2007

Bayesian inference in dynamic discrete choice models

Andriy Norets
University of Iowa

BAYESIAN INFERENCE IN DYNAMIC DISCRETE CHOICE MODELS

by

Andriy Norets

An Abstract

Of a thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Economics
in the Graduate College of
The University of Iowa

July 2007

Thesis Supervisor: Professor John Geweke

ABSTRACT

In this dissertation, I develop methods for Bayesian inference in dynamic discrete choice models (DDCMs.) Chapter 1 proposes a reliable method for Bayesian estimation of DDCMs with serially correlated unobserved state variables. Inference in these models involves computing high-dimensional integrals that are present in the solution to the dynamic program (DP) and in the likelihood function. First, the chapter shows that Markov chain Monte Carlo (MCMC) methods can handle the problem of multidimensional integration in the likelihood, which was previously considered infeasible for DDCMs with serially correlated unobservables. Second, the chapter presents an efficient algorithm for solving the DP suitable for use in conjunction with the MCMC estimation procedure. The algorithm utilizing random grids and nearest neighbor approximations iterates the Bellman equation only once for each parameter draw. The chapter evaluates the method's performance on two different DDCMs using real and artificial datasets. The experiments demonstrate that ignoring serial correlation in unobservables of DDCMs can lead to serious misspecification errors. Experiments on dynamic multinomial logit models, for which analytical integration is also possible, show that the estimation accuracy of the proposed method is good.

Chapter 2 presents a proof of the complete (and thus a.s.) uniform convergence of the DP solution approximations proposed in Chapter 1 to the true values under mild assumptions on the primitives of DDCMs. It also establishes the complete convergence of the corresponding approximated posterior expectations.

Chapter 3 proposes a method for inference in DDCMs that combines MCMC and artificial neural networks (ANN.) MCMC is intended to handle high dimensional integration in the likelihood function of richly specified DDCMs. ANNs approximate the DP solution as a function of the parameters and state variables beforehand of the estimation procedure to reduce the computational burden. Potential applications of the proposed methodology include inference in DDCMs with random coefficients, serially correlated unobservables, and dependent observations. The chapter discusses MCMC estimation of DDCMs, provides relevant background on ANNs, and derives a theoretical justification of the method. Experiments suggest that application of ANNs in the MCMC estimation of DDCMs is a promising approach.

Abstract Approved: ______________________________
Thesis Supervisor

____
Title and Department

____
Date

BAYESIAN INFERENCE IN DYNAMIC DISCRETE CHOICE MODELS

by

Andriy Norets

A thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Economics
in the Graduate College of
The University of Iowa

July 2007

Thesis Supervisor: Professor John Geweke

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

PH.D. THESIS

This is to certify that the Ph.D. thesis of

Andriy Norets

has been approved by the Examining Committee for the thesis requirement for the Doctor of Philosophy degree in Economics at the July 2007 graduation.

Thesis Committee: ______________________________
John Geweke, Thesis Supervisor

____
Charles Whiteman

____
Beth Ingram

____
Elena Pastorino

____
Luke Tierney

To my parents.

ACKNOWLEDGEMENTS

I am grateful to my committee members Charles Whiteman, Elena Pastorino, Beth Ingram, Luke Tierney, and especially my advisor, John Geweke, for guidance and helpful suggestions. Each stage of this research benefited from invaluable comments of John Geweke. My understanding of econometrics and economics were influenced by him to a great extent. Many faculty members and fellow students at the University of Iowa contributed to my learning and research. I would like to thank Marianna Kudlyak, Yaroslav Litus, Maksym Obrizan, and Latchezar Popov for their friendship, help, and encouragement throughout the years at the PhD program. I also would like to acknowledge financial support from the Economics Department graduate fellowship and the Seashore Dissertation fellowship at the University of Iowa. Finally, I am grateful to my parents and my wife, Keiko, for their love and support.

ABSTRACT

In this dissertation, I develop methods for Bayesian inference in dynamic discrete choice models (DDCMs.) Chapter 1 proposes a reliable method for Bayesian estimation of DDCMs with serially correlated unobserved state variables. Inference in these models involves computing high-dimensional integrals that are present in the solution to the dynamic program (DP) and in the likelihood function. First, the chapter shows that Markov chain Monte Carlo (MCMC) methods can handle the problem of multidimensional integration in the likelihood, which was previously considered infeasible for DDCMs with serially correlated unobservables. Second, the chapter presents an efficient algorithm for solving the DP suitable for use in conjunction with the MCMC estimation procedure. The algorithm utilizing random grids and nearest neighbor approximations iterates the Bellman equation only once for each parameter draw. The chapter evaluates the method's performance on two different DDCMs using real and artificial datasets. The experiments demonstrate that ignoring serial correlation in unobservables of DDCMs can lead to serious misspecification errors. Experiments on dynamic multinomial logit models, for which analytical integration is also possible, show that the estimation accuracy of the proposed method is good.

Chapter 2 presents a proof of the complete (and thus a.s.) uniform convergence of the DP solution approximations proposed in Chapter 1 to the true values under mild assumptions on the primitives of DDCMs. It also establishes the complete convergence of the corresponding approximated posterior expectations.

Chapter 3 proposes a method for inference in DDCMs that combines MCMC and artificial neural networks (ANN.) MCMC is intended to handle high dimensional integration in the likelihood function of richly specified DDCMs. ANNs approximate the DP solution as a function of the parameters and state variables beforehand of the estimation procedure to reduce the computational burden. Potential applications of the proposed methodology include inference in DDCMs with random coefficients, serially correlated unobservables, and dependent observations. The chapter discusses MCMC estimation of DDCMs, provides relevant background on ANNs, and derives a theoretical justification of the method. Experiments suggest that application of ANNs in the MCMC estimation of DDCMs is a promising approach.

TABLE OF CONTENTS

LIST OF TABLES viii

LIST OF FIGURES ix

CHAPTER

1 INFERENCE IN DYNAMIC DISCRETE CHOICE MODELS WITH SERIALLY CORRELATED UNOBSERVED STATE VARIABLES . 1

1.1 Introduction 1
1.2 Setup and estimation of DDCMs 6
1.3 Algorithm for solving the DP 14
1.3.1 Algorithm description 14
1.3.2 Comparison with Imai et al. (2005) 18
1.3.3 Theoretical results 19
1.4 Convergence of posterior expectations 24
1.5 Experiments 27
1.5.1 Gilleskie's (1998) model 27
1.5.2 Rust's (1987) model 49
1.6 Conclusion and future work 74

2 PROOFS OF THE THEORETICAL RESULTS 78

2.1 Lemmas 78
2.2 Extension to the uniform convergence 85
2.3 Convergence of posterior expectations and ergodicity of the Gibbs sampler 89
2.4 Auxiliary results 98

3 ESTIMATION OF DYNAMIC DISCRETE CHOICE MODELS WITH DP SOLUTION APPROXIMATED BY ARTIFICIAL NEURAL NETWORKS 113

3.1 Introduction 113
3.2 DDCM and MCMC 116
3.3 Feedforward ANN 120
3.3.1 Definition of feedforward ANN 120
3.3.2 Consistency of FFANN approximations 124
3.4 Experiments 127
3.4.1 Evaluating approximation quality 127

3.4.2 Estimation results 134

REFERENCES 138

LIST OF TABLES

Table

1.1 Estimation results for artificial data 42

1.2 Joint distribution test 48

1.3 Estimation results for artificial data 61

1.4 Exact and approximate estimation results. 64

1.5 Estimation results for the dynamic logit model and the model with Gaussian serially correlated unobservables. 67

1.6 Estimation results for data from Rust's paper, group 4. 70

LIST OF FIGURES

Figure

1.1 Flowchart of a direct search procedure for finding a fixed point. 36

1.2 Estimated densities of V_w. The tightest density corresponds to $\hat{N} = 1000$, the most widespread to $\hat{N} = 100$. The dashed lines are fitted normal densities. 38

1.3 Estimated densities of $\hat{E}^{(m)}[V(s';\theta)|s,d_1;\theta] - \hat{E}^{(m)}[V(s';\theta)|s,d_2;\theta]$. The tightest density corresponds to $\hat{N} = 1000$, the most widespread to $\hat{N} = 100$. The dashed lines are fitted normal densities. 39

1.4 Estimated posterior densities of (a) α_1, (b) α_2, (c) α_3, (d) α_4, (e) ρ, (f) η_1, (g) η_2, (h) η_3, (k) η_4, (l) $h_\epsilon^{-1/2}$. The dashed lines are prior densities. The vertical lines show the actual parameter values. 43

1.5 Posterior densities of η: (a) η_1, (b) η_2, (c) η_3, (d) η_4. The solid lines show the densities estimated by probit, the dotted lines by the full model. The vertical lines show the actual parameter values. 45

1.6 Posterior densities of α, ρ, and $h_\epsilon^{-1/2}$: (a) α_1, (b) α_2, (c) α_3, (d) α_4, (e) ρ, (f) $h_\epsilon^{-1/2}$. The solid lines show the densities estimated with fixed η (two simulator runs,) the dotted lines by the full model. The vertical lines show the actual parameter values. 46

1.7 Joint distribution tests: (a) α_1, (b) α_2, (c) α_3, (d) α_4, (e) ρ, (f) η_1, (g) η_2, (h) η_3, (k) η_4, (l) $h_\epsilon^{-1/2}$. 49

1.8 Estimated posterior densities: solid lines for $I = 500$, dotted lines for $I = 50$. (a) α_1, (b) α_2, (c) ρ, (d) η_1, (e) η_2, (f) η_3. The dashed lines are prior densities. The vertical lines show the actual parameter values. . . . 62

1.9 Comparison with exact and approximate estimation algorithms. Estimated posterior densities: (a) α_1, (b) α_2, (c) η_1, (d) η_2, (e) η_3. The vertical lines show the actual parameter values.The solid line shows the posterior for exact estimation procedure, the dashed line – approximate estimation procedure 66

1.10 The posterior means of the hazard functions. Panel (a)—the data generated from the model with the serially correlated unobservables, panel (b)—the data generated from the dynamic logit model. The vertical axis is for the probability of engine replacement, the horizontal axis is for the mileage interval. The solid line is for the model with serially correlated unobservables. The dashed line—for dynamic multinomial logit, the dotted line—data hazard. 68

1.11 Estimated posterior densities for different grids: (a) α_1, (b) α_2, (c) ρ, (d) η_1, (e) η_2, (f) η_3. The dashed lines are prior densities. The solid lines are posterior densities averaged over all simulator runs. The dotted lines show posterior densities averaged for runs 1–2, 3–4, and 5–6. 72

2.1 Nearest neighbors. 100

3.1 Multi-layer feed-forward neural network 121

3.2 Densities of the difference between the exact solution and the solution on random grids. The model with extreme value iid unobservables. The dashed line - the density of $[F(x^{ji}, \theta^j) - F^{ji}_{1,100}]$, the dotted line - the density of $[F(x^{ji}, \theta^j) - F^{ji}_{10,100}]$, and the solid line - the density of $[F(x^{ji}, \theta^j) - F^{ji}_{100,100}]$. 129

3.3 Densities of residuals e^{ji} (the dotted line is for the validation part of the sample. 130

3.4 Fitted values $\hat{F}(x^{ji}, \epsilon^j, \theta^j; w)$ (the dotted lines) and $F^{ji}_{100,100}$ (the solid lines) as functions of one input component. The model with serially correlated unobservables. The horizontal axes: (a) α_1, (b) α_2, (c) ρ, and (d) ϵ. . . 131

3.5 Densities of the approximation error for FFANNs and DP solutions on random grids. The model with extreme value iid unobservables. The dashed line - for a FFANN trained on exact data, the dotted line - for a FFANN trained on $F^{ji}_{100,100}$, the solid line - for $F^{ji}_{100,100}$. 132

3.6 Densities of e from neural network (the dotted line) and from DP solution on a random grid (the solid line.) The model with serially correlated unobservables. 133

3.7 Estimated posterior densities: (a) α_1, (b) α_2, (c) η_1, (d) η_2. The solid lines for the algorithm using the exact DP solutions, the dashed for the algorithm using the FFANN. 136

3.8 Estimated posterior densities: (a) α_1, (b) α_2. The solid lines - the simulator using the exact DP solutions, the other lines - the simulators using DP solutions on different random grids. 136

CHAPTER 1
INFERENCE IN DYNAMIC DISCRETE CHOICE MODELS WITH SERIALLY CORRELATED UNOBSERVED STATE VARIABLES

1.1 Introduction

Dynamic discrete choice models (DDCMs) describe the behavior of a forward-looking economic agent who chooses between several available alternatives repeatedly over time. Estimation of the deep structural parameters of such decision problem is a theoretically appealing and promising area in empirical economics. In contrast to conventional statistical modeling of discrete data, it does not fall under the Lucas critique and often produces better behavior forecasts. Structural estimation of dynamic models though, is very complex computationally. This fact substantially limits the ability of estimable models to capture essential features of the real world. One such important feature that had mainly to be assumed away in the literature is the presence of serial correlation in unobserved state variables. Although introducing serial dependence in modelled productivity, health status, or taste idiosyncrasies would improve the credibility of obtained quantitative results, general feasible estimation methods for dealing with serially correlated unobservables in dynamic discrete choice models are yet to be developed, according to Rust (1994). This chapter attempts to develop such a feasible general method.

Advances in simulation methods and computing speed over the last two decades made the Bayesian approach to statistical inference practical. Bayesian methods are now applied to many problems in statistics and econometrics that could not be tackled by the classical approach. Static discrete choice models, and more generally, models with latent variables, are one of those areas where the Bayesian approach was extremely fruitful, see for example McCulloch and Rossi (1994) and Geweke et al. (1994). In these models, the likelihood function is often an intractable integral over the latent variables. In Bayesian inference, the posterior distribution of the model parameters is usually explored by simulating a sequence of parameter draws that represents the posterior distribution. A simulation technique called the Gibbs sampler is particularly convenient for exploring posterior distributions in models with latent variables. This sampler simulates the parameters conditional on the data and the latent variables, and then simulates the latent variables conditional on the data and the parameters. The resulting sequence of the simulated parameters and latent variables is a Markov chain with the stationary distribution equal to the joint posterior distribution of the parameters and the latent variables. Thus, the high-dimensional integration required at each step of classical likelihood maximization can be replaced with sequential simulation from low-dimensional distributions in the Bayesian approach. In DDCMs, the likelihood function is an integral over the unobserved state variables. If the unobserved state variables are serially correlated, computing this integral is generally infeasible. Standard tools of Bayesian inference—the Gibbs sampler and the Metropolis-Hastings algorithm—are employed in this chapter to successfully handle this issue.

One of the main obstacles for Bayesian estimation of dynamic discrete choice models is the computational burden of solving the dynamic program at each iteration of the estimation procedure. Imai et al. (2005) were the first to attack this problem and consider application of Bayesian methods for estimation of dynamic discrete choice models with iid unobserved state variables. Their method uses a Markov chain Monte Carlo (MCMC) algorithm that solves the DP and estimates the parameters at the same time. The Bellman equation is iterated only once for each draw of the parameters. To obtain the approximations of the expected value functions for the current MCMC draw of the parameters, the authors use kernel smoothing over the approximations of the value functions from the previous MCMC iterations. The authors also provide a proof that for discrete observed state variables and deterministic observed state transitions their approximations of the value functions converge in probability to the true values.

This chapter extends the work of Imai et al. (2005) in several dimensions. First, it introduces a different parameterization of the Gibbs sampler and Metropolis-within-Gibbs steps to account for the effect of change in parameters on the expected value functions. Second, it allows for serial correlation in unobservables. Third, instead of kernel smoothing it uses nearest neighbors from previously generated parameter draws for approximating the expected value functions for the current parameter draw. The complete (and thus a.s., see Hsu and Robbins (1947)) uniform convergence of these nearest neighbor approximations is established for a more general model setup: a compact state space, random state transitions and less restrictive assumptions on the Gibbs sampler transition density. In addition to the wider theoretical applicability of this proposed DP solution method, there might be a substantial practical advantage since kernel smoothing does not work well in many dimensions: e.g., Scott (1992), pp. 189–190, shows that the nearest neighbor algorithm outperforms the usual kernel smoothing method in density estimation for Gaussian data if the number of dimensions exceeds four.

The proposed Gibbs sampler estimation procedure uses the approximations described above instead of the actual DP solutions. How this might affect inference results is an important issue. In Bayesian analysis, most inference exercises involve computing posterior expectations of some functions. For example, the posterior mean and the posterior standard deviation of a parameter can be expressed in terms of posterior expectations. Moreover, the answers to the policy questions that DDCMs address also take this form. Using the uniform complete convergence of the approximations of the expected value functions, I prove the complete convergence of the approximated posterior expectations under weak assumptions on a kernel of the joint posterior distribution of the parameters and the latent variables in the Gibbs sampler.

The estimation method is experimentally evaluated on two different DDCMs: the Rust (1987) binary choice model of optimal bus engine replacement and the Gilleskie (1998) model of medical care use and work absence. Serially correlated unobserved state variables are introduced into these models instead of the original extreme value iid unobservables. Model simplicity and availability of the data[1] make Rust's model very attractive for computational experiments. Experiments on Gilleskie's

[1] http://gemini.econ.umd.edu/jrust/nfxp.html

model in turn show that the method works when the number of alternatives exceeds two.

Estimation experiments presented in the chapter are meant to demonstrate the utility of the proposed method. Experiments on data from Rust (1987) confirm Rust's conclusion of weak evidence of the presence of serial correlation in unobservables for his model and dataset. However, experiments on artificial data show that the estimated choice probabilities implied by a dynamic logit model and a model with serially correlated unobservables can behave quite differently. More generally, the experiments demonstrate that ignoring serial correlation in unobservables of DDCMs can lead to serious misspecification errors.

The proposed theoretical framework is flexible and leaves room for experimentation. Experiments with the algorithm for solving the DP led to a discovery of modifications that provided increases in speed and precision beyond those anticipated directly by the theory. First, iterating the Bellman equation on several smaller random grids and combining the results turns out to be a very efficient alternative to iterating the Bellman equation on one larger random grid. Second, the approximation error for a difference of expected value functions is considerably smaller than the error for an expected value function by itself (this can be taken into account in the construction of the Gibbs sampler.) Finally, iterating the Bellman equation several times for each parameter draw, using the Gauss-Seidel method and a direct search procedure, also produces significant performance improvement.

A verification of the algorithm implementation is provided in the chapter. For example, to assess the accuracy of the proposed DP solving algorithm I apply it to a dynamic multinomial logit model, in which unobservables are extreme value iid and the exact DP solution can be quickly computed. The design and implementation of the posterior, prior, and data simulators are checked by joint distribution tests (see Geweke (2004).) Multiple posterior simulator runs are used to check the convergence of the MCMC estimation procedure. The proposed estimation algorithm can be applied to dynamic multinomial logit models, for which an exact algorithm is also available. A comparison of the estimation results for the proposed algorithm and the exact algorithm suggests that the estimation accuracy is excellent.

Section 1.2 of the chapter sets up a general dynamic discrete choice model, constructs its likelihood function, and outlines classical and Bayesian estimation procedures. The algorithm for solving the DP and corresponding convergence results are presented in Section 1.3. Section 1.4 states the convergence result for the approximated posterior expectations. The proofs are given in Chapter 2. The models used in experiments are described in Section 1.5. This section also provides a verification of the method and implementation details. The last section concludes with a summary of findings and directions for future work.

1.2 Setup and estimation of DDCMs

Eckstein and Wolpin (1989) and Rust (1994) survey the literature on the classical estimation of dynamic discrete choice models. Below, I briefly introduce a general model setup and emphasize possible advantages of the Bayesian approach to the estimation of these models, especially in treating the time dependence in unobservables.

Dynamic discrete choice models describe the behavior of an optimizing forward-looking economic agent who chooses between several available alternatives repeatedly over time taking into account her expectations about unknown future developments and her optimal future choices. Each period t the agent chooses an alternative d_t from a finite set of available alternatives $D(s_t)$. The per-period utility $u(s_t, d_t; \theta)$ depends on the chosen alternative, current state variables $s_t \in S$, and a vector of parameters $\theta \in \Theta$ that we want to estimate. The state variables are assumed to evolve according to a controlled first order Markov process with a transition law denoted by $f(s_{t+1}|s_t, d_t; \theta)$ for $t \geq 1$; the distribution of the initial state is denoted by $f(s_1|\theta)$. Time is discounted with a factor β. In the recursive formulation of the problem, the lifetime utility of the agent or the value function is given by the maximum of the alternative-specific value functions:

$$V(s_t; \theta) = \max_{d_t \in D(s_t)} \mathcal{V}(s_t, d_t; \theta) \tag{1.1}$$

$$\mathcal{V}(s_t, d_t; \theta) = u(s_t, d_t; \theta) + \beta E\{V(s_{t+1}; \theta)|s_t, d_t; \theta\} \tag{1.2}$$

This formulation embraces a finite horizon case if time t is included in the vector of the state variables.

In an estimable dynamic discrete choice model it is usually assumed that some state variables are unobserved by econometricians. Let's denote the unobserved part of the state variables by y_t and the observed part by x_t. All the state variables $s_t = (x_t, y_t)$ are known to the agent at time t when they are realized. No model can perfectly predict human behavior. Using the unobserved state variables is an attractive way to structurally incorporate random errors in the model. The unobserved state variables can be interpreted as shocks, taste idiosyncrasy, unobserved heterogeneity, or measurement errors. They may also be more specific: e.g. health status or returns to patents. The unobservables play an important role in the estimation. The likelihood function of a DDCM is an integral over the unobservables. In a static case, as few as n unobservables can be used in a model with 2^n alternatives to produce non-zero choice probabilities for all the alternatives and for any parameter vector in Θ given that the support of the distribution for the unobservables is sufficiently large relative to Θ. It would happen, for example, if a distinct combination of the components of n-dimensional y_t additively enters the utility function for each alternative. However, it is often more convenient to assume a larger number of the unobservables, e.g., a dynamic multinomial logit model has one unobservable for each alternative.

The set of the available alternatives $D(s_t)$ is assumed to depend only on the observed state variables. Hereafter, it will be denoted by D to simplify the notation. This is without loss of generality since we could set $D = \cup_{x_t \in X} D(x_t)$ and the alternatives unavailable at state x_t could be assigned a low per-period utility value.

A data set that is usually used for the estimation of a dynamic discrete choice model consists of a panel of I individuals. The observed part of the state and the decisions are known for each individual $i \in \{1, \ldots, I\}$ for T_i periods: $\{x_{t,i}, d_{t,i}\}_{t=1}^{T_i}$. Assuming that the state variables are independent for the individuals in the sample, the likelihood for the model can be written as

$$p(\{x_{t,i},d_{t,i}\}_{t=1}^{T_i}, i\in\{1,\ldots,I\}|\theta) = \prod_{i=1}^{I} p(x_{T_i,i},d_{T_i,i},\ldots,x_{1,i},d_{1,i}|\theta) = \tag{1.3}$$

$$\prod_{i=1}^{I}\int p(y_{T_i,i},x_{T_i,i},d_{T_i,i},\ldots,y_{1,i},x_{1,i},d_{1,i}|\theta)dy_{T_i,i},\ldots,dy_{1,i}$$

The joint density $p(y_{T_i,i},x_{T_i,i},d_{T_i,i},\ldots,y_{1,i},x_{1,i},d_{1,i}|\theta)$ could be decomposed as follows

$$p(y_{T_i,i},x_{t,i},d_{t,i},\ldots,y_{1,i},x_{1,i},d_{1,i}|\theta) = \prod_{t=1}^{T_i} p(d_{t,i}|y_{t,i},x_{t,i};\theta)f(x_{t,i},y_{t,i}|x_{t-1,i},y_{t-1,i},d_{t-1,i};\theta) \tag{1.4}$$

where $f(.|.;\theta)$ is the state transition density, $\{x_{0,i},y_{0,i},d_{0,i}\}=\emptyset$, and $p(d_{t,i}|y_{t,i},x_{t,i};\theta)$ is an indicator function:

$$p(d_{t,i}|y_{t,i},x_{t,i};\theta) = 1_{\{\mathcal{V}(y_{t,i},x_{t,i},d_{t,i};\theta)\geq\mathcal{V}(y_{t,i},x_{t,i},d;\theta),\forall d\in D\}}(y_{t,i},x_{t,i},d_{t,i};\theta) \tag{1.5}$$

In general, evaluation of the likelihood function in (1.3) involves computing multidimensional integrals of an order equal to T_i times the number of components in y_t, which becomes infeasible for large T_i and/or multi-dimensional unobservables y_t. That is why in previous literature the unobservables were mainly assumed to be iid. In a series of papers, John Rust developed a dynamic multinomial logit model, where he assumed that the utility function of the agents is additively separable in the unobservables and that the unobservables are extreme value iid. In this case, the integration in (1.3) can be performed analytically. Pakes (1986) used Monte Carlo simulations to approximate the likelihood function in a model of binary choice with a serially correlated one-dimensional unobservable.

In a Bayesian framework, the high dimensional integration over y_t for each parameter value can be circumvented by employing Gibbs sampling and data augmentation. In models with latent variables, the Gibbs sampler typically has two types of blocks: (a) parameters conditional on other parameters, latent variables and the data; (b) latent variables conditional on other latent variables, parameters and the data (this step is sometimes called data augmentation.) The draws simulated from this Gibbs sampler form a Markov chain with the stationary distribution equal to the joint distribution of the parameters and the latent variables conditional on the data. The densities for both types of the blocks are proportional to the joint density of the data, the latent variables, and the parameters. Therefore, in order to construct the Gibbs sampler in our case, we need to obtain an analytical expression for the joint density of the data, the latent variables, and the parameters.

By a parameterization of the Gibbs sampler I mean a set of parameters and latent variables used in constructing the sampler. One parameterization is obtained from another by a change of variables. The number of the variables does not have to be the same for different parameterizations: some variables could just have degenerate distributions given other variables in the parameterization. This section illustrates that although any parameterization validly describes the econometric model, the parameterization choice could be crucial for the Gibbs sampler performance. For a simple example, consider parameterizing a multinomial probit model by the error terms and the parameters instead of the latent utilities and the parameters.

It is straightforward to obtain an analytical expression for the joint density of the data, the latent variables, and the parameters under the parameterization of the Gibbs sampler in which the unobserved state variables are directly used as the latent variables in the sampler:

$$p(\theta; \{d_{t,i}; y_{t,i}; x_{t,i}\}_{t=1}^{T_i}; i = 1, \ldots, I) =$$
$$p(\theta) \prod_{i=1}^{I} \prod_{t=1}^{T_i} p(d_{t,i}|x_{t,i}, y_{t,i}; \theta) f(x_{t,i}, y_{t,i}|x_{t-1,i}, y_{t-1,i}, d_{t-1,i}; \theta) \tag{1.6}$$

where $p(\theta)$ is a prior density for the parameters and $p(d_{t,i}|x_{t,i}, y_{t,i}; \theta)$ is an indicator function defined in (1.5). It is evident from (1.6) that in this Gibbs sampler, the parameter blocks will be drawn subject to the observed choice optimality constraints:

$$\mathcal{V}(y_{t,i}, x_{t,i}, d_{t,i}; \theta) \geq \mathcal{V}(y_{t,i}, x_{t,i}, d; \theta), \forall d \in D, \forall t \in \{1, \ldots, T_i\}, \forall i \in \{1, \ldots, I\} \tag{1.7}$$

For realistic sample sizes, the number of these constraints is very large and the algorithm becomes impractical. The same situation occurs under the parameterization in which $u_{t,d,i} = u(y_{t,i}, x_{t,i}, d_{t,i}; \theta)$ are used as the latent variables in the sampler instead of some or all of the components of $y_{t,i}$[2].

The complicated truncation region (1.7) in drawing the parameter blocks could be avoided if we use $\mathcal{V}_{t,i} = \{\mathcal{V}_{t,d,i} = \mathcal{V}(s_{t,i}, d; \theta), d \in D\}$ as latent variables in the sampler. However, then some extra assumptions on the unobserved state variables are needed so that the joint density of the data, the latent variables, and the parameters could be specified analytically. A way to achieve this when an analytical solution to the DP is not available is to assume that the unobserved part of the state vector includes some serially conditionally independent components that do not affect the distribution of the future state. Let's denote them by ν_t and the other (possibly

[2] Imai et al. (2005) seem to use this parameterization, but they omit the observed choice optimality constraints (1.7) in drawing the parameters. From my communication with Professor Imai, I understand that it will be changed in the next version of their paper.

serially correlated) ones by ϵ_t; so, $y_t = (\nu_t, \epsilon_t)$ and

$$f(x_{t+1}, \nu_{t+1}, \epsilon_{t+1} | x_t, \nu_t, \epsilon_t, d; \theta) = p(\nu_{t+1} | x_{t+1}, \epsilon_{t+1}; \theta) p(x_{t+1}, \epsilon_{t+1} | x_t, \epsilon_t, d; \theta) \tag{1.8}$$

Then, the expected value function $E\{V(s_{t+1}; \theta) | s_t, d; \theta)\}$ will not depend on the unobservables ν_t. The alternative specific value functions $\mathcal{V}_{t,i} = \{u(\nu_{t,i}, \epsilon_{t,i}, x_{t,i}, d; \theta) + \beta E[V(s_{t+1}; \theta) | \epsilon_{t,i}, x_{t,i}, d; \theta)], d \in D\}$ will have analytical expressions as functions of ν_t. Thus, the density of the distribution of $\mathcal{V}_{t,i} | \theta, x_{t,i}, \epsilon_{t,i}$ could have an analytical expression in contrast to the case when ν_t are serially conditionally dependent and the expectation term depends on them.

A simple example of an analytical expression for the density $p(\mathcal{V}_{t,i} | \theta, x_{t,i}, \epsilon_{t,i})$ is obtained for normal iid $\nu_t = \{\nu_{t,d}\}_{d \in D}$ and $u(\nu_{t,i}, \epsilon_{t,i}, x_{t,i}, d; \theta) = u(\epsilon_{t,i}, x_{t,i}, d; \theta) + \nu_{t,d,i}$. The serially correlated unobservables $\epsilon_{t,i}$ could follow an AR(1) process and also enter the utility function additively:

$$u(y_{t,i}, x_{t,i}, d; \theta) = u(x_{t,i}, d; \theta) + \nu_{t,d,i} + \epsilon_{t,d,i} \tag{1.9}$$

This formulation could be seen as a simple way of introducing time persistent unobserved heterogeneity in the model. The serially correlated unobservables could also have a more meaningful economic interpretation, e.g. health status, and enter the utility function differently. In general, the number of components in ν_t and ϵ_t does not have to be the same and they do not have to enter the utility additively.

The requirement of the presence of the serially conditionally independent unobservables and the existence of a convenient analytical expression for $p(\mathcal{V}_{t,i} | \theta, x_{t,i}, \epsilon_{t,i})$ does restrict the class of the DDCMs that can be estimated by the proposed method.

However, this restriction does not seem to be strong since the process for the unobservables can still be made quite flexible.

Assuming that a convenient analytical expression for $p(\mathcal{V}_{t,i}|\theta, x_{t,i}, \epsilon_{t,i})$ exists, the joint distribution of the data, the parameters and the latent variables can be decomposed into parts with known analytical expressions:

$$p(\theta; \{d_{t,i}; \mathcal{V}_{t,i}; x_{t,i}; \epsilon_{t,i}\}_{t=1}^{T_i}; i = 1, \ldots, I) =$$
$$p(\theta) \prod_{i=1}^{I} \prod_{t=1}^{T_i} p(d_{t,i}|\mathcal{V}_{t,i}) p(\mathcal{V}_{t,i}|x_{t,i}, \epsilon_{t,i}; \theta) p(x_{t,i}, \epsilon_{t,i}|x_{t-1,i}, \epsilon_{t-1,i}, d_{t-1,i}; \theta) \tag{1.10}$$

Under this parameterization, the observed choice optimality constraints

$$p(d_{t,i}|\mathcal{V}_{t,i}; \theta; x_{t,i}; \epsilon_{t,i}) = p(d_{t,i}|\mathcal{V}_{t,i}) = 1_{\{\mathcal{V}_{t,d_{t,i},i} \geq \mathcal{V}_{t,d,i}, d \in D\}}(d_{t,i}, \mathcal{V}_{t,i}) \tag{1.11}$$

will not depend on the parameters and will be present only in the blocks for $\mathcal{V}_{t,d,i}| \ldots$. This could be easily handled since there will be only one constraint for each block $\mathcal{V}_{t,d,i}| \ldots$. Complete specifications of the Gibbs sampler constructed along these lines are given in Section 1.5 for the models used in experiments.

Further simplification of the Gibbs sampler is possible if we assume that the per-period utility function is given by (1.9) and that the unobservables $\nu_{t,d,i}$ are extreme value iid. Then, $\mathcal{V}_{t,i}$ can be integrated out analytically as in dynamic multinomial logit models. This slight simplification is not pursued here.

The Gibbs sampler outlined above requires computing the expected value functions for each new parameter draw θ^m from the MCMC iteration m and each observation in the sample:

$$E[V(s_{t+1}; \theta^m)|x_{t,i}, \epsilon^m_{t,i}, d; \theta^m)], \forall i, t, d$$

The following section describes how the approximations of the expected value functions are obtained.

1.3 Algorithm for solving the DP

For a discussion of methods for solving the DP in (1.1) and (1.2) for a given parameter vector θ, see the literature surveys by Eckstein and Wolpin (1989) and Rust (1994). Models used in the previous literature were mostly amenable to a significant analytical simplification. For example, in Rust's dynamic multinomial logit model, the integration in computing expected value functions could be performed analytically. Below, I introduce a method of solving the dynamic program suitable for use in conjunction with the Bayesian estimation of a general dynamic discrete choice model. This method uses an idea from Imai et al. (2005) of iterating the Bellman equation only once at each step of the estimation procedure and using information from previous steps to approximate the expectations in the Bellman equation. However, the way the previous information is used differs for the two methods. A detailed comparison is given in Section 1.3.2.

1.3.1 Algorithm description

In contrast to conventional value function iteration, this algorithm iterates the Bellman equation only once for each parameter draw. First, I will describe how the DP solving algorithm works and then how the output of the DP solving algorithm is used to approximate the expected value functions in the Gibbs sampler.

The DP solving algorithm takes a sequence of parameter draws θ^m, $m =$

$1, 2, \dots$ as an input from the Gibbs sampler, where m denotes the Gibbs sampler iteration. For each θ^m, the algorithm generates random states $s^{m,j} \in S$, $j = 1, \dots, \hat{N}(m)$. At each random state, the approximations of the value functions $V^m(s^{m,j}; \theta^m)$ are computed by iterating the Bellman equation once. At this one iteration of the Bellman equation, the future expected value functions are computed by importance sampling over value functions $V^k(s^{k,j}; \theta^k)$ from previous iterations $k < m$.

The random states $s^{m,j}$ are generated from a density $g(.) > 0$ on S. This density $g(.)$ is used as an importance sampling source density in approximating the expected value functions. The collection of the random states $\{s^{m,j}\}_{j=1}^{\hat{N}(m)}$ will be referred below as the random grid[3]. The number of points in the random grid at iteration m is denoted by $\hat{N}(m)$ and it will be referred below as the size of the random grid (at iteration m.)

For each point in the current random grid $s^{m,j}$, $j = 1, \dots, \hat{N}(m)$, the approximation of the value function $V^m(s^{m,j}; \theta^m)$ is computed according to

$$V^m(s; \theta) = \max_{d \in D} \{u(s, d; \theta) + \beta \hat{E}^{(m)}[V(s'; \theta) | s, d; \theta]\} \tag{1.12}$$

Not all of the previously computed value functions $V^k(s^{k,j}; \theta^k)$, $k < m$ are used in importance sampling for computing $\hat{E}^{(m)}[V(s'; \theta) | s, d; \theta]$ in (1.12). In order to converge the algorithm has to forget the remote past. Thus, at each iteration m, I keep track only of the history of length $N(m)$: $\{\theta^k; s^{k,j}, V^k(s^{k,j}; \theta^k), j = 1, \dots, \hat{N}(k)\}_{k=m-N(m)}^{m-1}$.

[3] Rust (1997) shows that value function iteration on random grids from a uniform distribution breaks the curse of dimensionality for DDCMs. The Keane and Wolpin (1994) procedure of evaluating expectations only for some grid points and using interpolation for the rest could be used to increase the speed of the algorithm when the dimension of the state space is large.

In this history, I find $\tilde{N}(m)$ closest to θ parameter draws. Only the value functions corresponding to these nearest neighbors are used in importance sampling. Formally, let $\{k_1, \ldots, k_{\tilde{N}(m)}\}$ be the iteration numbers of the nearest neighbors of θ in the current history:

$$\begin{aligned} k_1 &= \arg\min_{i \in \{m-N(m),\ldots,m-1\}} \left|\left|\theta - \theta^i\right|\right| \\ k_j &= \arg\min_{i \in \{m-N(m),\ldots,m-1\} \setminus \{k_1,\ldots,k_{j-1}\}} \left|\left|\theta - \theta^i\right|\right|,\ j = 2, \ldots, \tilde{N}(m) \end{aligned} \tag{1.13}$$

If the arg min returns a multivalued result, I use the lexicographic order for $(\theta^i - \theta)$ to decide which θ^i is chosen first. If the result of the lexicographic selection is also multivalued: $\theta^i = \theta^j$, then I choose θ^i over θ^j if $i > j$. This particular way of resolving the multivaluedness of the arg min might seem irrelevant for implementing the method in practice; however, it is important for the proof of the measurability of the supremum of the approximation error, which is necessary for the uniform convergence results. A reasonable choice for the norm would be $||\theta|| = \sqrt{\theta^T \underline{H}_\theta \theta}$, where $\underline{H}_\theta$ is the prior precision for the parameters. Importance sampling is performed as follows:

$$\begin{aligned} &\hat{E}^{(m)}[V(s';\theta)|s,d;\theta] \\ &= \sum_{i=1}^{\tilde{N}(m)} \sum_{j=1}^{\hat{N}(k_i)} V^{k_i}(s^{k_i,j};\theta^{k_i}) \frac{f(s^{k_i,j} \mid s,d;\theta)/g(s^{k_i,j})}{\sum_{r=1}^{\tilde{N}(m)} \sum_{q=1}^{\hat{N}(k_r)} f(s^{k_r,q} \mid s,d;\theta)/g(s^{k_r,q})} \qquad (1.14) \\ &= \sum_{i=1}^{\tilde{N}(m)} \sum_{j=1}^{\hat{N}(k_i)} V^{k_i}(s^{k_i,j};\theta^{k_i}) W_{k_i,j,m}(s,d,\theta) \qquad (1.15) \end{aligned}$$

The target density for importance sampling is the state transition density $f(.|s,d;\theta)$. The source density is the density $g(.)$ from which the random grid on the state space is generated. The computation of the weights $W_{k_i,j,m}(s,d,\theta)$ could be simplified if a part of the state vector is serially independent and its distribution does not depend on the parameters and the other state variables. Both models used for experiments contain examples of that: the unobservables ν_t are Gaussian iid with zero mean and the variance fixed for normalization. In this case the source density for ν_t could be the same as the density according to which ν_t are distributed in the model. Then, the part of the weight $W_{k_i,j,m}(s,d,\theta)$ corresponding to ν_t would be equal to 1. In general, $g(.)$ should give reasonably high probabilities to all parts of the state space that are likely under $f(.|s,d;\theta)$ with reasonable values of the parameter θ. To reduce the variance of the approximation of expectations produced by importance sampling[4], one should make $g(.)$ relatively high for the states that result in larger value functions.

