M-m) \sup_{s' \in S}\left|A_1^{(t+M-m)}\left(\theta^{*(t+M-m)}\right)\right| \\
& + \sum_{n=1}^{N(t)} \sup_{\widehat{s} \in S}\left|V(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)}) - V^{(t-n)}(\widehat{s},\epsilon^{(t-n)},\theta^{*(t-n)})\right| \\
& \left[\widehat{W}(t+M+1, t+1, t)\widetilde{W}(t, t-n) + \widehat{W}(t+M+1, t+1, t-n)\right] \qquad \text{(A18)}
\end{aligned}
$$

Now, we claim that, for any $M \geq 1$,

$$
\begin{aligned}
& \widehat{W}(t+M, t+1, t)\,\widetilde{W}(t, t-n) + \widehat{W}(t+M, t+1, t-n) \\
=\ & \widehat{W}(t+M, t, t-n) \qquad \text{(A19)}
\end{aligned}
$$

Proof of the Claim:

Let

$$
\begin{aligned}
& \Psi_{m,1}(t+M, t, \tau) \\
\equiv\ & \left\{J_m = (t_m, t_{m-1}, ..., t_1, t_0) : t_m = t+M > t_{m-1} > .... > t_2 \geq t+1, t_1 = t, t_0 = \tau\right\}.
\end{aligned}
$$

Notice that

$$
\begin{aligned}
& \Psi_m(t+M, t+1, \tau) \\
\equiv\ & \left\{J_m = (t_m, t_{m-1}, ..., t_1, t_0) : t_m = t+M > t_{m-1} > .... > t_2 > t_1 \geq t+1, t_0 = \tau\right\}.
\end{aligned}
$$

Then,

$$\Psi_m(t+M,t,\tau) = \Psi_{m,1}(t+M,t,\tau) \cup \Psi_m(t+M,t+1,\tau)$$

and

$$\Psi_{m,1}(t+M,t,\tau) \cap \Psi_m(t+M,t+1,\tau) = \varnothing.$$

Also,

$$\Psi_{M+1}(t+M,t+1,\tau) = \varnothing$$

Therefore,

$$\begin{aligned}
& \widehat{W}(t+M,t,\tau) \\
\equiv\; & \sum_{m=1}^{M+1} \left\{ \sum_{\Psi_m(t+M,t,\tau)} \prod_{k=1}^{m} \widetilde{W}(t_k,t_{k-1}) \right\} \\
=\; & \sum_{m=1}^{M+1} \left\{ \sum_{\Psi_{m,1}(t+M,t,\tau)} \prod_{k=1}^{m} \widetilde{W}(t_k,t_{k-1}) \right\} + \sum_{m=1}^{M+1} \left\{ \sum_{\Psi_m(t+M,t+1,\tau)} \prod_{k=1}^{m} \widetilde{W}(t_k,t_{k-1}) \right\} \\
=\; & \sum_{m=2}^{M+1} \left\{ \sum_{\Psi_{m-1}(t+M,t+1,t)} \prod_{k=1}^{m-1} \widetilde{W}(t_k,t_{k-1}) \right\} \widetilde{W}(t,\tau) \\
& + \sum_{m=1}^{M} \left\{ \sum_{\Psi_m(t+M,t+1,\tau)} \prod_{k=1}^{m} \widetilde{W}(t_k,t_{k-1}) \right\} \\
=\; & \sum_{m=1}^{M} \left\{ \sum_{\Psi_m(t+M,t+1,t)} \prod_{k=1}^{m} \widetilde{W}(t_k,t_{k-1}) \right\} \widetilde{W}(t,\tau) + \sum_{m=1}^{M} \left\{ \sum_{\Psi_m(t+M,t+1,\tau)} \prod_{k=1}^{m} \widetilde{W}(t_k,t_{k-1}) \right\} \\
=\; & \widehat{W}(t+M,t+1,t)\,\widetilde{W}(t,\tau) + \widehat{W}(t+M,t+1,\tau)
\end{aligned}$$

Hence, the claim holds. Substituting this into equation $A18$ yields the first part of the lemma by induction.

Next, suppose that $A17$ holds for $\underline{N} = M$. That is,

$$\sum_{n=1}^{N(t)} \widehat{W}(t+M,t,t-n) \leq \beta.$$

Then, denoting $t' = t+1$, we get

$$
\begin{aligned}
&\sum_{n=1}^{N(t)} \widehat{W}(t+M+1,t,t-n) \\
= &\sum_{n=1}^{N(t)} \widehat{W}\left(t+M+1,t+1,t\right)\widetilde{W}(t,t-n) + \sum_{n=1}^{N(t)} \widehat{W}(t+M+1,t+1,t-n) \\
\leq &\ \widehat{W}\left(t'+M,t',t\right) + \sum_{n=1}^{N(t)} \widehat{W}(t'+M,t',t-n) \\
= &\sum_{n=1}^{N(t')} \widehat{W}(t'+M,t',t'-n) \leq \beta
\end{aligned}
$$

Hence, induction holds and for any $\underline{N} > 0$,

$$
\sum_{n=1}^{N(t)} \widehat{W}(t+\underline{N},t,t-n) \leq \beta
$$

Therefore, from induction, Lemma 5 holds.

Now, for any $m = 1,...\widetilde{N}(l)$, if we substitute $t(l)-m$ for $t+\underline{N}$, $t(l-1)$ for t, then equation $A16$ becomes

$$
\begin{aligned}
&\left|V\left(s,\epsilon^{(t(l)-m)},\theta^{*(t(l)-m)}\right) - V^{(t(l)-m)}\left(s,\epsilon^{(t(l)-m)},\theta^{*(t(l)-m)}\right)\right| \\
\leq &\sup_{s'\in S}\left|A_1^{(t(l)-m)}\left(\theta^{*(t(l)-m)}\right)\right| \\
&+ \sum_{i=0}^{\widetilde{N}(l)-m-1} \widehat{W}\left(t(l)-m,t(l)-m-i,t(l)-m-i-1\right)\sup_{s'\in S}\left|A_1^{(t(l)-m-i-1)}\right| \\
&+ \sum_{n=1}^{\widetilde{N}(l-1)} \sup_{\widehat{s}\in S}\left|V(\widehat{s},\epsilon^{(t(l-1)-n)},\theta^{*(t(l-1)-n)}) - V^{(t(l-1)-n)}(\widehat{s},\epsilon^{(t(l-1)-n)},\theta^{*(t(l-1)-n)})\right| \\
&\widehat{W}(t(l)-m,t(l-1),t(l-1)-n)
\end{aligned}
$$

Now, we take weighted sum of $\left|V\left(s,\epsilon,\theta^{*(t(l)-m)}\right) - V^{(t(l)-m)}\left(s,\epsilon,\theta^{*(t(l)-m)}\right)\right|$, $m = 1,...\widetilde{N}(l)$, where the weights are defined to be $W^{\#}\left(l,t(l)-m\right)$. These weights satisfy $W^{\#}\left(l,t_l\right) > 0$ for t_l such that $t(l-1) \leq t_l < t(l)$ and 0 otherwise, and

$$
\sum_{t(l-1)\leq t_l < t(l)} W^{\#}\left(l,t_l\right) = 1 \tag{A20}
$$

Then,

$$\begin{aligned}
&\sum_{m=1}^{\widetilde{N}(l)} \left| V\left(s, \epsilon^{(t(l)-m)}, \theta^{*(t(l)-m)}\right) - V^{(t(l)-m)}\left(s, \epsilon^{(t(l)-m)}, \theta^{*(t(l)-m)}\right) \right| W^{\#}\left(l, t(l)-m\right) \\
\leq\ & \sum_{m=1}^{\widetilde{N}(l)} \left\{ \sup_{s' \in S} \left| A_1^{(t(l)-m)} \right| \right. \\
&+ \sum_{i=0}^{\widetilde{N}(l)-m-1} \widehat{W}\left(t(l)-m, t(l)-m-i, t(l)-m-i-1\right) \\
&\left. \sup_{s' \in S} \left| A_1^{(t(l)-m-i-1)} \right| \right\} W^{\#}\left(l, t(l)-m\right) \\
&+ \sum_{m=1}^{\widetilde{N}(l)} \sum_{n=1}^{\widetilde{N}(l-1)} \sup_{\widehat{s} \in S} \left| V(\widehat{s}, \epsilon^{(t(l-1)-n)}, \theta^{*(t(l-1)-n)}) - V^{(t(l-1)-n)}(\widehat{s}, \epsilon^{(t(l-1)-n)}, \theta^{*(t(l-1)-n)}) \right| \\
&\widehat{W}(t(l)-m, t(l-1), t(l-1)-n) W^{\#}\left(l, t(l)-m\right) \qquad \text{(A21)}
\end{aligned}$$

