Dynamic models of continuous and discrete outcomes; methods and applications

by

Jelmer Yeb Ypma

Thesis submitted to the Faculty of Social and Historical Sciences, University College London, for the degree of

Doctor of Philosophy

Department of Economics – University College London

November 2012

I, Jelmer Yeb Ypma, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Jelmer Yeb Ypma

Table of Contents

List of Tables

List of Figures

Acknowledgements

I would like to express my great appreciation to my advisors Lars Nesheim and Costas Meghir. Costas' enthousiasm has been influential in setting me on the path towards the field of labour economics. Lars has been extremely helpful throughout the process. His continued guidance, encouragement, and valuable insights, from the first days of the project up until the submission of this thesis, are very much appreciated.

During the research for this thesis, I have been a research scholar at the Centre for Microdata Methods and Practice (Cemmap). This has been an excellent research environment and I would like to thank my colleagues and fellow PhD students there for enhancing the PhD experience. I gratefully acknowledge financial support from the Economic and Social Research Council for the ESRC Centre for Microdata Methods and Practice (grant reference RES-589-28-0001 and ES/F015879/1).

Abstract

This thesis contains three chapters on dynamic models with discrete and continuous outcomes. In the first chapter, I focus on indirect inference estimation. Indirect inference is used to estimate parameters in models where evaluation of the objective function directly is complicated or infeasible. Indirect inference is typically formulated as an optimization problem nesting one or more other optimization problems. In some cases the solution to the inner optimization problems can be obtained in one step, but when such a solution is not available, indirect inference estimation is computationally demanding. I show how constrained optimization methods can be used to replace the nesting of optimization problems and I provide Monte Carlo evidence showing when this approach is beneficial.

The second chapter uses panel data from the United Kingdom to estimate a model of wage dynamics with labour participation where the variance in wages is decomposed in a permanent and a transitory component. Most studies that estimate similar models ignore non-participation; individuals without a wage are simply removed from the analysis. This leads to biased estimates of the parameters if working individuals are different in their unobservable characteristics compared to people that do not work. I use a dynamic selection model to include a discrete labour participation choice in a simple model of wage dynamics and compare the results to a version of the model that does not include labour participation.

In the third chapter, I show how some of the assumptions on the dynamics of the unobservables in the second chapter can be relaxed. High dimensional integrals have to be approximated to estimate the less restrictive models. I use sparse grids and simulation methods to approximate these integrals and compare their performance on simulated data.

1

Indirect inference estimation using MPEC

1.1 Introduction

Indirect inference (introduced by Smith, 1993; Gouriéroux, Monfort, & Renault, 1993) is an estimation method that was developed to be used in problems where evaluation of the objective function directly is difficult or infeasible, for example because there is no closed-form analytic expression of the likelihood. For instance, Magnac, Robin, and Visser (1995) use indirect inference to analyse censored duration data for which an explicit expression of the likelihood is not available. Gouriéroux, Phillips, and Yu (2010) apply indirect inference as a bias correction mechanism in panel data models, motivated by the fact that bias-correction formulas are generally not available for higher order auto-regressive processes.

The method works by choosing a (misspecified) auxiliary model that captures the features of the model of interest, but is easy to estimate. There is a one-to-one link between the estimated parameters from the auxiliary model and the parameters of the

model of interest, referred to as the binding function. The aim is to find values for the parameters of interest, that minimize an objective function measuring the distance between the auxiliary parameters obtained from the observed data and the theoretical values of the auxiliary parameters implied by the binding function. There are examples where the binding function can be derived analytically (see for instance example 1 in Smith (2008) or the examples in Jiang and Turnbull (2004)), but in many cases simulation is used to approximate the binding function.

One downside of indirect inference is that despite the simplification it brings, the method remains computationally intensive if we use simulation. The structural parameters minimizing the objective function are obtained using an iterative procedure. In an inner loop, the likelihood function of the auxiliary model is maximized, to get auxiliary parameter estimates for all simulations. In an outer loop, an algorithm searches the space spanning the structural parameters to minimize the objective function. This means that for every iteration of the outer loop, the auxiliary model needs to be estimated for all simulations. Adding to that, the gradient with respect to the structural parameters is commonly calculated using finite differences. This means that the auxiliary parameters have to be estimated repeatedly for each element of the gradient. One recommendation usually made in the literature is therefore to choose auxiliary models that are easy to calculate. Preferably a closed-form expression is available for the auxiliary estimator, in order for the inner loop to be evaluated as quickly as possible.

For auxiliary models that are computationally more intensive, a version of indirect inference aiming to solve this problem, has been proposed, where the objective function is based on the score of the auxiliary likelihood (Gallant & Tauchen, 1996). This method is also referred to as the efficient method of moments (EMM). By definition, the score of the likelihood of the observed data evaluated at the parameter estimates maximizing the likelihood is zero. The objective function of EMM tries to get the score of the likelihood evaluated using the simulated data as close as possible to zero. The benefit of this approach is that the likelihood of the auxiliary model is only maximized once, for the observed data. Then in every iteration, instead of maximizing the likelihood for the simulated data to get auxiliary parameter estimates, the score at the auxiliary parameters of the observed data

is evaluated using simulated datasets. This means that no auxiliary models have to be estimated in an inner loop, which saves computing time. One problem is that this method has been shown not to work well in small samples for some models (e.g. for models with very high persistence, Duffee & Stanton, 2008).

In this paper I propose a different method to decrease the computing time of indirect inference estimation, by formulating the optimization problem as a mathematical program with equilibrium constraints (MPEC, Su & Judd, 2010). In their paper, Su and Judd (2010) show how to solve a dynamic programming problem using MPEC. The economic model behind the dynamic programming problem implies conditions that must hold assuming the agent is acting optimally. These economic optimality conditions are introduced as constraints in the maximum likelihood optimization problem. The benefit of this approach is that there is no inner loop with an optimization problem to solve. A similar approach is taken by Dubé, Fox, and Su (2009) to replace the nested-fixed point algorithm in BLP demand estimation by one optimization procedure.

For indirect inference estimation, the inner loop that is used to approximate the binding function using simulations, can be replaced by adding constraints and parameters to the original problem. Instead of estimating the parameters of the auxiliary models in an inner loop, forcing the auxiliary parameters to equal the solution to the auxiliary estimation problem in every step, I add variables and constraints to the original problem, ensuring that in the optimal solution, i.e. when the optimization problem has converged, the auxiliary parameters are the solution to estimation of the auxiliary model. This re-formulation introduces many nuisance parameters and additional constraints to the optimization problem. Using Monte Carlo simulations I show that it's feasible to solve these optimization problems using a freely available non-linear constrained optimizer. The benefits of this procedure are that we can use analytic derivatives in the optimization procedure, and that we don't have to limit ourselves to simple auxiliary models that can be estimated in one step.

In addition, simulation methods lead to problems with non-smoothness if the model contains discrete outcomes. In some cases the binding function can be approximated without simulation. I give an example of a binary time series model, where we observe discrete

outcomes that are generated by a serially correlated latent process, where the binding function is approximated using sparse grid integration (Heiss & Winschel, 2008) instead of simulation. Sparse grid integration does not suffer from the curse of dimensionality, making integration in multiple dimensions feasible, and is more accurate than a simulated approximation when the same number of nodes is used to evaluate the function.

The results in this paper show that the constrained optimization approach to indirect inference estimation outperforms the regular approach when the estimator for the auxiliary model does not have a closed-form solution. In that case the CPU time that is needed to estimate the parameters is an order of magnitude lower. In addition, there is a small benefit to using the constrained optimization approach when the number of structural parameters is large. In other cases the difference in performance is not as clear. The different variations in the experiments below should be used to provide some help to decide which method to use when estimating similar models.

The organization of the remainder of the paper is as follows. First I will give an overview of the method of estimation by indirect inference, introducing the notation. Then an MPEC formulation of the problem is presented. In the following sections I show Monte Carlo simulations for different models, comparing the performance of MPEC relative to the regular estimation of indirect inference. The final section concludes.

1.2 Indirect inference estimation

The setup is as follows. Y is a random variable that is generated from a structural or economic model. The conditional distribution of Y on X is written as $F_{Y|X}(Y \mid X, \theta)$, with F_θ as shorthand notation. X contains variables that are exogenous to the process of Y. For instance, in a regression context, X can contain background characteristics that are exogenous; in a time series context, X can contain lagged values of (latent) Y. The conditional distribution of the data is known up to a finite dimensional parameter vector $\theta \in \mathbb{R}^p$, which is the object of interest and is referred to as the structural parameter vector. Assume that we observe data $(y, x) = \{y_i, x_i\}_{i=1,\ldots,N}$ generated as a random sample from probability model F_θ.

Estimation by indirect inference is used when we can not easily compute the probability

distribution of θ given our observed data. The idea behind indirect inference is to obtain a set of auxiliary parameters by estimating an approximate or auxiliary model which is typically simple to estimate. These auxiliary parameters are linked to the underlying structural parameters through a binding function. By inverting the binding function using a metric we obtain estimates for the structural parameters. Each of the three elements of indirect inference estimation, the auxiliary model, the binding function and the metric, are explained below.

1.2.1 Auxiliary model

To use indirect inference we specify a set of statistics or parameters, $\mu \in \mathbb{R}^r, r \geq p$, generated by an auxiliary model. Examples of auxiliary parameters are conditional means and variances, transition probabilities between states, correlations, or the parameters that are the result of estimating a linear model. To identify the structural parameters, θ, the dimension of the auxiliary parameter vector should be at least as large the dimension of θ. The auxiliary model should capture the features of the underlying structural model, but they do not need to follow directly from the true likelihood[1]; a misspecified model capturing many of the features of the structural model can be used as an auxiliary model. I refer to the exogenous variables in the auxiliary model as Z, since they can be different from those in the structural model. All X are in Z, but Z can contain additional variables, such as transformations of X.

Similar to Jiang and Turnbull (2004), I define the auxiliary estimator implicitly as the function $\mu = \hat{m}(y, z)$ that solves $G_N(y, z, \mu) = 0$, where $G_N(y, z, \mu) = \frac{1}{N}\sum_{i=1}^{N} G(y_i, z_i, \mu)$. For one observation, $G(y_i, z_i, \mu)$ is any vector-valued function relating y_i, z_i, and μ, such that its expectation evaluated at the true μ_0 is zero, $E\left[G(y_i, z_i, \mu_0)\right] = 0$. It is required that $\hat{m}$ converges to the true μ_0, as the number of observations, N, goes to infinity.

All extremum estimation procedures, such as method of moments or maximum likelihood can be written using this implicit definition. For example, if the auxiliary model is the linear model, $Y = Z\mu + \varepsilon$, we obtain an estimate for μ, by setting the following

[1] If the structural model is simple to estimate, the true likelihood can be used as auxiliary model to correct for finite sample bias (Gouriéroux et al., 2010).

moment equations to 0

$$G_N(y, z, \hat{m}) = 0 \Leftrightarrow \frac{1}{N}\sum_{i=1}^{N} G(y_i, z_i, \hat{m}) = 0 \Leftrightarrow \frac{1}{N}\sum_{i=1}^{N} z_{i,r}(y_i - z_i'\hat{m}) = 0 \quad \forall r,$$

where r moments identify the r auxiliary parameters. Another example follows from setting the score of a log-likelihood function for a Probit model to 0.

$$\begin{aligned} G_N(y, z, \hat{m}) = 0 \quad &\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} \frac{\partial \log \mathcal{L}\left(\hat{m} \mid y_i, z_i\right)}{\partial m} = 0 \\ &\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} z_{i,r}\left(y_i \frac{\phi(z_i'\hat{m})}{\Phi(z_i'\hat{m})} - (1 - y_i)\frac{\phi(z_i'\hat{m})}{1 - \Phi(z_i'\hat{m})}\right) = 0 \quad \forall r \end{aligned}$$

In the first case there is an explicit solution for the estimator of the auxiliary parameters, $\hat{m}(y, z) = (z'z)^{-1}z'y$. In the second case there is no explicit solution for $\hat{m}$ and an iterative procedure has to be used to find a numerical solution. In order to allow for the possibility of auxiliary models without a closed-form solution, I adopt the implicit definition of the auxiliary estimator given above.

1.2.2 Binding function

Given the structural model, F_θ, we can relate θ and μ, by taking the expectation of $G(Y, Z, \mu)$ with respect to the structural model conditional on $Z = z$

$$\begin{aligned} H_N(\theta, \mu \mid Z = z) &= \frac{1}{N}\sum_{i=1}^{N} E_{F_\theta}(G(Y_i, Z_i, \mu) \mid Z_i = z_i) \\ &= \frac{1}{N}\sum_{i=1}^{N} \int_{\mathcal{Y}_i} G(Y_i, z_i, \mu) f_{Y_i|Z_i}(Y_i|z_i, \theta) dY_i. \end{aligned}$$

Conditioning on observing $Z = z$ means that we calculate a 'finite sample binding function' (see footnote 1, Gouriéroux et al., 1993). Our interest is not in the data generating process for the exogenous variables, so we use the empirical distribution of Z. Intuitively, the $\mu = \tilde{\mu}(\theta)$ that uniquely solves $H_N(\theta, \mu \mid Z = z) = 0$, returns the values of μ that we expect to observe, given the structural model, a value for θ, and given a set of exogenous variables, by integrating out all possible values of Y. This function is called the binding function. The full set of conditions on the binding function that result in a consistent

estimator for the true θ_0 is given in Gouriéroux et al. (1993)

When there is an explicit solution for the auxiliary estimator, $\hat{\mu} = \hat{m}(Y, Z)$, the $\mu = \tilde{\mu}(\theta)$ that solves $H_N(\theta, \mu \mid Z = z) = 0$ can be rewritten as

$$\tilde{\mu}(\theta \mid Z = z) = E_{F_\theta}(\hat{m}(Y, Z) \mid Z = z) = \int_{\mathcal{Y}} \hat{m}(Y, z) f_{Y|Z}(Y|z, \theta) dY.$$

In some cases an analytic solution to the binding function is available, but in most cases the value of the binding function needs to be approximated at given values of θ.

Fuleky and Zivot (2010b) mention three simulation methods that can be used to approximate the binding function. The first method can only be used only if the model does not contain covariates. In that case a single very long time-series is simulated according to the model for a specific value of θ. These simulated values are used to obtain an approximation for the binding function at the same value for θ.

The other two methods can be used with covariates. Both methods use multiple datasets, $h = 1, \ldots, H$, that are simulated with the same size as the observed data, such that every observation in the data is associated with H simulated paths (Gouriéroux et al., 1993). The simulated outcome variables of set h are denoted by $y^h(\theta) = \{y_i^h(\theta)\}$, which reflects that simulations depend on specific values for the structural parameters θ. Since z can contain lagged outcome variables, z may be partly simulated, which is denoted by z^h. The sets of random draws that are used to simulate outcome data has to be kept fixed throughout the estimation procedure.

These simulated datasets can be combined in two different ways to get an approximation for the binding function. The two methods are asymptotically the same, but have different finite sample and computational properties. In the first case, we find $\tilde{\mu}^h$ for each h that solves

$$H_N^h(\theta, \mu \mid Z = z^h) = \frac{1}{N} \sum_{i=1}^{N} G(y_i^h(\theta), z_i^h(\theta), \mu) = 0.$$

The expectation in the definition of the binding function is replaced by the sum over all simulations

$$\tilde{\mu}(\theta) = \frac{1}{H} \sum_{h=1}^{H} \tilde{\mu}^h(\theta).$$

Since the simulated $y^h(\theta)$ are constructed according to the structural model F_θ, this sum approximates the expectation above. This method of combining the different simulated datasets is computationally intensive, because the auxiliary parameters $\tilde{\mu}^h$ are calculated for each simulated dataset separately. Depending on the auxiliary model this can be computationally demanding. The benefit of this approach is that it has a built-in finite sample bias correction. The examples in section 1.4 and 1.5 use simulation to approximate the binding function.

In the second case, we find $\tilde{\mu}$ that solves the following approximation

$$\begin{aligned} H_N(\theta, \mu \mid Z = z) &= \frac{1}{N}\sum_{i=1}^{N}\int_{\mathcal{Y}_i} G(Y_i, z_i, \mu) f_{Y_i|Z_i}(Y_i|z_i, \theta) dY_i \\ &\approx \frac{1}{N}\sum_{i=1}^{N}\frac{1}{H}\sum_{h=1}^{H} G(y_i^h(\theta), z_i^h(\theta), \mu) = 0. \end{aligned}$$

Asymptotically this will give the same results as above. However, this method of approximating the binding function can not be used to remove finite sample bias, at the benefit of reduced computational cost. Simulation is not the only way to approximate the integral in this case. Gallant and Tauchen (1996) mention that instead of simulation, quadrature methods can be used in some cases to approximate the integral defining the binding function. For integration in a single dimension, quadrature methods, such as Gauss-Hermite quadrature, are well understood, and the nodes and weights of integration can be found for instance in Judd (1998). Gaussian quadrature integrates a polynomial of degree $2k-1$ exactly by summing k function evaluations at specified nodes multiplied by corresponding weights.

For integration in multiple dimensions a tensor product of the nodes and weights that are used for integration in a single dimension is typically used. The problem of this product rule to generate nodes and weights, is the exponentially increasing number of nodes that is needed if the number of dimensions of integration increases. To integrate a d-dimensional function with k nodes in each dimension, one needs to evaluate the function at k^d nodes.

Sparse grid integration (SGI) aims to solve this problem (Heiss & Winschel, 2008). SGI combines Gaussian quadrature rules for a single dimension without an exponentially

growing amount of nodes[2]. For $k = 5$ the number of nodes that are needed to achieve the required accuracy is shown for different dimensions in table 1.1. At low dimensions (1, 2 and 3) it is more efficient to use a product rule to generate integration nodes, but at higher dimensions the benefit of SGI is evident. In section 1.6, I use SGI to approximate the binding function in a binary time series model.

Table 1.1 – Number of nodes used by SGI or the product rule, $k = 5$

d	1	2	3	4	5	6	7	8	9	10
SGI	5	53	165	385	781	1433	2437	3905	5965	8761
k^d	5	25	125	625	3125	15625	78125	390625	1953125	9765625

1.2.3 Metric

If we have the same number of auxiliary and structural parameters, then the problem is to find θ, such that $H(\theta, \hat{\mu} \mid Z) = 0$. This reduces to $\hat{\mu} - \tilde{\mu}(\theta \mid Z) = 0$ when there is an explicit expression for the auxiliary estimator. This is analogous to method of moments estimation, where $\hat{\mu}$ are moments calculated from the data, and $\tilde{\mu}(\theta)$ are theoretical moments. To obtain estimates for the structural parameters, θ, the binding function needs to be inverted. If the auxiliary models exactly identify the structural parameters there is a unique inversion. This is the case if we have the same number of structural parameters and auxiliary parameters, $r = p$. If the structural parameters are over-identified by the auxiliary models, $r > p$, then there are three standard choices to map the auxiliary statistics to the structural parameters (Smith, 2008).

The Wald approach maximizes a weighted difference between estimated auxiliary parameters from the data and the simulations. The parameter estimates are the solution to the following optimization problem

$$\hat{\theta}^{Wald} = \arg\min_{\theta} \left(\hat{m} - \tilde{\mu}(\theta)\right)' \Omega \left(\hat{m} - \tilde{\mu}(\theta)\right),$$

where $\hat{m}$ contains the estimated auxiliary parameters from the observed data. There is a

[2]The intuition behind the smaller number of nodes is that SGI is exact for a polynomial of total order $2k - 1$ if k nodes are used in a single dimension. The total order for a tensor product of univariate polynomials each with maximum order $2k - 1$ is in general higher than $2k - 1$ and more nodes are needed to be exact to that level.

weighting matrix, Ω, and $\tilde{\mu}(\theta)$ is (an approximation to) the theoretical binding function as defined before. Smith (1993) calls this method the 'extended method of simulated moments', or EMSM, because each auxiliary parameter forms a moment.

A second method, referred to as Likelihood Ratio (LR) by Smith (2008), uses the likelihood of the auxiliary model, $\log \tilde{\mathcal{L}}(\cdot)$, as a metric

$$\hat{\theta}^{LR} = \arg\min_{\theta} \left(\log \tilde{\mathcal{L}}(\hat{m} \mid y, z) - \log \tilde{\mathcal{L}}(\tilde{\mu}(\theta) \mid y, z) \right).$$

This metric chooses θ such that the likelihood of $\hat{m}$ given the observed data is as close as possible to the likelihood of $\tilde{\mu}(\theta)$ given the observed data. The first term on the right-hand side does not depend on the structural parameters θ, which means it is just a constant from an optimization perspective. The value of this constant is important when testing between different models. This is the same as the method (called 'simulated quasi-maximum likelihood', or SQML) in Smith (1993), which in his example uses the likelihood function associated with the VAR as a quasi-likelihood function for the structural model.

A third method, Lagrange Multiplier (LM) in the terminology of Smith (2008) and referred to as efficient method of moments (EMM) by others, uses the score vector of the likelihood defined by the auxiliary model.

$$\hat{\theta}^{LM} = \arg\min_{\theta} S(\theta)' V S(\theta),$$

with $S(\theta)$ the score of the log-likelihood function of the auxiliary model using simulated data

$$S(\theta) = \frac{1}{H} \sum_{h=1}^{H} \frac{\partial}{\partial \mu} \log \tilde{\mathcal{L}} \left(\hat{\mu} \,\middle|\, y^h(\theta), x \right).$$

The score is evaluated using simulated data $y^h(\theta)$, but with the auxiliary parameters estimated from the data, $\hat{\mu}$. The score evaluated using the observed data is 0 by definition, and θ is chosen to make the score using simulated data as close to 0 as possible. The benefit of this method is that only the score has to be evaluated in every step, without estimating auxiliary models on simulated data in every step. The downside is that this method does not work well in small samples (see Fuleky & Zivot, 2010b, 2010a, for an overview).

Fuleky and Zivot (2010b, 2010a) propose a method based on EMM, claiming it has better finite sample properties. This method plugs the simulated binding function into the score vector and evaluates the score using observed data. I.e. they define $S(\theta)$ as

$$S(\theta) = \frac{1}{H} \sum_{h=1}^{H} \frac{\partial}{\partial \mu} \log \tilde{\mathcal{L}} \left(\tilde{\mu}(\theta) \mid y, x \right).$$

This function depends on $\tilde{\mu}(\theta)$. Depending on how the different simulated datasets are combined, this method can be used to correct for finite sample bias. However, this means that the auxiliary models again have to be estimated in every step to get an approximation of the binding function, and the computational advantages of EMM vanish[3].

1.3 MPEC formulation

This section describes how we can get the solution to the indirect inference problem by solving it as a mathematical program with equilibrium constraints (MPEC, Su & Judd, 2010). First, I'll summarise how an estimate for θ is usually obtained. I'll refer to this method as nested-fixed point (NFXP), since this method solves multiple optimization problems nested inside an outer optimization problem. As inputs the algorithm needs a random seed, initial values for the structural parameters, θ_0, and observed data, y and z. After estimating $\hat{m}$ on the observed data, and generating simulated errors, a loop starts to find optimal values for θ.

In each step of the loop, H sets of outcome data y^h are simulated for the current value of θ, using the structural model. Since some of the covariates in the auxiliary model may contain previous period outcome data, z^h may contain simulated values. Then we estimate $\tilde{\mu}^h$ for all h. These optimizations have to be performed in every step. Therefore, to save computing time it is usually recommended to use an auxiliary model with a closed-form solution for the estimator. By averaging over all $\tilde{\mu}^h$ we get an approximation for the binding function for the current value of θ, $\tilde{\mu}$. This value is used to evaluate the objective function, which is based on one of the metrics defined in the previous section.

[3] In their specific example, their new EMM method converges more quickly than the old EMM version, which is counterintuitive. They explain this through the irregular shape of the objective function in the old method, which might be specific to their example

If the optimum hasn't been found yet, the gradient of the objective function is approximated using finite differences. In order to approximate the gradient, we have to repeat the simulations and estimation of the auxiliary parameters with a slightly changed structural parameter, for each of the elements in the structural parameter vector. Then an optimal step is found, for instance using Newton's method. The process is repeated with a new value for the structural parameter vector, until the optimum is reached (up to some pre-determined tolerance).

The intuition behind MPEC is that we are not interested in the value of the binding function in intermediate steps of the optimization process. We only need to ensure that for the final solution, the binding function holds. This is achieved by adding variables and constraints to the optimization problem, defining the binding function. The exact formulation depends on the method that is used to approximate the binding function. First I'll give the formulation if there is an analytic expression for the binding function, or if it can be approximated using numerical methods. Then, the formulation is given when simulation is used to approximate the binding function.

The optimization problem that has to be solved for indirect inference estimation, formulated as an MPEC problem, is

$$\begin{aligned} \left(\hat{\theta}, \tilde{\mu}\right) &= \arg\min_{\theta,\mu} Q_N\left(\mu \mid y, z\right) \\ &\text{s.t.} \\ & H_N\left(\theta, \mu \mid Z = z\right) = 0 \end{aligned}$$

In this formulation μ is a vector of parameters that needs to be estimated, instead of a function of θ. In the optimum, μ equals the theoretical binding function for the estimated value of θ, $\tilde{\mu}(\hat{\theta})$. The objective function that we are minimizing, is one of the metrics defined above. For instance, in the case of the LR metric the objective function is defined as

$$Q_N\left(\mu \mid y, z\right) = \log \tilde{\mathcal{L}}\left(\hat{m} \mid y, z\right) - \log \tilde{\mathcal{L}}\left(\mu \mid y, z\right),$$

which is the same as before; the difference between the log-likelihood of the auxiliary parameters, $\hat{m}$, calculated from the observed data, and the log-likelihood of the theoretical

value of the auxiliary parameters, μ, evaluated using the observed data. The structural parameters, θ, do not enter the objective function. To guarantee the link between the structural and auxiliary parameters, the binding function is added as a set of constraints to the optimization problem. These constraints ensure that in the optimum $\tilde{\mu}$ equals the auxiliary parameter that we expect to see for the value of θ in the optimum, $\hat{\theta}$.

This optimization problem may look more difficult to solve than the nested fixed point formulation, where we don't have constraints, and use fewer variables. However, in the MPEC formulation the objective function is a simple function instead of a composite function including $\mu(\theta)$. Depending on the auxiliary model, analytic derivatives of the objective function with respect to all the parameters can be easily derived. The derivatives with respect to μ depend on the auxiliary model, but the derivatives with respect to θ are zero. One of the benefits of using analytic derivatives over approximating derivatives by finite differences is speed. To approximate a derivative, we need to evaluate the function at different values of the parameter vector. This can be time-consuming, especially if simulation is used and simulation of the datasets takes a long time; new datasets have to be simulated for each element of the parameter vector to obtain an approximation for the gradient[4].

The constraints are more complex, because these link the structural parameters to the auxiliary parameters. The constraints defining the binding function are usually based on moment conditions, or the score of the likelihood function. Analytic gradients of these with respect to the auxiliary parameters are straightforward to derive. For instance, if the score is used to define the binding function, the gradient equals the Hessian of the auxiliary likelihood. The derivative with respect to the structural parameters, θ, is more difficult to derive analytically and it depends on the structural model, whether calculating analytic derivatives is feasible.

The burden on the programmer is not increased by much if standard solvers are used. With standard solvers, non-linear optimization problems with constraints, can be solved in a straightforward manner; the objective function, the constraints and the gradients

[4]Another potential benefit of being able to derive the gradients analytically, is to look for different approximations of the gradient directly, either through simulation or some other approximation method. Instead of approximating the derivative by finite differences applied to a function which was approximated using simulation.

have to be programmed, but the solver takes care of finding optimal step sizes etc. The solver that I use here, is Ipopt (Wächter & Biegler, 2006), which is free and open-source. Other optimization packages that can be called from standard programming languages are SNOPT (Gill, Murray, & Saunders, 2002) and KNITRO (Byrd, Nocedal, & Waltz, 2006). The performance of each of these solvers depend on the underlying optimization problem.

The following example shows the NFXP and MPEC formulations graphically using a simple probit model with one structural parameter. There is a latent variable, Y_i^*, that depends linearly on a single covariate, X_i

$$Y_i^* = X_i\theta + \varepsilon_i,$$

where $\varepsilon_i \sim$ i.i.d. $N(0,1)$. We observe a discrete outcome Y_i defined as

$$Y_i = \begin{cases} 1 & \text{if } Y_i^* > 0 \\ 0 & \text{otherwise.} \end{cases}$$

The parameter of interest is the one-dimensional structural parameter, θ. This model can be estimated using a standard probit, but as an example I show how to estimate it using indirect inference. As an auxiliary model, I use a linear probability model

$$Y_i = Z_i'\mu + \eta_i,$$

where $Z_i = (1, X_i)$. The auxiliary parameters are $\mu = (\mu_0, \mu_1)$, combined with the variance of η_i, σ_η^2. The likelihood for the linear probability model is

$$\log \tilde{\mathcal{L}}\left(\mu, \sigma_\eta^2 \mid Y, Z\right) = -\frac{N}{2}\ln 2\pi - \frac{N}{2}\ln \sigma_\eta^2 - \frac{1}{N}\sum_{i=1}^{N}\frac{(Y_i - Z_i'\mu)^2}{2\sigma_\eta^2}.$$

This auxiliary likelihood is also used in the objective function, which is based on the LR metric for this example. To convert the auxiliary likelihood to an implicit binding function, we use the estimating equations following from the first-order conditions of the auxiliary

likelihood with respect to μ, i.e. the score of the likelihood

$$\frac{\partial \log \tilde{\mathcal{L}}\left(\mu, \sigma_\eta^2 \mid Y, Z\right)}{\partial \mu} = 0 \Leftrightarrow \frac{1}{N}\sum_{i=1}^{N} \frac{Z_i(Y_i - Z_i'\mu)}{2\sigma_\eta^2} = 0 \Leftrightarrow \frac{1}{N}\sum_{i=1}^{N} Z_i(Y_i - Z_i'\mu) = 0,$$

which implies that $G(Y_i, Z_i, \mu) = Z_i(Y_i - Z_i'\mu)$. Since in this case, $\sigma_\eta^2 > 0$, the binding between θ and μ is independent of σ_η^2. To simplify the graphs below, I use this independence and don't include the equation following from the first-order condition with respect to σ_η^2[5]. The implicit binding function follows, by taking the expectation with respect to the structural model

$$\begin{aligned}
H_N(\theta, \tilde{\mu} \mid Z) = 0 \quad &\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} E_{F_\theta}(G(Y_i, Z_i, \tilde{\mu}) \mid Z_i = z_i) = 0 \\
&\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} \int_{\mathcal{Y}_i} G(Y_i, z_i, \tilde{\mu}) f_{Y_i|Z_i}(Y_i|z_i, \theta) dY_i = 0 \\
&\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} \int_{\mathcal{Y}_i} z_i(Y_i - z_i'\tilde{\mu}) f_{Y_i|Z_i}(Y_i|z_i, \theta) dY_i = 0 \\
&\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} z_i z_i'\tilde{\mu} - \frac{1}{N}\sum_{i=1}^{N} z_i \int_{\mathcal{Y}_i} Y_i f_{Y_i|Z_i}(Y_i|z_i, \theta) dY_i = 0 \\
&\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} z_i z_i'\tilde{\mu} - \frac{1}{N}\sum_{i=1}^{N} z_i \int_{\mathcal{Y}^*{}_i} 1(Y_i^* > 0) f_{Y_i^*|X_i}(Y_i^*|x_i, \theta) dY_i^* = 0 \\
&\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} z_i z_i'\tilde{\mu} - \frac{1}{N}\sum_{i=1}^{N} z_i \int_{-\infty}^{x_i\theta} f_{\varepsilon_i}(\varepsilon_i) d\varepsilon_i = 0 \\
&\Leftrightarrow \quad \frac{1}{N}\sum_{i=1}^{N} z_i z_i'\tilde{\mu} - \frac{1}{N}\sum_{i=1}^{N} z_i \Phi(x_i\theta) = 0.
\end{aligned}$$

Because there is a closed-form solution for the auxiliary estimator, there is an equivalent direct formulation of the binding function in this case

$$\tilde{\mu} = \left(\frac{1}{N}\sum_{i=1}^{N} z_i z_i'\right)^{-1} \frac{1}{N}\sum_{i=1}^{N} z_i \Phi(x_i\theta),$$

which relates $\tilde{\mu}$ to θ for a specific sample.

Figure 1.1 shows the NFXP objective function that we aim to minimize in the left panel,

[5] Instead, since there is a closed-form expression for σ_η^2 as a function of μ and the data I substitute $\sigma_\eta^2 = \frac{1}{N}\left(y_i - Z_i'\mu\right)^2$ in the objective function.

and its first derivative in the right panel. This is a function of only the single structural parameter θ, because the binding function is substituted directly in the objective function. The value of θ that minimizes this function is the indirect inference estimate, shown as $\hat{\theta}$ in the figure.

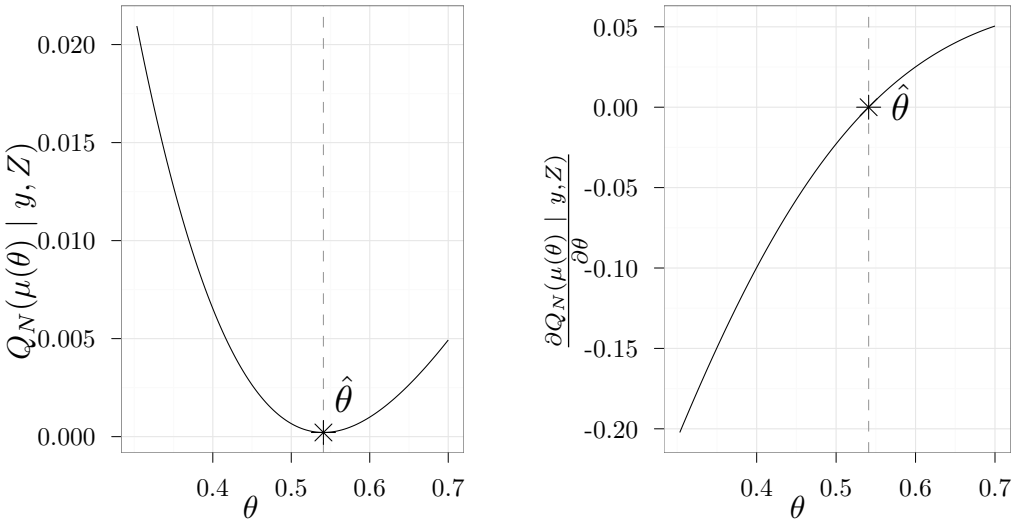

Figure 1.1 – Objective and gradient in terms of θ for NFXP formulation

Figure 1.2 shows the MPEC optimization problem. The horizontal and vertical axis correspond to the auxiliary parameters μ_0 and μ_1. The binding function, $H_N(\mu, \theta \mid Z) = 0$ mapping θ to μ is shown as the black line. This line shows the constraints of the optimization problem. Different values of θ are shown next to the line. For example, when the true value of θ is 0, then the expected value for μ_0 is 0.5 and for μ_1 is 0. If $\theta = 0$, then X_i has no effect on Y_i and there is a fifty-fifty chance of observing a 0 or a 1.

The dark grey contour lines belong to the objective function that we want to minimize, $Q_N(\mu \mid y, Z)$. The objective function is a function of the auxiliary parameters μ (and implicitly σ_η^2). The objective function is a well-behaved function that attains its minimum, by definition, at $\mu = \hat{m}$. However, this point does not lie on the binding function, so there is no value of θ that generates $\hat{m}$ as a possible solution. The point that is in the feasible set, where the objective function is minimized, is marked by a star. At this point the constraint is tangent to the contours of the objective function.

The two figures are related. The NFXP objective function is the slice of the MPEC objective function along the binding function. Since NFXP forces the binding function to

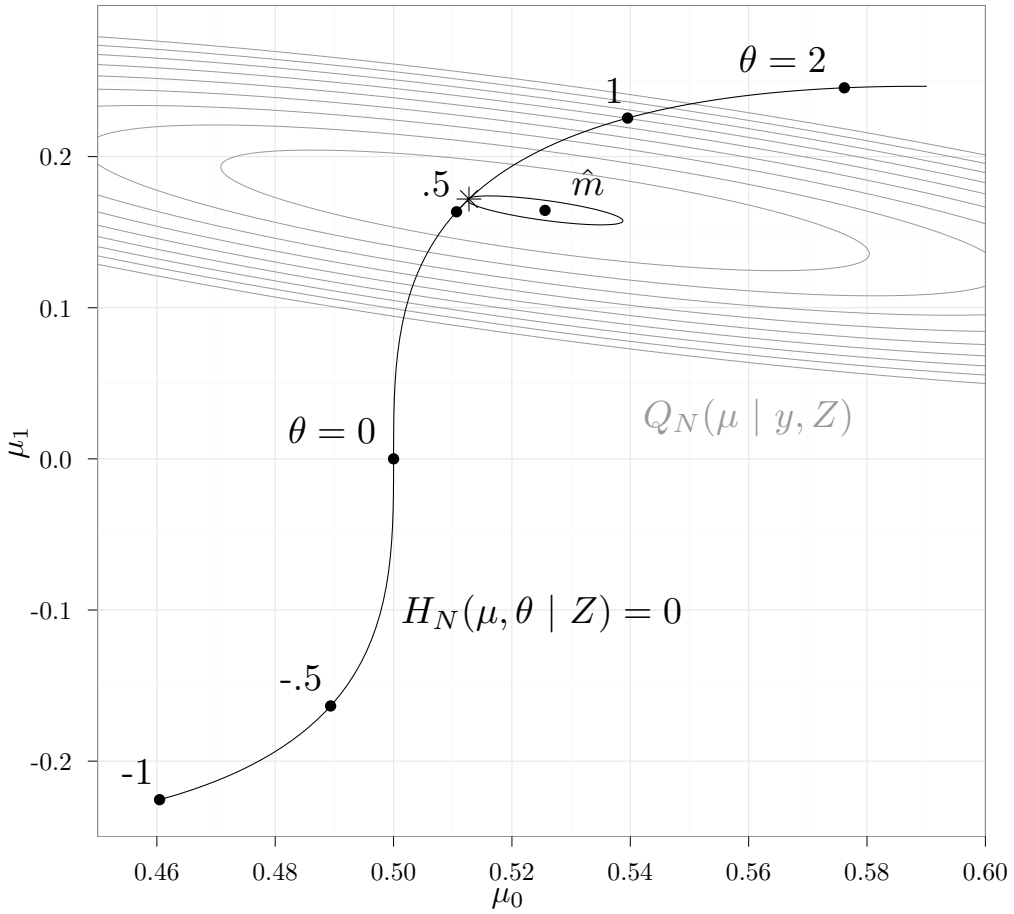

Figure 1.2 – Objective function and constraints for MPEC formulation

hold in every step of the optimization procedure, we are walking along the black line of figure 1.2 in the direction of the optimum. Steps outside of the black line are not allowed in the NFXP formulation. The solution algorithm for MPEC doesn't force the constraint to hold at the intermediate steps, but it should hold at the optimum. Steps outside the feasible set are allowed.

Figure 1.2 also shows a natural starting value for μ. The auxiliary parameters calculated from the data, $\hat{m}$, can be calculated before the optimization problem is solved. If the structural model is correct, these values will be close to the optimal value of μ and $\hat{m}$ will contain good starting values for μ. Starting values for θ can be chosen in the same way as they are chosen for the NFXP formulation.

1.3.1 Formulation with simulation

When simulation is used to approximate the binding function, the MPEC formulation of indirect inference is different. Additional parameters $\tilde{\mu}^h$ corresponding to the auxiliary parameters for each of the H simulated sets are added to the problem. Each of the simulated datasets results in an estimated auxiliary parameter, which needs to be reflected in the constraints.

$$\begin{aligned}\left(\hat{\theta}, \tilde{\mu}, \tilde{\mu}^h\right) \quad = \quad & \arg\min_{\theta,\mu,\mu^h} \arg\min_{\theta,\mu} Q_N\left(\mu \mid y, z\right) \\ \text{s.t.} \quad & \\ & \mu - \frac{1}{H}\sum_{h=1}^{H} \mu^h = 0 \\ & \frac{\partial}{\partial \mu^h} \log \tilde{\mathcal{L}}\left(\mu^h \;\middle|\; y^h(\theta), x\right) = 0 \quad \forall h\end{aligned}$$

The first set of equality constraints defines how the binding function is approximated by averaging over the auxiliary parameters of H simulations. This is the same equation that we have in the NFXP formulation, except that it enters the problem as a constraint here. The second set of constraints define that each $\tilde{\mu}^h$ is the solution to the optimization problem estimating the auxiliary model on a set of simulated data, h. Compared to the case where simulation is not needed, this optimization problem has even more variables and constraints. This can be handled by Ipopt, because the problem is very sparse; the auxiliary estimates for one simulated dataset are independent of all other auxiliary estimates which means that there are many zeroes in the Jacobian of the constraints.

The problem contains $p + r \cdot (1 + H)$ parameters and $r \cdot (1 + H)$ constraints, where p is the dimension of θ and r is the dimension of μ as before. This seems large compared to p parameters in the outer-loop minimization in the NFXP formulation. However, for each step updating those p parameters, H separate minimization problems with r parameters have to be solved in an inner loop. Also, the MPEC formulation contains a lot of sparseness, which makes the problem easier to solve than a general non-linear optimization problem with the same amount of parameters and constraints. The first-order conditions of the auxiliary model are independent for the different simulated sets, which

means that there are many zeros in the Jacobian of the constraints. In total there are $r \cdot (1 + H) + r \cdot (p + r) \cdot H$ non-zero elements in the Jacobian of the constraints, whereas the total number of elements is $r \cdot (H + 1) \cdot (p + r \cdot (1 + H))$. This means that the fraction of non-zero elements is $\frac{1}{(p+r)(1+H)} + \frac{H}{(1+H)^2}$.

If the structural model contains discrete outcomes, simulation of the binding function can lead to problems with optimization (see for instance Magnac et al., 1995; An & Liu, 2000). In the NFXP formulation, simulated discrete outcomes lead to a non-smooth objective function. A solution has been proposed by Keane and Smith (2004). They introduce a smoothing parameter λ to smooth the simulated discrete outcomes, and consequently to smooth the objective function. This results in biased estimates of the structural parameters, but Keane and Smith (2004) show that the bias can be reduced by adding one step to the optimization procedure.

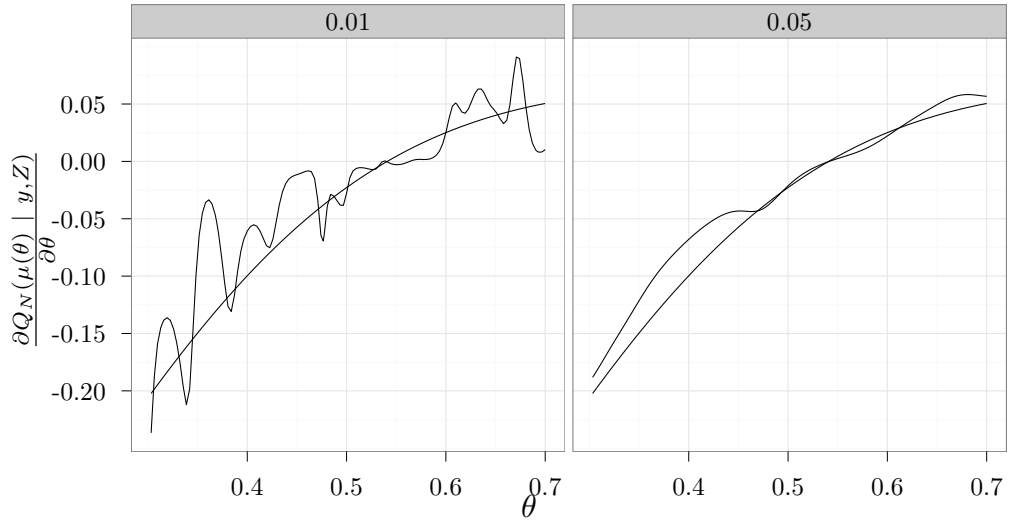

Figure 1.3 – Gradient of NFXP objective function with $\lambda = .01$ (left) and $\lambda = .05$ (right)

The amount of smoothing that should be chosen for a particular model is a matter of experimentation. With a small amount of smoothing, optimization problems still occur, because the objective function is now continuous, but contains local minima. Also, convergence can be slow, because the first derivative is very volatile, which leads to wrong stepsizes in the optimization algorithm. This can be seen in figure 1.3, where the gradient of the NFXP objective function is shown for two levels of smoothing for the same example as above, $H = 1$. The figure in the panel on the left shows that there are two local

minimum in this particular case, where the gradient is zero for two different values of θ.