To obtain the convergence of the DP solution approximations as $m \to \infty$, we have to impose some obvious restrictions on the size of the random grid $\hat{N}(m)$, the length of the tracked history $N(m)$, and the number of the nearest neighbors $\tilde{N}(m)$. The length of the tracked history $N(m)$ has to go to infinity so that when we pick the nearest neighbors from this history they get very close to the current parameter. For the same reason the number of the nearest neighbors $\tilde{N}(m)$ has to be small relative to $N(m)$. The length of the forgotten history $m - N(m)$ has to go to infinity so that early imprecise approximations would not contaminate the future ones. A lower bound on the number of the random states used in importance sampling

[4] Importance sampling is used as a variance reduction technique for Monte Carlo simulations

$[\tilde{N}(m) \cdot \min_{i \in \{m-N(m),\ldots,m-1\}} \hat{N}(i)]$ should go to infinity so that the importance sampling approximations of the integrals converge. More specific assumptions on $N(m)$, $\tilde{N}(m)$, and $\hat{N}(m)$ are made in the current version of the algorithm convergence proof. They are described along with the assumptions on the model primitives in Section 1.3.3, which formally presents convergence results.

After $V^m(s^{m,j};\theta^m)$ are computed, formula (1.14) is used to obtain the approximations of the expectations $E[V(s_{t+1};\theta^m)|x_{t,i},\epsilon^m_{t,i},d;\theta^m)]$ $\forall i,t,d$ in the Gibbs sampler.

1.3.2 Comparison with Imai et al. (2005)

Imai et al. (2005) use kernel smoothing over all $N(m)$ previously computed value functions to approximate the expected value functions. They do not need the importance sampling for the iid unobserved states; they also generate only one new state at each iteration, $\hat{N}(m) = 1, \forall m$. In contrast, I use the nearest neighbor (NN) algorithm instead of kernel smoothing. The advantage of the NN algorithm seems to be twofold. First, it was shown to outperform kernel smoothing in density estimation when the number of dimensions exceeds four. Thus, it might work better in practice for the DP solving algorithm as well. Second, the NNs seem to be easier to deal with mathematically. First of all, to prove the convergence of the DP solution approximations I do not have to impose the requirement of a uniform upper bound on the Gibbs sampler transition density (used by Imai et al. (2005) in their Lemma 2), which I have not managed to establish for the actual Gibbs sampler. Second, the Imai et al. (2005) assumption of finiteness of the observed states space X can be substituted by compactness. Third, Imai et al. (2005) assumed deterministic transition for the observed states in the proof and iid unobserved states. With NN approximations, random state transitions can be used. Imai et al. (2005) proved the convergence in probability for their DP solution approximations with bounds on the probabilities that are uniform over the parameter space. For the NN algorithm, I establish a much stronger type of convergence: the complete uniform convergence. Most importantly, the strong convergence results for the NN approximations of the DP solutions are shown to imply the convergence of the approximated posterior expectations, which provides a complete theoretical justification for the proposed Bayesian estimation algorithm.

1.3.3 Theoretical results

The following assumptions on the model primitives and the algorithm parameters are made:

Assumption 1.1. *$\Theta \subset R^{J_\Theta}$ and $S \subset R^{J_S}$ are bounded rectangles.*

Assumption 1.2. *$u(s, d; \theta)$ is bounded, $\beta \in (0, 1)$ is known.*

Assumption 1.3. *$V(s; \theta)$ is continuous in (θ, s).*

Assumption 1.3 will hold, for example, under the following set of restrictions on the primitives of the model: Θ and S are compact, $u(s, d; \theta)$ is continuous in (s, θ), and $f(s' \mid s, d; \theta)$ is continuous in (θ, s, s') (for a proof see Proposition 2.4.)

Assumption 1.4. *The density of the state transition $f(.|.)$ and the source importance density $g(.)$ are bounded above and away from zero, which gives:*

$$\inf_{\theta,s',s,d} f(s'|s,d;\theta)/g(s') = \underline{f} > 0$$

$$\sup_{\theta,s',s,d} f(s'|s,d;\theta)/g(s') = \overline{f} < \infty$$

Assumption 1.5. $\exists \hat{\delta} > 0$ *such that* $P(\theta^{m+1} \in A|\omega^m) \geq \hat{\delta}\lambda(A)$ *for any Borel measurable* $A \subset \Theta$*, any* m*, and any feasible history* $\omega^m = \{\omega_1, \ldots, \omega_m\}$ *where* λ *is the Lebesgue measure. The history includes all the parameter and latent variable draws from the Gibbs sampler and all the random grids from the DP solving algorithm:* $\omega_t = \{\theta^t, \Delta\mathcal{V}^t, \epsilon^t; s^{t,j}, j = 1, \ldots, \hat{N}(t)\}$.

Assumption 1.5 means that at each iteration of the algorithm, the parameter draw can get into any part of Θ. This assumption should be verified for each specific DDCM and the corresponding parameterization of the Gibbs sampler. The assumption is only a little stronger than standard conditions for convergence of the Gibbs sampler, see Corollary 4.5.1 in Geweke (2005). Since a careful practitioner of MCMC would have to establish convergence of the Gibbs sampler, a verification of Assumption 1.5 should not require much extra effort. Even if the assumption is not satisfied for the Gibbs sampler, the DP solving algorithm can be theoretically justified if the parameter draws from the Gibbs sampler are mixed with parameter draws from a positive on Θ density for creating the input sequence $\theta^1, \theta^2, \ldots$ for the DP solving algorithm.

Assumption 1.6. *Let $1 > \gamma_0 > \gamma_1 > \gamma_2 \geq 0$ and $N(t) = [t^{\gamma_1}]$, $\tilde{N}(t) = [t^{\gamma_2}]$, $\hat{N}(t) = [t^{\gamma_1 - \gamma_2}]$, and $\hat{N}(0) = 1$, where $[x]$ is the integer part of x.*

Multiplying the functions of t in Assumption 1.6 by positive constants will not affect any of the theoretical results below.

Theorem 1.1. *Under Assumptions 1.1-1.6, the approximation to the expected value function in (1.14) converges completely (and thus a.s.) to the true value with probability bounds that are uniform over parameter and state spaces: that is for any $\tilde{\epsilon} > 0$ there exists a sequence $\{z_t\}$ such that $\sum_{t=0}^{\infty} z_t < \infty$ and for any $\theta \in \Theta$, $s \in S$, and $d \in D$:*

$$P(|\hat{E}^{(t)}[V(s';\theta) \mid s, d; \theta] - E[V(s';\theta) \mid s, d; \theta]| > \tilde{\epsilon}) \leq z_t \tag{1.16}$$

Assumption 1.4 could be relaxed when a part of the state vector is discrete and the number of possible discrete states is finite. Let's denote such discrete part of the state vector by s_f. If the transition for the discrete part of the state is deterministic then Assumption 1.4 would be required to hold for each discrete state s_f and importance sampling would be performed only for the continuous part of the state space. If the transition is not deterministic and does not satisfy Assumption 1.4 then for each discrete part of the state and possible decision d we could introduce a separate space of possible future states $S(s_f, d)$. On each of those spaces we would define an importance sampling source density $g(.|s_f, d)$. Then, the DP solution convergence can also be established if an analog of Assumption 1.4 is satisfied for each discrete part of the state s_f and decision d and the corresponding space of possible future states $S(s_f, d)$. For a formal statement of these results see Proposition 2.7.

Theorem 1.1 gives uniform bounds on the probabilities that the approximation error for fixed (θ, s) exceeds some positive number. However, the uniform convergence for random functions, which seems to be easier to apply but harder to establish, is defined differently in the literature (see Bierens (1994)). A uniform version of Theorem 1.1 can be obtained given an extra assumption:

Assumption 1.7. *Fix a combination* $m = \{m_1, \ldots, m_{\tilde{N}(t)}\}$ *from* $\{t - N(t), \ldots, t-1\}$. *Let*

$$X(\omega^{t-1}, \theta, s, d, m) = \left| \sum_{i=1}^{\tilde{N}(t)} \sum_{j=1}^{\hat{N}(m_i)} \frac{(V(s^{m_i,j}; \theta) - E[V(s'; \theta) \mid s, d; \theta]))f(s^{m_i,j} \mid s, d; \theta)/g(s^{m_i,j})}{\sum_{r=1}^{\tilde{N}(t)} \sum_{q=1}^{\hat{N}(m_r)} f(s^{m_r,q} \mid s, d; \theta)/g(s^{m_r,q})} \right| \tag{1.17}$$

Assume that family of functions $\{X(\omega^{t-1}, \theta, s, d, m)\}_{\omega^{t-1}}$ *is equicontinuous in* (θ, s).

This assumption will be satisfied, for example, if $u(s, d; \theta)$ is continuous in (θ, s) on the compact set $\Theta \times S$ and $f(s' \mid s, d; \theta)$ and $g(s')$ are continuous in (θ, s, s') and satisfy Assumption 1.4 (for a proof see Propositions 2.4 and 2.5.)

Theorem 1.2. *Under Assumptions 1.1-1.7, the approximation to the expected value function in (1.14) converges uniformly and completely to the true value: that is*

(i) $\sup_{s,\theta,d} |\hat{E}^{(t)}[V(s'; \theta) \mid s, d; \theta] - E[V(s'; \theta) \mid s, d; \theta]|$ *is measurable,*

(ii) *for any* $\tilde{\epsilon} > 0$ *there exists a sequence* $\{z_t\}$ *such that* $\sum_{t=0}^{\infty} z_t < \infty$ *and*

$$P(\sup_{s,\theta,d} |\hat{E}^{(t)}[V(s'; \theta) \mid s, d; \theta] - E[V(s'; \theta) \mid s, d; \theta]| > \tilde{\epsilon}) \leq z_t \tag{1.18}$$

The proof of Theorem 1.2 is given in Chapter 2, Section 2.2. It is a modification of the proof of Theorem 1.1, the main steps of which are given below.

Proof. (Theorem 1.1) Let's decompose the error of approximation into three parts:

$$\begin{aligned}
&\left|\hat{E}^{(t)}[V(s';\theta)|s,d;\theta] - E[V(s';\theta) \mid s,d;\theta]\right| \\
= &\left|\sum_{i=1}^{\tilde{N}(t)}\sum_{j=1}^{\hat{N}(k_i)} V^{k_i}(s^{k_i,j};\theta^{k_i})W_{k_i,j,t}(s,d,\theta) - E[V(s';\theta) \mid s,d;\theta]\right| \\
\le &\left|\sum_{i=1}^{\tilde{N}(t)}\sum_{j=1}^{\hat{N}(k_i)} V(s^{k_i,j};\theta)W_{k_i,j,t}(s,d,\theta) - E[V(s';\theta) \mid s,d;\theta]\right| \\
&+\left|\sum_{i=1}^{\tilde{N}(t)}\sum_{j=1}^{\hat{N}(k_i)} (V(s^{k_i,j};\theta^{k_i}) - V(s^{k_i,j};\theta))W_{k_i,j,t}(s,d,\theta)\right| \\
&+\left|\sum_{i=1}^{\tilde{N}(t)}\sum_{j=1}^{\hat{N}(k_i)} (V^{k_i}(s^{k_i,j};\theta^{k_i}) - V(s^{k_i,j};\theta^{k_i}))W_{k_i,j,t}(s,d,\theta)\right| \\
= \quad & A_1^t(\theta,s,d) + A_2^t(\theta,s,d) + A_3^t(\theta,s,d) \\
\le \quad & \max_d A_1^t(\theta,s,d) + \max_d A_2^t(\theta,s,d) + \max_d A_3^t(\theta,s,d) \\
= \quad & A_1^t(\theta,s) + A_2^t(\theta,s) + A_3^t(\theta,s) \qquad (1.19)
\end{aligned}$$

In Lemma 2.1, I show that $A_1^t(\theta,s)$ converges to zero completely with bounds on probabilities that are independent of θ and s. The proof uses Hoeffding's inequality implying a SLLN for bounded random variables. However, some additional work is required since $s^{k_i,j}$ do not constitute a random sample. Using the continuity of the value function $V(.)$, the compactness of the parameter space Θ, and the assumption that each parameter draw can get into any point in Θ (Assumption 1.5,) I show analogous result for $A_2^t(\theta,s)$ in Lemma 2.2. In Lemma 2.3, I bound $A_3^t(\theta,s)$ by a weighted sum of $A_1^t(\theta,s)$ and $A_2^t(\theta,s)$ from previous iterations. Due to very fast convergence of $A_1^t(\theta,s)$ and $A_2^t(\theta,s)$, $A_3^t(\theta,s)$ also converges to zero completely. Thus, from the three Lemmas the result follows. Formally, according to Lemmas 2.1, 2.2, 2.3, there exist $\delta_1 > 0$, $\delta_2 > 0$, $\delta_3 > 0$ and T such that $\forall \theta \in \Theta$, $\forall s \in S$, and $\forall t > T$:

$$P(|A_1^t(\theta, s)| > \tilde{\epsilon}/3) \leq e^{-0.5\delta_1 t^{\gamma_1}}$$

$$P(|A_2^t(\theta, s)| > \tilde{\epsilon}/3) \leq e^{-0.5\delta_2 t^{\gamma_1}}$$

$$P(|A_3^t(\theta, s)| > \tilde{\epsilon}/3) \leq e^{-\delta_3 t^{\gamma_0 \gamma_1}}$$

Combining the above equations gives:

$$\begin{aligned}
&P(|\hat{E}^{(t)}[V(s';\theta) \mid s, d; \theta] - E[V(s';\theta) \mid s, d; \theta]| > \tilde{\epsilon}) \\
&\leq P(A_1^t(\theta, s) + A_2^t(\theta, s) + A_3^t(\theta, s) > \tilde{\epsilon}) \\
&\leq P(|A_1^t(\theta, s)| > \tilde{\epsilon}/3) + P(|A_2^t(\theta, s)| > \tilde{\epsilon}/3) + P(|A_3^t(\theta, s)| > \tilde{\epsilon}/3) \\
&\leq e^{-0.5\delta_1 t^{\gamma_1}} + e^{-0.5\delta_2 t^{\gamma_1}} + e^{-\delta_3 t^{\gamma_0 \gamma_1}}, \ \forall t > T \\
&= z_t, \ \forall t > T
\end{aligned} \tag{1.20}$$

For $t \leq T$ set $z_t = 1$. Proposition 2.10 shows that $\sum_{t=0}^{\infty} z_t < \infty$. The Lemmas are stated and proved in Chapter 2. □

1.4 Convergence of posterior expectations

In Bayesian analysis, most inference exercises involve computing posterior expectations of some functions. For example, the posterior mean and the posterior standard deviation of a parameter and the posterior probability that a parameter belongs to a set can all be expressed in terms of posterior expectations. More importantly, the answers to the policy questions that DDCMs address also take this form. Examples of such policy questions for the models I use in experiments include: (i) in Gilleskie's model, investigators might be interested in how the average number of doctor visits and/or work absences would be affected by changes in the coinsurance rates and in the proportion of the wage that sick leave replaces; (ii) in Rust's model, investigators could care about how the annual number of bus engine replacements would be affected by a change in the engine replacement cost. Using the uniform complete convergence of the approximations of the expected value functions, I prove the complete convergence of the approximated posterior expectations under mild assumptions on a kernel of the posterior distribution.

Assumption 1.8. *Assume that $\epsilon_{t,i} \in E$, $\theta \in \Theta$, and $\nu_{t,k,i} \in [-\overline{\nu}, \overline{\nu}]$, where $\nu_{t,k,i}$ denotes the k^{th} component of $\nu_{t,i}$. Let the joint posterior distribution of the parameters and the latent variables be proportional to a product of a continuous function and indicator functions:*

$$\begin{aligned} p(\theta, \mathcal{V}, \epsilon; F|d, x) \quad &\propto \quad r(\theta, \mathcal{V}, \epsilon; F(\theta, \epsilon)) \cdot 1_{\Theta}(\theta) \cdot \left(\prod_{i,t} 1_E(\epsilon_{t,i}) p(d_{t,i}|\mathcal{V}_{t,i}) \right) \\ &\cdot \quad \left(\prod_{i,t,k} 1_{[-\overline{\nu},\overline{\nu}]}(q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}(\theta, \epsilon_{t,i}))) \right) \end{aligned} \tag{1.21}$$

where $r(\theta, \mathcal{V}, \epsilon; F)$ and $q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i})$ are continuous in $(\theta, \mathcal{V}, \epsilon, F)$, $F = \{F_{t,d,i}, \forall i, t, d\}$ stands for a vector of the expected value functions, and $F_{t,i}$ are the corresponding subvectors. Also, assume that the level curves of $q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i})$ corresponding to $\overline{\nu}$ and $-\overline{\nu}$ have zero Lebesgue measure:

$$\lambda[(\theta, \mathcal{V}, \epsilon) : q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) = \overline{\nu}] = \lambda[(\theta, \mathcal{V}, \epsilon) : q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) = -\overline{\nu}] = 0 \tag{1.22}$$

This assumption is likely to be satisfied for most models formulated on a bounded state space, in which truncation is used for distributions with unbounded support. If in the two examples from the next section the Gaussian distributions were truncated to satisfy the boundedness requirements of the theorems: $\nu_{t,d,i}$, $\epsilon_{t,i}$, and the prior for θ were truncated to bounded sets $[-\overline{\nu}, \overline{\nu}]$, E, and Θ, then the kernels of the joint distribution for both models would have the form in (1.21). Condition (1.22) is also easy to verify. In both models, $q_d(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) = \Delta u(x_{t,i}, d) + \epsilon_{t,d,i} + F_{t,d,i}(\theta, \epsilon_{t,i}) - \mathcal{V}_{t,d,i} = \overline{\nu}$ defines a continuous function $\mathcal{V}_{t,d,i} = \Delta u(x_{t,i}, d) + \epsilon_{t,d,i} + F_{t,d,i}(\theta, \epsilon_{t,i}) - \overline{\nu}$. Since the Lebesgue measure of the graph of a continuous function is zero, (1.22) will be satisfied.

Theorem 1.3. *Let $h(\theta, \mathcal{V}, \epsilon)$ be a bounded function. Under Assumptions 1–1.8, the expectation of $h(\theta, \mathcal{V}, \epsilon)$ with respect to the approximated posterior that uses the DP solution approximations $\hat{F}^n$ from step n of the DP solving algorithm converges completely (and thus a.s.) to the true posterior expectation of $h(\theta, \mathcal{V}, \epsilon)$ as $n \to \infty$: for any $\varepsilon > 0$ there exists a sequence $\{z_n\}$ such that $\sum_{n=0}^{\infty} z_n < \infty$ and*

$$\begin{aligned} P\Bigg(\bigg| \int & h(\theta, \mathcal{V}, \epsilon) p(\theta, \mathcal{V}, \epsilon; F|d, x) d(\theta, \mathcal{V}, \epsilon) \\ & - \int h(\theta, \mathcal{V}, \epsilon) p(\theta, \mathcal{V}, \epsilon; \hat{F}^n|d, x) d(\theta, \mathcal{V}, \epsilon) \bigg| > \varepsilon \Bigg) \leq z_n \end{aligned} \tag{1.23}$$

The proof is given in Chapter 2, Section 2.3. The theorem can be extended to the case when we are interested in $p(W|x, d)$, where W is called the object of interest, see Geweke (2005); in particular, W can denote the answer to a policy question. If the implications of the model for W are specified by a density $p(W|\theta, \mathcal{V}, \epsilon, d, x)$, then

$$p(W|d,x) = \int p(W|\theta, \mathcal{V}, \epsilon, d, x) p(\theta, \mathcal{V}, \epsilon; F|d, x) d(\theta, \mathcal{V}, \epsilon) \tag{1.24}$$

If $p(W|\theta, \mathcal{V}, \epsilon, d, x)$ has the same properties as the kernel of $p(\theta, \mathcal{V}, \epsilon; F|d, x)$ in Assumption 1.8, then the theorem holds for $p(W|x, d)$.

1.5 Experiments

To implement the algorithm I wrote a program in C. The program uses BACC[5] interface to libraries LAPACK, BLAS, and RANLIB for performing matrix operations and random variates generation. Higher level interpreted languages like Matlab would not provide necessary computation speed since the algorithm cannot be sufficiently vectorized. As a matter of future work, the algorithm could be easily parallelized with very significant gains in speed (this is is not necessarily possible or easy for an arbitrary algorithm.) A short discussion of algorithm parallelization is given in Section 1.5.2.5.

1.5.1 Gilleskie's (1998) model

1.5.1.1 Setup

For experiments, I used a simplified version of Gilleskie's model. Only one type of sickness was included and some parameters were fixed. For the extreme value iid process for taste shocks in the original model I substituted a serially correlated process.

[5] BACC is an open source software for Bayesian Analysis, Computation, and Communication available at www2.cirano.qc.ca/˜bacc

In the model, an agent can be sick or well. If sick, every period she has the following alternatives to choose from: $d = 1$ – work and do not visit a doctor, $d = 2$ – work and visit a doctor, $d = 3$ – do not work and do not visit a doctor, $d = 4$ – do not work and visit a doctor. The observed state x for a sick agent includes: t – the time since the illness started, v_t – the number of doctor visits since the illness started, and a_t – the number of work absences accumulated since the illness started. For a well agent $x = (0, 0, 0)$.

The per-period utility function of a well agent is equal to her income Y, which is known; so, the marginal utility of consumption when well is fixed to 1. The per-period utility function of an ill agent is additively separable in the unobserved state variables $y_t = \{y_{t,d}, d \in D\}$ and linear in parameters:

$$u(x_t, y_t, d) = z(x_t, d) \cdot \alpha + y_{t,d}$$

where $\alpha = (\alpha_1, \alpha_2, \alpha_3, \alpha_4)$, α_1 is the disutility of illness, α_2 is the direct utility of doctor visit, α_3 is the direct utility of attending work when ill, and α_4 is the marginal utility of consumption when ill. As a function of the observed state and the decision, the 1×4 matrix $z(x_t, d)$ is given by

$$z(x_t, 1) = (1, 0, 1, Y),\ z(x_t, 2) = (1, 1, 1, Y - C)$$

$$z(x_t, 3) = (1, 0, 0, Y\phi(a_t + 1)),\ z(x_t, 4) = (1, 1, 0, Y\phi(a_t + 1) - C)$$

where C is a known out-of-pocket cost of a doctor visit; $\phi(a_t)$ is a proportion of the daily wage that sick leave replaces for the accumulated number of absences a_t. In the original model $\phi(a_t)$ depends on some parameters; here, I just fix those.

The unobserved states in the model are interpreted as taste shocks. As was discussed in Section 1.2, the unobserved state variables should include some serially conditionally independent components in addition to the serially correlated ones so that the joint distribution of the data, the parameters, and the latent variables could be specified analytically. I chose a very simple specification of the unobserved states that satisfies this condition:

$$y_{t,d} = \epsilon_{t,d} + \nu_{t,d}$$

where $\nu_{t,d}$ is iid $N(0, h_\nu^{-1})$, $\epsilon_{t,d}$ is $N(\rho\epsilon_{t-1,d}, h_\epsilon^{-1})$ and $\epsilon_{0,d} = 0$. The structure of interdependence between the unobserved state variables could be more general and it is a subject for future work.

The probability of contracting a sickness π^s is assumed to be known. The probability of getting well for a sick agent $\pi(x_t, d, \eta)$ depends on the parameters η and is given by

$$\pi(x_t, d, \eta) = \begin{cases} \Phi(\eta e_{t+1}) & \text{if } t = 1, \ldots, T-1 \\ 1 & \text{if } t = T \end{cases} \tag{1.25}$$

where $\Phi(.)$ is a standard normal cdf and

$$\eta e_{t+1} = \eta_1 + \eta_2 v_{t+1} + \eta_3 a_{t+1} + \eta_4 t \tag{1.26}$$

The maximum sickness duration is T. For $t < T$ the transition could be described by a probit model with an unobserved recovery index RI_t:

$$RI_{t+1} = \eta e_{t+1} + N(0, 1) \tag{1.27}$$

Conditional on RI_{t+1} the transition for x_t is deterministic:

$$x_{t+1}|x_t, d, RI_{t+1} = \begin{cases} (0,0,0), \text{ if } RI_{t+1} > 0 \text{ or } t = T \\ x'(x_t, d) = (t+1, a_t + 1_{\{3,4\}}(d), v_t + 1_{\{2,4\}}(d)), \text{ otherwise} \end{cases} \tag{1.28}$$

where $x'(.)$ denotes the future state as a deterministic function of the current state and the decision given that the agent remains sick.

The life-time value of being well is

$$V_w = Y + \beta(1 - \pi^s)V_w + \beta\pi^s EV(x_1, y_1) \tag{1.29}$$

where $x_1 = (1, 0, 0)$.

The lifetime value of being sick is

$$V(x_t, y_t) = \max_{d \in D} \mathcal{V}(x_t, y_t, d) \tag{1.30}$$

$$\begin{aligned} \mathcal{V}(x_t, y_t, d) &= u(x_t, y_t, d) + \beta\pi(x_t, d, \eta)V_w \\ &+ \beta(1 - \pi(x_t, d, \eta))E[V(x'(x_t, d), y_{t+1})|\epsilon_t; \theta] \end{aligned} \tag{1.31}$$

1.5.1.2 Gibbs sampler

In the model formulation above, the assumed distributions for the unobserved states have unbounded support. It is also more convenient to use distributions with unbounded support in constructing the Gibbs sampler. To reconcile this with the theory, which requires the parameters and the states to be in bounded spaces, we could assume the existence of bounds for all the parameters and the states. If these bounds are large enough, then the Gibbs sampler that takes them into account would produce the same results as the Gibbs sampler that does not. Thus, not to clutter the notation I present the Gibbs sampler assuming no bounds. For an example of the Gibbs sampler that imposes the bounds see the Gibbs sampler for Rust's model in Section 1.5.2.3.

For each individual i, one illness episode of length T_i is observed. The observables in the model are $\{x_{t,i}\}_{t=1}^{T_i+1}$ and $\{d_{t,i}\}_{t=1}^{T_i}$ for $i = 1, \ldots, I$. The parameters are $\theta = (\alpha, \eta, \rho, h_\epsilon)$; h_ν is fixed for normalization. From experiments with the DP solution (Section 1.5.1.4,) I find that the approximation error for the expected value functions $\hat{E}^m[V(s';\theta)|s,d;\theta]$ is much bigger than for the differences of expectations $\hat{E}^m[V(s';\theta)|s,d_1;\theta] - \hat{E}^m[V(s';\theta)|s,d_2;\theta]$. Thus, instead of using $\mathcal{V}_{t,d,i}$ as latent variables in the estimation procedure I use the following latent variables:

$$\begin{aligned} \Delta\mathcal{V}_{t,d,i} &= \mathcal{V}_{t,d,i} - z(x_{t,i}, \overline{d})\alpha - \beta E[V(s';\theta)|\epsilon_{t,i}, x_{t,i}, \overline{d};\theta] \\ &= \Delta z_{t,d,i}\alpha + \epsilon_{t,d,i} + \nu_{t,d,i} + F_{t,d,i}(\theta, \epsilon_{t,i}) \qquad (1.32) \end{aligned}$$

where $\overline{d}$ is some fixed alternative in D, $\Delta z_{t,d,i} = z(x_{t,i}, d) - z(x_{t,i}, \overline{d})$, and

$$F_{t,d,i}(\theta, \epsilon_{t,i}) = \beta E[V(s';\theta)|\epsilon_{t,i}, x_{t,i}, d;\theta] - \beta E[V(s';\theta)|\epsilon_{t,i}, x_{t,i}, \overline{d};\theta] \qquad (1.33)$$

Note that $\Delta\mathcal{V}_{t,d,i}$ is not a difference of alternative specific value functions. If it were then the Gibbs sampler blocks for $\Delta\mathcal{V}_{t,d,i}$ would be more complicated.

In addition, $\epsilon = \{\epsilon_{t,d,i}\}_{t=1}^{T_i}$ and $\{RI_{t,i}\}_{t=1}^{T_i+1}$ are also treated as latent variables.

The joint distribution of the data, the parameters, and the latent variables is

$$\begin{aligned}
&p(\theta; \{d_{t,i}; \Delta\mathcal{V}_{t,1,i}, \ldots, \Delta\mathcal{V}_{t,D,i}; \epsilon_{t,1,i}, \ldots, \epsilon_{t,D,i}\}_{t=1}^{T_i}; \{x_{t,i}; RI_{t,i}\}_{t=1}^{T_i+1}; i = 1, \ldots, I) = \\
&p(\theta) \prod_{i=1}^{I} \prod_{t=1}^{T_i} [p(x_{t+1,i}|x_{t,i}; d_{t,i}; RI_{t+1,i}) p(RI_{t+1,i}|x_{t,i}; d_{t,i}; \eta) \\
&p(d_{t,i}|\Delta\mathcal{V}_{t,1,i}, \ldots, \Delta\mathcal{V}_{t,D,i}) \prod_{d=1}^{D} p(\Delta\mathcal{V}_{t,d,i}|x_{t,i}, \epsilon_{t,i}; \theta) p(\epsilon_{t,d,i}|\epsilon_{t-1,d,i}, \rho, h_\epsilon)] \qquad (1.34)
\end{aligned}$$

where $p(\theta)$ is a prior density for parameters; $x_{0,i} = \emptyset$; $p(d_{t,i}|\Delta\mathcal{V}_{t,1,i}, \ldots, \mathcal{V}_{t,D,i})$ is an indicator function, which is equal to 1 when $\Delta\mathcal{V}_{t,d_{t,i},i} \geq \Delta\mathcal{V}_{t,d,i}, \forall d$.

$$p(\Delta\mathcal{V}_{t,d,i}|x_{t,i}, \epsilon_{t,i}; \theta) \propto \exp\{-0.5 h_\nu (\Delta\mathcal{V}_{t,d,i} - \Delta z_{t,d,i}\alpha - \epsilon_{t,d,i} - F_{t,d,i}(\theta, \epsilon_{t,i}))^2\}$$

$$p(\epsilon_{t,d,i}|\epsilon_{t-1,d,i}, \theta) \propto h_\epsilon^{-1/2} \exp\{-0.5 h_\epsilon (\epsilon_{t,d,i} - \rho\epsilon_{t-1,d,i})^2\}$$

Gibbs sampler blocks

The block for $\Delta\mathcal{V}_{t,d,i}| \ldots$ is $N(\Delta z_{t,d,i}\alpha + \epsilon_{t,d,i} + F_{t,d,i}(\theta, \epsilon_{t,i}), h_\nu)$ truncated to $\Delta\mathcal{V}_{t,d_{t,i},i} \geq \Delta\mathcal{V}_{t,\tilde{d},i} \forall \tilde{d} \in D$. The block for $RI_{t+1,i}| \ldots$ is $N(e_{t+1,i}\eta, 1)$ truncated to $(0, \infty)$ if $x_{t+1,i} = (0, 0, 0)$ and to $(-\infty, 0)$ otherwise, where $e_{t+1,i}$ is a vector depending on $x_{t,i}$ and $d_{t,i}$ that was defined in (1.26). The density for $\epsilon_{t,d,i}| \ldots$ block:

$$\begin{aligned}
p(\epsilon_{t,d,i}| \ldots) \;&\propto\; \exp\{-0.5 h_\nu (\Delta\mathcal{V}_{t,d,i} - F_{t,d,i}(\theta, \epsilon_{t,i}) - \Delta z_{t,d,i}\alpha - \epsilon_{t,d,i})^2\} && (1.35) \\
&\times\; \exp\{-0.5 h_\nu \sum_{\tilde{d} \neq d} (\Delta\mathcal{V}_{t,\tilde{d},i} - F_{t,\tilde{d},i}(\theta, \epsilon_{t,i}) - \Delta z_{t,d,i}\alpha - \epsilon_{t,\tilde{d},i})^2\} && (1.36) \\
&\times\; \exp\{-0.5 h_\epsilon (\epsilon_{t+1,d,i} - \rho\epsilon_{t,d,i})^2 - 0.5 h_\epsilon (\epsilon_{t,d,i} - \rho\epsilon_{t-1,d,i})^2\} && (1.37)
\end{aligned}$$

To draw from this density I use a Metropolis step with a normal transition density proportional to (1.37). Blocks for $\epsilon_{t,d,i}$ with $t = 0$ and $t = T_i$ will be similar. Blocks for $\alpha| \ldots$, $\eta| \ldots$, $\rho| \ldots$, $h_\epsilon| \ldots$ are drawn by the Metropolis-Hastings (MH) random walk algorithm since an analytical expression for the difference in expected value functions $F_{t,d,i}(\theta, \epsilon)$ is unknown and it could only be approximated numerically. The proposal density of the MH random walk algorithm is normal with mean equal to the current parameter draw and a fixed variance. The variances are chosen so that the acceptance probability would be between $0.2 - 0.8$. If a vector of parameters is drawn as one block by the MH random walk it is important to make the variances for all the components of the vector as large as possible keeping the acceptance rate reasonable. Nevertheless, reasonable acceptance rates do not guarantee fast convergence. While drawing vector α by the MH as one block worked well, drawing η as one block resulted in too slow mixing of the chain. Thus, on every other iteration the components of η are drawn one at a time. This significantly accelerated convergence. For larger sample sizes ($I = 1000$,) acceptance rates in the range 0.2-0.3 worked the best. For Rust's model, I explore an alternative to the random walk chain, in which the MH transition densities are proportional to the familiar parts of the posterior. This alternative seems to work remarkably well for the state transition parameters that are strongly identified by the data (see Section 1.5.2.)

1.5.1.3 Approximating the value functions

The sequential structure of the model was exploited in computing the approximations of the value functions. In experiments, only one nearest neighbor was picked for approximating the expectations: $\tilde{N}(m) = 1$. First, a random grid $\{y^{m,j} = (\nu^{m,j}, \epsilon^{m,j})\}_{j=1}^{\hat{N}(m)}$ is generated on the continuous part of the state space:

$\nu_d^{m,j} \sim N(0, h_\nu^{-1})$ and $\epsilon^{m,j} \sim g(.)$, where $g(.)$ is normal with zero mean and the precision equal to the prior mean of h_ϵ. The approximations of the value functions for each $x \in X$ and $y^{m,j}$, $j = 1, \ldots, \hat{N}(m)$ are computed as follows:

$$\begin{aligned} V^m(x, y^{m,j}; \theta^m) &= \max_{d \in D} u(x, y^{m,j}, d, \alpha^m) \\ &+ \beta\pi(x, d, \eta^m)V_w^{k_1} + \beta(1 - \pi(x, d, \eta^m)) \times \\ &\times \sum_{i=1}^{\hat{N}(m)} V^m(x'(x, d), y^{m,i}; \theta^m)\frac{f(\epsilon^{m,i} \mid \epsilon^{m,j}; \theta^m)/g(\epsilon^{m,i})}{\sum_{r=1}^{\hat{N}(m)} f(\epsilon^{m,r} \mid \epsilon^{m,j}; \theta^m)/g(\epsilon^{m,r})} \end{aligned} \tag{1.38}$$

where $f(.|\epsilon; \theta)$ is a $N(\rho\epsilon, h_\epsilon^{-1})$ density. Note that $\nu_d^{m,j}$ have the same distribution as $\nu_{t,d}$ in the model. Thus, the corresponding density values cancel each other in the numerator and denominator of the importance sampling weight. The approximations of the value functions with larger t are computed first. That is why only the value functions already updated at the current iteration m, $V^m(.; \theta^m)$ (as opposed to $V^{k_1}(.; \theta^{k_1})$,) are used for approximating the expectations in (1.38). Note, that for x with $t = T$, the recovery is certain, $\pi(x, d, \eta^m) = 1$, and only $V_w^{k_1}$ is required for computing the expectation. This procedure is similar to the backward induction or the Gauss-Seidel method.

After (1.38), the approximation of the value of being well is computed.

$$\begin{aligned} V_w^m &= [1/(1 - \beta(1 - \pi^s))][Y + \\ &+ \beta\pi^s \sum_{i=1}^{\hat{N}(m)} V^m((1, 0, 0), y^{m,i}; \theta^m)\frac{f(\epsilon^{m,i}|0; \theta^m)/g(\epsilon^{m,i})}{\sum_{r=1}^{\hat{N}(m)} f(\epsilon^{m,r}|0; \theta^m)/g(\epsilon^{m,r})}] \end{aligned} \tag{1.39}$$

Experiments with a sequence of θ^m, which was drawn from a prior distribution one component of θ at a time, showed that performing only one Bellman equation iteration might not provide a sufficient approximation precision for feasible run times.