Now, let,

$$B_1(l,l) = \sum_{m=1}^{\widetilde{N}(l)} \sup \left| A_1^{(t(l)-m)} \right| W^{\#}\left(l, t(l)-m\right),$$

$$\begin{aligned}
B_2(l,l) \ \equiv\ & \sum_{m=1}^{\widetilde{N}(l)} W^{\#}\left(l, t(l)-m\right) \\
& \times \sum_{j=0}^{\widetilde{N}(l)-m-1} \left\{ \widehat{W}\left(t(l)-m, t(l)-m-j, t(l)-m-j-1\right) \sup \left| A^{(t(l)-m-j-1)} \right| \right\}
\end{aligned}$$

and,

$$A(l,l) \equiv B_1(l,l) + B_2(l,l).$$

Lemma 6

$$A(l,l) \xrightarrow{P} 0 \quad \text{as} \quad l \to \infty.$$

Proof: We first show that $B_1(l,l) \xrightarrow{P} 0$. Recall that

$$A_1^{(t)}(\theta) = \beta \left[\int V(s', \epsilon', \theta) dF_{\epsilon'}(\epsilon', \theta) - \sum_{n=1}^{N(t)} V(s', \epsilon^{(t-n)}, \theta) W_{N(t),h}(\theta, \theta^{*(t-n)}) \right]$$

$$+\beta \left[\sum_{n=1}^{N(t)} \left[V(s', \epsilon^{(t-n)}, \theta) - V(s', \epsilon^{(t-n)}, \theta^{(t-n)}) \right] W_{N(t),h}(\theta, \theta^{*(t-n)}) \right]$$

Because $\int V(s',\epsilon',\theta)dF_{\epsilon'}(\epsilon',\theta)$, and $V(s',\epsilon^{(t-n)},\theta)$ are uniformly bounded and the parameter space is compact, $A_1^{(t)}$ is uniformly bounded. Hence, there exists $\overline{A}>0$ such that $A_1^{(t)}\leq\overline{A}$ for any t. Because $A_1^{(t)}\xrightarrow{P}0$ uniformly over Θ, for any $\eta_1>0$, $\eta_2>0$, there exists T such that for any $t>T$,

$$\sup_{\theta\in\Theta}\Pr\left[\sup_{s'\in S}\left|A_1^{(t)}(\theta)\right|<\eta_1\right]>1-\eta_2$$

Therefore,

$$\begin{aligned}
E\left[\sup_{s'\in S,\theta\in\Theta}\left|A_1^{(t)}(\theta)\right|\right] &\leq \eta_1\Pr\left[\sup_{s'\in S}\left|A_1^{(t)}(\theta)\right|<\eta_1\right]+\overline{A}\Pr\left[\sup_{s'\in S,\theta\in\Theta}\left|A_1^{(t)}(\theta)\right|\geq\eta_1\right]\\
&\leq \eta_1(1-\eta_2)+\overline{A}\eta_2 \qquad \text{(A22)}
\end{aligned}$$

Hence,

$$E\left[B_1(l,l)\right]=E\left[\sum_{m=1}^{\widetilde{N}(l)}\sup\left|A_1^{(t(l)-m)}\right|W^{\#}(l,t(l)-m)\right]$$

$$\begin{aligned}
&\leq \sum_{m=1}^{\widetilde{N}(l)}W^{\#}(l,t(l)-m)\left[\eta_1(1-\eta_2)+\overline{A}\eta_2\right]\\
&= \left[\eta_1(1-\eta_2)+\overline{A}\eta_2\right]
\end{aligned}$$

Now, from Chebychev's Inequality,

$$\Pr\left[\frac{1}{\widetilde{N}(l)}\sum_{m=1}^{\widetilde{N}(l)}W^{\#}(l,t(l)-m)\sup_{s',\theta^{(t(l)-m)}\in\Theta}\left|A_1^{(t(l)-m)}\right|>\delta\right]$$

$$\leq \frac{\left[\eta_1(1-\eta_2)+\eta_2\overline{A}\right]}{\delta} \qquad \text{(A23)}$$

For any given δ, the RHS can be made arbitrarily small by choosing $\eta_1$and η_2. Thus, $B_1(l,l)\xrightarrow{P}0$ as $t\to\infty$.

We now show that

$$\begin{aligned}
&B_2(l,l)\\
&= \sum_{m=1}^{\widetilde{N}(l)}W^{\#}(l,t(l)-m)\times\\
&\quad\sum_{j=0}^{\widetilde{N}(l)-m-1}\left\{\widehat{W}(t(l)-m,t(l)-m-j,t(l)-m-j-1)\sup\left|A_1^{(t(l)-m-j-1)}\right|\right\}\xrightarrow{P}0
\end{aligned}$$

as $t\to\infty$.

For any $t' > t > 0$, let,

$$\widetilde{K}\left(t', t\right) \equiv K_h\left(\theta^{*(t')} - \theta^{*(t)}\right)$$

For $\tau_1 > \tau_2 > \tau$, define $W^*\left(\tau_1, \tau_2, \tau, j\right)$ recursively to be as follows.

$$\begin{aligned}
W^*\left(\tau_1, \tau_2, \tau, 1\right) &\equiv \widetilde{W}(\tau_1, \tau) \\
W^*\left(\tau_1, \tau_2, \tau, 2\right) &\equiv \sum_{j=1}^{\tau_1-\tau_2} \widetilde{W}(\tau_1, \tau_1 - j) W^*\left(\tau_1 - j, \tau_2, \tau, 1\right) \\
&\vdots \\
W^*\left(\tau_1, \tau_2, \tau, k\right) &\equiv \sum_{j=1}^{\tau_1-\tau_2-(k-2)} \widetilde{W}(\tau_1, \tau_1 - j) W^*\left(\tau_1 - j, \tau_2, \tau, k-1\right)
\end{aligned}$$

Notice that for $\tau < \tau_2 - N\left(\tau_2\right)$,

$$W^*\left(\tau_1, \tau_2, \tau, k\right) = 0$$

for all k. Similarly,

$$\begin{aligned}
K^*\left(\tau_1, \tau_2, \tau, 1\right) &\equiv \frac{1}{N(\tau_1)} \widetilde{K}(\tau_1, \tau) \\
K^*\left(\tau_1, \tau_2, \tau, 2\right) &\equiv \sum_{j=1}^{\tau_1-\tau_2} \frac{1}{N(\tau_1)} \widetilde{K}(\tau_1, \tau_1 - j) K^*\left(\tau_1 - j, \tau_2, \tau, 1\right) \\
&\vdots \\
K^*\left(\tau_1, \tau_2, \tau, k\right) &\equiv \sum_{j=1}^{\tau_1-\tau_2-(k-2)} \frac{1}{N(\tau_1)} \widetilde{K}(\tau_1, \tau_1 - j) K^*\left(\tau_1 - j, \tau_2, \tau, k-1\right)
\end{aligned}$$

and for $\tau < \tau_2 - N\left(\tau_2\right)$,

$$K^*\left(\tau_1, \tau_2, \tau, k\right) = 0$$

Then, for any $\tau_1 > \tau_2 > \tau$,

$$\begin{aligned}
\widehat{W}(\tau_1, \tau_2, \tau) &\equiv \sum_{m=1}^{\widetilde{N}(l)+1} \left\{ \sum_{\Psi_m(\tau_1, \tau_2, \tau)} \prod_{k=1}^{m} \widetilde{W}\left(t_k, t_{k-1}\right) \right\} \\
&= \sum_{k=1}^{\tau_1-\tau_2+1} W^*(\tau_1, \tau_2, \tau, k) \qquad \text{(A24)}
\end{aligned}$$

Hence,

$$
\begin{aligned}
& \sum_{i=0}^{\widetilde{N}(l)-m-1} \left\{ \widehat{W}\left(t(l)-m, t(l)-m-i, t(l)-m-i-1\right) \sup_{s' \in S} \left| A_1^{(t(l)-m-i-1)} \right| \right\} \\
= & \sum_{i=0}^{\widetilde{N}(l)-m-1} \left\{ \sum_{k=1}^{i+1} W^*(t(l)-m, t(l)-m-i, t(l)-m-i-1, k) \sup_{s' \in S} \left| A_1^{(t(l)-m-i-1)} \right| \right\} \\
= & \sum_{k=1}^{\widetilde{N}(l)} \left\{ \sum_{i=k-1}^{\widetilde{N}(l)-m-1} W^*(t(l)-m, t(l)-m-i, t(l)-m-i-1, k) \sup_{s' \in S} \left| A_1^{(t(l)-m-i-1)} \right| \right\}
\end{aligned}
$$