In the NFXP formulation, increasing the number of simulations improves the smoothness of the objective function, because the non-smoothness in the auxiliary parameters is smoothed by averaging over all simulations. In the MPEC formulation the objective function does not depend on simulations and this function is smooth. However, the constraints are non-smooth, which can be seen in figure 1.4. Moreover, every additional simulation adds a new set of constraints. In the MPEC case, adding simulations does not improve the smoothness of the constraints, but instead adds non-smooth constraints to the problem. In section 1.5, I investigate whether this leads to problems in one of the models presented by Keane and Smith (2004).

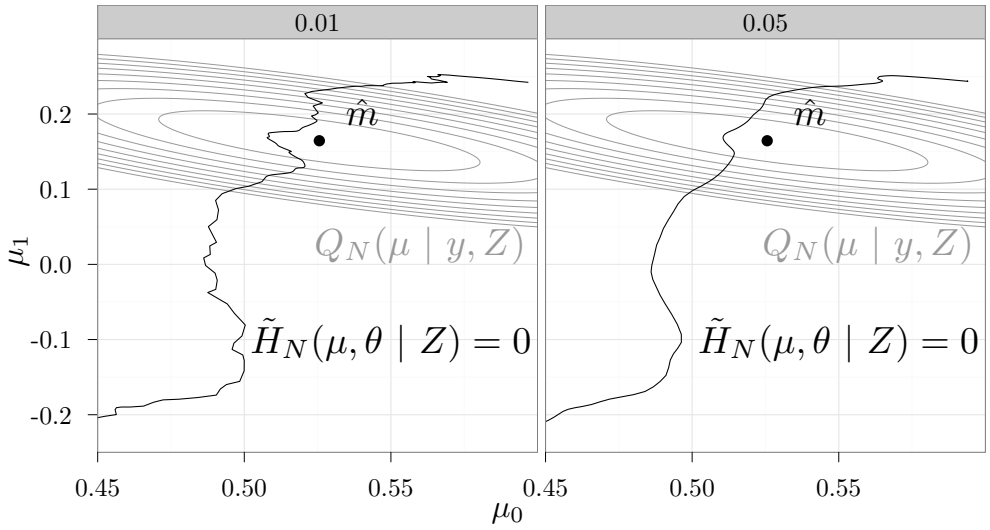

Figure 1.4 – Objective function and constraints for MPEC formulation with $\lambda = .01$ (left) and $\lambda = .05$ (right)

1.4 Example 1: Linear panel data model

In this section I use indirect inference applied to a linear panel data model. Gouriéroux et al. (2010) show that indirect inference can be used as a bias-correction mechanism for panel data, where the bias is a result of having a short time dimension, i.e. T is small. The model in this section is the same as the one estimated by Gouriéroux et al. (2010), but with added covariates. For individual $i = 1, \ldots, N$ at time $t = 1, \ldots, T$, we observe

outcome y_{it}, which follows an AR(1) process

$$y_{it} = \alpha_i + \phi y_{it-1} + x'_{it}\gamma + \varepsilon_{it},$$

where α_i is a fixed effect, x_{it} is a vector of covariates of dimension K, and $\varepsilon_{it} \sim$ i.i.d. $N(0, \sigma_\varepsilon^2)$. The initial value that we observe, y_{i0}, is assumed to be drawn from the stationary distribution of the AR(1) process above

$$y_{i0} = \frac{\alpha_i}{1-\phi} + \frac{\varepsilon_{i0}}{\sqrt{1-\phi^2}},$$

where $\varepsilon_{i0} \sim N(0, \sigma_\varepsilon^2)$ independent of all other ε_{it}. In the simulations below, α_i are drawn independently from $N(0,1)$[6]. The value of σ_ε is set to 1. The way x_{it} is constructed depends on the experiment, and will be described for each case below.

In this example, the object of interest is a vector of structural parameters $\theta = (\phi, \gamma)$[7]. In order to estimate these parameters using indirect inference, we need to simulate values for the endogenous variables, $y_{i,t}$, given the exogenous variables, $x_{i,t}$, according to the structural model above. I start by drawing α_i^h and $\varepsilon_{i,t}^h$ for simulation $h = 1, \ldots, H$. If the optimization procedure consists of multiple steps, we need to ensure that the same random draws are used in every step. This could be done by simulating the data using the same seed for the random generator, or, as I do here, by simply saving the draws in memory. For a given value of the structural parameters θ, we can then calculate $y_{i,0}^h$ and its subsequent values, $y_{i,1}^h, \ldots, y_{i,T}^h$. The part that depends on the exogenous covariates, $x'_{it}\gamma$, is the same for all of the H simulated datasets and is calculated only once every time the parameter γ is updated.

The auxiliary model that I use, is based on a commonly used biased estimator for

[6]Note that we're implicitly assuming that we know the true distribution of α_i, including the true values of the parameters describing this distribution. Even though it looks like taking the differences of y_{it} will remove the effect of α_i from the estimation, the binding function will depend on the distribution that was chosen for α_i. This means that if α_i in the data is not $N(0,1)$, for instance because its variance is not equal to 1, or because its distribution is non-normal, then the wrong binding function will be used when α_i is drawn from $N(0,1)$. Also, if α_i is correlated with x_{i0} without taking this into account in the simulated values for α_i, this results in the wrong binding function being used, and thus in biased estimates. Since this paper is about the computational aspects of indirect inference estimation, we assume that we know the true distribution of α_i.

[7]As in Gouriéroux et al. (2010), the variance of the error, σ^2 is assumed to be known. The method presented here is applicable to the case where σ_ε^2 is unknown.

the structural parameters, and is the same as the one used by Gouriéroux et al. (2010). One way to estimate the parameters ϕ and γ, is by transforming the data to eliminate α_i from the equation. Define the following time means, $y_{i*} = \frac{1}{T}\sum_{t=1}^{T} y_{it}$, $y_{i*-1} = \frac{1}{T}\sum_{t=0}^{T-1} y_{it}$ and $x_{i*} = \frac{1}{T}\sum_{t=1}^{T} x_{it}$. Transformed variables can then be defined as $\tilde{y}_{it} = y_{it} - y_{i*}$, and similarly for the other variables. The auxiliary model we consider is the linear model

$$\tilde{y}_{i,t} = \tilde{z}_{i,t}\kappa + \tilde{\varepsilon}_{i,t},$$

where $\tilde{z}_{i,t} = (1, \tilde{y}_{i,t-1}, \tilde{x}_{i,t}),\ t = 1, \ldots, T$. The maximum likelihood estimator for κ returns biased estimates for ϕ and γ, as shown in Nickell (1981), because the transformation introduces a correlation between $\tilde{z}_{i,t}$ and $\tilde{\varepsilon}_{it}$, through $\tilde{y}_{it-1}$. This correlation disappears as $N \to \infty$ and $T \to \infty$. For panel data T is fixed, resulting in an inconsistent estimator.

The set of statistics that are generated using this auxiliary model are a combination of the regression parameters and the estimated variance of the residuals, $\mu = (\kappa, \sigma^2)$. In this specific case, the addition of an intercept to the auxiliary model, and adding σ^2 to the set of statistics is not necessary for identification of the structural parameters. The inclusion of σ is necessary if we want to estimate the variance parameter σ_ε. Since the auxiliary model contains more parameters than the structural model, the model is overidentified and we choose the LR metric to link the two sets of parameters together[8].

The auxiliary model is estimated by maximizing the log-likelihood implied by the normal linear model. The contribution to the log-likelihood is, for given i, t,

$$\log \tilde{\mathcal{L}}\left(\kappa, \sigma^2 \mid \tilde{y}_{it}, \tilde{z}_{it}\right) = -\frac{1}{2}\ln 2\pi - \frac{1}{2}\ln \sigma^2 - \frac{(\tilde{y}_{it} - \tilde{z}_{it}{}'\kappa)^2}{2\sigma^2},$$

where we use $\tilde{y}^h_{i,t}$ and $\tilde{z}^h_{i,t}$ to calculate the log-likelihood for simulated data. In the NFXP formulation of indirect inference, we obtain estimates of the auxiliary parameters for each

[8] This is different from Gouriéroux et al. (2010), who use only one parameter, ϕ, in the structural model. Their auxiliary model contains one parameter, μ, which is the regression coefficient on lagged y. They then minimize the distance between $\hat{\mu}$ obtained from the observed data and $\tilde{\mu}(\phi)$ obtained from simulated data. Because they have exactly the same number of structural and auxiliary parameters, they do not need to choose a metric to link the two together.

of the simulated datasets as the solution to the following optimization problem

$$\tilde{\mu}^h(\theta) = \arg\max_{\kappa^h,\sigma^h} \sum_{t=1}^{T}\sum_{i=1}^{N} \log \tilde{\mathcal{L}}\left(\mu^h \;\middle|\; \tilde{y}^h_{it}(\theta), \tilde{z}^h_{it}(\theta)\right).$$

The estimator for κ^h is the regular OLS estimator, which has the well-known closed form solution, $\hat{\kappa}^h = \left(\tilde{z}^{h\prime}\tilde{z}^h\right)^{-1}\tilde{z}^{h\prime}y^h$.

$$\hat{\theta}^{LR} = \arg\min_{\theta} \sum_{t=1}^{T}\sum_{i=1}^{N} \left(\log \tilde{\mathcal{L}}\left(\hat{\mu} \mid \tilde{y}_{i,t}, \tilde{z}_{i,t}\right) - \log \tilde{\mathcal{L}}\left(\tilde{\mu}(\theta) \mid \tilde{y}_{i,t}, \tilde{z}_{i,t}\right)\right),$$

where

$$\tilde{\mu}(\theta) = \frac{1}{H}\sum_{h=1}^{H} \tilde{\mu}^h(\theta).$$

For the MPEC formulation we have the following optimization problem

$$\begin{aligned}
\hat{\theta}^{LR} \quad = \quad & \arg\min_{\theta,\tilde{\mu},\tilde{\mu}^h} \sum_{t=1}^{T}\sum_{i=1}^{N} \left(\log \tilde{\mathcal{L}}\left(\hat{\mu} \mid \tilde{y}_{i,t}, \tilde{z}_{i,t}\right) - \log \tilde{\mathcal{L}}\left(\tilde{\mu} \mid \tilde{y}_{i,t}, \tilde{z}_{i,t}\right)\right) \\
& \text{s.t.} \\
& \tilde{\mu} - \frac{1}{H}\sum_{h=1}^{H} \tilde{\mu}^h = 0 \\
& \sum_{t=1}^{T}\sum_{i=1}^{N} \frac{\partial}{\partial \tilde{\mu}^h} \log \tilde{\mathcal{L}}\left(\tilde{\mu}^h \;\middle|\; \tilde{y}^h_{it}(\theta), \tilde{z}^h_{it}(\theta)\right) = 0 \quad \forall h.
\end{aligned}$$

In this case the second set of constraints consists of the derivative of the likelihood of the normal linear model with respect to κ and with respect to σ^2

$$\frac{\partial}{\partial \tilde{\mu}^h} \log \tilde{\mathcal{L}}\left(\tilde{\mu}^h \;\middle|\; \tilde{y}^h_{it}(\theta), \tilde{z}^h_{it}(\theta)\right) = \begin{pmatrix} \frac{1}{\sigma^2}\left(\tilde{y}^h_{i,t}(\theta) - \tilde{z}^h_{i,t}(\theta)'\kappa\right)\tilde{z}^h_{i,t}(\theta) \\ -\frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}\left(\tilde{y}^h_{i,t}(\theta) - \tilde{z}^h_{i,t}(\theta)'\kappa\right)^2 \end{pmatrix} = 0.$$

The first rows of these constraints of course correspond to the OLS estimator of κ. The last row corresponds to the estimator for σ^2. These constraints can be simplified by multiplying the first rows by σ^2, and multiplying the last row by $2\sigma^4$. This simplification introduces some additional sparseness in the Jacobian of the constraints, since σ^2 drops from the first rows[9]. Since the auxiliary model can be calculated quickly, we don't expect

[9]This simplifications turns out to result in a slightly smaller number of iterations for the MPEC method to reach the optimum.

to see a large benefit from using MPEC in this case.

As initial values for the auxiliary parameters we use $\hat{\mu}$, which is obtained from estimating the auxiliary model on the observed data. For the structural parameters there is a direct link between the coefficient on lagged y and the coefficient on x in the auxiliary model to ϕ and γ. We take these biased estimates from the auxiliary model as initial values for the structural parameters.

Changing the value of ϕ

In the first experiment I estimate the linear panel data above with one covariate, $x_{i,t} \sim$ i.i.d. $N(0,1)$. The two structural parameters that we estimate, are ϕ and γ. In this set of experiments the true value for γ is 1, and ϕ is taken to be 0, .4, .85, and .99. For all of the experiments below we simulate data for 1,000 individuals and replicate the estimations 10,000 times with different simulated sets of observed data. Table 1.2 shows the results when we estimate using indirect inference with 20 simulated paths for each individual.

The first row shows the mean of the estimated parameters for the different Monte Carlo replications and its standard deviation in parentheses, for fixed effect estimation. The estimates for γ and ϕ are clearly biased in this case. The bias of γ is higher for larger values of ϕ, because a larger value of ϕ introduces a higher correlation between $\tilde{y}_{i,t-1}$ and $\tilde{\varepsilon}_{i,t}$. Furthermore, we see that the means and standard deviations of the NFXP, MPEC1 and MPEC2 estimates are the same, and that these provide unbiased estimates of γ and ϕ. This is as expected, given the results of Gouriéroux et al. (2010).

The three methods solve the same optimization problem, which means that the solution we obtain should be the same, not only on average, but for every simulated dataset. The first twelve digits of the value of the objective function in the optimum are the same for all Monte Carlo replications for NFXP, MPEC1 and MPEC2. The first four digits of the parameter estimates are the same for virtually all Monte Carlo replications. Most of the parameter estimates agree in more places, especially when comparing MPEC1 and MPEC2. Except for small rounding errors, there are no differences between estimating via NFXP, MPEC1 or MPEC2; the same optimum is found. Also, each of the methods converge successfully for all of the simulated datasets except when $\phi = .99$, when the

Table 1.2 – Comparison of FE, NFXP, MPEC1 and MPEC2 estimation

	$\phi = 0$		$\phi = .4$		$\phi = .85$		$\phi = .99$	
method	γ	ϕ	γ	ϕ	γ	ϕ	γ	ϕ
FE	0.977	-0.115	0.951	0.222	0.886	0.584	0.860	0.706
	(0.016)	(0.008)	(0.016)	(0.009)	(0.016)	(0.009)	(0.015)	(0.009)
NFXP	1.000	0.001	1.000	0.400	0.998	0.849	0.998	0.989
	(0.016)	(0.009)	(0.016)	(0.010)	(0.016)	(0.009)	(0.016)	(0.007)
MPEC1	1.000	0.001	1.000	0.400	0.998	0.849	0.998	0.989
	(0.016)	(0.009)	(0.016)	(0.010)	(0.016)	(0.009)	(0.016)	(0.007)
MPEC2	1.000	0.001	1.000	0.400	0.998	0.849	0.998	0.989
	(0.016)	(0.009)	(0.016)	(0.010)	(0.016)	(0.009)	(0.016)	(0.007)

Mean estimates are based on 10,000 replications of simulated datasets with $N = 1000$ individuals and $T = 5$ periods, standard deviations in parentheses. $H = 20$ simulated paths were used for each individual to approximate the binding function. A value of $\gamma = 1$ was used to simulate the data.

optimization fails in 362, 440 and 569 cases for MPEC1, MPEC2 and NFXP respectively[10].

To compare the performance of the different formulations, the number of iterations for the successful optimizations are shown in the bottom panel of figure 1.5. NFXP uses one iteration fewer than MPEC1 and MPEC2 on average, except when $\phi = .99$. The number of iterations for MPEC1 and MPEC2 are the same, except for $\phi = .99$, where MPEC2 uses fewer iterations in some cases.

The total CPU time used is shown in the top panel of the same figure. The CPU time largely reflects the differences in number of iterations. The number of seconds per iteration are comparable for MPEC2 and NFXP, with MPEC2 being approximately 3 percent slower for this experiment. In the current implementation MPEC1 uses approximately 30 percent more time per iteration.

Changing the value of H

In the second experiment, I change the number of simulated paths used for each estimation. The model again includes one covariate, $x_{i,t} \sim$ i.i.d. $N(0, 1)$. In these experiments the true value for γ is 1, and ϕ is fixed to .85. The number of simulated paths, H, is varied from 20, 200, 2000 to 5000. For NFXP the number of control variables is equal to the number of structural parameters, and independent from the number of simulated paths. A priori

[10] Failure to find a solution in this case means that the solver didn't find the optimal solution within 100 iterations from the chosen starting value.

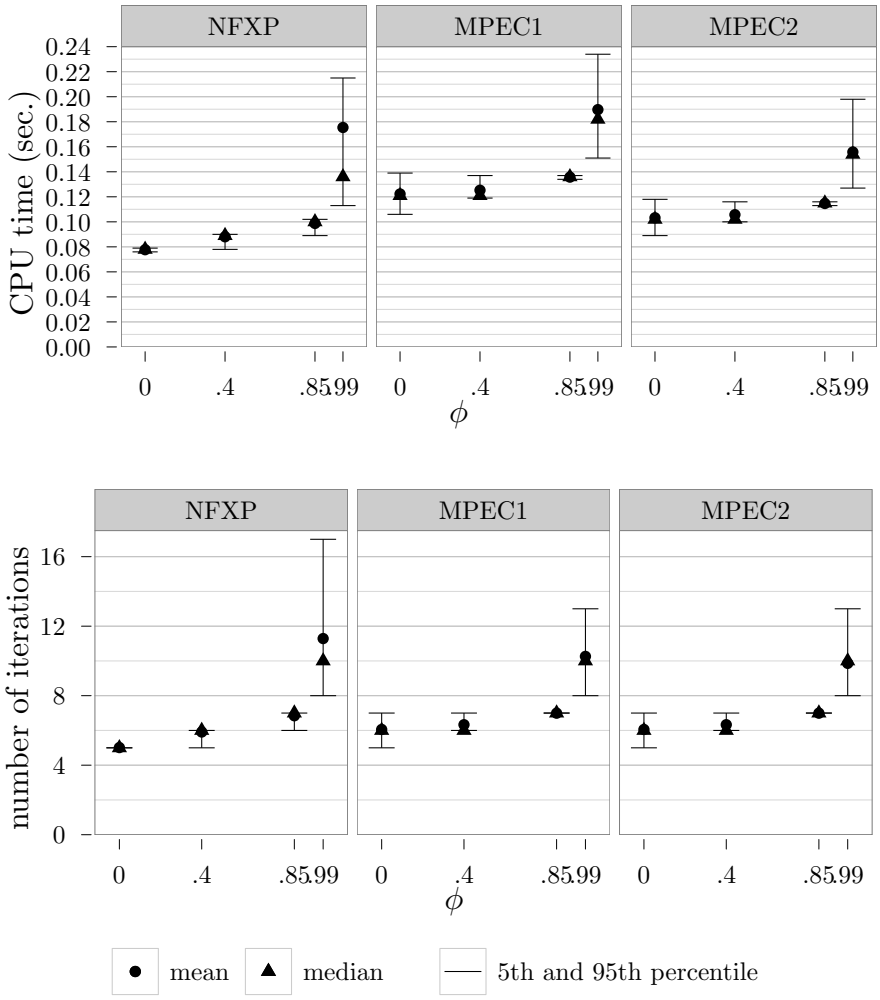

Figure 1.5 – Comparison of CPU time in seconds and number of iterations for NFXP, MPEC1 and MPEC2, $H = 20$

we expect to see no difference in performance for different values for H. A change in the number of simulated paths, implies a change in the number of control variables and constraints for MPEC1 and MPEC2. For each additional simulation, we add control variables and constraints to the problem equal to the number of auxiliary parameters, $r = 4$.

The bottom panel of figure 1.6 shows that the number of iterations does not depend on the number of simulated paths for NFXP. For MPEC1 and MPEC2 the number of iterations increases when we increase the number of simulated paths. Increasing the number of simulated paths, increases the number of constraints. All the constraints are linked together through the structural parameters. The effect of a change in γ or ϕ on the

value of the constraints is non-linear, because $y_{i,t}$ is a function of ϕ, γ and $y_{i,t-1}$, which in itself is again a function of ϕ and γ etc. From the top panel we see that the number of seconds per iteration per simulation is not increasing for both NFXP and MPEC. From these experiments it looks like the MPEC methods are slightly less scalable in terms of number of simulations than NFXP, but for a large number of simulations the methods are still feasible.

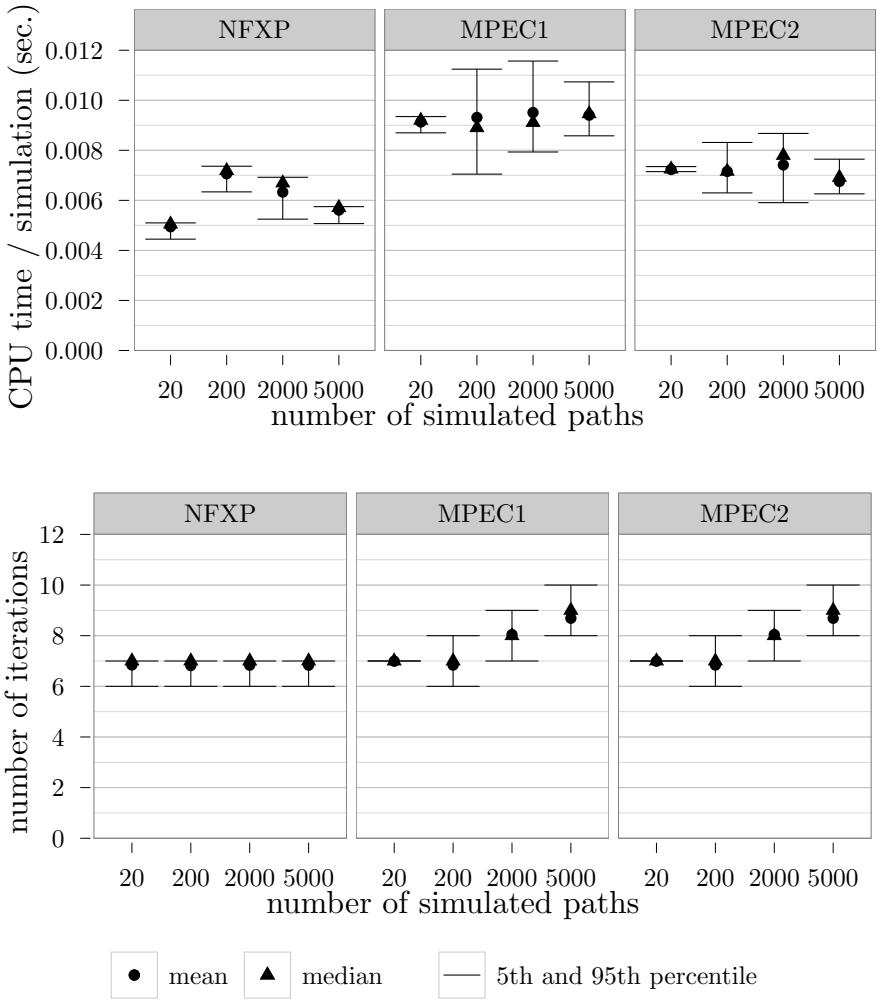

Figure 1.6 – Comparison of CPU time in seconds and number of iterations for NFXP, MPEC1 and MPEC2

Increasing the number of structural parameters

In the third experiment we look at the effect of increasing the number of structural parameters on the computing time. The number of parameters is increased by changing K,

the dimension of x. The variables x_{itk}, for individual i, time period t and index k are drawn independently from $N(0,1)$. The number of structural parameters is $K+1$, i.e. the dimension of x plus one for ϕ. Note that the number of auxiliary parameters also increases in order to identify all the parameters. This implies that the number of variables and constraints in the MPEC formulation increases, and for NFXP we need to solve a linear system with an increasing dimension to obtain estimates of κ. We use 20 simulated paths.

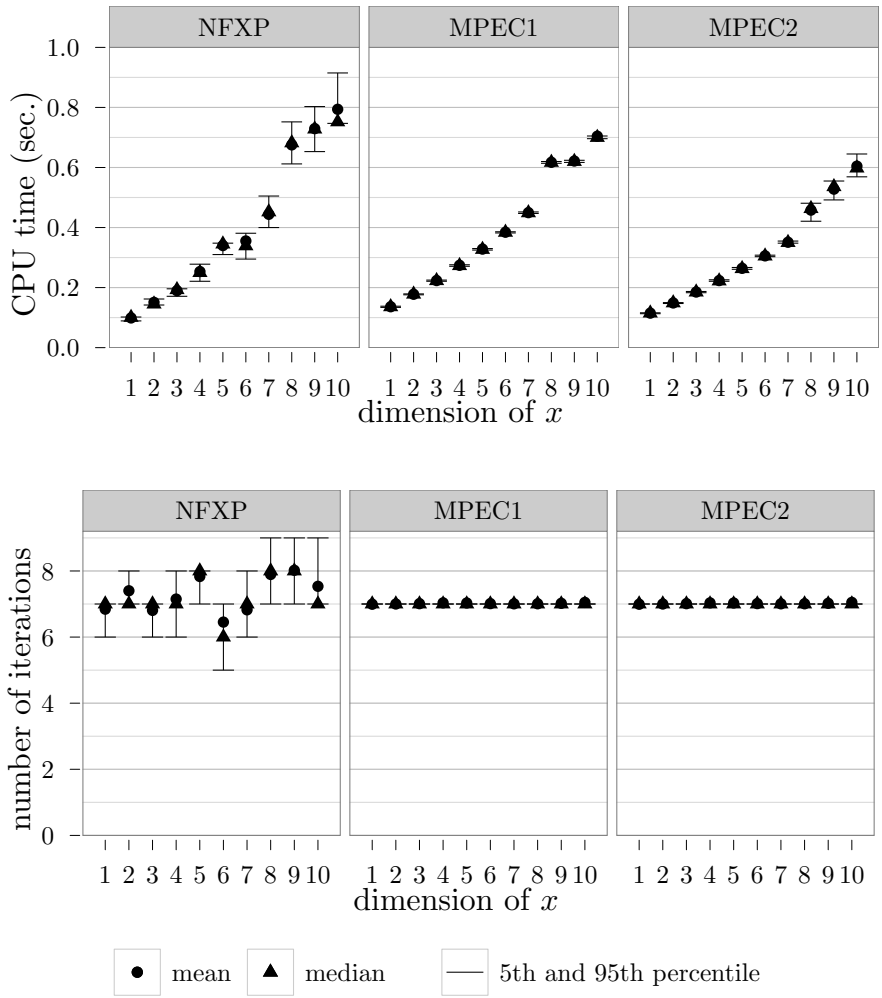

Figure 1.7 – Comparison of CPU time in seconds and number of iterations for NFXP, MPEC1 and MPEC2

Figure 1.7 shows the results. There is no clear relation between the number of iterations and different values of K. When we look at the CPU time spent to estimate the model, we see that NFXP increases quickest, followed by MPEC1 and MPEC2. NFXP and MPEC1

approximate the gradient with respect to the structural parameters by finite differences. To approximate the gradient with respect to one of the structural parameters, θ_j, we re-simulate all paths using the slightly changed value $\theta_j + h$. In the case of NFXP, we obtain new estimates for κ, calculate the value of the objective function and compare this with the value of the objective function in the original parameter vector. For MPEC1, we calculate the value of constraints using the newly simulated data, and compare this with the value of the constraints in the original structural parameter vector. We have to do this for each of the structural parameters in θ.

MPEC1 is faster than NFXP, even when we look at the CPU time spent in each iteration. I suspect this is because we need to solve $Z'Z\kappa = Z'y$ in every step of the gradient approximation for NFXP, but for MPEC1, we only calculate $Z'Z\kappa - Z'y$, without solving the system. MPEC2 takes less time than the other two methods, because we don't have to re-simulate the data in order to approximate the gradient with respect to the structural parameters. Analytic gradients are calculated instead.

Increasing the condition number of $Z'Z$

In the fourth experiment we want to see the effect of increased correlation between the covariates. We have one variable $x_{i,t} \sim$ i.i.d. $N(0, 1)$, and include powers of x_{it} in our structural (and auxiliary) model. For instance, when the maximum power is 2, we include x_{it} and x_{it}^2 in the structural model. The true coefficient on x_{it} is 1 in all cases. The coefficients on the powers of x are all 0. The average condition numbers of the matrix $Z'Z$ are 2.46, 2.53, 43.50, and 276.61 for powers 1 to 4 respectively. From figure 1.8 we see that the number of iterations increases a lot when the condition number increases. MPEC needs more iterations than NFXP, but NFXP spends more CPU time, which is due to the fact that NFXP needs more function evaluations during the line search to find a good enough update step for the parameters. Both MPEC1 and MPEC2 do not converge within 100 iterations for 49 cases out of 10,000 when we include x^4.

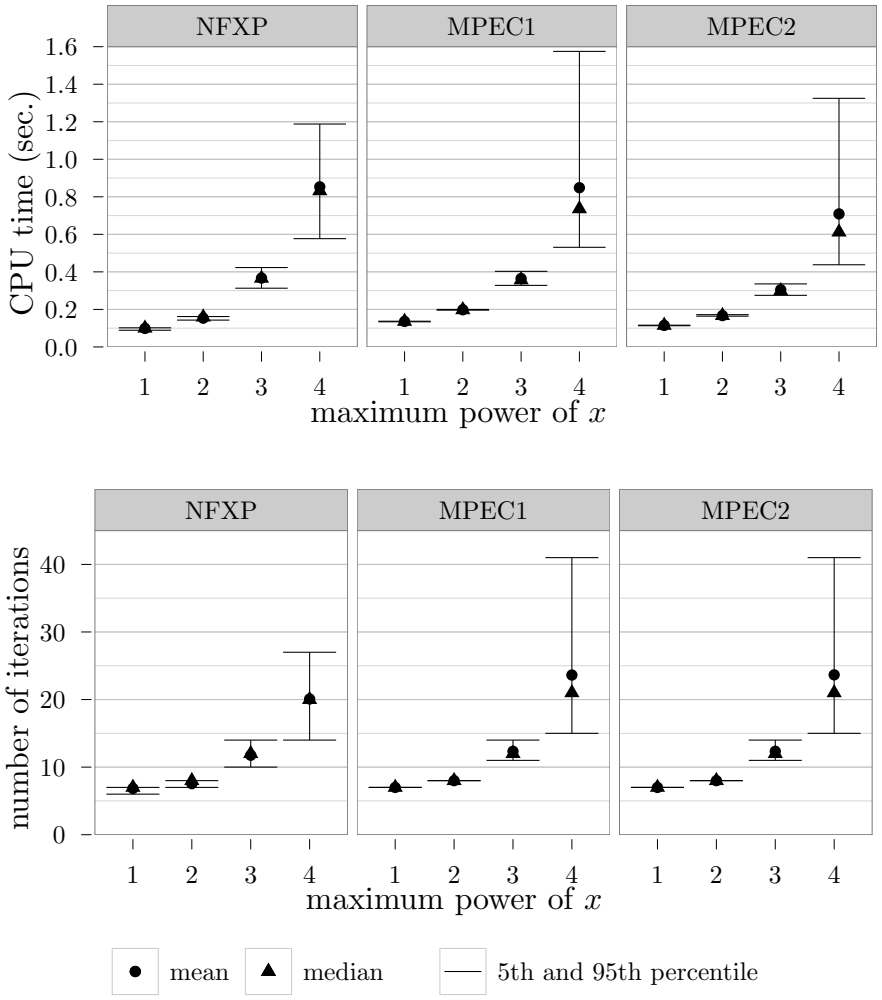

Figure 1.8 – Comparison of CPU time in seconds and number of iterations for NFXP, MPEC1 and MPEC2

1.5 Example 2: Dynamic probit model

In this section I use the first model presented in Keane and Smith (2004) as an example. This is a dynamic probit model with serially correlated errors where individual $i, i = 1, \ldots N$ chooses between two alternatives in periods $t = 1, \ldots, T$. They define a latent utility, u_{it}, that is attached to choosing the first alternative[11]

$$u_{it} = \gamma_0 + \gamma_1 x_{it} + \varepsilon_{it},$$

[11] Keane and Smith (2004) don't include the intercept γ_0.

where x_{it} is a scalar. The other alternative has utility 0. The errors are serially correlated, for $t > 1$,

$$\varepsilon_{it} = \rho\varepsilon_{it-1} + \eta_{it}.$$

The exogenous variables $x_{i,t}$ are drawn independently from the standard normal distribution. The innovation to the disturbances, $\eta_{i,t}$, are i.i.d. $N(0,1)$, and independent of $x_{i,t}$. We do not observe the latent utilities, but instead observe the chosen alternative

$$y_{i,t} = \begin{cases} 1 & \text{if} \quad u_{i,t} > 0 \\ 0 & \text{otherwise.} \end{cases}$$

The object of interest is a vector of structural parameters, $\theta = (\gamma_0, \gamma_1, \rho)$. These parameters can be estimated by indirect inference, by simulating values for the endogenous variables, $y_{i,t}$, given the exogenous variables, $x_{i,t}$, according to the structural model above, and using these to approximate the binding function between the structural model and the auxiliary model. To capture the auto-correlation in $\varepsilon_{i,t}$, lagged values of $y_{i,t}$ are included as covariates in the auxiliary model.

For a given value of the structural parameters θ, we have to simulate outcomes, using $\varepsilon^h_{i,t}$ and $u^h_{i,t}$. The simulated choice $y^h_{i,t}$ is a non-continuous function of the structural parameters θ, which translates into a non-smooth objective function for the NFXP formulation. For the MPEC formulation, the objective function does not depend on the structural parameters, so this function is smooth. The constraints are a function of the structural parameters, resulting in non-smooth constraints for MPEC. Keane and Smith (2004) use a smooth version of the simulated choice, to get a smooth objective function

$$y^h_{i,t} = \frac{1}{1 + e^{-\frac{u^h_{i,t}}{\lambda}}},$$

where λ is smoothing parameter. The same smoothing function is used to smooth the constraints in the MPEC formulation. Smoothing the simulated variables results in bias, because the auxiliary model estimated on the observed data and the auxiliary model estimated on the simulated data are not the same. The smoothing procedure adds non-classical measurement error to the auxiliary model estimated on the simulated data, result-

ing in biased estimates for the structural parameters. As λ goes to 0, the transformation becomes closer to the step function, and the bias goes to 0.

In a first step Keane and Smith (2004) find estimates for the structural parameters using a small number of simulation of $H = 10$, and a large value for the smoothing parameter, $\lambda = .03$. To improve on this estimate, they use a large number of simulations, $H = 300$ and a smaller value for the smoothing parameter, $\lambda = .003$, and take one additional Newton step from the previous estimate. The smaller value of λ reduces the bias introduced by smoothing. A larger number of simulations reduces the effect of simulation error in the standard errors and it makes the objective function more smooth.

In the MPEC formulation, increasing the number of simulations doesn't smooth the constraints, but adds additional constraints. This can cause problems if the score of the simulated likelihood is very non-smooth. A way to circumvent this problem is to use MPEC in the first step, where we have a high value of the smoothing parameter. The additional Newton step can then be taken in the same way as proposed by Keane and Smith (2004). This forces the algorithm to walk along the feasible set.

Keane and Smith (2004) propose different sets of auxiliary models, all of them based on the linear probability model $y_{i,t} = z_{i,t}\alpha_t + \nu_{i,t}$. For the simplest example $z_{i,t} = (1, x_{i,t}, y_{i,t-1})$ and $\alpha_t = \alpha_s \forall t, s$. The initial value for $y_{i,0}$, is defined to be 0. As long as the definition for the unobserved $y_{i,0}$ is the same for simulated and observed data, putting an arbitrary value will not affect the results. This is the first auxiliary model that I use below, which has four auxiliary parameters; an intercept, coefficients for $x_{i,t}$ and $y_{i,t-1}$, and the variance of the residuals of the linear model, σ^2_ν.

The second auxiliary model has different coefficients α_t for different time periods, and different variances of the residuals, making the total number of auxiliary parameters 19; 2 parameters for the first period, where there is no lagged $y_{i,t}$, 3 parameters for each of the subsequent periods, and 1 parameter for the variance of the residuals of the linear model in all periods. For the third auxiliary model, I include additional lags of x and y, when they are observed. For instance, at $t = 3$, I include $x_{t-1}, x_{t-2}, y_{t-1}$ and y_{t-2}. This model has 35 auxiliary parameters.

I include three additional auxiliary models, based on a Probit auxiliary model, $y^*_{i,t} =$

$z_{i,t}\alpha_t + \nu_{i,t}$, where $\nu_{i,t} \sim N(0,1)$, and $y_{i,t} = 1$ if $y^*_{i,t} > 0$, and 0 otherwise. The same three variations as above are used, except the Probit model does not estimate the variance of $\nu_{i,t}$ as auxiliary parameter. This auxiliary model is expected to better capture the non-linearity in the structural model leading to a reduction in the standard errors of the estimated structural parameters.

1.5.1 Results

The estimations in this section are based on 2,500 different Monte Carlo datasets where $N = 1000$ and $T = 5$. All of the results use $H = 20$ as the number of simulations. The true value of both γ_0 and γ_1 is set to 1. Instead of taking only one Newton step for the smaller value of λ, the optimization problem is solved until convergence for all λ. Estimation results for the first auxiliary model are shown in table 1.3, with different values for ρ and different values for the smoothing parameter, λ. Results for the two sets of auxiliary models are shown next to each other; the linear probability model (LPM) and Probit.

When looking at the first columns of table 1.3, we see that for LPM there is substantial bias in the parameter estimates for γ_0, γ_1 and ρ for $\lambda = 0.03$. The bias is worse when there is more auto-correlation, ρ is higher. The bias in the parameter estimates decreases when we use less smoothing, and there is virtually no bias if $\lambda = 0.003$. The rightmost columns show that for Probit, there is substantially less bias in the parameter estimates for all values of λ.

I ran the experiments above using both NFXP and MPEC1. The initial values for θ are set to 0. A comparison of the performance of the two methods is shown in table 1.4, where the median number of iterations and CPU time in seconds are given. Not all optimizations completed within 100 iterations. Those optimizations were not included in the calculations of the coefficient estimates, and for the calculation of the number of iterations and CPU time. The percentage of failed optimizations is shown in the third and sixth column.

From the table we see that for the linear probability model, MPEC1 takes slightly more iterations, and slightly more CPU time to find a solution. The number of seconds

Table 1.3 – Comparing different values of λ for dynamic probit estimation

		LPM			Probit		
	λ	γ_0	γ_1	ρ	γ_0	γ_1	ρ
$\rho = 0$							
	0.03	0.954	0.947	0.001	1.003	1.002	0.001
		(0.025)	(0.029)	(0.060)	(0.027)	(0.031)	(0.064)
	0.01	0.985	0.982	0.001	1.001	1.001	0.002
		(0.026)	(0.031)	(0.065)	(0.027)	(0.031)	(0.065)
	0.003	0.997	0.995	0.001	1.002	1.001	0.002
		(0.027)	(0.031)	(0.066)	(0.030)	(0.034)	(0.068)
$\rho = 0.4$							
	0.03	0.946	0.941	0.363	1.003	1.002	0.394
		(0.036)	(0.036)	(0.061)	(0.040)	(0.040)	(0.065)
	0.01	0.984	0.982	0.388	1.003	1.003	0.400
		(0.039)	(0.039)	(0.065)	(0.042)	(0.042)	(0.067)
	0.003	0.998	0.997	0.398	1.004	1.004	0.402
		(0.042)	(0.043)	(0.068)	(0.045)	(0.044)	(0.069)
$\rho = 0.85$							
	0.03	0.907	0.905	0.773	0.990	0.988	0.830
		(0.071)	(0.064)	(0.065)	(0.081)	(0.074)	(0.067)
	0.01	0.970	0.968	0.822	1.000	0.998	0.843
		(0.080)	(0.073)	(0.068)	(0.082)	(0.074)	(0.067)
	0.003	0.996	0.994	0.842	1.005	1.003	0.848
		(0.083)	(0.076)	(0.069)	(0.085)	(0.076)	(0.068)

Mean estimates are based on 2,500 replications of simulated datasets with $N = 1000$ individuals and $T = 5$ periods, standard deviations in parentheses. $H = 20$ simulated paths were used for each individual to approximate the binding function. Values of $\gamma_0 = 1$ and $\gamma_1 = 1$ were used to simulate the data. Results are based on auxiliary model 1.

per iteration is also slightly higher, which is in line with what we saw in the previous section. When there is little smoothing, λ is small, we see from the number of failed optimizations that both NFXP and MPEC1 have more trouble finding an optimum.

For the probit auxiliary model, there is a larger difference between NFXP and MPEC1. MPEC1 uses more iterations to find an optimum, almost twice as many when $\lambda = .003$. However, the number of seconds per iteration is much smaller for MPEC1 than for NFXP. The difference arises because with NFXP, $H = 20$ probit estimations have to be performed in every step. For MPEC1 instead of finding estimates for the probit model, the algorithm only checks whether the first-order conditions are satisfied for the current set of auxiliary parameters. This difference will become larger if the number of structural parameters

Table 1.4 – Performance of NFXP versus MPEC1 for different λ

		LPM			Probit		
	λ	num. iter.	CPU time	% failed	num. iter.	CPU time	% failed
NFXP							
	0.03	12	0.41	0.12	11	60.31	1.04
	0.01	13	0.45	0.97	12	63.90	0.82
	0.003	15	0.61	1.52	14	74.77	1.07
MPEC1							
	0.03	13	0.53	0.11	14	2.76	1.16
	0.01	15	0.60	1.01	18	3.33	1.77
	0.003	19	0.87	1.72	28	5.24	2.03

Median number of iterations, median CPU time and percentage of failed optimzations are based on 2,500 replications of simulated datasets. An optimization failed if the problem did not converge within 100 iterations or if the exit code of the optimization procedure showed another failure. Results are obtained with auxiliary model 1.

increases, because the gradient is calculated using finite differences. If H increases, the number of auxiliary parameters for MPEC1 will increase, possibly leading to more iterations that are needed to converge. For NFXP, every additional simulation set, implies an additional probit estimation that needs to be solved in every step.

Keane and Smith (2004) focus in their simulations on the effect of an auxiliary model not capturing all the dynamic features of the data generating process. In their case, for the dynamic probit model, the different auxiliary models increasingly manage to capture better the features of the dynamics. The same effect can be observed in table 1.5 from the decreasing standard error on ρ, going from auxiliary model 2, to model 3. There is an even larger improvement in the standard errors going from model 1 to 2. As described above, auxiliary model 2 introduces time-specific coefficients in the auxiliary model. These time-specific coefficients are important in this model, because the distribution of the errors is non-stationary. When we compare LPM and Probit, we see that the standard errors are slightly smaller for the Probit auxiliary model.