For $N(m) = 1000$ and $\hat{N}(m) = 100$ the average approximation error for $F_{t,d,i}(\theta, \epsilon)$ was three times as large as the standard deviation of the taste shocks $h_\epsilon^{-.5}$. The approximation error for the kernel smoothing algorithm of Imai et al. (2005) was on average twice as large as for the nearest neighbors algorithm[6].

The approximation precision could be improved by repeating (1.38) and (1.39) several times. For that purpose we can separate iterations of the Gibbs sampler and the DP solving algorithm. For each iteration m of the Gibbs sampler we perform several iterations of the DP solving algorithm keeping the parameter vector fixed at θ^m. Only the approximations of the value functions obtained on the last repetition are used for approximating the expectations in the Gibbs sampler at iteration m. Note that for $\tilde{N}(m) = 1$ this procedure could still fit the proposed theoretical framework with the modification that at each iteration of the DP solving algorithm the parameter vector is drawn with a small probability p from a density $p(\theta) > 0$ on Θ or, otherwise, taken to be the current Gibbs sampler draw θ^m with a probability $1 - p$. This augmentation would guarantee that Assumption 1.5 holds.

The value function iteration algorithm has linear convergence rates and convergence may slow down significantly near the fixed point. That is why employing the following non-linear optimization procedure might help in obtaining a good approximation precision at reduced computational costs. Performing one iteration of the DP solving algorithm (computations in (1.38) and (1.39)) for the fixed parameter

[6] Imai et al. (2005) do not provide numerical results characterizing the accuracy of their DP solution approximations. It might be possible to improve the results obtained here for the kernel smoothing algorithm by varying the kernel smoothing band width.

vector θ^m and the random grid $\{y^{m,j}\}_{j=1}^{\hat{N}(m)}$ could be seen as a mapping that takes V_w as an input and updates it. Iterating this mapping produces a sequence of V_w's that converges monotonically. Taking this into account improves the performance. Figure 1.1 presents a flowchart of a direct search procedure for finding the mapping fixed point V_w^m.

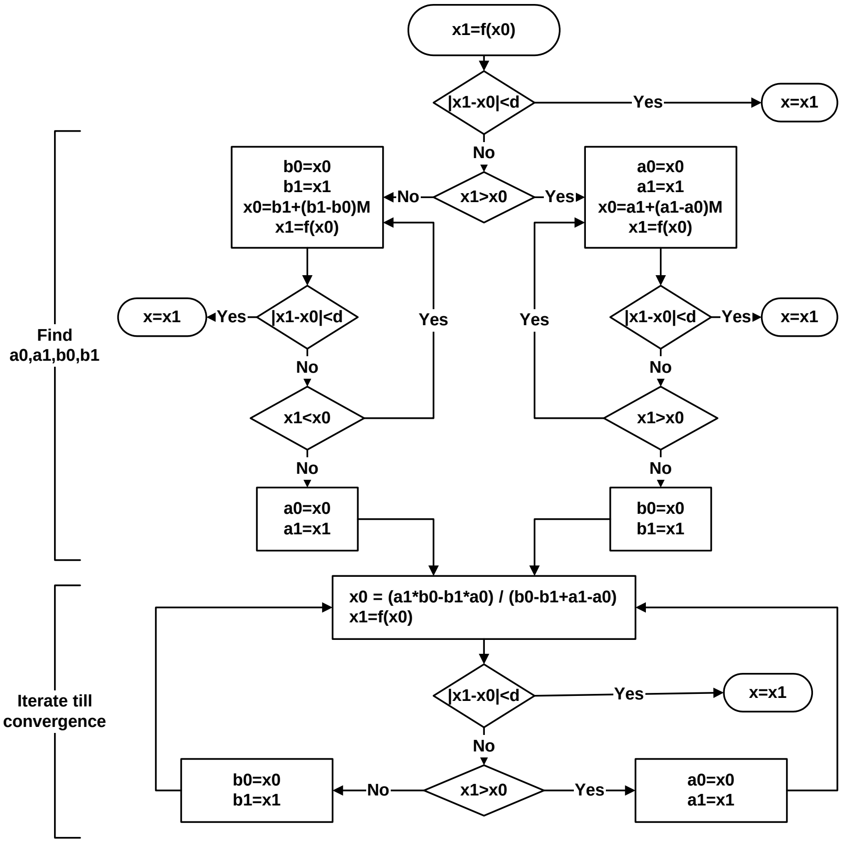

Figure 1.1: Flowchart of a direct search procedure for finding a fixed point.

In the flowchart, $f(.)$ denotes a mapping that takes V_w as an input and returns an updated value of V_w iterating the Bellman equations once. The algorithm searches for a fixed point $x = f(x)$. First, the algorithm finds bounds a_0, a_1, b_0, b_1: $a_0 < a_1 = f(a_0) \leq x \leq b_1 = f(b_0) < b_0$ starting with x_0. A scaling factor M is chosen experimentally. Updated x is obtained by cutting interval $[a_1, b_1]$ in proportions $(a_1 - a_0) : (b_0 - b_1)$. After each iteration the difference $f(x_0) - x_0$ is compared to a tolerance parameter d. If the convergence has not been achieved a_0, a_1, b_0, b_1 are updated and the procedure is repeated. This procedure is used in the estimation experiments presented below. In these experiments, starting from the nearest neighbor, the procedure required only 2-4 passages over (1.38) and (1.39) to find the fixed point V_w^m or, equivalently, to solve the DP for θ^m on the random grid $\{y^{m,j}\}_{j=1}^{\hat{N}(m)}$.

Since the Gibbs sampler changes only one or few components of the parameter vector at a time, the previous parameter draw θ^{m-1} turned out to be the nearest neighbor of the current parameter θ^m in most cases. Taking advantage of this observation and keeping track only of one previous iteration saves a significant amount of computer memory.

In the Gibbs sampler, the approximations of the differences in the expectations are computed as follows:

$$\begin{aligned}
&F_{t,d,i}(\theta^m, \epsilon_{t,i}^m) = \\
&\quad \beta(\pi(x_{t,i}, d, \eta^m) - \pi(x_{t,i}, \overline{d}, \eta^m))V_w^m \\
&+ \quad \beta(1 - \pi(x_{t,i}, d, \eta^m)) \sum_{i=1}^{\hat{N}(m)} V^m(x'(x_{t,i}, d), y^{m,i}; \theta^m) \frac{f(\epsilon^{m,i} \mid \epsilon_{t,i}^m; \theta^m)/g(\epsilon^{m,i})}{\sum_{r=1}^{\hat{N}(m)} f(\epsilon^{m,r} \mid \epsilon_{t,i}^m; \theta^m)/g(\epsilon^{m,r})} \\
&- \quad \beta(1 - \pi(x_{t,i}, \overline{d}, \eta^m)) \sum_{i=1}^{\hat{N}(m)} V^m(x'(x_{t,i}, \overline{d}), y^{m,i}; \theta^m) \frac{f(\epsilon^{m,i} | \epsilon_{t,i}^m; \theta^m)/g(\epsilon^{m,i})}{\sum_{r=1}^{\hat{N}(m)} f(\epsilon^{m,r} | \epsilon_{t,i}^m; \theta^m)/g(\epsilon^{m,r})}
\end{aligned}$$

1.5.1.4 Experiments with DP solution

A simulation study was conducted to assess the quality of the DP solution approximations. The study explores how the randomness of the grid affects the approximations for fixed parameters and how these effects change with the random grid size. The parameter values for this experiment are the same as for the estimation experiments described below in Section 1.5.1.5. First, I generated 1000 random grids $\{y^{m,j}\}_{j=1}^{\hat{N}}$, $m = 1, \ldots, 1000$. Then, for each random grid m, I solved the DP as described in the previous section and computed the approximation of the value of being well V_w^m and the approximation of the difference in expectations $\hat{E}^{(m)}[V(s';\theta)|s, d_1;\theta] - \hat{E}^{(m)}[V(s';\theta)|s, d_2;\theta]$.

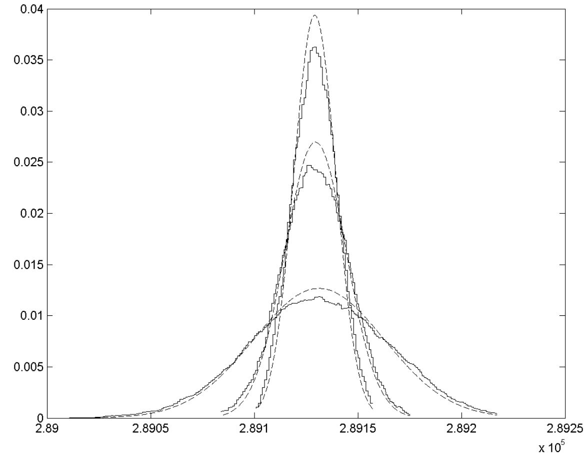

Figure 1.2: Estimated densities of V_w. The tightest density corresponds to $\hat{N} = 1000$, the most widespread to $\hat{N} = 100$. The dashed lines are fitted normal densities.

The approximation V_w^m is a measurable function of the random grid realization $\{y^{m,j}\}_{j=1}^{\hat{N}}$ and thus itself is a random variable. Using kernel smoothing, I estimated densities of those approximations. The estimated densities for $\hat{N} = 100, 500, 1000$ are presented in Figures 1.2 and 1.3.

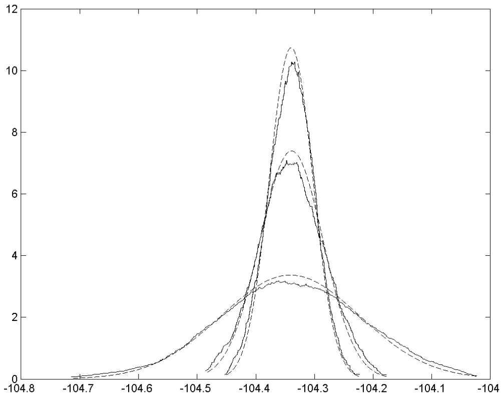

Figure 1.3: Estimated densities of $\hat{E}^{(m)}[V(s';\theta)|s,d_1;\theta] - \hat{E}^{(m)}[V(s';\theta)|s,d_2;\theta]$. The tightest density corresponds to $\hat{N} = 1000$, the most widespread to $\hat{N} = 100$. The dashed lines are fitted normal densities.

Visual inspection of the figures suggests that the approximations converge as the number of the points in the random grid increases. The mean of the distribution seems to be the same for $\hat{N} = 100, 500, 1000$. The variances are roughly proportional to $\hat{N}^{-1}$. The densities are close to the fitted normal densities. All this hints that an analog to a CLT might hold for this problem. Comparison of the two figures shows that the maximal approximation error for the expected value function is larger by two orders of magnitude than the maximal approximation error for the difference in the expected value functions. This result seems to have a simple intuitive explanation. An approximal DP solution computed on a random grid could be far from the actual solution. However, the errors resulting from discretization and numerical integration very similarly affect the approximations of the future expected value functions for the same current state but different decisions. It probably happens because numerical integration over the future states is performed on the same random grid no matter which alternative is chosen in the current period. Thus, the approximations of the expected value functions have very high positive correlation and their variances are of similar magnitude. This results in a small variance for their difference. As I mentioned earlier, these findings motivate the choice of the Gibbs sampler parameterization, in which only the differences of the expected value functions are used.

Comparing the approximation precision with the magnitude of the taste shocks in the model seems to be a reasonable way of judging the approximation quality. The maximal approximation error for the differences in the expected value functions for $\hat{N} = 100$ was smaller than the standard deviation of the taste shocks $h_\nu^{-1} = 10$ by a factor of $15 - 30$.

To further verify that the method is implemented correctly I conducted a similar simulation study using the extreme value iid unobservables instead of the serially correlated unobservables. The results were analogous to the ones reported in the figures. The actual DP solution for the iid extreme value unobservables can be easily computed with a high precision as described in Rust (1994). As expected, the exact solutions were right at the means of the distributions obtained from the simulation study.

These experiments also suggest an improvement in the algorithm performance. Solving the DP on several small random grids and combining the results seems to be a very efficient alternative to using one large grid. I separate the series of the approximations of V_w for $\hat{N} = 100$ into batches of size 10. For each batch I compute the mean and then use these means in kernel smoothing to obtain the estimated density for such approximations of V_w. The resulting density practically coincide with the density obtained for $\hat{N} = 1000$ and no batching. Thus, the approximation precision for these procedures is about the same. The time of iterating the Bellman equation on a grid of size $\hat{N}$ is proportional to $\hat{N}^2$. Therefore, the time required for iterating the Bellman equation on a grid of size $\hat{N} = 100$ for ten different grids will be smaller by a factor of 10 than the time required for iterating the Bellman equation on one grid of size $\hat{N} = 1000$. These experimental results are intriguing. Investigating theoretical properties of this improved procedure, e.g. deriving complexity bounds, seems to be of great interest and is a subject of future work. This improvement has not been incorporated into the estimation experiments in this chapter. However, I employ it in Chapter 3 that uses artificial neural networks to approximate the expected value function as a function of the parameters θ and the state variables.

1.5.1.5 Estimation for artificial datasets

The generated sample contained $I = 100$ observations. The maximal length of an illness episode was $T = 5$, the standard deviation of the uncorrelated taste shocks was $h_\nu^{-0.5} = 10$, and the time discount factor was $\beta = 0.9997$. The size of the random grid for solving the DP was equal to $\hat{N} = 100$, and the number of the picked nearest neighbors was equal to $\tilde{N} = 1$. Data generation and each iteration of the estimation procedure use the same random grid for solving the DP (Proposition 2.6 justifies using the same random grid at each iteration of the algorithm if the number of the nearest neighbors is constant.) Experiments with different grids are conducted on real data for the Rust (1987)'s model. The approximation error for the differences in the expected value functions was smaller than the standard deviation of the taste shocks by a factor of $15 - 30$. Under these settings, it takes about 30 seconds to produce 100 draws from the posterior on a 2002 vintage PC. The priors are specified together with estimation results in Table 1.1.

Table 1.1: Estimation results for artificial data

Param-	True	Posterior			Prior
eter	value	Mean	SD	NSE	
α_1	-1000	-1225.1	458.26	14.931	$N(-1000, 3333.3^2)$
α_2	-50	-0.5944	174.96	5.6386	$N(-50, 333.3^2)$
α_3	90	87.298	120.52	5.0465	$N(90, 333.3^2)$
α_4	0.2	1.939	5.8549	0.1924	$N(0.2, 16.7^2)$
ρ	0.7	0.66199	0.1368	0.0068	$N(0.5, 1000^2)$, s.t.[0, 0.99]
η_1	-4.5	-4.5983	0.3107	0.0479	$N(-4.5, 0.67^2)$
η_2	0.1	0.09767	0.0192	0.0036	$N(0.1, 0.067^2)$
η_3	0.1	0.13571	0.0348	0.0101	$N(0.1, 0.067^2)$
η_4	1.5	1.4872	0.1112	0.0187	$N(1.5, 0.67^2)$
$h_\epsilon^{-0.5}$	20	19.4959	5.2168	0.67	$800\chi^2_2$, mean=sd=20

The data and the parameters that were fixed: $Y = 98$, $C = 30$, $\pi^s = 0.0034$, $\phi(a) = 1/(1 + \exp(-20 + 5a))$, and $\overline{d} = 4$. The parameter values used for data simulation, the priors and the posteriors are also presented graphically. The parameter values for data simulation were chosen so that all the decisions and most of the possible observed states $x \in X$ were present in the simulated data. The chain convergence is checked by comparing the estimation results for several different posterior simulator runs. Figure 1.4 gives estimated posterior marginal densities of the parameters for five different posterior simulator runs that were started from random initial values.

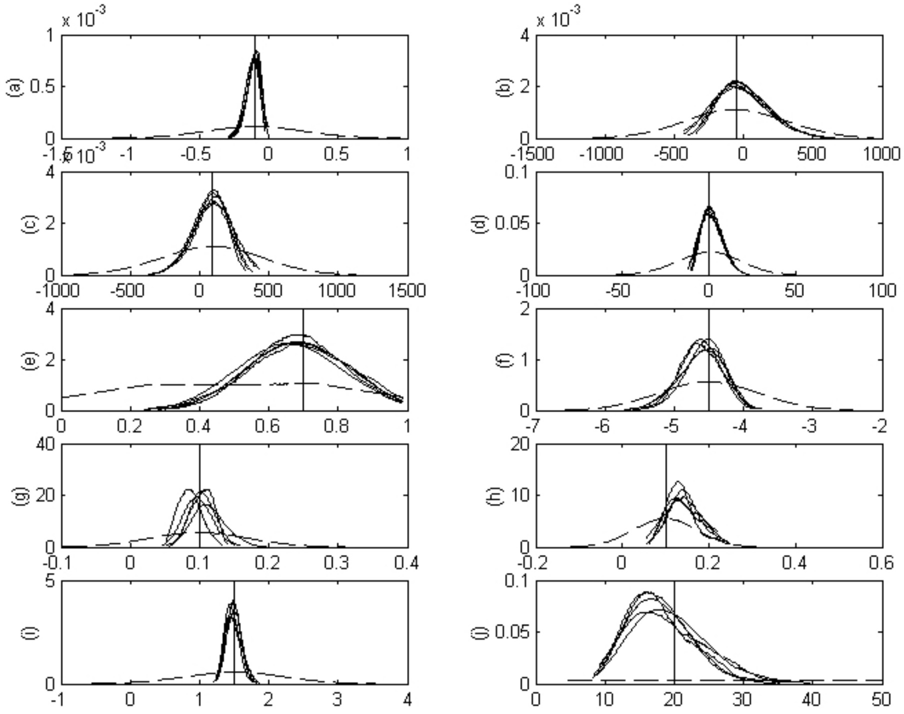

Figure 1.4: Estimated posterior densities of (a) α_1, (b) α_2, (c) α_3, (d) α_4, (e) ρ, (f) η_1, (g) η_2, (h) η_3, (k) η_4, (l) $h_\epsilon^{-1/2}$. The dashed lines are prior densities. The vertical lines show the actual parameter values.

The length of the runs was $300000 - 1000000$ draws. The posterior densities estimated by kernel smoothing seem to converge to the same stationary distributions for different posterior runs.

As can be seen from the figure, α and ρ converge much faster than η and h_ϵ. Overall, the amount of serial correlation in the sampler draws is quite large. This is typical for the Metropolis-Hastings random walk algorithm. The situation might be complicated here by the presence of a lot of latent variables and by the sensitivity of the expected value functions to changes in parameters. Thus, long simulator runs are necessary to estimate the posterior distributions with sufficient precision. Also, some experimental work is required for choosing the variance of the transition densities for the MH random walk.

Another apparent feature of the estimation results is that the uncertainty about the parameter values is huge. One reason being that the model is not very parsimonious and the parameters are weakly identified. Changes in different parameters might lead to similar changes in the observables. This also creates difficulties for classical maximum likelihood estimation of dynamic discrete choice models. The standard errors of the parameter estimates are often large and difficult to compute precisely. In contrast to the classical approach, Bayesian inference takes into account the uncertainty about the parameters and this advantage seems to be important for dynamic discrete choice models. As will be seen in the experiments with Rust's model, an increase in the sample size does not necessarily cure the problem of weak identification.

1.5.1.6 Two stage estimation

The following experiment is motivated by the two stage classical estimation procedure in which the state transition parameters η are estimated first from the partial likelihood and then these estimates are used in the estimation of the rest of the parameters. The experiment below could give some clues on how the full maximum likelihood estimation results would differ from the ones of the two stage procedure. Using the priors and the artificial dataset from the experiments in the previous section, I estimate η by a standard probit model with the assumption that $x_{t,i}$ are fixed. The results of this procedure are contrasted with the full model estimation results in Figure 1.5.

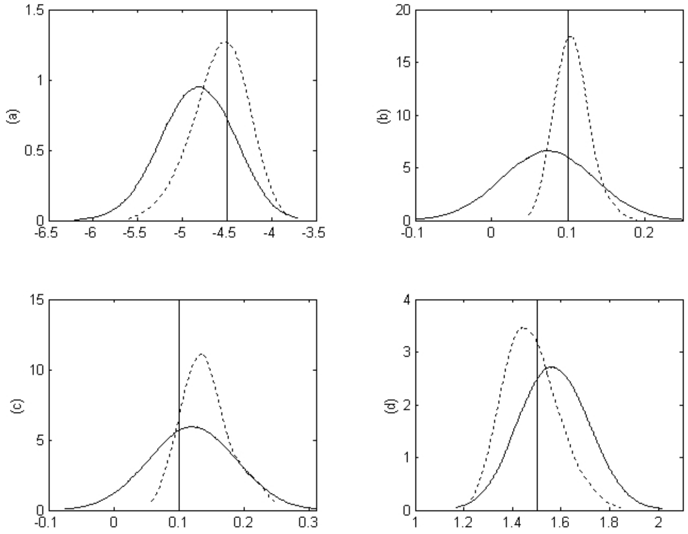

Figure 1.5: Posterior densities of η: (a) η_1, (b) η_2, (c) η_3, (d) η_4. The solid lines show the densities estimated by probit, the dotted lines by the full model. The vertical lines show the actual parameter values.

Then, fixing η at the posterior mean from the probit estimation, I estimate the rest of the parameters. The results are compared with the full model in Figure 1.6. As can be seen from Figure 1.5, the posterior standard deviations for η are significantly smaller for the full model estimation. Thus, there seem to be significant efficiency gains from the full model estimation. Surprisingly, the estimation results for the rest of the parameters are only slightly affected by fixing η. The "second stage posteriors", depicted in Figure 1.6, noticeably differ from the full model posteriors only for α_1 and $h_\epsilon^{-1/2}$.

Overall, this experiment does not seem to suggest that the two stage classical estimation procedure would produce unreasonable results for this model and sample size. However, the standard errors of the parameters might be considerably affected.

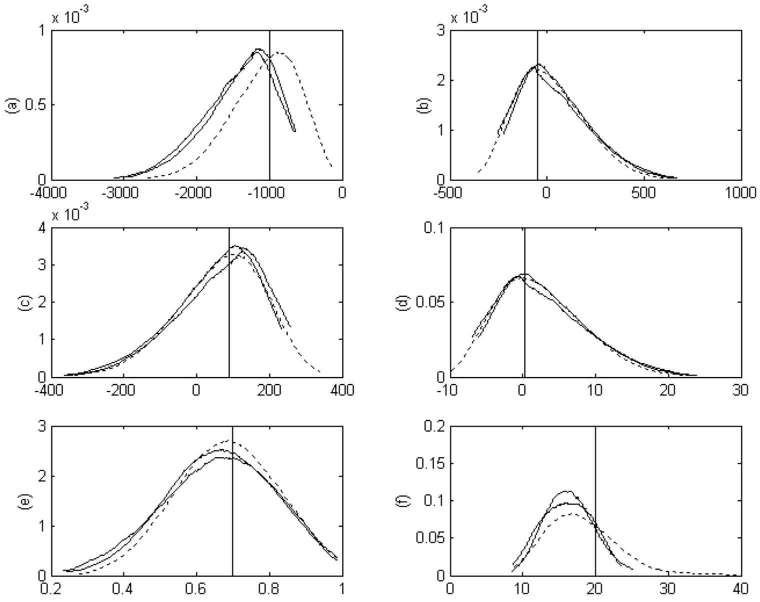

Figure 1.6: Posterior densities of α, ρ, and $h_\epsilon^{-1/2}$: (a) α_1, (b) α_2, (c) α_3, (d) α_4, (e) ρ, (f) $h_\epsilon^{-1/2}$. The solid lines show the densities estimated with fixed η (two simulator runs,) the dotted lines by the full model. The vertical lines show the actual parameter values.

1.5.1.7 Joint distribution tests

To verify that the Gibbs sampler is implemented correctly I employ joint distribution tests developed in Geweke (2004). The test works as follows. First, start from the joint prior distribution for the observables and the unobservables: $\theta^0 \sim p(\theta)$ and $\Delta\mathcal{V}^0, \epsilon^0, x^0, RI^0, d^0 \sim p(data|\theta^0)$. Then, generate a draw from the posterior simulator $\Delta\tilde{\mathcal{V}}^1, \tilde{\epsilon}^1, \tilde{RI}^1, \theta^1$ and generate the observed and the augmented data from the data simulator $\Delta\mathcal{V}^1, \epsilon^1, x^1, RI^1, d^1 \sim p(data|\theta^1)$ and continue repeating these two steps. The invariant stationary distribution of this Markov chain is the joint prior distribution of the observables and the unobservables. This successive conditional simulator uses the posterior and data simulators. If the posterior and data simulators are derived and implemented correctly, then, for example, the sample mean of θ^m converges to the prior mean of θ, which could be tested formally using a central limit theorem.

Using smaller size of the artificial dataset results in better mixing of the chain in the successive conditional simulator. In this experiment $I = 2$. Tighter priors also increase the speed of convergence. Some experimental work was required to choose the variances for the Metropolis-Hastings transition densities. The acceptance rates were in 0.2-0.7 interval.

The test uses 10000 draws from the prior simulator and 75000 draws from the successive conditional simulator. The hypothesis of means equality was not rejected by the standard means equality test performed for the parameters and their squares. The test p-values are reported in Table 1.2. The numerical standard errors were computed by batching with a first order time series correction.[7]

Table 1.2: Joint distribution test

Parameter	Prior simulator mean	Successive conditional simulator mean	Means equality p-value
α_1	-999.99	-1000	0.25
α_2	-49.997	-50.03	0.42
α_3	90.009	89.946	0.10
α_4	0.20014	0.19997	0.67
ρ	0.50209	0.50122	0.60
η_1	-4.5033	-4.4989	0.24
η_2	0.099921	0.099911	0.99
η_3	0.099814	0.10013	0.63
η_4	1.4989	1.5034	0.72
$h_\epsilon^{-.5}$	20.306	20.296	0.73

These tests are useful not only at the program debugging stage, but also help to catch some conceptual errors in the posterior simulator implementation: e.g., in drawing ρ by the MH random walk a truncation constant was not taken into account in computing the acceptance probability (prior for ρ is truncated to $[0, 0.99]$.) The

[7] The sequence of the draws is divided into batches. It is assumed that the means of the batches follow an AR(1) process and the corresponding standard error is computed.

results of the tests are also presented graphically in Figure 1.7.

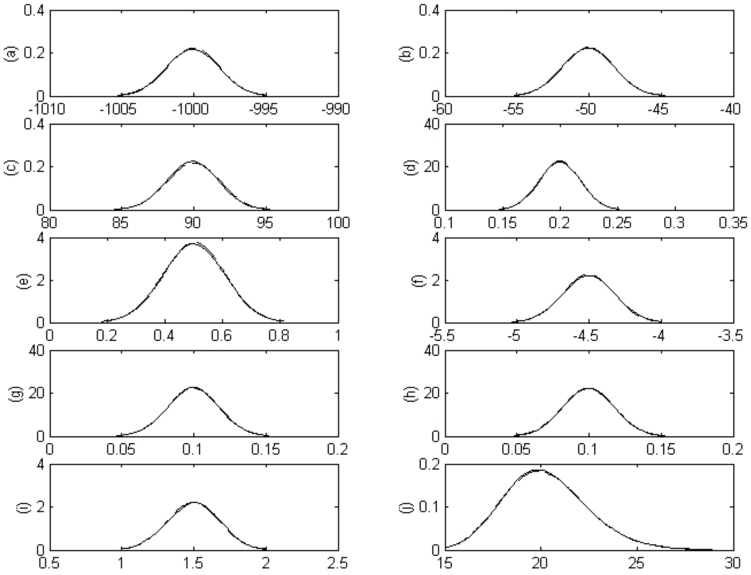

Figure 1.7: Joint distribution tests: (a) α_1, (b) α_2, (c) α_3, (d) α_4, (e) ρ, (f) η_1, (g) η_2, (h) η_3, (k) η_4, (l) $h_\epsilon^{-1/2}$.

The solid lines are the densities of the parameter draws from the successive conditional simulator estimated by kernel smoothing. These densities practically coincide with the dashed lines that show the prior densities.

1.5.2 Rust's (1987) model

1.5.2.1 Setup

Rust (1987) used a binary choice model of optimal bus engine replacement to demonstrate his dynamic logit model. In this model a maintenance superintendent of a bus transportation company decides every time period whether to replace a bus engine. The observed state variable is the bus mileage x_t since the last engine replacement. The control variable d_t takes on two values: 2 if the engine is replaced at t and 1 otherwise. The per-period utility function of the superintendent is the negative of per-period costs:

$$u(x_t, \epsilon_t, \nu_t, d_t; \alpha) = \begin{cases} \alpha_1 x_t + \epsilon_t & \text{if } d_t = 1 \\ \alpha_2 + \nu_t & \text{if } d_t = 2 \end{cases} \tag{1.40}$$

where ϵ_t and ν_t are the unobserved state variables, α_1 is the negative of per-period maintenance costs per unit of mileage, α_2 is the negative of the costs of engine replacement. Rust assumes that ϵ_t and ν_t are extreme value iid. I assume ν_t is iid $N(0, h_\nu^{-1})$ truncated to $[-\overline{\nu}, \overline{\nu}]$, ϵ_t is $N(\rho\epsilon_{t-1}, h_\epsilon^{-1})$ truncated to $E = [-\overline{\epsilon}, \overline{\epsilon}]$, and $\epsilon_0 = 0$. The bus mileage since the last replacement is discretized into M intervals $X = \{1, \ldots, M\}$. The observed state x_t evolves according to

$$P(x_{t+1}|x_t, d_t; \eta) = \begin{cases} \pi(x_{t+1} - x_t; \eta) & \text{if } d_t = 1 \\ \pi(x_{t+1} - 1; \eta) & \text{if } d_t = 2 \end{cases} \tag{1.41}$$

and

$$\pi(\Delta x; \eta) = \begin{cases} \eta_1 & \text{if} \quad \Delta x = 0 \\ \eta_2 & \text{if} \quad \Delta x = 1 \\ \eta_3 & \text{if} \quad \Delta x = 2 \\ 0 & \text{if} \quad \Delta x \geq 3 \end{cases} \tag{1.42}$$

Rust assumes that if the mileage reaches the state M it stays in this state with probability 1. I instead assume that the engine is replaced at t if x_t exceeds $M - 1$, which slightly simplifies the DP solution. In the recursive formulation, the life-time utility for $x_t < M$ is given by

$$\begin{aligned} V(x_t, \epsilon_t, \nu_t; \theta) = \max\{ \quad & \alpha_1 x_t + \epsilon_t + \beta \sum_{k=1}^{3} \eta_k E[V(x_t + k - 1, \epsilon', \nu'; \theta) | \epsilon_t; \theta], \\ & \alpha_2 + \nu_t + \beta E V_2(\theta) \ \} \end{aligned} \tag{1.43}$$

where

$$EV_2(\theta) = \sum_{k=1}^{3} \eta_k E[V(k, \epsilon', \nu'; \theta) | 0; \theta] \tag{1.44}$$

$$E[V(x_{t+1}, \epsilon', \nu'; \theta) | \epsilon_t; \theta] = \int V(x_{t+1}, \epsilon', \nu'; \theta) dP(\epsilon', \nu' | \epsilon_t; \theta) \tag{1.45}$$

For $x_t \geq M$:

$$V(x_t, \epsilon_t, \nu_t; \theta) = \alpha_2 + \nu_t + \beta E V_2(\theta) \tag{1.46}$$

1.5.2.2 Approximating value functions

The algorithm of approximating the value functions is similar to the one for Gilleskie's model, where the role of V_w is played by EV_2. Only one nearest neighbor is used, $\tilde{N}(m) = 1$. The random grid $\{y^{m,j} = (\nu^{m,j}, \epsilon^{m,j})\}_{j=1}^{\hat{N}(m)}$ is generated from a normal distribution: $\nu_d^{m,j} \sim N(0, h_\nu^{-1})$ and $\epsilon^{m,j} \sim g(.)$, where $g(.)$ is a normal density. First, one iteration of the DP solving algorithm is described as it should be performed according to the theory. Then, improvements in the algorithm performance are discussed.

An iteration of the DP solving algorithm is performed as follows. For a given initial $EV_2^{k_1}(\theta^{k_1})$ corresponding to the nearest neighbor, Bellman equations (1.43) are iterated for x in descending order and $j = 1, \ldots, \hat{N}(m)$:

$$\begin{aligned} V^m(x, y^{m,j}; \theta^m) = \max\{ \quad & \alpha_1^m x + \epsilon^{m,j} + \beta \sum_{k=1}^{3} \eta_k \hat{E}^m[V(x+k-1, y'; \theta^m) | y^{m,j}; \theta^m], \\ & \alpha_2^m + \nu^{m,j} + \beta E V_2^{k_1}(\theta^{k_1})\} \end{aligned} \tag{1.47}$$

where

$$\hat{E}^m[V(x+k-1, y'; \theta^m) | y^{m,j}; \theta^m] = \sum_{r=1}^{\hat{N}(k_1)} V^{k_1}(x+k-1, y^{k_1,r}; \theta^{k_1}) W(\epsilon^{k_1,r}, \epsilon^{k_1,j}, \theta^{k_1}) \tag{1.48}$$

Note that for $k > 1$ the value functions $V^m(x+k-1, y^{m,r}; \theta^m)$ have already been computed. Thus, for $k > 1$, k_1 will be equal to m in (1.48). Next, $EV_2^m(\theta^m)$ is computed:

$$EV_2^m(\theta^m) = \sum_{k=1}^{3} \eta_k \sum_{r=1}^{\hat{N}(m)} V^m(k, y^{m,r}; \theta^m) W(\epsilon^{m,r}, 0, \theta^m) \tag{1.49}$$

In practice, iterating (1.47) several times in a row for one x before going to the next significantly improves the convergence speed. It happens because the expression for the value function at the mileage x includes the expected value function at the same x. If Bellman equation (1.47) is iterated several times the approximation error in $V^m(x, y^{m,j}; \theta^m)$ becomes smaller and affects the value functions for $\{1, \ldots, x-1\}$ much less. Using only already updated $V^m(x, y^{m,j}; \theta^m)$ in computing the expectations further improves the performance. Thus, when (1.47) is iterated first time for a given x the expectations for $k = 1$ are approximated as follows:

$$\hat{E}^m[V(x+k-1, y'; \theta^m) | y^{m,j}; \theta^m] = \sum_{r=1}^{j-1} V(x, y^{m,r}; \theta^m) W(y^{m,r}, y^{m,j}, \theta^m) \tag{1.50}$$

For $j = 1$, $\hat{E}^m[V(x+0, y'; \theta^m)|y^{m,1}; \theta^m]$ is a solution of the following equation:

$$\hat{E}^m[V(x+0, y'; \theta^m)|y^{m,1}; \theta^m] = \max\{\alpha_2^m + \beta EV_2^{k_1}(\theta^{k_1}),$$

$$\alpha_1^m x + \beta(\eta_1^m \hat{E}^m[V(x+0, y'; \theta^m)|y^{m,1}; \theta^m] \tag{1.51}$$

$$+\eta_2^m \hat{E}^m[V(x+1, y'; \theta^m)|y^{m,1}; \theta^m] + \eta_3^m \hat{E}^m[V(x+2, y'; \theta^m)|y^{m,1}; \theta^m])\} \tag{1.52}$$

This equation is obtained by interchanging the places of the expectation and the max in the Bellman equation. After the first iteration on (1.47) for a given x, all the expectations are computed according to (1.48) on the subsequent iterations. This procedure can be seen as a mapping taking EV_2 as an input and updating it. The fixed point of this mapping can be found by a direct search procedure similar to the one described in Section 1.5.1.3.

1.5.2.3 Gibbs sampler

Each bus i is observed over T_i time periods: $\{x_{t,i}, d_{t,i}\}_{t=1}^{T_i}$ for $i = 1, \ldots, I$. The parameters are $\theta = (\alpha, \eta, \rho, h_\epsilon)$; h_ν is fixed for normalization. The latent variables are $\{\Delta\mathcal{V}_{t,i}, \epsilon_{t,i}\}_{t=1}^{T_i}$ $i = 1, \ldots, I$.

$$\Delta\mathcal{V}_{t,i} = x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} - \nu_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})$$

where

$$F_{t,i}(\theta, \epsilon) = \beta \sum_{j=1}^{3} \eta_j (E[V(x_{t,i} + j - 1, \epsilon', \nu'; \theta)|\epsilon; \theta] - EV_2(\theta))$$

The compact space for parameters Θ is defined as follows: $\alpha_i \in [-\overline{\alpha}, \overline{\alpha}]$, $\rho \in [-\overline{\rho}, \overline{\rho}]$, $h_\epsilon \in [\overline{h_\epsilon^l}, \overline{h_\epsilon^r}]$, and η belongs to a three dimensional simplex. The joint distribution of the data, the parameters, and the latent variables is

$$p(\theta; \{x_{t,i}, d_{t,i}; \Delta\mathcal{V}_{t,i}, \epsilon_{t,i}\}_{t=1}^{T_i}; i = 1, \ldots, I) =$$
$$p(\theta) \prod_{i=1}^{I} \prod_{t=1}^{T_i} [p(d_{t,i}|\Delta\mathcal{V}_{t,i}) p(\Delta\mathcal{V}_{t,i}|x_{t,i}, \epsilon_{t,i}; \theta) p(x_{t,i}|x_{t-1,i}; d_{t-1,i}; \eta) p(\epsilon_{t,i}|\epsilon_{t-1,i}, \rho, h_\epsilon)]$$

where $p(\theta)$ is a prior density for the parameters; $p(x_{t,i}|x_{t-1,i}; d_{t-1,i}; \eta)$ is given in (1.41) and $p(x_{1,i}|x_{0,i}; d_{0,i}; \eta) = 1_{\{1\}}(x_{1,i})$—all the buses start with a new engine;

$$p(d_{t,i}|\Delta\mathcal{V}_{t,i}) = \begin{cases} 1, \text{ if } d_{t,i} = 1, \Delta\mathcal{V}_{t,i} \geq 0 \text{ or } d_{t,i} = 2, \Delta\mathcal{V}_{t,i} \leq 0 \\ 0, \text{ if } d_{t,i} = 1, \Delta\mathcal{V}_{t,i} < 0 \text{ or } d_{t,i} = 2, \Delta\mathcal{V}_{t,i} > 0 \end{cases} \tag{1.53}$$

$$p(\Delta\mathcal{V}_{t,i}|x_{t,i}, \epsilon_{t,i}; \theta) = \exp\{-0.5h_\nu(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})])^2\} \tag{1.54}$$

$$\cdot 1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})]) \tag{1.55}$$

$$\cdot \frac{h_\nu^{0.5}}{\sqrt{2\pi}[\Phi(\overline{\nu}h_\nu^{0.5}) - \Phi(-\overline{\nu}h_\nu^{0.5})]}$$

$$p(\epsilon_{t,i}|\epsilon_{t-1,i}, \theta) = \frac{h_\epsilon^{1/2} \exp\{-0.5h_\epsilon(\epsilon_{t,i} - \rho\epsilon_{t-1,i})^2\}}{\sqrt{2\pi}[\Phi([\overline{\epsilon} - \rho\epsilon_{t-1,i}]h_\epsilon^{0.5}) - \Phi([-\overline{\epsilon} - \rho\epsilon_{t-1,i}]h_\epsilon^{0.5})]} 1_E(\epsilon_{t,i}) \tag{1.56}$$

Gibbs sampler blocks

The Gibbs sampler blocks for $\Delta\mathcal{V}_{t,i}|\ldots$ will have a normal truncated distribution proportional to (1.54) and (1.55), and also truncated to R^+ if $d_{t,i} = 1$ or to R^- otherwise. An algorithm from Geweke (1991) is used to simulate efficiently from the normal distribution truncated to R^+ (or R^-.) Acceptance sampling handles the truncation in (1.55).