Also, notice that, for any $\widetilde{t}$ such that $t(l-1) \leq \widetilde{t} \leq t(l)$

$$
\begin{aligned}
& W^*(\widetilde{t}, \widetilde{t}-i, \widetilde{t}-i-1, k) \\
= & \sum_{\Psi_k(\widetilde{t}, \widetilde{t}-i, \widetilde{t}-i-1)} \prod_{j=1}^{k} \widetilde{W}\left(t_j, t_{j-1}\right) \\
= & \sum_{\Psi_k(\widetilde{t}, \widetilde{t}-i, \widetilde{t}-i-1)} \prod_{j=1}^{k} \beta \frac{\widetilde{K}\left(t_j, t_{j-1}\right)}{\sum_{i=1}^{N(t_j)} \widetilde{K}\left(t_j, t_{j-i}\right)} \\
\leq & \beta^k \left[\inf_{t(l-1) \leq t \leq t(l)} \sum_{i=1}^{N(t)} \widetilde{K}\left(t, t-i\right) \right]^{-k} \sum_{\Psi_k(\widetilde{t}, \widetilde{t}-i, \widetilde{t}-i-1)} \prod_{j=1}^{k} \widetilde{K}\left(t_j, t_{j-1}\right) \\
= & \beta^k \left[\frac{1}{\widetilde{N}(l)} \inf_{t(l-1) \leq t \leq t(l)} \sum_{i=1}^{N(t)} \widetilde{K}\left(t, t-i\right) \right]^{-k} \sum_{\Psi_k(\widetilde{t}, \widetilde{t}-i, \widetilde{t}-i-1)} \prod_{j=1}^{k} \frac{\widetilde{K}\left(t_j, t_{j-1}\right)}{\widetilde{N}(l)} \\
= & \beta^k \left[\frac{1}{\widetilde{N}(l)} \inf_{t(l-1) \leq t \leq t(l)} \sum_{i=1}^{N(t)} \widetilde{K}\left(t, t-i\right) \right]^{-k} K^*(\widetilde{t}, \widetilde{t}-i, \widetilde{t}-i-1, k)
\end{aligned}
$$

Hence, we get

$$
\begin{aligned}
& \Pr\left[\sum_{k=1}^{\widetilde{N}(l)}\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m)\right. \\
& \sum_{i=k-1}^{\widetilde{N}(l)-m-1} W^{*}(t(l)-m,t(l)-m-i,t(l)-m-i-1,k) \\
& \left.\sup_{s'\in S}\left|A_1^{(t(l)-m-i-1)}\right| \geq \frac{\delta-\delta^{\widetilde{N}(l)+2}}{1-\delta}\right] \\
\leq\ & \Pr\left[\bigcup_{k=1}^{\widetilde{N}(l)}\left\{\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m)\right.\right. \\
& \sum_{i=k-1}^{\widetilde{N}(l)-m-1} W^{*}(t(l)-m,t(l)-m-i,t(l)-m-i-1,k) \\
& \left.\left.\sup_{s'\in S}\left|A_1^{(t(l)-m-i-1)}\right| \geq \delta^{k}\right\}\right] \\
\leq\ & \sum_{k=1}^{\widetilde{N}(l)} \Pr\left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(t(l),t(l)-m)\right. \\
& \sum_{i=k-1}^{\widetilde{N}(l)-m-1} W^{*}(t(l)-m,t(l)-m-i,t(l)-m-i-1,k) \\
& \left.\sup_{s'\in S}\left|A_1^{(t(l)-m-i-1)}\right| \geq \delta^{k}\right] \\
\leq\ & \sum_{k=1}^{\widetilde{N}(l)} \Pr\left\{\left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(t(l),t(l)-m)\right.\right. \\
& \sum_{i=k-1}^{\widetilde{N}(l)-m-1} K^{*}(t(l)-m,t(l)-m-i,t(l)-m-i-1,k) \\
& \left.\sup_{s'\in S}\left|A_1^{(t(l)-m-i-1)}\right| \geq \left[\frac{\delta}{4A\beta}\varepsilon_0 \inf_{\theta} g\left(\theta\right)\right]^{k}\right] \\
& \left.\bigcup\left[\inf_{t(l-1)\leq t\leq t(l)}\left[\frac{1}{\widetilde{N}(l)}\sum_{i=1}^{N(t)}\widetilde{K}(t,t-i)\right] < \frac{1}{4A}\varepsilon_0 \inf_{\theta} g\left(\theta\right)\right]\right\}
\end{aligned}
$$

$$
\begin{aligned}
\leq \; & \sum_{k=1}^{\widetilde{N}(l)} \Pr \left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(t(l), t(l)-m) \right. \\
& \sum_{i=k-1}^{\widetilde{N}(l)-m-1} K^{*}(t(l)-m, t(l)-m-i, t(l)-m-i-1, k) \\
& \left. \sup_{s' \in S} \left| A_1^{(t(l)-m-i-1)} \right| \geq \left[\frac{\delta}{4A\beta} \varepsilon_0 \inf_{\theta} g(\theta) \right]^k \right] \\
& + \Pr \left[\inf_{t(l-1) \leq t \leq t(l)} \left[\frac{1}{\widetilde{N}(l)} \sum_{i=1}^{N(t)} \widetilde{K}(t, t-i) \right] < \frac{1}{4A} \varepsilon_0 \inf_{\theta} g(\theta) \right] \qquad \text{(A25)}
\end{aligned}
$$

First, we consider the first term of the RHS of equation $A25$.

Claim 1: The following inequality holds.

$$
\begin{aligned}
& E \left\{ \sum_{i=k-1}^{\widetilde{N}(l)-m-1} K^{*}(t(l)-m, t(l)-m-i, t(l)-m-i-1, k) \right\} \\
\leq \; & \varepsilon_1^{k+1} \left\{ \sup_{\theta' \in \Theta} E_{\theta} \left[K_h \left(\theta' - \theta(\widetilde{g}) \right) \right] \right\}^k \frac{1}{(k-1)!} \qquad \text{(A26)}
\end{aligned}
$$

Proof: First, by definition of K^{*}, note that,

$$
\begin{aligned}
& E \left\{ \sum_{i=k-1}^{\widetilde{N}(l)-m-1} K^{*}(t(l)-m, t(l)-m-i, t(l)-m-i-1, k) \right\} \\
= \; & \frac{1}{\widetilde{N}(l)^k} \sum_{i=k-1}^{\widetilde{N}(l)-m-1} \sum_{j_1, \ldots, j_{k-1}} I \left(j_0 = t(l)-m-i-1, \right. \\
& \left. t(l)-m-i \leq j_1 < j_2 < \ldots < j_k = t(l)-m \right) \\
& E \left[\left\{ \prod_{s=0}^{k-1} \left[K_h \left(\theta^{*(j_{s+1})} - \theta^{*j_s} \right) \right] \right\} \right] \qquad \text{(A27)}
\end{aligned}
$$

Because $\theta'(\widetilde{g})$ and $\theta(\widetilde{g})$ are assumed to be independent,

$$
\begin{aligned}
E_{\theta', \theta} \left[K_h \left(\theta'(\widetilde{g}) - \theta(\widetilde{g}) \right) \right] &= E_{\theta'} \left[E_{\theta} \left\{ K_h \left(\theta'(\widetilde{g}) - \theta(\widetilde{g}) \right) \right\} \right] \\
&\leqq \sup_{\widetilde{\theta} \in \Theta} E_{\theta} \left[K_h \left(\widetilde{\theta} - \theta(\widetilde{g}) \right) \right] \qquad \text{(A28)}
\end{aligned}
$$

Now, for $k \geq 1$, let $(j_0, j_1, \ldots, j_k)$ satisfy $t(l)-m-i-1 = j_0 < j_1 < j_2 < \ldots < j_{k-1} < j_k = t(l)-m$. Now, denote the conditional transition probability from $\theta^{*(t)}$ to

$\theta^{*(t+1)}$ given $\Omega^{(t)}$ as $f^*\left(\theta^{*(t)},\theta^{*(t+1)}|\Omega^{(t)}\right)$, or , in shorthand, $f^{*(t+1)}$. Notice that from Lemma 2, for any l,

$$\left\{\prod_{s=2}^{t(l)-m} f^*\left(\theta^{*(s-1)},\theta^{*(s)}|\Omega^{(s-1)}\right)\right\}$$

$$\leq \left[\prod_{m=0}^{k-1}\varepsilon_1\widetilde{g}(\theta^{*(j_m)})\right]$$

$$\left\{\prod_{s=2}^{t(l)-m}\left[f^*\left(\theta^{*(s-1)},\theta^{*(s)}|\Omega^{(s-1)}\right)1\left(s\neq\{j_m\}_{m=0}^{k-1}\right)+1\left(s\neq\{j_m\}_{m=0}^{k-1}\right)\right]\right\} \tag{A29}$$

Because $K_h() \geq 0$, for any $0<t<t'$

$$\begin{aligned}E\left[K_h(\theta^{*(t')}-\theta^{*(t)})\right] &= E\left[K_h\left(\theta^{*(t')}(f^{*(t')})-\theta^{*(t)}(f^{*(t)})\right)\right]\\ &\leq \varepsilon_1^2 E\left\{E\left[K_h\left(\theta^{*(t')}(\widetilde{g})-\theta^{*(t)}(\widetilde{g})\right)\right]\right\}.\end{aligned}$$