In table 1.6 we compare the performance for the different auxiliary models. The results for auxiliary model 1, are the same as those for $\lambda = 0.003$ in table 1.4. The Probit auxiliary models 1 and 2 were not estimated for NFXP, because the expected time this would take,

Table 1.5 – Comparing three auxiliary models for dynamic probit estimation

		LPM			Probit		
	aux	γ_0	γ_1	ρ	γ_0	γ_1	ρ
$\rho = 0$							
	1	0.997	0.995	0.001	1.002	1.001	0.002
		(0.027)	(0.031)	(0.066)	(0.030)	(0.034)	(0.068)
	2	0.994	0.992	0.000	0.997	0.995	0.002
		(0.027)	(0.031)	(0.048)	(0.027)	(0.031)	(0.047)
	3	0.991	0.989	0.000	0.993	0.991	0.001
		(0.027)	(0.031)	(0.040)	(0.027)	(0.031)	(0.040)
$\rho = 0.4$							
	1	0.998	0.997	0.398	1.004	1.004	0.402
		(0.042)	(0.043)	(0.068)	(0.045)	(0.044)	(0.069)
	2	0.993	0.992	0.394	0.997	0.996	0.397
		(0.035)	(0.036)	(0.043)	(0.034)	(0.035)	(0.042)
	3	0.990	0.989	0.395	0.993	0.992	0.398
		(0.033)	(0.034)	(0.035)	(0.033)	(0.032)	(0.033)
$\rho = 0.85$							
	1	0.996	0.994	0.842	1.005	1.003	0.848
		(0.083)	(0.076)	(0.069)	(0.085)	(0.076)	(0.068)
	2	0.993	0.992	0.845	0.998	0.996	0.848
		(0.049)	(0.047)	(0.030)	(0.048)	(0.044)	(0.029)
	3	0.989	0.991	0.845	0.995	0.992	0.846
		(0.046)	(0.041)	(0.024)	(0.045)	(0.038)	(0.023)

Mean estimates are based on 2,500 replications of simulated datasets with $N = 1000$ individuals and $T = 5$ periods, standard deviations in parentheses. $H = 20$ simulated paths were used for each individual to approximate the binding function. Values of $\gamma_0 = 1$ and $\gamma_1 = 1$ were used to simulate the data. Auxiliary models 1, 2 and 3 for LPM, use 4, 19 and 35 auxiliary parameters respectively. Results are shown for $\lambda = 0.003$

would be too long. For the LPM auxiliary models, increasing the number of auxiliary parameters from 4 to 19 to 35 in models 1, 2 and 3, results in an increase in CPU time. A larger number of auxiliary parameters means that a larger system of equations has to be solved to approximate the binding function. For MPEC1, the relative difference in CPU time for the different auxiliary models is smaller for the LPM models, than for the Probit models. The Probit models show a larger increase in CPU time when the number of auxiliary parameters is increased. Finally, there are no obvious differences in the number of iterations or the percentage of failed optimizations.

Table 1.6 – Performance of NFXP vs. MPEC1 for 3 auxiliary models

		LPM			Probit		
	aux	num. iter.	CPU time	% failed	num. iter.	CPU time	% failed
NFXP							
	1	15	0.61	1.52	14	74.77	1.07
	2	14	0.86	0.00			
	3	15	1.23	0.00			
MPEC1							
	1	19	0.87	1.72	28	5.24	2.03
	2	22	1.39	0.39	24	32.03	0.72
	3	24	1.72	0.69	26	34.86	2.00

Median number of iterations, median CPU time and percentage of failed optimzations are based on 2,500 replications of simulated datasets. An optimization failed if the problem did not converge within 100 iterations or if the exit code of the optimization procedure showed another failure. Results are shown for $\lambda = 0.003$.

1.6 Example 3: Binary time series

In the final example I look at different ways to approximate the binding function. I approximate the binding function as before, where auxiliary parameters are estimated from H simulated datasets. The mean of these H sets of auxiliary parameters is then used as the binding function, which corrects for any finite sample bias. In addition, I approximate the integral defining the binding function using sparse grid integration. This results in a binding function that does not correct for finite sample bias.

The model is a binary time series

$$\begin{aligned} y_t^* &= \phi y_{t-1} + x_t\gamma + \varepsilon_t \\ \varepsilon_t &= \rho\varepsilon_{t-1} + \eta_t, \end{aligned}$$

where $\eta_t \sim$ i.i.d. $N(0, \sigma_\eta^2)$. We observe $y_t = 1$, if $y_t^* > 0$, and $y_t = 0$ otherwise. The variance of the stationary distribution of ε_t is normalized to 1, which implies a value of $\sigma_\eta^2 = 1 - \rho^2$ for the variance of the innovation. We assume that the initial shock is drawn from the stationary distribution for the AR process for ε_t, $\varepsilon_0 \sim N(0, 1)$. The structural parameter is defined as $\theta = (\gamma, \rho)$. The likelihood of observing a specific sequence of

discrete outcomes, is defined by a high-dimensional integral, because we have to integrate over all possible latent values in the previous periods, which makes this model difficult to estimate. Applying indirect inference reduces the dimension of this integral at the cost of some loss in efficiency.

To capture the auto-correlation in ε_t and the state-dependence in y_t, I add L lagged values of y_t to the linear regression that is used as an auxiliary model. I estimate two versions of the model below. In one version we fix ϕ to 0. In this model we need $L \geq 1$ to identify ρ. In the other version, ϕ is a parameter to be estimated, and we require $L \geq 2$.

The binding function from a linear auxiliary model, excluding the first L observations, is

$$\begin{aligned} H(\theta, \mu \mid Z) = 0 \quad &\Leftrightarrow \quad E_{F_\theta}(G(Y, Z, \hat{\mu}) \mid Z = z) = 0 \\ &\Leftrightarrow \quad \frac{1}{T-(L-1)} \sum_{t=L}^{T} E_{F_\theta}(Z_t(Y_t - Z_t'\hat{\mu}) \mid Z_t = z_t) = 0 \\ &\Leftrightarrow \quad \frac{1}{T-(L-1)} \sum_{t=L}^{T} z_t z_t'\hat{\mu} - \frac{1}{T-(L-1)} \sum_{t=L}^{T} z_t E_{F_\theta}(Y_t \mid Z_t = z_t) = 0, \end{aligned}$$

where Z_t contains $X_t, \ldots, X_{t-L}$ and $y_{t-1}, \ldots, y_{t-L}$. The auxiliary parameters from the linear model can be interpreted as the elements of a Markov transition matrix. To find an expression for the binding function, we calculate the expectation of Y conditional on Z

$$\begin{aligned} E(y_t \mid Z_t = z_t) &= P(y_t^* > 0 \mid x_t, \ldots, x_{t-L}, y_{t-1}, \ldots y_{t-L}) \\ &= \frac{P(y_t^* > 0, y_{t-1}, \ldots, y_{t-L} \mid x_t, \ldots, x_{t-L})}{P(y_{t-1}, \ldots, y_{t-L} \mid x_{t-1}, \ldots, x_{t-L})} \\ &= \frac{\int_{-\infty}^{-x_t\gamma} \int_{a_{t-1}}^{b_{t-1}} \cdots \int_{a_{t-L}}^{b_{t-L}} f_{L+1}(\varepsilon_t, \ldots, \varepsilon_{t-L}) d\varepsilon_{t-L} \ldots d\varepsilon_t}{\int_{a_{t-1}}^{b_{t-1}} \cdots \int_{a_{t-L}}^{b_{t-L}} f_L(\varepsilon_{t-1}, \ldots, \varepsilon_{t-L}) d\varepsilon_{t-L} \ldots d\varepsilon_{t-1}}, \end{aligned}$$

where a_s, b_s are the upper and lower bounds of integration for time period $s = t-1, \ldots, t-L$. These bounds depend on the value of y_s that we observe in the data. If $y_s = 0$, then the integration runs from $a_s = -\infty$ to $b_s = -\phi y_{s-1} - x_s\gamma$, and if $y_s = 1$, then integration runs from $a_s = -\phi y_{s-1} - x_s\gamma$ to $b_s = \infty$. For $s = t - L$, the lagged value y_{s-1} is not known, because we do not condition on this value. In that case we substitute $y_{s-1} = .5$.

The function $f_{L+1}(\varepsilon_t, \ldots, \varepsilon_{t-L})$ is the joint density of $L + 1$ variables. The auto-

regressive structure for ε implies that we can write

$$\begin{pmatrix} \varepsilon_t \\ \vdots \\ \varepsilon_{t-(L-1)} \\ \varepsilon_{t-L} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & \dots & \rho^{L-1} & \rho^L \\ & \ddots & & \vdots \\ & & 1 & \rho \\ 0 & & & 1 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} \sigma_\eta & & & 0 \\ & \ddots & & \\ & & \sigma_\eta & \\ 0 & & & 1 \end{pmatrix}}_{D} \begin{pmatrix} \eta_t \\ \vdots \\ \eta_{t-(L-1)} \\ \varepsilon_{t-L} \end{pmatrix}.$$

Without conditioning on periods before $t-L$, ε_{t-L} follows a standard normal distribution. Since the η_t's are independently distributed from each other and from ε_{t-L}, $(\varepsilon_t, \dots, \varepsilon_{t-L})$ follows a multivariate normal distribution with variance-covariance matrix $\Sigma = ADD'A'$. A similar procedure can be used if the error follows a moving average process.

Applying indirect inference reduces the problem from evaluating a T dimensional integral in the final period, a $T-1$ dimensional integral in the period before, etc., to evaluating $L+1$ dimensional integrals in every period. This reduction in the dimension of the integral is obtained by conditioning only on the previous L periods, instead of the complete history. This will result in a loss of efficiency of the estimator. On the other hand, it also results in a numerically more accurate estimator, since an $L+1$ dimensional integral can be evaluated at higher accuracy than a T dimensional integral.

There are different options to evaluate the multivariate normal probabilities defined above. Genz (2004) has formulas for 2 and 3 dimensional multivariate normals. For higher dimensions one option is to use simulation, as proposed by Hajivassiliou, McFadden, and Ruud (1996). I use a combination of sparse grid integration (Heiss & Winschel, 2008) and Genz (1992) who shows a transformation simplifying the integral. After applying the transformation in Genz (1992), the multivariate normal probabilities can be evaluated as an integral with bounds of integration from 0 to 1. The same transformation also reduces the dimension of integration by 1; to evaluate the $L+1$ dimensional probability, an L dimensional integration problem has to be solved. Genz (1992) uses random draws from a uniform distribution to evaluate the integral. I use a sparse grid based on single-dimension Gauss-Legendre nodes.

Results

Table 1.7 shows the estimation results using a linear probability model from a binary time series of length, $T = 1000$, where we have three parameters; an intercept, γ_0, which takes the value 1, the coefficient on a covariate x_t, γ_1, which takes the value 1, and the auto-correlation parameter, ρ, which takes on values $0, .3, .6, .7, .8, .9$ and $.95$. The covariate x_t is i.i.d. $N(0, 1)$. The average of the estimated coefficients, based on 1000 Monte Carlo replications, are given in the table, with the standard deviation in brackets below. The number of lags of y_t that are used in the auxiliary model is denoted by L.

Table 1.7 – Comparing different L for binary time series with LPM as auxiliary model

		$\rho = 0$	$\rho = 0.3$	$\rho = 0.6$	$\rho = 0.7$	$\rho = 0.8$	$\rho = 0.9$	$\rho = 0.95$
$L = 1$								
	γ_0	1.003	1.002	1.002	1.002	1.001	1.007	1.016
		(0.061)	(0.069)	(0.087)	(0.099)	(0.120)	(0.166)	(0.235)
	γ_1	1.003	1.003	1.004	1.004	1.005	1.012	1.023
		(0.071)	(0.072)	(0.074)	(0.076)	(0.083)	(0.102)	(0.126)
	ρ	-0.006	0.294	0.598	0.697	0.796	0.895	0.941
		(0.087)	(0.081)	(0.071)	(0.065)	(0.061)	(0.052)	(0.046)
$L = 2$								
	γ_0	1.004	1.004	1.009	1.011	1.011	1.017	1.030
		(0.061)	(0.070)	(0.093)	(0.108)	(0.134)	(0.183)	(0.244)
	γ_1	1.004	1.005	1.011	1.013	1.015	1.020	1.034
		(0.071)	(0.073)	(0.079)	(0.083)	(0.094)	(0.115)	(0.134)
	ρ	-0.006	0.293	0.598	0.698	0.798	0.897	0.942
		(0.088)	(0.077)	(0.061)	(0.055)	(0.048)	(0.038)	(0.034)
$L = 3$								
	γ_0	1.004	1.004	1.010	1.013	1.014	1.022	1.034
		(0.061)	(0.070)	(0.095)	(0.112)	(0.140)	(0.193)	(0.255)
	γ_1	1.004	1.005	1.012	1.015	1.017	1.026	1.036
		(0.071)	(0.073)	(0.081)	(0.087)	(0.099)	(0.122)	(0.144)
	ρ	-0.006	0.293	0.598	0.698	0.797	0.898	0.946
		(0.088)	(0.077)	(0.058)	(0.051)	(0.043)	(0.032)	(0.027)

Mean estimates are based on 1,000 replications of simulated datasets with $T = 1000$ periods, standard deviations in parentheses. L denotes the number of lags of y_t in the auxiliary model. Values of $\gamma_0 = 1$ and $\gamma_1 = 1$ were used to simulate the data. Sparse grid integration was used to approximate the binding function.

The approximation of the binding function used to obtain the estimates in this table does not correct for finite sample bias. From the table it looks like there is no apparent bias in any of the parameter estimates, except perhaps for large values of ρ. For values of ρ close to 1, a shock in one period has a long-lasting effect on the subsequent periods. A

large value for ρ has a similar effect as having a small number of observations, T, in the time series, since more observations are needed to get an accurate estimate for ρ in that case. The slight bias in ρ when its true value is close to 1, could therefore be interpreted as a small amount of finite sample bias. On the other hand, the standard errors on γ_0 and γ_1 are larger for large values of ρ, which means that the apparent bias could go away if we estimate the same model on a larger number of Monte Carlo simulations.

Increasing the number of lags in the auxiliary model adds information about the auto-correlation, resulting in more precise estimates for ρ. This comes at the cost of a loss of precision in the other two structural parameters.

Instead of using a linear probability model, we can again use a Probit estimation as auxiliary model. The results of this experiment are shown in table 1.8. Again, we observe no bias in the estimates, except possibly for values of ρ close to 1. One benefit of using Probit as an auxiliary model seems to be that it has smaller standard errors[12]. Also, if we increase the number of lags in the auxiliary model, the precision of ρ increases, without a decrease in the precision of the other two parameters.

Table 1.9 compares the performance of the different variations. Estimating a Probit instead of an LPM as auxiliary model does not lead to a large increase in computing time. Increasing the number of lags is more computationally intensive, since we have to approximate a higher dimensional integral. For $L = 1$, a 1-dimensional (sparse) grid using 5 nodes is used to approximate the integral. For $L = 2$ and $L = 3$, a sparse grid used more nodes than a grid defined by the product rule (see table 1.1), so I use the product rule. This implies that every increase in L results in a 5-fold (the number of nodes in a single dimension) increase in the number of nodes that we use to evaluate the function. The timings increase by slightly more than a factor 5 if L is increased.

Instead of using a product rule grid to obtain integration nodes, I could have used a random grid of points, in effect using Monte Carlo simulation to approximate the integral. In that case the number of integration nodes can be chosen to be any number, where lower numbers of integration nodes lead to slightly higher standard errors, but quicker

[12]Besides the efficiency of an estimator, its robustness to misspecification may also be a consideration when deciding which auxiliary model should be used. To the best of my knowledge, no comparison has been made between LPM and Probit auxiliary models, but a general discussion of the subject is given in section 2.5 of Jiang and Turnbull (2004)

Table 1.8 – Comparing different L for binary time series with Probit as auxiliary model

		$\rho = 0$	$\rho = 0.3$	$\rho = 0.6$	$\rho = 0.7$	$\rho = 0.8$	$\rho = 0.9$	$\rho = 0.95$
$L = 1$								
	γ_0	1.003	1.001	1.001	1.001	0.999	1.005	1.017
		(0.060)	(0.067)	(0.085)	(0.097)	(0.118)	(0.165)	(0.235)
	γ_1	1.003	1.003	1.003	1.003	1.003	1.010	1.023
		(0.069)	(0.070)	(0.072)	(0.073)	(0.078)	(0.098)	(0.124)
	ρ	-0.006	0.294	0.598	0.697	0.796	0.895	0.941
		(0.086)	(0.080)	(0.070)	(0.064)	(0.059)	(0.050)	(0.043)
$L = 2$								
	γ_0	1.003	1.003	1.006	1.006	1.004	1.011	1.027
		(0.060)	(0.068)	(0.085)	(0.098)	(0.119)	(0.170)	(0.246)
	γ_1	1.004	1.005	1.009	1.009	1.009	1.016	1.033
		(0.069)	(0.070)	(0.072)	(0.073)	(0.078)	(0.100)	(0.131)
	ρ	-0.006	0.293	0.597	0.696	0.796	0.896	0.944
		(0.087)	(0.076)	(0.059)	(0.052)	(0.044)	(0.032)	(0.029)
$L = 3$								
	γ_0	1.003	1.003	1.006	1.006	1.004	1.007	1.022
		(0.060)	(0.068)	(0.085)	(0.098)	(0.119)	(0.169)	(0.240)
	γ_1	1.004	1.005	1.009	1.009	1.010	1.015	1.028
		(0.069)	(0.070)	(0.072)	(0.073)	(0.078)	(0.100)	(0.128)
	ρ	-0.006	0.293	0.597	0.696	0.796	0.896	0.946
		(0.088)	(0.076)	(0.056)	(0.048)	(0.039)	(0.027)	(0.023)

Mean estimates are based on 1,000 replications of simulated datasets with $T = 1000$ periods, standard deviations in parentheses. L denotes the number of lags of y_t in the auxiliary model. Values of $\gamma_0 = 1$ and $\gamma_1 = 1$ were used to simulate the data. Sparse grid integration was used to approximate the binding function.

computations. This way of using simulation to approximate the binding function has the benefit of providing a gradient that is continuous in the parameters, whereas the method shown previously needs some form of smoothing to obtain a continuous gradient. This comes at the cost of not having the finite sample correction property that the other method has.

In a second set of experiments, I estimate the same model as above, but with ϕ included, and its true value set to 0.5. Table 1.10 shows the results of 1000 Monte Carlo simulations, using an LPM or Probit as auxiliary model. The number of lags in the auxiliary model is set 2, 3, or 4. The parameter ϕ is not very precisely estimated, and compared with the previous model, the precision of ρ is also smaller. A similar improvement of efficiency as in the previous model can be seen, going from LPM to Probit.

Table 1.9 – Performance of LPM and Probit

	LPM			Probit		
L	num. iter.	CPU time	% failed	num. iter.	CPU time	% failed
1	9	0.14	0.21	8	0.20	0.08
2	9	0.93	0.89	8	1.08	0.26
3	9	6.42	0.57	8	7.29	0.24

Median number of iterations, median CPU time and percentage of failed optimzations are based on 1,000 replications of simulated datasets. An optimization failed if the problem did not converge within 200 iterations or if the exit code of the optimization procedure showed another failure.

Table 1.10 – Comparing different L for binary time series with state dependence

		LPM				Probit			
	ρ	ϕ	γ_0	γ_1	ρ	ϕ	γ_0	γ_1	ρ
$L=2$									
	0	0.479	1.018	1.004	0.004	0.478	1.020	1.005	0.005
		(0.273)	(0.239)	(0.080)	(0.199)	(0.273)	(0.240)	(0.079)	(0.198)
	0.3	0.521	0.985	1.008	0.276	0.524	0.983	1.008	0.275
		(0.272)	(0.221)	(0.083)	(0.180)	(0.271)	(0.220)	(0.080)	(0.179)
	0.6	0.497	0.995	1.006	0.596	0.499	0.995	1.007	0.594
		(0.187)	(0.155)	(0.093)	(0.094)	(0.188)	(0.155)	(0.088)	(0.095)
	0.9	0.495	0.978	1.004	0.887	0.504	0.991	1.019	0.889
		(0.159)	(0.187)	(0.135)	(0.043)	(0.153)	(0.187)	(0.123)	(0.042)
$L=3$									
	0	0.481	1.018	1.004	0.003	0.480	1.020	1.005	0.005
		(0.275)	(0.238)	(0.080)	(0.198)	(0.274)	(0.238)	(0.079)	(0.197)
	0.3	0.524	0.985	1.008	0.275	0.526	0.983	1.008	0.274
		(0.268)	(0.217)	(0.083)	(0.176)	(0.268)	(0.217)	(0.080)	(0.176)
	0.6	0.500	1.002	1.009	0.595	0.499	1.000	1.007	0.593
		(0.181)	(0.152)	(0.094)	(0.089)	(0.184)	(0.153)	(0.088)	(0.093)
	0.9	0.504	0.995	1.022	0.892	0.497	0.995	1.015	0.890
		(0.159)	(0.200)	(0.158)	(0.039)	(0.140)	(0.185)	(0.121)	(0.035)
$L=4$									
	0	0.480	1.019	1.005	0.004	0.479	1.021	1.005	0.004
		(0.273)	(0.237)	(0.082)	(0.197)	(0.273)	(0.237)	(0.079)	(0.196)
	0.3	0.523	0.987	1.009	0.276	0.527	0.983	1.009	0.274
		(0.266)	(0.217)	(0.084)	(0.175)	(0.266)	(0.216)	(0.080)	(0.175)
	0.6	0.503	1.004	1.010	0.595	0.503	1.004	1.010	0.594
		(0.180)	(0.152)	(0.097)	(0.088)	(0.178)	(0.149)	(0.088)	(0.087)
	0.9	0.514	1.012	1.036	0.892	0.506	1.005	1.027	0.890
		(0.167)	(0.207)	(0.168)	(0.037)	(0.152)	(0.196)	(0.148)	(0.034)

Mean estimates are based on 1,000 replications of simulated datasets with $T = 1000$ periods, standard deviations in parentheses. L denotes the number of lags of y_t in the auxiliary model. Values of $\gamma_0 = 1$, $\gamma_1 = 1$, and $\phi = 0.5$ were used to simulate the data. Sparse grid integration was used to approximate the binding function. LPM or Probit are used as auxiliary models.

Table 1.11 – Comparing different L for binary time series using NFXP

		$H = 10, \lambda = 0.03$				$H = 300, \lambda = 0.003$			
	ρ	ϕ	γ_0	γ_1	ρ	ϕ	γ_0	γ_1	ρ
$L = 2$									
	0	0.506	1.000	1.006	-0.002	0.482	1.020	1.005	0.009
		(0.272)	(0.235)	(0.088)	(0.186)	(0.255)	(0.223)	(0.083)	(0.183)
	0.3	0.521	0.989	1.007	0.280	0.509	0.998	1.007	0.294
		(0.262)	(0.220)	(0.092)	(0.166)	(0.234)	(0.198)	(0.087)	(0.157)
	0.6	0.504	1.009	1.011	0.588	0.516	0.997	1.012	0.595
		(0.209)	(0.166)	(0.102)	(0.107)	(0.200)	(0.160)	(0.097)	(0.102)
	0.9	0.511	1.004	1.016	0.887	0.506	0.997	1.001	0.890
		(0.198)	(0.194)	(0.150)	(0.058)	(0.166)	(0.176)	(0.132)	(0.056)
$L = 3$									
	0	0.503	1.003	1.007	0.001	0.478	1.024	1.007	0.015
		(0.268)	(0.231)	(0.090)	(0.182)	(0.242)	(0.211)	(0.083)	(0.172)
	0.3	0.518	0.993	1.008	0.284	0.503	1.004	1.007	0.299
		(0.241)	(0.201)	(0.095)	(0.152)	(0.224)	(0.190)	(0.082)	(0.145)
	0.6	0.504	1.008	1.011	0.588	0.513	1.000	1.012	0.596
		(0.201)	(0.161)	(0.108)	(0.095)	(0.184)	(0.150)	(0.090)	(0.090)
	0.9	0.502	1.004	1.010	0.889	0.497	0.999	0.998	0.891
		(0.174)	(0.190)	(0.136)	(0.051)	(0.158)	(0.174)	(0.134)	(0.049)
$L = 4$									
	0	0.501	1.004	1.006	0.002	0.478	1.024	1.006	0.015
		(0.256)	(0.222)	(0.091)	(0.173)	(0.232)	(0.203)	(0.082)	(0.164)
	0.3	0.511	0.999	1.008	0.290	0.508	1.004	1.011	0.298
		(0.225)	(0.194)	(0.096)	(0.142)	(0.237)	(0.186)	(0.097)	(0.148)
	0.6	0.508	1.008	1.013	0.588	0.510	0.999	1.007	0.596
		(0.199)	(0.161)	(0.111)	(0.095)	(0.181)	(0.148)	(0.094)	(0.088)
	0.9	0.500	1.007	1.010	0.892	0.497	1.000	0.999	0.891
		(0.171)	(0.191)	(0.138)	(0.047)	(0.163)	(0.174)	(0.135)	(0.046)

Mean estimates are based on 1,000 replications of simulated datasets with $T = 1000$ periods, standard deviations in parentheses. L denotes the number of lags of y_t in the auxiliary model. Values of $\gamma_0 = 1$, $\gamma_1 = 1$, and $\phi = 0.5$ were used to simulate the data. $H = 10$ or $H = 300$ simulated paths were used to approximate the binding function. LPM is used as auxiliary model.

Table 1.11 shows estimations of the same model where I use the NFXP formulation combined with simulations to approximate the binding function. For both values of λ no apparent bias is present in the parameter estimates. Increasing the number of simulations improves the standard errors of the estimates.

The performance for the LPM auxiliary model, using sparse grid integration or simulation is shown in table 1.12. In terms of speed, using a small number of simulations and a large value for the smoothing parameter, the second column, is by far the quickest. When simulation is used to approximate the binding function, only a small increase in

computing time is observed, when L is increased. This can be explained by the similarity of the problems that are being solved. The only difference is in the number of auxiliary parameters. This difference is small, as increasing L by one, increases the auxiliary parameters by one as well. From the right set of columns, we also see that a small value for the smoothing parameter leads to convergence issues. The maximum number of iterations is set to 200, and 10% of the Monte Carlo simulations fail to converge in that case.

For the sparse grid integration we see much larger differences in CPU time when increasing L, which is due to the increasing number of integration nodes used, as explained above. We see that the relative increase from including 3 to 4 lags, versus 2 to 3 lags, is smaller. For $L = 4$, the benefit of sparse grid integration becomes visible, because the number of nodes used in SGI is smaller compared to the product rule (385 to 625).

Table 1.12 – Comparing performance of SGI versus simulation

	SGI			$H = 10, \lambda = .03$			$H = 300, \lambda = .003$		
L	num. iter.	CPU time	% failed	num. iter.	CPU time	% failed	num. iter.	CPU time	% failed
2	15	1.78	0.65	19	0.14	0.12	37	10.96	8.32
3	15	12.50	0.55	20	0.17	0.12	38	14.10	10.97
4	15	54.98	2.23	20	0.21	0.15	39	17.88	10.60

Median number of iterations, median CPU time and percentage of failed optimzations are based on 1,000 replications of simulated datasets. An optimization failed if the problem did not converge within 200 iterations or if the exit code of the optimization procedure showed another failure.

1.7 Conclusion

This paper presents an MPEC formulation to indirect inference estimation. The MPEC formulation simplifies introduction of analytic derivatives, which is shown to reduce computing time relative to the NFXP formulation if the number of structural parameters grows. The MPEC formulation combined with simulation introduces many new variables and constraints, but because of the sparseness structure of the problem, using a large number of simulations, $H = 5000$, did not present a significant decrease in performance for MPEC.

When using a Probit model as auxiliary model, for which no closed-form solution exists, MPEC is an order of magnitude faster than NFXP. This is an encouraging result for cases

where the preferred auxiliary model has to be estimated using a recursive optimization procedure.

Finally, sparse grid integration can prove useful as an alternative to approximating the binding function by simulation. Especially when the model contains discrete outcomes, the non-smoothness introduced by simulation results in convergence problems, which can be mitigated if a sparse grid approximation to the binding function is used. In that case the binding function does not correct for finite sample bias. For the model and the parameter values that were presented in section 1.6 this appeared to be no problem, but this does not hold in general.

None of the methods presented here consistently outperforms the other methods in all cases. Which formulation should be used depends on the model at hand, and the requirements on speed and robustness. The results described here should serve as guidance as to which method is most likely to perform best in which case.

2

Wage dynamics with labour participation

2.1 Introduction

Many studies have looked at the evolution of wages of individuals over time or over the life-cycle. A large number of papers decompose the residuals of a wage regression in a permanent and a transitory component. These papers look at wages for males that are working and ignore selection into work. Including work status in the model is important for two reasons. First, instead of wages, we are typically interested in income. Wages contribute to a large part of income for individuals that work, but if we do not model labour participation, then we can not say anything about individuals that do not have wages. Non-participation is more common among females and the analysis is therefore usually restricted to males. Second, from the literature on static models of labour supply,

The data used in this chapter were made available through the ESRC Data Archive. The data were originally collected by the ESRC Research Centre on Micro-social Change at the University of Essex now incorporated within the Institute for Social and Economic Research. Neither the original collectors of the data nor the Archive bear any responsibility for the analyses or interpretations presented here.

it is known that the selection in to work is non-random; more productive individuals are more likely to work than less productive individuals. Not correcting for this non-random selection leads to biased estimates in these models. In this paper I combine these different approaches, by adding a selection equation to a simple model of wage dynamics to account for non-random selection. I then estimate this model on wage and labour participation data from the United Kingdom for males and females and assess the amount of bias.

The data show a high level of persistence in log-wages and in participation status. One of the main findings of this paper is that the persistence in log-wages can not fully explain the persistence in labour participation. When I control for additional persistence in participation independent from log-wages, by either including lagged participation or by including a persistent unobservable that only affects participation, the non-random selection into work becomes very small. This is opposite to the conclusion that is reached by most papers with static models of labour participation and holds for both males and females and for all education groups. This also implies that the bias in the parameter estimates is small in those cases when non-participation is not included in the model.

Since wages are an important part of household income, the statistical process underlying wages and earnings has been used to analyse inequality and social mobility. Lillard and Willis (1978) estimate an earnings function using U.S. panel data on male log earnings and then decompose the residual from this regression in a permanent and a transitory component. Assuming normality for the transitory and persistent shock, they use this model to look at persistence in poverty status, where an arbitrary threshold in earnings is used to define poverty. MaCurdy (1982) specifies a more general model for the structure of the unobservables. In addition to earnings, Abowd and Card (1989) analyse the covariance structure of earnings and hours for males with positive values for both.

Gottschalk and Moffitt (1994) and Moffitt and Gottschalk (1995)[1] used a similar decomposition to look at changes in parameters of the underlying stochastic process over time and conclude that the increase in the variance of earnings for men in the U.S. in the 1970s and 1980s can be attributed to equal increases in the permanent and the transitory

[1]This working paper has been around for a long time and cited many times, and as explained in Jenkins and Lambert (2011) has recently been published as Moffitt and Gottschalk (2011). I'll refer to the paper as Moffitt and Gottschalk (1995).

part. This research has subsequently been repeated many times with different data sets, for different countries, and using different stochastic processes for the decomposed unobservable (some examples include Haider, 2001; Baker & Solon, 2003; Meghir & Pistaferri, 2004). Relevant papers for the UK, the country that I study in this paper, are Dickens (2000), Ramos (2003), and Blundell and Etheridge (2010).

Instead of looking at time-varying parameters, Browning, Ejrnæs, and Alvarez (2010) take a slightly different approach. They start with a standard ARMA model, and introduce some additional parameters. More importantly, they specify a joint distribution for a subset of the parameters to allow for additional individual heterogeneity. They then estimate this model on data from the Panel Study of Income Dynamics (PSID). The heterogeneity in the model parameters turns out to be important in their case. In this paper I do not include as much heterogeneity in the parameters, but I find that heterogeneity in the level of earnings and especially heterogeneity in the propensity to work is important.

In addition to inequality, another reason to look at wages, is to get a sense of uncertainty. Most dynamic economic models of household decisions contain uncertainty over future wages, and the decomposition of wages or earnings gives some idea of uncertainty. These decompositions can not be directly interpreted as uncertainty, because the fluctuations in earnings might be perfectly forecasted by the agents, even though the econometrician does not observe this. To characterize how much of the earnings is forecasted by the agent, and how much is uncertainty, a model relating current economic decisions to expectations about the future has to be formulated, see for instance (Blundell & Preston, 1998) and Blundell, Pistaferri, and Preston (2008). This is beyond the scope of the current paper.

All of the wage dynamics studies mentioned above have in common that they focus on males with positive wages. In this paper I add a selection equation to a simple model for wage dynamics and compare this to the same model without correcting for selection. From static models of labour participation it is known that selection in to work is non-random and this non-random selection causes bias in estimation results if the selection process is ignored (Heckman, 1979; Blundell, Reed, & Stoker, 2003). Bias is not only present in estimates of the levels, but also in estimates of variance parameters. In this paper I

estimate a model for wage dynamics, where I assess the bias of the variance parameters resulting from ignoring selection if selection into work is non-random.

From the data we see that labour participation status is very persistent; individuals that are working in one period have a high probability of working in the next period. The estimation results show that this persistence in participation status can not be explained by persistence in wages alone. An additional source of persistence is required to fit the pattern of persistence in labour participation. Inluding a random effect in the participation equation improves the match between the empirical work status transition probabilities and the transition probabilities predicted by the model. Allowing for state dependence in the participation equation further improves the fit. At the same time, once we control for persistence in participation heterogeneity, non-random selection into work does not seem to be an important problem.

There are two papers that I know of, that include discrete choices in a wage dynamics model. Altonji, Smith, and Vidangos (2009) estimate a very rich model of wages, hours, job mobility and participation using indirect inference. Their estimation procedure is computationally very expensive. They do not allow for changing parameters over time, and a comparison of their model with models of earnings dynamics that do not include hours, job mobility and participation, but that do allow parameters to change over time can not be made directly.

Low, Meghir, and Pistaferri (2010) estimate parameters for a wage process as an input to their structural model of consumption, labour supply and job mobility. The process for wages in their paper contains additional features to this paper; most notably a random walk in the persistent part of earnings, and dynamics related to job changes. They correct for non-random selection into work by including an inverse Mills ratio for employment in the moment equations of the difference in log-wages. The inverse Mills ratio is obtained from a first stage probit regression of employment status. Importantly, the error in this first stage selection equation is independent over time. The error can be correlated with the innovation to the persistent unobservable of wages, with the transitory shock, or with the job innovation, but not with the persistent part of the unobservable. Consequently, two individuals with the same values for the shocks, but with a different level of the

cumulated persistent unobservable, will have the same probability of participating. The results in my paper suggest that persistence in the unobservables are an important part of the selection process.

The remainder of the paper is organised as follows. In the next section I will describe the data that I use. Then I will describe the model, the estimation procedure and the results. The final section concludes.

2.2 Patterns in the BHPS

To estimate the model in this paper, I use data from the British Household Panel Survey (BHPS). The BHPS was introduced in 1991 and since then a representative sample of the United Kingdom has been surveyed annually. The first wave of the panel consists of approximately 5,500 households and 10,300 individuals. In later waves, additional samples of households have been added to the main sample.

For this study, I use 16 years of data, from 1991 to 2006. The wage data has been converted to 2008 prices. To minimize the effect of outliers, we remove the top and bottom percent from the non-zero wage data, by gender and education group. These observations are completely removed from the data, resulting in a mssing value for both participation and wages for the individual for that year.

Self-employed individuals are removed from the data, because it is difficult to get a measure for hourly wages for these individuals. In total this leads to a reduction of about 10% of the number of observations, where the reduction is larger for males (16%) than for females (5%). If an individual was self-employed in one year, and works as an employee in the next, then this individual is only included in the sample when she works as an employee. If an individual is self-employed for some periods, and not working in other periods, the individual is included only for those periods where she is not working. This inflates the proportion of individuals that are not working in two ways; the total number of individuals in the sample is smaller than the number of individuals in the total population, because self-employed individuals are dropped, which makes the denominator smaller. And, individuals are included in the sample if they are non-working, but not when they are self-employed. This understates the proportion of individuals that are working.

In principle an additional layer could be added to the model to account for selection into self-employment, similar to the way non-participation is included currently, but due to the difficulty in getting an accurate measure for self-employed wages I do not explore this approach here.

Observations of individuals between age 25, when most individuals have finished education, and the legal retirement age, 60 for women and 65 for men, are included in the sample. The models will be estimated separately by education level and gender. Since the education status changes during the survey for about 10% of the individuals, everyone is classified into education groups based on the lowest education level that we observed for them. For instance, if we observe someone with O-Levels for three years, and then A-Levels for two years, this individual is coded as having O-Levels for the entire sample.

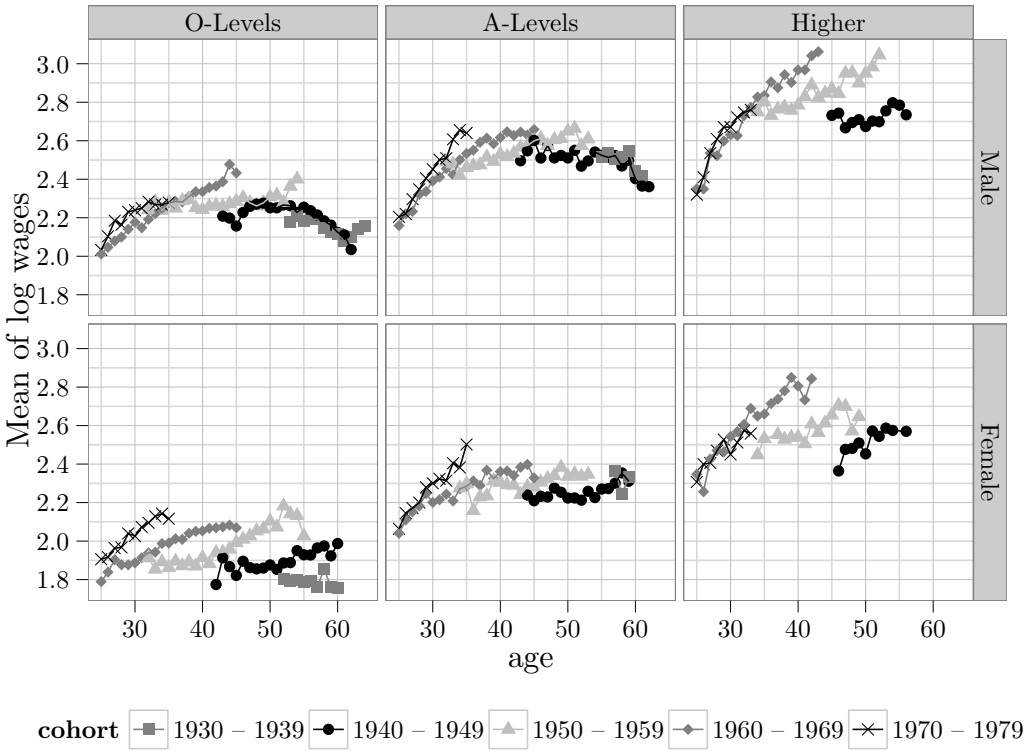

Figure 2.1 – Mean log-wages by cohort, education group and gender

Figure 2.1 shows the mean of log-wages by age for males and females for different education groups. Each panel shows the mean for different cohorts. Individuals that are not working at a certain age, are not used in the calculation of the mean for that

age. For the figures below, values are only shown when they are based on at least 25 observations. For example, mean log-wages is not shown for the high education group for cohort 1930–1939, because there are too few observations.

The wages for the different cohorts follow roughly the same inverted-U shaped pattern over the life-cycle. For males there are hardly any changes in the level between the cohorts. For females the differences between cohorts are more pronounced. The level of mean wages at a given age differ by cohort, which can be explained by an increase in the mean wage over time, which can be seen in figure 2.2. It is not quite clear whether the shape of the life-cycle pattern of wages is the same for different cohorts for women, especially for women with education up to O-Levels.

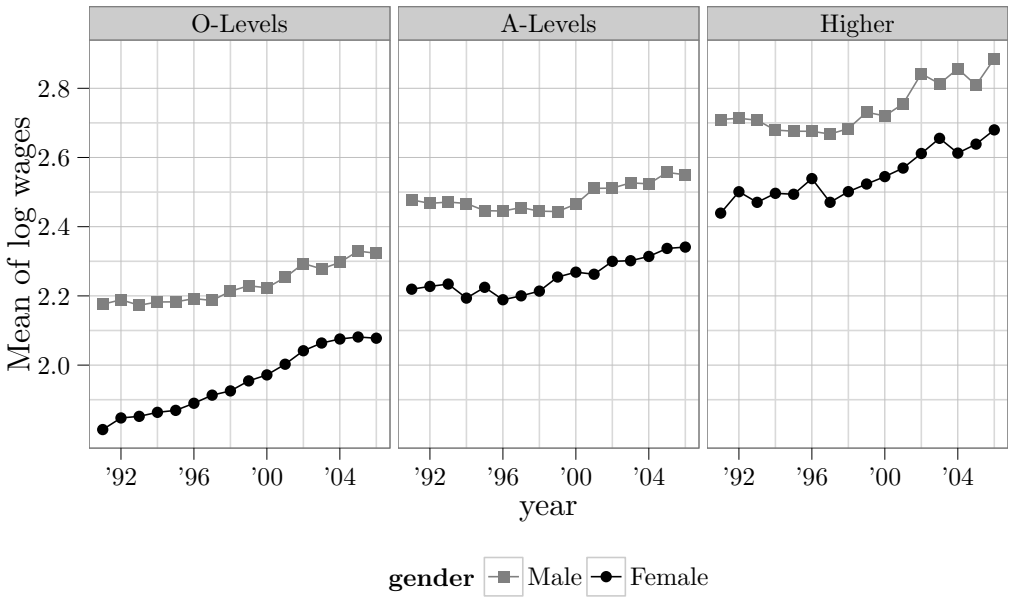

Figure 2.2 – Mean log-wages by education group and gender

Figure 2.2 shows the mean of log-wages by year. In a given year the wages for individuals from different cohorts are not directly comparable, because the composition of the groups are different; the respective cohorts are at different points in the life-cycle. Wages for the different cohorts are pooled in this figure to mask age and cohort effects. Both from figure 2.1 and from figure 2.2 we see that for most of their career wages of working males with A-Levels are on average about 25% higher than wages of working males with O-Levels. For males with higher education, this difference is about 60%. Figure 2.1 shows

that the wage-growth at young ages is steeper for individuals with more education. We see a similar pattern for the wages of females. For males there do not seem to be large differences in the level of wages between different cohort at the same age, except for the higher education group. For females the differences between cohorts seem to be larger.

Figure 2.2 shows that wages of males have been relatively constant during the 90's, even falling a bit, and then increased in later years. The wages of females are about 20% lower than those of males. The gap between male and female wages appears to have been constant over the sample period, with a small decrease in the gap for individuals with O-Levels. There are many different potential reasons for this gap, such as differences in industries and full-time or part-time jobs between men and women.