The density for $\epsilon_{t,i}|\ldots$ is proportional to

$$
\begin{aligned}
p(\epsilon_{t,i}|\ldots) \quad \propto \quad & \frac{\exp\{-0.5h_{\nu}(\Delta\mathcal{V}_{t,i}-[x_{t,i}\alpha_1-\alpha_2+\epsilon_{t,i}+F_{t,i}(\theta,\epsilon_{t,i})])^2\}}{\Phi([\overline{\epsilon}-\rho\epsilon_{t-1,i}]h_{\epsilon}^{0.5})-\Phi([-\overline{\epsilon}-\rho\epsilon_{t-1,i}]h_{\epsilon}^{0.5})} \\
\cdot \quad & 1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i}-[x_{t,i}\alpha_1-\alpha_2+\epsilon_{t,i}+F_{t,i}(\theta,\epsilon_{t,i})]) \\
\cdot \quad & \exp\{-0.5h_{\epsilon}(\epsilon_{t+1,i}-\rho\epsilon_{t,i})^2-0.5h_{\epsilon}(\epsilon_{t,i}-\rho\epsilon_{t-1,i})^2\}\cdot 1_{E}(\epsilon_{t,i}) \qquad (1.57)
\end{aligned}
$$

Draws from this density are obtained from a Metropolis step with a normal truncated transition density proportional to (1.57). The blocks for $\epsilon_{t,i}$ with $t=0$ and $t=T_i$ will be similar.

Assuming a normal prior $N(\underline{\rho},\underline{h}_{\rho})$ truncated to $[-\overline{\rho},\overline{\rho}]$,

$$
\begin{aligned}
p(\rho|\ldots) \quad \propto \quad & \frac{\exp\{-0.5h_{\nu}\sum_{i,t}(\Delta\mathcal{V}_{t,i}-[x_{t,i}\alpha_1-\alpha_2+\epsilon_{t,i}+F_{t,i}(\theta,\epsilon_{t,i})])^2\}}{\prod_{i,t}\Phi([\overline{\epsilon}-\rho\epsilon_{t-1,i}]h_{\epsilon}^{0.5})-\Phi([-\overline{\epsilon}-\rho\epsilon_{t-1,i}]h_{\epsilon}^{0.5})} \\
\cdot \quad & \prod_{i,t}1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i}-[x_{t,i}\alpha_1-\alpha_2+\epsilon_{t,i}+F_{t,i}(\theta,\epsilon_{t,i})]) \\
\cdot \quad & \exp\{-0.5\overline{h}_{\rho}(\rho-\overline{\rho})^2\}\cdot 1_{[-\overline{\rho},\overline{\rho}]}(\rho) \qquad (1.58)
\end{aligned}
$$

where $\overline{h}_{\rho}=\underline{h}_{\rho}+\sum_i\sum_{t=2}^{T_i}\epsilon_{t-1,i}^2$ and $\overline{\rho}=\overline{h}_{\rho}^{-1}(\underline{h}_{\rho}\underline{\rho}+h_{\epsilon}\sum_i\sum_{t=2}^{T_i}\epsilon_{t,i}\epsilon_{t-1,i})$. To draw from this density I use a Metropolis step with a normal truncated transition density proportional to (1.58).

Assuming a gamma prior $\underline{s}^2h_{\epsilon}\sim\chi^2(\underline{df})$, truncated to $[\overline{h_{\epsilon}^{l}},\overline{h_{\epsilon}^{r}}]$,

$$
\begin{aligned}
p(h_{\epsilon}|\ldots) \quad \propto \quad & \frac{\exp\{-0.5h_{\nu}\sum_{i,t}(\Delta\mathcal{V}_{t,i}-[x_{t,i}\alpha_1-\alpha_2+\epsilon_{t,i}+F_{t,i}(\theta,\epsilon_{t,i})])^2\}}{\prod_{i,t}\Phi([\overline{\epsilon}-\rho\epsilon_{t-1,i}]h_{\epsilon}^{0.5})-\Phi([-\overline{\epsilon}-\rho\epsilon_{t-1,i}]h_{\epsilon}^{0.5})} \\
\cdot \quad & \prod_{i,t}1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i}-[x_{t,i}\alpha_1-\alpha_2+\epsilon_{t,i}+F_{t,i}(\theta,\epsilon_{t,i})]) \\
\cdot \quad & h_{\epsilon}^{(\overline{df}-2)/2}\exp\{-0.5\overline{s}^2h_{\epsilon}\}\cdot 1_{[\overline{h_{\epsilon}^{l}},\overline{h_{\epsilon}^{r}}]}(h_{\epsilon}) \qquad (1.59)
\end{aligned}
$$

where $\overline{df}=\underline{df}+\sum_i T_i$ and $\overline{s}^2=\underline{s}^2+\sum_i\left(\sum_{t=2}^{T_i}(\epsilon_{t,i}-\rho\epsilon_{t-1,i})^2+\epsilon_{1,i}^2\right)$. For this block, I employ a Metropolis step with a truncated gamma transition density proportional to (1.59); draws from this density are obtained by acceptance sampling.

Assuming a Dirichlet prior with parameters (a_1, a_2, a_3),

$$\begin{aligned} p(\eta|\ldots) &\propto \exp\{-0.5h_\nu \sum_{i,t}(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta,\epsilon_{t,i})])^2\} \\ &\cdot \prod_{i,t} 1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta,\epsilon_{t,i})]) \\ &\cdot \prod_{j=1}^{3} \eta_j^{n_j+a_j-1} \end{aligned} \tag{1.60}$$

where $n_j = \sum_i \sum_{t=2}^{T_i} 1_{\{j-1\}}(x_{t,i} - x_{t-1,i})$. A Metropolis step with a Dirichlet transition density proportional to (1.60) is used in this block.

$$\begin{aligned} p(\alpha|\ldots) &\propto p(\alpha)\exp\{-0.5h_\nu \sum_{i,t}(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta,\epsilon_{t,i})])^2\} \\ &\cdot \quad 1_{[-\overline{\alpha},\overline{\alpha}]\times[-\overline{\alpha},\overline{\alpha}]}(\alpha) \cdot \prod_{i,t} 1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta,\epsilon_{t,i})]) \end{aligned}$$

To draw from this density I use the Metropolis-Hastings random walk algorithm. The proposal density is normal truncated to $[-\overline{\alpha},\overline{\alpha}] \times [-\overline{\alpha},\overline{\alpha}]$ with a mean equal to the current parameter draw and a fixed variance. The variance matrix is chosen so that the acceptance probability would be between $0.2 - 0.3$.

1.5.2.4 Uniform ergodicity

The draws from the Gibbs sampler are used for approximating posterior expectations by sample averages. Under certain conditions, the sample averages converge almost surely to the posterior expectations and a corresponding central limit theorem holds. Uniform ergodicity of the Gibbs sampler—a sufficient condition for these results (see Tierney (1994))—is established by the following theorem.

Theorem 1.4. *Consider the Gibbs sampler with the following order of blocks at each iteration: 1)* $(\Delta\tilde{\mathcal{V}}_{t,i}^{m+1}|\theta^m, \epsilon^m, d, x)$*,* $\forall t, i$*; 2)* $(\rho^{m+1}|\theta^m, \epsilon^m, \Delta\tilde{\mathcal{V}}^{m+1}, d, x)$*,* $(\alpha^{m+1}|\ldots)$*,* $(\eta^{m+1}|\ldots)$*,* $(h_\epsilon^{m+1}|\ldots)$*; 3)* $(\epsilon_{t,i}^{m+1}|\epsilon^m, \theta^{m+1}, \Delta\tilde{\mathcal{V}}^{m+1})$*,* $\forall t, i$*; 4)* $(\Delta\mathcal{V}_{t,i}^{m+1}|\theta^{m+1}, \epsilon^{m+1}, d, x)$*,* $\forall t, i$*; where the blocks were described above. Block 4) is redundant but simplifies the proof. Assume that the support of* $\nu_{t,i}$ *is sufficiently large relative to the support of* ϵ *and* θ*:* $\Phi(-h_\nu^{0.5}\overline{\nu}) < 0.25$ *and* $\overline{\nu} > 2(\overline{u} + \overline{\epsilon} + \beta\overline{EV})$*, where* $\overline{u}$ *is an upper bound on the absolute value of the deterministic part of the per-period utility function,* $\overline{\epsilon}$ *is an upper bound on the absolute value of* $\epsilon_{t,i}$*, and* $\overline{EV} = [\overline{u} + \overline{\epsilon} + 1 + 2h_\epsilon^{-1}]/(1-\beta)$ *is an upper bound on the absolute value of the expected value function (see the proof.) Then, the Gibbs sampler is uniformly ergodic. Thus, by Theorems 3 and 5 in Tierney (1994), for any integrable (w.r.t. posterior) function* $z(\Delta\mathcal{V}, \theta, \epsilon)$ *the sample average* $\overline{z}_n = 1/n\sum_m z(\Delta\mathcal{V}^m, \theta^m, \epsilon^m)$ *converges a.s. to the posterior expectation* $E(z|d, x)$*. If* $E(z^2|d, x) < \infty$ *then there exists a real number* $\sigma^2(z)$ *such that* $\sqrt{n}(\overline{z}_n - E(z|d, x))$ *converges in distribution to* $N(0, \sigma^2(z))$*.*

The theorem is proven in Chapter 2, Section 2.3. For the Gibbs sampler that uses the approximations instead of the actual value functions the uniform ergodicity holds in probabilistic sense. Suppose that the DP solving algorithm stops at iteration n and that the Gibbs sampler uses the output from the DP solving algorithm up to iteration n for approximating the value functions on all the subsequent iterations. Then, this Gibbs sampler is uniformly ergodic with a probability that converges to 1 very fast as n goes to infinity. That is it becomes uniformly ergodic a.s. This statement follows from the proof of Theorem 1.4 and the fact that the approximated expected value functions will be bounded by $\overline{EV}$ plus a small positive number with a probability that converges to 1.

Most of the distributions in the described Gibbs sampler are truncated distributions. In practice, it is easier to use distributions with unbounded support and ignore the truncation. To reconcile this with the theory, we could assume that the truncation bounds are very large. Then, the Gibbs sampler that takes them into account would produce the same results as the Gibbs sampler that does not: e.g. simulating $N(0,1)$ (or $N(10^6, 10^6)$) truncated to $(-1.7 \cdot 10^{308}, 1.7 \cdot 10^{308})$ will give the same results as simulating $N(0,1)$ (or $N(10^6, 10^6)$.) Therefore, in the experiments below, the truncation to the bounded parameter and state spaces is not enforced.

1.5.2.5 Algorithm parallelization

Opportunities for algorithm parallelization are abundant and it is a possible subject for future work. Approximating the expectations of the value functions either in the DP solving algorithm or in the Gibbs sampler is a very frequent and time consuming task. The computations of the expected value functions for different current states are not interrelated. Thus, they could be computed simultaneously on different computers/processors in the cluster. In designing a parallel implementation of an algorithm, it is crucial to take into account the time required for communication and data transfer between computers. As was indicated in Section 1.5.1.4 solving the DP on several smaller random grids and combining the results is a very efficient alternative to using one big random grid. If, in addition, different processors are used for solving the DP on each random grid, the performance would be increased considerably since the need for data transfer between processors will be decreased relative to just computing expectations simultaneously. In the Gibbs sampler, the volume of data exchange between machines in the cluster could also be minimized. Each processor in the cluster could be assigned to a part of the dataset. Then, simulating latent variables pertaining to these parts of the dataset could be performed simultaneously by different processors and no data exchange is needed given that the current parameters and the results of the DP solving algorithm are copied to the memory of each computer in the cluster. In drawing parameters there is no need for transferring all the latent variables and the corresponding expectations between computers. A computer assigned to a part of the dataset could compute aggregates corresponding to its part of the dataset and transfer only these aggregates to the computer that performs drawing the parameters: e.g. in drawing $\rho| \ldots$ parts of the sum in (1.58) could be computed by the computers to which the corresponding part of the sample is assigned. The proposed parallelization scheme is relatively easy to implement, and the potential gains in performance seem to be considerable.

1.5.2.6 Estimation for artificial datasets

Estimation was performed for two different artificial datasets. One artificial sample consisted of $I = 500$ observations, the other one of $I = 50$. Each bus i is observed $T_i \in \{1, \ldots, 200\}$ months until the engine is replaced (if the engine is not replaced at $t = 200$ the observation i is censored, there were no such observations in simulated data.) As in Rust's paper, the mileage is divided into 90 discrete intervals corresponding to $[0, 5000], \ldots, [445000, 450000]$ miles. The parameter values used for data generation were taken from Rust's paper. The precision for the correlated unobserved state variables h_ϵ was fixed: $h_\epsilon = h_\nu = 0.6$, which roughly corresponds to the precision of the extreme value distributed errors in Rust's paper. The time discount factor was equal to $\beta = 0.999$. Joint distribution tests did not reject the hypothesis of correct implementation of the prior, the data, and the posterior simulators.

Table 1.3 shows the estimation results for six posterior simulator runs. The first three runs (1-3) were produced for the sample of size $I = 500$, the other three runs (4-6) for the sample of size $I = 50$. The length of each run was equal to 1000000. Figure 1.8 illustrates the estimation results graphically. As can be seen from the figure and the table, the posterior distributions are tighter for the larger sample size. This expected behavior is more pronounced for η than for the rest of the parameters. There are some other respects in which the results are different for η. First, the uncertainty about η seems to be much smaller than for the rest of the parameters. Second, the acceptance rate for η was about 90%, which is quite high compared to 10-20% for ρ (for alpha it was about 27%, but it should not be compared with the other two since it is affected by the preset variance of the MH transition density.) Third, the amount of serial correlation in draws of η was insignificant, which resulted in very fast convergence for η. For the rest of the parameters the observations made for Gilleskie's model, such as large uncertainty about parameters and slow convergence are unchanged. The most important reason for this difference seems to be a

Table 1.3: Estimation results for artificial data

	Run	α_1	α_2	ρ	η_1	η_2	η_3
Post mean	1	-.0026	-9.65	.683	0.33925	0.63972	0.021033
	2	-.0027	-10.36	.703	0.33926	0.63971	0.021032
	3	-.0029	-10.35	.699	0.33927	0.6397	0.021032
	4	-.0035	-9.82	.649	0.34686	0.63296	0.020176
	5	-.0036	-11.1	.684	0.34692	0.6329	0.020182
	6	-.003	-8.69	.599	0.34681	0.63302	0.020176
Post SD	1	.00099	1.56	.049	0.003814	0.003869	0.001156
	2	.00095	1.66	.043	0.003813	0.003868	0.001155
	3	.00101	1.68	.051	0.003816	0.003869	0.001156
	4	.00146	2.17	.103	0.011592	0.011751	0.003421
	5	.0015	2.89	.104	0.011585	0.011734	0.003424
	6	.00129	2.07	.126	0.011589	0.011731	0.003423
NSE for post mean	1	.00026	.774	.022	5.00E-06	4.92E-06	1.19E-06
	2	.00024	.903	.021	4.85E-06	4.81E-06	1.31E-06
	3	.00026	.792	.024	4.54E-06	4.48E-06	1.27E-06
	4	.00022	.466	.019	1.85E-05	1.87E-05	3.78E-06
	5	.00025	.817	.024	1.79E-05	1.78E-05	4.36E-06
	6	.00017	.392	.020	1.75E-05	1.79E-05	3.80E-06
True		-.003	-10	.7	.34	.64	.02
Prior		N(-.0035, $.0017^2$)	$N(-12, 5^2)$	$N(.5, 10^6)$ s.t.[0,.99]	Dirichlet $a_1 = 34$	prior $a_2 = 64$	for η: $a_3 = 2$

great deal of information that the data on bus mileage $x_{t,i}$ contain about η. Information about the rest of the parameters is mainly contained in the observed decisions and obscured by the presence of a lot of latent variables. Thus, long posterior simulator runs are also required for estimation of Rust's model due to considerable amount of serial correlation in the posterior simulator draws for the weakly identified part of the parameter vector.

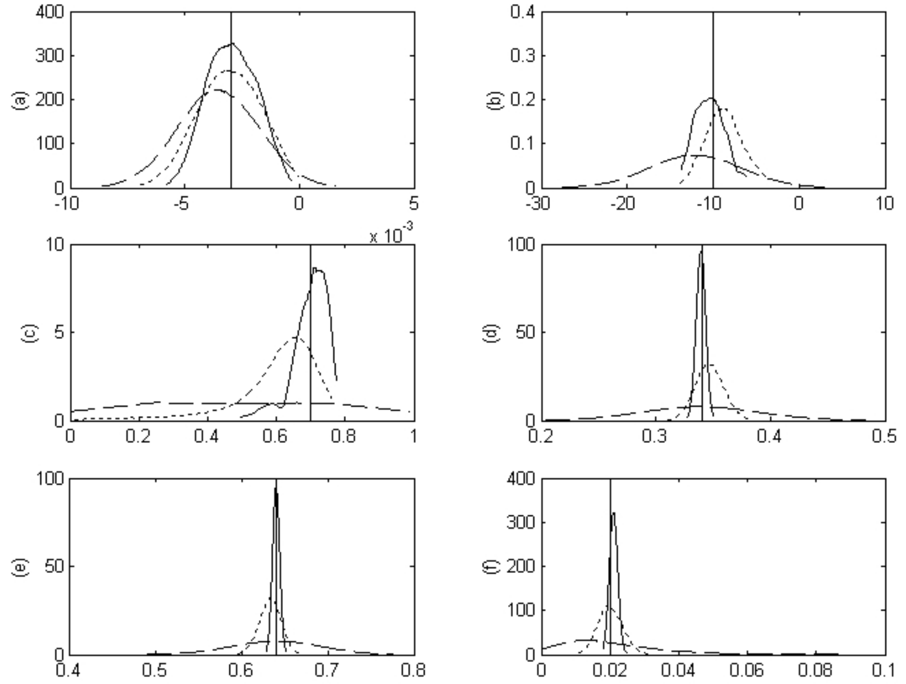

Figure 1.8: Estimated posterior densities: solid lines for $I = 500$, dotted lines for $I = 50$. (a) α_1, (b) α_2, (c) ρ, (d) η_1, (e) η_2, (f) η_3. The dashed lines are prior densities. The vertical lines show the actual parameter values.

An attempt to decrease the amount of serial correlation was made. Instead of drawing the correlated unobservables $(\epsilon_{t,i}|\epsilon_{t-1,i}, \epsilon_{t+1,i}, \ldots)$ one at a time, I tried to draw them in blocks $(\epsilon_{t_1,i}, \epsilon_{t_1+1,i}, \ldots, \epsilon_{t_2,i}|\epsilon_{t_1-1,i}, \epsilon_{t_2+1,i}, \ldots)$. For the blocks of size

$t_2 - t_1 + 1 = 2$ the acceptance rate for the correlated unobservables dropped from 0.6 to 0.4 and the amount of serial correlation in the sampler did not seem to change much. For the blocks of size $t_2 - t_1 + 1 = 5$ the acceptance rate for the correlated unobservables and ρ decreased to less than 0.001. Thus, grouping parameters in larger Gibbs sampler blocks does not seem to work as a technique of decreasing the amount of serial correlation in posterior simulator draws for this problem. Looking for other techniques is a subject of future work.

1.5.2.7 Exact and approximate estimation

To evaluate the quality of the estimation results I conduct experiments on the model with extreme value unobservables—the dynamic logit model. For this model, the integration over the unobservables in solving the DP and in the likelihood function can be performed analytically. The estimation method that integrates the unobservables analytically in the likelihood and in the DP solution will be referred below as the exact algorithm. The posterior simulator for this method also uses the Metropolis-Hastings algorithm since the logit-like choice probabilities comprising the likelihood function contain the expected value functions that can only be computed numerically. The approximate algorithm will refer to the algorithm proposed in this chapter. The Gibbs sampler for the approximate algorithm is the same as the one for the Gaussian unobservables described in Section 1.5.2.3; except here the Gaussian probability densities are replaced by the densities for the extreme value distribution. Table 1.4 gives the estimation results for the exact and approximate algorithms.

Table 1.4: Exact and approximate estimation results.

	Run	α_1	α_2	η_1	η_2	η_3
Post mean	1	-.00228	-9.0721	0.34433	0.63394	0.021736
	2	-.00247	-9.4999	0.34435	0.63392	0.021731
	3	-.00203	-9.1815	0.3443	0.63397	0.021733
	4	-.00207	-9.2569	0.34433	0.63394	0.021732
	5	-.00229	-8.7955	0.34435	0.63393	0.021727
	6	-.00241	-9.0610	0.34435	0.63392	0.021733
	7	-.00229	-9.0519	0.34434	0.63392	0.02174
	8	-.00231	-9.0797	0.34432	0.63395	0.021733
Post SD	1	.00044	0.8538	0.006311	0.006399	0.001939
	2	.00049	0.9795	0.006315	0.006403	0.001938
	3	.00046	0.9681	0.006314	0.0064	0.00194
	4	.00047	0.9655	0.00631	0.006394	0.001932
	5	.00042	0.7790	0.006302	0.006396	0.001938
	6	.00051	0.9789	0.006298	0.006395	0.001938
	7	.00051	1.0028	0.006327	0.006412	0.001941
	8	.00049	0.9680	0.006306	0.006396	0.001941

Table 1.4 – Continued

NSE for post mean	1	.00015	0.3444	7.42E-06	7.31E-06	2.23E-06
	2	.00019	0.4404	8.48E-06	8.12E-06	2.11E-06
	3	.00007	0.1892	2.27E-05	2.10E-05	4.16E-06
	4	.00007	0.2015	2.63E-05	2.36E-05	4.61E-06
	5	.00005	0.1196	1.90E-05	1.74E-05	3.95E-06
	6	.00007	0.1957	1.82E-05	1.71E-05	3.88E-06
	7	.00007	0.1926	2.11E-05	1.97E-05	3.93E-06
	8	.00006	0.1776	2.04E-05	1.88E-05	3.81E-06
Prior		N(-.003,.0017)	N(-10, 5)	Dirichlet	prior	
				$a_1 = 34$	$a_2 = 64$	$a_3 = 2$
Actual	param	-.003	-10	.34	.64	.02

The experiments use an artificial dataset consisting of observations on $I = 70$ buses. Runs 1–2 in the table are the runs of the exact posterior simulator started from different random initial values for the parameters. The length of runs 1–2 was 1000000. For the approximate algorithm three different realization of the random grid for solving the DP were used. Each grid realization corresponds to a pair of simulator runs: 3–4, 5–6, and 7–8. The random number generator was initialized differently for each run. The length of runs 3–8 was about 500000.

Figure 1.9 shows the marginal posterior densities for the parameters obtained from the exact and approximate algorithms.

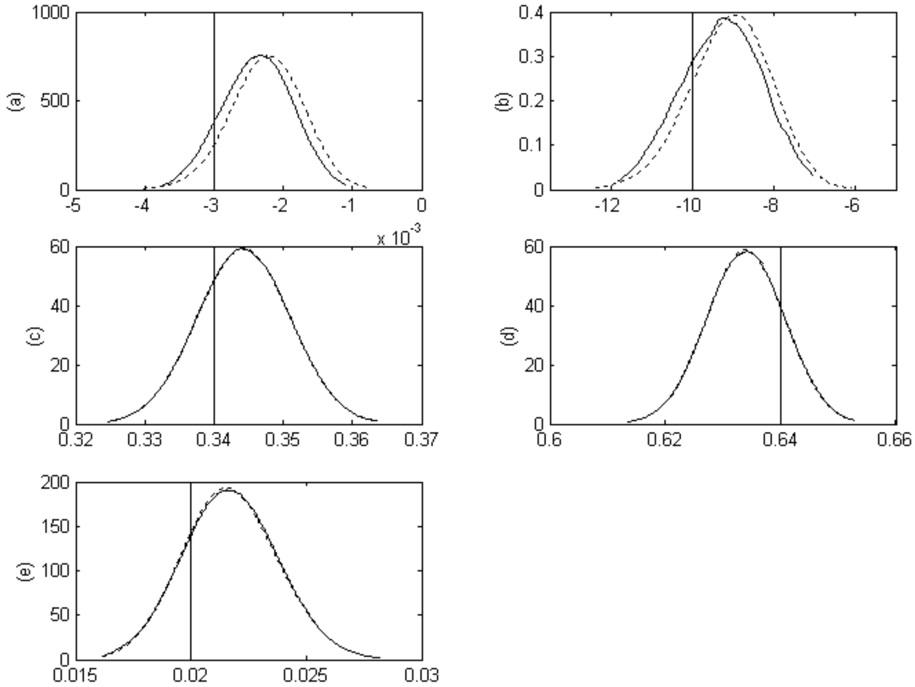

Figure 1.9: Comparison with exact and approximate estimation algorithms. Estimated posterior densities: (a) α_1, (b) α_2, (c) η_1, (d) η_2, (e) η_3. The vertical lines show the actual parameter values.The solid line shows the posterior for exact estimation procedure, the dashed line – approximate estimation procedure

The densities were obtained by kernel smoothing over all available simulator runs: 2 runs for the exact algorithm and 6 runs for the approximate algorithm. The results in the figure and table above suggest that the approximation quality of the proposed algorithm is very good.

1.5.2.8 The role of serial correlation

In this section, I show how the presence of serial correlation in unobservables in the data generation process affects the estimation results for the dynamic logit model. For this purpose I use an artificial dataset simulated from the model with Gaussian serially correlated unobservables described in this chapter (it will be referred in this section as the true model.) Then, I use this data in estimation of the dynamic logit model and the true model. The results are shown in the table below.

Table 1.5: Estimation results for the dynamic logit model and the model with Gaussian serially correlated unobservables.

	Run	α_1	α_2	ρ	η_1	η_2	η_3
Post	logit	-.0009	-3.1431		.35883	.6273	.0138
mean	true	-.0028	-10.7342	.843	.35887	.6273	.0138
Post	logit	.00065	0.2275		.012606	.012703	.003
SD	true	.00098	1.3262	.061	.012607	.012719	.003
NSE post	logit	.00001	.005		1.4E-05	1.4E-05	3.2E-06
mean	true	.00006	0.1848	.019	2.0E-05	2.0E-05	3.8E-06
Actual		-.003	-10.0	.85	.34	.64	.02
Prior		N(-.003,	N(-10,5^2)	N(0.5,10^6)	Dirichlet	prior	
		.0017^2)		s.t.[-.99,.99]	$a_1 = 34$	$a_2 = 64$	$a_3 = 2$

From this table, it might seem that the presence of serial correlation in unobservables produces the same effect as an increase in the variance of these unobservables would. The utility function parameters are almost proportional for both cases. To get more insight into the effects of serial correlation in unobservables, I compute the posterior means of the hazard function for each of the models.

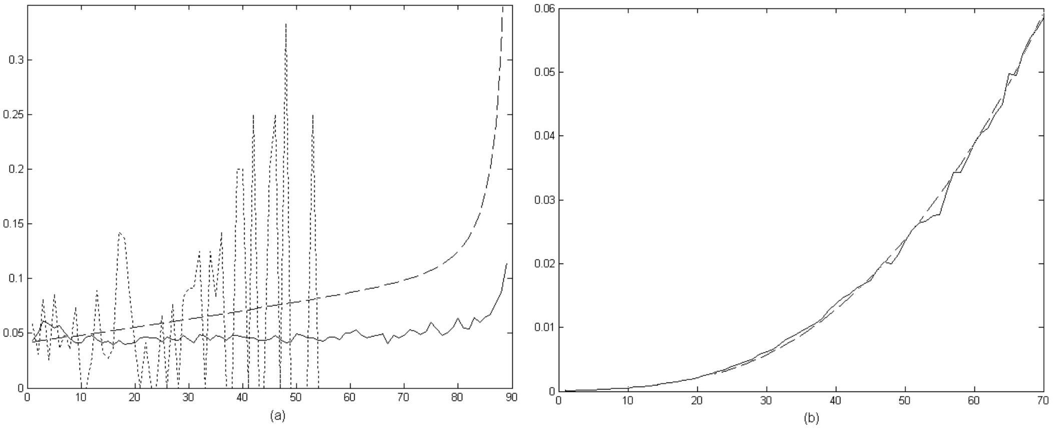

Figure 1.10: The posterior means of the hazard functions. Panel (a)—the data generated from the model with the serially correlated unobservables, panel (b)—the data generated from the dynamic logit model. The vertical axis is for the probability of engine replacement, the horizontal axis is for the mileage interval. The solid line is for the model with serially correlated unobservables. The dashed line—for dynamic multinomial logit, the dotted line—data hazard.

As can be seen from panel (a) in Figure 1.10, the dynamic logit model would underestimate the probability of the engine replacement for low mileage and considerably overestimate the probability for high mileage if serial correlation is present in the data generation process. Moreover, the shape of the hazard function is also different. In the dynamic logit case, the hazard function is increasing, while for the true model it is decreasing at first. Although the estimated hazard is noisy, the decrease at the beginning was observed for several posterior simulator runs; thus it is not a result of the noise. For comparison, panel (b) shows the posterior means of the hazard functions estimated from the artificial data that were simulated from the dynamic logit model. In this case, the hazards for the dynamic logit model and for the model with Gaussian serially correlated unobservables seem to be very close and have the same shape. These results support the claim that the disparities in the hazards observed in panel (a) are driven by the presence of serial correlation in the data but not by the different distributional assumptions on unobservables: Gaussian vs. extreme value. These experiment demonstrate that ignoring serial correlation in unobservables might lead to serious misspecification errors.

1.5.2.9 Estimation for a real dataset

The data set is group 4 from Rust's paper. It contains observations on 37 buses that could be divided into $I = 70$ individual spells containing one engine replacement (or censored at the last observed x_t), which gives $\sum_i T_i = 4329$ monthly mileage/decision points. It takes about 50 seconds to produce 100 draws from the posterior on a 2002 vintage PC.

To start the Gibbs sampler I used the parameter estimates from Rust's paper. The algorithm also works if the Gibbs sampler is started from a draw from the prior distribution or from the zero vector for the utility function parameters α and the data frequencies for the state transition probabilities η. The initial values for the latent variables are adjusted so that the observed decisions are optimal. In particular, given the parameter values, the serially correlated unobservables $\epsilon_{t,i}$ are simulated from

Table 1.6: Estimation results for data from Rust's paper, group 4.

	Run	α_1	α_2	ρ	η_1	η_2	η_3
Post mean	1	-.00275	-9.56	-.1763	.40325	.58388	.012874
	2	-.00289	-10.04	.0945	.40325	.58388	.012869
	3	-.00248	-11.28	.1545	.40329	.58385	.012868
	4	-.00228	-9.89	-.1000	.40328	.58385	.012868
	5	-.00224	-10.13	-.0526	.40318	.58395	.012875
	6	-.00233	-10.37	-.0573	.4032	.58392	.012872
Post SD	1	.00123	2.4267	.4229	.00736	.00739	.001696
	2	.00146	2.8896	.2879	.00738	.00741	.001694
	3	.00084	3.2330	.4170	.0074	.00743	.001693
	4	.00075	2.5904	.4871	.00739	.00742	.001694
	5	.00066	2.9959	.4805	.00737	.00740	.001694
	6	.00075	2.8908	.4894	.00738	.00741	.001694
NSE for post mean	1	.00027	.6762	.0916	1.7E-05	1.6E-05	2.3E-06
	2	.00039	.9795	.0451	1.7E-05	1.7E-05	2.3E-06
	3	.00012	1.203	.1175	1.9E-05	1.9E-05	2.4E-06
	4	.00013	.9552	.1463	2.2E-05	2.1E-05	2.9E-06
	5	.00008	.9910	.1225	1.5E-05	1.4E-05	2.2E-06
	6	.00010	.9460	.1258	1.4E-05	1.4E-05	2.4E-06
Prior		N(-.003, $.0017^2$)	N(-10, 5^2)	N(0.5, 10^6) s.t.[-.99,.99]	Dirichlet $a_1 = 34$	prior $a_2 = 64$	$a_3 = 2$

the corresponding AR(1) process. Then, $\Delta\mathcal{V}_{t,i}$ are adjusted to satisfy the observed choice optimality constraints with a small margin. It is also possible to adjust $\epsilon_{t,i}$ together with $\Delta\mathcal{V}_{t,i}$. If the initial values for $\epsilon_{t,i}$ are not simulated from the AR(1) process or if the starting value for ρ is far in the tail of the posterior, a procedure similar to the simulated annealing might be helpful in starting the Gibbs sampler with high acceptance rates: the acceptance probabilities for the parameters are multiplied by a decreasing quantity on the first hundred iterations.

Estimation results for 6 posterior simulator runs are presented in Table 1.6 and Figure 1.11. The number of draws for each simulator run was equal to 1000000. Three different random grids for solving the DP were used in these experiments (the random grid is generated before the estimation procedure starts and it stays fixed through the simulator run, Proposition 2.6 justifies using the same random grid at each iteration of the algorithm if the number of the nearest neighbors is constant.) One grid was used for runs 1–2, another one for runs 3–4, and the last one for runs 5–6. The random number generator was initialized differently for each run.

Convergence of the Gibbs sampler draws to the stationary distribution could be judged by comparing the posteriors obtained for the same realizations of the random grid. For all the parameters but ρ convergence was attained in all the runs. Convergence for ρ was clearly attained only for runs 5–6. To reduce the role of the MCMC slow convergence for ρ in evaluating the effects of the random grid on the estimation results I combine the draws from the simulator runs corresponding to the same realizations of the random grid. The three posterior densities corresponding to runs 1–2, 3–4, and 5–6 are presented in Figure 1.11. The figure and table above suggest that only the estimation results for ρ are significantly affected by the random grid realization. The results for strongly identified η are not affected at all.

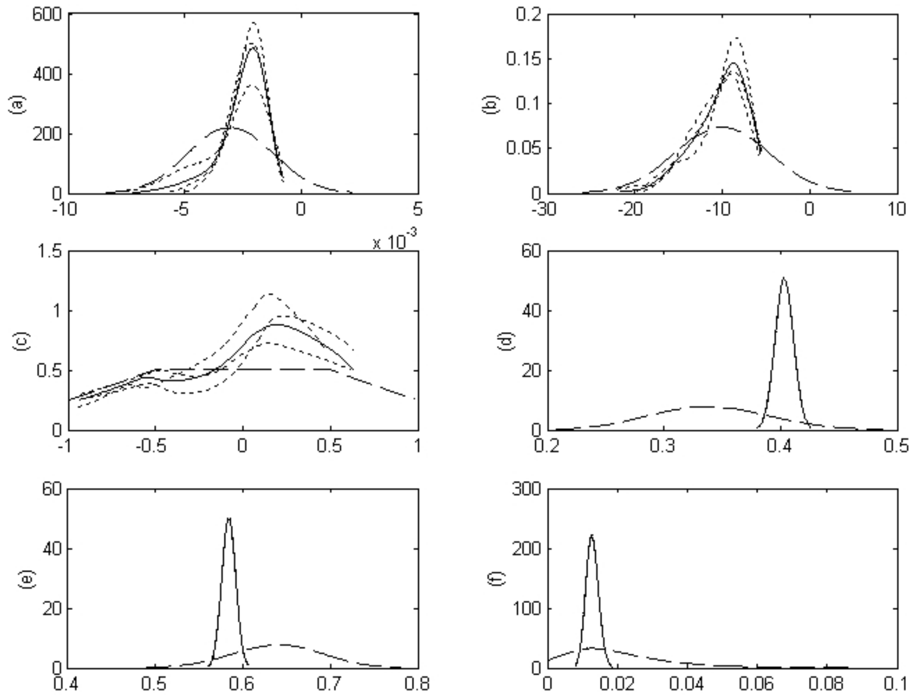

Figure 1.11: Estimated posterior densities for different grids: (a) α_1, (b) α_2, (c) ρ, (d) η_1, (e) η_2, (f) η_3. The dashed lines are prior densities. The solid lines are posterior densities averaged over all simulator runs. The dotted lines show posterior densities averaged for runs 1–2, 3–4, and 5–6.

The qualitative results for ρ do not seem to depend on the grid realization. The posterior distribution for ρ is bimodal. The higher mode is positive and located at about 0.2, the lower mode is at about -0.6. The posterior mean is close to 0. The posterior probability of $\rho > 0$ is in 0.54–0.66 range. Overall, there seems to be no strong evidence that Rust's assumption of no serial correlation in the unobservables is invalid.