By $A28$ and $A29$,

$$\begin{aligned}& E\left[\prod_{i=0}^{k-1}\left[K_h\left(\theta^{*(j_{i+1})}(f^{*(j_{i+1})})-\theta^{*(j_i)}(f^{*(j_i)})\right)\right]|\Omega^{(j_0)}\right]\\ \leq\;& \varepsilon_1^{k+1}E\left[\prod_{i=0}^{k-1}\left[K_h\left(\theta^{*(j_{i+1})}(\widetilde{g})-\theta^{*(j_i)}(\widetilde{g})\right)\right]\right]\\ \leq\;& \varepsilon_1^{k+1}E\left[\prod_{i=0}^{k-1}\sup_{\theta'\in\Theta}\left[K_h\left(\theta'-\theta^{*(j_i)}(\widetilde{g})\right)\right]\right]\\ =\;& \varepsilon_1^{k+1}\left\{\sup_{\theta'\in\Theta}E_\theta\left[K_h\left(\theta'-\theta^*(\widetilde{g})\right)\right]\right\}^k\end{aligned} \tag{A30}$$

Furthermore, for any i,m such that $0<m+i\leq\widetilde{N}(l)$ and for any $k>1$ such that $k\leq m+i$,

$$\begin{aligned}& \frac{1}{\widetilde{N}(l)^{k-1}}\sum_{j_1,\ldots,j_{k-1}} I(t(l)-m-i\leq j_1<\ldots<j_{k-1}<t(l)-m)\\ =\;& \frac{1}{\widetilde{N}(l)^{k-1}}\left(\frac{[i]!}{(k-1)!(i-(k-1))!}\right)\\ \leq\;& \frac{\left[[i]/\widetilde{N}(l)\right]^{k-1}}{(k-1)!}\leq\frac{1}{(k-1)!}\end{aligned} \tag{A31}$$

Substituting $A30$ and $A31$ into $A27$, $A26$ follows and hence Claim 1 is proved.

Now, by $\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m) = 1$, the law of iterated expectations and the results obtained in $A22$ and $A25$,

$$E\left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m) \sum_{i=k-1}^{\widetilde{N}(l)-m-1} K^{*}(t(l)-m,t(l)-m-i,t(l)-m-i-1,k)\right.$$

$$\left.\sup_{s'\in S}\left|A_1^{(t(l)-m-i-1)}\right|\right]$$

$$= E\left\{\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m)\right.$$

$$E\left[\sum_{i=k-1}^{\widetilde{N}(l)-m-1} K^{*}(t(l)-m,t(l)-m-i,t(l)-m-i-1,k)|\Omega^{(t(l)-m-i-1)}\right]$$

$$\left.\sup_{s',\theta}\left|A_1^{(t(l)-m-i-1)}(\theta)\right|\right\}$$

$$\leq \sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m)\left[\varepsilon_1^{k+1}\sup_{\theta'\in\Theta} E_\theta\left[K_h\left(\theta'-\theta(\widetilde{g})\right)\right]^k \frac{1}{(k-1)!}\right]\left[\eta_1(1-\eta_2)+\eta_2\overline{A}\right]$$

$$= \left[\varepsilon_1^{k+1}\sup_{\theta'\in\Theta} E_\theta\left[K_h\left(\theta'-\theta(\widetilde{g})\right)\right]^k \frac{1}{(k-1)!}\right]\left[\eta_1(1-\eta_2)+\eta_2\overline{A}\right]$$

Chebychev Inequality implies,

$$\Pr\left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m) \sum_{i=k-1}^{\widetilde{N}(l)-m-1} K^{*}(t(l)-m,t(l)-m-i,t(l)-m-i-1,k)\right.$$

$$\left.\sup_{s'\in S}\left|A_1^{(t(l)-m-i-1)}\right| > \left[\frac{\delta}{4A\beta}\varepsilon_0\inf_\theta g(\theta)\right]^k\right]$$

$$\leq \frac{\left[\eta_1(1-\eta_2)+\eta_2\overline{A}\right]\varepsilon_1^{k+1}\sup_{\theta'\in\Theta} E\left[K_h\left(\theta'-\theta(\widetilde{g})\right)\right]^k \frac{1}{(k-1)!}}{\left[\frac{\delta}{4A\beta}\varepsilon_0\inf_\theta g(\theta)\right]^k} \tag{A32}$$

Next, we consider the second term of the RHS of equation A25.

Claim 2: For any $t(l-1)\leq t\leq t(l)$, either $\left[t(l-1)-\widetilde{N}(l-1)/2,t(l-1)\right]\subseteq [t-N(t),t]$ or $\left[t(l-1),t(l-1)+\widetilde{N}(l-1)/2\right]\subseteq[t-N(t),t]$ or both.

Proof: First, we show that for t satistying $t(l-1)\leq t\leq t(l-1)+\widetilde{N}(l-1)/2$,

$$\left[t(l-1)-\widetilde{N}(l-1)/2,t(l-1)\right]\subseteq[t-N(t),t] \tag{A33}$$

Because $N()$ is a nondecreasing function, $N(t) \geq \widetilde{N}(l-1)$. Hence,

$$t - t(l-1) \leq \widetilde{N}(l-1)/2 = \widetilde{N}(l-1) - \widetilde{N}(l-1)/2 \leq N(t) - \widetilde{N}(l-1)/2$$

Thus,

$$t - N(t) \leq t(l-1) - \widetilde{N}(l-1)/2$$

Since $t(l-1) \leq t$, $A33$ holds.

Next, we show that for t satisfying $t(l-1) + \widetilde{N}(l-1)/2 < t \leq t(l)$,

$$\left[t(l-1), t(l-1) + \widetilde{N}(l-1)/2\right] \subseteq [t - N(t), t]\,. \tag{A34}$$

From the definition of $\widetilde{N}()$,

$$t(l) - \widetilde{N}(l) = t(l-1)$$

Furthermore, because $N(s)$ is increasing at most by one with unit increase in s, $s - N(s)$ is nondecreasing in s. Hence,

$$t - N(t) \leq t(l) - \widetilde{N}(l) = t(l-1).$$

Furthermore, $t > t(l-1) + \widetilde{N}(l-1)/2$. Therefore, $A34$ holds. Hence, Claim 2 is proved.

Now, from $A6$, we know that for any $\eta_3 > 0$, there exists L such that for any $l > L$, $t_1 = t(l-1)$ and for $t_2 = t(l-1) + \widetilde{N}(l-1)/2$,

$$\Pr\left[\frac{1}{\widetilde{N}(l)/2}\sum_{k=1}^{\widetilde{N}(l)/2} K_h(\theta - \theta^{*(t_i-k)}) \leq \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g\left(\theta\right)\right] \leq \eta_3,\ \ i = 1, 2$$

Therefore,

$$\begin{aligned}
& \Pr\left[\left\{\frac{1}{\widetilde{N}(l)/2}\sum_{k=1}^{\widetilde{N}(l)/2} K_h(\theta - \theta^{*(t_1-k)}) \leq \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g\left(\theta\right)\right\} \bigcup \right. \\
& \left. \left\{\frac{1}{\widetilde{N}(l)/2}\sum_{k=1}^{\widetilde{N}(l)/2} K_h(\theta - \theta^{*(t_2-k)}) \leq \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g\left(\theta\right)\right\}\right] \\
\leq\ & \Pr\left[\frac{1}{\widetilde{N}(l)/2}\sum_{k=1}^{\widetilde{N}(l)/2} K_h(\theta - \theta^{*(t_1-k)}) \leq \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g\left(\theta\right)\right] \\
& + \Pr\left[\frac{1}{\widetilde{N}(l)/2}\sum_{k=1}^{\widetilde{N}(l)/2} K_h(\theta - \theta^{*(t_2-k)}) \leq \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g\left(\theta\right)\right] \\
\leq\ & 2\eta_3
\end{aligned}$$

Therefore,

$$\Pr\left[\left\{\frac{1}{\widetilde{N}(l)/2}\sum_{k=1}^{\widetilde{N}(l)/2} K_h(\theta-\theta^{*(t_1-k)}) > \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g(\theta)\right\}\bigcap\right.$$
$$\left.\left\{\frac{1}{\widetilde{N}(l)/2}\sum_{k=1}^{\widetilde{N}(l)/2} K_h(\theta-\theta^{*(t_2-k)}) > \frac{1}{2}\varepsilon_0 \inf_{\theta\in\Theta} g(\theta)\right\}\right] > 1-2\eta_3$$

Now, from Claim 2, for any t such that $t(l-1)\leq t\leq t(l)$,

$$\frac{1}{\widetilde{N}(l)}\sum_{n=1}^{N(t)} K_h(\theta-\theta^{*(t-k)}) \geq \frac{\widetilde{N}(l-1)/2}{\widetilde{N}(l)}\frac{1}{\widetilde{N}(l-1)/2}\sum_{k=1}^{\widetilde{N}(l-1)/2} K_h(\theta-\theta^{*(s-k)}) \qquad (A35)$$

where either $s=t_1=t(l-1)$ or $s=t_2=t(l-1)+\widetilde{N}(l-1)/2$ or both. Furthermore, notice that $\frac{\widetilde{N}(l-1)/2}{\widetilde{N}(l)}\geq\frac{1}{2A}$. Therefore,

$$\Pr\left[\inf_{t(l-1)\leq t\leq t(l)}\frac{1}{\widetilde{N}(l)}\sum_{n=1}^{N(t)} K_h(\theta-\theta^{*(t-n)}) \geq \frac{1}{4A}\varepsilon_0 \inf_{\theta\in\Theta} g(\theta)\right]$$
$$> \; 1-2\eta_3$$