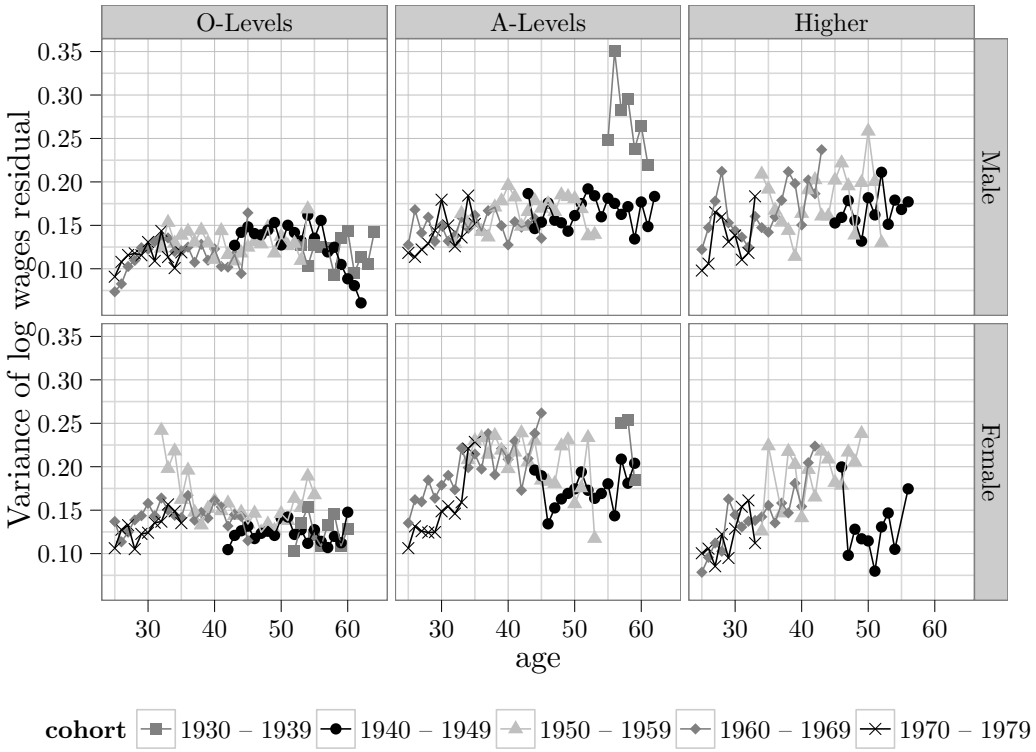

Figure 2.3 – Variance of log-wages residuals by cohort, education group and gender

To look at the variation of log-wages, I create log-wage residuals from a regression of log-wages on time dummies and age dummies. These regressions are estimated by education group, gender and cohort for figure 2.3 and by education group and gender for figure 2.4. These figures show the variance of the wage residuals by age or year. In

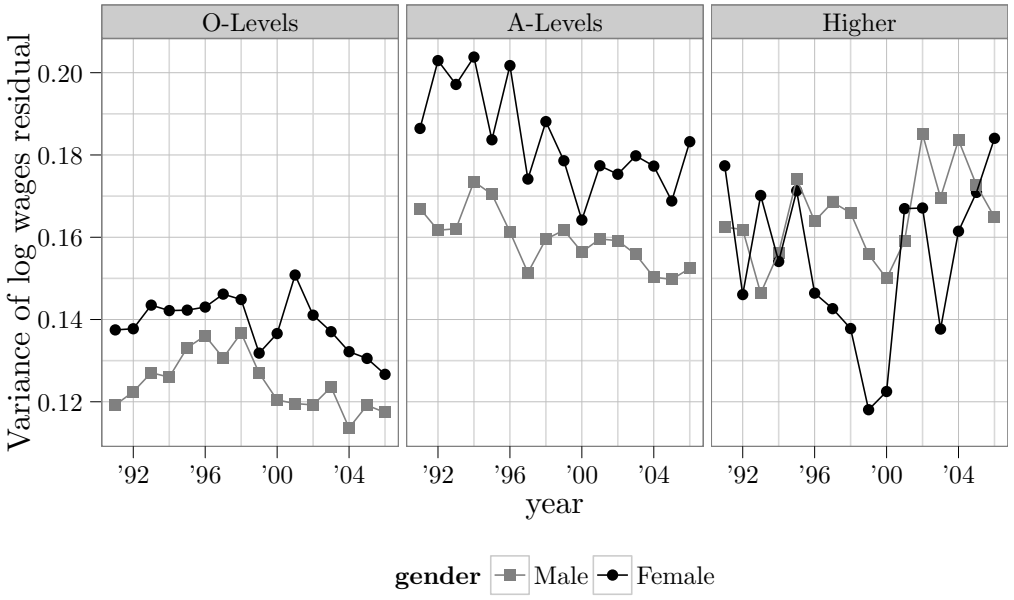

Figure 2.4 – Variance of log-wages residuals by education group and gender

most papers where earnings or wages are decomposed in separate parts, moments of these residuals, such as variances, variances of differences and auto-covariances, are used directly for the analysis.

We see that, depending on the education group, variances increase slightly with age. For females, the variance seems to decrease at later ages. Because of heterogeneity in earnings growth, or some accumulation of past shocks, the stochastic processes underlying most earnings dynamics models do not allow for a decreasing variance with age. Including non-participation and individuals non-randomly dropping out of work can explain a decrease in the variance. If at later ages individuals on one side of the earnings distribution are more likely to stop working, the sample of individuals that is observed to be working becomes more homogeneous, with a smaller variance in observed earnings.

To get an interpretation for these variances, we can compare the 90th percentile of wages with the 50th percentile of wages[2]. Using a variance of .15 for log-wages, an

[2]If wages follow a log-normal distribution, which seems to be a reasonable assumption from looking at histograms, we can write the random variable determining wages as $\text{wages}_i = \exp(\mu + \sigma\varepsilon_i)$. We can compare two percentiles in the distribution of wages, by using the corresponding z-score and the standard deviation of log-wages. For instance, approximating the percentage change in wages going from the 50th to the 90th percentile, we have $\frac{\exp(\mu+\sigma z(.9))-\exp(\mu+\sigma z(.5))}{\exp(\mu+\sigma z(.5))} = \exp(\sigma(z(.9) - z(.5))) - 1 \approx \sigma(z(.9) - z(.5))$, where I used that $exp(x) \approx 1 + x$ for small values of x. Plugging in values for $z(.5) = 0$, and $z(.9) \approx 1.3$, we get that the 90th percentile of wages is approximately $100 \cdot 1.3 \cdot \sigma$ percent larger than the 50th percentile

individual at the 90th percentile of the wage distribution has a wage approximately $100 \cdot 1.3 \cdot \sqrt{.15} \approx 50$ percent higher than an individual at the mean of the distribution.

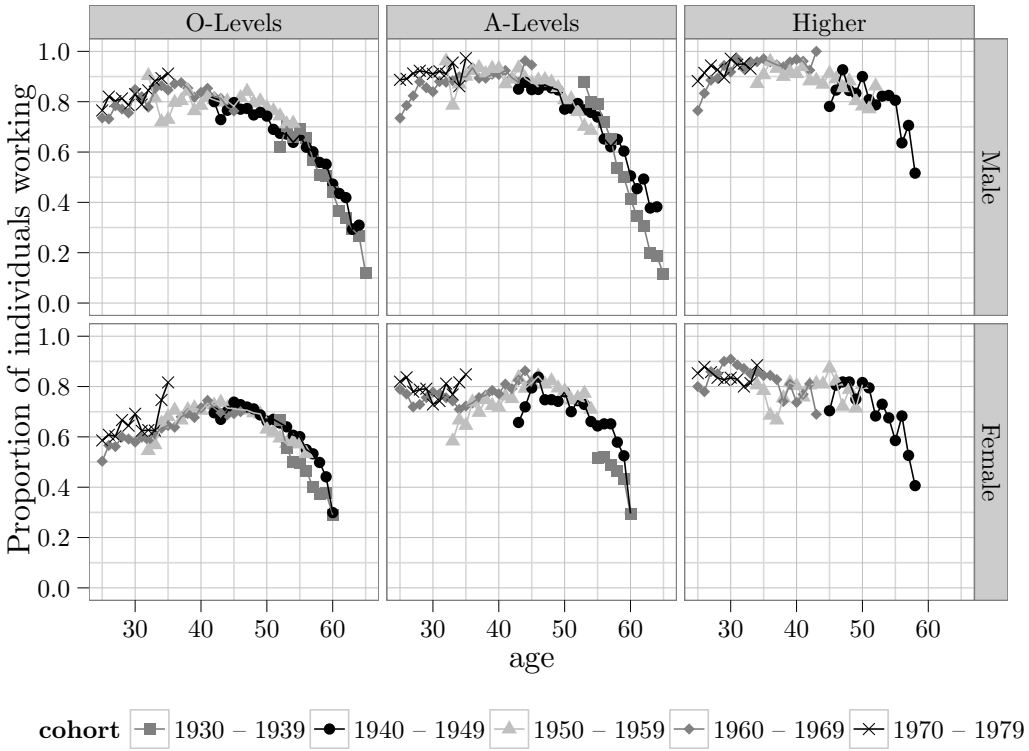

Figure 2.5 – Participation by cohort, education group and gender

Non-participation is important as we can see from figures 2.5 and 2.6. For both males and females there is a significant group of individuals that is not working. These proportions are not to be interpreted as unemployment rates, because of my definition of participation. As explained above, self-employed individuals are not included in the sample. Also, individuals that are not working are not nescessarliy looking for work.

The pattern of participation over the life-cycle that is the same for all cohorts. For men, the proportion of individuals that are working is roughly constant up to age 50, when the participation rate starts to decline. Individuals with higher levels of education are more likely to work, with over 90% of Higher educated individuals working, compared to 80% for individuals with O-Levels. Participation rates for women are lower than for men, and we see a slight drop in participation rates around age 30.

of wages, which is equal to the mean for the normal distribution.

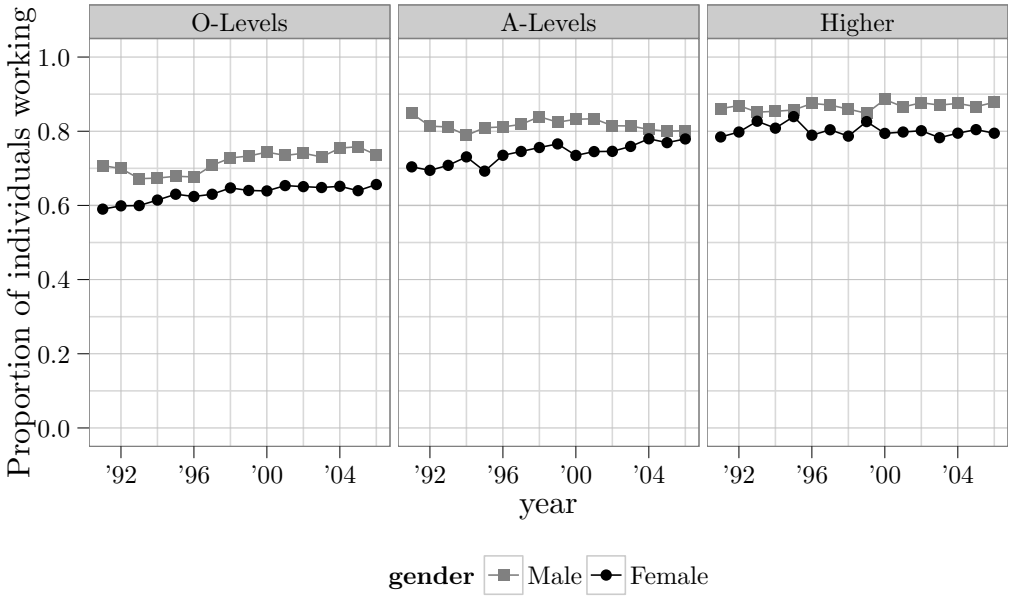

Figure 2.6 – Participation by education group and gender

Participation rates for men have been relatively constant over time, with a slight increase for men with O-Levels, as can be seen from figure 2.6. For females we see an increase in participation rates over the observed period for O-Levels and A-Levels.

Transition probabilities from work to work and for non-work to non-work are shown in figures 2.7 and 2.8. These figures show the proportion of individuals that are observed to stay in the same state from one period to the next. Only observations from individuals that are observed in two consecutive periods are included. Both of these figures tell us that work-status is very persistent from one period to the next. For both males and females, over 90% of individuals that are working in one period, are observed to be working in the next period. The transition probabilities from non-work to non-work are slightly lower for individuals with O-Levels, and significantly lower for higher educated individuals. Although these transition probabilities are not constant over the life-cycle, they appear to be constant over time.

Figure 2.7 – Transition probability by education group and gender

2.3 Model description

The goal of this paper is to develop and estimate a model for earnings dynamics that includes a labour participation decision. In this section I describe a simple permanent-transitory model for the dynamics of wages and labour participation. The model described in this section is estimated separately by education group and gender. I assume the following process for latent potential wages, in logs, for individual i, in year t

$$\ln y^*_{ith} = X_{it}\theta_1 + \pi_t \xi_{it} + \tau_t \varepsilon_{it}. \tag{2.3.1}$$

X_{it} contains a set of observable covariates, which in the estimation below consists a set of time dummies, and a set of age dummies. The component of wages that is not explained by observables, is decomposed in two parts; a permanent part, $\pi_t \xi_{it}$, and a transitory part, $\tau_t \varepsilon_{it}$. The variance of ξ_{it} is normalized to 1, which means that π_t can be interpreted as the standard deviation of the persistent part. The variance of wages that is explained by the

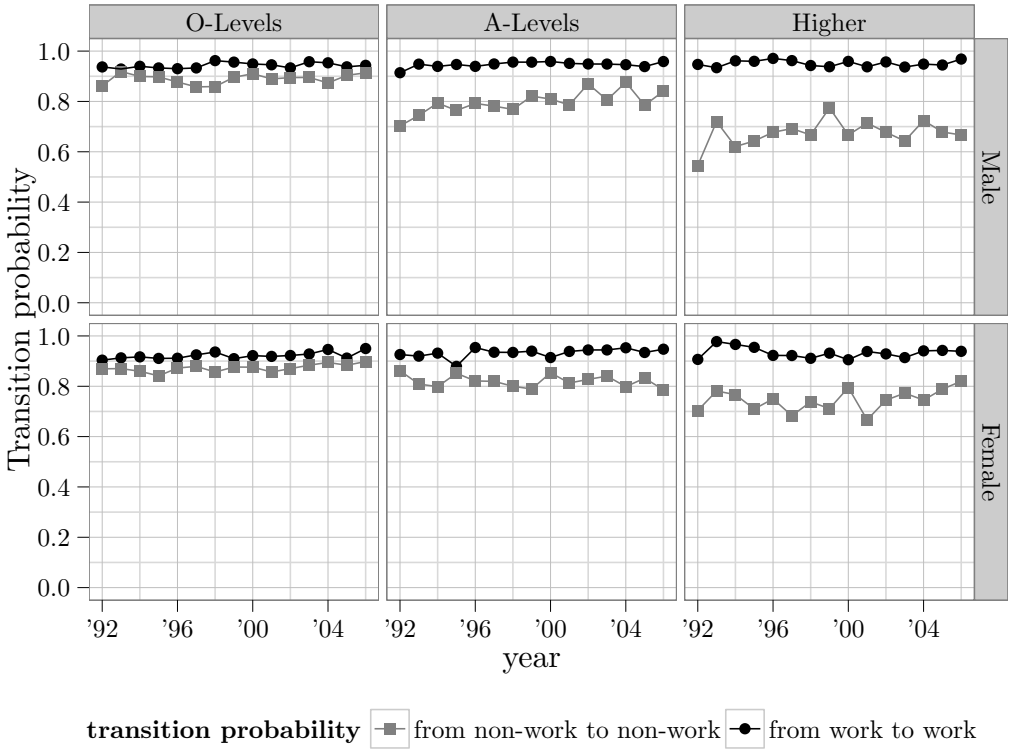

Figure 2.8 – Transition probability by education group and gender

permanent part can change over time, because π_t has a t subscript. The random draws ε_{it} are independent of ξ_{it} and independent over time and have unit variance as well. The contribution to the variance in log wages of this component may change over time through the parameter τ_t, which can be interpreted as the standard deviation of the transitory part. In some formulations of earnings dynamics models the transitory part is written just as ε_{it}, without τ_t. The standard deviation of ε_{it} would then be defined as σ_{ε_t}, i.e. with a subscript t, in order to have time-varying variance parameters for the transitory shock. This is equivalent to my formulation, where the variance of ε_{it} is normalized to one. The parameters π_t and τ_t can be interpreted as prices, or aggregate productivity shocks.

The permanent part is written as

$$\xi_{it} = \alpha_i + \sigma_\beta \beta_i W_{it}, \tag{2.3.2}$$

where the variable α_i is an individual effect, corresponding to individual i, and constant

over time. In the baseline models the second element in this formula, the part with W_{it}, is zero. This means that in the baseline model, the persistent part is constant over time. As a variation I also present results for a model where W_{it} is a scalar observable. In the first case W_{it} is equal to the age of individual i at time t, and in the second case W_{it} equals the square root of her age. In those variations β_i is an individual specific growth rate, i.e. we allow for a heterogeneous trend.

Any variable that is not included in X_{it}, but with an effect on log wages, enters one of the two unobservables. For example, if industry or region are not included in X_{it} and these variables are constant over time, they enter α_i. Changes in returns to working in a certain industry will increase the relative differences between latent wages, which will be reflected in an increase in π_t. On the other hand, an increase in the level of log-wages for all sectors will be captured by the time-dummies in X_{it}.

This is not the only possible way to decompose wages or earnings in a permanent and transitory part. Many variations have been introduced where the process for ξ_{it} or ε_{it} is more complex (see for instance Moffitt & Gottschalk, 1995, for some examples). One variation commonly made, assumes that ξ_{it} follows a random walk, and ε_{it} contains a bit of persistence, for instance by modelling it as an auto-regressive (AR) process, a moving-average (MA) process or a combination of both (e.g Meghir & Pistaferri, 2004; Blundell & Etheridge, 2010). Other authors prefer a heterogeneous trend over a random walk for the permanent component (Guvenen, 2007, 2009), or include both a heteregeneous trend and a random walk (Baker & Solon, 2003; Hryshko, 2012).

In this paper I choose to use a heterogeneous trend as one of the variations on the baseline model. I do not to include a random walk to keep estimation of the wage dynamics model tractable when I add a labour participation equation to the model. Including a random walk to the wage dynamics model with selection is not a straightforward extension to the model described here when it comes to estimation, as I will explain below. Allowing for some growth of the variance of wages with age could be important, given that we saw a slight increase in the variance of wages over the life-cycle in the BHPS data, and adding a heterogeneous trend to the model is the computationally less burdensome way to achieve that.

The random walk and heterogeneous trend specification imply different theoretical moments. Table 2.1 shows the theoretical moments for the variance and auto-covariance of ξ_{it} and the variance and auto-covariance for the change in ξ_{it}, $\Delta\xi_{it} = \xi_{it} - \xi_{it-1}$, where t in this table should be seen as age. The variance for ξ_{it} increases quadratically in age for the heterogeneous trend in t, whereas it increases linearly in age for the other two specifications. In the data we see no evidence for a quadratic increase in the variance of log wages. The moments for the growth in ξ_{it} as implied by the heterogeneous trend in the square-root of age go to 0 as age increases, which does not completely match the random walk model.

Table 2.1 – Moments for heterogeneous trend and random walk specifications

Model		Moments
Heterogeneous trend in t		
$\xi_{it} = \alpha_i + \sigma_\beta \beta_i t$	$E\left[\xi_{it}^2\right]$	$= \sigma_\alpha^2 + t^2\sigma_\beta^2$
	$E\left[\xi_{it}\xi_{it-1}\right]$	$= \sigma_\alpha^2 + t(t-1)\sigma_\beta^2$
	$E\left[\Delta\xi_{it}^2\right]$	$= \sigma_\beta^2$
	$E\left[\Delta\xi_{it}\Delta\xi_{it-1}\right]$	$= \sigma_\beta^2$
Heterogeneous trend in $\sqrt{t}$		
$\xi_{it} = \alpha_i + \sigma_\beta \beta_i \sqrt{t}$	$E\left[\xi_{it}^2\right]$	$= \sigma_\alpha^2 + t\sigma_\beta^2$
	$E\left[\xi_{it}\xi_{it-1}\right]$	$= \sigma_\alpha^2 + \sqrt{t(t-1)}\sigma_\beta^2$
	$E\left[\Delta\xi_{it}^2\right]$	$= \sigma_\beta^2\left(\sqrt{t} - \sqrt{t-1}\right)^2$
	$E\left[\Delta\xi_{it}\Delta\xi_{it-1}\right]$	$= \sigma_\beta^2\left(\sqrt{t} - \sqrt{t-1}\right)\left(\sqrt{t-1} - \sqrt{t-2}\right)$
Random walk		
$\xi_{it} = \alpha_i + u_{it}$	$E\left[\xi_{it}^2\right]$	$= \sigma_\alpha^2 + t\sigma_v^2$
$u_{it} = u_{it-1} + \sigma_v v_{it}$	$E\left[\xi_{it}\xi_{it-1}\right]$	$= \sigma_\alpha^2 + (t-1)\sigma_v^2$
	$E\left[\Delta\xi_{it}^2\right]$	$= \sigma_v^2$
	$E\left[\Delta\xi_{it}\Delta\xi_{it-1}\right]$	$= 0$

To calculate these moments, for simplicity, the correlation between α_i and β_i is assumed to be 0.

Similarly to the derivation of the moments for different specifications of the persistent part, we can derive moments for different specifications of the transitory part. Two of these specifications are given in table 2.2. The single period transitory shock is the specification that I use in this paper. By allowing for a moving average model with one lag, the additional parameter can be used to better fit the data. For instance, in the single period transitory shock, we have the following equality $E\left[\Delta\varepsilon_{it}^2\right] = 2E\left[\varepsilon_{it}^2\right]$. In the moving average model with one lag, this equality does not need to hold. In this paper I do not include

a moving average specification for the transitory part for the same reason that I do not include a random walk in the persistent part; introducing this type of dependence between two periods results in a computationally demanding estimation procedure when a labour participation equation is added to the model. In the section describing the results, I assess how this simplification affects the fit of the model.

Table 2.2 – Moments for different specifications for transitory part

Model	Moments	
Single period transitory shock		
$\varepsilon_{it} = \sigma_\eta \eta_{it}$	$E\left[\varepsilon_{it}^2\right]$	$= \sigma_\eta^2$
	$E\left[\varepsilon_{it}\varepsilon_{it-1}\right]$	$= 0$
	$E\left[\Delta\varepsilon_{it}^2\right]$	$= 2\sigma_\eta^2$
	$E\left[\Delta\varepsilon_{it}\Delta\varepsilon_{it-1}\right]$	$= -\sigma_\eta^2$
Moving average with one lag		
$\varepsilon_{it} = \theta_0\eta_{it} + \theta_1\eta_{it-1}$	$E\left[\varepsilon_{it}^2\right]$	$= \theta_0^2 + \theta_1^2$
	$E\left[\varepsilon_{it}\varepsilon_{it-1}\right]$	$= \theta_0\theta_1$
	$E\left[\Delta\varepsilon_{it}^2\right]$	$= 2 \cdot (\theta_0^2 - \theta_0\theta_1 + \theta_1^2)$
	$E\left[\Delta\varepsilon_{it}\Delta\varepsilon_{it-1}\right]$	$= 2\theta_1\theta_0 - \theta_0^2 - \theta_1^2$

2.3.1 Modeling labour participation

From the figures of the BHPS data on participation, we saw that non-participation is non-trivial. Latent wages are not observed for individuals that do not participate. Because ignoring potential non-random participation leads to biased estimates (e.g. Blundell et al., 2003), we want to include non-participants in the model. Below, I explicitly write down the decision process of the agents that leads to a dynamic version of the standard selection model by Heckman (1979). This facilitates explaining why the model may not capture some features that we observe in the data.

We assume that individuals work if their latent wage is larger than a reservation wage, $\ln w_{it}^*$, that we write as

$$\ln w_{it}^* = X_{it}\tilde{\theta}_{2,t} + Z_{it}\tilde{\delta}_t + \tilde{\sigma}_{\gamma,t}\gamma_i + \sigma_{\eta,t}\eta_{it}. \tag{2.3.3}$$

The reservation wage depends on the same observable variables that we included in the wage regression, X_{it}, some variables that affect the reservation wage, but not the wage

itself, Z_{it}, and an unobservable, which is decomposed in a permanent part γ_i, and a transitory part, η_{it}. Typically, variables that are thought to enter Z_{it} are household composition, assets or savings, non-labour income or your health status. Not all of these variables are observed, which means that they will enter the unobservable, either via γ_i or via η_{it}.

The decision to work depends on a comparision between the latent potential wage and the reservation wage. If the potential wage is higher than the reservation wage, the individual decides to work, and otherwise she does not work

$$d_{it} = \begin{cases} 1 & \text{if } d_{it}^* > 0 \\ 0 & \text{otherwise,} \end{cases}$$

where d_{it} is observed participation, and d_{it}^* is defined as the difference between the wage and the reservation wage

$$d_{it}^* = \ln y_{it}^* - \ln w_{it}^*. \tag{2.3.4}$$

After substituting our definition of the latent wage from (2.3.1) and the reservation wage from (2.3.3) in (2.3.4), we get

$$d_{it}^* = X_{it}(\theta_1 - \tilde{\theta}_{2,t}) - Z_{it}\tilde{\delta}_t + \pi_t \xi_{it} + \tau_t \varepsilon_{it} - \tilde{\sigma}_{\gamma,t}\gamma_i - \sigma_{\eta,t}\eta_{it}. \tag{2.3.5}$$

In this specification, all the coefficients related to the reservation wage depend on t. Since the participation outcome, d_{it}, is a discrete variable, for identification, we have to normalize the scale of the participation equation; multiplying all elements of the participation equation by the same amount does not change the outcome, d_{it}, that we observe. For instance, we can normalize all the $\sigma_{\eta,t}$ to one. Another way to achieve identification is by removing the time-subscript from the $\tilde{\theta}_{2,t}$ and $\tilde{\delta}_t$ and normalizing one of the $\sigma_{\eta,t}$ to one, e.g. only the $\sigma_{\eta,t}$ corresponding to the first period. To decrease the number of parameters that we need to estimate, I make a stronger assumption, by combining the two; i.e. removing the time subscripts of the coefficients and relating $\sigma_{\eta,t}$ to one parameter. First, I define

$$\sigma_{\eta,t} = -\frac{\sqrt{1-\rho^2}}{\rho}\tau_t,$$

and multiply all elements in (2.3.5) by $\frac{\rho}{\tau_t}$, resulting in

$$d_{it}^* = X_{it}\frac{\rho}{\tau_t}(\theta_1 - \tilde{\theta}_{2,t}) - Z_{it}\frac{\rho}{\tau_t}\tilde{\delta}_t + \frac{\rho}{\tau_t}\pi_t\xi_{it} + \rho\varepsilon_{it} - \frac{\rho}{\tau_t}\sigma_{\gamma,t}\gamma_i + \sqrt{1-\rho^2}\eta_{it}.$$

This transformation introduces a new variable ρ, which is constant over time, with $|\rho| < 1$. A value of $\rho = 0$, implies an infinite value for $\sigma_{\eta,t}$. In that case, the reservation wage of the individual will be plus or minus infinity, resulting in a decision to work that is independent from her wage. Also, defining the scale of $\sigma_{\eta,t}$ with respect to τ_t implies that we assume that the relative importance of ε_{it} and η_{it} for the participation decision stays constant over time. Since we are not interested in the values of the separate parts that contribute to the coefficients, we rename some of the parameters, by defining θ_2, δ and σ_γ as the combination of coefficients multiplying X_{it}, Z_{it} and γ_i to end up with

$$d_{it}^* = X_{it}\theta_2 + Z_{it}\delta + \frac{\rho}{\tau_t}\pi_t\xi_{it} + \rho\varepsilon_{it} + \sigma_\gamma\gamma_i + \sqrt{1-\rho^2}\eta_{it}.$$

This results in a convenient expression for the likelihood as can be seen below. Other normalizations can be chosen and whether the choice of a particular normalization matters for the estimation results is an empirical issue that is left for future research.

In one of the variations I include previous period participation, d_{it-1}, in the participation equation. Introducing state dependence complicates the estimation, as explained below. However, allowing for state dependence has been shown to be important in practice, for instance by Hyslop (1999) who estimates a dynamic model of female labour participation. One rationalization for adding this variable, is that previous period participation is part of the reservation wage, because if you start working you have to pay some fixed cost, such as buying a car or spending time to arrange child support. Individuals that were working in the previous period, already paid this fixed cost and their reservation wage is lower. We therefore expect that individuals that worked in the previous period are more likely to be working in this period, suggesting a positive coefficient on d_{it-1} in the participation equation.

Combining both the log-wage equation and the participation equation from above,

results in the following set of equations

$$\begin{aligned} \ln y_{it}^* &= X_{it}\theta_1 + \pi_t\xi_{it} + \tau_t\varepsilon_{it} \\ d_{it}^* &= X_{it}\theta_2 + Z_{it}\delta + \phi d_{it-1} + \tfrac{\rho}{\tau_t}\pi_t\xi_{it} + \sigma_\gamma\gamma_i + \rho\varepsilon_{it} + \sqrt{1-\rho^2}\eta_{it}. \end{aligned}$$

Here we see again that if $\rho = 0$, the unobservables from the latent wage do not enter the participation equation, and there is no non-random selection into work. In that case there will be no bias in the parameter estimates, when the model for log-wages is estimated using observed wages alone. The direction of the bias due to non-random selection for the level parameters is well-known. Depending on the sign of ρ parameters are biased upwards or downwards. In a simplified version of the model, where there is no heteregeneous growth in wages, and without covariates, the variance of latent wages is simply

$$\text{var}\left[y_{it}^*\right] = \pi_t^2 + \tau_t^2.$$

Similarly the auto-covariance of the latent wages is

$$\text{cov}\left[y_{it}^*, y_{it-1}^*\right] = \pi_t\pi_{t-1}.$$

Using variances and covariances of non-randomly selected observed wage data, will lead to downward biased estimates for the variance parameters, independent of whether the correlation between the unobservables in the selection and outcome equation is positive or negative

$$\text{var}\left[y_{it}^* \mid d_{it}^* > 0\right] \leq \text{var}\left[y_{it}^*\right] = \pi_t^2 + \tau_t^2.$$

The intuition is that, because individuals on one side of the distribution (either the high end or the low end, depending on the sign of ρ) are more likely to be unobserved, the variance of the distribution decreases. If we restrict our sample data to individuals for which we have positive earnings in multiple periods, then the truncation gets worse if there is correlation over time between the participation probabilities

$$\text{var}\left[y_{it}^* \;\middle|\; d_{it}^* > 0, d_{it-1}^* > 0\right] \leq \text{var}\left[y_{it}^* \mid d_{it}^* > 0\right] \leq \text{var}\left[y_{it}^*\right].$$

Also, the bias in estimating the underlying variance process for the latent wages is worse if the participation rates are lower. We therefore expect to see more bias for females and for lower education groups, because they have lower participation rates.

Any additional change to the participation equation should be rationalized by a change in the model for wages or the model for the reservation wage. For example, if we wanted the permanent and transitory component of wages to enter the selection equation with different coefficients, then we need to explain why we want to include one or the other in the reservation wage. One possibility would be to allow for correlation between γ_{it} and ξ_{it}. If these unobservables were assumed to be correlated, and γ_i and ξ_{it} are drawn from a normal distribution, then we can write γ_i as a linear combination of ξ_{it} and a normally distributed variable that is independent of ξ_{it}. This would result in a different relation between the coefficients on ξ_{it} and ε_{it}. Another way to get distinct coefficients on the permanent and transitory shock, is to allow for instance the permanent part of wages to enter the reservation wage directly, for instance as a proxy for assets.

The model presented above can be seen as a reduced form approximation to the policy function describing the work decision of an individual that would follow from a dynamic optimization model with a forward looking agent. The parameters in this model capture the beliefs and preferences of such an agent without giving them a structural interpretation. For instance, some degree of forward looking behaviour is implicitly present in the reservation wage of individuals. For instance, if younger individuals decide to work now, because that increases their human capital, which in turn will increase their expected future earnings, then this will result in a lower reservation wage for young individuals. This is captured by age dummies in the reservation wage. However, to disentangle preferences and beliefs, or to answer questions such as whether individuals distinguish between ξ_{it} and ε_{it} when making their decisions, further structure has to be imposed.

2.4 Estimation

To estimate the model described above, I make some additional assumptions on the distributions of the unobservables, described below. Note that all these distributional assumptions are conditional on education and gender, because estimations are performed

separately by education and gender groups.

[**A-1**] *Joint normality and independence of permanent shocks, α_i and β_i.* I assume that α_i and β_i follow a bivariate normal distribution. The diagonal elements of the variance-covariance matrix are 1, which is again a normalization. These parameters capture the individual specific level and growth of earnings. For older individuals this also includes accumulated human capital. In addition, (α_i, β_i) is assumed to be independent of all variables, including X_{it} and Z_{it}, i.e. α_i and β_i are random effects. This independence is often perceived as a strong assumption, especially if X_{it} contains endogenous variables. For instance, if X_{it} contains education, we might be worried that the decision to go to college is associated with higher values of α_i, leading to correlation between one of the regressors, and the unobservable. Also, dependence is mechanically introduced when the participation equation contains lagged values of participation, because these are by definition correlated with α_i and β_i, if $\rho \neq 0$. In my case, X_{it} contains only age and time dummies, and the estimation is performed conditional on gender and education, which reduces this problem. Correlation with variables in Z_{it}, e.g. the number of kids or marital status will be problematic.

In linear panel data models the fixed effect estimator is therefore often preferred. In wage dynamics models the analysis is usually performed on the differences in residuals from the wage equation. Taking differences removes the fixed effect, α_i, without having to make any assumptions on its stochastic process. Taking differences does not remove the random growth term, so this could cause problems in earnings dynamics models without participation that have education or other endogenous variables in the regression to determine wage residuals. For the participation decision, the level of wages is important, which is why we cannot take differences.

[**A-2**] *Normality and independence of permanent shock to reservation wage, γ_i.* The random effect in the reservation wage, γ_i, that only affects participation is assumed to follow a standard normal distribution independent of the other variables. The parameter that determines its variance, σ_γ, is assumed to be constant over time and age. The random effect should therefore be seen to capture the initial difference in participation probability between individuals.

[**A-3**] *Normality and independence of transitory shock, ε_{it}.* The transitory part of the unobservable in the outcome equation is assumed to follow a standard normal distribution, $\varepsilon_{it} \sim N(0,1)$, independent of all other variables in the model and independent over time. Fixing the variance of this distribution to 1 is a normalization that we need to identify the changing price over time, τ_t, which captures a changing variance. The assumption that there is no serial correlation in ε_{it} is strong and does not hold in practice.

[**A-4**] *Normality and independence of shock to reservation wage, η_{it}.* Similary to the variables above, η_{it} is assumed to follow a standard normal distribution, independent of the other variables and without serial correlation. Since η_{it}, together with γ_i, captures all the variation in participation between individuals not explained by observables or their latent wage, the credibility of the assumption of no serial correlation depends on the instruments that are included in the participation equation. Since assets are not included in the participation equation, it will enter γ_i or η_{it}. As far as there is no variation in assets over time, γ_i can capture the effect, but in other cases we would require some form of auto-correlation in η_{it}.

The normality assumptions introduced above can be used to estimate the model by maximum likelihood. The distributional assumptions on α_i, β_i and γ_i can be relaxed. Since the distributions of these variables are integrated out it is useful to choose a (parametric) distribution that can be accurately integrated over. An example of these would be to assume a mixture of normals for these variables, as I explain below. Another example would be a distribution that takes only a finite number of points, with associated probabilities. This last option would make more sense for γ_i than for the other variables, because γ_i only enters in the discrete choice.

First, I consider the case where there is no state dependence, i.e. lagged participation does not enter the participation equation. To simplify the notation, I define

$$\begin{aligned} u_{1,it} &= \tau_t \varepsilon_{it} \\ u_{2,it} &= \rho \varepsilon_{it} + \sqrt{1-\rho^2}\eta_{it}. \end{aligned}$$

Combining [**A-3**] and [**A-4**] implies a joint normal distribution for $u_{1,it}$ and $u_{2,it}$

$$\begin{pmatrix} u_{1,it} \\ u_{2,it} \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tau_t^2 & \lambda\tau_t^2 \\ \lambda\tau_t^2 & 1 \end{pmatrix}\right).$$

Since we assume that there is no serial correlation in ε_{it} and η_{it}, this means that observations at different time periods, t, for the same individual i are correlated through α_i, β_i and γ_i only. If the values of α_i, β_i and γ_i were observed, the observations for individual i would be independent, similar to Butler and Moffitt (1982), with likelihood contribution for individual i at time t defined as the standard likelihood in a Heckman selection model,

$$\begin{aligned}
&\mathcal{L}\left(\theta \mid d_{it}, y_{it}, X_{it}, Z_{it}, \alpha_i, \beta_i, \gamma_i\right) = \\
&= \begin{cases} 1 - \Phi\left(X_{it}\theta_2 + Z_{it}\delta + \frac{\rho}{\tau_t}\pi_t\left(\alpha_i + \sigma_\beta\beta_i W_{it}\right) + \sigma_\gamma\gamma_i\right) & \text{if } d_{it} = 0 \\ \frac{1}{\tau_t}\phi\left(\frac{y_{it} - X_{it}\theta_1 - \pi_t\left(\alpha_i + \sigma_\beta\beta_i W_{it}\right)}{\tau_t}\right)\Phi\left(\frac{X_{it}\theta_2 + Z_{it}\delta + \sigma_\gamma\gamma_i + \frac{\rho}{\tau_t}\left(y_{it} - X_{it}\theta_1\right)}{\sqrt{1-\rho^2}}\right) & \text{if } d_{it} = 1. \end{cases}
\end{aligned}$$

Given our assumed distributions for α_i, β_i and γ_i, we can integrate out their unobserved values to obtain the likelihood contribution for individual i

$$\mathcal{L}_i = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\prod_{t=1}^{T}\mathcal{L}\left(\theta \mid d_{it}, y_{it}, X_{it}, Z_{it}, \alpha_i, \beta_i, \gamma_i\right) f_{\alpha,\beta,\gamma}(\alpha_i, \beta_i, \gamma_i) d\alpha_i d\beta_i d\gamma_i.$$

Taking the log and summing over all individuals gives the following log-likelihood, where the three-dimensional integral is approximated by Gaussian quadrature, since an analytic expression for the integral is not available

$$\begin{aligned}
\log\mathcal{L} &= \sum_{i=1}^{N}\log\left(\mathcal{L}_i\right) \\
&\approx \sum_{i=1}^{N}\log\left(\sum_{h=1}^{H} w_h \prod_{t=1}^{T}\mathcal{L}\left(\theta \mid d_{it}, y_{it}, X_{it}, Z_{it}, \alpha_h, \beta_h, \gamma_h\right)\right).
\end{aligned}$$

The quadrature weights are denoted by w_h, with corresponding nodes α_h, β_h, and γ_h for the three-dimensional Normal distribution.

For the variations of the model that allow for state dependence, this procedure has to be adapted slightly. We do not observe lagged participation when an individual is observed

for the first time. Also, for some of the individuals no information is available in some of the intermediate waves. Even though we know the participation status of this individual at the start of the survey, for some waves we do not know the participation status for the period that immediately precedes the current period, because the information from that wave may be missing.

Simply removing the first observation from the analysis and using only the outcome value from the first period as lagged participation in the second period results in biased estimates, because the first period outcome is correlated with the unobservable random effects in the participation equation. This problem is known as the initial conditions problem. I take one of the solutions proposed in Heckman (1981) and Stewart (2007), where a separate equation is specified to model the participation status in the first period. In addition, I use an additional equation to model the participation status following a wave with missing data.

For the first period observations, the participation status is modeled as

$$P(d_{i1}=1) = P\left(X_{i1}\theta_{2,\text{ic}} + Z_{i1}\delta_{\text{ic}} + \frac{\rho_{\text{ic}}}{\tau_1}\pi_1\xi_{i1} + \sigma_{\gamma,\text{ic}}\gamma_i + \rho_{\text{ic}}\varepsilon_{i1} + \sqrt{1-\rho_{\text{ic}}^2}\eta_{i1} > 0\right).$$

Note that all the parameters in this equation, except π_1 and τ_1, have an ic (initial conditions) subscript. They can take on different values than the parameters in subsequent periods. A similar equation, with different coefficients, is used to model the participation status when no information from a previous wave is available, because of missing data. In that equation I also include the last value of participation status that was observed in any of the previous waves. Because we do not have enough observations to identify all coefficients on time and age dummies for these initial condition equations, I only include dummies in these two equations that divide the age data in 5-year periods.

To get some idea about the performance of this method and the accuracy of the integral approximation, I run a small Monte Carlo experiment. I simulate observations for 500 individuals that are between age 30 and 50 when we start observing them. They enter the workforce at age 25, so they have between 5 and 25 periods of of observations affecting the initial conditions. Furthermore, 20% of the observations are missing at random, to simulate individuals with information from intermediate waves missing. The model contains one

Table 2.3 – Monte Carlo simulations showing bias in parameter estimates, $T = 6$

	Number of integration nodes ($\alpha_i \times \gamma_i$)						
	3×3	7×7	11×11	17×17	25×25	49×25	99×25
$\log \mathcal{L}$	–2333.344	–2181.449	–2154.983	–2144.827	–2143.130	–2142.934	–2142.896
	(62.017)	(52.673)	(51.794)	(51.308)	(51.577)	(51.556)	(51.529)
$\theta_{1,1}$	0.028	0.000	–0.002	–0.001	0.000	0.000	0.000
	(0.046)	(0.051)	(0.042)	(0.033)	(0.027)	(0.025)	(0.026)
$\theta_{1,2}$	–0.001	0.000	0.000	0.001	0.001	0.001	0.001
	(0.008)	(0.008)	(0.008)	(0.008)	(0.008)	(0.008)	(0.008)
$\theta_{2,1}$	–0.147	–0.014	–0.003	0.009	0.014	0.015	0.015
	(0.165)	(0.207)	(0.214)	(0.212)	(0.207)	(0.207)	(0.207)
$\theta_{2,2}$	–0.023	0.004	0.006	0.008	0.008	0.008	0.008
	(0.061)	(0.067)	(0.070)	(0.069)	(0.069)	(0.069)	(0.069)
δ	–0.047	0.001	0.006	0.009	0.010	0.010	0.010
	(0.077)	(0.087)	(0.090)	(0.090)	(0.089)	(0.089)	(0.089)
ϕ	0.078	0.017	0.014	0.012	0.012	0.011	0.011
	(0.134)	(0.145)	(0.145)	(0.144)	(0.144)	(0.144)	(0.144)
τ	0.056	0.015	0.007	0.002	0.001	0.001	0.001
	(0.009)	(0.006)	(0.005)	(0.005)	(0.005)	(0.005)	(0.005)
π	–0.105	–0.068	–0.036	–0.010	–0.002	–0.001	–0.001
	(0.017)	(0.019)	(0.020)	(0.019)	(0.020)	(0.019)	(0.019)
σ_γ	–0.156	–0.012	–0.006	–0.006	–0.006	–0.006	–0.006
	(0.114)	(0.160)	(0.168)	(0.164)	(0.164)	(0.164)	(0.164)
ρ	0.021	0.011	0.007	0.007	0.007	0.007	0.007
	(0.085)	(0.072)	(0.065)	(0.062)	(0.061)	(0.061)	(0.061)

Mean bias is shown based on 100 replications of simulated datasets with 500 individuals, standard deviation in parentheses. These parameters were used to simulate the data $(\theta_{1,1}, \theta_{1,2}, \theta_{2,1}, \theta_{2,2}, \delta, \phi, \tau, \pi, \sigma_\gamma, \rho) = (5, 1, 1, 0.5, 1, 1, 0.3, 0.5, 1, 0.4)$.

covariate, $x_{1,it} \sim$ i.i.d. $N(0, 1)$, and one instrument, $z_{1,it} \sim$ i.i.d. $N(0, 1)$. The vector of parameters that was used to simulate the data is $(\theta_{1,1}, \theta_{1,2}, \theta_{2,1}, \theta_{2,2}, \delta, \phi, \tau, \pi, \sigma_\gamma, \rho) = (5, 1, 1, 0.5, 1, 1, 0.3, 0.5, 1, 0.4)$. The model does not contain a heterogeneous trend, and therefore we do not integrate over β_i.