A more objective criterion for studying the effects of the grid realization on the estimation results would be to check how it affects conditional choice probabilities or results of some policy changes. If the effects are still present then there are several alternative ways to reduce them. The first one is to estimate the posterior densities from several posterior simulator runs corresponding to different grids. This was done for the experiment above and the resulting densities are shown by the solid lines in Figure 1.11.

Increasing the size of the grid seems to be a more attractive way to obtain better approximations for the posterior distribution. However, it would increase the computational burden of solving the DP and approximating the expectations in the Gibbs sampler. In both cases this burden can be ameliorated. As was described in Section 1.5.1.4, solving the DP on several small random grids and combining the results produces about the same approximation precision as solving the DP on one big random grid. However, using several smaller grids requires much less time. To speed up the approximation of the expectations in the Gibbs sampler a strategy proposed by Keane and Wolpin (1994) could be used. In solving the DP, the authors compute expectations using Monte Carlo integration only for a subset of states in the discretized state space. For the rest of the states the expectations are computed by interpolation. Such an interpolation function could be used for approximating the expectations in the Gibbs sampler.

Chapter 3 presents another approach to estimation of dynamic discrete choice models that avoids the problem of estimation results dependence on the random grid realization. The future expected value function can be seen as a function of the parameters, the current state, and the chosen alternative. Experiments in this chapter showed that kernel smoothing does not provide sufficiently good approximations of this function. It turns out that artificial neural networks do. The DP solving algorithm presented in this chapter can produce very precise approximations of the value functions as described in Section 1.5.1.4. It is not feasible to get such precision for a million of parameter draws required to reasonably approximate the posterior distributions. However, it is feasible for several thousand draws from a prior distribution. These precise approximations can be used to train an artificial neural network beforehand of the estimation procedure. Then this neural network can be used in the Gibbs sampler for approximating the expectations of the value functions. Experiments in Chapter 3 suggest that this indeed is a promising alternative.

1.6 Conclusion and future work

This chapter presents a method for Bayesian inference in dynamic discrete choice models with serially correlated unobserved state variables. It constructs the Gibbs sampler, employing data augmentation and Metropolis steps, that can successfully handle multidimensional integration in the likelihood function of these models. The computational burden of solving the DP at each iteration of the estimation algorithm can be reduced by efficient use of the information obtained on previous iterations. A proof of the complete uniform convergence of the proposed DP solution approximations to the true values is obtained under mild assumptions on the primitives of the model. In Bayesian analysis, inference results are often represented in terms of posterior expectations. The chapter establishes the complete convergence of the posterior expectations approximated by the proposed method.

Serially correlated unobservables are not the only possible source of intractable integrals in the likelihood function of DDCMs. The Gibbs sampler algorithm can be extended to tackle other cases as well. First, missing observations could be handled by data augmentation in this framework. An example of that is different observation frequencies for different variables in a dataset, e.g. HRS interviews are conducted biannually but the attached data from the Social Security Administration and Medicare records are available monthly or even daily. Also, adding a macro shock or a cohort effect into a model is equally easy. It would amount to adding another block in the Gibbs sampler.

The method is experimentally evaluated on two different dynamic discrete choice models. First of all, estimation experiments show that ignoring serial correlation in unobservables can lead to serious misspecification errors. Second, parameters in DDCMs are often weakly identified. In Bayesian inference, uncertainty about parameters is treated explicitly. This advantage of Bayesian methods seems to be very important for DDCMs. Since the proposed estimation method allows for serial correlation in unobservables and accounts for uncertainty about parameters, its application is likely to improve the reliability of the answers to the policy questions that a DDCM can provide.

There are several directions in which the method could be improved. First, the amount of serial correlation in posterior simulator draws is very large. Thus, long posterior simulator runs are required for exploring the posterior distributions. Developing strategies for decreasing the amount of serial correlation in the Gibbs sampler draws is an important area for future research. Due to the high computational demand the method is implemented in C. Application of parallel computing seems to be a fruitful way to achieve higher performance of the method in the future.

Experiments with the DP solving algorithm led to a discovery of several significant practical improvements in the algorithm. First, the approximations of the expected value function obtained for fixed parameters and different realizations of the random grid behave as if an analog of a CLT with respect to the size of the random grid holds. This suggests that solving the DP on several small random grids and combining the results is a very efficient alternative to using one large grid. Theoretical justification of this observed improvement in the algorithm performance could be a subject of future work. Second, a difference of expected value functions can be approximated by the DP solving algorithm with a much higher precision than an expected value function by itself. This is taken into account in the construction of the Gibbs sampler. As a result, the realization of the random grid on which the DP is solved does not seem to seriously alter the results of estimation even for small grid sizes.

The flexibility of the framework and extensive experimentation were crucial for making the proposed approach successful. Nevertheless, combined with efficient

DP solution strategies, standard tools of Bayesian analysis—Gibbs sampling, data augmentation, and the Metropolis-Hastings algorithm—seem to be very promising in making more elaborate dynamic discrete choice models estimable.

CHAPTER 2 PROOFS OF THE THEORETICAL RESULTS

2.1 Lemmas

Lemma 2.1. *Given $\tilde{\epsilon} > 0$, there exist $\delta > 0$ and T such that for any $\theta \in \Theta$, $s \in S$, and $t > T$:*

$$P(|A_1^t(\theta, s)| > \tilde{\epsilon}) \leq e^{-\delta \tilde{N}(t)\hat{N}(t-N(t))} \leq e^{-0.5\delta t^{\gamma_1}} \tag{2.1}$$

Proof. Fix a combination $m = \{m_1, \ldots, m_{\tilde{N}(t)}\}$ from $\{t-N(t), \ldots, t-1\}$. Assumption 1.7 defines $X(\omega^{t-1}, \theta, s, d, m)$. Since the importance sampling weights are bounded away from zero by $\underline{f} > 0$ (see Assumption 1.4),

$$\begin{aligned}
&[X(\omega^{t-1}, \theta, s, d, m) > \tilde{\epsilon}] \\
\subset \; & \left[|\sum_{i=1}^{\tilde{N}(t)} \sum_{j=1}^{\hat{N}(m_i)} \frac{(V(s^{m_i,j};\theta) - E[V(s';\theta) \mid s, d; \theta])f(s^{m_i,j} \mid s, d; \theta)/g(s^{m_i,j})}{\sum_{r=1}^{\tilde{N}(t)} \hat{N}(m_r) \inf_{\theta,s,s',d} f(s'|s,d;\theta)/g(s')}| > \tilde{\epsilon} \right] \\
= \; & \left[|\sum_{i=1}^{\tilde{N}(t)} \sum_{j=1}^{\hat{N}(m_i)} (V(s^{m_i,j};\theta) - E[V(s';\theta) \mid s, d; \theta])f(s^{m_i,j} \mid s, d; \theta)/g(s^{m_i,j})| > \right. \\
& \left. \tilde{\epsilon}\underline{f} \sum_{i=1}^{\tilde{N}(t)} \hat{N}(m_i) \right]
\end{aligned} \tag{2.2}$$

Using (2.2) and then applying Hoeffding (1963)'s inequality we get

$$\begin{aligned}
&P(X(\omega^{t-1}, \theta, s, d, m) > \tilde{\epsilon}) \\
\leq \; & P\left[|\sum_{i=1}^{\tilde{N}(t)} \sum_{j=1}^{\hat{N}(m_i)} (V(s^{m_i,j};\theta) - E[V(s';\theta) \mid s, d; \theta])f(s^{m_i,j} \mid s, d; \theta)/g(s^{m_i,j})| > \right. \\
& \left. \tilde{\epsilon}\underline{f} \sum_{i=1}^{\tilde{N}(t)} \hat{N}(m_i) \right] \leq 2\exp\left\{ \frac{-2\underline{f}^2\tilde{\epsilon}^2}{(b-a)^2} \sum_{r=1}^{\tilde{N}(t)} \hat{N}(m_r) \right\}
\end{aligned} \tag{2.3}$$

where a and b are correspondingly the lower and upper bounds on $(V(s^{m_i,j};\theta) - E[V(s';\theta) \mid s; \theta])f(s^{m_i,j} \mid s, d; \theta)/g(s^{m_i,j})$. Hoeffding's inequality applies since $s^{m_i,j}$ are independent, the summands have expectations equal to zero:

$$\int \frac{(V(s^{m_i,j};\theta) - E[V(s';\theta) \mid s;\theta])f(s^{m_i,j} \mid s,d;\theta)}{g(s^{m_i,j})} g(s^{m_i,j}) ds^{m_i,j} = 0 \tag{2.4}$$

and a and b are finite by Assumptions 1.1, 1.3, and 1.4.

Since $\hat{N}(.)$ is non-decreasing, (2.3) implies

$$\begin{aligned} & P(X(\omega^{t-1},\theta,s,d,m) > \tilde{\epsilon}) \leq 2\exp\left\{\frac{-2\underline{f}^2\tilde{\epsilon}^2}{(b-a)^2}\tilde{N}(t)\hat{N}(t-N(t))\right\} \\ & = 2\exp\left\{-4\delta\tilde{N}(t)\hat{N}(t-N(t))\right\} \end{aligned} \tag{2.5}$$

where the last equality defines $\delta > 0$.

Since $|A_1^t(\theta,s,d)| < \max_m X(\omega^{t-1},\theta,s,d,m)$:

$$\begin{aligned} P(|A_1^t(\theta,s,d)| > \tilde{\epsilon}) & \leq P[\max_m X(\omega^{t-1},\theta,s,d,m) > \tilde{\epsilon}] \\ = & \; P(\cup_m[X(\omega^{t-1},\theta,s,d,m) > \tilde{\epsilon}]) \leq \sum_m P[X(\omega^{t-1},\theta,s,d,m) > \tilde{\epsilon}] \\ \leq & \; 2\exp\left\{-4\delta\tilde{N}(t)\hat{N}(t-N(t))\right\}\frac{N(t)!}{(N(t)-\tilde{N}(t))!\tilde{N}(t)!} \end{aligned} \tag{2.6}$$

where the summation, the maximization, and the union are taken over all possible combinations m and $N(t)!/((N(t)-\tilde{N}(t))!\tilde{N}(t)!)$ is the number of the possible combinations.

Assumption 1.6 and Proposition 2.8 show that $\exists T_1$ such that $\forall t > T_1$,

$$\exp\left\{-4\delta\tilde{N}(t)\hat{N}(t-N(t))\right\}\frac{N(t)!}{(N(t)-\tilde{N}(t))!\tilde{N}(t)!} \leq \exp\left\{-2\delta\tilde{N}(t)\hat{N}(t-N(t))\right\} \tag{2.7}$$

Finally,

$$\begin{aligned} P(|A_1^t(\theta,s)| > \tilde{\epsilon}) &= P(\max_{d\in D}|A_1^t(\theta,s,d)| > \tilde{\epsilon}) = P(\cup_{d\in D}[|A_1^t(\theta,s,d)| > \tilde{\epsilon}]) \\ &\leq \text{card}(D)2\exp\{-2\delta\tilde{N}(t)\hat{N}(t-N(t))\}, \forall t > T_1 \\ &\leq \exp\{-\delta\tilde{N}(t)\hat{N}(t-N(t))\}, \forall t > T_2 \geq T_1 \qquad (2.8) \end{aligned}$$

where such T_2 exists since $\text{card}(D)2\exp\{-\delta\tilde{N}(t)\hat{N}(t-N(t))\} \to 0$. The last inequality in (2.1) follows since $\tilde{N}(t)\hat{N}(t-N(t)) \geq t^{\gamma_1} - t^{\gamma_1-\gamma_2} \geq 0.5t^{\gamma_1}$ for any t larger than some $T \geq T_2$.

□

Lemma 2.2. *Given $\tilde{\epsilon} > 0$, there exist $\delta > 0$ and T such that for any $\theta \in \Theta$, $s \in S$, and $t > T$:*

$$P(|A_2^t(\theta,s)| > \tilde{\epsilon}) \leq e^{-\delta(N(t)-\tilde{N}(t))} \leq e^{-0.5\delta t^{\gamma_1}} \qquad (2.9)$$

Proof.

$$\begin{aligned} \left[\left|A_2^t(\theta,s,d)\right| > \tilde{\epsilon}\right] &= \left[\left|\sum_{i=1}^{\tilde{N}(t)}\sum_{j=1}^{\hat{N}(t)}(V(s^{k_i,j};\theta^{k_i}) - V(s^{k_i,j};\theta))W_{k_i,j,t}(s,d,\theta)\right| > \tilde{\epsilon}\right] \\ &\subset \left[\sum_{i=1}^{\tilde{N}(t)}\sum_{j=1}^{\hat{N}(t)}|V(s^{k_i,j};\theta^{k_i}) - V(s^{k_i,j};\theta)|W_{k_i,j,t}(s,d,\theta) > \tilde{\epsilon}\right] \\ &\subset \left[\exists k_i, j : |V(s^{k_i,j};\theta^{k_i}) - V(s^{k_i,j};\theta)| > \tilde{\epsilon}\right] \qquad (2.10) \end{aligned}$$

Since $V(s;\theta)$ is continuous, and $\Theta \times S$ is a compact, $\exists\tilde{\delta}_{\tilde{\epsilon}} > 0$ such that $||(s_1,\theta_1) - (s_2,\theta_2)|| \leq \tilde{\delta}_{\tilde{\epsilon}}$ implies $|V(s_1;\theta_1) - V(s_2;\theta_2)| \leq \tilde{\epsilon}$. Therefore,

$$\begin{aligned} &\left[\exists k_i, j : |V(s^{k_i,j};\theta^{k_i}) - V(s^{k_i,j};\theta)| > \tilde{\epsilon}\right] \\ &\subset \ [\exists k_i, j : ||(s^{k_i,j},\theta^{k_i}) - (s^{k_i,j},\theta)|| > \tilde{\delta}_{\tilde{\epsilon}}] = [\exists k_i : ||\theta^{k_i} - \theta|| > \tilde{\delta}_{\tilde{\epsilon}}] \qquad (2.11) \end{aligned}$$

Because k_i are the indices of the parameters from the previous iterations that are the closest to θ:

$$\begin{aligned}
&[\exists k_i : ||\theta^{k_i} - \theta|| > \tilde{\delta}_{\tilde{\epsilon}}] \\
&\quad \subset \quad [\forall j \in \{t - N(t), \dots, t-1\} \setminus \{k_1, \dots, k_{\tilde{N}(t)}\} : ||\theta^j - \theta|| > \tilde{\delta}_{\tilde{\epsilon}}] \\
&\quad \subset \quad \bigcup_{(j_1, \dots, j_{N(t)-\tilde{N}(t)}) : j_m \in \{t-N(t), \dots, t-1\}, m \neq l \Rightarrow j_m \neq j_l} \quad \bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[||\theta^{j_m} - \theta|| > \tilde{\delta}_{\tilde{\epsilon}} \right] \qquad (2.12)
\end{aligned}$$

Fix some $(j_1, \dots, j_{N(t)-\tilde{N}(t)})$, then by Assumption 1.5:

$$\begin{aligned}
&P([||\theta^{j_m} - \theta|| > \tilde{\delta}_{\tilde{\epsilon}}]|\omega^{j_m - 1}) \\
&\quad = \quad 1 - P([||\theta^{j_m} - \theta|| < \tilde{\delta}_{\tilde{\epsilon}}]|\omega^{j_m - 1}) \leq 1 - \hat{\delta}\lambda[B_{\tilde{\delta}_{\tilde{\epsilon}}}(\theta) \cap \Theta] \\
&\quad \leq \quad 1 - \hat{\delta}[\tilde{\delta}_{\tilde{\epsilon}}/J_{\Theta}^{0.5}]^J = \exp\{-4(-0.25 \log(1 - \hat{\delta}[\tilde{\delta}_{\tilde{\epsilon}}/J_{\Theta}^{0.5}]^J))\} = e^{-4\delta} \qquad (2.13)
\end{aligned}$$

where the last equality defines $\delta > 0$, J_Θ is the dimensionality of rectangle Θ, $B_{.}(.)$ is a ball in R^{J_Θ}. It holds for any history $\omega^{j_m - 1}$; thus for fixed $(j_1, \dots, j_{N(t)-\tilde{N}(t)})$

$$P\left(\bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[||\theta^{j_m} - \theta|| > \tilde{\delta}_{\tilde{\epsilon}} \right] \right) \leq e^{-4\delta(N(t) - \tilde{N}(t))} \qquad (2.14)$$

Since the union in (2.12) is taken over $N(t)!/(\tilde{N}(t)!(N(t) - \tilde{N}(t))!)$ events

$$\begin{aligned}
P\left[\left| A_2^t(x, \theta, \epsilon) \right| > \tilde{\epsilon} \right] \quad &\leq \quad e^{-4\delta(N(t)-\tilde{N}(t))} \frac{N(t)!}{\tilde{N}(t)!(N(t) - \tilde{N}(t))!} \\
&\leq \quad e^{-2\delta(N(t)-\tilde{N}(t))}, \forall t > T_2 \qquad (2.15)
\end{aligned}$$

where the second inequality and existence of T_2 follows from Assumption 1.6 and Proposition 2.8. Finally,

$$\begin{aligned}
P(|A_2^t(\theta, s)| > \tilde{\epsilon}) \quad &= \quad P(\max_{d \in D} |A_2^t(\theta, s, d)| > \tilde{\epsilon}) = P(\cup_{d \in D} [|A_2^t(\theta, s, d)| > \tilde{\epsilon}]) \\
&\leq \quad \text{card}(D) e^{-2\delta(N(t) - \tilde{N}(t))}, \forall t > T_2 \\
&\leq \quad e^{-\delta(N(t) - \tilde{N}(t))}, \forall t > T_3 \geq T_2 \qquad (2.16)
\end{aligned}$$

where such T_3 exists since $\text{card}(D)e^{-\delta(N(t)-\tilde{N}(t))} \to 0$. The last inequality in (2.9) follows since $N(t)-\tilde{N}(t) \geq [t^{\gamma_1} - [t^{\gamma_2}] \geq t^{\gamma_1} - 1 - t^{\gamma_2} \geq 0.5t^{\gamma_1}$ for any t larger than some $T \geq T_3$. □

Lemma 2.3. *Given $\tilde{\epsilon} > 0$, there exist $\delta > 0$ and T such that $\forall \theta \in \Theta$, $\forall s \in S$, and $\forall t > T$:*

$$P(|A_3^t(\theta, s)| > \tilde{\epsilon}) \leq e^{-\delta t^{\gamma_0 \gamma_1}} \tag{2.17}$$

Proof. First let's show that for any positive integer m, $\forall \theta \in \Theta$, and $\forall s \in S$

$$A_3^t(\theta, s) \leq \frac{\beta}{1-\beta}\left[\max_{i=t-mN(t),t-1}\left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j})\right) + \max_{i=t-mN(t),t-1}\left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j})\right)\right] + \beta^m \max_{i=t-mN(t),t-1}\left(\max_{j=1,\hat{N}(i)} A_3^i(\theta^i, s^{i,j})\right) \tag{2.18}$$

By definition

$$A_3^t(\theta, s, d) = \left|\sum_{i=1}^{\tilde{N}(t)} \sum_{j=1}^{\hat{N}(k_i)} (V^{k_i}(s^{k_i,j}; \theta^{k_i}) - V(s^{k_i,j}; \theta^{k_i}))W_{k_i,j,t}(s, d, \theta)\right| \tag{2.19}$$

Since $\max_d a(d) - \max_d b(d) \leq \max_d\{a(d) - b(d)\}$

$$\begin{aligned}
&|V^{k_i}(s^{k_i,j}; \theta^{k_i}) - V(s^{k_i,j}; \theta^{k_i})| \\
= &\ |\max_{d\in D}\left\{u(s^{k_i,j}, d) + \beta\hat{E}^{(k_i)}[V(s'; \theta^{k_i}) \mid s^{k_i,j}, d; \theta^{k_i}]\right\} \\
&- \max_{d\in D}\left\{u(s^{k_i,j}, d) + \beta E[V(s'; \theta^{k_i}) \mid s^{k_i,j}, d; \theta^{k_i}]\right\}| \\
\leq &\ |\max_{d\in D}\left\{\beta\hat{E}^{(k_i)}[V(s'; \theta^{k_i}) \mid s^{k_i,j}, d\theta^{k_i}]] - \beta E[V(s'; \theta^{k_i}) \mid s^{k_i,j}, d; \theta^{k_i}]\right\}| \qquad (2.20)
\end{aligned}$$

From (2.20) and definition of $A_l^t(.)$ given in Theorem 1.1,

$$\begin{aligned}
&|V^{k_i}(s^{k_i,j}; \theta^{k_i}) - V(s^{k_i,j}; \theta^{k_i})| \\
\leq &\ \beta \max_{d\in D}\left(A_1^{k_i}(\theta^{k_i}, s^{k_i,j}, d) + A_2^{k_i}(\theta^{k_i}, s^{k_i,j}, d) + A_3^{k_i}(\theta^{k_i}, s^{k_i,j}, d)\right) \\
\leq &\ \beta\left(A_1^{k_i}(\theta^{k_i}, s^{k_i,j}) + A_2^{k_i}(\theta^{k_i}, s^{k_i,j}) + A_3^{k_i}(\theta^{k_i}, s^{k_i,j})\right) \qquad (2.21)
\end{aligned}$$

Combining (2.19) and (2.21) gives:

$$\begin{aligned} A_3^t(\theta, s, d) &\leq \beta \sum_{i=1}^{\tilde{N}(t)} \sum_{j=1}^{\hat{N}(k_i)} \left(A_1^{k_i}(\theta^{k_i}, s^{k_i,j}) + A_2^{k_i}(\theta^{k_i}, s^{k_i,j}) + A_3^{k_i}(\theta^{k_i}, s^{k_i,j}) \right) W_{k_i,j,t}(s, d, \theta) \\ &\leq \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j}) \right) + \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j}) \right) \\ &\quad + \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_3^i(\theta^i, s^{i,j}) \right) \qquad (2.22) \end{aligned}$$

where the second inequality follows from the fact that $\forall i \in \{1, \ldots, \tilde{N}(t)\}$, $k_i \in \{t - N(t), \ldots, t-1\}$ and the weights sum to one. Since the r.h.s. of (2.22) does not depend on d:

$$\begin{aligned} & A_3^t(\theta, s) \leq \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j}) \right) + \qquad (2.23) \\ & \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j}) \right) + \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_3^i(\theta^i, s^{i,j}) \right) \end{aligned}$$

To facilitate the description of the iterative process on (2.23) that will lead to (2.18) let $M(t, 0) = t$ and $M(t, i) = M(t, i-1) - N(M(t, i-1))$, then

$$\begin{aligned} A_3^t(\theta, s) &\leq \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j}) \right) + \beta \max_{i=t-N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j}) \right) \\ &\quad + \beta^2 \max_{i=t-N(t)-N[t-N(t)],t-2} \left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j}) \right) \\ &\quad + \beta^2 \max_{i=t-N(t)-N[t-N(t)],t-2} \left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j}) \right) \\ &\quad + \beta^2 \max_{i=t-N(t)-N[t-N(t)],t-2} \left(\max_{j=1,\hat{N}(i)} A_3^i(\theta^i, s^{i,j}) \right) \end{aligned}$$

$$\begin{aligned} \leq & \sum_{k=1}^{m} \beta^k \left[\max_{i=M(t,k),t-k} \left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j}) \right) + \max_{i=M(t,k),t-k} \left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j}) \right) \right] \\ & + \beta^m \max_{i=M(t,m),t-m} \left(\max_{j=1,\hat{N}(i)} A_3^i(\theta^i, s^{i,j}) \right) \qquad (2.24) \end{aligned}$$

from which (2.18) follows since $\sum_{k=1}^{m} \beta^k < \beta/(1-\beta)$, and $M(t, m) \geq t - mN(t)$ $\forall m$.

Inequality in (2.18) is shown to hold for any m. Let $m(t) = [(t - t^{\gamma_0})/N(t)]$ ($[x]$ is the integer part of x.) and notice that $M(t, m(t)) \geq t - m(t)N(t) \geq t^{\gamma_0}$. Since $A_3^i(\theta^i, s^{i,j})$ is bounded above by some $\bar{A}_3 < \infty$ (utility function and state and parameter spaces are bounded):

$$
\begin{aligned}
& P[|A_3^t(\theta, s)| > \tilde{\epsilon}] \\
& \quad \leq \; P[\frac{\beta}{1-\beta}\left\{ \max_{i=t-m(t)N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j}) \right) + \right. \\
& \qquad \left. \max_{i=t-m(t)N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j}) \right) \right\} \\
& \qquad + \beta^{m(t)} \max_{i=t-m(t)N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_3^i(\theta^i, s^{i,j}) \right) > \tilde{\epsilon}] \\
& \leq P\left[\max_{i=t-m(t)N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_1^i(\theta^i, s^{i,j}) \right) > \frac{\tilde{\epsilon}(1-\beta)}{3\beta} \right] \\
& \quad + P\left[\max_{i=t-m(t)N(t),t-1} \left(\max_{j=1,\hat{N}(i)} A_2^i(\theta^i, s^{i,j}) \right) > \frac{\tilde{\epsilon}(1-\beta)}{3\beta} \right] \\
& \quad + P\left[\beta^{m(t)} \bar{A}_3 > \frac{\tilde{\epsilon}}{3} \right] \\
\leq & \sum_{i=t-m(t)N(t)}^{t-1} \sum_{j=1}^{\hat{N}(i)} \left\{ P\left[A_1^i(\theta^i, s^{i,j}) > \frac{\tilde{\epsilon}(1-\beta)}{3\beta} \right] + P\left[A_2^i(\theta^i, s^{i,j}) > \frac{\tilde{\epsilon}(1-\beta)}{3\beta} \right] \right\}
\end{aligned}
$$

The last inequality holds for $t > T_3$, where T_3 satisfies $P(\beta^{m(t)} \bar{A}_3 > \tilde{\epsilon}/3) = 0, \forall t > T_3$. Such T_3 exists since $m(t) \to \infty$.

Since $t - m(t)N(t) \to \infty$, by Lemma 2.1 and Lemma 2.2, exist $\delta_1 > 0$, $\delta_2 > 0$, T_1, and T_2, such that for $\forall t > \max(T_1, T_2, T_3)$:

$$
P(|A_3^t(\theta, s)| > \tilde{\epsilon}) \leq \sum_{i=t-m(t)N(t)}^{t-1} \hat{N}(i) \left[e^{-\delta_1 \tilde{N}(i) \hat{N}(i - N(i))} + e^{-\delta_2 (N(i) - \tilde{N}(i))} \right] \qquad (2.25)
$$

Proposition 2.9 shows that exist $\delta > 0$ and T_4 such that the r.h.s of (2.25) is no larger than $\exp(-\delta t^{\gamma_0 \gamma_1}), \forall t > T_4$. Thus, setting $T = \max(T_1, T_2, T_3, T_4)$ completes the proof. □

2.2 Extension to the uniform convergence

First, note that the approximation error is not a continuous function of (θ, s). Thus, we cannot apply the standard results to show the measurability of the supremum of the approximation error over the state and parameter spaces. Proposition 2.1 and Proposition 2.3 can be used to establish the measurability in this case. Below, Lemma 2.4 shows that a uniform version of Lemma 2.1 holds given extra Assumption 1.7. Lemma 2.5 shows that a uniform version of Lemma 2.2 also holds. A uniform version of Lemma 2.3 holds trivially since the right hand side of the key inequality (2.18) does not depend on (θ, s). Theorem 1.2 follows from the uniform versions of the Lemmas in the same way as Theorem 1.1 follows from Lemmas 2.1-2.3.

Proposition 2.1. *Let $f(\omega, \theta)$ be a measurable function on $(\Omega \times \Theta, \sigma(\mathcal{A} \times \mathcal{B}))$ with values in R. Assume that Θ has a countable subset $\tilde{\Theta}$ and that for any $\omega \in \Omega$ and any $\theta \in \Theta$ there exists a sequence in $\tilde{\Theta}$, $\{\tilde{\theta}_n\}$ such that $f(\omega, \tilde{\theta}_n) \to f(\omega, \theta)$. Then, $\sup_{\theta \in \Theta} f(\omega, \theta)$ is measurable w.r.t. $(\Omega, \mathcal{A})$ (the proposition can be used to show that the supremum of a random function with some simple discontinuities, e.g. jumps, on a separable space is measurable.)*

To apply the proposition for establishing the measurability of the supremum of the approximation errors, let the set of rational numbers contained in $\Theta \times S$ play the role of the countable subset $\tilde{\Theta}$. Proposition 2.3 shows that for any given history ω^{t-1} and any (θ, s) it is always possible to find a sequence with rational coordinates $(\tilde{\theta}_n) \to \theta$ such that for all n, $(\tilde{\theta}_n)$ and θ have the same iteration indices for the nearest neighbors. For a given history ω^{t-1}, the approximation error is continuous in (θ, s)

on the subsets of $\Theta \times S$ that give the same iteration indices of the nearest neighbors. Using any rational sequence $s^n \to s$ gives $f(\omega, \widetilde{(\theta, s)}_n) \to f(\omega, (\theta, s))$ required in the proposition. Thus, the supremum of the approximation error is measurable.

Proof. First, let's show that for an arbitrary t

$$\cup_{\theta \in \Theta}[f(\omega, \theta) > t] = \cup_{\theta \in \tilde{\Theta}}[f(\omega, \theta) > t] \tag{2.26}$$

Assume $\omega_1 \in \cup_{\theta \in \Theta}[f(\omega, \theta) > t]$. It means there exists $\theta_1 \in \Theta$ such that $f(\omega_1, \theta_1) > t$. By the theorem's assumption $\exists\{\tilde{\theta}_n\}$ such that $f(\omega_1, \tilde{\theta}_n) \to f(\omega_1, \theta_1)$. Then, $\exists n$, $f(\omega_1, \tilde{\theta}_n) > t$. Thus, $\omega_1 \in \cup_{\theta \in \tilde{\Theta}}[f(\omega, \theta) > t]$ and (2.26) is proven.

Note that $[\sup_{\theta \in \Theta} f(\omega, \theta) > t] = \cup_{\theta \in \Theta}[f(\omega, \theta) > t] = \cup_{\theta \in \tilde{\Theta}}[f(\omega, \theta) > t]$ is a countable union of sets from $\mathcal{A}$ and thus also belongs to $\mathcal{A}$. □

Lemma 2.4. *Given $\tilde{\epsilon} > 0$, there exist $\delta > 0$ and T such that $\forall t > T$:*

$$P(\sup_{\theta \in \Theta, s \in S} |A_1^t(\theta, s)| > \tilde{\epsilon}) \leq e^{-\delta \tilde{N}(t) \hat{N}(t - N(t))} \leq e^{-0.5 \delta t^{\gamma_1}} \tag{2.27}$$

Proof. Fix a combination $m = \{m_1, \ldots, m_{\tilde{N}(t)}\}$ from $\{t - N(t), \ldots, t-1\}$. Assumption 1.7 defines $X(\omega^{t-1}, \theta, s, d, m)$. By Assumption 1.7, $\{X(\omega^{t-1}, \theta, s, d, m)\}_{\omega^{t-1}}$ are equicontinuous on $\Theta \times S$: there exists $\tilde{\delta}(\tilde{\epsilon}) > 0$ such that $||(\theta_1, s_1) - ((\theta_2, s_2))|| < \tilde{\delta}(\tilde{\epsilon})$ implies $|X(\omega^{t-1}, \theta_1, s_1, d, m) - X(\omega^{t-1}, \theta_2, s_2, d, m)| < \tilde{\epsilon}/2$. Since $\Theta \times S$ is a compact set it can be covered by M balls: $\Theta \times S \subset \cup_{i=1}^{M} B_i$ with radius $\tilde{\delta}(\tilde{\epsilon})$ and centers at (θ_i, s_i), where $M < \infty$ depends only on $\tilde{\epsilon}$. It follows that

$$\begin{aligned}
&[\sup_{\theta, s} X(\omega^{t-1}, \theta, s, d, m) > \tilde{\epsilon}] = \cup_{\theta, s}[X(\omega^{t-1}, \theta, s, d, m) > \tilde{\epsilon}] = \\
&\cup_{i=1}^{M} \cup_{(\theta, s) \in B_i} [X(\omega^{t-1}, \theta, s, d, m) > \tilde{\epsilon}]
\end{aligned} \tag{2.28}$$

Let's show that

$$\cup_{(\theta,s)\in B_i}[X(\omega^{t-1},\theta,s,d,m)>\tilde{\epsilon}]\subset[X(\omega^{t-1},\theta_i,s_i,d,m)>\frac{\tilde{\epsilon}}{2}] \tag{2.29}$$

If $\omega_*^{t-1}\in\cup_{(\theta,s)\in B_i}[X(\omega^{t-1},\theta,s,d,m)>\tilde{\epsilon}]$, then $\exists(\theta^*,s^*)\in B_i(\theta_i,s_i)$ such that $X(\omega_*^{t-1},\theta^*,s^*,d,m)>\tilde{\epsilon}$. Since $||(\theta^*,s^*)-(\theta_i,s_i)||\leq\tilde{\delta}(\tilde{\epsilon})$, $X(\omega_*^{t-1},\theta_i,s_i,d,m)\geq X(\omega_*^{t-1},\theta^*,s^*,d,m)-\tilde{\epsilon}/2$. This implies $\omega_*^{t-1}\in[X(\omega^{t-1},\theta_i,s_i,d,m)>\frac{\tilde{\epsilon}}{2}]$.

Since $\sup_{\theta,s}|A_1^t(\theta,s,d)|<\max_m\sup_{\theta,s}X(\omega^{t-1},\theta,s,d,m)$:

$$\begin{aligned}
&P(\sup_{\theta,s}|A_1^t(\theta,s,d)|>\tilde{\epsilon})\\
&\leq\ P[\max_m\sup_{\theta,s}X(\omega^{t-1},\theta,s,d,m)>\tilde{\epsilon}]\text{ (max is over all possible combinations } m)\\
&\leq\ P(\cup_m[\sup_{\theta,s}X(\omega^{t-1},\theta,s,d,m)>\tilde{\epsilon}]\\
&\leq\ \sum_m P[\sup_{\theta,s}X(\omega^{t-1},\theta,s,d,m)>\tilde{\epsilon}]\\
&\leq\ \sum_m P(\cup_{i=1}^M[X(\omega^{t-1},\theta_i,s_i,d,m)>\frac{\tilde{\epsilon}}{2}])\text{ (by (2.28) and (2.29))}\\
&\leq\ M\frac{N(t)!}{(N(t)-\tilde{N}(t))!\tilde{N}(t)!}2\exp\left\{-4\delta\tilde{N}(t)\hat{N}(t-N(t))\right\} \qquad (2.30)
\end{aligned}$$

where $N(t)!/((N(t)-\tilde{N}(t))!\tilde{N}(t)!)$ is the number of different combinations m and $2\exp\{-4\delta\tilde{N}(t)\hat{N}(t-N(t))\}$ is the bound from (2.5) in Lemma 2.1. From the last inequality, the proof follows steps of the argument starting after (2.6) in the proof of Lemma 2.1. □

Lemma 2.5. *Given $\tilde{\epsilon}>0$, there exist $\delta>0$ and T such that $\forall t>T$:*

$$P(\sup_{\theta,s}|A_2^t(\theta,s)|>\tilde{\epsilon})\leq e^{-\delta(N(t)-\tilde{N}(t))}\leq e^{-0.5\delta t^{\gamma_1}} \tag{2.31}$$

Proof. From Lemma 2.2

$$\left[\left|A_2^t(\theta,s,d)\right| > \tilde{\epsilon}\right]$$
$$\subset \bigcup_{(j_1,\ldots,j_{N(t)-\tilde{N}(t)}):j_m\in\{t-N(t),\ldots,t-1\},m\neq l\Rightarrow j_m\neq j_l} \bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[||\theta^{j_m}-\theta|| > \tilde{\delta}_{\tilde{\epsilon}}\right] \quad (2.32)$$

This implies that

$$\left[\sup_{\theta,s}\left|A_2^t(\theta,s,d)\right| > \tilde{\epsilon}\right] = \bigcup_{\theta,s}\left[\left|A_2^t(\theta,s,d)\right| > \tilde{\epsilon}\right] \quad (2.33)$$
$$\subset \bigcup_{\theta\in\Theta}\left\{\bigcup_{(j_1,\ldots,j_{N(t)-\tilde{N}(t)}):j_m\in\{t-N(t),\ldots,t-1\},m\neq l\Rightarrow j_m\neq j_l} \bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[||\theta^{j_m}-\theta|| > \tilde{\delta}_{\tilde{\epsilon}}\right]\right\}$$

Since Θ is a rectangle in R^{J_Θ} it can be covered by a finite number of balls with radius $\frac{\tilde{\delta}_{\tilde{\epsilon}}}{2}$:

$$\Theta \subset \cup_{i=1}^{M} B(\theta_i),\ M = \text{const}\cdot(\tilde{\delta}_{\tilde{\epsilon}}/2)^{-J_\Theta} \quad (2.34)$$

Let's prove the following fact:

$$\bigcup_{\theta\in B(\theta_i)} \bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[||\theta^{j_m}-\theta|| > \tilde{\delta}_{\tilde{\epsilon}}\right] \subset \bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[\theta^{j_m} \notin B(\theta_i)\right] \quad (2.35)$$

Assume $\omega^{t-1} \in \left(\bigcap_{m=1}^{N(t)-\tilde{N}(t)}[\theta^{j_m}\notin B(\theta_i)]\right)^c$. Then $\exists m$ such that $\theta^{j_m}\in B(\theta_i)$. It follows that $\forall\theta\in B(\theta_i), \exists\theta^{j_m} : ||\theta^{j_m}-\theta|| \leq \tilde{\delta}_{\tilde{\epsilon}}$. Thus, ω^{t-1} belongs to the following set:

$$\bigcap_{\theta\in B(\theta_i)} \bigcup_{m=1}^{N(t)-\tilde{N}(t)} \left[||\theta^{j_m}-\theta|| \leq \tilde{\delta}_{\tilde{\epsilon}}\right] = \left(\bigcup_{\theta\in B(\theta_i)} \bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[||\theta^{j_m}-\theta|| > \tilde{\delta}_{\tilde{\epsilon}}\right]\right)^c \quad (2.36)$$

Therefore, the claim in (2.35) is proven.