Thus,

$$\Pr\left[\inf_{t(l-1)\leq t\leq t(l)}\left[\frac{1}{\widetilde{N}(l)}\sum_{n=1}^{N(t)} \widetilde{K}(t,t-n)\right] < \frac{1}{4A}\varepsilon_0 \inf_{\theta} g(\theta)\right] \leq 2\eta_3 \qquad (A36)$$

By $A32$ and $A36$,

$$\begin{aligned}
& RHS \text{ of } A25 \\
\leq\; & \sum_{k=1}^{\widetilde{N}(l)} \frac{\left[\eta_1(1-\eta_2)+\eta_2\overline{A}\right]\varepsilon_1^{k+1}\sup_{\theta'\in\Theta} E_\theta\left[K_h(\theta'-\theta(\widetilde{g}))\right]^k \frac{1}{(k-1)!}}{\left[\frac{\delta}{4A\beta}\varepsilon_0\inf_\theta g(\theta)\right]^k} + 2\eta_3 \\
=\; & \varepsilon_1\left[\eta_1(1-\eta_2)+\eta_2\overline{A}\right]e^{\lambda}\lambda\sum_{k=1}^{\widetilde{N}(l)}\left[e^{-\lambda}\frac{\lambda^{(k-1)}}{(k-1)!}\right] + 2\eta_3
\end{aligned}$$

where,

$$\lambda = \frac{4A\beta\varepsilon_1\sup_{\theta'\in\Theta} E_\theta\left[K_h(\theta'-\theta(\widetilde{g}))\right]}{\delta\varepsilon_0\inf_\theta g(\theta)} > 0$$

Notice that $e^{-\lambda}\frac{\lambda^k}{k!}$ is the formula for the distribution function of the Poisson distribution. Hence,

$$\sum_{k=1}^{\widetilde{N}(l)} e^{-\lambda}\frac{\lambda^{(k-1)}}{(k-1)!} \leq \sum_{k=1}^{\infty} e^{-\lambda}\frac{\lambda^{(k-1)}}{(k-1)!} = 1$$

Together, we have shown that,

$$\begin{aligned} & LHS \text{ of } A25 \\ \leq \ & \varepsilon_1 \left[\eta_1 (1-\eta_2) + \eta_2 \overline{A}\right] \lambda \exp(\lambda) + 2\eta_3 \end{aligned} \tag{A37}$$

Now,

$$E_\theta \left\{K_h(\theta', \theta(\widetilde{g}))\right\} \rightarrow \widetilde{g}(\theta') \text{ as } h \rightarrow 0.$$

Hence, for any $B > \sup_{\theta \in \Theta} [\widetilde{g}(\theta)]$, there exists $H > 0$ such that for any positive $h < H$,

$$E_\theta \left\{K_h(\theta', \theta(\widetilde{g}))\right\} < B$$

Therefore, for $h < H$, λ is uniformly bounded. Hence, the RHS of $A37$ can be made arbitrarily small by choosing η_1, η_2 and η_3 small enough.

Thus, Lemma 6 is proved. That is, we have shown that

$$A(l, l) \rightarrow 0 \text{ as } l \rightarrow \infty$$

Let

$$\begin{aligned} & \Xi(l, l_1 + 1) \\ \equiv \ & \left\{(t_l, t_{l-1}, ..., t_{l_1+1}) : t(l_1) \leq t_{l_1+1} < t(l_1+1), ..., t_{l-1} \leq t(l-1) \leq t_l < t(l)\right\}. \end{aligned}$$

Now, define, $\overrightarrow{W}(t(l), t(l_1), t_{l_1})$ as follows: For $l_1 = l$,

$$\overrightarrow{W}(t(l), t(l), t_l) \equiv W^{\#}(l, t_l).$$

For $l_1 = l - 1$,

$$\begin{aligned} & \overrightarrow{W}(t(l), t(l-1), t_{l-1}) \\ = \ & \sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l, t(l) - m) \widehat{W}(t(l) - m, t(l-1), t_{l-1}). \end{aligned}$$

For $l_1 \leq l - 2$,

$$\begin{aligned} & \overrightarrow{W}(t(l), t(l_1), t_{l_1}) \\ \equiv \ & \sum_{(t_l, t_{t-1}, ..., t_{l_1+1}) \in \Xi(l, l_1+1)} W^{\#}(t(l), t_l) \left\{\prod_{j=l_1+1}^{l-1} \widehat{W}(t_{j+1}, t(j), t_j)\right\} \widehat{W}(t_{l_1+1}, t(l_1), t_{l_1}) \end{aligned}$$

Recursively, we can express for $l_1 < l$,

$$\overrightarrow{W}(t(l), t(l_1), t_{l_1}) = \sum_{m=1}^{\widetilde{N}(l_1+1)} \overrightarrow{W}(l, t(l_1+1), t(l_1+1) - m) \widehat{W}(t(l_1+1) - m, t(l_1), t_{l_1}).$$

Hence, $A21$ can be written as follows.

$$\begin{aligned}
& \sum_{m=1}^{\widetilde{N}(l)} \left| V\left(s, \epsilon^{(t(l)-m)}, \theta^{(t(l)-m)}\right) - V^{(t(l)-m)}\left(s, \epsilon^{(t(l)-m)}, \theta^{(t(l)-m)}\right)\right| \\
& \overrightarrow{W}(t(l), t(l), t(l)-m) \\
\leq\ & \sum_{m=1}^{\widetilde{N}(l)} \overrightarrow{W}(t(l), t(l), t(l)-m) \sup_{s' \in S} \left| A_1^{(t(l)-m)}\left(\theta^{(t(l)-m)}\right)\right| \\
& + \sum_{m=1}^{\widetilde{N}(l)} \overrightarrow{W}(t(l), t(l), t(l)-m) \\
& \times \sum_{i=0}^{N(l)-m-1} \widehat{W}(t(l)-m, t(l)-m-i, t(l)-m-i-1) \sup_{s' \in S} \left| A_1^{(t(l)-m-i-1)}\right| \\
& + \sum_{m=1}^{\widetilde{N}(l-1)} \sup_{\widehat{s} \in S} \left| V(\widehat{s}, \epsilon^{(t(l-1)-m)}, \theta^{(t(l-1)-m)}) - V^{(t(l-1)-m)}(\widehat{s}, \epsilon^{(t(l-1)-m)}, \theta^{(t(l-1)-m)})\right| \\
& \overrightarrow{W}(t(l), t(l-1), t(l-1)-m) \qquad \text{(A38)}
\end{aligned}$$

Furthermore, by Lemma 5,

$$\sum_{m=1}^{\widetilde{N}(l_1)} \widehat{W}(t_{l_1+1}, t(l_1), t(l_1)-m) \leq \beta$$

Applying these inequalities to $\overrightarrow{W}$ yields,

$$\sum_{m=1}^{\widetilde{N}(l_1)} \overrightarrow{W}(t(l), t(l_1), t(l_1)-m) \leq \beta^{(l-l_1)} \qquad \text{(A39)}$$

Now, let

$$A(l, l_1) \equiv B_1(l, l_1) + B_2(l, l_1)$$

where,

$$B_1(l, l_1) \equiv \sum_{m=1}^{\widetilde{N}(l_1)} \overrightarrow{W}(t(l), t(l_1), t(l_1)-m) \sup_{s' \in S} \left| A_1^{(t(l_1)-m)}\right|$$

and

$$\begin{aligned}
B_2(l, l_1) \equiv\ & \sum_{m=1}^{\widetilde{N}(l_1)} \overrightarrow{W}(t(l), t(l_1), t(l_1)-m) \sum_{j=0}^{N(l_1)-m-1} \\
& \left\{ \widehat{W}(t(l_1)-m, t(l_1)-m-j, t(l_1)-m-j-1) \sup_{s' \in S} \left| A_1^{(t(l_1)-m-j-1)}\right| \right\}
\end{aligned}$$

Then, for $l_1 \leq l$,

$$\begin{aligned}
&\sum_{m=1}^{\widetilde{N}(l_1)} \left| V\left(s, \epsilon^{(t(l_1)-m)}, \theta^{(t(l_1)-m)}\right) - V^{(t(l_1)-m)}\left(s, \epsilon^{(t(l_1)-m)}, \theta^{(t(l_1)-m)}\right) \right| \\
&\overrightarrow{W}\left(t(l), t(l_1), t(l_1)-m\right) \\
\leq\ & A(l, l_1) \\
&+ \sum_{m=1}^{\widetilde{N}(l_1-1)} \sup_{\widehat{s}\in S} \left| V\left(\widehat{s}, \epsilon^{(t(l_1-1)-m)}, \theta^{(t(l_1-1)-m)}\right) \right. \\
&\left. \overrightarrow{W}\left(t(l), t(l_1-1), t(l_1-1)-m\right) \right| \qquad \text{(A40)}
\end{aligned}$$