Table 2.3 shows the bias in the parameter estimates for different numbers of nodes that were used to approximate the integral. The results are based on 100 Monte Carlo replications and use $T = 6$ observed waves. The same results are shown for the case where we have more observations for each individual, $T = 16$, in table 2.4. From these tables we see that the bias in all parameters is substantial when a small number of integration nodes is used, 3×3, in the first column. When the number of integration nodes increases, the bias decreases. Because we are interested in estimates for the variance parameters, it is important to use a high number of integration nodes to approximate the integral. For the estimations on the BHPS data, I use 99 quadrature nodes in the α_i dimension, and 25

Table 2.4 – Monte Carlo simulations showing bias in parameter estimates, $T = 16$

	Number of integration nodes ($\alpha_i \times \gamma_i$)						
	3×3	7×7	11×11	17×17	25×25	49×25	99×25
$\log \mathcal{L}$	–5391.377	–4791.423	–4691.189	–4634.537	–4606.182	–4590.960	–4590.367
	(120.802)	(89.528)	(88.798)	(90.434)	(90.535)	(89.913)	(90.044)
$\theta_{1,1}$	0.053	0.013	0.011	0.003	0.008	0.004	0.003
	(0.053)	(0.067)	(0.055)	(0.053)	(0.048)	(0.035)	(0.028)
$\theta_{1,2}$	–0.002	–0.001	–0.001	0.000	0.000	0.000	0.000
	(0.005)	(0.004)	(0.004)	(0.004)	(0.004)	(0.004)	(0.004)
$\theta_{2,1}$	–0.176	–0.052	–0.020	–0.017	–0.007	–0.007	–0.008
	(0.096)	(0.110)	(0.116)	(0.109)	(0.115)	(0.101)	(0.100)
$\theta_{2,2}$	–0.034	–0.009	–0.004	–0.003	–0.002	–0.002	–0.002
	(0.033)	(0.032)	(0.033)	(0.033)	(0.034)	(0.033)	(0.033)
δ	–0.062	–0.013	–0.005	–0.003	–0.002	–0.001	–0.001
	(0.037)	(0.039)	(0.040)	(0.040)	(0.040)	(0.040)	(0.040)
ϕ	0.083	0.020	0.008	0.005	0.004	0.004	0.004
	(0.076)	(0.077)	(0.079)	(0.078)	(0.078)	(0.078)	(0.078)
τ	0.059	0.015	0.009	0.005	0.003	0.001	0.001
	(0.007)	(0.004)	(0.004)	(0.003)	(0.003)	(0.003)	(0.003)
π	–0.132	–0.134	–0.102	–0.067	–0.038	–0.005	–0.001
	(0.016)	(0.018)	(0.020)	(0.020)	(0.020)	(0.020)	(0.018)
σ_γ	–0.226	–0.053	–0.014	–0.008	–0.009	–0.010	–0.010
	(0.048)	(0.061)	(0.073)	(0.076)	(0.075)	(0.075)	(0.074)
ρ	–0.007	–0.013	–0.007	–0.003	–0.002	–0.001	–0.001
	(0.067)	(0.046)	(0.042)	(0.038)	(0.038)	(0.038)	(0.038)

Mean bias is shown based on 100 replications of simulated datasets with 500 individuals, standard deviation in parentheses. These parameters were used to simulate the data $(\theta_{1,1}, \theta_{1,2}, \theta_{2,1}, \theta_{2,2}, \delta, \phi, \tau, \pi, \sigma_\gamma, \rho) = (5, 1, 1, 0.5, 1, 1, 0.3, 0.5, 1, 0.4)$.

nodes in the γ_i dimension. In the models with a heterogeneous trend, I use 25 integration nodes in the β_i dimension.

2.4.1 Mixture of normal distributions

To allow for a more flexible distribution than the normal distribution, I estimate variations of the models that contain only α_i as a random effect using a mixture of two normal distributions. This mixture has probability density function

$$f_\alpha(\alpha_i) = \kappa_1 f_1(\alpha_i) + (1 - \kappa_1) f_2(\alpha_i),$$

where f_1 and f_2 are probability density functions of a univariate normal distribution with different means and variances. The probability that an observation is drawn from the first component of the mixture is κ_1, and because the probabilities should add to one,

the probability that an observation comes from the second component of the mixture is $\kappa_2 = 1 - \kappa_1$.

Allowing for a normal mixture distribution increases the computational cost in two ways. First, we need more integration nodes to accurately approximate the integral. In the case of a mixture of two normal distributions, I approximate two integrals, conditional on whether the observation is drawn from the first component of the mixture, or from the second component. This means that it takes twice as long to approximate the log-likelihood. The second way in which the problem requires more computing time, is because the log-likelihood function is more difficult to optimize, which typically means that more iterations are needed to converge to the optimum.

Finally, in the original formulation, α_i was normalized to have zero mean and unit variance. In this variation, α_i is defined as a mixture of two non-standard normal distributions; $\alpha_{i1} \sim N(\mu_1, \sigma_1^2)$ and $\alpha_{i2} \sim N(\mu_2, \sigma_2^2)$. In line with the original formulation, I normalize the mean and variance of α_i; $E[\alpha_i] = 0$, and $E[\alpha_i^2] = 1$. We can then express μ_2 and σ_2 in terms of μ_1 and σ_1 to obtain this normalization, where

$$E[\alpha_i] = 0 \quad \Leftrightarrow \quad \mu_2 = -\frac{\kappa_1}{1-\kappa_1}\mu_1.$$

and

$$E[\alpha_i^2] = 1 \quad \Leftrightarrow \quad \sigma_2^2 = \frac{1}{1-\kappa_1}\left(1 - \frac{\kappa_1}{1-\kappa_1}\mu_1^2 - \kappa_1\sigma_1^2\right).$$

The parameters μ_2 and σ_2 are not estimated, but their values follow directly from the normalization.

2.4.2 Serial correlation in the unobservables

The factor structure that I assume for the dependence of unobservables over time simplifies the analysis, because in that case the log-likelihood can be approximated by a low-dimensional integral. Inluding serial correlation for individual i in a different form is an order of magnitude more complicated. This occurs if ξ_{it} is not a function of two

unobserved factors, as above, but follows a random walk

$$\xi_{it} = \xi_{it-1} + v_{it},$$

or when $u_{1,it}$, or $u_{2,it}$ follow some auto-regressive or moving-average process. In principle we can condition on ξ_{i0}, and $v_{i1}, \dots v_{iT}$, to integrate out these unobservables using the product of multiple one-dimensional integration grids. In practice this approach quickly becomes intractible if the number of time periods, T, grows. Especially since the integral has to be approximated accurately enough to get informative estimates for the variance parameters.

2.5 Results

This section describes the estimation results for the different variations of the earnings dynamics models that I described above. The identifiers and characteristics of the different variations are shown in table 2.5. The models are divided in three parts; the base models, and two variations that use either mixture distributions or a heterogeneous trend. There is a distinction between models that include wages only, starting with the letter W, and models that model both wages and participation at the same time. These models include a selection equation and start with the letter S.

The wage models are estimated using maximum likelihood, where the normality assumption is used equivalently to the selection model, so that we can easily compare the results from those models. The W models can be seen as simple benchmark wage dynamics models that are commonly estimated. The normality assumption is stronger than the assumptions that are usually made in models focusing on wages or earnings only, as most of these models use a set of low-order moments to estimate variance parameters of the wage process. Also, instead of running a linear regression on some observables first, to obtain wage residuals that can be used in the subsequent analysis, I include these observables in the estimation directly.

All models include a persistent part, $\xi_{it} = \alpha_i$, that affects the outcome and, if present, the selection equation. In the base models, two selection models have an additional random

Table 2.5 – Model variations

	Selection	Persistence in participation	State Dependence	Mixture distribution	Heterog. trend
Base models					
W	No	–	–	No	No
S^{Base}	Yes	No	No	No	No
S^{REP}	Yes	Yes	No	No	No
S^{SD}	Yes	No	Yes	No	No
$S^{REP,SD}$	Yes	Yes	Yes	No	No
Variations with mixtures					
W^{mix}	No	–	–	Yes	No
$S^{SD,mix}$	Yes	No	Yes	Yes	No
Variations with heterogeneous trend					
W^{age}	No	–	–	No	Yes
$W^{\sqrt{age}}$	No	–	–	No	Yes
$S^{SD,age}$	Yes	No	Yes	No	Yes
$S^{SD,\sqrt{age}}$	Yes	No	Yes	No	Yes

effect, γ_i, that enters the selection equation only. These models are labelled S^{REP} and $S^{REP,SD}$, where REP refers to the random effect in the participation equation. For the other variations the effect of γ_i is removed, i.e. $\sigma_\gamma = 0$ in those cases.

The third column of table 2.5 shows whether the model allows for state dependence, marked by the superscript SD. It is assumed that lagged participation only enters the selection equation. Therefore, the wage model does not have a variation including state dependence.

In one set of variations, I estimate versions of model W and model S^{SD} with a mixture of two normal distributions as the distribution for α_i. These variations are referred to as W^{mix} and $S^{SD,mix}$. The final set of variations introduces a heterogeneous trend in the persistent part of the unobservable. I.e. ξ_{it} contains $\sigma_\beta \beta_i W_{it}$ in addition to α_i. The variable W_{it} defines the heterogeneous trend and is equal to age or to the square-root of age.

2.5.1 Results for base models

The models estimated in this section contain many parameters. There are time and year dummies in both the outcome and the selection equation, additional parameters for instrumental variables affecting participation but not wages, time-varying variance

parameters for the persistent and transitory part of the unobservable, and some additional parameters related to the variances and correlation of unobservables. Not all of these parameters are of direct interest, nor is it easy to interpret them when shown in a large table. Therefore I present only a subset of the parameter estimates.

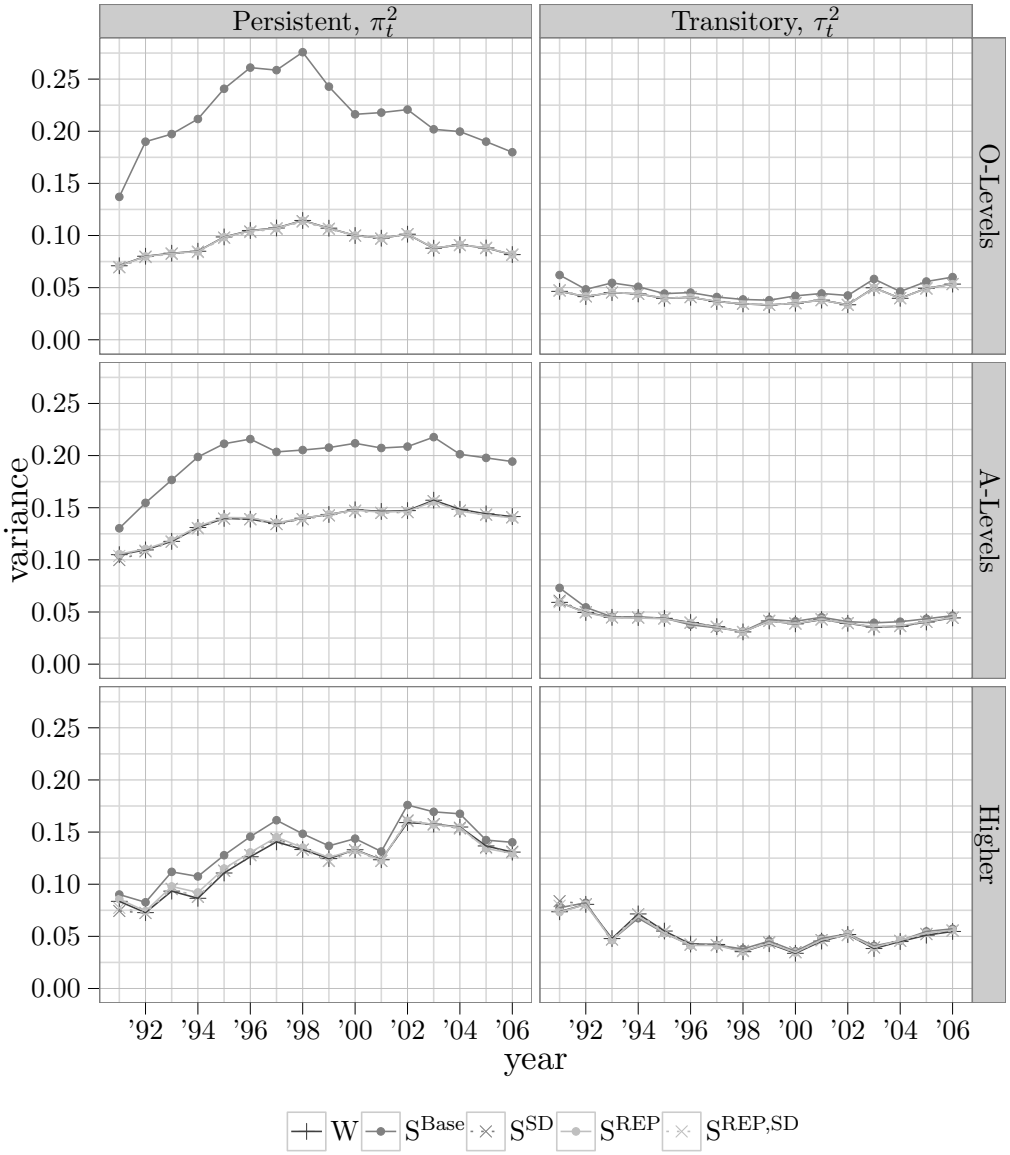

Figure 2.9 – Estimated transitory and persistent variances for males by year

Figures 2.9 and 2.10 show the estimated variance parameters for the persistent variance, π_t^2, and the transitory variance, τ_t^2, in different years for models W, $^{\text{Base}}$, $^{\text{SD}}$, $^{\text{REP}}$ and $\text{S}^{\text{REP,SD}}$. The squared parameters are shown in order to simplify comparisons with other studies, where these parameters are usually presented as variances instead of standard

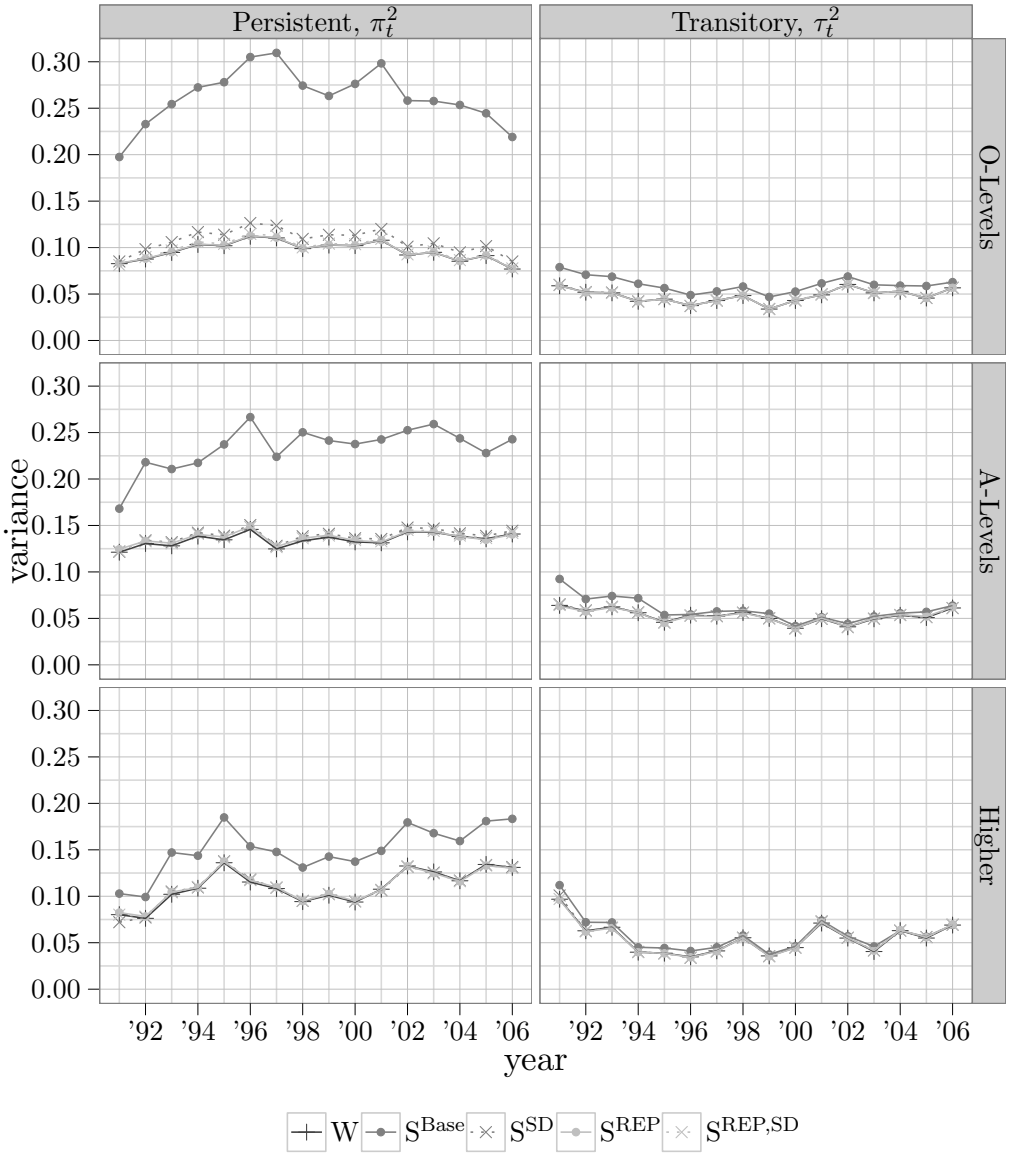

Figure 2.10 – Estimated transitory and persistent variances for females by year

deviations.

When looking at figures 2.9 and 2.10 we see that the estimates for the transitory variances, the graphs on the right, are very similar for all variations of the model. The transitory variances are somewhat higher in the beginning of the 1990's especially for the higher education group, and they increase again slightly towards the end of the sample. The transitory variances are of similar magnitude for all education levels for males and females.

The second thing to note is the difference in the contribution of the persistent variance

for the different models. We see that for most models, the estimates for the persistent variance lie almost completely on top of the estimates for model W. The exception is S^{Base}, where the estimated persistent variance is visibly larger. We see further that the differences in estimates for π_t between model W and model S^{Base} are larger for the lower education groups. The effect of not including non-random selection into work is more important for these groups, because they contain a larger proportion of individuals that do not work.

As we saw from the plots of transition probabilities in the BHPS data, selection into work is very persistent. If this persistence is not due to persistence in observables, the only way for model S^{Base} to capture the persistence, is by having large values for π_t, the coefficients governing the persistence of latent wages. The only way to explain persistence in participation is to have a high persistence in latent wages. The other selection models however, include a second channel that affects the individual differences in the propensity to participate. Model S^{REP} contains a random effect, γ_i, that only enters the selection equation and that can capture individual differences in the propensity to participate independent from the persistence in latent wages. This factor captures all the persistence in participation status, and a large value of π_t is not required to improve the log-likelihood for this model. Similarly, S^{SD} contains lagged participation in the participation equation, to allow for persistence in participation status independently from persistence in latent wages.

We can compare the results for males to Blundell and Etheridge (2010). They estimate a wage dynamics model for males with a slightly different specification; most notably they pool all education groups, and only include wages of employed individuals, as I do in model W. In their model the persistent shock follows a random walk, whereas the transitory shock follows a moving average process. They use data from the BHPS from a similar period, 1991 – 2004. From the results of this model they calculate the implied variance for the persistent and the transitory part of log-earnings which is shown in the top panel of figure 6.1 in their paper. Because earnings consists of a combination of hours and wages, I

deocmpose the variance of earnings as follows

$$\begin{aligned} \text{var}\left[\log\left(\text{earnings}\right)\right] &= \text{var}\left[\log\left(\text{hours}\cdot\text{wage}\right)\right] \\ &= \text{var}\left[\log\left(\text{hours}\right)\right] + \text{var}\left[\log\left(\text{wage}\right)\right] + \text{cov}\left[\log\left(\text{hours}\right),\log\left(\text{wage}\right)\right], \end{aligned}$$

which implies that the variance of log wages is a lower bounds for the variance of log earnings, as the covriance between log hours and log wages can be assumed to be nonnegative. The variance of earnings and the variance of log wages are roughly the same if the variance of log hours and the covariance between log hours and log wages is small, which is plausible for males. Taking this into account it looks like the estimates from Blundell and Etheridge (2010) for the contribution of the transitory variance are very similar to mine; their estimates are around 0.04. The contribution of the persistent variance are higher in my case, compared to their estimates around 0.06. Also, in Blundell and Etheridge (2010) the contribution of the persistent variance decreases over the sample period, whereas this seems to increase for A-Levels and the Higher education group in my results.

Appendix 2.B contains tables with estimation results for a subset of the parameters. The tables contain three sets of values, separated by horizontal rows. The top rows show the coefficients on the instrumental variables in the selection equation. These are not present in the wage model, W. The middle set of rows contain estimates for the parameters related to unobservables that are not shown in figures 2.9 and 2.10. Not all of these parameters are present in every model. The final set of rows present some additional information, such as the log-likelihood, $\log \mathcal{L}$, the number of individuals, N, and the total number of observations $N \cdot T$, used in the estimation, where not all individuals are observed in every wave.

Finally, the implied correlation is a measure of correlation between the unobservable in the outcome equation and the unobservable in the selection equation. These unobservables are a combination of the separate unobservable parts, and in these calculations I use the average over all time-periods for π_t, and τ_t. This implied correlation can be interpreted as the correlation between the unobservables in the outcome and selection equation for static selection models, which measures the amount of non-random selection. See appendix 2.A

for the exact definition.

Looking at the estimate for ρ in model S^{Base} in the second column of tables 2.6, 2.7, and 2.8 in appendix 2.B, we see that for O-Levels the non-random selection is stronger than for the A-Levels or Higher education group. The implied correlation between the unobservables in the outcome and the selection equation is 0.8 for O-Levels and 0.5, and 0.4 for A-Levels and Higher respectively. This relates directly to the differences we saw in the estimates for π_t when comparing model W and model S^{Base}. The models with a larger implied correlation show a larger difference.

In comparison to S^{Base}, models S^{SD}, S^{REP}, $S^{REP,SD}$ contain additional parameters to allow for persistence in participation that is independent from persistence in wages. S^{Base} is nested in S^{SD}, and S^{Base} is nested in S^{REP}. All three of these models are nested in the more complicated model $S^{REP,SD}$. A likelihood ratio test can be used to compare the models and in all cases the simpler models are rejected in favour of the more complex model.

The size of the random effect in the participation equation in S^{REP}, σ_γ, is large compared to the contribution of the persistent and the transitory shock, α_i and ε_{it}. Also, the estimate for ρ as well as the implied correlation is considerably smaller than the corresponding estimate in S^{Base}. This suggests that it is important to capture the persistence in work status separately from persistence in log wages. This also suggests that the non-random selection in to work largely goes away once we condition on some underlying propensity to work that is not correlated to an unobserved factor determining log wages. Depending on the specification of the model, the bias in the variance parameters compared to a wage dynamics model without participation is large (S^{Base}) or virtually non-existent (S^{REP}).

The same changes occur when we introduce lagged participation in the participation equation in model S^{SD}, where the estimate for ρ is again smaller than in model S^{Base}. The estimated coefficient on lagged participation is large compared to the other coefficients in the participation equation, ranging between 1.8 for Higher educated males and 2.6 for males with O-Levels. For an easier interpretation of the magnitude of this coefficient, I calculate the probability of working in year 2000, for a 40-year old married male, without

kids, who owned a house in the first period. For an individual with O-Levels, this probability changes from 16% to 95% when we change his status from not working in the previous period, to working in the previous period. Similarly, the percentage increase from 44% to 98% for A-Levels and from 58% to 98% for Higher education. The same probability can be calculated for model $S^{REP,SD}$, where the values go from 50% to 90% for O-Levels, from 76% to 96% for A-Levels, and from 76% to 97% for Higher education.

The results for females are shown in tables 2.9, 2.10, and 2.11. The presence of kids lowers the probability that an individual works, and the effect is stronger for young kids than for older kids. The same as we saw in the case of males, if we introduce a random effect or state dependence in the participation equation, the parameter estimate for ρ decreases, and the issue of non-random selection decreases.

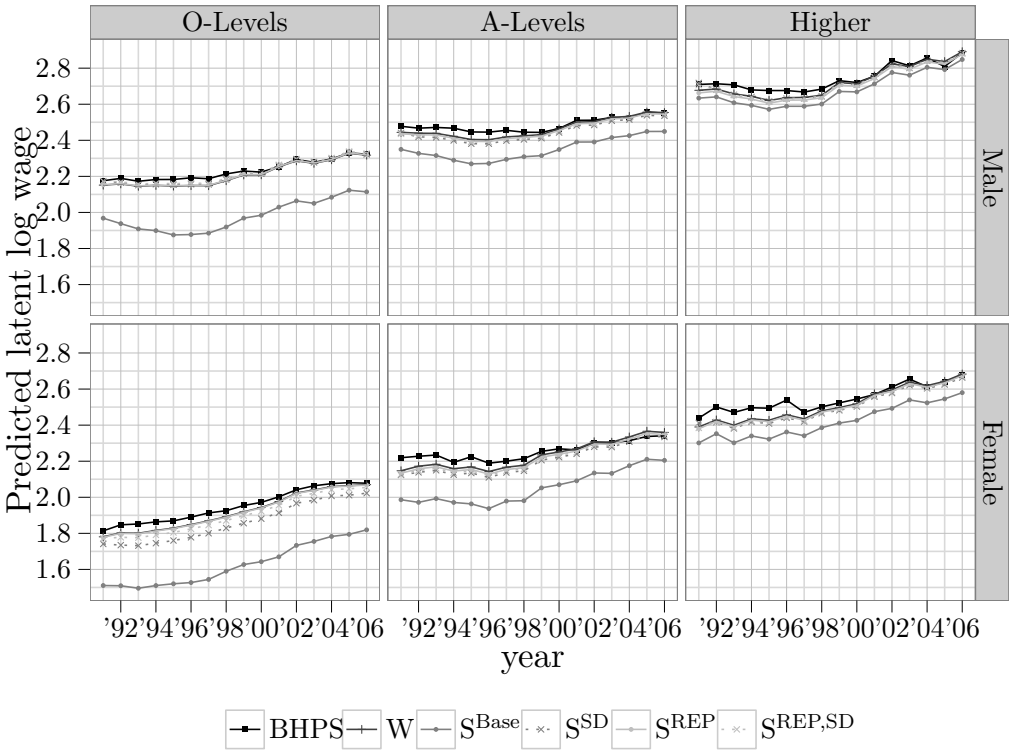

Figure 2.11 – Estimated latent wage by year

Another way to look at the results of non-random selection on the level of wages is by plotting predicted latent wages for the different models together with the observed wage in the BHPS, see figures 2.11 and 2.12. These figures show the amount of selection present

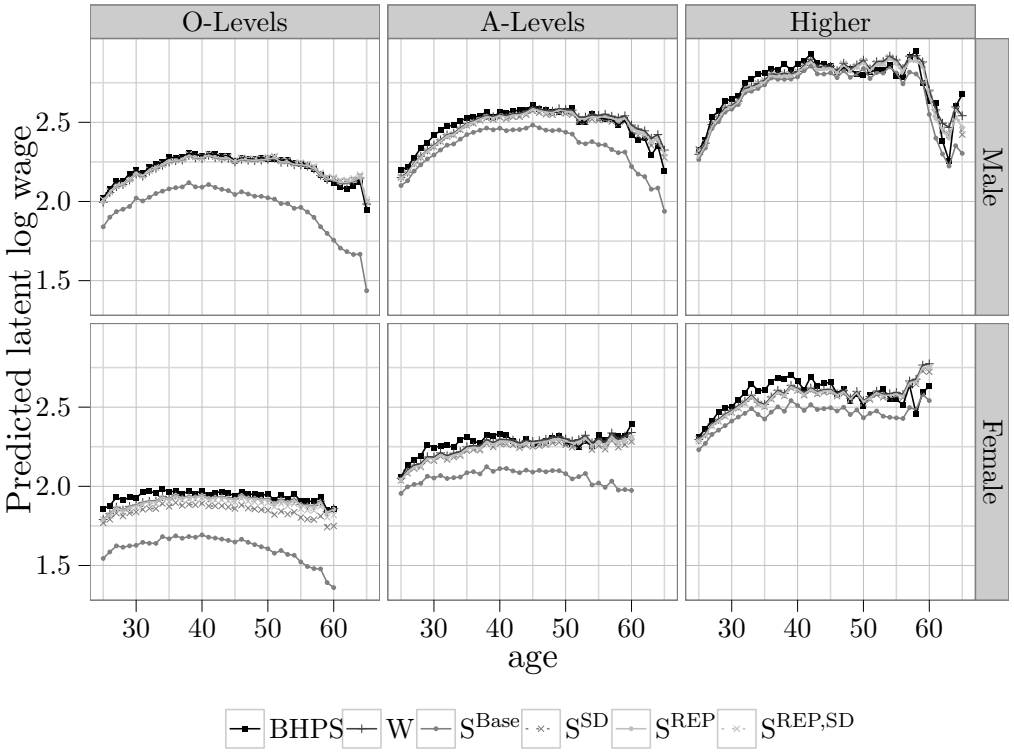

Figure 2.12 – Estimated latent wage by age

in each of the variations. As before, we see that the non-random selection in model S^{Base} is strong. The latent wage in this case is up to 25% smaller than the wage we observe for working individuals. For the other selection models there is still some effect of the non-random selection, especially in the early 90's and for younger individuals, but the effect is smaller. This can be explained by the increase in labour participation that we observed for these groups over this period.

2.5.2 Assessment of fit

To assess the fit of the different variations, I show some figures of simulated moments in appendix 2.C. For each individual 1000 random paths of the unobservables are drawn to construct simulated datasets. From these simulated datasets, moments can be calculated both by year and by age. The comparison between the wage model and the selection models is not entirely obvious, because they operate on a different set of data. The wage model does not include any data on individuals that are not working, and therefore we do

not have predicted participation or transition probabilities for this model. The selection models can utilize more parameters, but at the same time these models have to explain more features of the data, i.e. patterns in participation. If the selection models can replicate the log-wages in a similar way as the wage model, then we prefer the selection models, because these also help explain an additional part.

Figure 2.13 shows the mean of log wages for the BHPS data and the different models. The data from the BHPS is shown in black. The wage model without selection, W, is shown as a solid line in a dark shade of gray with a plus-sign marking the years. A slightly lighter shade of gray is used for S^{Base} and S^{SD}, where the distinction between the two is made in the type of line, solid versus dashed, and the type of marker used, a dot versus a cross. Finally S^{REP} and $S^{REP,SD}$ are shown in the lightest shade of gray, with the same distinction in line type and the type of marker as for the other selection models. This coding of colours, line types and markers, is the same for the next few figures.

We see from figure 2.13 that the mean of log-wages is reasonably reproduced by all models. In the first couple of years log-wages predicted by the models are lower than observed log-wages. Because all lines are very close to each other, figure 2.14 shows the difference between the simulated data and the observed data, where we can see more clearly that the prediction in the first set of years is too low. The discrepancy between observed wages and the wages predicted by W can be attributed the normality distribution being violated, e.g. because there is some skewness in the distribution of the errors.

Figure 2.15 shows the variance of log-wages. We see that the variance for females can be replicated reasonably well for all models except S^{Base}. For O-Level and A-Level males, the predicted variance is on top of the BHPS data for the first few years, but is too large in the next years. The predicted variance of the difference in log-wages in figure 2.16 is almost twice as high as the variance that we observe in the data. The introduction of an MA component in the transitory shock, could reduce this difference as can be seen from table 2.2, whereas introducing a random walk for the persistent component would only increase the difference. According to the same reasoning, introducing an MA component in the transitory shock would to better match the auto-covariance of the difference in log wages in figure 2.18.

The proportion of individuals that are working in each year is shown in figure 2.19. The difference between simulated and observed moments are given in figure 2.20. Model W is not shown in these figures, as participation is not present in the model. The predicted participation is too low, but there is not one particular variation that performs better than the others. However, if we look at the transition probabilities from non-work to non-work in figure 2.21 and the transition probabilities from work to work in figures 2.22 it is clear that the predicted probabilities of S^{SD} and $S^{REP,SD}$ are closer to the probabilities that we observe in the data, than the probabilities predicted by the selection models without state dependence.

As seen before, S^{Base} can not replicate the transition probabilities. The probability of staying in the same state for two consecutive periods is high. The probability of not working in the next period when not working in the current period is somewhere between 0.7 and 0.9, and the probability of working in the next period given that you're working in the current period is larger than 0.9. The only variable that affects persistence in work status in model S^{Base} is α_i. Increasing the coefficient on $alpha_i$ in the participation equation, π_t, however, also increases the variance of log-wage, as α_i enters the equation for log-wage with the same coefficient. The log-likelihood estimation makes a trade-off between a large deviation from the observed variance and auto-covariance, and a large deviation in the transition probabilities. Model S^{REP} has an additional random effect, γ_i, and its associated parameter, σ_γ, to separately influence the variance and auto-covariance of wages and the persistence in labour participation. This brings the prediction closer to the observed probabilities, but it is still about 10 percentage points too low.

Figures 2.23 to 2.32 show the same moments by age and the same remarks can be made about these figures. In figure 2.24 we see that the mean of log wages is particularly difficult to estimate for older individuals. Their participation rates are lower, which means that we do have as many observations. The inclusion of age dummies in the model does not solve this. Because the base models include only a random effect in the wage model, but not random walk, the predicted variance is constant over age in figure 2.25. The variance for S^{Base} decreases with age, because of a combination of decreasing participation rates and non-random selection. Clearly a richer structure has to be assumed for the unobservables

to better capture the pattern of the changing variance with age.

2.5.3 Variation: mixture distribution

In these variations I relax the distributional assumptions of the base models, by assuming a mixture of two normal distributions for α_i. I re-estimate models W and S^{SD} using a mixture distribution. The results are given in tables 2.12 and 2.13 in appendix 2.D under W^{mix} and $S^{SD,mix}$. The parameters $\mu_{1,\alpha}$ and $\sigma_{1,\alpha}$ are the mean and standard deviation of the first component of the mixture. The probability that the element is drawn from the first component is $\pi_{1,\alpha}$. The estimates for the mean and standard deviation from the second component, $\mu_{2,\alpha}$ and $\sigma_{2,\alpha}$ follow from the normalization that I chose for the mean and standard deviation of α_i. Their standard errors are calculated using the Delta-method.

Introducing the mixture improves the log-likelihood of the models, for instance because some skewness in the distribution of the persistent unobservable be captured by the mixture. However, when comparing figures of simulated moments, not reproduced here, we see virtually no difference between the base models and their mixture distribution counterpart.

2.5.4 Variation: heterogeneous trend

As a second variation I estimate model W and model S^{SD} with a heterogeneous trend in age, or a heterogeneous trend in $\sqrt{\text{age}}$. In this case, the parameters σ_β and $\rho_{\alpha\beta}$ can be different from 0. The tables in appendix 2.E show selected parameter estimates. The estimate correlation between α_i and β_i is negative, which is similar to what has been found in other studies (such as Baker & Solon, 2003). Based on the likelihood, in most cases the version including a heterogeneous trend in age, is preferred over the version with a heterogeneous trend in $\sqrt{\text{age}}$.

I introduced the heterogeneous trend to see if we can better match the variance of log-wages and and the variance of the difference in log-wages. Simulated moments for these by age are shown in figure 2.33 and 2.34 in appendix 2.F. The predicted variance of log-wages is increasing in both cases, but the predicted levels are much higher than the observed variance, especially for the models with a heterogeneous trend in age. From

comparing figure 2.34 with 2.26 the difference between observed and predicted variance of the difference in log-wages is smaller in the models with heterogeneous trend than in the base models. The large discrepancy in the predicted variance of log-wages leads me to prefer the base models.

2.6 Conclusion

In this paper I formulated a model of wage dynamics where I included the decision to work in a selection equation. I then estimated different variations of the model on wage and labour participation data from the United Kingdom. From the comparison of simulated moments for different models we saw that high auto-correlation in log-wages can not by itself explain the persistence in work status. Allowing for a random effect in the participation equation brings the predicted transition probabilities closer to the transition probabilities that we observe in the data. Including state dependence in the participation equation further improves the model fit. It was also found that conditional on work status in the previous period, non-random selection into work is small.

Two additional experiments where performed to relax some of the assumptions I made. First, the random effect in the latent log-wage equation was allowed to follow a mixture of two normal distributions, instead of coming from a normal distribution. Although this addition improves the log-likelihood of the models, no visible improvement was observed in the simulated moments. As a second experiment, a heterogeneous trend was introduced in the log-wage equation. This improved the predicted variance of the difference in log-wages slightly, at the cost of highly overestimating the variance of log-wages.

Not all of the dynamics of the wage process are captured by the models presented here. It is expected that extending the structure of the unobservables will improve the models further, for instance by introducing a random walk in the persistent part or by introducing a moving average component in the transitory part. Further research combining models of wage dynamics with models of labour supply is welcomed to get a better understanding of the importance of different drivers of heterogeneity and their persistence.

2.A Calculation of implied correlation

From the equations in the main text we obtain the total unobservable in the outcome equation, which I define in this section as $\omega_{1,it}$, and the total unobservable in the selection equation, defined here as $\omega_{2,it}$

$$\begin{aligned}\omega_{1,it} &= \pi_t \xi_{it} + \tau_t \varepsilon_{it} \\ \omega_{2,it} &= \frac{\rho}{\tau_t}\pi_t \xi_{it} + \sigma_\gamma \gamma_i + \rho \varepsilon_{it} + \sqrt{1-\rho^2}\eta_{it},\end{aligned}$$

with the expectation of both variables equal to 0, $E[\omega_{1,it}] = E[\omega_{2,it}] = 0$. We want to calculate the correlation between $\omega_{1,it}$ and $\omega_{2,it}$, which by definition is

$$\text{cor}[\omega_{1,it}, \omega_{2,it}] = \frac{\text{cov}[\omega_{1,it}, \omega_{2,it}]}{\sqrt{\text{var}[\omega_{1,it}]}\sqrt{\text{var}[\omega_{2,it}]}}.$$

The variances in this equation are given by

$$\text{var}[\omega_{1,it}] = E\left[\omega_{1,it}^2\right] = E\left[(\pi_t \xi_{it} + \tau_t \varepsilon_{it})^2\right] = \pi_t^2 + \tau_t^2,$$

and

$$\begin{aligned}\text{var}[\omega_{2,it}] &= E\left[\omega_{2,it}^2\right] \\ &= E\left[\left(\frac{\rho}{\tau_t}\pi_t \xi_{it} + \sigma_\gamma \gamma_i + \rho \varepsilon_{it} + \sqrt{1-\rho^2}\eta_{it}\right)^2\right] \\ &= \left(\frac{\rho}{\tau_t}\pi_t\right)^2 + \sigma_\gamma^2 + 1.\end{aligned}$$

The covariance is calculated as

$$\begin{aligned}\text{cov}[\omega_{1,it}, \omega_{2,it}] &= E[\omega_{1,it} \cdot \omega_{2,it}] \\ &= E\left[(\pi_t \xi_{it} + \tau_t \varepsilon_{it}) \cdot \left(\frac{\rho}{\tau_t}\pi_t \xi_{it} + \sigma_\gamma \gamma_i + \rho \varepsilon_{it} + \sqrt{1-\rho^2}\eta_{it}\right)\right] \\ &= \frac{\rho}{\tau_t}\pi_t^2 + \tau\rho.\end{aligned}$$

These are combined to get the correlation between the unobservables in the two equations

$$\text{cor}\left[\omega_{1,it}, \omega_{2,it}\right] = \frac{\frac{\rho}{\tau_t}\pi_t^2 + \tau\rho}{\sqrt{\pi_t^2 + \tau_t^2}\sqrt{\left(\frac{\rho}{\tau_t}\pi_t\right)^2 + \sigma_\gamma^2 + 1}}.$$

Since the parameters π_t and τ_t are varying with time, I take the average of both over all time periods to calculate the implied correlation. This gives a good indication if the parameters do not vary too much over time.

2.B Estimation results: base models

Table 2.6 – Selected estimates for base models for males with O-Levels

	W	S^{Base}	S^{REP}	S^{SD}	$S^{REP,SD}$
Married		0.147**	0.277***	−0.018	0.082
		(0.048)	(0.065)	(0.065)	(0.098)
Coupled		−0.396***	−0.182*	−0.001	−0.005
		(0.060)	(0.087)	(0.079)	(0.116)
Has kids aged 0 – 2		−0.037	−0.175*	−0.191**	−0.238**
		(0.059)	(0.101)	(0.078)	(0.098)
Has kids aged 3 – 4		0.016	−0.046	0.160*	0.115
		(0.059)	(0.099)	(0.081)	(0.099)
Has kids aged 5 – 11		−0.125**	−0.172**	−0.113*	−0.185**
		(0.043)	(0.061)	(0.055)	(0.075)
Has kids aged 12 – 15		−0.070	−0.062	−0.092	−0.101
		(0.047)	(0.071)	(0.060)	(0.078)
Spouse has job		0.325***	0.349**	0.225*	0.304*
		(0.087)	(0.121)	(0.114)	(0.155)
Log-wage for spouse		−0.048	0.058	0.025	0.040
		(0.031)	(0.049)	(0.035)	(0.046)
Log hrs/week spouse		0.125***	0.142***	0.074*	0.107*
		(0.028)	(0.036)	(0.035)	(0.048)
Owns house period 1		0.011	1.043	0.233***	0.569***
		(0.035)		(0.043)	(0.089)
Lagged participation				2.603***	1.861***
				(0.042)	(0.067)
σ_γ			1.994***		1.017***
			(0.041)		(0.075)
ρ		0.493***	−0.015	−0.024	−0.065*
		(0.014)	(0.014)	(0.016)	(0.031)
$\log \mathcal{L}$	−53.553	−5538.018	−4231.078	−3900.261	−3784.119
N	1529	1943	1943	1943	1943
$N \cdot T$	9836	13752	13752	13752	13752
Implied correlation		0.798	−0.012	−0.044	−0.075

This table shows selected estimates for the wage model (W) and four selection models. S^{REP} contains a random effect in the participation equation, S^{SD} allows for state dependence, and ^{REP,SD} allows for both.

All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.

The models S^{SD} and ^{REP,SD} have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.

Standard errors are given in parentheses.

Table 2.7 – Selected estimates for base models for males with A-Levels

	W	S^{Base}	S^{REP}	S^{SD}	$S^{REP,SD}$
Married		0.190***	0.222*	0.186**	0.248*
		(0.056)	(0.109)	(0.072)	(0.108)
Coupled		−0.188**	0.023	−0.102	−0.051
		(0.067)	(0.121)	(0.088)	(0.127)
Has kids aged 0 – 2		−0.028	−0.131	−0.005	−0.043
		(0.069)	(0.102)	(0.091)	(0.114)
Has kids aged 3 – 4		−0.101	−0.092	−0.122	−0.143
		(0.069)	(0.096)	(0.087)	(0.108)
Has kids aged 5 – 11		0.013	−0.017	0.080	0.040
		(0.051)	(0.081)	(0.065)	(0.087)
Has kids aged 12 – 15		−0.096*	−0.072	−0.044	−0.038
		(0.056)	(0.085)	(0.071)	(0.091)
Spouse has job		0.179*	0.198	0.150	0.076
		(0.102)	(0.160)	(0.131)	(0.172)
Log-wage for spouse		−0.104**	0.042	−0.013	0.032
		(0.039)	(0.053)	(0.042)	(0.054)
Log hrs/week spouse		0.150***	0.083*	0.053	0.075
		(0.032)	(0.049)	(0.040)	(0.052)
Owns house period 1		−0.048	0.631***	0.111*	0.296**
		(0.046)	(0.142)	(0.059)	(0.113)
Lagged participation				2.209***	1.479***
				(0.049)	(0.075)
σ_γ			1.678***		0.966***
			(0.072)		(0.080)
ρ		0.266***	0.058*	0.004	−0.001
		(0.010)	(0.034)	(0.015)	(0.028)
$\log \mathcal{L}$	−278.761	−4297.736	−3398.629	−3245.505	−3152.567
N	1342	1532	1532	1532	1532
$N \cdot T$	9048	11075	11075	11075	11075
Implied correlation		0.543	0.062	0.008	−0.001

This table shows selected estimates for the wage model (W) and four selection models. S^{REP} contains a random effect in the participation equation, S^{SD} allows for state dependence, and ^{REP,SD} allows for both.