By the same argument as for (2.14) from Lemma 2.2, we can establish that

$$P\left(\bigcap_{m=1}^{N(t)-\tilde{N}(t)} \left[\theta^{j_m}\notin B(\theta_i)\right]\right) \leq e^{-4\delta(N(t)-\tilde{N}(t))} \quad (2.37)$$

for some positive δ.

From (2.33), (2.34) and (2.35)

$$\left[\sup_{\theta,s}\left|A_2^t(\theta,s,d)\right| > \tilde{\epsilon}\right]$$

$$\subset \bigcup_{(j_1,\ldots,j_{N(t)-\tilde{N}(t)}):j_m\in\{t-N(t),\ldots,t-1\},m\neq l\Rightarrow j_m\neq j_l} \bigcup_{i=1}^{M}\left(\bigcap_{m=1}^{N(t)-\tilde{N}(t)}[\theta^{j_m}\notin B(\theta_i)]\right) \tag{2.38}$$

Using (2.37) and (2.38) gives

$$P\left[\sup_{\theta,s}\left|A_2^t(\theta,s,d)\right| > \tilde{\epsilon}\right] \leq \frac{N(t)!}{\tilde{N}(t)!(N(t)-\tilde{N}(t))!}Me^{-4\delta(N(t)-\tilde{N}(t))} \tag{2.39}$$

The rest of the proof follows the corresponding steps in Lemma 2.2.

□

2.3 Convergence of posterior expectations and ergodicity of the Gibbs sampler

Proof. (Theorem 1.3.) First, let's introduce some notation shortcuts:

$$r = r(\theta,\mathcal{V},\epsilon;F(\theta,\epsilon))$$

$$\hat{r} = r(\theta,\mathcal{V},\epsilon;\hat{F}^n(\theta,\epsilon))$$

$$1_{\{\}} = 1_{\Theta}(\theta)\cdot\left(\prod_{i,t}1_E(\epsilon_{t,i})p(d_{t,i}|\mathcal{V}_{t,i})\right)\cdot\left(\prod_{i,t,k}1_{[-\overline{\nu},\overline{\nu}]}(q(\theta,\mathcal{V}_{t,i},\epsilon_{t,i},F_{t,i}(\theta,\epsilon_{t,i})))\right)$$

$$\hat{1}_{\{\}} = 1_{\Theta}(\theta)\cdot\left(\prod_{i,t}1_E(\epsilon_{t,i})p(d_{t,i}|\mathcal{V}_{t,i})\right)\cdot\left(\prod_{i,t,k}1_{[-\overline{\nu},\overline{\nu}]}(q(\theta,\mathcal{V}_{t,i},\epsilon_{t,i},\hat{F}^n_{t,i}(\theta,\epsilon_{t,i})))\right)$$

$$\int h(\theta,\mathcal{V},\epsilon)d(\theta,\mathcal{V},\epsilon) = \int h$$

$$p = p(\theta,\mathcal{V},\epsilon;F|d,x) = \frac{r\cdot 1_{\{\}}}{\int r\cdot 1_{\{\}}}$$

$$\hat{p} = p(\theta, \mathcal{V}, \epsilon; \hat{F}^n | d, x) = \frac{\hat{r} \cdot \hat{1}_{\{\}}}{\int \hat{r} \cdot \hat{1}_{\{\}}}$$

The probability that the approximation error exceeds $\varepsilon > 0$ can be bounded by the sum of two terms:

$$P\left[\left|\int h \cdot p - \int h \cdot \hat{p}\right| > \varepsilon\right] \leq P(||F - \hat{F}|| > \delta_F) \tag{2.40}$$

$$+P\left([\left|\int h \cdot p - \int h \cdot \hat{p}\right| > \varepsilon] \cap [||F - \hat{F}|| \leq \delta_F]\right) \tag{2.41}$$

where $||F - \hat{F}|| = \sup_{s,\theta,d} |F(s, \theta, d) - \hat{F}(s, \theta, d)|$ and $F(s, \theta, d)$ is the expected value function (or the difference of expected value functions, depending on the parameterization of the Gibbs sampler) and $\hat{F}$ is the approximation to F from the DP solving algorithm on its iteration n (fixed in this proof.) I will show that for a sufficiently small $\delta_F > 0$, the set in (2.41) is empty. Then, by Theorem 1.2, the term in (2.40) can be bounded by z_n corresponding to δ_F.

$$[\left|\int h \cdot p - \int h \cdot \hat{p}\right| > \varepsilon] \cap [||F - \hat{F}|| \leq \delta_F]$$

$$\subset \left[\int |p - \hat{p}| > \varepsilon/||h||\right] \cap [||F - \hat{F}|| \leq \delta_F]$$

$$\subset \left(\left[\int_{\hat{1}_{\{\}}=1_{\{\}}} |p - \hat{p}| > \varepsilon/(2||h||)\right] \cap [||F - \hat{F}|| \leq \delta_F]\right) \tag{2.42}$$

$$\cup \left(\left[\int_{\hat{1}_{\{\}}\neq 1_{\{\}}} |p - \hat{p}| > \varepsilon/(2||h||)\right] \cap [||F - \hat{F}|| \leq \delta_F]\right) \tag{2.43}$$

Let's start with (2.42):

$$\left(\left[\int_{\hat{1}_{\{\}}=1_{\{\}}} |p - \hat{p}| > \varepsilon/(2||h||)\right] \cap [||F - \hat{F}|| \leq \delta_F]\right)$$

$$= \left[\int_{\hat{1}_{\{\}}=1_{\{\}}} |\frac{r}{\int r \cdot 1_{\{\}}} - \frac{\hat{r}}{\int \hat{r} \cdot \hat{1}_{\{\}}}| > \frac{\varepsilon}{2||h||}\right] \cap [||F - \hat{F}|| \leq \delta_F]$$

$$\subset \left[||\frac{r}{\int r \cdot 1_{\{\}}} - \frac{\hat{r}}{\int \hat{r} \cdot \hat{1}_{\{\}}}|| > \frac{\varepsilon}{2||h||\overline{\lambda}}\right] \cap [||F - \hat{F}|| \leq \delta_F] \tag{2.44}$$

where $\overline{\lambda} < \infty$ is the Lebesgue measure of the space for the parameters and the latent variables. For $\delta_{Sp} \in (0, \int r \cdot 1_{\{\}})$:

$$\begin{aligned}
&\left[||\frac{r}{\int r \cdot 1_{\{\}}} - \frac{\hat{r}}{\int \hat{r} \cdot \hat{1}_{\{\}}}|| > \frac{\varepsilon}{2||h||\overline{\lambda}}\right] \cap [||F - \hat{F}|| \leq \delta_F] = \\
&\left(\left[||\frac{r}{\int r \cdot 1_{\{\}}} - \frac{\hat{r}}{\int \hat{r} \cdot \hat{1}_{\{\}}}|| > \frac{\varepsilon}{2||h||\overline{\lambda}}\right] \cap [||F - \hat{F}|| \leq \delta_F]\right. \\
&\qquad \left. \cap \left[\left|\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}\right| > \delta_{Sp}\right]\right) \qquad (2.45) \\
&\bigcup \left(\left[||\frac{r}{\int r \cdot 1_{\{\}}} - \frac{\hat{r}}{\int \hat{r} \cdot \hat{1}_{\{\}}}|| > \frac{\varepsilon}{2||h||\overline{\lambda}}\right] \cap [||F - \hat{F}|| \leq \delta_F]\right. \\
&\qquad \left. \cap \left[\left|\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}\right| \leq \delta_{Sp}\right]\right) \qquad (2.46)
\end{aligned}$$

By Proposition 2.2 for δ_{Sp} there exists $\delta_F^1 > 0$ such that $[\left|\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}\right| > \delta_{Sp}] = \emptyset$. Thus, (2.45) (the whole two-line expression in parentheses) is the empty set for any $\delta_F < \delta_F^1$. Now, let's work with (2.46) (again, both lines in parentheses.)

$$\begin{aligned}
||\frac{r}{\int r \cdot 1_{\{\}}} - \frac{\hat{r}}{\int \hat{r} \cdot \hat{1}_{\{\}}}|| &\leq \frac{||r|| \cdot |\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}|}{\int r \cdot 1_{\{\}} \cdot \int \hat{r} \cdot \hat{1}_{\{\}}} + \frac{||\hat{r} - r||}{\int \hat{r} \cdot \hat{1}_{\{\}}} \\
&\leq \frac{||r|| \cdot |\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}|}{\int r \cdot 1_{\{\}} \cdot (\int r \cdot 1_{\{\}} - \delta_{Sp})} + \frac{||\hat{r} - r||}{\int r \cdot 1_{\{\}} - \delta_{Sp}} \qquad (2.47)
\end{aligned}$$

This inequality shows that (2.46) is a subset of the union of the following two sets:

$$\left[\frac{||r|| \cdot |\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}|}{\int r \cdot 1_{\{\}} \cdot (\int \int r \cdot 1_{\{\}} - \delta_{Sp})} > \frac{\varepsilon}{4||h||\overline{\lambda}}\right] \cap [||F - \hat{F}|| \leq \delta_F] \cap \left[\left|\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}\right| \leq \delta_{Sp}\right] \tag{2.48}$$

and

$$\left[\frac{||\hat{r} - r||}{\int r \cdot 1_{\{\}} - \delta_{Sp}} > \frac{\varepsilon}{4||h||\overline{\lambda}}\right] \cap [||F - \hat{F}|| \leq \delta_F] \cap \left[\left|\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}\right| \leq \delta_{Sp}\right] \tag{2.49}$$

I will show that both of them are empty for sufficiently small δ_F. By Proposition 2.2 there exists $\delta_F^2 > 0$ such that

$$\left[\left|\int r \cdot 1_{\{\}} - \int \hat{r} \cdot \hat{1}_{\{\}}\right| > \frac{\varepsilon \cdot \int r \cdot 1_{\{\}} \cdot (\int r \cdot 1_{\{\}} - \delta_{Sp})}{4||h||\overline{\lambda}||r||}\right] = \emptyset$$

whenever $||F - \hat{F}|| \leq \delta_F^2$. Therefore, (2.48) is equal to the empty set for $\delta_F \leq \delta_F^2$. Since r is continuous in components of F, there exists $\delta_F^3 > 0$ such that

$$||\hat{r} - r|| < \frac{\varepsilon \cdot (\int r \cdot 1_{\{\}} - \delta_{Sp})}{4||h||\overline{\lambda}||r||}$$

whenever $||F - \hat{F}|| \leq \delta_F^3$. Therefore, for $\delta_F \leq \delta_F^3$, (2.49) is equal to the empty set and so is (2.46). Thus, so far we showed that (2.42) is equal to the empty set for $\delta_F \leq \min_{i=1,2,3}(\delta_F^i)$.

Now, let's work with (2.43). Note that

$$\int_{\hat{1}_{\{\}} \neq 1_{\{\}}} |p - \hat{p}| \leq (\frac{||r||}{\int r \cdot 1_{\{\}}} + \frac{||\hat{r}||}{\int \hat{r} \cdot \hat{1}_{\{\}}}) \int_{\hat{1}_{\{\}} \neq 1_{\{\}}} 1 \leq (\frac{||r||}{\int r \cdot 1_{\{\}}} + \frac{||\hat{r}||}{\int r \cdot 1_{\{\}} - \delta_{Sp}}) \int_{\hat{1}_{\{\}} \neq 1_{\{\}}} 1$$

Thus, (2.43) is a subset of the following set:

$$\begin{aligned} &\left(\left[\int_{\hat{1}_{\{\}} \neq 1_{\{\}}} |p - \hat{p}| > \varepsilon/(2||h||)\right] \cap [||F - \hat{F}|| \leq \delta_F]\right) \\ \subset &\left(\left[\int_{\hat{1}_{\{\}} \neq 1_{\{\}}} 1 > \frac{\varepsilon}{2||h||(\frac{||r||}{\int r \cdot 1_{\{\}}} + \frac{||\hat{r}||}{\int r \cdot 1_{\{\}} - \delta_{Sp}})}\right] \cap [||F - \hat{F}|| \leq \delta_F]\right) \end{aligned} \tag{2.50}$$

Using the same argument as the one starting from (2.54) in Proposition 2.2, I can show that there exists $\delta_F^4 > 0$ such that $\forall \delta_F < \delta_F^4$, (2.43) will be the empty set. Setting $\delta_F = \min_{i=1,2,3,4}\{\delta_F^i\}$ completes the proof of the theorem.

□

Proposition 2.2. *For any $\varepsilon > 0$ there exists $\delta_F > 0$ such that*

$$\left[||F - \hat{F}|| < \delta_F\right] \cap \left[\left|\int \hat{r} \cdot \hat{1}_{\{\}} - \int r \cdot 1_{\{\}}\right| > \varepsilon\right] = \emptyset \tag{2.51}$$

Proof.

$$\begin{aligned}
\left[\left|\int \hat{r} \cdot \hat{1}_{\{\}} - \int r \cdot 1_{\{\}}\right| > \varepsilon\right] &\subset \left[\int |\hat{r} \cdot \hat{1}_{\{\}} - r \cdot 1_{\{\}}| > \varepsilon\right] \\
&\subset \left[\int_{\hat{1}_{\{\}}=1_{\{\}}} |\hat{r} \cdot \hat{1}_{\{\}} - r \cdot 1_{\{\}}| > \varepsilon/2\right] &(2.52) \\
&\quad \cup \left[\int_{\hat{1}_{\{\}}\neq 1_{\{\}}} |\hat{r} \cdot \hat{1}_{\{\}} - r \cdot 1_{\{\}}| > \varepsilon/2\right] &(2.53)
\end{aligned}$$

Let's show that the intersection of (2.52) and $[||F - \hat{F}|| < \delta_F]$ is the empty set for a sufficiently small δ_F.

$$\left[\int_{\hat{1}_{\{\}}=1_{\{\}}} |\hat{r} \cdot \hat{1}_{\{\}} - r \cdot 1_{\{\}}| > \varepsilon/2\right] \subset \left[\int_{\hat{1}_{\{\}}=1_{\{\}}} |\hat{r} - r| > \varepsilon/2\right] \subset \left[||\hat{r} - r|| > \varepsilon/(2\overline{\lambda})\right]$$

where $\overline{\lambda} < \infty$ is the Lebesgue measure of the bounded space for the parameters and the latent variables on which the integration is performed: $\Theta \times E \times \ldots \times E \times \mathbf{V} \times \ldots \times \mathbf{V}$, where $\mathbf{V} \subset R$ is the space for the alternative specific value functions $\mathcal{V}_{t,d,i}$. By Assumption 1.8, r is continuous in components of F. Thus, $\exists \delta_F^1 > 0$ such that $||F - \hat{F}|| < \delta_F^1$ implies $||\hat{r} - r|| < \varepsilon/(2\overline{\lambda})$, which means that the intersection of (2.52) and $[||F - \hat{F}|| < \delta_F]$ is the empty set for $\forall \delta_F < \delta_F^1$.

Let's show that the intersection of (2.53) and $[||F - \hat{F}|| < \delta_F]$ is the empty set for a sufficiently small δ_F. First, note that

$$\int_{\hat{1}_{\{\}}\neq 1_{\{\}}} |\hat{r} \cdot \hat{1}_{\{\}} - r \cdot 1_{\{\}}| \leq (||r|| + ||\hat{r}||) \int_{\hat{1}_{\{\}}\neq \cdot 1_{\{\}}} 1 \tag{2.54}$$

where $||r|| < \infty$ and $||\hat{r}|| < \overline{r} < \infty$ for any $\hat{F}$ (everything is bounded in the model.) Thus,

$$\begin{aligned}
&[||F-\hat{F}|| < \delta_F] \cap \left[\int_{\hat{1}_{\{\}} \neq 1_{\{\}}} |\hat{r} \cdot \hat{1}_{\{\}} - r \cdot 1_{\{\}}| > \varepsilon/2 \right] \\
&\subset [||F-\hat{F}|| < \delta_F] \cap \left[\int_{\hat{1}_{\{\}} \neq 1_{\{\}}} 1 > \varepsilon/(2(||r|| + ||\hat{r}||)) \right] \\
&= [||F-\hat{F}|| < \delta_F] \cap \left[\lambda[\hat{1}_{\{\}} \neq 1_{\{\}}] > \varepsilon/(2(||r|| + ||\hat{r}||)) \right]
\end{aligned} \tag{2.55}$$

where $\lambda(.)$ is the Lebesgue measure on the space of the parameters and the latent variables.

By Assumption 1.8, q_k is continuous in components of F. Thus, for any $\delta_q > 0$ there exists $\delta_F(\delta_q) > 0$ such that $||F-\hat{F}|| < \delta_F(\delta_q)$ implies $\max_k ||\hat{q}_k - q_k|| < \delta_q$. On the space of the parameters and the latent variables (these are not subsets of the underlying probability space):

$$[(\theta, \mathcal{V}, \epsilon) : \hat{1}_{\{\}} \neq 1_{\{\}}] \subset \bigcup_{i,t,k} [(\theta, \mathcal{V}, \epsilon) : q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) \in B_{\delta_q}(\overline{\nu}) \cup B_{\delta_q}(-\overline{\nu})] \tag{2.56}$$

if $||F-\hat{F}|| < \delta_F(\delta_q)$. To prove this claim assume $\forall i,t,k$ $q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) \notin B_{\delta_q}(\overline{\nu}) \cup B_{\delta_q}(-\overline{\nu})$. So, the distance between q_k and the truncation region edges $-\overline{\nu}$ and $\overline{\nu}$ is larger than δ_q for all i,t,k. But then, since $||\hat{q}_k - q_k|| < \delta_q$, $\hat{1}_{\{\}} = 1_{\{\}}$ and the claim (2.56) is proven.

Note that

$$\begin{aligned}
&\lim_{\delta_q \to 0} \lambda \left(\bigcup_{i,t,k} [(\theta, \mathcal{V}, \epsilon) : q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) \in B_{\delta_q}(\overline{\nu}) \cup B_{\delta_q}(-\overline{\nu})] \right) \qquad (2.57) \\
&\leq \sum_{i,t,k} \lim_{\delta_q \to 0} \lambda[(\theta, \mathcal{V}, \epsilon) : q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) \in B_{\delta_q}(\overline{\nu}) \cup B_{\delta_q}(-\overline{\nu})] \\
&= \sum_{i,t,k} \lambda[(\theta, \mathcal{V}, \epsilon) : q_k(\theta, \mathcal{V}_{t,i}, \epsilon_{t,i}, F_{t,i}) \in \{\overline{\nu}, -\overline{\nu}\}]
\end{aligned}$$

where the last equality holds by the monotone property of measures (the Lebesgue measure in this case) and by the fact that $\cap_{\delta_q>0}[q_k \in B_{\delta_q}(\overline{\nu})] = [q_k = \overline{\nu}]$.

By Assumption 1.8, $\lambda[(\theta, \mathcal{V}, \epsilon) : q_k = \overline{\nu}] = \lambda[(\theta, \mathcal{V}, \epsilon) : q_k = -\overline{\nu}] = 0$. Therefore, the limit in (2.57) is equal to zero and there exists $\delta_q^* > 0$ such that if $||F - \hat{F}|| < \delta_F(\delta_q^*)$ then

$$\lambda[\hat{1}_{\{\}} \neq 1_{\{\}}] < \varepsilon/(2(||r|| + ||\hat{r}||))$$

So, $\forall \delta_F \in (0, \delta_F(\delta_q^*)]$ the intersection of (2.53) and $[||F - \hat{F}|| < \delta_F]$ is the empty set. Setting $\delta_F = \min\{\delta_F(\delta_q^*), \delta_F^1\}$ completes the proof of the proposition. □

Proof. **(Theorem 1.4)**

Consider the following uniform probability density:

$$\begin{aligned} q(\Delta\mathcal{V}, \theta, \epsilon) &= c \cdot 1_\Theta(\theta) \prod_{i,t} [1_E(\epsilon_{t,i}) \cdot p(d_{t,i}|\Delta\mathcal{V}_{t,i}) \\ &\quad \cdot 1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})])] \qquad (2.58) \end{aligned}$$

where c is a normalization constant. The corresponding probability measure is denoted by $Q(.)$.

Let's show that the transition probability measure for the Gibbs sampler satisfies the marginalization condition w.r.t. $Q(.)$:

$$P((\mathcal{V}^{m+1}, \theta^{m+1}, \epsilon^{m+1}) \in A | \mathcal{V}^m, \theta^m, \epsilon^m, d, x) \geq bQ(A), \forall \mathcal{V}^m, \theta^m, \epsilon^m$$

where $b > 0$ is a constant. Then, the uniform ergodicity follows from Proposition 2 in Tierney (1994).

$$
\begin{aligned}
& P((\mathcal{V}^{m+1}, \theta^{m+1}, \epsilon^{m+1}) \in A | \mathcal{V}^m, \theta^m, \epsilon^m, d, x) \qquad (2.59)\\
= & \int_{R\times\ldots\times R} \int_A \prod_{t,i} p(\Delta\tilde{\mathcal{V}}_{t,i}^{m+1} | \theta^m, \epsilon^m, d, x)\\
& p(\rho^{m+1} | \theta^m, \epsilon^m, \Delta\tilde{\mathcal{V}}^{m+1}, d, x) \cdot p(\alpha^{m+1} | \rho^{m+1} \eta^m, h_\epsilon^m, \alpha^m, \epsilon^m, \Delta\tilde{\mathcal{V}}^{m+1}, d, x)\\
& p(\eta^{m+1} | \ldots) \cdot p(h_\epsilon^{m+1} | \ldots) \prod_{t,i} p(\epsilon_{t,i}^{m+1} | \ldots)\\
& \prod_{t,i} p(\Delta\mathcal{V}_{t,i}^{m+1} | \theta^{m+1}, \epsilon^{m+1}, d, x)\, d(\Delta\tilde{\mathcal{V}}, \theta^{m+1}, \epsilon^{m+1}, \Delta\mathcal{V}^{m+1})
\end{aligned}
$$

where $p(.|.)$ are the densities for the Gibbs sampler blocks. Even though the Metropolis-Hastings is used in the blocks, the densities can be expressed by using the Dirac delta function (see, for example, chapter 4 in Geweke (2005).)

Given the assumptions on the support of $\nu_{t,i}$ let's show that there exist $\delta_1 > 0$ such that $\Delta\mathcal{V}_{t,i} \in [-\delta_1, \delta_1]$ implies $(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})]) \in [-\overline{\nu}, \overline{\nu}]$, $\forall\theta, \epsilon$. It was stated in the formulation of the theorem that $\overline{EV}$ is an upper bound on the absolute value of the expected value function. Note that an upper bound on the expected value function EV^{ub} exists. Let's show that it is no greater than $\overline{EV}$.

$$
\begin{aligned}
E[|V(s';\theta)|;\|s,d;\theta] &= E[|\max\{\alpha_1 x + \epsilon + \beta E[V(s'';\theta)|s', d_1;\theta],\\
&\qquad \alpha_2 + \nu + \beta E[V(s'';\theta)|s', d_2;\theta]\}|]\\
&\leq \overline{u} + \overline{\epsilon} + E[|\nu|] + \beta \mathrm{EV}^{ub} \qquad (2.60)
\end{aligned}
$$

It was also assumed in the theorem that $\Phi(-\overline{\nu}) < 0.25$, which implies $E[|\nu|] \leq 1 + E[\nu^2] \leq 1 + 2h_\nu^{-1}$. Since (2.60) holds for any (s, d, θ):

$$
\mathrm{EV}^{ub} \leq \frac{\overline{u} + \overline{\epsilon} + (1 + 2h_\epsilon^{-1})}{1-\beta} = \overline{EV}
$$

Therefore,

$$|[x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})]| \leq 2(\overline{u} + \overline{\epsilon} + \beta\overline{EV})$$

Let $\delta_1 = \overline{\nu} - 2(\overline{u} + \overline{\epsilon} + \beta\overline{EV})$, which is positive by the assumption of the theorem. Thus, for $|\Delta\mathcal{V}_{t,i}| \leq \delta_1$,

$$|\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})]| \leq \delta_1 + 2(\overline{u} + \overline{\epsilon} + \beta\overline{EV}) = \overline{\nu}$$

To find a lower bound on the integral in (2.59), let's restrict the integration over $\Delta\tilde{\mathcal{V}}_{t,i}^{m+1}$ to $|\Delta\tilde{\mathcal{V}}_{t,i}^{m+1}| \leq \delta_1$ and use only the parts of the block densities corresponding to the accepted draws. The parts of the block densities for the accepted draws are equal to the MH transition densities multiplied by the acceptance probabilities. For $(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})]) \in [-\overline{\nu}, \overline{\nu}]$, these densities for the accepted draws are positive and continuous on Θ, E, and $[\Delta\mathcal{V}_{t,i} \geq 0]$ (or $[\Delta\mathcal{V}_{t,i} < 0]$ depending on $d_{t,i}$) for all blocks, and thus bounded away from zero. Let's denote the common bound by $\delta > 0$. Then,

$$\begin{aligned}
P((\mathcal{V}^{m+1}, \theta^{m+1}, \epsilon^{m+1}) \in A | \mathcal{V}^m, \theta^m, \epsilon^m, d, x) \geq (\prod_{t,i} \delta_1 \delta) \\
\cdot \int_A 1_\Theta(\theta^{m+1}) \cdot \delta^4 \cdot \prod_{t,i}[\delta \cdot 1_E(\epsilon_{t,i})] \cdot \prod_{t,i} \delta \cdot p(d_{t,i}|\Delta\mathcal{V}_{t,i}) \\
\cdot 1_{[-\overline{\nu},\overline{\nu}]}(\Delta\mathcal{V}_{t,i} - [x_{t,i}\alpha_1 - \alpha_2 + \epsilon_{t,i} + F_{t,i}(\theta, \epsilon_{t,i})]) \\
d(\theta^{m+1}, \epsilon^{m+1}, \Delta\mathcal{V}^{m+1}) = \frac{1}{c}(\prod_{t,i} \delta)^2 \cdot \delta^4 \cdot \prod_{t,i} \delta_1 \cdot Q(A)
\end{aligned} \tag{2.61}$$

Also, since $Q(.)$ is absolutely continuous w.r.t. the posterior probability measure, the transition probability measure for the Gibbs sampler is irreducible w.r.t. the posterior probability measure. This completes the proof of the uniform ergodicity of the Gibbs sampler. □

2.4 Auxiliary results

Proposition 2.3. *For any $\{\theta^1, \dots, \theta^N\}$ and θ in R^n and any $\tilde{N} \leq N$, there exists a sequence of rational numbers $q_m \to \theta$ such that for any m, q_m and θ have the same set of indices for the nearest neighbors: $\{k_1, \dots, k_{\tilde{N}}\}$ defined by (1.13).*

Proof. The outcomes of selecting the nearest neighbors can be classified into two cases. The trivial one occurs when there exists a ball around θ with radius r such that $||\theta^{k_i} - \theta|| < r$ and $||\theta^j - \theta|| > r + d$ for $d > 0$ and $j \neq k_i$. Then, applying the triangle inequality twice we get $\forall q \in B_{d/4}(\theta)$, $||\theta^{k_i} - q|| < r + d/2 < ||\theta^j - q||$ $\forall j \neq k_i$. For this case the proposition holds trivially.

The other case occurs when there exists a ball at θ with radius r_1 such that the closure of the ball includes all the nearest neighbors and the boundary of the ball includes one or more θ^j that are not included in the set of the nearest neighbors. For this case, I will construct a ball in the vicinity of θ such that it can be made as close to θ as needed and such that for any point inside this ball the set of the nearest neighbors is the same as for θ.

As described in the paper body (see (1.13)), the selection of the nearest neighbors on the boundary of $B_{r_1}(\theta)$ is conducted by the lexicographic comparison of $(\theta^j - \theta)$. Let's denote vectors $(\theta^j - \theta)$ such that θ^j is on the boundary of $B_{r_1}(\theta)$: $||\theta^j - \theta|| = r_1$ by $x^{0,i}$, $i = 1, \dots, M_x^0$. The results of the lexicographic selection process can be represented as follows:

$$\begin{aligned} z^{k,i} &= (r_1 - a_1, \ldots, r_{k-1} - a_{k-1}, \quad z_k^{k,i} \quad, \ldots, z_n^{k,i}) \\ x^{k,i} &= (r_1 - a_1, \ldots, r_{k-1} - a_{k-1}, \quad r_k - a_k \quad, x_{k+1}^{k,i}, \ldots, x_n^{k,i}) \\ y^{k,i} &= (r_1 - a_1, \ldots, r_{k-1} - a_{k-1}, \quad y_k^{k,i} \quad, \ldots, y_n^{k,i}) \end{aligned} \tag{2.62}$$

where a geometric interpretation of variables r_k and a_k is given in the figure below,

$$z_k^{k,i} > r_k - a_k > y_k^{k,i} \tag{2.63}$$

and $k = 1, \ldots, K$ for some $K \leq n$. Vectors $z^{k,i}, i = 1, \ldots, M_z^k$ are those vectors included in the set of the nearest neighbors for which the decision of the inclusion was obtained from the lexicographic comparison for the coordinate k. Vectors $x^{k,i}, i = 1, \ldots, M_x^k$ are the vectors for which the decision has not yet been made after comparing coordinates k. Vectors $y^{k,i}, i = 1, \ldots, M_y^k$ are the vectors for which the decision of not including them in the set of the nearest neighbors was obtained from comparing coordinate k. Vectors $x^{k+1,i}, y^{k+1,i}, z^{k+1,i}$ are all selected from $x^{k,i}$. The lexicographic selection will end at some coordinate K with unique x^K. This vector is denoted by x not by z to emphasize the fact that if there are multiple repetitions of $\theta + x^K = \theta^i = \theta^j, \ i \neq j$ in the history, then not all the repetitions have to be selected for the set of the nearest neighbors (the ones with larger iteration number will be selected first.) Of course, this is true only for the last selected nearest neighbor, for all the previous ones all the repetitions are included. Note that vectors $z^{k,i}, x^{k,i}, y^{k,i}$ are constructed in the system of coordinates with the origin at θ; so, we should add θ to all of them to get back to the original coordinate system.

A graphical illustration might be helpful for understanding the idea of the proof (the proof was actually constructed from similar graphical examples in R^2 and R^3.)

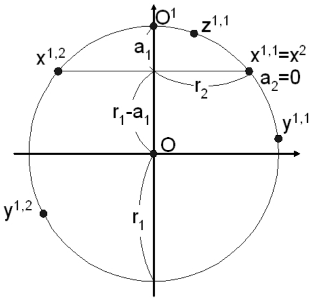

Figure 2.1: Nearest neighbors.

The figure shows an example, in which 2 nearest neighbors have to be chosen for point O. Since the required number of the nearest neighbors is smaller than the number of the points on the circle, we can always find a_1 such that all the points with the first coordinate strictly above $r_1 - a_1$ will be included in the set of the nearest neighbors and all the points with the first coordinate strictly below $r_1 - a_1$ will not be. For the points with the coordinate equal to $r_1 - a_1$, the selection process continues to the next dimension.

If we did not use the lexicographic comparison and just resolved the multivaluedness of $\arg\min$ by choosing vectors with larger iteration numbers first, than the proposition would not hold (a counterexample could be easily found in R^2.)

If the following conditions hold than the same nearest neighbors from the surface of $B_{r_1}(\theta)$ will be chosen for $(\theta+b)$ and θ:

$$||b-y^{k,i}|| > ||b-x^K|| > ||b-z^{k,i}||, \forall k,i \tag{2.64}$$

The condition says that $(x^K+\theta)$, which is the last nearest neighbor selected for θ, also has to be selected last for $(\theta+b)$ and that vectors on the boundary of $B_{r_1}(\theta)$ that are not the selected nearest neighbors for θ $(y^{k,i}, \forall k, i)$ should not be the selected nearest neighbors for $(\theta+b)$. Since $||y^{k,i}|| = ||x^K|| = ||z^{k,i}|| = r_1$, these conditions are equivalent to the following:

$$b^T(x^K - y^{k,i}) > 0 \text{ and } b^T(z^{k,i} - x^K) > 0 \tag{2.65}$$

Define

$$d = \min_{k=1,K} \min\{\min_i [z_k^{k,i} - (r_k - a_k)],\ \min_i [(r_k - a_k) - y_k^{k,i}]\},\ d > 0 \text{ by construction.}$$

For given $\epsilon_1 > 0$, let

$$\begin{aligned} \epsilon_{k+1} &= \min\{\epsilon_k,\ \epsilon_k d/(4nr_1)\}, \epsilon(\epsilon_1) = (\epsilon_1, \ldots, \epsilon_n) \\ \delta(\epsilon_1) &= \epsilon_n d/(8nr_1) \end{aligned} \tag{2.66}$$

Let $b \in B_{\delta(\epsilon_1)}(\epsilon(\epsilon_1))$ and $l = b - \epsilon(\epsilon_1)$. Let's show that (2.65) holds for any such b.

$$b^T(x^K - y^{k,i}) = (r_k - a_k - y_k^{k,i})\epsilon_k + \sum_{m=k+1}^{n} (x_m^K - y_m^{k,i})\epsilon_m + \sum_{m=k}^{n} (x_m^K - y_m^{k,i}) l_m \tag{2.67}$$

Note that $|l_k| \leq \delta(\epsilon_1)$ and $|x_m^K - y_m^{k,i}| \leq 2r_1$:

$$\begin{aligned} b^T(x^K - y^{k,i}) &\geq (r_k - a_k - y_k^{k,i})\epsilon_k - n2r_1 \max_{m=k+1,n} \epsilon_m - n2r_1\delta(\epsilon_1) \\ &\geq d\epsilon_k - n2r_1 \frac{\epsilon_k d}{4nr_1} - n2r_1 \frac{\epsilon_k d}{8nr_1} = d\epsilon_k/4 > 0 \end{aligned} \tag{2.68}$$

Analogously,

$$\begin{aligned} b^T(z^{k,i} - x^K) &\geq [z_k^{k,i} - (r_k - a_k)]\epsilon_k + \sum_{m=k+1}^{n} (z_m^{k,i} - x_m^K)\epsilon_m + \sum_{m=k}^{n} (z_m^{k,i} - x_m^K) l_m \\ &\geq d\epsilon_k - n2r_1 \max_{m=k+1,n} \epsilon_m - n2r_1\delta(\epsilon_1) \geq d\epsilon_k/4 > 0 \qquad (2.69) \end{aligned}$$

Thus, the order of selecting the nearest neighbors on the surface of $B_{r_1}(\theta)$ is the same for θ and any $\theta + b$ if $b \in B_{\delta(\epsilon_1)}(\epsilon(\epsilon_1))$ for any $\epsilon_1 > 0$. Making ϵ_1 sufficiently small, we can guarantee that all θ^j satisfying $||\theta^j - \theta|| < r_1$ will be chosen as the nearest neighbors for $\theta + b$ before the vectors on the surface of $B_{r_1}(\theta)$ and that θ^j satisfying $||\theta^j - \theta|| > r_1$ will not be chosen at all. For any $\epsilon_1 > 0$, $B_{\delta(\epsilon_1)}(\theta + \epsilon(\epsilon_1))$ will contain rational numbers. Letting a positive sequence $\{\epsilon_1^m\}$ go to zero and choosing $q_m \in B_\delta(\theta + \epsilon^m) \cap Q$ will give the sought sequence $\{q_m\}$. □

Proposition 2.4. *If Θ and S are compact, $u(s, d; \theta)$ is continuous in (s, θ), and $f(s' \mid s, d; \theta)$ is continuous in (θ, s, s'), then $V(s; \theta)$ and*
$E\{V(s'; \theta)|s, d; \theta\}$ are continuous in (θ, s).

Proof. The proof of the proposition follows closely the standard proof of the continuity of value functions with respect to the state variables (see, for example, chapters 3 and 4 of Stokey and Lucas (1989).) Let's consider the Bellman operator Γ on the Banach space of bounded functions B with sup norm: $V : \Theta \times S \to X$, where X is a bounded subset of R:

$$\Gamma(V)(s; \theta) = \max_d \{u(s, d; \theta) + \beta \int V(s'; \theta) f(s'|s, d; \theta) ds'\}$$

Blackwell's sufficient conditions for contraction are satisfied for this operator; so, Γ is a contraction mapping on B. The set of continuous functions C is a closed subset in

B. Thus, it suffices show that $\Gamma(C) \subset C$ (this trivially implies that the fixed point of Γ is a continuous function.)