Lemma 7

Given $\Delta = l - l_1 \geq 0$

$$A(l, l-\Delta) \xrightarrow{P} 0 \ \text{ as } \ l \to \infty.$$

Proof: Lemma 6 proves it with $\Delta = 0$. By definition of $\overrightarrow{W}$,

$$\begin{aligned}
&\overrightarrow{W}(t(l), t(l_1), t(l_1)-m) \\
=\ & \left[\sum_{t(l-1)\leq t_l < t(l)} W^{\#}\left(t(l), t_l\right) \left\{ \sum_{t(l-2)\leq t_{l-1} < t(l-1)} \widehat{W}\left(t_l, t(l-1), t_{l-1}\right) \right.\right. \\
&\left.\left. \ldots \left\{ \sum_{t(l_1)\leq t_{l_1+1} < t(l_1+1)} \widehat{W}\left(t_{l_1+2}, t(l_1+1), t_{l_1+1}\right) \widehat{W}\left(t_{l_1+1}, t(l_1), t(l_1)-m\right) \right\} \right\} \right] \\
&\qquad \text{(A41)}
\end{aligned}$$

We prove convergence of $B_1(l, l_1)$. We follow steps that are similar to the proof of

Lemma 6. First, we derive

$$\begin{aligned}
& \Pr\left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m) \sum_{k=l-l_1}^{t(l)-m-t(l_1)+1} W^{*}(t(l)-m,t(l_1),t_{l_1},k)\right. \\
& \left.\sup_{s'\in S}\left|A_1^{(t_{l_1})}\right| \geq \delta^{l-l_1}\frac{1-\delta^{\widetilde{N}(l)+1}}{1-\delta}\right] \\
\leq & \sum_{k=l-l_1}^{t(l)-t(l_1)+1} \Pr\left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}(l,t(l)-m) W^{*}(t(l)-m,t(l_1),t_{l_1},k) \sup_{s'\in S'}\left|A_1^{(t_{l_1})}\right| \geq \delta^{k}\right] \\
\leq & \sum_{k=l-l_1}^{t(l)-t(l_1)+1} \Pr\left[\sum_{m=1}^{\widetilde{N}(l_1)} W^{\#}(l,t(l)-m) K^{*}(t(l)-m,t(l_1),t_{l_1},k) \sup_{s'\in S}\left|A^{(t_{l_1})}\right|\right. \\
& \left. > \left[\frac{\delta}{4A^{l+1-l_1}}\varepsilon_0 \inf_{\theta} g(\theta)\right]^{k}\right] \\
& + \Pr\left[\inf_{t(l_1-1)\leq t\leq t(l)}\left[\frac{1}{\widetilde{N}(l)}\sum_{i=1}^{N(t)}\widetilde{K}(t,t-i)\right] < \frac{1}{4A^{l+1-l_1}}\varepsilon_0 \inf_{\theta} g(\theta)\right] \qquad \text{(A42)}
\end{aligned}$$

We again use arguments similar to Claim 1 to show that,

$$\begin{aligned}
& E\left[\sum_{m=1}^{\widetilde{N}(l)} W^{\#}\left(l,t(l)-m\right) \sum_{k=l-l_1}^{t(l)-m-t(l_1)+1} K^{*}(t(l)-m,t(l_1),t_{l_1},k) \sup_{s'\in S}\left|A_1^{(t_{l_1})}\right|\right] \\
\leq & \ \varepsilon_1\left[\eta_1\left(1-\eta_2\right)+\eta_2\overline{A}\right]\lambda\exp\left(\lambda\right)
\end{aligned}$$

where

$$\lambda = \frac{4A^{l+1-l_1}\beta\varepsilon_1 \sup_{\theta'\in\Theta} E_{\theta}\left[K_h\left(\theta'-\theta(\widetilde{g})\right)\right]}{\delta\varepsilon_0 \inf_{\theta} g\left(\theta\right)} > 0.$$

Next, let $t_1(l) \equiv t(l-1)$ and $t_2(l) = t(l-1) + \widetilde{N}(l-1)/2$. Then, arguments similar to ones used in deriving equation $A35$ can be used to derive the inequality below.

$$\begin{aligned}
& \inf_{t(l_1-1)\leq t\leq t(l)}\left[\frac{1}{\widetilde{N}(l)}\sum_{i=1}^{N(t)}\widetilde{K}(t,t-i)\right] \\
\geq & \min_{l_1-1\leq \widetilde{l}<l}\left\{\frac{\widetilde{N}(\widetilde{l})/2}{\widetilde{N}(l)}\frac{1}{\widetilde{N}(\widetilde{l})/2}\min\left\{\sum_{k=1}^{\widetilde{N}(\widetilde{l})/2} K_h(\theta-\theta^{*(t_1(\widetilde{l})-k)}), \sum_{k=1}^{\widetilde{N}(\widetilde{l})/2} K_h(\theta-\theta^{*(t_2(\widetilde{l})-k)})\right\}\right\} \\
\geq & \frac{1}{2A^{l+1-l_1}}\frac{1}{\widetilde{N}(l^{*})/2}\min\left\{\sum_{k=1}^{\widetilde{N}(l^{*})/2} K_h(\theta-\theta^{*(t_1(l^{*})-k)}), \sum_{k=1}^{\widetilde{N}(l^{*})/2} K_h(\theta-\theta^{*(t_2(l^{*})-k)})\right\}
\end{aligned}$$

where,

$$l^* \equiv \arg \min_{\tilde{l}: l_1 - 1 \leq \tilde{l} < l} \left\{ \frac{1}{2A^{l+1-\tilde{l}}} \frac{1}{\widetilde{N}(\tilde{l})/2} \min \left\{ \sum_{k=1}^{\widetilde{N}(\tilde{l})/2} K_h(\theta - \theta^{*(t_1(\tilde{l})-k)}), \sum_{k=1}^{\widetilde{N}(\tilde{l})/2} K_h(\theta - \theta^{*(t_2(\tilde{l})-k)}) \right\} \right\}$$

Hence, similarly to Lemma 6, we get

$$\begin{aligned} & RHS \text{ of } A42 \\ \leq \; & \varepsilon_1 \left[\eta_1 (1 - \eta_2) + \eta_2 \overline{A} \right] e^{\lambda} \sum_{k=1}^{\infty} \left[e^{-\lambda} \frac{\lambda^k}{(k-1)!} \right] + 2(l + 1 - l_1) \eta_3 \end{aligned}$$

whose RHS can be made arbitrarily close to zero by having η_1, η_2 and η_3 arbitrarily small by choosing l to be large enough, for any arbitarily positive δ, RHS can be made arbitarrily small by increasing l, while keeping $\Delta = l - l_1$ constant,

$$B_1(l, l - \Delta) \xrightarrow{P} 0$$

as $l \to \infty$.

Next, we prove convergence of $B_2(l, l_1)$. Again, the arguments are very similar to that of Lemma 6. That is,

$$\Pr\left[\sum_{\left(t_l,t_{t-1},\ldots,t_{l_1+1}\right)\in\Xi(l,l_1+1)} W^{\#}\left(l,t_l\right)\left\{\prod_{j=l_1+1}^{l-1}\widehat{W}\left(t_{j+1},t(j),t_j\right)\right\}\widehat{W}\left(t_{l_1+1},t(l_1),t_{l_1}\right)\right.$$
$$\left.\left\{\sum_{k=1}^{\widetilde{N}(l_1)}\sum_{j=k-1}^{t_{l_1}-t(l_1-1)-1}\widehat{W}(t_{l_1},t_{l_1}-j,t_{l_1}-j-1,k)\sup_{s'\in S}\left|A_1^{(t_{l_1}-j-1)}\right|\right\}\geq\delta^{l-l_1}\frac{1-\delta^{\widetilde{N}(l)+1}}{1-\delta}\right]$$
$$\leq\ \Pr\left[\sum_{\left(t_l,t_{t-1},\ldots,t_{l_1+1}\right)\in\Xi(l,l_1+1)} W^{\#}\left(l,t_l\right)\left\{\prod_{j=l_1+1}^{l-1}\sum_k W^{*}\left(t_{j+1},t(j),t_j,k\right)\right\}\sum_k W^{*}\left(t_{l_1+1},t(l_1),t_{l_1},k\right)\right.$$
$$\left.\left\{\sum_{k=1}^{\widetilde{N}(l_1)}\sum_{j=k-1}^{t_{l_1}-t(l_1-1)-1}W^{*}(t_{l_1},t_{l_1}-j,t_{l_1}-j-1,k)\sup_{s'\in S}\left|A_1^{(t_{l_1}-j-1)}\right|\right\}\geq\delta^{l-l_1}\frac{1-\delta^{\widetilde{N}(l)+1}}{1-\delta}\right]$$
$$\leq\ \sum_{k=l-l_1}^{t(l)-t(l_1-1)}\Pr\left[\sum_{m=1}^{\widetilde{N}(l)}W^{\#}(l,t(l)-m)\sum_{j=\max\{0,k-[t(l)-t(l_1)]+m-1\}}^{t_{l_1}-t(l_1-1)-1}K^{*}(t(l)-m,t(l_1)-j,t_{l_1}-j-1,k)\right.$$
$$\left.\sup_{s'\in S}\left|A_1^{(t_{l_1}-j-1)}\right|>\left[\frac{\delta}{4A^{l+1-l_1}}\varepsilon_0\inf_{\theta}g\left(\theta\right)\right]^k\right]$$
$$+\Pr\left[\inf_{t(l_1-1)\leq t\leq t(l)}\left[\frac{1}{\widetilde{N}(l)}\sum_{i=1}^{N(t)}\widetilde{K}(t,t-i)\right]<\frac{1}{4A^{l+1-l_1}}\varepsilon_0\inf_{\theta}g\left(\theta\right)\right] \tag{A43}$$