All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.

The models S^{SD} and ^{REP,SD} have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.

Standard errors are given in parentheses.

Table 2.8 – Selected estimates for base models for males with Higher education

	W	S^{Base}	S^{REP}	S^{SD}	$S^{REP,SD}$
Married		−0.030	−0.203	−0.085	−0.179
		(0.092)	(0.157)	(0.122)	(0.164)
Coupled		−0.173	−0.074	−0.173	−0.167
		(0.114)	(0.187)	(0.155)	(0.203)
Has kids aged 0 – 2		0.040	0.000	−0.049	−0.027
		(0.114)	(0.150)	(0.136)	(0.160)
Has kids aged 3 – 4		−0.047	−0.105	0.090	0.092
		(0.111)	(0.144)	(0.140)	(0.164)
Has kids aged 5 – 11		−0.026	−0.098	0.012	−0.014
		(0.083)	(0.121)	(0.102)	(0.129)
Has kids aged 12 – 15		0.023	−0.023	0.065	0.023
		(0.092)	(0.129)	(0.113)	(0.140)
Spouse has job		−0.062	−0.080	−0.205	−0.207
		(0.139)	(0.203)	(0.168)	(0.212)
Log-wage for spouse		0.017	0.070	0.072	0.099*
		(0.039)	(0.052)	(0.045)	(0.055)
Log hrs/week spouse		0.095*	0.091	0.079	0.091
		(0.041)	(0.059)	(0.050)	(0.062)
Owns house period 1		−0.008	0.253	−0.100	−0.051
		(0.069)	(0.158)	(0.092)	(0.140)
Lagged participation				1.833***	1.395***
				(0.086)	(0.116)
σ_γ			1.169***		0.689***
			(0.087)		(0.102)
ρ		0.220***	0.151***	0.085***	0.106**
		(0.020)	(0.043)	(0.023)	(0.035)
$\log \mathcal{L}$	−387.268	−1786.572	−1598.455	−1510.499	−1490.213
N	597	643	643	643	643
$N \cdot T$	3893	4491	4491	4491	4491
Implied correlation		0.396	0.180	0.155	0.152

This table shows selected estimates for the wage model (W) and four selection models. S^{REP} contains a random effect in the participation equation, S^{SD} allows for state dependence, and ^{REP,SD} allows for both.

All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.

The models S^{SD} and ^{REP,SD} have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.

Standard errors are given in parentheses.

Table 2.9 – Selected estimates for base models for females with O-Levels

	W	S^{Base}	S^{REP}	S^{SD}	$S^{REP,SD}$
Married		0.025	−0.115	−0.024	−0.067
		(0.040)	(0.072)	(0.048)	(0.067)
Coupled		−0.325***	−0.305***	−0.283***	−0.322***
		(0.051)	(0.084)	(0.061)	(0.083)
Has kids aged 0 – 2		−0.843***	−1.144***	−0.711***	−0.868***
		(0.042)	(0.059)	(0.054)	(0.065)
Has kids aged 3 – 4		−0.563***	−0.815***	−0.237***	−0.424***
		(0.039)	(0.054)	(0.051)	(0.061)
Has kids aged 5 – 11		−0.219***	−0.480***	−0.134***	−0.227***
		(0.029)	(0.044)	(0.036)	(0.046)
Has kids aged 12 – 15		0.107***	−0.003	−0.004	−0.026
		(0.030)	(0.043)	(0.037)	(0.047)
Spouse has job		0.522***	0.724***	0.491***	0.624***
		(0.049)	(0.079)	(0.060)	(0.082)
Log-wage for spouse		−0.021	0.036*	−0.003	0.013
		(0.014)	(0.020)	(0.017)	(0.021)
Log hrs/week spouse		0.038**	0.024	0.021	0.027
		(0.013)	(0.020)	(0.016)	(0.021)
Owns house period 1		0.013	0.919***	0.105***	0.342***
		(0.026)	(0.093)	(0.032)	(0.061)
Lagged participation				2.361***	1.789***
				(0.031)	(0.042)
σ_γ			1.841***		0.873***
			(0.047)		(0.047)
ρ		0.560***	0.014	0.104***	0.065**
		(0.010)	(0.046)	(0.013)	(0.023)
$\log \mathcal{L}$	−918.205	−10983.954	−9065.976	−8242.910	−8039.648
N	1944	2599	2599	2599	2599
$N \cdot T$	13649	21677	21677	21677	21677
Implied correlation		0.842	0.012	0.185	0.083

This table shows selected estimates for the wage model (W) and four selection models. S^{REP} contains a random effect in the participation equation, S^{SD} allows for state dependence, and ^{REP,SD} allows for both.

All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.

The models S^{SD} and ^{REP,SD} have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.

Standard errors are given in parentheses.

Table 2.10 – Selected estimates for base models for females with A-Levels

	W	S^{Base}	S^{REP}	S^{SD}	$S^{REP,SD}$
Married		-0.102^{*}	-0.293^{**}	-0.091	-0.166^{*}
		(0.055)	(0.106)	(0.071)	(0.095)
Coupled		-0.310^{***}	-0.315^{*}	-0.234^{**}	-0.270^{*}
		(0.078)	(0.136)	(0.099)	(0.131)
Has kids aged 0 – 2		-0.789^{***}	-1.117^{***}	-0.581^{***}	-0.743^{***}
		(0.051)	(0.074)	(0.066)	(0.079)
Has kids aged 3 – 4		-0.601^{***}	-0.815^{***}	-0.266^{***}	-0.420^{***}
		(0.050)	(0.069)	(0.066)	(0.077)
Has kids aged 5 – 11		-0.286^{***}	-0.655^{***}	-0.166^{***}	-0.318^{***}
		(0.041)	(0.066)	(0.052)	(0.068)
Has kids aged 12 – 15		0.062	0.027	-0.017	-0.023
		(0.050)	(0.075)	(0.064)	(0.079)
Spouse has job		0.431^{***}	0.458^{***}	0.461^{***}	0.576^{***}
		(0.083)	(0.135)	(0.108)	(0.140)
Log-wage for spouse		-0.033^{*}	0.040	0.008	0.031
		(0.018)	(0.028)	(0.021)	(0.026)
Log hrs/week spouse		0.063^{***}	0.017	-0.021	-0.030
		(0.020)	(0.032)	(0.026)	(0.032)
Owns house period 1		-0.037	0.376^{**}	0.028	0.101
		(0.042)	(0.128)	(0.052)	(0.085)
Lagged participation				2.317^{***}	1.838^{***}
				(0.047)	(0.066)
σ_γ			1.652^{***}		0.721^{***}
			(0.070)		(0.065)
ρ		0.375^{***}	0.091^{**}	0.057^{***}	0.065^{**}
		(0.011)	(0.036)	(0.016)	(0.024)
$\log \mathcal{L}$	-1009.534	-5385.392	-4506.126	-4108.965	-4050.322
N	1123	1311	1311	1311	1311
$N \cdot T$	7240	9760	9760	9760	9760
Implied correlation		0.669	0.089	0.109	0.098

This table shows selected estimates for the wage model (W) and four selection models. S^{REP} contains a random effect in the participation equation, S^{SD} allows for state dependence, and ^{REP,SD} allows for both.

All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.

The models S^{SD} and ^{REP,SD} have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.

Standard errors are given in parentheses.

Table 2.11 – Selected estimates for base models for females with Higher education

	W	S^{Base}	S^{REP}	S^{SD}	$S^{REP,SD}$
Married		−0.094	0.019	0.111	0.136
		(0.074)	(0.153)	(0.095)	(0.137)
Coupled		−0.186	−0.205	−0.236	−0.274
		(0.116)	(0.200)	(0.155)	(0.204)
Has kids aged 0 – 2		−0.585***	−0.818***	−0.583***	−0.749***
		(0.078)	(0.116)	(0.099)	(0.122)
Has kids aged 3 – 4		−0.661***	−0.884***	−0.404***	−0.628***
		(0.078)	(0.112)	(0.098)	(0.121)
Has kids aged 5 – 11		−0.261***	−0.439***	−0.117	−0.201*
		(0.070)	(0.116)	(0.088)	(0.117)
Has kids aged 12 – 15		0.104	−0.116	0.042	−0.042
		(0.086)	(0.129)	(0.110)	(0.138)
Spouse has job		0.020	−0.274	0.018	−0.077
		(0.124)	(0.198)	(0.163)	(0.211)
Log-wage for spouse		−0.104**	−0.014	−0.054	−0.030
		(0.035)	(0.049)	(0.042)	(0.051)
Log hrs/week spouse		0.152***	0.144**	0.079*	0.089*
		(0.033)	(0.049)	(0.042)	(0.052)
Owns house period 1		−0.052	0.245	0.039	0.068
		(0.064)	(0.196)	(0.083)	(0.139)
Lagged participation				2.125***	1.560***
				(0.074)	(0.103)
σ_γ			1.421***		0.782***
			(0.102)		(0.097)
ρ		0.306***	0.068	0.059**	0.040
		(0.018)	(0.063)	(0.024)	(0.039)
$\log \mathcal{L}$	−437.097	−2196.915	−1852.781	−1730.186	−1694.545
N	540	604	604	604	604
$N \cdot T$	3301	4119	4119	4119	4119
Implied correlation		0.520	0.068	0.103	0.051

This table shows selected estimates for the wage model (W) and four selection models. S^{REP} contains a random effect in the participation equation, S^{SD} allows for state dependence, and ^{REP,SD} allows for both.

All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.

The models S^{SD} and ^{REP,SD} have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.

Standard errors are given in parentheses.

2.C Figures with moments for base models

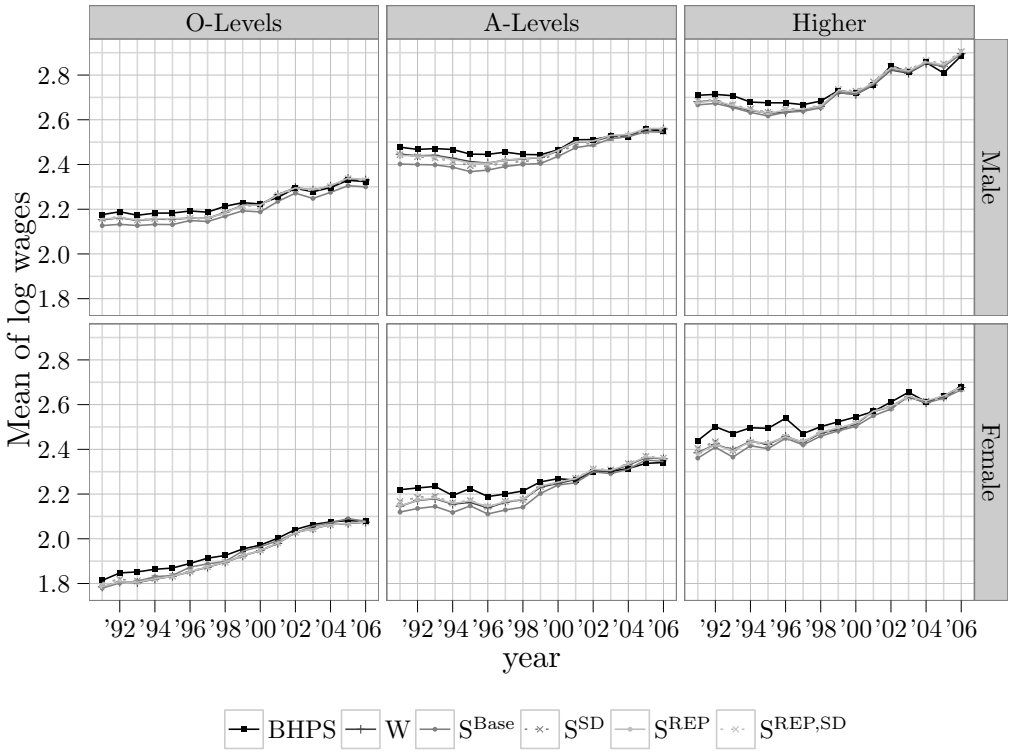

Figure 2.13 – Observed and simulated mean of log-wages by education group and gender

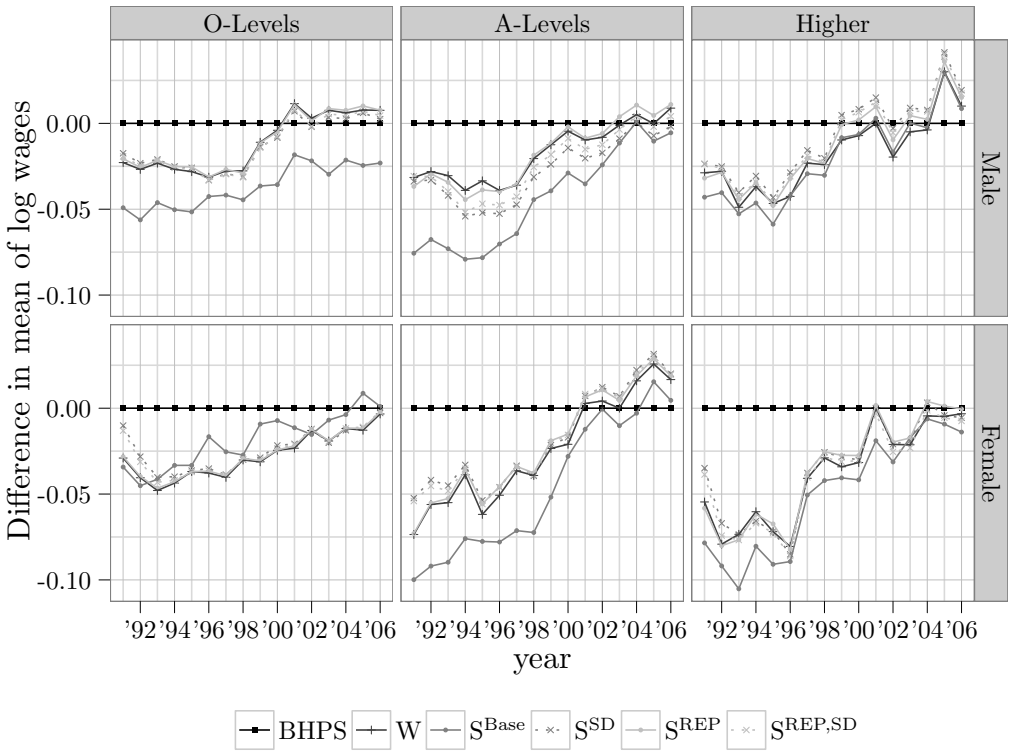

Figure 2.14 – Difference between simulated and observed mean of log-wages by education group and gender

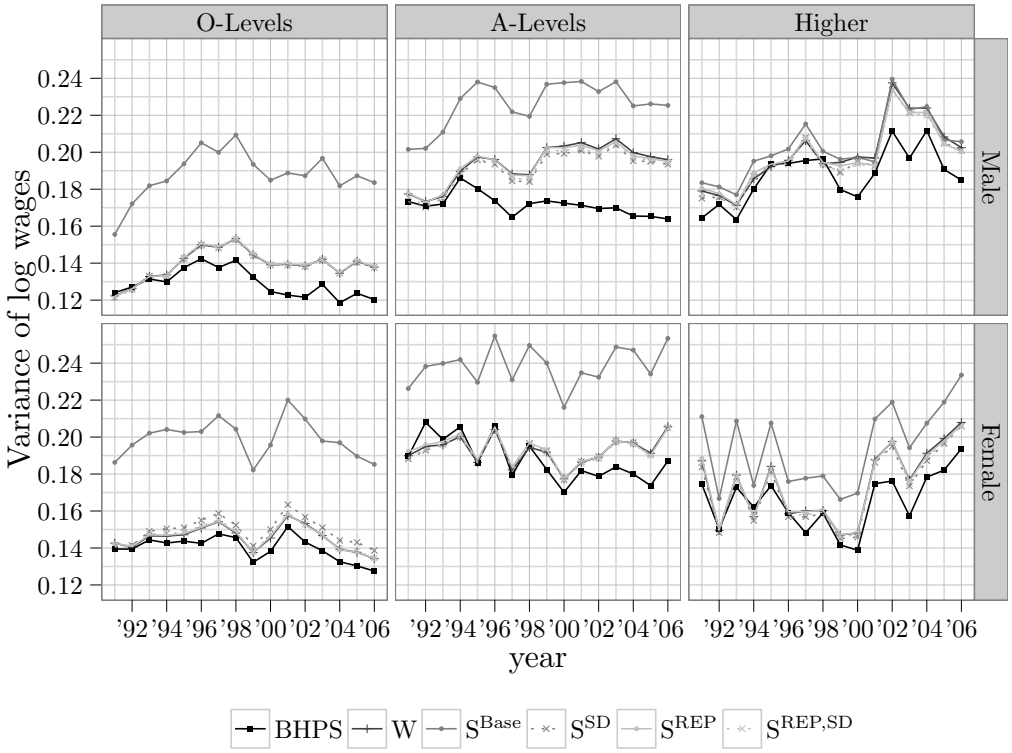

Figure 2.15 – Observed and simulated variance of log-wages by education group and gender

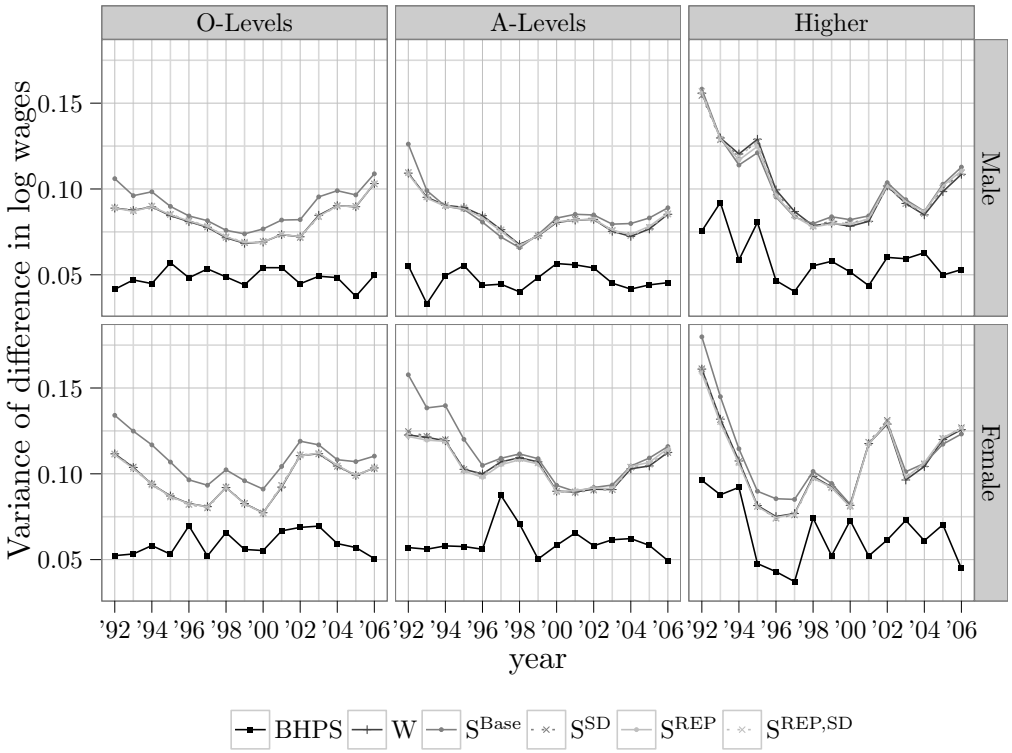

Figure 2.16 – Observed and simulated variance of difference in log-wages by education group and gender

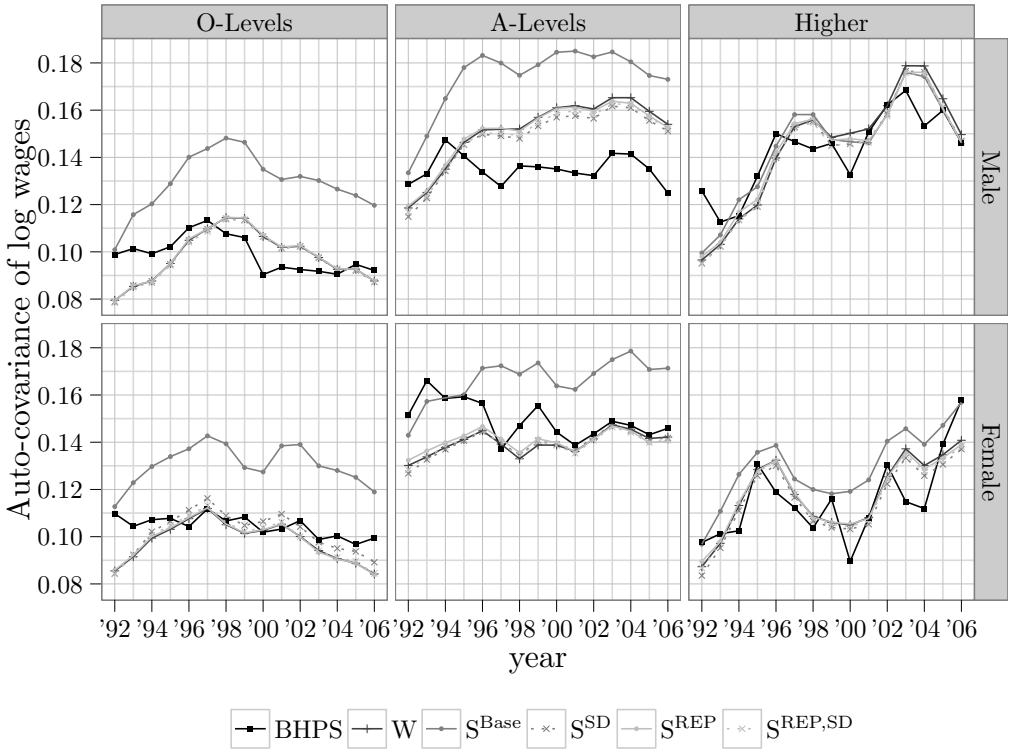

Figure 2.17 – Observed and simulated auto-covariance of log-wages by education group and gender

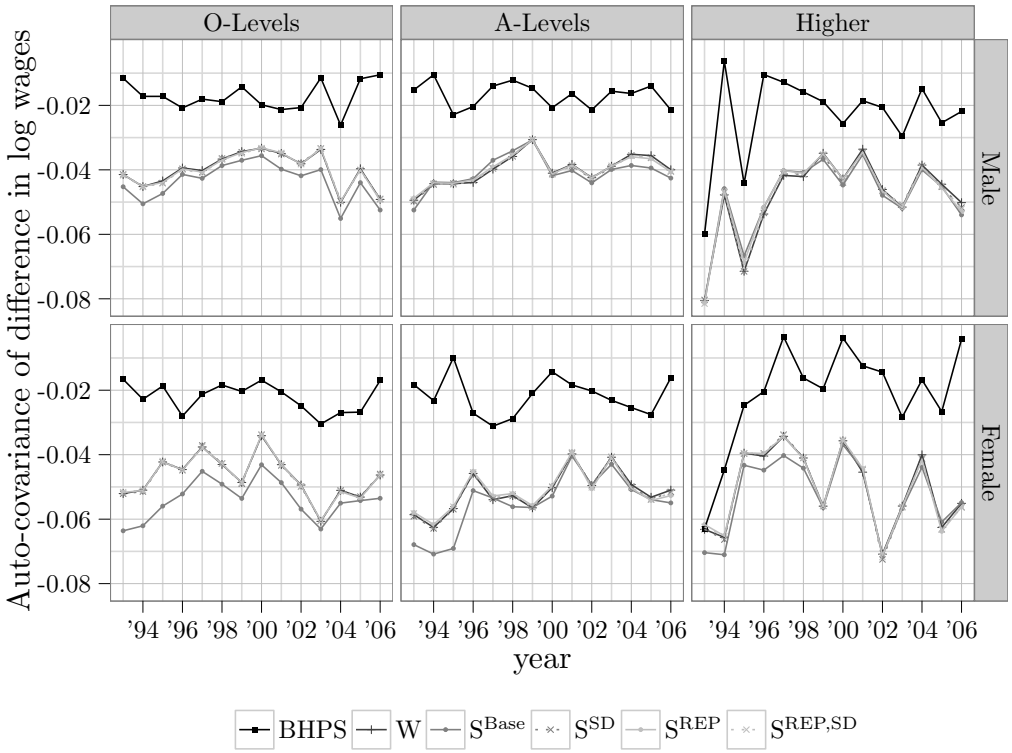

Figure 2.18 – Observed and simulated auto-covariance of difference in log-wages by education group and gender

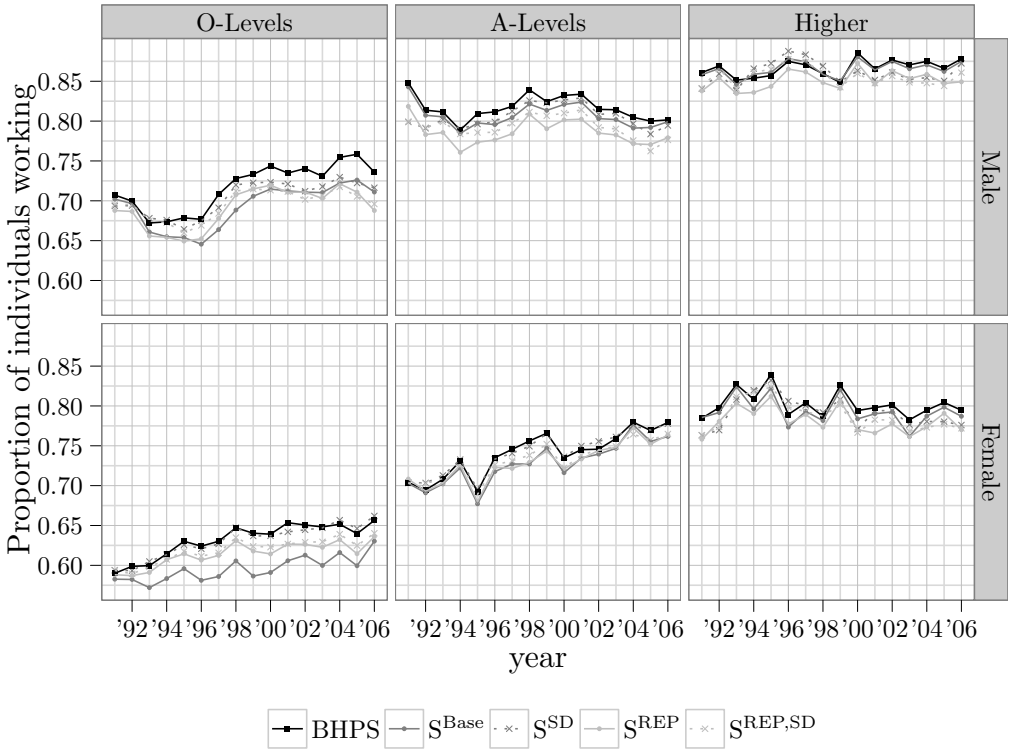

Figure 2.19 – Observed and simulated participation by education group and gender

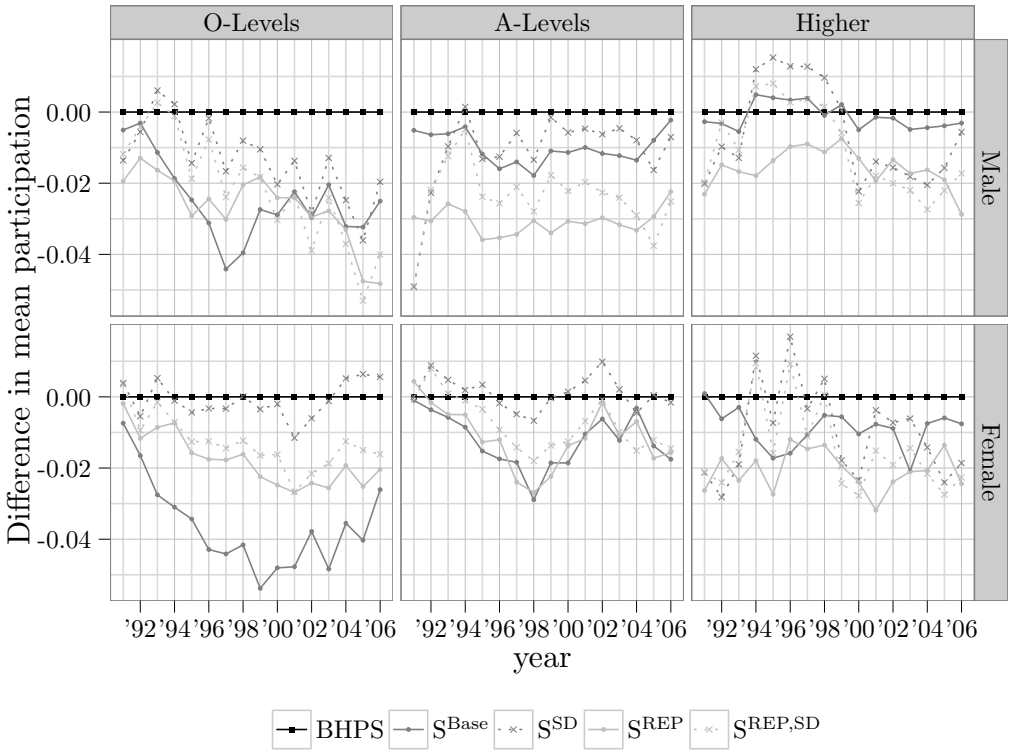

Figure 2.20 – Difference between simulated and observed participation by education group and gender

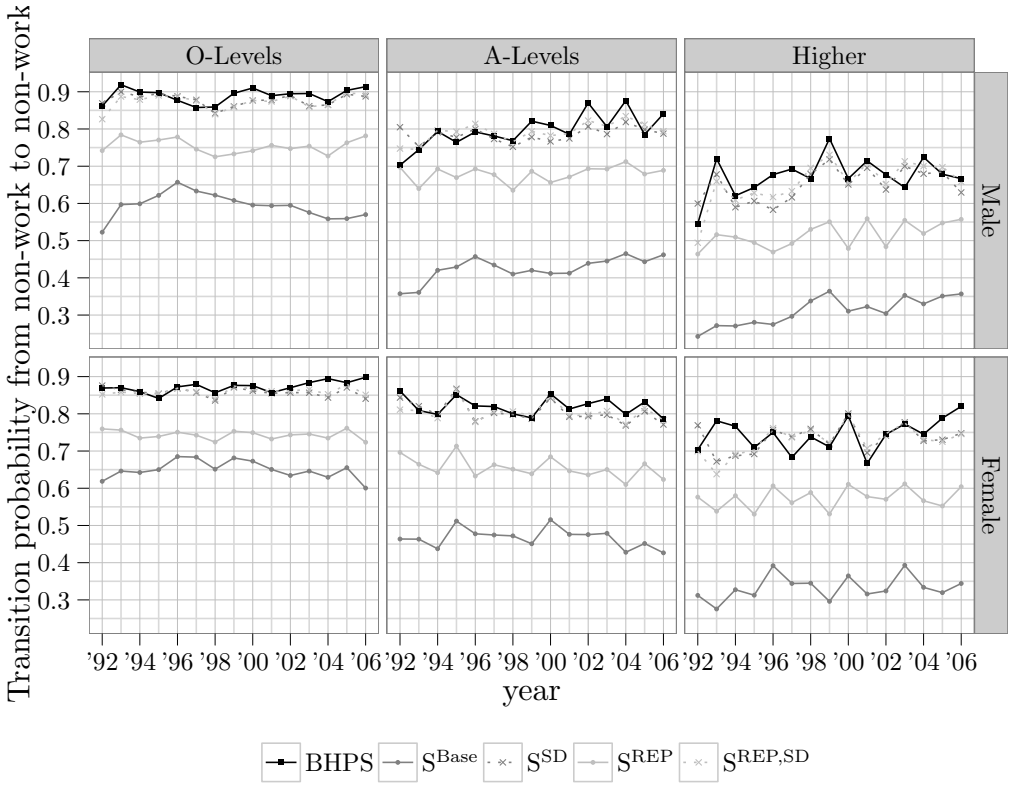

Figure 2.21 – Observed and simulated transition probability from non-work to non-work

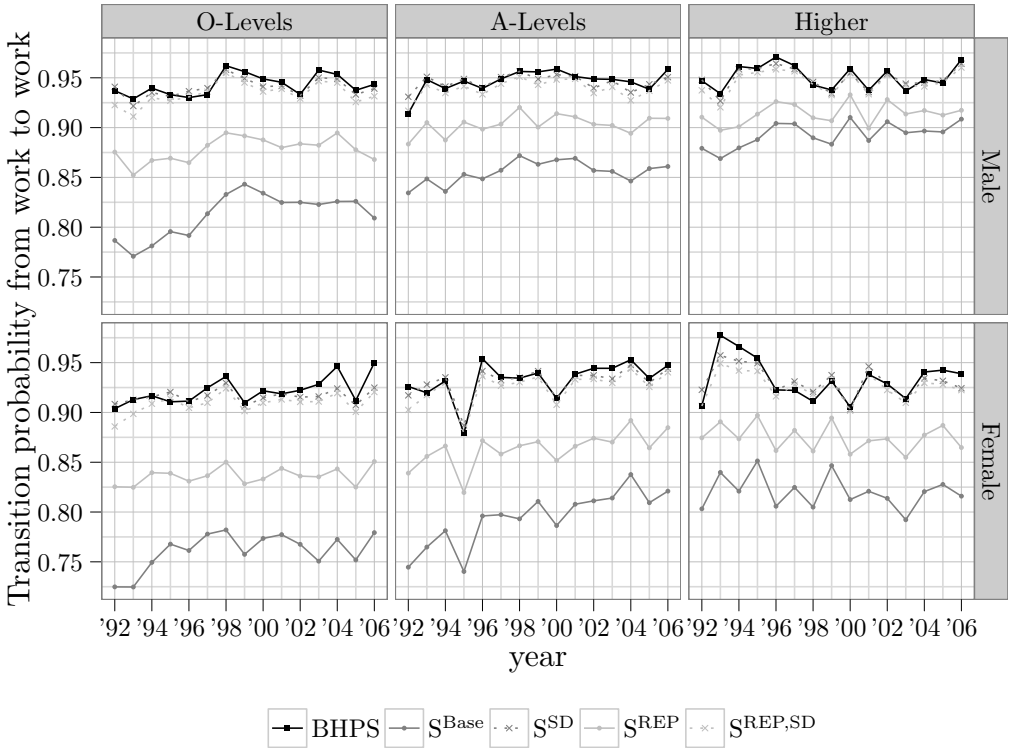

Figure 2.22 – Observed and simulated transition probability from work to work

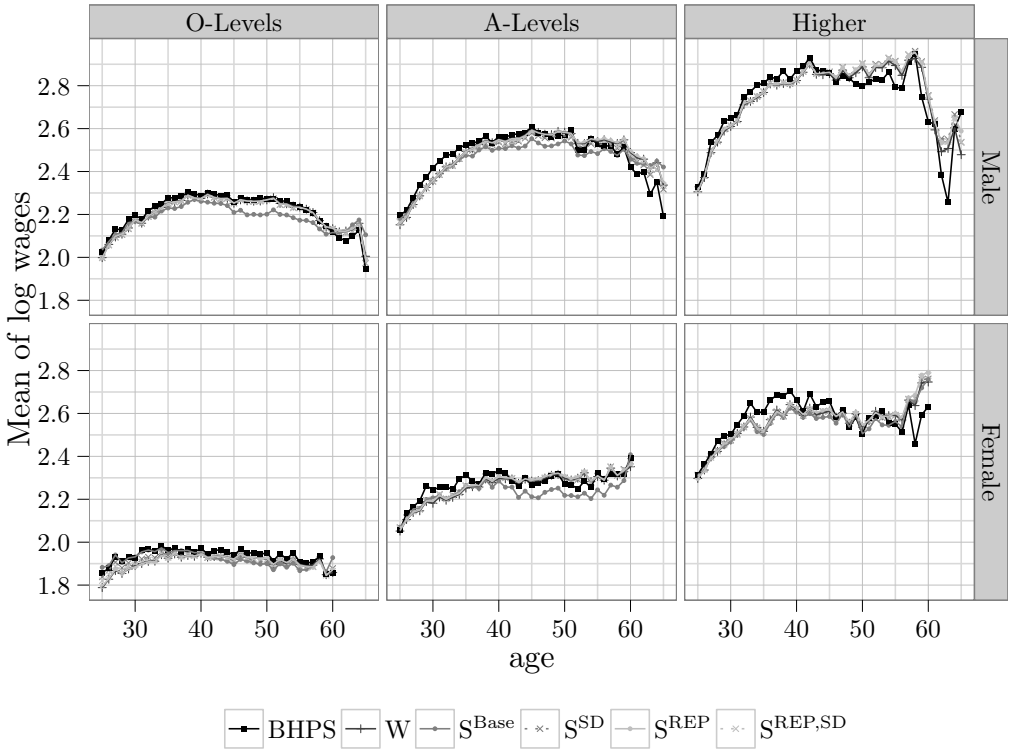

Figure 2.23 – Observed and simulated mean of log-wages by education group and gender

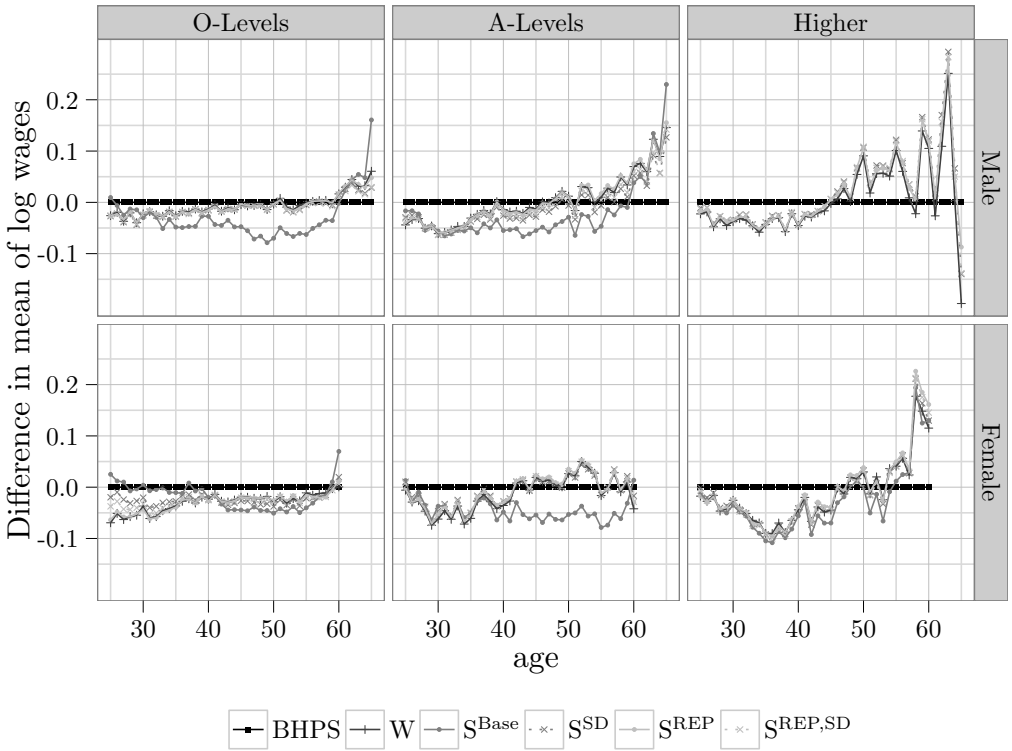

Figure 2.24 – Difference between simulated and observed mean of log-wages by education group and gender

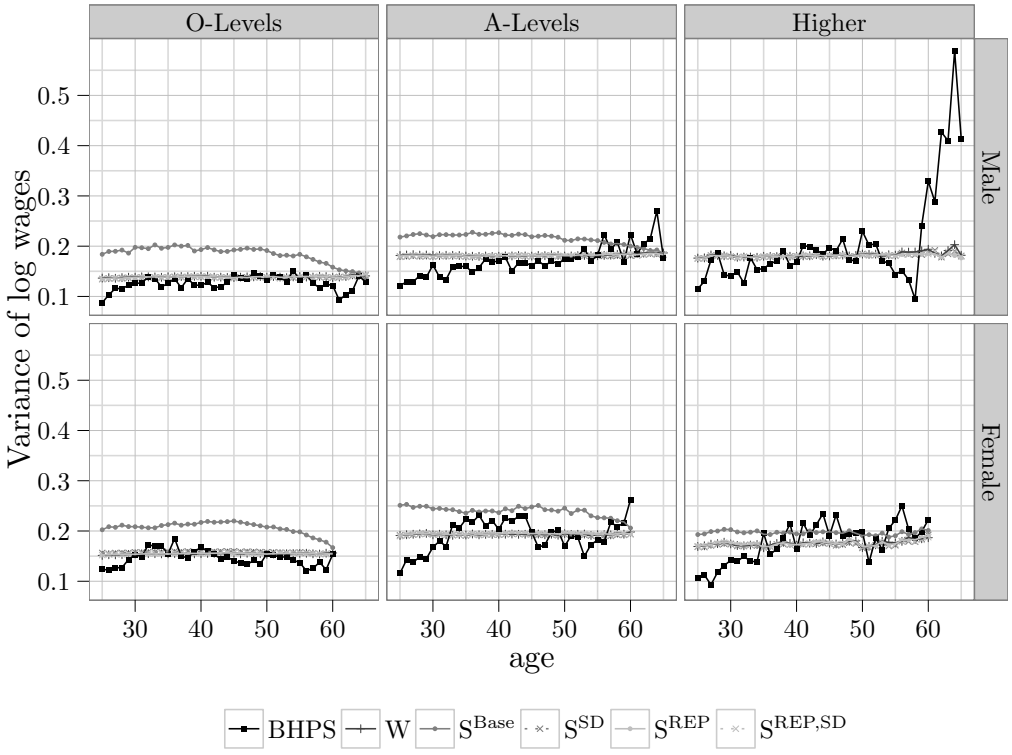

Figure 2.25 – Observed and simulated variance of log-wages by education group and gender

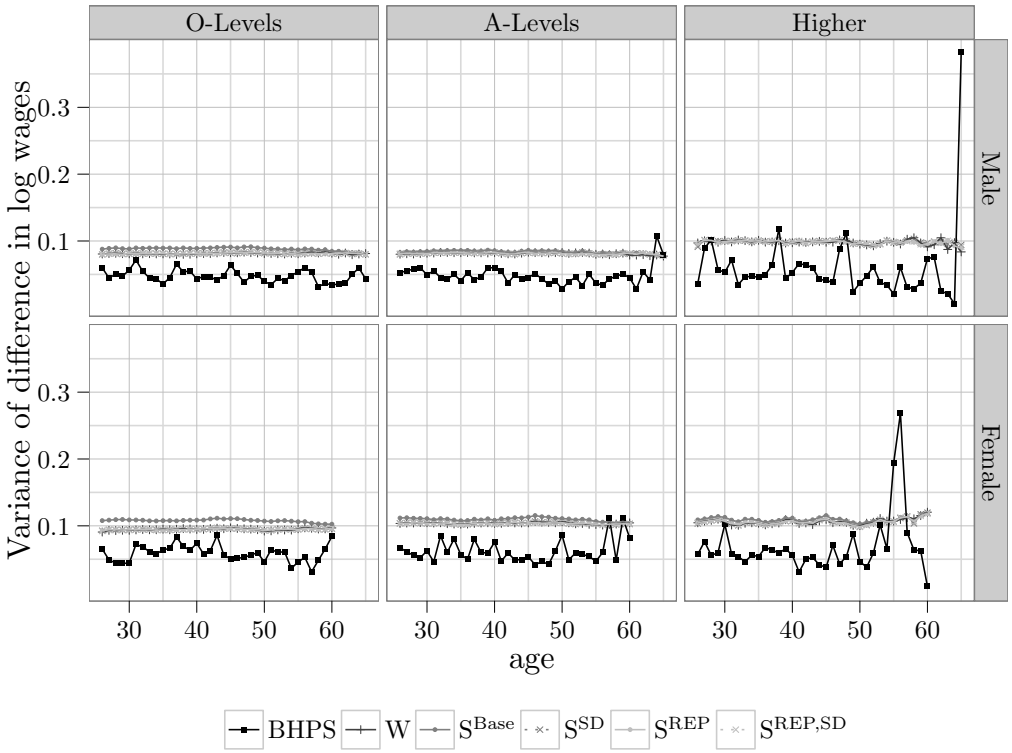

Figure 2.26 – Observed and simulated variance of difference in log-wages by education group and gender