Let $V(s;\theta)$ be a continuous function in B ($V \in C$). Let's show that $\Gamma(V)$ is also continuous.

$$\begin{aligned}
&|\Gamma(V)(s_1;\theta_1) - \Gamma(V)(s_2;\theta_2)| \leq \max_d |u(s_1,d;\theta_1) - u(s_2,d;\theta_2) \\
&+\beta \int V(s';\theta_1) f(s'|s_1,d;\theta_1)ds' - \beta \int V(s';\theta_2) f(s'|s_2,d;\theta_2)ds'| \\
&\leq \max_d |u(s_1,d;\theta_1) - u(s_2,d;\theta_2)| \\
&+\beta \max_d |\int [V(s';\theta_1) f(s'|s_1,d;\theta_1) - V(s';\theta_2) f(s'|s_2,d;\theta_2)]ds'|
\end{aligned} \tag{2.70}$$

Given $\epsilon > 0$ there exists $\delta_1 > 0$ such that $||(s_1;\theta_1) - (s_2;\theta_2)|| < \delta_1$ implies $\max_d |u(s_1,d;\theta_1) - u(s_2,d;\theta_2)| < \epsilon/2$.

$$\begin{aligned}
&|\int [V(s';\theta_1) f(s'|s_1,d;\theta_1) - V(s';\theta_2) f(s'|s_2,d;\theta_2)]ds'| \\
&\leq \max_d \sup_{s'} |V(s';\theta_1) f(s'|s_1,d;\theta_1) - V(s';\theta_2) f(s'|s_2,d;\theta_2)| \cdot \lambda(S)
\end{aligned} \tag{2.71}$$

Since $V(s';\theta) f(s'|s,d;\theta)$ is continuous on compact $\Theta \times S \times S$, for $\epsilon > 0$ there exists $\delta_2^d > 0$ such that $||(s_1,s';\theta_1) - (s_2,s';\theta_2)|| = ||(s_1;\theta_1) - (s_2;\theta_2)|| < \delta_2^d$ implies

$$\sup_{s'} |V(s';\theta_1) f(s'|s_1,d;\theta_1) - V(s';\theta_2) f(s'|s_2,d;\theta_2)| < \frac{\epsilon}{2\lambda(S)}$$

Thus, for $\delta = \min\{\delta_1, \min_d \delta_2^d\}$, $||(s_1;\theta_1) - (s_2;\theta_2)|| < \delta$ implies $|\Gamma(V)(s_1;\theta_1) - \Gamma(V)(s_2;\theta_2)| < \epsilon$. So, $\Gamma(V)$ is a continuous function. The continuity of $E\{V(s';\theta)|s,d;\theta\}$ follows from the continuity of $V(s';\theta)$ by an analogous argument.

□

Proposition 2.5. *Assumption 1.7 holds if Θ and S are compacts, $V(s;\theta)$ and $E[V(s';\theta) \mid s, d; \theta]$ are continuous in (θ, s), and $f(s' \mid s, d; \theta)/g(s')$ is continuous in (θ, s, s') and satisfies Assumption 1.4.*

Proof. Let's introduce the following notation shortcuts. T will denote the number of terms in the sum defining $X(\omega^{t-1}, \theta, s, d, m)$. Consider two arbitrary points: (θ_1, s_1) and (θ_2, s_2), let $V_j^i = V(s^j; \theta_i) - EV(\theta_i, s_i)$ and

$$W_j^i = \frac{f_j^i/g_j^i}{\sum f_k^i/g_k^i} = \frac{f(s^j \mid s_i, d; \theta_i)/g(s^j)}{\sum f(s^k \mid s_i, d; \theta_i)/g(s^k)}$$

Then,

$$\begin{aligned} |X(\omega^{t-1}, \theta_1, s_1, d, m) - X(\omega^{t-1}, \theta_2, s_2, d, m)| &= |\sum_{j=1}^{T} V_j^1 W_j^1 - \sum_{j=1}^{T} V_j^2 W_j^2 \pm \sum_{j=1}^{T} V_j^2 W_j^1| \\ &\leq |\sum_{j=1}^{T} (V_j^1 - V_j^2) W_j^1| \qquad (2.72) \\ &\quad +|\sum_{j=1}^{T} V_j^2 (W_j^1 - W_j^2)| \qquad (2.73) \end{aligned}$$

By the proposition's hypothesis, $V(s;\theta)$ and $E[V(s';\theta) \mid s, d; \theta]$ are continuous in (θ, s) on a compact set. Thus, given $\epsilon > 0$ $\exists \delta_1 > 0$ such that

$$||(\theta_1, s_1, s^j) - (\theta_2, s_2, s^j)|| = ||(\theta_1, s_1) - (\theta_2, s_2)|| < \delta_1$$

implies $|V(s^j; \theta_1) - EV(\theta_1, s_1) - (V(s^j; \theta_2) - EV(\theta_2, s_2))| < \epsilon/2$. Since the weights sum to one, (2.72) is bounded above by $\epsilon/2$. Let's similarly bound (2.73):

$$
\begin{aligned}
&|\sum_{j=1}^{T} V_j^2(W_j^1 - W_j^2)| = \left|\sum_{j=1}^{T} V_j^2\left(\frac{f_j^1/g_j^1}{\sum f_k^1/g_k^1} - \frac{f_j^2/g_j^2}{\sum f_k^2/g_k^2}\right)\right| \\
&= \left|\frac{(\sum f_k^2/g_k^2)\left[\sum V_j^2(f_j^1/g_j^1 - f_j^2/g_j^2)\right] + \left[\sum f_k^2/g_k^2 - \sum f_k^1/g_k^1\right]\left(\sum V_j^2 f_j^2/g_j^2\right)}{\sum f_k^1/g_k^1 \cdot \sum f_k^2/g_k^2}\right| \\
&\leq \frac{\overline{V} \cdot \max_j |f_j^1/g_j^1 - f_j^2/g_j^2| \cdot T}{\underline{f}T} + \frac{T \cdot \max_j |f_j^1/g_j^1 - f_j^2/g_j^2| \cdot \overline{V} \cdot \overline{f} \cdot T}{\underline{f}^2 T^2} \\
&\leq \max_j |f_j^1/g_j^1 - f_j^2/g_j^2| \cdot \overline{V}\left(\frac{1}{\underline{f}} + \frac{\overline{f}}{\underline{f}^2}\right) \qquad (2.74)
\end{aligned}
$$

where $\overline{f}$ and $\underline{f}$ are the upper and lower bounds on f/g introduced in Assumption 1.4; $\overline{V} < \infty$ is an upper bound on V_j^i. Since $f(s' \mid s, d; \theta)/g(s')$ is continuous in (θ, s, s') on compact $\Theta \times S \times S$, for any $\epsilon > 0$ there exists $\delta_2 > 0$ such that $||(\theta_1, s_1, s^j) - (\theta_2, s_2, s^j)|| = ||(\theta_1, s_1) - (\theta_2, s_2)|| < \delta_2$ implies

$$
|f(s^j \mid s_1, d; \theta_1)/g(s^j) - f(s^j \mid s_2, d; \theta_2)/g(s^j)| < \frac{\epsilon/2}{||V||(\frac{1}{\underline{f}} + \frac{\overline{f}}{\underline{f}^2})}, \quad \forall j
$$

Thus, (2.73) is also bounded above by $\epsilon/2$. For given $\epsilon > 0$, let $\delta = \min\{\delta_1, \delta_2\}$. Then, $||(\theta_1, s_1) - (\theta_2, s_2)|| < \delta$ implies $|X(\omega^{t-1}, \theta_1, s_1, d, m) - X(\omega^{t-1}, \theta_1, s_1, d, m)| < \epsilon/2 + \epsilon/2 = \epsilon$ □

Proposition 2.6. *Assume, that in the DP solving algorithm, the same random grid over the state space is used at each iteration: $s^{m_1,j} = s^{m_2,j} = s^j$ for any m_1, m_2, and j, where $s^j \overset{iid}{\sim} g(.)$. If the number of the nearest neighbors is constant: γ_2 in Assumption 1.6 is equal to zero and $\tilde{N}(t) = \tilde{N}$, then all the theoretical results proven in the paper will hold.*

Proof. Only the proof of Lemma 2.1 is affected by the change since in the other parts

I use only one fact about the weights in the importance sampling: the weights are in $[0,1]$. Thus let's show that Lemma 2.1 holds.

In Lemma 2.1 the terms in the sum (2.2) corresponding to the same s^j should be grouped into one term multiplied by the number of such terms:

$$\begin{aligned} & P(X(\omega^{t-1},\theta,s,d,m) > \tilde{\epsilon}) && (2.75) \\ = \ & P\left[| \sum_{j=1}^{\hat{N}(\max\{m_i\})} \frac{M_j(t,m)(V(s^j;\theta) - E[V(s';\theta) \mid s,d;\theta])f(s^j \mid s,d;\theta)/g(s^j)}{\sum_{r=1}^{\hat{N}(\max\{m_r\})} M_r(t,m) f(s^r \mid s,d;\theta)/g(s^r)} | > \tilde{\epsilon} \right] \\ \leq \ & P\left(| \sum_{j=1}^{\hat{N}(\max\{m_i\})} M_j(t,m)(V(s^j;\theta) - E[V(s';\theta) \mid s,d;\theta])f(s^j \mid s,d;\theta)/g(s^j)| \right. \\ & \quad \left. > \tilde{\epsilon}\underline{f}\hat{N}(\max\{m_i\}) \right) \end{aligned}$$

where $M_j(t,m) \in \{1,\ldots,\tilde{N}(t)\}$ denotes the number of the terms corresponding to s^j and $\hat{N}(\max\{m_r\})$ is the largest grid size. The inequality above follows since

$$\sum_{j=1}^{\hat{N}(\max\{m_i\})} M_j(t,m) f(s^j \mid s,d;\theta)/g(s^j) \geq \underline{f}\hat{N}(\max\{m_i\})$$

The summands in (2.75) are bounded by $(\tilde{N}a, \tilde{N}b)$, where a and b where defined in Lemma 2.1. Application of Hoeffding's inequality to (2.75) gives

$$\begin{aligned} & P(X(\omega^{t-1},\theta,s,d,m) > \tilde{\epsilon}) \\ \leq \ & 2\exp\left\{-4\delta\tilde{N}\hat{N}(\max\{m_i\}\right\} \leq 2\exp\left\{-4\delta\tilde{N}\hat{N}(t - N(t))\right\} \qquad (2.76) \end{aligned}$$

where $0 < \delta = \tilde{\epsilon}^2\underline{f}^2/(2(b-a)^2\tilde{N}^3)$. The rest of the argument follows the steps in Lemma 2.1 starting after (2.5). □

Proposition 2.7. *This proposition shows how to relax Assumption 1.4 for the state transition density and correspondingly change the DP solving algorithm so that the* *theoretical results proved in the paper would hold. Let the state space be a product of a finite set and a bounded rectangle in* $R^{J_{S_c}}$ $S = S_f \times S_c$. *Let* $f(s'_f, s'_c|s_f, s_c;\theta)$ *be the state transition density with respect to the product of the counting measure on* S_f *and the Lebesgue measure on* S_c. *Assume for any* $s_f \in S_f$ *and* $d \in D$ *we can define* $S(s_f, d) \subset S$ *such that* $f(s'_f, s'_c|s_f, s_c, d;\theta) > 0$ *for any* $(s'_f, s'_c) \in S(s_f, d)$ *and any* $s_c \in S_c$ *and for any* $(s'_f, s'_c) \notin S(s_f, d)$ *and any* $s_c \in S_c$ $f(s'_f, s'_c|s_f, s_c, d;\theta) = 0$. *For each* $s_f \in S_f$ *let density* $g_{s_f}(.)$ *be such that for any* $s_f \in S_f$

$$\inf_{\theta\in\Theta, s'_f, s'_c \in S(s_f), s_c \in S_c} f(s'_f, s'_c|s_f, s_c, d;\theta)/g_{s_f,d}(s'_f, s'_c) = \underline{f} > 0$$

$$\sup_{\theta\in\Theta, s'_f, s'_c \in S(s_f), s_c \in S_c, d\in D} f(s'_f, s'_c|s_f, s_c, d;\theta)/g_{s_f,d}(s'_f, s'_c) = \overline{f} < \infty$$

In the DP solving algorithm generate the random grid over the state space for each discrete state $s_f \in S_f$ *and decision* $d \in D$: $s^{m,j}_{s_f,d} \sim g_{s_f,d}(.)$ *and use these grids in computing the approximations of the expectations* $E(V(s';\theta)|s_f, s_c, d;\theta)$. *Then, all the theoretical results stated in the paper hold.*

If the transition for the discrete states is independent from the the other states, then a more efficient alternative would also work. Let's denote the transition probability for the discrete states by $f(s'_f|s_f, d;\theta)$. *Suppose that for* $f(s'_c|s_c, d;\theta)$ *and some* $g(.)$ *defined on* S_c *Assumption 1.4 holds, and the random grid* $s^{m,j}_c$ *is generated only on* S_c *from* $g(.)$. *Consider the following approximation of the expectations in the DP solving algorithm:*

$$\begin{aligned}&\hat{E}^{(m)}[V(s';\theta)|s_f, s_c, d;\theta] \qquad (2.77)\\ &= \sum_{s'_f \in S_f(s_f,d)} f(s'_f|s_f, d;\theta) \sum_{i=1}^{\tilde{N}(m)} \sum_{j=1}^{\hat{N}(k_i)} \frac{V^{k_i}(s'_f, s^{k_i,j};\theta^{k_i}) f(s^{k_i,j} \mid s, d;\theta)/g(s^{k_i,j})}{\sum_{r=1}^{\tilde{N}(m)} \sum_{q=1}^{\hat{N}(k_r)} f(s^{k_r,q} \mid s, d;\theta)/g(s^{k_r,q})}\end{aligned}$$

where $S_f(s_f, d)$ denotes the set of possible future discrete states given the current state s_f and decision d. Then, all the theoretical results stated in the paper hold.

Proof. Given the assumptions made in the first part of this proposition, the proofs of Lemma 2.1 and its uniform extension Lemma 2.4 apply without any changes. The rest of the results are not affected at all.

If (2.77) is used for approximating the expectations then in the proof of Lemmas 2.1 and 2.4 let's separate the expression for $X(.)$ into $K = \mathrm{card}(S_f(s_f, d))$ terms corresponding to each possible future discrete state:

$$\begin{aligned}
&X(\omega^{t-1}, \theta, s, d, m) = \\
&\quad f(s'_{f,1}|s_f, d; \theta) \quad \{\sum_{i=1}^{\tilde{N}(m)} \sum_{j=1}^{\hat{N}(k_i)} \frac{V^{k_i}(s'_{f,1}, s^{k_i,j}; \theta^{k_i}) f(s^{k_i,j}|s_c, d; \theta)/g(s^{k_i,j})}{\sum_{r=1}^{\tilde{N}(m)} \sum_{q=1}^{\hat{N}(k_r)} f(s^{k_r,q}|s_c; \theta)/g(s^{k_r,q})} \\
&\qquad -E[V(s'; \theta)|s'_f = s'_{f,1}, s_c, d; \theta]\} \\
&+ \qquad \ldots \qquad\qquad (2.78) \\
&+ \quad f(s'_{f,K}|s_f, d; \theta) \quad \{\sum_{i=1}^{\tilde{N}(m)} \sum_{j=1}^{\hat{N}(k_i)} \frac{V^{k_i}(s'_{f,K}, s^{k_i,j}; \theta^{k_i}) f(s^{k_i,j}|s_c, d; \theta)/g(s^{k_i,j})}{\sum_{r=1}^{\tilde{N}(m)} \sum_{q=1}^{\hat{N}(k_r)} f(s^{k_r,q}|s_c; \theta)/g(s^{k_r,q})} \\
&\qquad -E[V(s'; \theta)|s'_f = s'_{f,K}, s_c, d; \theta]\}
\end{aligned}$$

Then, applying the argument from Lemmas 2.1 and 2.4 we can bound the following probabilities for $k = 1, \ldots, K$:

$$\begin{aligned}
&P[|f(s'_{f,k}|s_f, d; \theta) \sum_{i=1}^{\tilde{N}(m)} \sum_{j=1}^{\hat{N}(k_i)} \frac{V^{k_i}(s'_{f,k}, s^{k_i,j}; \theta^{k_i}) f(s^{k_i,j}|s_c, d; \theta)/g(s^{k_i,j})}{\sum_{r=1}^{\tilde{N}(m)} \sum_{q=1}^{\hat{N}(k_r)} f(s^{k_r,q}|s_c; \theta)/g(s^{k_r,q})} && (2.79) \\
&\qquad -E[V(s'; \theta)|s'_f = s'_{f,k}, s_c, d; \theta]| > \frac{\epsilon}{K}] && (2.80)
\end{aligned}$$

and Lemmas 2.1 and 2.4 will hold. The proofs of the other Lemmas are not affected at all since the weights on the value functions in expectation approximations are still non-negative and sum to 1. □

Proposition 2.8. *If x_t, z_t, and y_t are integer sequences with $\lim_{t\to\infty} y_t/z_t = 0$, $\lim_{t\to\infty} z_t = \infty$, and $\limsup_{t\to\infty} z_t/x_t < \infty$ then $\forall\delta > 0$ $\exists T$ such that $\forall t > T$*

$$e^{-\delta x_t}\frac{z_t!}{(z_t-y_t)!y_t!} \le e^{-0.5\delta x_t}$$

Proof.

$$\begin{aligned}
\log\left[e^{-\delta x_t}\frac{z_t!}{(z_t-y_t)!y_t!}\right] &= -\delta x_t + \sum_{i=z_t-y_t+1}^{z_t}\log(i) - \sum_{i=1}^{y_t}\log(i)\\
&\le -\delta x_t + \int_{z_t-y_t+1}^{z_t+1}\log(i)di - \int_1^{y_t}\log(i)di\\
&= -\delta x_t + (z_t+1)\log(z_t+1) - (z_t-y_t+1)\log(z_t-y_t+1) -\\
&\quad [(z_t+1)-(z_t-y_t+1)] - \{y_t\log(y_t) - 1\log(1) - [y_t-1]\}\\
&= -\delta x_t + z_t[\log(z_t+1) - \log(z_t-y_t+1)] + y_t[\log(z_t-y_t+1) - \log(y_t)]\\
&\quad + \log(z_t+1) - \log(z_t-y_t+1) - y_t\log(y_t) - 1\\
&\le -\delta x_t + z_t\log\frac{z_t+1}{z_t-y_t+1} + y_t\log\frac{z_t-y_t+1}{y_t} + \log\frac{z_t+1}{z_t-y_t+1} = x_t\Bigg[-\delta +\\
&+ \frac{z_t}{x_t}\log\frac{z_t+1}{z_t-y_t+1} + \frac{(z_t-y_t+1)y_t}{x_t(z_t-y_t+1)}\log\frac{z_t-y_t+1}{y_t} + \frac{1}{x_t}\log\frac{z_t+1}{z_t-y_t+1}\Bigg] \quad (2.81)\\
&\le -0.5\delta x_t, \forall t > T
\end{aligned}$$

There exists such T that the last inequality holds since all the terms in (2.81) converge to zero. Exponentiating the obtained inequality completes the proof. □

Proposition 2.9. *For any $\delta_1 > 0$ and $\delta_2 > 0$ there exist $\delta > 0$ and T such that $\forall t > T$:*

$$\sum_{i=t-m(t)N(t)}^{t-1}\hat{N}(i)\left[e^{-\delta_1\tilde{N}(i)\hat{N}(i-N(i))} + e^{-\delta_2(N(i)-\tilde{N}(i))}\right] \le \exp\{-\delta t^{\gamma_0\gamma_1}\} \qquad (2.82)$$

Proof. The following inequalities will be used below:

$$t - m(t)N(t) \geq t - \frac{t - t^{\gamma_0}}{N(t)} N(t) = t^{\gamma_0} \tag{2.83}$$

$$t - m(t)N(t) \leq t - (\frac{t - t^{\gamma_0}}{N(t)} - 1)N(t) \leq t^{\gamma_0} + t^{\gamma_1} < 2t^{\gamma_0} \tag{2.84}$$

$$\tilde{N}(t-m(t)N(t)) = [(t-m(t)N(t))^{\gamma_2}] \geq [t^{\gamma_0\gamma_2}] \geq t^{\gamma_0\gamma_2}-1 \geq 0.5t^{\gamma_0\gamma_2},\ \forall t > T_1 = 2^{1/(\gamma_0\gamma_2)} \tag{2.85}$$

$$N(t - m(t)N(t)) = [(t - m(t)N(t))^{\gamma_1}] \leq (2t^{\gamma_0})^{\gamma_1} \leq 2^{\gamma_1} t^{\gamma_0\gamma_1} \tag{2.86}$$

$$\begin{aligned}
&\hat{N}(t - m(t)N(t) - N(t - m(t)N(t))) = [(t - m(t)N(t) - N(t - m(t)N(t)))^{\gamma_1-\gamma_2}] \\
&\geq (t^{\gamma_0} - 2^{\gamma_1} t^{\gamma_0\gamma_1})^{\gamma_1-\gamma_2} - 1, \text{ by (2.83) and (2.86)} \\
&\geq \frac{t^{\gamma_0(\gamma_1-\gamma_2)}}{2^{\gamma_1-\gamma_2}} - 1,\ \forall t > 2^{(1+\gamma_1/(\gamma_0(1-\gamma_1))} \\
&\geq \frac{t^{\gamma_0(\gamma_1-\gamma_2)}}{2^{1+\gamma_1-\gamma_2}},\ \forall t > T_2 = \max\{2^{(1+\gamma_1-\gamma_2/(\gamma_0(\gamma_1-\gamma_2))}, 2^{(1+\gamma_1/(\gamma_0(1-\gamma_1))}\}
\end{aligned} \tag{2.87}$$

Combining (2.85) and (2.87) gives:

$$\begin{aligned}
&\exp\{-\delta_1 \tilde{N}(t - m(t)N(t))\hat{N}(t - m(t)N(t) - N(t - m(t)N(t)))\} \\
&\leq \exp\{-\frac{\delta_1 t^{\gamma_0\gamma_1}}{2^{2+\gamma_1-\gamma_2}}\} = \exp\{-\tilde{\delta_1} t^{\gamma_0\gamma_1}\}
\end{aligned} \tag{2.88}$$

where the last equality defines $\tilde{\delta_1} > 0$.

$$\begin{aligned}
&N(t - m(t)N(t)) - \tilde{N}(t - m(t)N(t)) = [(t - m(t)N(t))^{\gamma_1}] - [(t - m(t)N(t))^{\gamma_2}] \\
&\geq [t^{\gamma_0\gamma_1}] - [2^{\gamma_2} t^{\gamma_0\gamma_2}], \text{ by (2.83) and (2.84)} \\
&\geq t^{\gamma_0\gamma_1} - 1 - 2^{\gamma_2} t^{\gamma_0\gamma_2} \\
&\geq 0.5t^{\gamma_0\gamma_1}, \text{ for } t \text{ larger than some } T_3
\end{aligned} \tag{2.89}$$

where such T_3 exists since $(0.5t^{\gamma_0\gamma_1} - 1 - 2^{\gamma_2}t^{\gamma_0\gamma_2}) \to \infty$.

Taking an upper bound on summands in (2.82) and multiplying it by the number of terms in the sum gives the following upper bound on the sum:

$$\begin{aligned}
&\sum_{i=t-m(t)N(t)}^{t-1} \hat{N}(i)\left[e^{-\delta_1\tilde{N}(i)\hat{N}(i-N(i))} + e^{-\delta_2(N(i)-\tilde{N}(i))}\right] \\
&\leq ((t-1)-(t-m(t)N(t))+1) \times \hat{N}(t-1) \times \\
&\qquad \times \left[e^{-\delta_1\tilde{N}(t-m(t)N(t))\hat{N}(t-m(t)N(t)-N(t-m(t)N(t)))}\right. \\
&\qquad\qquad \left. + e^{-\delta_2(N(t-m(t)N(t))-\tilde{N}(t-m(t)N(t)))}\right] \qquad (2.90)
\end{aligned}$$

Inequalities in (2.88), (2.89), and (2.90) imply:

$$\begin{aligned}
&\sum_{i=t-m(t)N(t)}^{t-1} \hat{N}(i)\left[e^{-\delta_1\tilde{N}(i)\hat{N}(i-N(i))} + e^{-\delta_2(N(i)-\tilde{N}(i))}\right] \\
&\leq t^{1+\gamma_1-\gamma_2}(\exp\{-\tilde{\delta_1}t^{\gamma_0\gamma_1}\} + \exp\{-0.5\delta_2 t^{\gamma_0\gamma_1}\}) \\
&\leq 2t^{1+\gamma_1-\gamma_2}\exp\{-\min(\tilde{\delta_1}, 0.5\delta_2)t^{\gamma_0\gamma_1}\} \qquad (2.91)
\end{aligned}$$

where $\tilde{\delta_1}$ was defined in (2.88).

Note that $(2t^{1+\gamma_1-\gamma_2}\exp\{-0.5\min(\tilde{\delta_1}, 0.5\delta_2)t^{\gamma_0\gamma_1}\}) \to \infty$ and therefore $\exists T \geq \max(T_1, T_2, T_3)$ such that $\forall t > T$

$$2t^{1+\gamma_1-\gamma_2}\exp\{-\min(\tilde{\delta_1}, 0.5\delta_2)t^{\gamma_0\gamma_1}\} \leq \exp\{-\delta t^{\gamma_0\gamma_1}\} \qquad (2.92)$$

where $\delta = 0.5\min(\tilde{\delta_1}, 0.5\delta_2)$. This completes the proof. □

Proposition 2.10. *For any $a > 0$ and $\delta > 0$, $\sum_{t=1}^{\infty}\exp\{-\delta t^a\} < \infty$*

Proof. Sketch. The sum above is a lower sum for the following improper integral $\int_0^\infty \exp\{-\delta t^a\}dt$. One way to show that it is finite is to do a transformation of variables $y = t^a$, then bound the obtained integral by an integral of the form $\int_0^\infty y^n \exp\{-\delta y\} dy$, where n is an integer. It follows by induction and integration by parts that this integral is finite. □

CHAPTER 3
ESTIMATION OF DYNAMIC DISCRETE CHOICE MODELS WITH DP SOLUTION APPROXIMATED BY ARTIFICIAL NEURAL NETWORKS

3.1 Introduction

The dynamic discrete choice model (DDCM) is a dynamic program (DP) with discrete controls. Estimation of these models is a growing area in econometrics with a wide range of applications. Labor economists employed DDCMs in modeling job search and occupational choice (Miller (1984), Wolpin (1987), Keane and Wolpin (1997)), retirement decisions (Stock and Wise (1990), Rust and Phelan (1997), French (2005)), fertility (Wolpin (1984), Hotz and Miller (1993)), and crime (Imai and Krishna (2004).) Health economists estimated DDCMs of medical care utilization (Gilleskie (1998)), health and financial decisions of elderly (Davis (1998)), and smoking addiction (Choo (2000).) In industrial organization DDCMs were used for studying optimal investment replacement (Rust (1987), Das (1992), Kennet (1994), Cho (2000).) Pakes (1986) estimated a DDCM of patent renewals. There is a growing interest to DDCMs in marketing literature (Erdem and Keane (1996), Osborne (2006).)

DDCMs are attractive for empirical research since they are grounded in economic theory. However, estimation of these models is very computationally expensive. The DP has to be solved at each iteration of an estimation procedure and the likelihood function of a richly specified DDCM contains high-dimensional integrals.

Chapter 1 shows that Markov chain Monte Carlo (MCMC) methods can handle the high dimensional integration in the likelihood function for DDCMs with serially correlated unobservables. In this chapter, I describe how to apply MCMC for dealing with such desirable features of DDCMs as random coefficients and dependent observations that were avoided in the literature because of the high computational burden. A posterior simulator for dynamic discrete choice models proposed in Chapter 1 produces a large amount of serial correlation in parameter draws. Thus, long simulator runs are required to estimate posterior densities with sufficient precision. The solution of the dynamic program that has to be obtained at each iteration of the estimation procedure constitutes a considerable part of the algorithm's computational burden. Algorithms for solving the DP that use information from the previous MCMC iterations to speed up the DP solution on the current iteration were proposed in Imai et al. (2005) and Chapter 1. However, even if the DP solution on a grid over the state space is available, computing the expected value functions by importance sampling for each observation in the dataset still requires a lot of time. The approach based on artificial neural networks (ANN) proposed here can ameliorate this problem.

The expected value function can be seen as a function of the parameters and the current state. Instead of obtaining the DP solution at each iteration of the estimation procedure one could beforehand approximate it by a function of the parameters and state variables and then use this function in the estimation procedure. Under this approach, there is no need to solve the DP at each iteration of a long posterior simulator run. Moreover, if the approximations can be computed faster than the importance sampling integration of the value functions then there will additional performance gains. The DP solving algorithm presented in Chapter 1 can produce precise solutions to the DP as described in Section 5.1.4. It is not feasible to get a very high precision for millions of parameter draws required to reasonably approximate the posterior distributions. However, it is feasible for several thousand draws from a prior distribution. These precise DP solutions on a randomly generated collection of parameters and states can be used to approximate the expected value function.

Approximating a function of several variables is a formidable task. Kernel smoothing did not perform well in experiments, see Chapter 1, Section 5.1.3. ANNs seem to be a method of choice for that. An intuitive explanation for excellent performance of ANNs in theory and practice might be that the basis functions in the ANN case can be tuned, which provides additional flexibility relative to many other approximation methods, e.g., approximation by polynomials, in which the basis functions are fixed. The theoretical properties of ANNs are very attractive: the number of neural network parameters required to obtain an approximation of functions in certain smooth classes grows only polynomially fast in the function argument dimension (Barron (1994).) This theoretical result suggests that relatively small neural networks might provide sufficient approximation precision and at the same time produce approximations faster than the importance sampling integration algorithm from Chapter 1. A word of caution is in order here. Although in experiments ANNs do perform very well it has not been proven that DDCM's expected value functions have the smoothness properties required for the Barron's results.

An important issue is whether we can use ANN function approximation properties to show that the estimation results, e.g., posterior expectations, that are obtained with approximated DP solutions converge to the true ones as the approximation precision improves. Although there are lot of different results available for the consistency and convergence rates for ANN function approximation, the result we could use to show the consistency of the estimated posterior expectations does not seem to be available in the ready-to-use form. In this chapter, I derive such a result from the contributions of White (1990), Hornik et al. (1989), and Chen (2005).

Section 3.2 of this chapter sets up a DDCM and outlines an MCMC estimation procedure. Section 3.3 introduces ANNs and derives necessary theoretical results. Experiments are conducted on the model from Rust (1987). The model and the corresponding MCMC algorithm are is described in Chapter 1. The ANN approximation quality is evaluated in Section 3.4.1. Section 3.4.2 presents estimation results.

3.2 DDCM and MCMC

A DDCM is a single agent model. Each time period t the agent chooses an alternative d_t from a finite set of available alternatives $D(s_t)$. The per-period utility $u(s_t, d_t; \theta)$ depends on the chosen alternative, current state variables $s_t \in S$, and a vector of parameters $\theta \in \Theta$. The state variables are assumed to evolve according to a controlled first order Markov process with a transition law denoted by $f(s_{t+1}|s_t, d_t; \theta)$ for $t \geq 1$; the distribution of the initial state is denoted by $f(s_1|\theta)$. Time is discounted with a factor β. In the recursive formulation of the problem, the lifetime utility of the agent or the value function is given by the maximum of the alternative-specific value functions:

$$V(s_t;\theta) = \max_{d_t \in D(s_t)} \mathcal{V}(s_t, d_t;\theta) \tag{3.1}$$

$$\mathcal{V}(s_t, d_t;\theta) = u(s_t, d_t;\theta) + \beta E\{V(s_{t+1};\theta)|s_t, d_t;\theta\} \tag{3.2}$$

This formulation embraces a finite horizon case if time t is included in the vector of the state variables.

In estimable DDCMs, some extra assumptions are usually made. First of all, some of the state variables are assumed to be unobservable for econometricians (the agent observes s_t at time t.) Let's denote the unobserved state variables by y_t and the observed ones by x_t. Examples of unobservables include taste idiosyncrasy, ability, and health status. Using the unobserved state variables is a way to incorporate random errors in DDCMs structurally. Some of the state variables could be common to all individuals in a dataset. Let's denote these common states by z_t. We assume that z_t are unobserved (the case of observed z_t would be simpler.) To avoid modeling the interactions between agents it is assumed that the evolution of z_t is not affected by individual states and decisions. Introducing common states z_t is a way to model dependence across observations in the sample. Thus, the state variables are separated into three parts $s_t = (z_t, x_t, y_t)$ and they evolve according to $f(s_{t+1}|s_t, d;\theta) = p(z_{t+1}|z_t;\theta)p(x_{t+1}, y_{t+1}|x_t, y_t, z_t, d;\theta)$. The set of the available alternatives $D(s_t)$ is assumed to depend only on the observed state variables. Hereafter, it will be denoted by D without loss of generality.

There is a consensus in the literature that it is desirable to allow for individual heterogeneity in panel data models. Examples of individual heterogeneity in DDCMs include individual specific time discount rates and individual specific intercepts or coefficients in the per period utility function that would represent taste idiosyncrasies. To allow for that let's assume that the parameter vector θ contains individual specific components θ_1^i and common components θ_2 and the prior $p(\theta_1^i|\theta_2)p(\theta_2)$ is specified. The common parameters θ_2 may include components that define $p(\theta_1|\theta_2)$ and do not affect the DP.

A data set that is usually used for the estimation of a dynamic discrete choice model consists of a panel of I individuals. The observed part of the state and the decisions are known for each individual $i \in \{1, \dots, I\}$ for T periods: $(x, d) = \{x_{t,i}, d_{t,i}; t = 1, \dots, T; i = 1, \dots, I\}$. The likelihood function is given by the integral over the latent variables:

$$p(x, d|\theta_2) = \int p(x, d, y, \theta_1, z|\theta_2) d(y, \theta_1, z) \tag{3.3}$$

where $y = \{y_{t,i}; t = 1, \dots, T; i = 1, \dots, I\}$, $z = \{z_t; t = 1, \dots, T\}$, and $\theta_1 = \{\theta_1^i; i = 1, \dots, I\}$. Because of the high dimensionality of the integral computing the likelihood function is infeasible for richly specified DDCMs.

In a Bayesian framework, the high dimensional integration over the latent variables can be handled by employing MCMC for exploring the joint posterior distribution of the latent variables and parameters. As was shown in Chapter 1, it is convenient to use the differences in alternative specific value functions

$$\Delta\mathcal{V} = \{\Delta\mathcal{V}_{t,d,i} = u(s_{t,i}, d; \theta) + \beta E[V(s_{t+1};\theta)|s_{t,i}, d; \theta)] - E[V(s_{t+1};\theta)|s_{t,i}, \overline{d}; \theta)], \forall i, t, d\} \tag{3.4}$$

as the latent variables in the MCMC algorithm instead of a part of $y_{t,i}$, where $\overline{d}$ is a chosen base alternative. Let's denote the part of $y_{t,i}$ substituted with $\Delta\mathcal{V}_{t,i}$ by $\nu_{t,i}$ and the remaining part by $\epsilon_{t,i}$; thus $y_{t,i} = (\nu_{t,i}, \epsilon_{t,i})$. To save space it is assumed below that $p(\nu_{t,i}|z_t, x_{t,i}, \epsilon_{t,i}, z_{t-1}, x_{t-1,i}, \epsilon_{t-1,i}, d_{t-1,i}; \theta^i) = p(\nu_{t,i}|z_t, x_{t,i}, \epsilon_{t,i}; \theta^i)$. However, this assumption is not necessary.

The joint posterior distribution of the parameters and latent variables will be proportional to the joint distribution of the data, the parameters and the latent variables:

$$p(\theta_1, \theta_2, \Delta\mathcal{V}, \epsilon, z|x, d) \propto p(d, \Delta\mathcal{V}, \theta_1, \theta_2, \epsilon, z, x)$$

which in turn can be decomposed into the product of marginals and conditionals:

$$\begin{aligned} p(d, \Delta\mathcal{V}, \theta_1, \theta_2, \epsilon, z, x) = \prod_{t=1}^{T} \Bigg[\prod_{i=1}^{I} \Bigg(& p(d_{t,i}|\Delta\mathcal{V}_{t,i}) p(\Delta\mathcal{V}_{t,i}|x_{t,i}, \epsilon_{t,i}, z_t; \theta_1^i, \theta_2) \\ \cdot p(x_{t,i}, \epsilon_{t,i}|x_{t-1,i}, \epsilon_{t-1,i}, z_{t-1}, d_{t-1,i}; \theta_1^i, \theta_2) \Bigg) & \cdot p(z_t|z_{t-1}, \theta_2) \Bigg] \cdot \left[\prod_{i=1}^{I} p(\theta_1^i|\theta_2) \right] \cdot p(\theta_2) \end{aligned} \tag{3.5}$$

The Gibbs sampler can be used to simulate a Markov chain which would have the stationary distribution equal to the posterior. The densities of the Gibbs sampler blocks: $p(\theta_1^i|\Delta\mathcal{V}_i, \theta_2, \epsilon_i, z, d_i, x_i)$, $p(\theta_2|\Delta\mathcal{V}, \epsilon, z, d, x)$, $p(\Delta\mathcal{V}_{t,i}|\theta_1^i, \theta_2, \epsilon_{t,i}, z_{t,i}, d_{t,i}, x_{t,i})$, $p(\epsilon_{t,i}|\Delta\mathcal{V}_{t,i}, \theta_1^i, \theta_2, \epsilon_{t-1,i}, \epsilon_{t+1,i}, z_{t,i}, d_{t,i}, x_{t,i})$, and $p(z_t|\Delta\mathcal{V}_t, \theta_1, \theta_2, \epsilon_t, z_{t+1}, z_{t-1}, d_t, x_t)$ are proportional to (3.5). If $p(\Delta\mathcal{V}_{t,i}|\theta, x_{t,i}, \epsilon_{t,i}, z_t)$ can be quickly computed then (3.5) (and, thus, the kernels of the densities of the Gibbs sampler blocks) can be quickly computed as well. Therefore, it is possible to use the Metropolis-within-Gibbs algorithm to simulate the chain.

As evident from (3.4), computing the value of the joint density (3.5) will require computing the differences in expected value functions $F(s, d, \theta) = E[V(s_{t+1}; \theta)|s, d; \theta)] - E[V(s_{t+1}; \theta)|s, \overline{d}; \theta)]$. Let $F(s, \theta) = \{F(s, d, \theta), d \in D\}$ be a vector of the differences in expected value functions corresponding to all available alternatives, the same current state s, and the parameter vector θ. Solving the DP and computing $F(s, \theta_1^i, \theta_2)$ for each observation $i = 1, \ldots, I$ at each MCMC iteration would be infeasible. Instead, one could approximate $F(.)$ beforehand of the estimation procedure by ANNs. The following section gives a relevant background on ANNs.

3.3 Feedforward ANN

3.3.1 Definition of feedforward ANN

It is beyond the scope of this chapter to survey the literature on artificial neural networks and their (potential) applications in economics. For general information, history, and econometric perspective on ANN the reader is referred to the work by Kuan and White (1994). Rust (1996) discusses the application of neural networks to function approximation problems in the context of numerical dynamic programming. Cho and Sargent (1996) consider applications of neural networks in dynamic economics and game theory.