and

$$\sum_{k=l-l_1}^{t(l)-t(l_1-1)}\Pr\left[\sum_{m=1}^{\widetilde{N}(l_1)}W^{\#}(l,t(l)-m)\sum_{j=\max\{0,k-[t(l)-t(l_1)]+m-1\}}^{t_{l_1}-t(l_1-1)-1}K^{*}(t(l)-m,t(l_1)-j,t_{l_1}-j-1,k)\right.$$
$$\left.\sup_{s'\in S}\left|A^{(t_{l_1}-j-1)}\right|>\left[\frac{\delta}{4A^{l+1-l_1}}\varepsilon_0\inf_{\theta}g\left(\theta\right)\right]^k\right]$$
$$\leq\ \varepsilon_1\left[\eta_1\left(1-\eta_2\right)+\eta_2\overline{A}\right]\lambda\exp\left(\lambda\right)$$

where,

$$\lambda=\frac{4A^{l+1-l_1}\beta\varepsilon_1\sup_{\theta'\in\Theta}E_{\theta}\left[K_h\left(\theta'-\theta(\widetilde{g})\right)\right]}{\delta\varepsilon_0\inf_{\theta}g\left(\theta\right)}>0$$

Furthermore, let $t_1(l)\equiv t(l-1)$ and $t_2(l)=t(l-1)+\widetilde{N}(l-1)/2$. Then, arguments similar to ones used in deriving equation $A35$ can be used to derive the inequality below.

$$\begin{aligned}
&\inf_{t(l_1-1)\le t\le t(l)}\left[\frac{1}{\widetilde{N}(l)}\sum_{i=1}^{N(t)}\widetilde{K}(t,t-i)\right]\\
\ge\;& \min_{l_1-1\le\widetilde{l}<l}\left\{\frac{\widetilde{N}(\widetilde{l})/2}{\widetilde{N}(l)}\frac{1}{\widetilde{N}(\widetilde{l})/2}\min\left\{\sum_{k=1}^{\widetilde{N}(\widetilde{l})/2}K_h(\theta-\theta^{*(t_1(\widetilde{l})-k)},\sum_{k=1}^{\widetilde{N}(\widetilde{l})/2}K_h(\theta-\theta^{*(t_2(\widetilde{l})-k)})\right\}\right\}\\
\ge\;& \frac{1}{2A^{l+1-l_1}}\frac{1}{\widetilde{N}(l^*)/2}\min\left\{\sum_{k=1}^{\widetilde{N}(l^*)/2}K_h(\theta-\theta^{*(t_1(l^*)-k)}),\sum_{k=1}^{\widetilde{N}(l^*)/2}K_h(\theta-\theta^{*(t_2(l^*)-k)})\right\}
\end{aligned}$$

where,

$$\begin{aligned}
l^* \equiv\;& \arg\min_{\widetilde{l}:l_1-1\le\widetilde{l}<l}\left\{\frac{1}{2A^{l+1-\widetilde{l}}}\frac{1}{\widetilde{N}(\widetilde{l})/2}\right.\\
&\left.\min\left\{\sum_{k=1}^{\widetilde{N}(\widetilde{l})/2}K_h(\theta-\theta^{*(t_1(\widetilde{l})-k)}),\sum_{k=1}^{\widetilde{N}(\widetilde{l})/2}K_h(\theta-\theta^{*(t_2(\widetilde{l})-k)})\right\}\right\}
\end{aligned}$$

Hence, using $A24$,

$$\begin{aligned}
&RHS \text{ of } A43\\
\le\;& \varepsilon_1\left[\eta_1(1-\eta_2)+\eta_2\overline{A}\right]e^{\lambda}\sum_{k=1}^{\infty}\left[e^{-\lambda}\frac{\lambda^k}{(k-1)!}\right]+2(l+1-l_1)\eta_3
\end{aligned}$$

where,

$$\lambda=\frac{4\beta A^{l+1-l_1}\varepsilon_1\sup_{\theta'\in\Theta}E\left[K_h\left(\theta'-\theta(g)\right)\right]}{\delta\varepsilon_0\inf_\theta g(\theta)}>0$$

which can be made arbitrarily close to zero by increasing l while keeping $\Delta l\equiv l-l_1$ constant. Therefore,

$$B_2(l,l-\Delta l)\xrightarrow{P}0$$

Hence, Lemma 7 holds.

Now, let,

$$\Delta V(m,n)\equiv\sup_{s\in S}\left|V(s,\epsilon^{(t(m)-n)},\theta^{*(t(m)-n)})-V^{(t(m)-n)}(s,\epsilon^{(t(m)-n)},\theta^{*(t(m)-n)})\right|$$

$$\Delta V(m)\equiv\left[\Delta V(m,1),...,\Delta V(m,\widetilde{N}(m))\right]$$

$$\overline{W}(l,k)\equiv\left[\overrightarrow{W}(l,t(l+1-k),t(l+1-k)-m)\right]_{m=1}^{\widetilde{N}(l+1-k)}$$

Then, by $A39, \overline{W}(l,k)'\iota \le \beta^{k-1}$ and from $A39$, we obtain the following.

$$\begin{aligned}\Delta V(l)'\overline{W}(l,1) &\le A(l,l) + \Delta V(l-1)'\overline{W}(l,2) \\ &\le \ldots \le \sum_{i=0}^{k-1} A(l,l-i) + \Delta V(l-k)'\overline{W}(l,k+1).\end{aligned}$$

By Lemma 7, given k, the first term on the RHS, $\sum_{i=0}^{k-1} A(l,l-i)$ converge to 0 in probability as $l \to \infty$, and since $\Delta V(l+1-k)$ is bounded and $\overline{W}(l,k)'\iota \le \beta^{k-1}$ from $A39$, the second term can be made arbitrarily small by chosing a large enough k. Therefore, $\Delta V(l)' W(l,1)$ converges to zero in probability as $l \to \infty$.

Lemma 8:

$$\left|V(s,\epsilon^{(t)},\theta^{(t)}) - V^{(t)}(s,\epsilon^{(t)},\theta^{(t)})\right| \xrightarrow{P} 0 \text{ as } t \to \infty$$

Suppose not. Then, there exists a positive δ, η and a sequence $\{t_k\}$ such that

$$\Pr\left(\left|V(s,\epsilon^{(t_k)},\theta^{(t_k)}) - V^{(t_k)}(s,\epsilon^{(t_k)},\theta^{(t_k)})\right| \ge \delta\right) > \eta. \tag{A44}$$

Set the weights $W^{\#}$ be as follows: If there is t_k such that $t(l-1) \le t_k < t(l)$, then, let

$$t^*(l) = \min_{t(l-1)\le t_k < t(l)} \{t_k\}.$$

Otherwise, let

$$t^*(l) = t(l-1).$$

Let

$$W^{\#}(t(l), t_l) = I(t_l = t^*(l))$$

Then, because $\Delta V(l)'\overline{W}(l,1) \xrightarrow{P} 0$ as $l \to \infty$,

$$\left|V(s,\epsilon^{(t^*(l))},\theta^{(t^*(l))}) - V^{(t^*(l))}(s,\epsilon^{(t^*(l))},\theta^{(t^*(l))})\right| \xrightarrow{P} 0 \text{ as } l \to \infty$$

which contradicts $A44$. Hence, Lemma 8 holds, and thus we have proved Theorem 1.