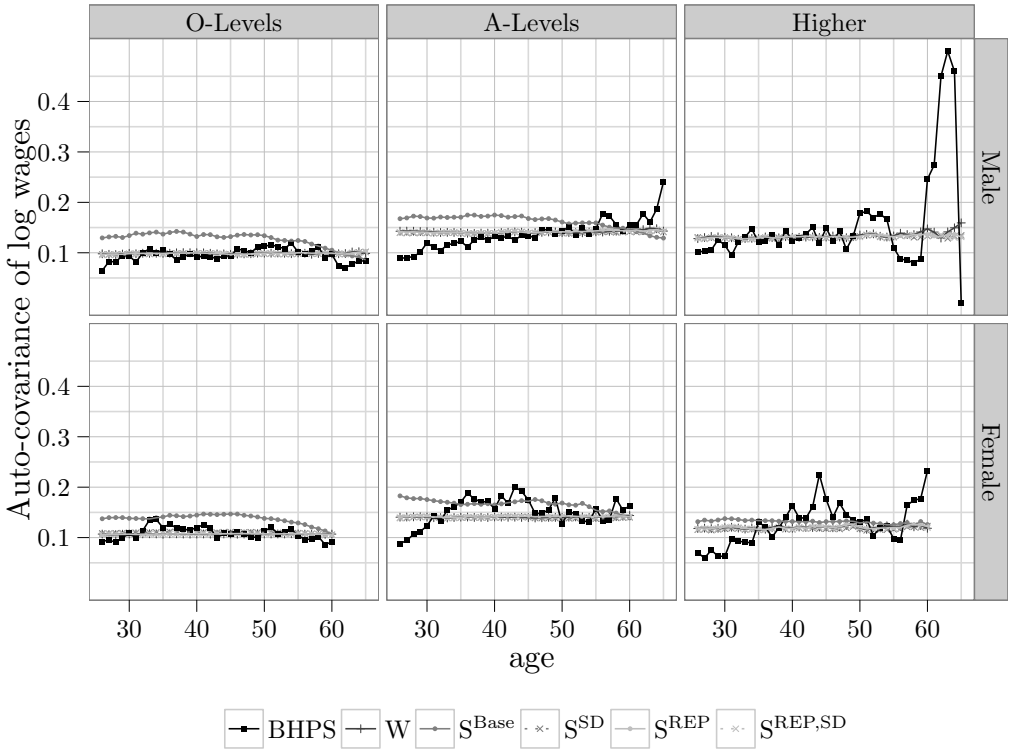

Figure 2.27 – Observed and simulated auto-covariance of log-wages by education group and gender

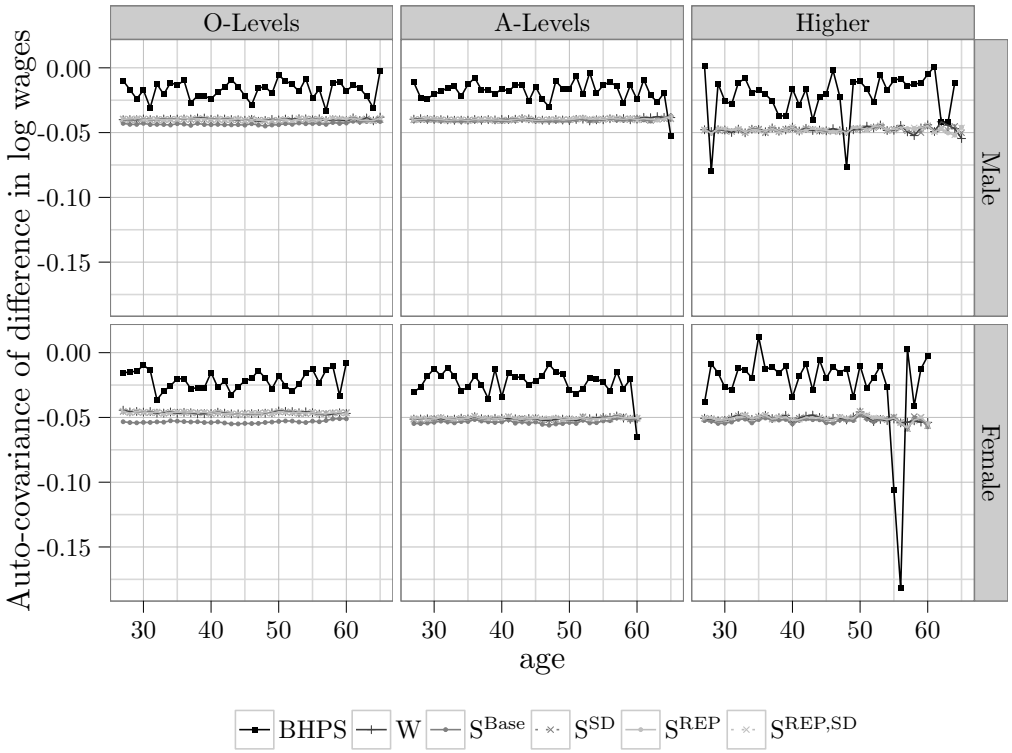

Figure 2.28 – Observed and simulated auto-covariance of difference in log-wages by education group and gender

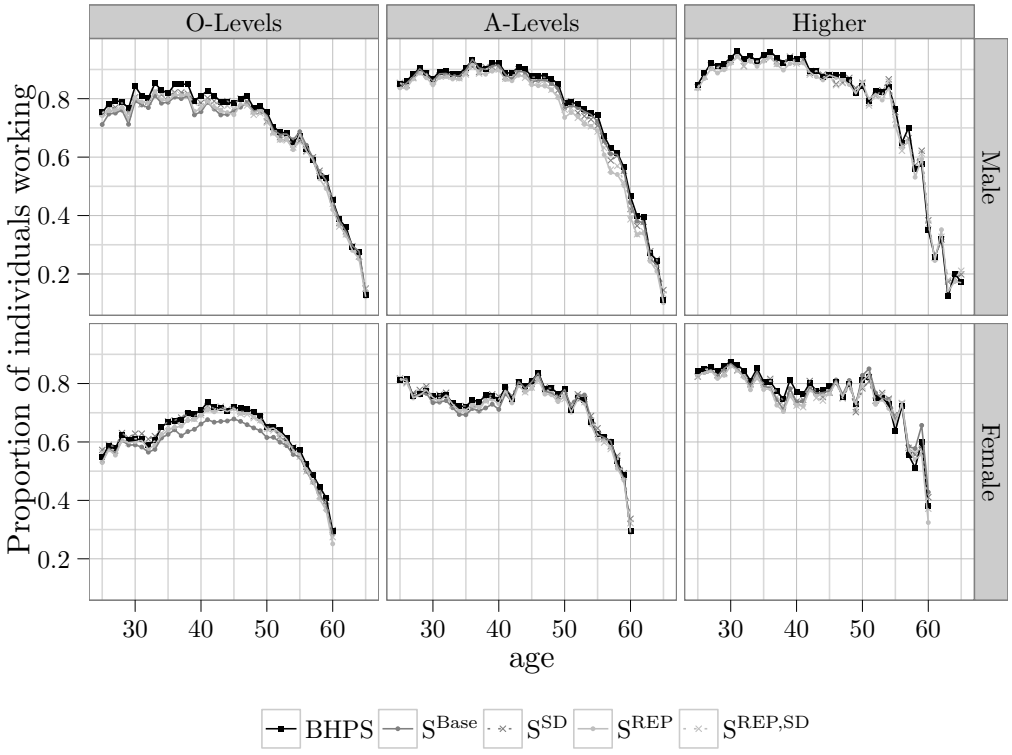

Figure 2.29 – Observed and simulated participation by education group and gender

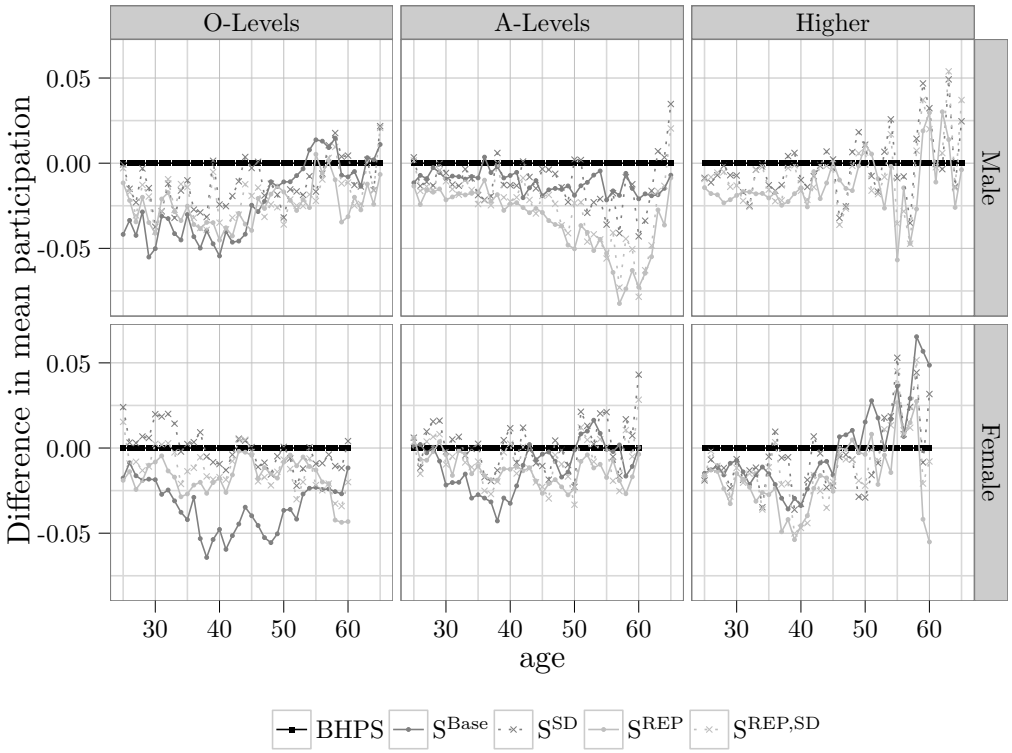

Figure 2.30 – Difference between simulated and observed participation by education group and gender

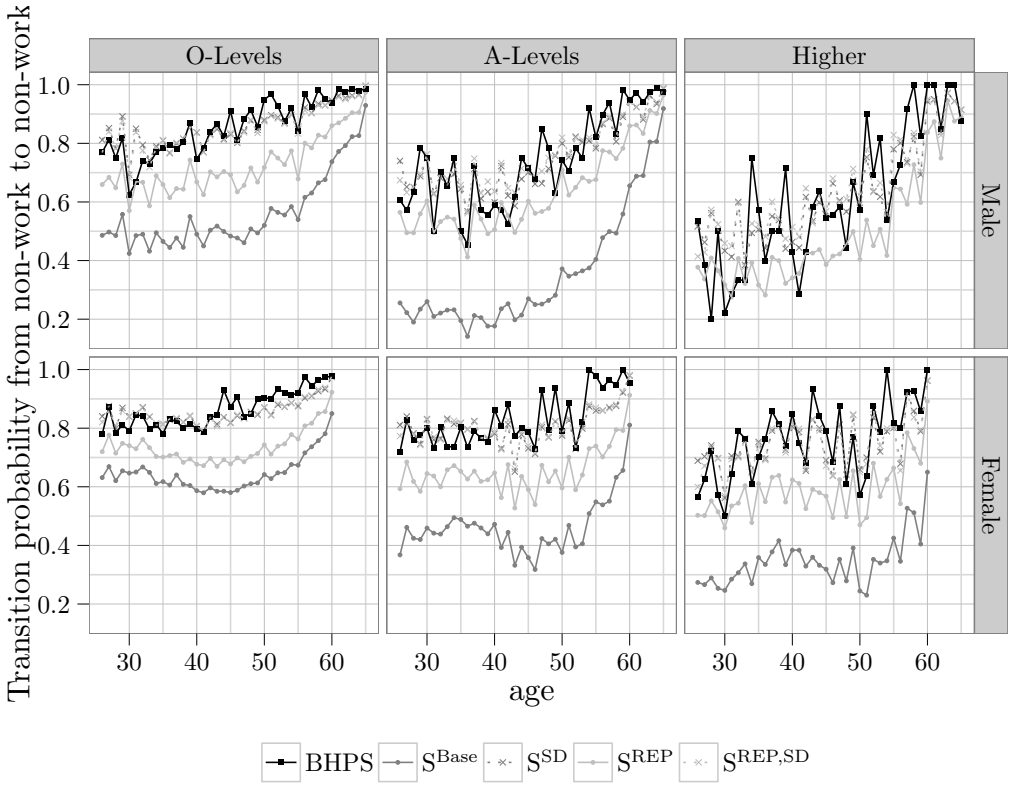

Figure 2.31 – Observed and simulated transition probability from non-work to non-work

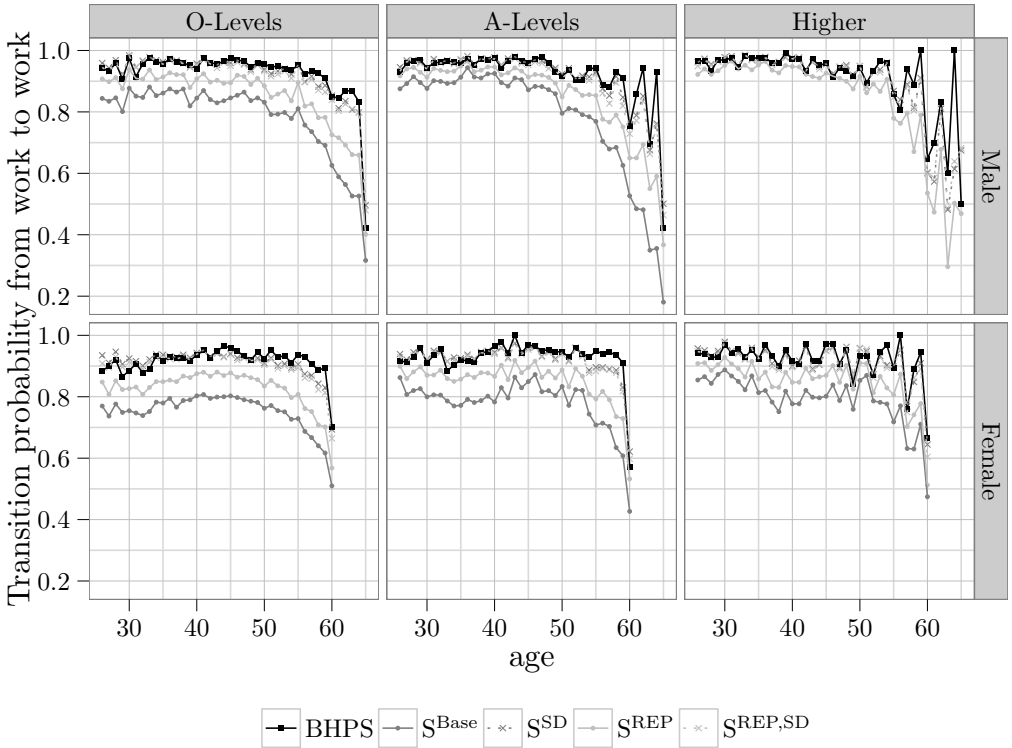

Figure 2.32 – Observed and simulated transition probability from work to work

2.D Estimation results: mixture distribution

Table 2.12 – Selected estimates for mixture distribution models for males

	O-Levels		A-Levels		Higher	
	W^{mix}	$S^{SD,mix}$	W^{mix}	$S^{SD,mix}$	W^{mix}	$S^{SD,mix}$
Married		−0.017		0.186**		−0.084
		(0.065)		(0.072)		(0.122)
Coupled		0.001		−0.102		−0.171
		(0.079)		(0.088)		(0.155)
Has kids aged 0 – 2		−0.192**		−0.006		−0.049
		(0.078)		(0.091)		(0.136)
Has kids aged 3 – 4		0.160*		−0.122		0.089
		(0.081)		(0.087)		(0.140)
Has kids aged 5 – 11		−0.113*		0.080		0.011
		(0.055)		(0.065)		(0.102)
Has kids aged 12 – 15		−0.093		−0.044		0.064
		(0.060)		(0.071)		(0.113)
Spouse has job		0.225*		0.150		−0.206
		(0.114)		(0.131)		(0.168)
Log-wage for spouse		0.026		−0.013		0.072
		(0.035)		(0.042)		(0.045)
Log hrs/week spouse		0.073*		0.052		0.078
		(0.035)		(0.040)		(0.050)
Owns house period 1		0.236***		0.112*		−0.100
		(0.043)		(0.059)		(0.092)
Lagged participation		2.604***		2.209***		1.833***
		(0.042)		(0.049)		(0.086)
$\mu_{1,\alpha}$	0.476	0.604**	0.465***	0.502***	0.012	0.010
	(0.309)	(0.230)	(0.077)	(0.074)	(0.041)	(0.042)
$\sigma_{1,\alpha}$	0.973***	0.938***	1.012***	1.024***	1.157***	1.157***
	(0.094)	(0.072)	(0.065)	(0.068)	(0.063)	(0.062)
$\mu_{2,\alpha}$	−0.656***	−0.619***	−0.413***	−0.391***	−0.025	−0.020
	(0.106)	(0.105)	(0.127)	(0.122)	(0.085)	(0.086)
$\sigma_{2,\alpha}$	0.576***	0.605***	0.784***	0.783***	0.537***	0.539***
	(0.110)	(0.068)	(0.058)	(0.059)	(0.124)	(0.119)
$\pi_{1,\alpha}$	0.580**	0.506***	0.471***	0.438***	0.677***	0.676***
	(0.189)	(0.131)	(0.083)	(0.080)	(0.122)	(0.120)
ρ		−0.030*		0.003		0.084***
		(0.016)		(0.014)		(0.023)
$\log \mathcal{L}$	−27.876	−3873.496	−270.996	−3235.938	−382.473	−1505.655
N	1529	1943	1342	1532	597	643
$N \cdot T$	9836	13752	9048	11075	3893	4491

This table shows selected estimates for models where α_i follows a mixture of two normal distributions. All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

Table 2.13 – Selected estimates for mixture distribution models for females

	O-Levels		A-Levels		Higher	
	W^{mix}	$S^{SD,mix}$	W^{mix}	$S^{SD,mix}$	W^{mix}	$S^{SD,mix}$
Married		−0.030		−0.091		0.111
		(0.048)		(0.071)		(0.095)
Coupled		−0.281***		−0.233**		−0.237
		(0.061)		(0.099)		(0.155)
Has kids aged 0 – 2		−0.705***		−0.581***		−0.584***
		(0.054)		(0.066)		(0.100)
Has kids aged 3 – 4		−0.233***		−0.266***		−0.405***
		(0.051)		(0.066)		(0.099)
Has kids aged 5 – 11		−0.139***		−0.165***		−0.116
		(0.036)		(0.052)		(0.089)
Has kids aged 12 – 15		−0.011		−0.016		0.042
		(0.037)		(0.064)		(0.110)
Spouse has job		0.491***		0.461***		0.018
		(0.060)		(0.108)		(0.163)
Log-wage for spouse		−0.001		0.008		−0.054
		(0.017)		(0.021)		(0.042)
Log hrs/week spouse		0.020		−0.021		0.080*
		(0.016)		(0.026)		(0.042)
Owns house period 1		0.114***		0.027		0.038
		(0.031)		(0.052)		(0.083)
Lagged participation		2.390***		2.315***		2.122***
		(0.030)		(0.047)		(0.075)
$\mu_{1,\alpha}$	−0.750***	−0.671***	0.417*	0.441*	0.282**	0.336*
	(0.065)	(0.057)	(0.196)	(0.202)	(0.104)	(0.151)
$\sigma_{1,\alpha}$	0.431***	0.444***	0.791***	0.793***	0.533***	0.500***
	(0.076)	(0.075)	(0.092)	(0.093)	(0.131)	(0.140)
$\mu_{2,\alpha}$	0.405***	0.370***	−1.094***	−1.052***	−0.167*	−0.165*
	(0.101)	(0.094)	(0.255)	(0.235)	(0.076)	(0.076)
$\sigma_{2,\alpha}$	0.986***	1.028***	0.575***	0.562***	1.161***	1.134***
	(0.028)	(0.027)	(0.111)	(0.110)	(0.063)	(0.069)
$\pi_{1,\alpha}$	0.351***	0.355***	0.724***	0.705***	0.372**	0.329**
	(0.071)	(0.067)	(0.139)	(0.140)	(0.125)	(0.128)
ρ		0.081***		0.059***		0.062**
		(0.013)		(0.016)		(0.025)
$\log \mathcal{L}$	−863.216	−8200.487	−1001.318	−4100.017	−430.855	−1724.075
N	1944	2599	1123	1311	540	604
$N \cdot T$	13649	21677	7240	9760	3301	4119

This table shows selected estimates for models where α_i follows a mixture of two normal distributions.
All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

2.E Estimation results: heterogeneous trend

Table 2.14 – Selected estimates for participation equation for males, O-Levels

	W^{age}	$W^{\sqrt{age}}$	$S^{SD,age}$	$S^{SD,\sqrt{age}}$
Married			−0.021	−0.022
			(0.065)	(0.065)
Coupled			−0.006	−0.008
			(0.079)	(0.079)
Has kids aged 0 – 2			−0.188**	−0.187**
			(0.078)	(0.078)
Has kids aged 3 – 4			0.161*	0.161*
			(0.081)	(0.081)
Has kids aged 5 – 11			−0.111*	−0.110*
			(0.055)	(0.055)
Has kids aged 12 – 15			−0.091	−0.091
			(0.060)	(0.060)
Spouse has job			0.224*	0.223*
			(0.113)	(0.113)
Log-wage for spouse			0.021	0.019
			(0.035)	(0.035)
Log hrs/week spouse			0.076*	0.077*
			(0.035)	(0.035)
Owns house period 1			0.223***	0.219***
			(0.043)	(0.043)
Lagged participation			2.601***	2.599***
			(0.042)	(0.042)
σ_β	0.042***	0.211***	0.042***	0.210***
	(0.001)	(0.004)	(0.001)	(0.004)
$\rho_{\alpha\beta}$	−0.597***	−0.801***	−0.597***	−0.803***
	(0.027)	(0.016)	(0.028)	(0.016)
ρ			−0.006	−0.000
			(0.016)	(0.016)
$\log \mathcal{L}$	−253.534	−186.543	−3594.102	−3659.352
N	1529	1529	1943	1943
$N \cdot T$	9836	9836	13752	13752

This table shows selected estimates for models with a heterogeneous trend in age or $\sqrt{age}$.
All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

Table 2.15 – Estimates males, A-Levels

	W^{age}	$W^{\sqrt{age}}$	$S^{SD,age}$	$S^{SD,\sqrt{age}}$
Married			0.179**	0.181**
			(0.072)	(0.072)
Coupled			−0.108	−0.107
			(0.088)	(0.088)
Has kids aged 0 – 2			−0.003	−0.003
			(0.092)	(0.092)
Has kids aged 3 – 4			−0.124	−0.123
			(0.087)	(0.087)
Has kids aged 5 – 11			0.080	0.080
			(0.066)	(0.066)
Has kids aged 12 – 15			−0.048	−0.047
			(0.071)	(0.071)
Spouse has job			0.142	0.144
			(0.131)	(0.131)
Log-wage for spouse			−0.020	−0.018
			(0.043)	(0.043)
Log hrs/week spouse			0.060	0.059
			(0.040)	(0.040)
Owns house period 1			0.084	0.091
			(0.059)	(0.059)
Lagged participation			2.187***	2.194***
			(0.050)	(0.050)
σ_β	0.050***	0.228***	0.050***	0.226***
	(0.002)	(0.006)	(0.002)	(0.006)
$\rho_{\alpha\beta}$	−0.425***	−0.660***	−0.417***	−0.660***
	(0.029)	(0.023)	(0.030)	(0.028)
ρ			0.024*	0.020
			(0.012)	(0.013)
$\log \mathcal{L}$	−82.012	−46.133	−2877.310	−2915.548
N	1342	1342	1532	1532
$N \cdot T$	9048	9048	11075	11075

This table shows selected estimates for models with a heterogeneous trend in age or $\sqrt{age}$.
All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

Table 2.16 – Estimate males, Higher

	W^{age}	$W^{\sqrt{age}}$	$S^{SD,age}$	$S^{SD,\sqrt{age}}$
Married			−0.084	−0.084
			(0.122)	(0.122)
Coupled			−0.176	−0.174
			(0.155)	(0.155)
Has kids aged 0 – 2			−0.055	−0.053
			(0.136)	(0.136)
Has kids aged 3 – 4			0.082	0.084
			(0.140)	(0.140)
Has kids aged 5 – 11			0.011	0.011
			(0.102)	(0.102)
Has kids aged 12 – 15			0.064	0.064
			(0.114)	(0.114)
Spouse has job			−0.210	−0.213
			(0.169)	(0.168)
Log-wage for spouse			0.070	0.070
			(0.046)	(0.046)
Log hrs/week spouse			0.083*	0.082
			(0.050)	(0.050)
Owns house period 1			−0.111	−0.107
			(0.092)	(0.092)
Lagged participation			1.810***	1.818***
			(0.086)	(0.086)
σ_β	0.052***	0.228***	0.053***	0.242***
	(0.002)	(0.007)	(0.002)	(0.009)
$\rho_{\alpha\beta}$	−0.499***	−0.728***	−0.523***	−0.668***
	(0.032)	(0.031)	(0.031)	(0.033)
ρ			0.088***	0.085***
			(0.019)	(0.020)
$\log \mathcal{L}$	−228.077	−278.606	−1345.782	−1398.317
N	597	597	643	643
$N \cdot T$	3893	3893	4491	4491

This table shows selected estimates for models with a heterogeneous trend in age or $\sqrt{age}$.
All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

Table 2.17 – Estimates females, O-Levels

	W^{age}	$W^{\sqrt{age}}$	$S^{SD,age}$	$S^{SD,\sqrt{age}}$
Married			−0.010	−0.017
			(0.053)	(0.050)
Coupled			−0.293***	−0.289***
			(0.064)	(0.068)
Has kids aged 0 – 2			−0.727***	−0.721***
			(0.052)	(0.061)
Has kids aged 3 – 4			−0.246***	−0.236***
			(0.050)	(0.057)
Has kids aged 5 – 11			−0.120**	−0.123***
			(0.043)	(0.036)
Has kids aged 12 – 15			0.007	−0.001
			(0.049)	(0.041)
Spouse has job			0.497***	0.494***
			(0.063)	(0.067)
Log-wage for spouse			−0.006	−0.004
			(0.019)	(0.019)
Log hrs/week spouse			0.022	0.021
			(0.017)	(0.018)
Owns house period 1			0.084***	0.097***
			(0.027)	(0.030)
Lagged participation			2.281***	2.337***
			(0.040)	(0.033)
σ_β	0.042***	0.227***	0.043	0.211***
	(0.001)	(0.002)		(0.000)
$\rho_{\alpha\beta}$	−0.713***	−0.905***	−0.664	−0.901***
	(0.019)	(0.007)		(0.001)
ρ			0.136***	0.111
			(0.006)	
$\log \mathcal{L}$	−488.071	−528.745	−7772.715	−7826.058
N	1944	1944	2599	2599
$N \cdot T$	13649	13649	21677	21677

This table shows selected estimates for models with a heterogeneous trend in age or $\sqrt{age}$.
All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

Table 2.18 – Estimates females, A-Levels

	W^{age}	$W^{\sqrt{age}}$	$S^{SD,age}$	$S^{SD,\sqrt{age}}$
Married			−0.089	−0.089
			(0.071)	(0.071)
Coupled			−0.237**	−0.236**
			(0.100)	(0.100)
Has kids aged 0 – 2			−0.586***	−0.584***
			(0.066)	(0.066)
Has kids aged 3 – 4			−0.271***	−0.268***
			(0.066)	(0.066)
Has kids aged 5 – 11			−0.154**	−0.157**
			(0.052)	(0.052)
Has kids aged 12 – 15			−0.003	−0.007
			(0.064)	(0.064)
Spouse has job			0.455***	0.457***
			(0.109)	(0.109)
Log-wage for spouse			0.004	0.005
			(0.021)	(0.021)
Log hrs/week spouse			−0.015	−0.017
			(0.026)	(0.026)
Owns house period 1			0.017	0.020
			(0.053)	(0.053)
Lagged participation			2.271***	2.289***
			(0.049)	(0.048)
σ_β	0.057***	0.260***	0.060***	0.248***
	(0.003)	(0.010)	(0.003)	(0.009)
$\rho_{\alpha\beta}$	−0.404***	−0.584***	−0.379***	−0.609***
	(0.032)	(0.030)	(0.037)	(0.044)
ρ			0.077***	0.070***
			(0.014)	(0.014)
$\log \mathcal{L}$	−730.172	−780.990	−3823.333	−3878.932
N	1123	1123	1311	1311
$N \cdot T$	7240	7240	9760	9760

This table shows selected estimates for models with a heterogeneous trend in age or $\sqrt{age}$.
All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

Table 2.19 – Estimates females, Higher

	W^{age}	$W^{\sqrt{age}}$	$S^{SD,age}$	$S^{SD,\sqrt{age}}$
Married			0.104	0.108
			(0.095)	(0.095)
Coupled			−0.241	−0.239
			(0.156)	(0.155)
Has kids aged 0 – 2			−0.592***	−0.589***
			(0.100)	(0.100)
Has kids aged 3 – 4			−0.418***	−0.413***
			(0.099)	(0.099)
Has kids aged 5 – 11			−0.105	−0.110
			(0.089)	(0.089)
Has kids aged 12 – 15			0.048	0.046
			(0.110)	(0.110)
Spouse has job			0.020	0.019
			(0.163)	(0.163)
Log-wage for spouse			−0.056	−0.055
			(0.042)	(0.042)
Log hrs/week spouse			0.081*	0.081*
			(0.042)	(0.042)
Owns house period 1			0.019	0.027
			(0.083)	(0.083)
Lagged participation			2.082***	2.101***
			(0.076)	(0.075)
σ_β	0.063***	0.272***	0.060***	0.263***
	(0.004)	(0.014)	(0.005)	(0.014)
$\rho_{\alpha\beta}$	−0.387***	−0.621***	−0.371***	−0.644***
	(0.046)	(0.041)	(0.063)	(0.042)
ρ			0.086***	0.076***
			(0.021)	(0.022)
$\log \mathcal{L}$	−330.484	−357.297	−1621.808	−1649.110
N	540	540	604	604
$N \cdot T$	3301	3301	4119	4119

This table shows selected estimates for models with a heterogeneous trend in age or $\sqrt{age}$.
All models include an intercept, age and year dummies in the wage equation, and in addititon the selection models have the same variables in the participation equation. These estimates are not reported in the table.
The selection models allow for state dependence and have two additional equations to model the initial conditions. The parameter estimates for these equations are also not reported.
Standard errors are given in parentheses.

2.F Figures for heterogeneous trend models

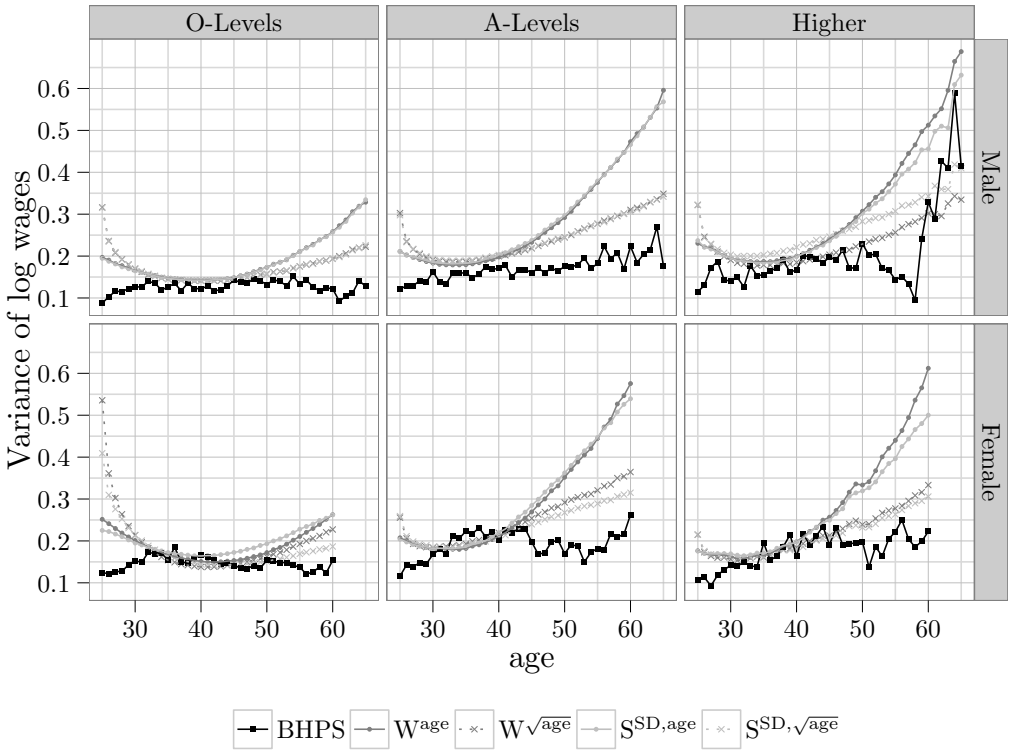

Figure 2.33 – Observed and simulated variance of log-wages by education group and gender

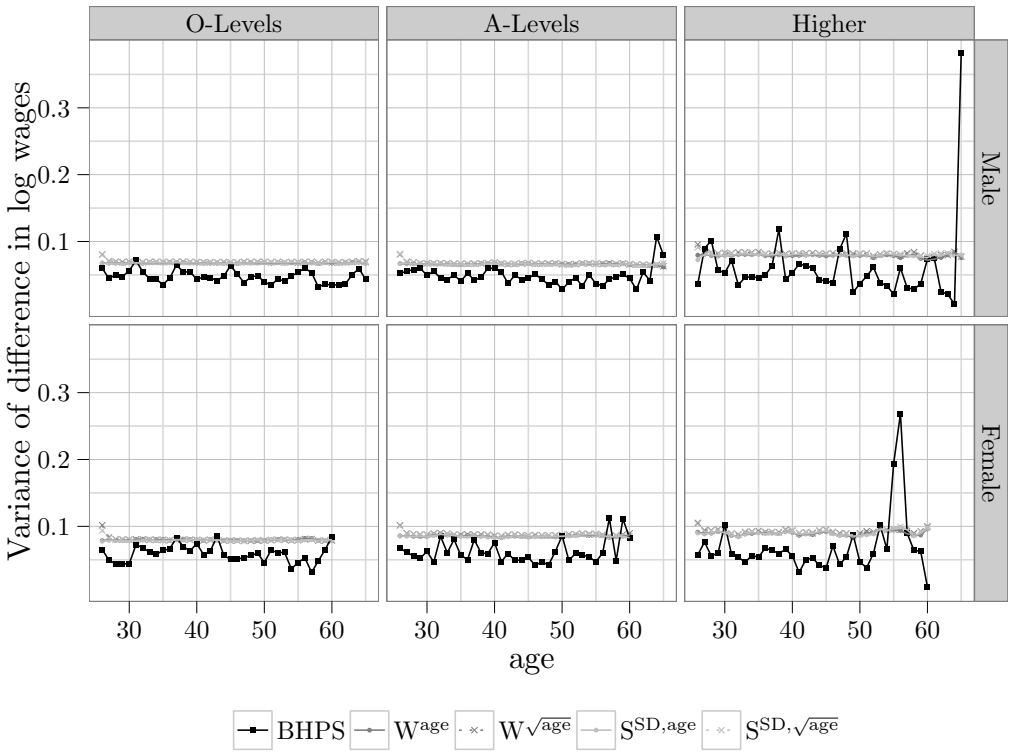

Figure 2.34 – Observed and simulated variance of difference in log-wages by education group and gender

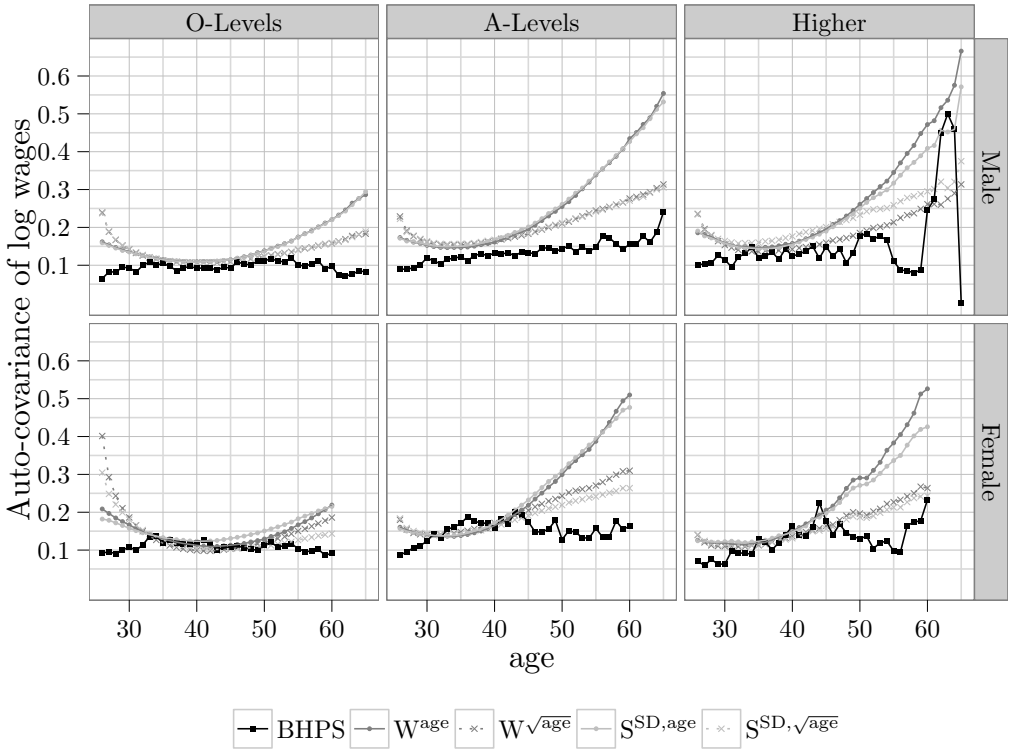

Figure 2.35 – Observed and simulated auto-covariance of log-wages by education group and gender

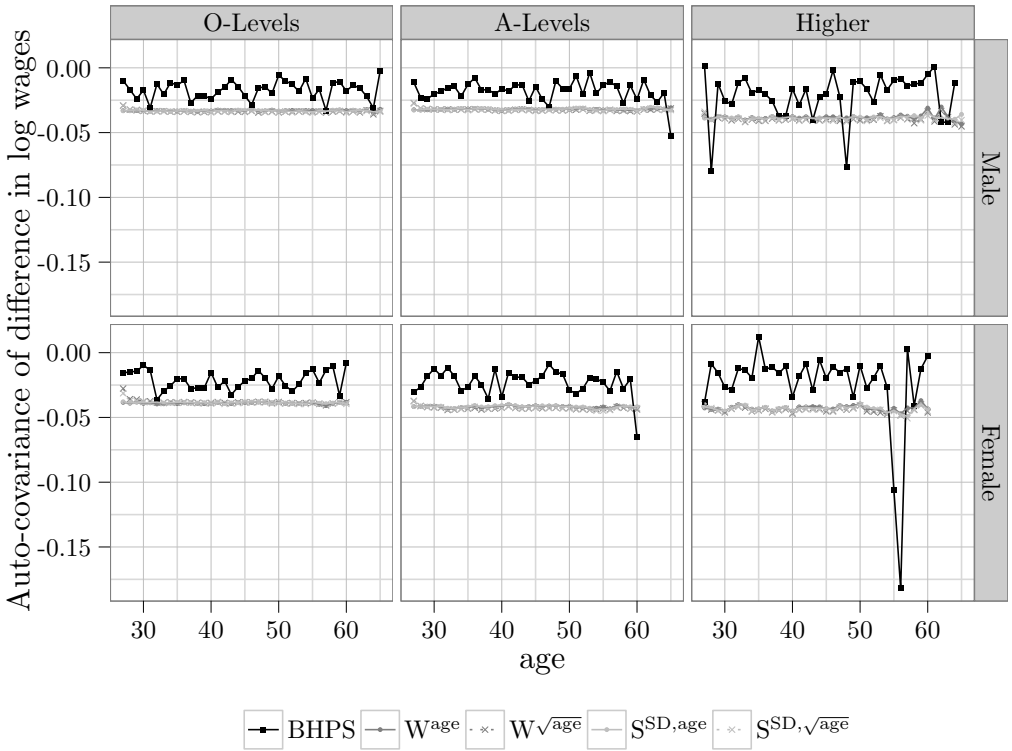

Figure 2.36 – Observed and simulated auto-covariance of difference in log-wages by education group and gender

3

Integration methods for dynamic selection models

3.1 Introduction

In this paper I focus on models with a combination of discrete and continuous outcomes. Following Heckman (1979) these models have been used in labour economics to estimate the joint process describing continuous log-wages and a discrete participation decision. In most cases a single continuous and a single discrete outcome variable is considered. These outcomes are then related through the bivariate normal distribution of two underlying latent variables. This structure is simple to estimate, and procedures to estimate this model are present in all standard statistical software packages. Allowing for multiple continuous and discrete outcomes results in a multi-dimensional integral that has to be approximated, which complicates the estimation procedure. In this paper I explain how these multi-dimensional integrals arise, and show Monte Carlo simulations comparing

different methods to approximate these integrals.

To approximate multi-dimensional integrals, I compare sparse grid integration (Heiss, 2010) to the use of pseudo Monte Carlo numbers and Halton sequences. The experiments in this paper show that sparse grid integration and Halton sequences provide the highest accuracy, when directly comparing the approximated value to the 'true' value. However, when the approximations are used inside a log-likelihood maximization procedure, the differences in the parameter values that maximize the log-likelihood are very small on average between the different methods.

In one example, where we can approximate the same integral by a one-dimensional integration in addition to the multi-dimensional integral approximation, we find that multi-dimensional integration is preferred when a small number of integration nodes is used. In that case, the one-dimensional approximation leads to biased estimates. However, the accuracy of the one-dimensional approximation increases more rapidly when the number of integration nodes is increased.

The same multi-dimensional integral as above is present in models that contain only discrete outcomes. An example where we have multiple discrete outcomes, is when we have panel data on employment status. For the same individual, employment status in different periods will be related. A model for discrete employment status is usually formulated in a latent variable framework, where the latent variables in different periods follow a joint distribution. Since we do not observe the latent variable related to employment status, we have to integrate over the underlying joint distribution of these latent variables to formulate a log-likelihood or moment conditions in terms of the observed discrete employment status.

In general, analytic expressions to evaluate these multi-dimensional integrals are not available, and numerical methods are used to obtain an approximation. Butler and Moffitt (1982) use a random effects specification to describe the covariance structure of the latent variables. In this way, they reduce the problem of approximating a multi-dimensional integral, to the problem of approximating a one-dimensional integral. By assuming that the random effect follows a normal distribution, Gauss-Hermite quadrature can be used to accurately approximate this integral.

There are cases where the error structure can not simply be reduced to a random effects specification. For instance, in addition to a random effect, Hyslop (1999) allows for an auto-regressive error component in the latent variables underlying employment status. In his case, the multi-dimensional integral can not be reduced to a one-dimensional integral. Similarly, when the discrete outcomes are not related to the same decision over time, but to purchases of different goods at the same time, a random effects specification for the unobservables may not be a plausible simplification.