The purpose of this section is to provide information on artificial neural networks relevant to applications in DDCM estimation. The section describes a particular type of artificial neural networks, feedforward networks (FFANN), that are well suited for function approximation problems. Figure 3.1 shows the structure of a multi-layer FFANN that transforms the input vector $x \in R^n$ into the output vector $y \in R^m$.

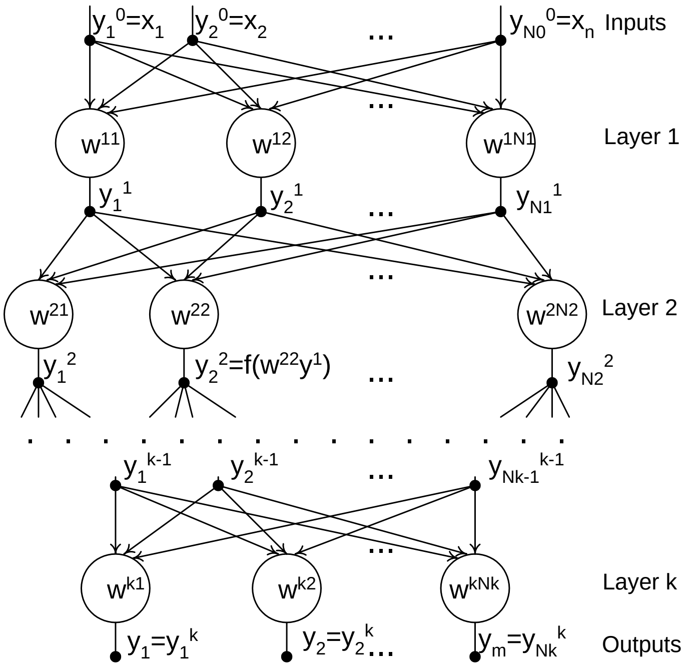

Figure 3.1: Multi-layer feed-forward neural network

The network consists from a number of nodes called neurons. The neurons are grouped into layers. The outputs of the neurons on the layer $i-1$ are used as the inputs for each neuron on the next layer i. The inputs for the first level are the network inputs x. The outputs of the last layer are the network outputs. The neuron j on the level i multiplies the inputs $y^{i-1} = (y_1^{i-1}, \ldots, y_{N_{i-1}}^{i-1})$ by the connection weights $w^{ij} = (w_l^{ij}, \ldots, w_{N_{i-1}}^{ij})^T$ and transforms the sum of the weighted inputs into a scalar output y_j^i:

$$y_j^i = f(\sum_{l=1}^{N_{i-1}} y_l^{i-1} w_l^{ij}) = f(y^{i-1} w^{ij}),\ i = 1, \dots, k,\ j = 1, \dots, N_i, \tag{3.6}$$

where k is the number of layers, N_i is the number of neurons in the layer i, and $f(.)$ is called activation function. The logistic sigmoid $f(z) = 1/(1 + \exp\{-z\})$ is a popular choice for the activation function. The activation functions do not have to be the same for all neurons. The identity function $f(z) = z$ is sometimes used for the neurons on the last (output) layer. It is standard practice to add an extra input equal to 1 to each neuron. This is a way to introduce intercepts (called biases in the ANN literature) in addition to the coefficients (weights) in (3.6).

An explicit formula for computing the output for a two-layer network with one-dimensional output might be helpful for understanding the general case:

$$y = y_1^2 = \hat{F}(x; w) = f\left(\sum_{l=1}^{N_1} w_l^{21} f\left(\sum_{j=1}^{m} w_j^{1l} x_j\right)\right)$$

Let $F(x)$ denote the function we wish to approximate by a neural network. The connection weights w are adjusted so that the neural network $\hat{F}(x; w)$ fits $F(x)$. The process of adjusting the weights is called learning or training. Training is performed on a dataset $\{x^j, y^j, j = 1, J\}$, where y^j is equal to $F(x^j)$ perhaps with some noise. The method of least squares is the most common way to adjust the weights:

$$\min_w S(w) = \min_w \sum_j [y^j - \hat{F}(x^j; w)]^2 = \min_w \sum_j e^j(w)^2$$

If the activation function is differentiable then gradient methods can be used to perform the minimization. In the ANN literature the gradient descent algorithm is referred to as the back error propagation. The derivatives are computed by the chain rule: $S'(w) = 2e'(w)^T e(w)$. More sophisticated optimization methods, such as conjugate gradient algorithms and quasi-Newton methods, can be used to increase the training speed. According to the Matlab manual, the Levenberg-Marquardt algorithm is the fastest method for training moderate-sized FFANN (up to several hundred weights). My experiments with a few other algorithms (not implemented in Matlab) confirm this claim. The Levenberg-Marquardt algorithm iteratively updates the weights according to

$$w^{q+1} = w^q - [e'(w^q)^T e'(w^q) + \mu I]^{-1} e'(w^q)^T e(w^q)$$

If the scalar μ is small then the method works as a quasi-Newton method with the Hessian approximated by $e'(w)^T e'(w)$ (computing actual Hessian would be very time consuming.) If μ is large then the method works as the gradient descent algorithm with a small step. The Newton method performs considerably better than the gradient descent algorithm near the optimum. Thus, after successful iterations (the ones that decrease $S(w)$) μ is decreased; otherwise it is increased. For large FFANNs conjugate gradient methods might perform better than the Levenberg-Marquardt algorithm.

The training methods mentioned above are local optimization methods. There is no guarantee that they will find the global minimum. The theoretical results on the consistency of ANN approximations do require finding the global minimum. Therefore, running training algorithms for several initial values is advisable. In experiments on approximating the expected value function by a FFANN the Matlab implementation of the Levenberg-Marquardt algorithm performed very well. No cases of getting stuck in a very bad local minimum were detected.

3.3.2 Consistency of FFANN approximations

The consistency and convergence rates for ANN function approximation were examined by a number of authors. However, the result we would need for approximation of the DDCM solutions does not seem to be available in the literature in the ready-to-use form. In this section we deduce the necessary result building on the existing theory of ANN approximation.

In Section 1.4 I show that the uniform (in sup norm) convergence in probability of the expected value function approximation would imply the consistency of the approximated posterior expectations (see Theorem 1.3 and its proof.) It was also shown that the expected value function is continuous in the state variables and parameters under suitable continuity and compactness assumptions on the primitives of DDCMs (see Section 1.3.3.) At the same time the differentiability of the expected value function with respect to the state variables and parameters does not seem to have been established for a general DDCM. Therefore, it would be desirable to have the consistency of the ANN approximations in sup norm for continuous functions on compact spaces. However, the consistency results in the ANN literature are available for convergence in L_p norm and/or for smooth functions (White (1990), Barron (1994), and Chen and White (1999).)

Following White (1990) let's consider a FFANN sieve estimator. For a survey of sieve estimation see Chen (2005). Let $\mathcal{F}$ denote a set of continuous functions on a compact space $\mathcal{X}$ with sup norm $||.||$. Let (X, Y) be random variables defined on a complete probability space. Assume that X has a positive density on $\mathcal{X}$, $E(Y|X = x) = F(x)$, and $F \in \mathcal{F}$. Let $\{x^j, y^j\}_{j=1}^n$ denote an i.i.d. sample of (X, Y). The consistency results are proven below for randomly generated $\{x^j\}_{j=1}^n$. Experiments show that using low discrepancy sequences on $\mathcal{X}$ instead of randomly generated ones does not improve the approximation quality.

A FFANN with one hidden layer consisting of q neurons and a linear activation function for the neuron on the output layer is described by

$$\hat{F}(x; w, q) = w_0^{21} + \sum_{k=1}^{q} w_k^{21} f\left(w_0^{1k} + \sum_j w_j^{1k} x_j\right)$$

Let

$$T(q, \Delta) = \{\hat{F}(.; w, q) : \sum_{k=0}^{q} |w_k^{21}| < \Delta \text{ and } \sum_{k=1}^{q} \sum_j |w_j^{1k}| < q\Delta\}$$

be a set of FFANNs with q neurons on the hidden layer and the weights satisfying a restriction on their sum norm. For specified sequences $\{q_n\}$ and $\{\Delta_n\}$, $T(q_n, \Delta_n)$ is called a sieve. The sieve estimator $\hat{F}_n(.)$ is defined as the solution to the least squares problem:

$$\min_{\hat{F} \in T(q_n, \Delta_n)} \frac{1}{n} \sum_{j=1}^{n} [y^j - \hat{F}(x^j)]^2 \tag{3.7}$$

The parameters q_n and Δ_n determine the flexibility of approximating functions $\hat{F} \in T(q_n, \Delta_n)$. As they increase to infinity the set $T(q_n, \Delta_n)$ will become dense in $\mathcal{F}$. The flexibility of approximating functions should depend on the number of observations n in such a way that overfitting and underfitting are avoided at the same time. Specific restrictions on q_n and Δ_n that achieve this are given below. Also, introducing a finite bound on the weights Δ_n makes $T(q_n, \Delta_n)$ compact. White (1990) proves a version of the following theorem for L_p norm. Here, I present a proof for sup norm.

Theorem 3.1. *Assume that the activation function f is Lipschitz continuous and it is a squashing function (f is non-decreasing, $\lim_{x\to-\infty} f(x) = 0$, $\lim_{x\to+\infty} f(x) = 1$). Also assume that $q_n, \Delta_n \nearrow \infty$, $\Delta_n = o(n^{1/4})$, and $\Delta_n^4 q_n \log(\Delta_n q_n) = o(n)$. Under these conditions there exists a measurable sieve estimator $\hat{F}_n(.)$ defined by (3.7) and for any $\epsilon > 0$*

$$\lim_{n\to\infty} P(||F - \hat{F}_n|| > \epsilon) = 0$$

Proof. Theorem 3.1 in Chen (2005) specifies five conditions under which an abstract extremum sieve estimator will be consistent. Let's show that these five conditions are satisfied.

Condition 3.1: $E[Y - g(X)]^2$ is uniquely minimized over $g \in \mathcal{F}$ at F and $E[Y - F(X)]^2 < \infty$. This identification condition is satisfied in our case because $F(x) \equiv E(Y|X = x)$ is a minimizer and it is unique since functions in $\mathcal{F}$ are continuous and the density of X is positive on $\mathcal{X}$.

Condition 3.2: the sequence of sieves is increasing ($T(q_n, \Delta_n) \subset T(q_{n+1}, \Delta_{n+1})$) and $\cup_{n=1}^{\infty} T(q_n, \Delta_n)$ is dense in $\mathcal{F}$. The denseness of the set of one hidden layer FFANNs with unconstrained weights $\cup_{n=1}^{\infty} T(n, \infty)$ in the set of continuous functions on compacts is proven in Hornik et al. (1989), Theorem 2.4. The condition is satisfied since $\cup_{n=1}^{\infty} T(q_n, \Delta_n) = \cup_{n=1}^{\infty} T(q_n, \infty)$ for $q_n, \Delta_n \to \infty$.

Condition 3.3: $-E[Y - g(X)]^2$ is upper semicontinuous in g w.r.t $||.||$. The condition is trivially satisfied since $E[Y - g(X)]^2$ is continuous.

Condition 3.4: $T(q_n, \Delta_n)$ is compact under $||.||$. Since any element in $T(q_n, \Delta_n)$ is defined by a vector of weights belonging to a compact set and the activation function f is continuous, any sequence in $T(q_n, \Delta_n)$ will have a convergent subsequence with the limit in $T(q_n, \Delta_n)$, thus $T(q_n, \Delta_n)$ is compact.

Condition 3.5: (uniform convergence over sieves) $\text{plim}_{n\to\infty} \sup_{g\in T(q_n,\Delta_n)} |\frac{1}{n}\sum_{j=1}^{n}[y^j - g(x^j)]^2 - E[Y - g(X)]^2| = 0$. This condition is proven in White (1990), pp.543-544. That is where the Lipschitz continuity of f and the specific conditions on q_n and Δ_n are used. □

3.4 Experiments

Experiments in this section demonstrate how well FFANNs can approximate expected value functions and what the performance gains of using FFANNs in MCMC estimation of DDCMs can be. The Rust (1987) model of optimal bus engine replacement is used for experiments.

3.4.1 Evaluating approximation quality

The posterior simulator for Rust's model described in Section 1.5.2.3 requires computing the differences in expected value functions $F(x_{t,i}, \epsilon^m_{t,i}, \theta^m)$ defined in (1.5.2.3) for each parameter draw θ^m and each observation (i, t) in the sample. This section shows how FFANNs can be used for approximating $F(.)$. A FFANN is trained and validated beforehand of the estimation procedure on a set of inputs and outputs. The inputs include parameters and states: $\{x^{ji}, \epsilon^j, \alpha^j_1, \alpha^j_2, \rho^j, \eta^j_1, \eta^j_2, \eta^j_3; i = 1, \ldots, 90; j = 1, \ldots, 2200\}$. In the experiments described below, the inputs are generated from the following distributions: $\alpha_1^j \sim U[-.006, 0]$, $\alpha_2^j \sim U[-25, -5]$, $\rho^j \sim U[0, .99]$, $\epsilon^j \sim U[-3.8, 3.8]$, $\eta \sim \text{Dirichlet}(34, 64, 2)$, and $x^{ji} = i$. For large and complicated models, more involved inputs could be used, e.g., some functions of states and parameters and the value functions for the DP in which shocks are replaced by zeros. The latter could also be subtracted from the difference in expected value function to obtain a better behaved output. Since the exact DP solution is not available for the model with serially correlated unobservables the following approximations are used as etalon outputs. For each θ^j the DP is solved on $\tilde{N} = 100$ different random grids. Each grid consists of $\hat{N} = 100$ randomly generated points on the space for ν and ϵ. The differences in the expected value functions is computed for each random grid. The average over the grids, denoted by $F^{ji}_{\tilde{N},\hat{N}}$, is used as the output. This procedure efficiently produces good approximations of $F(.)$. Let's illustrate this for the model with extreme value iid unobservables ϵ_t and ν_t.

Under the extreme value iid assumption, the integration over the unobservables can be performed analytically (see Rust (1994),) the exact DP solution can be quickly computed, and solutions on the random grids can be compared with the exact solution. Figure 3.2 shows densities of the difference between the exact solution and the solution on random grids $[F(x^{ji}, \theta^j) - F^{ji}_{\tilde{N},\hat{N}}]$ (for iid unobservables $F(.)$ does not depend on ϵ.) The densities were estimated by kernel smoothing.

The precision of the DP solution obtained by averaging the results over 100 random grids with 100 points in each grid is about the same as for the solution obtained on one random grid with 10000 points. However, the former algorithm works about 100 times faster since the number of operations performed for one Bellman equation iteration is roughly proportional to the square of the number of points in the grid. See Section 1.5.1.4 of Chapter 1 for further discussion of this issue. The maximal approximation error for $F^{ji}_{100,100}$ does not exceed 3% of the standard deviation of the iid shocks in the model.

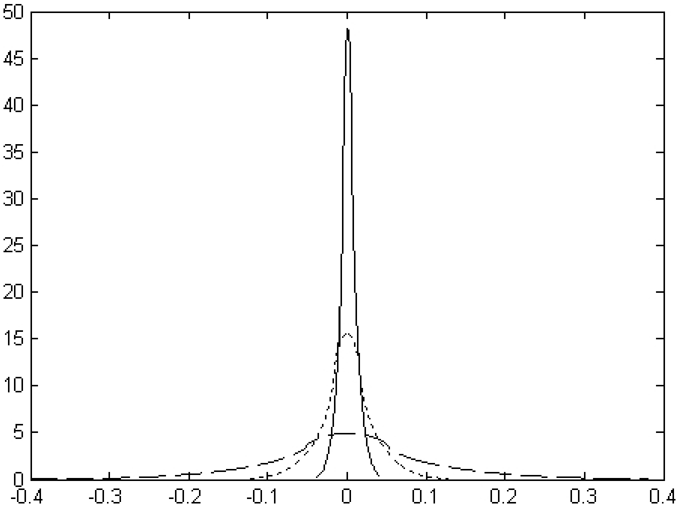

Figure 3.2: Densities of the difference between the exact solution and the solution on random grids. The model with extreme value iid unobservables. The dashed line - the density of $[F(x^{ji},\theta^j) - F^{ji}_{1,100}]$, the dotted line - the density of $[F(x^{ji},\theta^j) - F^{ji}_{10,100}]$, and the solid line - the density of $[F(x^{ji},\theta^j) - F^{ji}_{100,100}]$.

A three layer FFANN was trained in Matlab by the Levenberg-Marquardt algorithm for the data from the model with normal serially correlated unobservables. The network layers contained 8, 10, and 1 neurons correspondingly (it will be referred below as the 8-10-1 FFANN.) First 1500 points in the data were used for training, the remaining data were used for validation. Figure 3.3 shows the distributions of the residuals scaled by the standard deviation of the iid shocks in the model $e^{ji} = (F^{ij}_{100,100} - \hat{F}(x^{ji}, \epsilon^{j}, \theta^{j}; w))h_{\nu}^{0.5}$ for the training and validation parts of the data. Scaling is performed to facilitate the comparison of the approximation error and the magnitude of the random shocks in the model. As can be seen from the figure, the approximation quality for the validation part of the data is the same as for the training part. This suggest that no overfitting occurred.

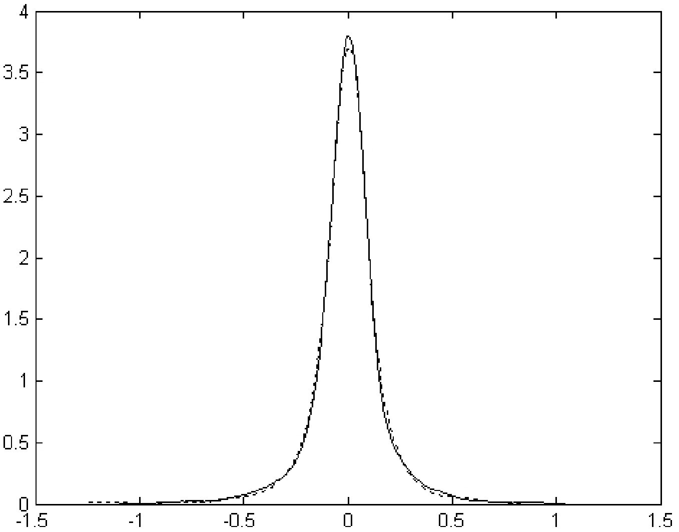

Figure 3.3: Densities of residuals e^{ji} (the dotted line is for the validation part of the sample.

In addition to the randomly generated validation part of the sample, $F^{ji}_{100,100}$ were computed for inputs with one component changing over a relevant range and the other components fixed at $(x, \epsilon, \alpha_1, \alpha_2, \rho, \eta_1, \eta_2, \eta_3) = (55, 0, -.003, -10, .5, .34, .64, .02)$. Figure 3.4 shows these $F^{ji}_{100,100}$ and the corresponding fitted values.

As Figure 3.4 and Figure 3.2 demonstrate, the values $F^{ji}_{100,100}$ used for neural network training are noisy approximations to the true differences in expected value functions. It is not surprising since they were obtained by solving the dynamic program on random grids. The exact difference in the expected value functions should be a smooth function. Since the fitted function $\hat{F}(.;w)$ tends to smooth out the noise, the actual error of the neural network approximation might be smaller on average then the residuals described by Figure 3.3.

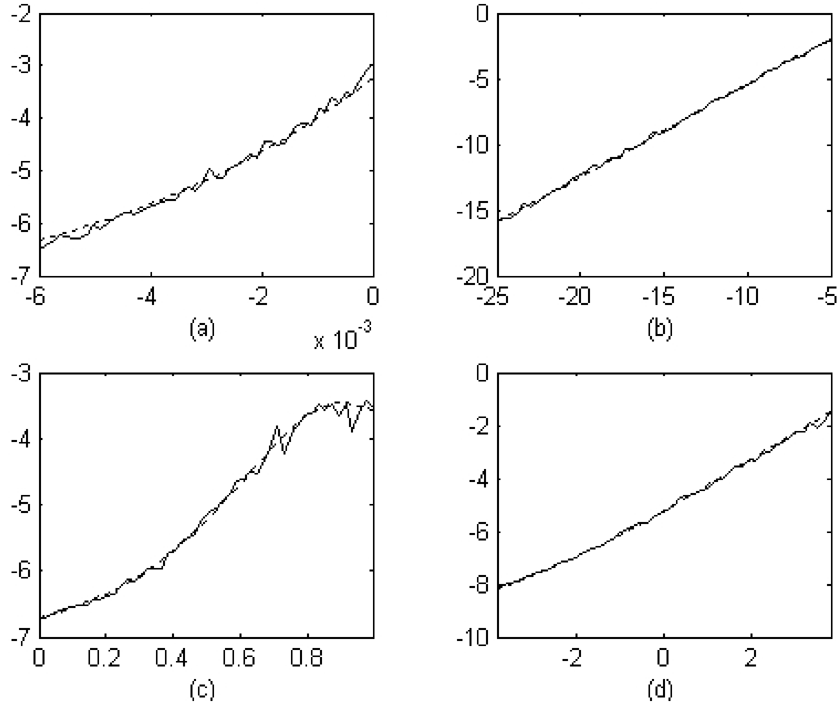

Figure 3.4: Fitted values $\hat{F}(x^{ji}, \epsilon^j, \theta^j; w)$ (the dotted lines) and $F^{ji}_{100,100}$ (the solid lines) as functions of one input component. The model with serially correlated unobservables. The horizontal axes: (a) α_1, (b) α_2, (c) ρ, and (d) ϵ.

The quality of FFANN approximations can be further explored for the model with extreme value iid unobservables since the actual approximation error can be computed in this case. Figure 3.5 compares the densities of the exact approximation error for FFANNs and DP solutions on random grids.

In this particular example, the noise in the training data does not affect the FFANN approximation quality as evidenced by similar results for FFANNs trained on exact and noisy data.

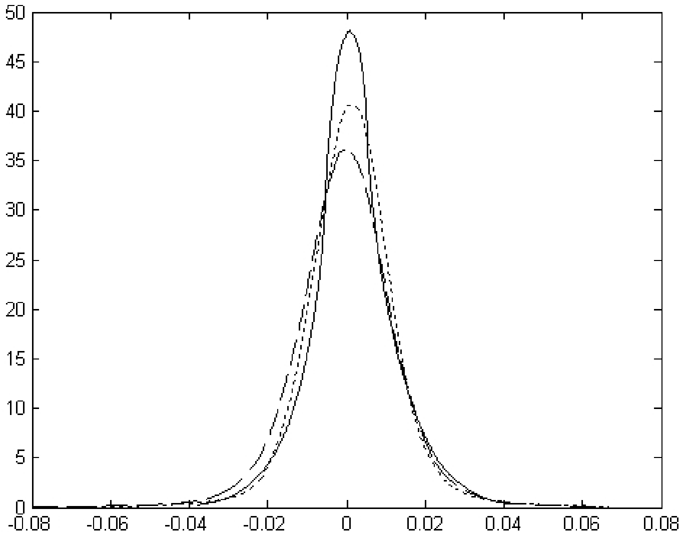

Figure 3.5: Densities of the approximation error for FFANNs and DP solutions on random grids. The model with extreme value iid unobservables. The dashed line - for a FFANN trained on exact data, the dotted line - for a FFANN trained on $F^{ji}_{100,100}$, the solid line - for $F^{ji}_{100,100}$.

Figure 3.6 compares the FFANN and random grid DP solution approximations for the model with serially correlated unobservables. Unfortunately, the exact DP solution and, thus, the exact approximation errors are not available for this model. Therefore, the figure shows the densities of the scaled residuals $e^{ji} = (F^{ji}_{100,100} - \hat{F}(x^{ji}, \epsilon^j, \theta^j; w))h_\nu^{0.5}$ for an 8-10-1 FFANN and the scaled differences $[F^{ji}_{100,100} - F^{ji}_{10,100}]h_\nu^{0.5}$.

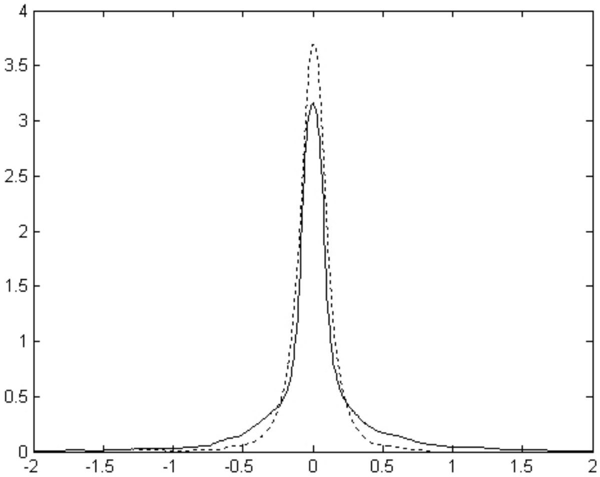

Figure 3.6: Densities of e from neural network (the dotted line) and from DP solution on a random grid (the solid line.) The model with serially correlated unobservables.

As can be seen from the figure, $\hat{F}(x^{ji}, \epsilon^j, \theta^j)$ and $F^{ji}_{10,100}$ provide comparable precision in approximating $F^{ji}_{100,100}$. Since the variance of $F(x^{ji}, \epsilon^j, \theta^j) - F^{ji}_{10,100}$ is considerably larger than the variance of $F(\theta^j, x^i) - F^{ji}_{100,100}$ (see Figure 3.2,) we can argue that $\hat{F}(x^{ji}, \epsilon^j, \theta^j)$ and $F^{ji}_{10,100}$ provide comparable precision in approximating $F(x^{ji}, \epsilon^j, \theta^j)$. Figure 3.5 shows that for the model with extreme value iid unobservables, the approximation precision of an 8-10-1 FFANN is comparable to the precision of $F^{ji}_{100,100}$, which is better than $F^{ji}_{10,100}$ we obtain for the model with serially correlated unobservables. This can be explained by the fact that the dimension of the input vector is smaller for the model with extreme value iid unobservables. Increasing the number of neurons and/or layers in a FFANN would improve the precision, e.g., adding another layer with 10 neurons decreased the approximation error by two times on average.

The posterior simulator for the model with serially correlated unobservables that uses the 8-10-1 FFANN works 4-5 times faster than the posterior simulator that uses DP solutions on one random grid with 100 points. Averaging DP solutions over 10 random grids (which would provide precision comparable to the 8-10-1 FFANN as we argued above) will increase the execution time by 10 times. Thus, for this particular example, the performance gains from using FFANNs in the posterior simulator could amount to about 40-50 times. In experiments, the MCMC estimation algorithm required at least 1 million draws to converge. The time required for preparing FFANN training data is less than 2% of the time required for solving the DP on 10 random grids for 1 million parameter draws. The time required for training an 8-10-1 FFANN is also of similar magnitude. Thus the overall time saving from using FFANN approximations in DDCM estimation seems considerable. This claim is confirmed by the performance comparison for the model with extreme value iid unobservables, which is presented with the estimation results in the next section.

3.4.2 Estimation results

This section presents estimation results for the model with extreme value iid unobservables. The advantage of using this model relative to the model with serially correlated unobservables is that we can characterize the true posterior distributions of parameters with a high precision. The integration over the unobservables in solving the DP and in the likelihood function can be performed analytically. Thus it would be easier to evaluate the quality of the posterior simulator that uses FFANNs. The posterior simulator for this model also uses the Metropolis-Hastings algorithm since the logit-like choice probabilities comprising the likelihood function contain the expected value functions that do not have an analytical representation.

Figure 3.7 shows the estimated posterior densities for the simulators that use the exact DP solutions and the 8-10-1 FFANN approximations. The experiments use an artificial dataset consisting of observations on $I = 70$ buses (about 4000 mileage/decision points.) The posterior densities were estimated by kernel smoothing over several simulator runs. The length of the runs was 3 million draws. The simulator using the 8-10-1 FFANN takes about 1.2 second to produce 1000 draws from the posterior on a 2002 vintage PC. The simulator that uses the exact DP solutions works 10 times slower. The estimated densities from both simulators are very similar.

The model with extreme value iid unobservables could also be estimated by the algorithm that performs integration numerically as in the case of the model with serially correlated unobservables. The Gibbs sampler for this algorithm is the same as the one for the Gaussian unobservables described in Section 1.5.2.3; except here the Gaussian probability densities are replaced by the densities for the extreme value distribution. Figure 3.8 compares the estimation results for the exact posterior simulator and the simulator that integrates unobservables numerically and solves the DP on a random grid with 100 points. The posteriors for η are not shown in the figure since they are identical for all simulators. For some random grids the estimated density can be far off as the figure demonstrates. The simulator that integrates unobservables numerically and solves the DP on a random grid with 100 points produce 1000 parameter draws in 102 seconds. The same task takes 14 seconds for the same simulator if it uses an 8-10-1 FFANN instead of DP solutions on a random grid. As Figure 3.5 from the previous section demonstrates, the approximation precision of an 8-10-1 FFANN is comparable to the average over 100 DP solutions on random grids

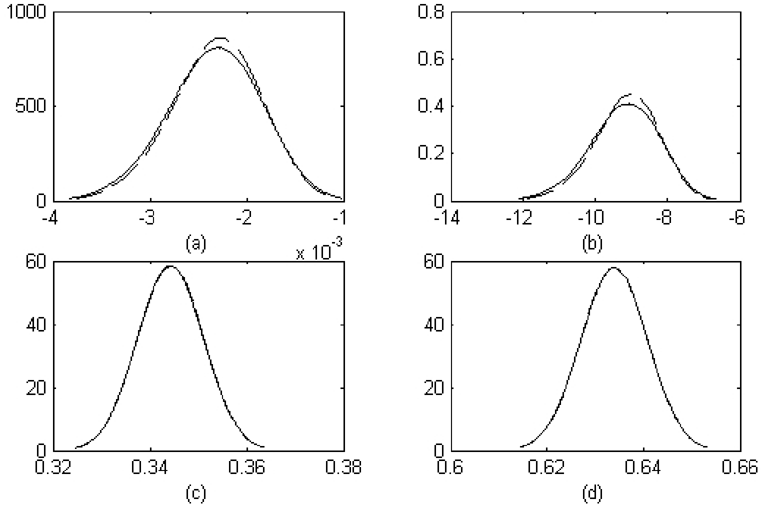

Figure 3.7: Estimated posterior densities: (a) α_1, (b) α_2, (c) η_1, (d) η_2. The solid lines for the algorithm using the exact DP solutions, the dashed for the algorithm using the FFANN.

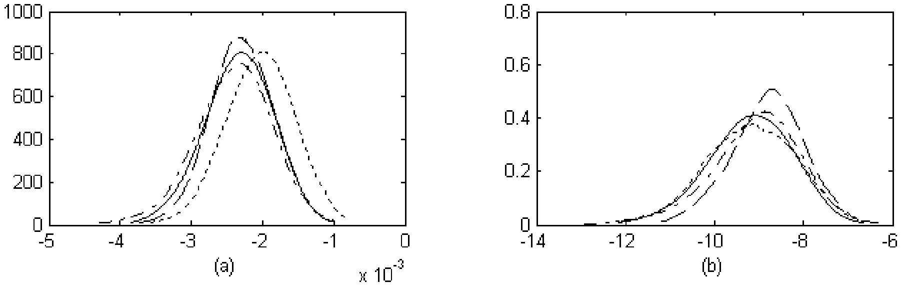

Figure 3.8: Estimated posterior densities: (a) α_1, (b) α_2. The solid lines - the simulator using the exact DP solutions, the other lines - the simulators using DP solutions on different random grids.

with 100 points. If the simulator uses averages of the DP solutions over several random grids the computing time will increase proportionally to the number of the random grids used. Thus, the performance gains from using FFANNs in this example can reach $729 = (102 \cdot 100/14)$ times. Similar estimation experiments in Section 1.5.2.7 of Chapter 1 suggest that averaging the posterior distributions estimated with the DP solved on different random grids improves estimation precision. Nevertheless, this posterior averaging strategy does not have a rigorous theoretical justification and it is still considerably outperformed by a simulator using FFANNs.

In summary, the experiments suggest that application of ANNs in the MCMC estimation of DDCMs is indeed a promising approach. It is fast and precise. It can also provide a feasible way to estimate rich DDCMs with different forms of individual heterogeneity, e.g., serially correlated unobserved state variables and individual specific parameters.

REFERENCES

Barron, A. Approximation and estimation bounds for artificial neural networks. *Machine Learning*, 14, 1994.

Bierens, H. *Topics in Advanced Econometrics.* Cambridge University Press, 1994.

Chen, X. Large sample sieve estimation of semi-nonparametric models, 2005.

Chen, X. and White, H. Improved rates and asymptotic normality for nonparametric neural network estimators. *IEEE Transactions on Information Theory*, (45), 1999.

Cho, I.-K. and Sargent, T. Neural networks for encoding and adapting in dynamic economies. In H. Amman, D. K. and Rust, J., editors, *Handbook of Computational Economics.* North Holland, 1996.

Cho, S.-J. An empirical model of mainframe computer replacement, 2000.

Choo, E. Rational addiction and rational cessation: A dynamic structural model of cigarette consumption, 2000.

Das, M. A micro-econometric model of capital utilization and retirement: The case of the cement industry. *Review of Economic Studies*, 59:277–297, 1992.

Davis, M. A. *The Health and Financial Decisions of the Elderly.* Ph.D. thesis, University of Pennsylvania, 1998.

Eckstein, Z. and Wolpin, K. The specification and estimation of dynamic stochastic discrete choice models: A survey. *Journal of Human Resources*, 24(58):562–598, 1989.

Erdem, T. and Keane, M. Decision-making under uncertainty: Capturing dynamic brand choice processes in turbulent consumer goods markets. *Marketing Science*, 15(1):1–20, 1996.

French, E. The effects of health, wealth, and wages on labour supply and retirement behaviour. *Review of Economic Studies*, 72:395–427, 2005.

Geweke, J. Efficient simulation from the multivariate normal and student-t distributions subject to linear constraints and the evaluation of constraint probabilities, 1991.

Geweke, J. Getting it right: Joint distribution tests of posterior simulators. *Journal of the American Statsitical Association*, 99:799–804, 2004.

Geweke, J. *Contemporary Bayesian Econometrics and Statistics.* Wiley-Interscience, 2005.

Geweke, J., Keane, M., and Runkle, D. Alternative computational approaches to inference in the multinomial probit model. *The review of economics and statistics*, 76(4):609–632, 1994.

Gilleskie, D. A dynamic stochastic model of medical care use and work absence. *Econometrica*, 6(1):1–45, 1998.

Hoeffding, W. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, (58):13–30, 1963.

Hornik, K., Stinchcombe, M., and White, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, (2):359–366, 1989.

Hotz, J. and Miller, R. Conditional choice probabilities and the estimation of dynamic models. *Re-view of Economic Studies*, 60(3):497–530, 1993.

Hsu, P. L. and Robbins, H. Complete convergence and the law of large numbers. In *Proc. Natl. Acad. Sci., February, 33(2)*, pages 25–31. USA, 1947.

Imai, S., Jain, N., and Ching, A. Bayesian estimation of dynamic discrete choice models, 2005.

Imai, S. and Krishna, K. Employment, deterrence, and crime in a dynamic model. *International Economic Review*, 45(3):845–872, 2004.

Keane, M. and Wolpin, K. The solution and estimation of discrete choice dynamic programming models by simulation and interpolation: Monte carlo evidence. *Review of Economics and Statistics*, 76(4):648–672, 1994.

Keane, M. and Wolpin, K. The career decisions of young men. *Journal of Political Economy*, 1997.

Kennet, M. A structural model of aircraft engine maintenance. *Journal of Applied Econometrics*, 9:351–368, 1994.

Kuan, C.-M. and White, H. Artificial neural networks: An econometric perspective. *Econometric Reviews*, (13):1–92, 1994.

McCulloch, R. and Rossi, P. An exact likelihood analysis of the multinomial probit model. *Journal of Econometrics*, 64(1-2):207–240, 1994.

Miller, R. Job matching and occupational choice. *Journal of Political Economy*, (92):1086–1120, 1984.

Osborne, M. Consumer learning, habit formation, and heterogeneity: A structural examination, 2006.

Pakes, A. Patents as options: Some estimates of the value of holding european patent stocks. *Econometrica*, 54(4):755–84, 1986.

Rust, J. Optimal replacement of gmc bus engines: an empirical model of harold zurcher. *Econometrica*, 55(5):999–1033, 1987.

Rust, J. Structural estimation of markov decision processes. In Engle, R. and McFadden, D., editors, *Handbook of Econometrics*. North Holland, 1994.

Rust, J. Numerical dynamic programming in economics. In H. Amman, D. K. and Rust, J., editors, *Handbook of Computational Economics*. North Holland, Available online at http://gemini.econ.umd.edu/jrust/sdp/ndp.pdf, 1996.

Rust, J. Using randomization to break the curse of dimensionality. *Econometrica*, 65(3):487–516, 1997.

Rust, J. and Phelan, C. How social security and medicare affect retirement behavior in a world of incomplete markets. *Econometrica*, 65(4):781–831, 1997.

Scott, D. *Multivariate Density Estimation*. Wiley-Interscience, 1992.

Stock, J. and Wise, D. The pension inducement to retire: An option value analysis. *NBER Working Papers*, (2660), 1990.

Stokey, N. and Lucas, R. *Recursive Methods in Economic Dynamics*. Harvard University Press, 1989. With Edward Prescott.

Tierney, L. Markov chains for exploring posterior distributions. *The Annals of Statistics, Vol. 22, No. 4, 1758-1762*, 22(4):1758–1762, 1994.

White, H. Connectionist nonparametric regression: Multilayer feedforward networks can learn arbitrary mappings. *Neural Networks*, (3):535–549, 1990.

Wolpin, K. An estimable dynamic stochastic model of fertility and child mortality. *Journal of Political Economy*, 1984.

Wolpin, K. Estimating a structural search model: The transition from school to work. *Econometrica*, 1987.