Proof of Theorem 2

We are given a random process with transition probability $f^{(t)}(.,.)$ which is

$$f^{(t)}\left(\theta^{(t)},\theta'\right) = \lambda\left(\theta^{(t)},\theta'|\Omega^{(t-1)}\right) q\left(\theta^{(t)},\theta'\right) + \left[1 - \int \lambda\left(\theta^{(t)},\theta'|\Omega^{(t-1)}\right) q\left(\theta^{(t)},\theta'\right)\right] \delta_{\theta^{(t)}}(\theta')$$

where $\delta_{\theta^{(t)}}$ is the Dirac mass at $\theta^{(t)}$. Because $\lambda\left(\theta,\theta'|\Omega^{(t-1)}\right)$ converges to $\lambda(\theta,\theta')$ uniformly in probability on $\theta,\theta' \in \Theta$, $f^{(t)}(.,.)$ converges to $f(.,.)$ in probability uniformly as $t \to \infty$. Because both $\lambda\left(\theta,.|\Omega^{(t-1)}\right)$ and $q(\theta,.)$ are uniformly positive functions for any $\theta \in \Theta$, using the results in Lemma 1, we can construct a density $g(.)$ and a constant $\varepsilon_0 > 0$ such that for any $\theta \in \Theta$,

$$\begin{aligned} f^{(t)}(\theta, .) &\geq \varepsilon_0 g(.) \\ f(\theta, .) &\geq \varepsilon_0 g(.) \end{aligned}$$

Define $\nu^{(t)}$ as follows.

$$v^{(t)}(\theta) = \min \left\{ \inf_{\theta' \in \Theta} \left\{ \frac{f^{(t)}(\theta, \theta')}{f(\theta, \theta')} \right\}, 1 \right\}$$

Then,

$$\begin{aligned} f^{(t)}(\theta, .) &\geq v^{(t)} f(\theta, .) \\ f(\theta, .) &\geq v^{(t)} f(\theta, .) \end{aligned}$$

Now, construct the following coupling scheme. Let $X^{(t)}$ be a random variable that follows the transition probability $f^{(t)}(x, .)$ given $X^{(t-1)} = x$, and $Y^{(t)}$ be a Markov process that follows the transition probability $f(y, .)$, given $Y^{(t-1)} = y$. Suppose $X^{(t)} \neq Y^{(t)}$. With probability $\varepsilon_0 > 0$, let

$$X^{(t+1)} = Y^{(t+1)} = Z^{(t+1)} \sim g(.)$$

and with probability $1 - \varepsilon_0$,

$$X^{(t+1)} \sim \frac{1}{1 - \varepsilon_0} \left[f^{(t)}\left(X^{(t)}, .\right) - \varepsilon_0 g(.) \right]$$

$$Y^{(t+1)} \sim \frac{1}{1 - \varepsilon_0} \left[f\left(Y^{(t)}, .\right) - \varepsilon g(.) \right]$$

Suppose $X^{(t)} = Y^{(t)} = Z^{(t)}$. With probability $v^{(t)}$,

$$X^{(t+1)} = Y^{(t+1)} \sim f(Z^{(t)}, .)$$

and with probability $\left(1 - v^{(t)}\right)$,

$$X^{(t+1)} \sim \frac{1}{1 - v^{(t)}} \left[f^{(t)}\left(X^{(t)}, .\right) - v^{(t)} f(Z^{(t)}, .) \right]$$

$$Y^{(t+1)} \sim \frac{1}{1 - v^{(t)}} \left[f\left(Y^{(t)}, .\right) - v^{(t)} f(Z^{(t)}, .) \right]$$

As $f^{(t)}(x, .) \xrightarrow{P} f(x, .)$ uniformly over the compact parameter set Θ, $v^{(t)}$ converges to 1 in probability. Let $w^{(t)} = 1 - v^{(t)}$. Then, $w^{(t)} \xrightarrow{P} 0$ as $t \to \infty$. Let $S^{(t)} \in \{1, 2\}$ be the state at iteration t, where state 1 is assumed to be the state in which $X^{(t)} = Y^{(t)}$, and state 2 the state in which $X^{(t)} \neq Y^{(t)}$. Then, $S^{(t)}$ follows the Markov process with the following transition matrix.

$$P = \begin{bmatrix} 1 - w^{(t)} & w^{(t)} \\ \varepsilon_0 & 1 - \varepsilon_0 \end{bmatrix}$$

Denote the unconditional probability of state 1 at time t as $\pi^{(t)}$. Then,

$$\left[\pi^{(t+1)}, 1 - \pi^{(t+1)}\right] = \left[\pi^{(t)}, 1 - \pi^{(t)}\right] \begin{bmatrix} 1 - w^{(t)} & w^{(t)} \\ \varepsilon_0 & 1 - \varepsilon_0 \end{bmatrix}$$

Hence,

$$\begin{aligned} \pi^{(t+1)} &= \pi^{(t)} \left[\left(1 - w^{(t)}\right) - \varepsilon_0\right] + \varepsilon_0 \\ &\geq \pi^{(t)} \left(1 - \varepsilon_0\right) + \varepsilon_0 - w^{(t)} \\ &\geq \pi^{(t-m)} \left(1 - \varepsilon_0\right)^{m+1} + 1 - \left(1 - \varepsilon_0\right)^{m+1} - \left[w^{(t)} + \left(1 - \varepsilon_0\right) w^{(t-1)} + ... + \left(1 - \varepsilon_0\right)^m w^{(t-m)}\right] \end{aligned}$$

We now prove that $\pi^{(t)} \xrightarrow{P} 1$.

Define W_{tm} to be

$$W_{tm} = w^{(t)} + \left(1 - \varepsilon_0\right) w^{(t-1)} + ... + \left(1 - \varepsilon_0\right)^m w^{(t-m)}$$

Because $w^{(t)} \xrightarrow{P} 0$, for any $\delta_1 > 0$, $\delta_2 > 0$, there exists $N > 0$ such that for any $t \geq N$,

$$\Pr\left[\left|w^{(t)} - 0\right| < \delta_1\right] > 1 - \delta_2$$

Now, given any $\overline{\delta}_1 > 0$, $\overline{\delta}_2 > 0$, let m be such that

$$\left(1 - \varepsilon_0\right)^m < \frac{\overline{\delta}_1}{5}$$

Also, let δ_1 satisfy $\delta_1 < \frac{\overline{\delta}_1}{5(m+1)}$, and δ_2 satisfy $\delta_2 < \frac{\overline{\delta}_2}{m+1}$. Then,

$$\begin{aligned} \Pr\left\{\left|W_{tm} - 0\right| < \frac{\overline{\delta}_1}{5}\right\} &\geq \Pr\left\{\bigcap_{j=t-m}^{t} \left|w^{(j)} - 0\right| < \delta_1\right\} \\ &= 1 - \Pr\left\{\bigcup_{j=t-m}^{t} \left|w^{(j)} - 0\right| \geq \delta_1\right\} \\ &\geq 1 - \sum_{j=t-m}^{t} \Pr\left\{\left|w^{(j)} - 0\right| \geq \delta_1\right\} \geq 1 - \overline{\delta}_2 \qquad \text{(A47)} \end{aligned}$$

Now, let $\overline{N}$ be defined as $\overline{N} = \max\{N, m\}$. Then, for each $k > \overline{N}$,

$$\begin{aligned} & \Pr\left[\left|\pi^{(t+1)} - 1\right| < \overline{\delta}_1\right] \\ \geq\ & \Pr\left[\left|\pi^{(t-m)} \left(1 - \varepsilon_0\right)^m - \left(1 - \varepsilon_0\right)^{m+1} + W_{tm}\right| < \overline{\delta}_1\right] \\ \geq\ & \Pr\left[\left|\pi^{(t-m)} \left(1 - \varepsilon_0\right)^m - \left(1 - \varepsilon_0\right)^{m+1}\right| < \frac{2\overline{\delta}_1}{5}, \left|\frac{\left(1 - P_\Xi\right)/\varepsilon_0}{1 + \left(1 - P_\Xi\right)/\varepsilon_0}\right| < \frac{\overline{\delta}_1}{5}, \left|W_{tm}\right| < \frac{\overline{\delta}_1}{5}\right] \\ =\ & \Pr\left[\left|W_{tm}\right| < \frac{\overline{\delta}_1}{5}\right] \qquad \text{(A48)} \end{aligned}$$

Last equality holds because $0 \leq \pi^{(t-m)} \leq 1$ and thus,

$$\left|\pi^{(t-m)}\left(1-\varepsilon_0\right)^m-\left(1-\varepsilon_0\right)^{m+1}\right| \leq \left|2(1-\varepsilon_0)^m\right| \leq \frac{2\overline{\delta}_1}{5}$$

From $(A47)$ and (A48), we conclude that

$$\Pr\left[\left|\pi^{(t+1)}-1\right|<\overline{\delta}_1\right] \geq 1-\overline{\delta}_2$$

Therefore, π_k converges to 1 in probability.

Therefore, for any $\delta > 0$, there exists M such that for any $t > M$,

$$\Pr\left[X^{(t)}=Y^{(t)}\right]>1-\delta$$

Since $Y^{(t)}$follows a stationary distribution, $X^{(t)}$ converges to a stationary process in probability.