Hyslop (1999) uses data from seven years of the Panel Study of Income Dynamics (PSID), which means that he has to approximate a seven-dimensional integral. Since the underlying random variables are assumed to follow a multivariate normal distribution, he uses the smooth recursive conditioning (CRS) simulator, also known as the GHK simulator to approximate this integral.

In a recent working paper, Altonji et al. (2009) use indirect inference to estimate a model of earnings dynamics that has both continuous and discrete outcomes. Indirect inference is usually implemented using simulations, making it a time-consuming procedure. Also, the simulation of discrete outcomes as used in that setup, leads to a non-smooth objective function with potential local extrema. Since they assume that the random variables defining the process are all normally distributed, their paper fits the framework presented here, and the parameters could be estimated using maximum likelihood. The dimension of integration will not be reduced, but since maximum likelihood estimation in this case usually results in a well-behaved maximization problem, I believe that the solution path will be quick. More CPU time can then be devoted to increase the accuracy of approximating the integral, by using more integration nodes (they use 20 integration nodes). Instead of maximum likelihood, a method of (simulated) moments approach can also be used, which would make the criterion function closer to indirect inference in their case.

In the next section of the paper, I write down a general structure for models with a combination of continuous and discrete outcomes. I also show what the likelihood looks like, if the underlying latent variables follow a normal distribution. Because the likelihood consists of high-dimensional integrals, I summarise some methods to approximate these

integrals in section 3.3. The next section shows for two specific examples how to construct the variance-covariance matrix for the latent variables in the general model; one example has a random effects error structure, and the other has an ARMA error structure. Section 3.5 compares the performance of the different approximation methods on simulated data that was generated using the two example models. The final section concludes.

3.2 Model

In this section I describe the general structure of the models that I consider. For ease of presentation I consider that the data has a panel data structure, where i refers to an individual, and t refers to time. In the examples below, the panel structure simplifies the dependence structure between the error terms. However, t does not have to be time, it can also correspond to the values of different variables in the same period; e.g. expenditure on different food categories. I start with a model with two outcomes for individual i in period t

$$\begin{aligned} Y_{it}^* &= x_{1,it}'\beta_1 + U_{1,it} \\ D_{it}^* &= x_{2,it}'\beta_2 + U_{2,it}, \end{aligned} \tag{3.2.1}$$

where Y_{it}^* is a latent random variable corresponding to a continuous outcome, and D_{it}^* is a latent variable corresponding to a discrete outome. An example in labour economics would be the case where Y_{it}^* corresponds to latent log-wages, and D_{it}^* corresponds to latent labour participation. $U_{1,it}$ and $U_{2,it}$ follow a joint distribution, which we assume to be normal here. The normality assumption is needed to be able to use the approximation methods described below. The same methods can be used with mixtures of normal distributions as well, in case a more flexible distribution is needed. The specification with respect to the covariates does not nescessary have to be linear, but the unobservables have to enter the model additively. Observations corresponding to different individuals are assumed to be independent.

Because of the joint normality in $U_{1,it}$ and $U_{2,it}$, we can write the joint distribution for

Y_{it}^* and D_{it}^* conditional on the observed covariates for T observations for individual i as

$$\begin{pmatrix} Y_i^* \\ D_i^* \end{pmatrix} = \begin{pmatrix} Y_{i1}^* \\ \vdots \\ Y_{iT}^* \\ D_{i1}^* \\ \vdots \\ D_{iT}^* \end{pmatrix} \sim N \left[\underbrace{\begin{pmatrix} M_Y \\ M_D \end{pmatrix}}_{M_i}, \underbrace{\begin{pmatrix} \Sigma_{YY} & \Sigma_{YD} \\ \Sigma_{DY} & \Sigma_{DD} \end{pmatrix}}_{\Sigma_i} \right]. \tag{3.2.2}$$

Y_i^* is a vector with the continuous latent outcomes for individual i in different time periods stacked on top of each other. D_i^* is analogously defined for the latent outcome corresponding to the discrete variables. In general a variable without the t-subscript denotes a vector or matrix with this variable for all time periods combined. M_i and Σ_i are a vector and matrix that depend on parameters and possibly covariates. For instance, with the linear in covariates specification in (3.2.1), we have

$$M_i = \big(\underbrace{x_{1,i1}'\beta_1, \ldots, x_{1,iT}'\beta_1}_{M_Y}, \underbrace{x_{2,i1}'\beta_2, \ldots, x_{2,iT}'\beta_2}_{M_D} \big),$$

where the i subscript is not repeated on M_Y, and M_D, for ease of notation, and similarly for $\Sigma_{YY}, \Sigma_{YD}, \Sigma_{DY}$, and Σ_{DD}. The structure of Σ_i depends on the structure of the dependence between the unobservables $U_{1,i}$ and $U_{2,i}$. We are silent about the exact dependence structure of the unobservables here, but examples of specific cases are provided in section 3.4 below.

The latent variables, Y_i^* and D_i^*, are not observed, but instead we assume that observed values are generated from the latent variables according to a Heckman selection model, referred to as type 2 Tobit by Amemiya (1984). The procedure can be adapted slightly to work with other limited dependent variable models. Instead of the continuous D_{it}^*, we observe a binary variable D_{it}

$$D_{it} = \begin{cases} 1 & \text{if } D_{it}^* > 0, \\ 0 & \text{otherwise.} \end{cases}$$

When $D_{it} = 1$, we observe the value of the latent Y_{it}^*. Otherwise we do not obtain any information about Y_{it}^* and normalize the observed value to 0. The observed Y_{it} is then defined using the following expression

$$Y_{it} = D_{it} Y_{it}^*.$$

In terms of the wage model above; we observe whether someone has a job ($D_{it} = 1$) or is unemployed ($D_{it} = 0$). Only if an individual has a job, we observe their log-wage (Y_{it}^*).

To obtain estimates for the parameters, β_1, β_2, and the set of parameters defining the elements of Σ_i, I use the joint distribution of the latent variables, to write down a likelihood. The probability density function of the latent variables is

$$\begin{aligned} & f_{Y_{i1}^*,\ldots,Y_{iT}^*,D_{i1}^*,\ldots,D_{iT}^*}(Y_{i1}^*,\ldots,Y_{iT}^*,D_{i1}^*,\ldots,D_{iT}^* \mid M_i,\Sigma_i) = \\ &= \frac{1}{(2\pi)^{k/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}\left(\begin{pmatrix} Y_i^* \\ D_i^* \end{pmatrix} - M_i\right)' \Sigma_i^{-1} \left(\begin{pmatrix} Y_i^* \\ D_i^* \end{pmatrix} - M_i\right)\right) \end{aligned}$$

Because the latent variables are not directly observed, we need to define the likelihood function in terms of realizations of the observable random variables. Realizations of random variables are denoted in lower case.

Similar to Yen (2005) it is instructive to consider three separate cases. In the first case none of the continuous outcome values are observed for individual i. In other words, all the realizations for d_{it} are equal to 0. There is no information about any of the y_{it}^* in this case, and the likelihood, $\mathcal{L}(y_{i1},\ldots,y_{it},d_{i1},\ldots,d_{it} \mid M_i,\Sigma_i)$, only depends on the marginal distribution of D_i^*

$$\begin{aligned} & \mathcal{L}(y_{i1}=0,\ldots,y_{iT}=0,d_{i1}=0,\ldots,d_{iT}=0 \mid M_i,\Sigma_i) = \\ &= \int_{-\infty}^{0}\cdots\int_{-\infty}^{0} f_{D_{i1}^*,\ldots,D_{iT}^*}(d_{i1}^*,\ldots,d_{iT}^*)dd_{i1}^*\cdots dd_{iT}^*, \end{aligned}$$

where $f_{D_{i1}^*,\ldots,D_{iT}^*}(d_{i1}^*,\ldots,d_{iT}^*)$ denotes the marginal distribution of the latent variable. This integral calculates exactly the probability that $d_{it} = 0$ for all t, or equivalently, the

probability that all the underlying latent variables are smaller than 0

$$P(d_{i1}^* \leq 0 \cap \cdots \cap d_{it}^* \leq 0).$$

Since no assumptions have been made about the dependence structure of $U_{2,i}$, the integral above is of dimension T. In section 3.3, I describe how to calculate an approximation to this integral.

The marginal distribution, $f_{D_{i1}^*,\ldots,D_{iT}^*}(d_{i1}^*,\ldots,d_{iT}^*)$, is easily derived because of the normality of the underlying variables. By selecting the elements in M_i that are related to D_i^*, i.e. M_D, and selecting the elements in the rows and columns of Σ_i that are related to D_i^*, i.e. Σ_{DD}, we obtain the marginal distribution of D_i^*, which is again normal, $D_i^* \sim N(M_D, \Sigma_{DD})$.

In the second case, $d_{it} = 1$ for all t, which means that we observe $y_{it} = y_{it}^*$ in every period. The likelihood is

$$\begin{aligned}
&\mathcal{L}\left(y_{i1} = y_{i1}^*, \ldots, y_{iT} = y_{iT}^*, d_{i1} = 1, \ldots, d_{iT} = 1 \mid M_i, \Sigma_i\right) = \\
&= \int_0^\infty \cdots \int_0^\infty f_{Y_i^*, D_i^*}(y_{i1}^*, \ldots, y_{iT}^*, d_{i1}^*, \ldots, d_{iT}^*) dd_{i1}^* \cdots dd_{iT}^* \\
&= f_{Y_i^*}(y_{i1}^*, \ldots, y_{iT}^*) \cdot \int_0^\infty \cdots \int_0^\infty f_{D_i^*|Y_i^*=y_i^*}(d_{i1}^*, \ldots, d_{iT}^*) dd_{i1}^* \cdots dd_{iT}^*,
\end{aligned}$$

where Bayes' rule is used to go from the second to the third equation, replacing the joint distribution by a marginal and a conditional distribution. Values for y_{it}^* are observed, which means that we do not have to integrate over all possible realizations of y_{it}^*. Analogously to the first case, we only know that all d_{it}^* are larger than 0, resulting in a T-dimensional integral over the possible values of d_{it}^*, where the domain of integration now runs from 0 to infinity.

The marginal distribution of Y_i^* can be obtained similarly as described above for the marginal distribution of D_i^*. The conditional distribution of $D_i^*|Y_i^* = y_i$ is a multivariate normal distribution with mean

$$M_{D|Y} = M_D + \Sigma_{DY}\Sigma_{YY}^{-1}(y_i - M_Y), \tag{3.2.3}$$

and variance-covariance matrix

$$\Sigma_{D|Y} = \Sigma_{DD} - \Sigma_{DY}\Sigma_{YY}^{-1}\Sigma_{YD}. \tag{3.2.4}$$

In the third and final case, some of the values for y_{it}^* are observed, but not all of them. Define $\tilde{y}_i^*$ as the subvector of y_i^*, where $d_i = 1$. The likelihood is simlar to the second case,

$$\begin{aligned}
&\mathcal{L}\left(y_{i1}, \ldots, y_{iT}, d_{i1}, \ldots, d_{iT} \mid M_i, \Sigma_i\right) = \\
&\quad = \quad f_{\tilde{Y}_i^*}(\tilde{y}_i^*) \cdot \int_{\mathcal{D}_T} \cdots \int_{\mathcal{D}_1} f_{D_i^*|\tilde{Y}_i^*=\tilde{y}_i^*}(d_{i1}^*, \ldots, d_{iT}^*) dd_{i1}^* \cdots dd_{iT}^*.
\end{aligned}$$

In this case we condition on $\tilde{Y}_i^* = \tilde{y}_i^*$, because only this subset of observations contains information about the value of y_i^*. The mean for the distribution of $D_i^*|\tilde{Y}_i^* = \tilde{y}_i^*$ is equivalent to (3.2.3), where y_i should be replaced by $\tilde{y}_i$, and M_Y should be replaced by $M_{\tilde{Y}}$. A similar replacement is needed in (3.2.4).

As a second point, $d_{it} = 0$ for some t, and $d_{it} = 1$ in other time periods. This results in different domains of integration. The domain of integration is given by $\mathcal{D}_t$, which is $(-\infty, 0)$ if $d_{it} = 0$, and $(0, \infty)$ otherwise.

For every individual, the observed outcomes can be combined with one of the three likelihood functions above, to get the likelihood contribution for this individual. By taking the logarithm and summing over all individuals, the total log-likelihood is obtained

$$\log \mathcal{L}\left(y, d \mid X, \theta\right) = \sum_{i=1}^{N} \log \mathcal{L}\left(y_i, d_i \mid M_i, \Sigma_i\right),$$

where y, d, and X are vectors and matrices with the observed outcome variables and covariates for all individuals, and θ is a vector of parameters, including β_1, β_2, and the set of parameters that determine Σ_i. Maximizing this log-likelihood with respect to the parameters θ, results in an estimate $\hat{\theta}$. In order to calculate the log-likelihood, we need a method to approximate the T-dimensional integrals that arise in the individual log-likelihood contributions. A description of such a method is given in the next section.

Finally, in some cases we may have continuous variables that are observed for all individuals, or we may have discrete variables that do not correspond to a continuous

outcome. The likelihood functions above can be easily adjusted to incorporate those cases. For instance, if some of the variables are not subject to selection, because a value for these variables is observed for all individuals, we do not have to integrate out the probability that this value is equal to 0, because it is 1 for everyone. The corresponding D_{it}^* can be removed from the analysis, reducing the dimension of integration by one.

3.3 Multivariate normal integration

To calculate the likelihood for the model in the previous section, we need to approximate a T-dimensional integral of the form

$$P(a_1 < X_1 < b_1, \ldots, a_T < X_T < b_T) = \int_{a_T}^{b_T} \cdots \int_{a_1}^{b_1} f_{X_1,\ldots,X_T}(x_1, \ldots, x_T) dx_1 \cdots dx_T,$$

where $f_{X_1,\ldots,X_T}(x_1, \ldots, x_T)$ is the joint probability density function for the random variables $X_1, \ldots, X_T$. These random variables are assumed to follow a joint normal distribution with vector of means M and variance-covariance matrix Σ.

Perhaps the most well-known procedure to approximate this integral is the Geweke-Hajivassiliou-Keane or GHK simulator. The main papers developing the simulator and testing its performance are Hajivassiliou et al. (1996), Börsch-Supan and Hajivassiliou (1993), Geweke (1996) and Keane (1994). Around the same time, Genz (1992) developed the method independently.

I will present a short explanation of the algorithm here, where I restrict myself to approximating a 2-dimensional integral. The approach can be generalized to more dimensions (see for instance Train, 2003, for an explanation). The GHK simulator starts from the fact random draws from a joint normal distribution can be written as a linear combination of random draws from independent standard normal distributions with a triangular structure

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} M_1 \\ M_2 \end{pmatrix} + \begin{pmatrix} \omega_{11} & 0 \\ \omega_{21} & \omega_{22} \end{pmatrix} \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix}, \tag{3.3.1}$$

where Z_1 and $Z_2 \sim$ i.i.d. $N(0,1)$, and $(X_1, X_2) \sim N(M, \Sigma)$. The matrix containing the ω elements is the lower Cholesky decomposition of Σ, i.e. $\Sigma = \Omega\Omega^T$. To simplify the

notation, I change the bounds on the domain of integration, where I choose $a_1 = a_2 = -\infty$, and $b_1 = b_2 = 0$. Similar to Train (2003), we can then rewrite the probability above

$$\begin{aligned}
&\int_{-\infty}^{0}\int_{-\infty}^{0} f_{X_1,X_2}(x_1,x_2)dx_1dx_2 = \\
&= P[X_1 < 0 \cap X_2 < 0] \\
&= P[X_1 < 0]\, P[X_2 < 0 \mid X_1 < 0] \\
&= P[M_1 + \omega_{11}Z_1 < 0]\, P[M_2 + \omega_{21}Z_1 + \omega_{22}Z_2 < 0 \mid M_1 + \omega_{11}Z_1 < 0] \\
&= P\left[Z_1 < -\frac{M_1}{\omega_{11}}\right] P\left[Z_2 < -\frac{M_2 + \omega_{21}Z_1}{\omega_{22}} \;\middle|\; Z_1 < -\frac{M_1}{\omega_{11}}\right] \\
&= \Phi\left(-\frac{M_1}{\omega_{11}}\right)\int_{-\infty}^{-\frac{M_1}{\omega_{11}}} \Phi\left(-\frac{M_2 + \omega_{21}z_1}{\omega_{22}}\right)\phi(z_1)dz_1, \qquad (3.3.2)
\end{aligned}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the probability density function and the cumulative distribution function for the standard normal distribution. This reduces the 2-dimensional integral to a 1-dimensional integral. The integral is then approximated using simulations

$$\int_{-\infty}^{-\frac{M_1}{\omega_{11}}} \Phi\left(-\frac{M_2 + \omega_{21}z_1}{\omega_{22}}\right)\phi(z_1)dz_1 \approx \frac{1}{R}\sum_{r=1}^{R} \Phi\left(-\frac{M_2 + \omega_{21}z_1^r}{\omega_{22}}\right),$$

where z_1^r is a draw from the truncated normal distribution with upper bound $-\frac{M_1}{\omega_{11}}$. A draw from the truncated normal distribution with lower bound a_1 and upper bound b_1, can be obtained from a uniform draw, $u_1^r \sim U(0,1)$, by inverting a transformation using the quantile function of the normal distribution

$$z_1^r = \Phi^{-1}\left[\Phi(a) + u_1^r\left(\Phi(b) - \Phi(a)\right)\right].$$

In the more general case, the T-dimensional integral will be replaced by a $T-1$-dimensional integral, with the benefit that the $z_1, z_2, \cdots, z_{T-1}$ are independent. The integral can still be seen as an integration over independent truncated normal variables, and the multi-dimensional integral can be approximated using random uniform draws. However, since we have more dimensions, more random draws need to be used to sample the domain of integration in order to get an accurate approximation.

The derivation in (3.3.2) is not unique. Instead of decomposing the elements using the

Cholesky decomposition in (3.3.1), we can swap the elements X_1 and X_2 and decompose the variables as follows

$$\begin{pmatrix} X_2 \\ X_1 \end{pmatrix} = \begin{pmatrix} M_2 \\ M_1 \end{pmatrix} + \begin{pmatrix} \tilde{\omega}_{22} & 0 \\ \tilde{\omega}_{12} & \tilde{\omega}_{11} \end{pmatrix} \begin{pmatrix} Z_2 \\ Z_1 \end{pmatrix}.$$

If we then condition on X_2 in the second step of the derivation for (3.3.2) instead of conditioning on X_1 we get

$$\Phi\left(-\frac{M_2}{\tilde{\omega}_{22}}\right) \int_{-\infty}^{-\frac{M_2}{\tilde{\omega}_{22}}} \Phi\left(-\frac{M_1 + \tilde{\omega}_{12} z_2}{\tilde{\omega}_{11}}\right) \phi(z_2) dz_2.$$

This results in a different value for the approximation. For example, let $M = (-2, 1)$ and Σ a matrix with ones on the diagonal and 0.7 as the off-diagonal element. Approximating the integral using seven Halton draws (see below), we get 0.1309 if we use the first approximation, and 0.1587 if we use the second.

Genz (1992) notes that in his experiments a more accurate approximation can usually be obtained with fewer nodes by reordering the variables such that the variables with largest domain of integration are the innermost variables in the integral. The mean of the variables is subtracted from the bounds of integration, which implies that the mean has an effect on the size of the domain of integration, if one of the bounds is infinite. This is the case in our setup.

As we saw in the example above, the difference in approximation can be substantial. However, I do not implement this re-ordering in this paper for the following reason. The probabilities approximated here, are used to approximate a log-likelihood function. This log-likelihood function depends on parameters, and we want to find the set of parameter values that maximize the likelihood. Finding this optimum is achieved using an iterative procedure, where we start with a guess for the parameters, calculate the log-likelihood at this point, and then update the parameters. This is repeated until we have reached the optimum. Since the mean, and therefore the bounds of integration usually depend on the parameters that we are estimating, a change in the parameters could result in a re-ordering of the integrals. The re-ordering could in that way cause a discontinuity in

the likelihood function that we are optimizing, thus causing convergence problems. The ordering of the integrals should therefore be fixed in advance.

Note that if one wanted to use the optimal ordering, this could be achieved in steps. The first step is to find the parameters maximizing the log-likelihood for a given ordering. After the procedure has converged, an optimal ordering can be defined for those parameters. Then, a new optimization procedure can then be started using the new ordering. One could repeat this procedure if needed. In this paper I do not use this method. Whether the improvement in accuracy you obtain is valuable in practice remains an open question.

Instead of using uniform random draws to sample from the domain of integration and approximate the multi-dimensional integral by simulation, other methods can be used. Heiss (2010) proposes to use sparse grids and compares the performance with Monte Carlo approximation, i.e. using uniform random draws, and an approximation obtained using Halton sequences. Both these methods have the same objective as Monte Carlo integration, sampling the domain of integration, but achieve this in different ways. I will first explain sparse grids, followed by Halton sequences.

A common way to approximate a one-dimensional integral is by Gaussian quadrature (Judd, 1998). When the integral consists of a combination of a function of interest $g(\cdot)$ and some nonnegative weighting function $f(\cdot)$, this integral can be approximated using

$$\int_a^b g(x)f(x)dx \approx \sum_{r=1}^{R} w^r g(x^r),$$

where x^r is an integration node and w^r is a weight. Gaussian quadrature refers to the way of choosing the integration nodes and weights in such a way that with R nodes, the polynomial can be approximated without error if $g(\cdot)$ is a polynomial of degree $2R-1$. There are different types of Gaussian quadrature that can be used depending on the domain of integration and the weighting function $f(\cdot)$. For instance, Gauss-Legendre quadrature can be used to integrate a function over a closed domain, $[a, b]$, when the weighting function is $f(x) = 1$. Gauss-Hermite quadrature is used to evaluate integrals where the domain runs from minus infinity to infinity, and the weighting function is a Gaussian probability density function.

In practice we also want to evaluate integrals where the function $g(\cdot)$ is not a polynomial. The same approach can be used, with the difference that the evaluation is no longer exact, i.e. some approximation error is introduced. The size of the approximation error depends on how well the function $g(\cdot)$ can be approximated by a polynomial of degree $2R-1$. If the function is close to being polynomial, the approximation error is small. When the function cannot be accurately approximated by a polynomial, for instance because the function is not continuous, the approximation error is larger.

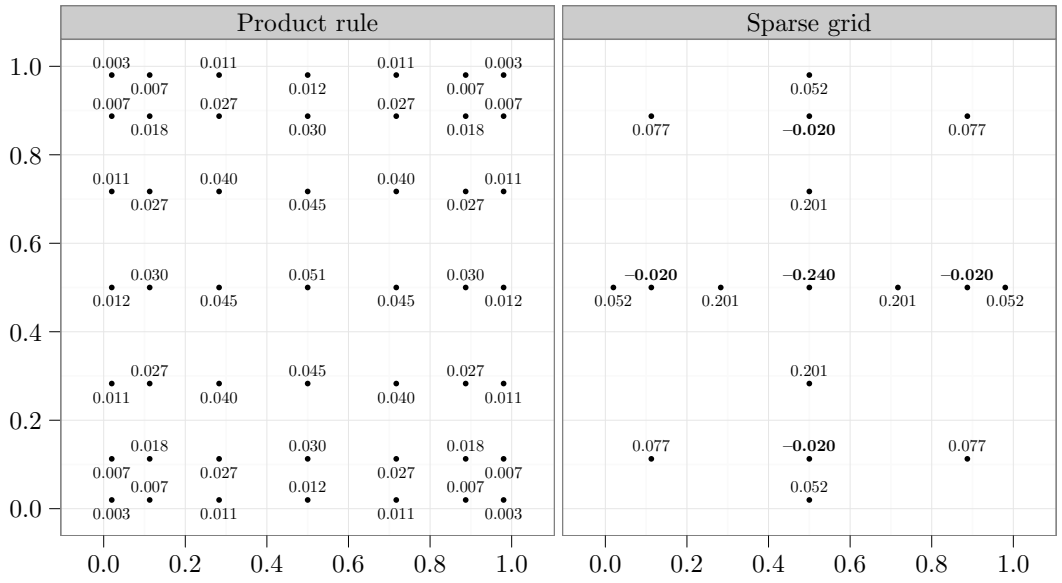

Figure 3.1 – Nodes of 2D integration grid with corresponding weights

One way to extend these methods for one-dimensional integrals to T-dimensional integrals, is by combining the nodes and weights for one dimension using a product rule. The resulting nodes and weights for 2-dimensional integral are shown in figure 3.1 on the left, where the numbers in small print correspond to the weight of a particular node. The problem with this approach is that the number of nodes increases exponentially in the dimension of the integral, leading to a grid with R^T nodes and weights to evaluate a T-dimensional integral.

Sparse grid integration is a different way of combining one-dimensional grids, resulting in a polynomial increase in the number of integration nodes instead of the exponential increase displayed by the product rule (Heiss & Winschel, 2008; Heiss, 2010). This comes at a cost. For instance, if we start with a one-dimensional grid that can exactly approximate

the integral where $g(\cdot)$ is a polynomial of order 2,

$$a_0 + a_1 x_1 + a_2 x_1^2,$$

then the integration grid that is constructed by combining two of these one-dimensional integration nodes using the product rule can exactly approximate an integral where the polynomial $g(\cdot)$ looks like

$$a_{00} + a_{10} x_1 + a_{01} x_2 + a_{20} x_1^2 + a_{02} x_2^2 + a_{11} x_1 x_2 + a_{21} x_1^2 x_2 + a_{12} x_1 x_2^2 + a_{22} x_1^2 x_2^2.$$

The maximal exponent that can enter the polynomial is 2. This bound follows directly from the order of the polynomial that the one-dimensional grid can approximate exactly.

The sparse grid that is created by combining the same one-dimensional integration nodes, results in an exact approximation of an integral where $g(\cdot)$ is

$$a_{00} + a_{10} x_1 + a_{01} x_2 + a_{20} x_1^2 + a_{02} x_2^2 + a_{11} x_1 x_2.$$

In this case the sum of the exponents of the separate terms in the polynomial is bounded by 2, instead of the maximum of the exponents. The smaller class of polynomials for which evaluation of the integral is exact, results in fewer integration nodes required by sparse grids. If the higher-order interaction terms are important elements to approximate the function $g(\cdot)$, then the approximation with sparse grids will suffer.

Heiss and Winschel (2008) assess the performance of sparse grid integration to approximate the probabilities in a mixed logit model. To evaluate the probabilities, a multi-dimensional integral has to be approximated. In experiments with simulated data they compare the accuracy of the approximation using sparse grids with approximations using Monte Carlo integration for integrals up to dimension 20. Similarly, Heiss (2010) shows the results of simulation experiments evaluating the probabilities in a dynamic probit model using sparse grids. They conclude that sparse grid integration gives more accurate approximations for the probabilities than pseudo-random or quasi-random Monte Carlo integration.

There are downsides to using sparse grid integration. First of all, the weights that are used in sparse grid integration can be negative. Figure 3.1 on the right shows the integration nodes that we obtain from combining the same one-dimensional grid that was used to create the product rule grid on the left. Instead of the $7 \times 7 = 49$ nodes that the product rule grid contains, the sparse grid has only 17 nodes. However, as can be seen from the figure, some of the weights, in bold, are negative; e.g. -0.240 for the node in the center. In practice this could lead to an approximated probability lower than 0, leading to problems when taking the log of this probability inside a log-likelihood estimation.

The second downside is that we can not choose an arbitrary number of integration nodes. To use sparse grid integration one specifies the accuracy of the one-dimensional polynomial that one wants to attain. The integration grid and the number of nodes that should be used follows from this accuracy. This means that the number of integration nodes can only be increased in pre-specified steps, e.g. from 21, to 201, to 1201, to 5281 nodes for integration in 10 dimensions. This can be problematic if a higher degree of accuracy is required, but there are not enough computing resources available to increase the number of nodes from one level of accuracy to the next. A potential solution for this problem, that to the best of my knowledge has not been tested in practice, is to augment the nodes from a sparse grid by a set of nodes obtained by some other method, e.g. a random grid of nodes, to improve the accuracy of the final approximation.

A different way to create a set of nodes to use in the approximation to the integral, are so-called quasi-random numbers. Since computers can not generate truly random numbers, the random numbers that are generated, such as the random draws from the uniform distribution above, are usually referred to as pseudo-random numbers. The approximation of an integral using pseudo-random numbers is called pseudo Monte Carlo integration, or simply Monte Carlo integration.

Quasi-random numbers or low discrepancy sequences differ from pseudo-random numbers in two respects. They cover the area that they are drawn from more evenly, and the different draws are correlated with each other (e.g. Train, 2003). An example of the more even coverage can be seen in figure 3.2, where 200 draws from a popular type of quasi-random numbers, Halton sequences, are shown on the left, and pseudo-random numbers

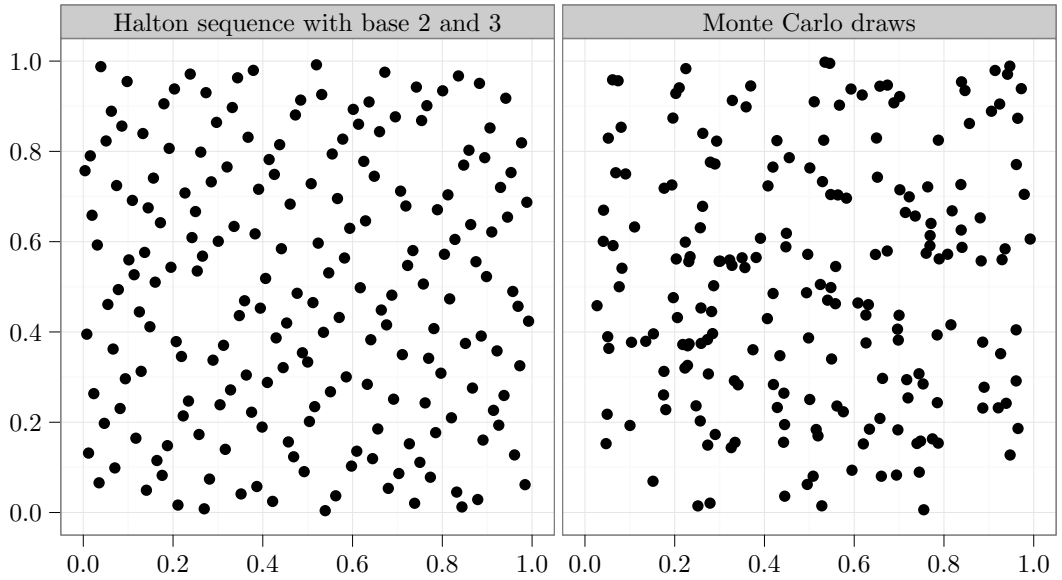

Figure 3.2 – Halton sequences versus pseudo-random numbers

are shown on the right. The Halton sequences seem to be more evenly distributed over the two-dimensional area.

A sequence of Halton draws that cover the one-dimensional grid between 0 and 1 uniformly, are created from a base, k. If required, these draws can be transformed to quasi-random numbers from a different distribution, e.g. the normal distribution, by using the quantile function for this distribution. Halton sequences are constructed by splitting the interval 0–1 into k equal parts. The values where the break between the parts occur, are the first numbers for the sequence, i.e., $\frac{1}{k}, \frac{2}{k}, \ldots, \frac{k-1}{k}$. For instance, if $k = 2$, this results in one number, $\frac{1}{2}$, if $k = 3$, we have $\frac{1}{3}, \frac{2}{3}$.

The next step is to divide each of the parts in k equal parts and add the newly obtained breakpoints to sequence we already have in a special way. With $k = 2$, we add $\frac{1}{4}$ and $\frac{3}{4}$. For $k = 3$, we first add $\frac{1}{9}$, $\frac{4}{9}$, and $\frac{7}{9}$, and then add $\frac{2}{9}$, $\frac{5}{9}$, and $\frac{8}{9}$. The process continuous by breaking these intervals into smaller parts in every step.

For every base k, we can construct a Halton sequence and sequences with different base numbers can be combined to form a multi-dimensional grid of nodes. The two-dimensional grid in figure 3.2 on the left was created by combining a Halton sequence with base 2 and a Halton sequence with base 3. Figure 3.3 shows that not all combinations of base numbers work well. Two sequences of length 1000 are shown. On the left, base 2 and base 3 are

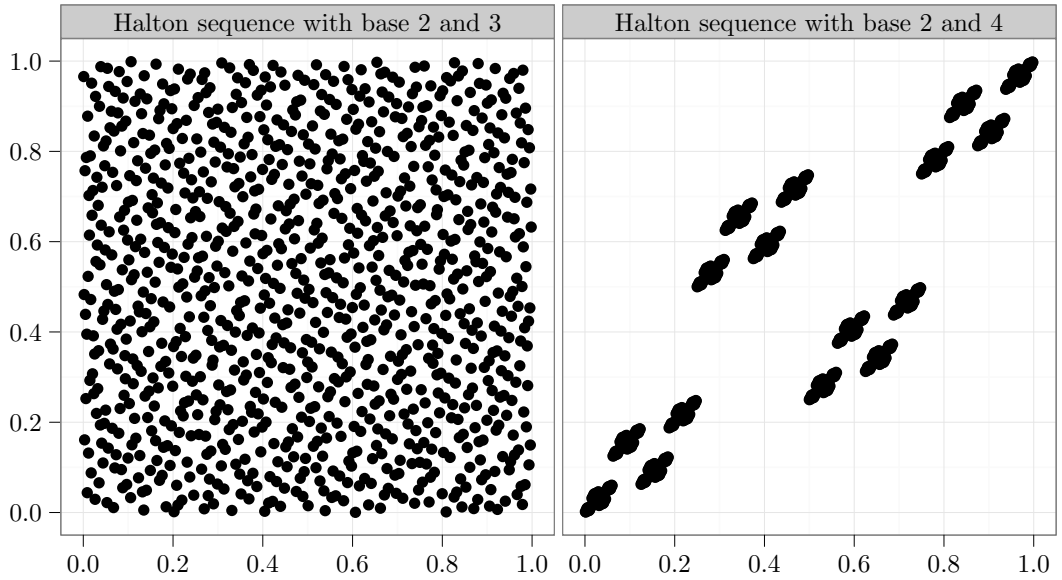

Figure 3.3 – Halton sequences with base 2 and 3, and with base 2 and 4

combined to get a two-dimensional grid. On the right, I combined base 2 and base 4. The two-dimensional grid constructed from base 2 and 4 covers only a small fraction of the area. Because 4 can be divided by 2, this means that the construction of the Halton sequences are very similar, resulting in a two-dimensional grid with highly correlated draws. To avoid this cycling, only prime numbers are used as a base to construct Halton sequences in practice.

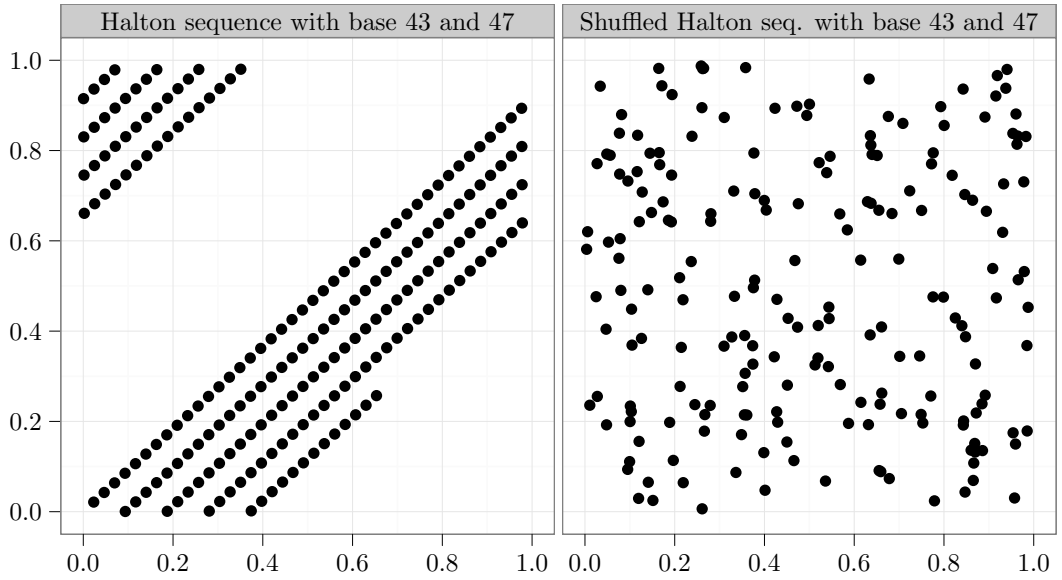

Figure 3.4 – Halton sequences with base 43 and 47 and its shuffled version

Bhat (2003) notes that the practice of using prime numbers can still result in problems if we construct sequences for higher dimensions. He gives an example when there are 15 dimensions. The 14^{th} prime number is 43, and the 15^{th} prime number is 47, so these numbers are used as the base for the Halton sequences corresponding to dimension 14 and 15. These sequences are correlated, as can be seen in figure 3.4 on the left, which shows the first 200 draws for Halton sequences with base 43 and base 47. A solution to this problem is to add the Halton numbers, $\frac{1}{k}, \frac{2}{k}, \ldots$, in a different order to the sequence. References to papers describing different methods to construct such permutations are given in Bhat (2003). For example, one of these methods gives $\frac{2}{3}, \frac{1}{3}, \frac{2}{9}, \frac{8}{9}, \frac{5}{9}, \frac{1}{9}, \frac{7}{9}$, and $\frac{4}{9}$ as the first numbers in a scrambled Halton sequence with base 3. Note that the order of the numbers is different than the order we have above.

Good scrambling methods that permute the draws, and thus prevent correlations, and at the same time cover the whole area, are difficult to find for higher dimensions (Hess, Polak, & Daly, 2003). They use a simple method, referred to as shuffling, and show that this works well for sequences of length 100 and 200 in a small simulation experiment. A shuffled Halton sequence can be obtained by starting from a regular Halton sequence. For every dimension the sequence corresponding to this dimension is replaced by a (pseudo-)random permutation of the sequence. Using a different permutation for every dimension breaks the correlation, as can be seen on the right in figure 3.4. These nodes correspond to the same one-dimensional sequences in the figure on the left, but they are combined in a different way to get the shuffled version. This shuffled version shows better coverage for the area, although we do see some of the nodes lumping together.

In practice, another solution that is often used to mitigate the problem of correlations, is to discard, or burn, the first set of Halton draws. This is what I do to construct the Halton sequences in the experiments below. The first 500 numbers are not used. In the experiments, I compare the use of sparse grids, pseudo Monte Carlo nodes, and both shuffled Halton sequences and regular Halton sequences, to approximate integrals of different dimensions.

3.4 Examples

In this section I discuss two example models with discrete and continuous outcomes. By defining the unobservables in the latent outcome process as a linear combination of independent random variables with a normal distribution, we can easily specify models with auto-regressive or moving average processes in the form described above.

It is important to note that although the examples here use observations on one continuous and one discrete outcome in different time periods for the same individual, they can easily be adapted to models where there are observations on multiple continuous and discrete outcomes within the same period. An example of such a model would be the labour participation decision of a couple. In every period there are two discrete outcomes. We see whether the female works, and whether the male works. If the female works, we also observe her continuous wage. We do not observe the wage, if she does not work, and similarly for the male.

The latent work decision of both spouses depend on their individual wages, but potentially on the wage of the spouse as well. In addition, there may be an underlying factor that affects the taste for work of both spouses. Some of these elements might be persistent over time. If the underlying factors are specified in terms of normal distributions, the same methods as described in the examples below can be used to derive the variance-covariance matrix that describes the stochastic structure of the latent variables. A likelihood can be formed involving the continuous and discrete outcomes of both spouses in multiple time periods, which can then be approximated using the methods described above.

3.4.1 Example 1: random effects

This first example is similar to the random effects model presented in chapter 2. The latent outcome variables follow the linear specification in (3.2.1). The unobservables are defined by the following random effect model

$$\begin{aligned} U_{1,it} &= \sigma_\alpha \alpha_i + \sigma_\varepsilon \varepsilon_{it} \\ U_{2,it} &= \frac{\rho}{\sigma_\varepsilon} \sigma_\alpha \alpha_i + \rho \varepsilon_{it} + \sqrt{1-\rho^2}\eta_{it}, \end{aligned} \tag{3.4.1}$$

where $\alpha_i \sim N(0,1)$ is the random effect, $\varepsilon_{it} \sim N(0,1)$ are transitory shocks to the continuous outcome, and $\eta_{it} \sim N(0,1)$ are transitory shocks to the latent variable driving the discrete outcome. $U_{1,it}$ and $U_{2,it}$ are correlated if ρ is not equal to zero. $U_{1,it}$ is correlated over time through α_i, and $U_{2,it}$ is correlated over time through α_i if $\rho \neq 0$.

The same relation can be written in matrix notation

$$\begin{pmatrix} U_{1,i} \\ U_{2,i} \end{pmatrix} = \begin{pmatrix} U_{1,iT} \\ \vdots \\ U_{1,i1} \\ U_{2,iT} \\ \vdots \\ U_{2,i1} \end{pmatrix} = \underbrace{\begin{pmatrix} \sigma_\varepsilon & & 0 & 0 & & 0 & \sigma_\alpha \\ & \ddots & & & \ddots & & \vdots \\ 0 & & \sigma_\varepsilon & 0 & & 0 & \sigma_\alpha \\ \rho & & 0 & \sqrt{1-\rho^2} & & 0 & \frac{\rho}{\sigma_\varepsilon}\sigma_\alpha \\ & \ddots & & & \ddots & & \vdots \\ 0 & & \rho & 0 & & \sqrt{1-\rho^2} & \frac{\rho}{\sigma_\varepsilon}\sigma_\alpha \end{pmatrix}}_{A} \begin{pmatrix} \varepsilon_{iT} \\ \vdots \\ \varepsilon_{i1} \\ \eta_{iT} \\ \vdots \\ \eta_{i1} \\ \alpha_i \end{pmatrix}.$$

Since the errors α_i, ε_{it}, and η_{it} are i.i.d. and follow a standard normal distribution, the vector $(U_{1,i}, U_{2,i})$ is also normally distributed, with mean 0, and variance-covariance matrix AA'. This can be directly related to the joint distribution of Y_i^* and D_i^* in (3.2.2), when we define $\Sigma_i = AA'$, and we can use the method described in that section to find maximum likelihood estimates for the parameters β_1, β_2, σ_ε, σ_α, and ρ.

3.4.2 Example 2: ARMA specification

In this example, the unobservable in the continuous outcome equation follows an ARMA(1,1) process,

$$\begin{aligned} U_{1,it} &= \xi_{it} + \zeta_{it} \\ U_{2,it} &= \frac{\rho}{\sqrt{\theta_1^2 + \theta_2^2}}\xi_{it} + \frac{\rho}{\sqrt{\theta_1^2 + \theta_2^2}}\zeta_{it} + \sqrt{1-\rho^2}\eta_{it}, \end{aligned} \tag{3.4.2}$$

where ξ_{it} follows an AR(1) process and ζ_{it} an MA(1) process as defined below, and $\eta_{it} \sim N(0,1)$. The two unobservables are related through the parameter ρ and the contributions of ξ_{it} and ζ_{it} in the discrete outcome equation are scaled in such a way that the combination of ζ_{it} and η_{it} has unit variance.

We have observations for individual i in periods $t = 1, \ldots, T$. The first period value of ξ_{it} is assumed to follow a normal distribution with standard deviation σ_{ξ_0}, $\xi_{i1} \sim N(0, \sigma_{\xi_0}^2)$. To have all expressions in terms of standard normal random variables, I write for $t = 1$

$$\xi_{i1} = \sigma_{\xi_0} \varepsilon_{i1},$$

where $\varepsilon_{i1} \sim N(0, 1)$. For $t > 1$, the AR(1) process can be written as

$$\begin{aligned} \xi_{it} &= \phi \xi_{it-1} + \sigma_\varepsilon \varepsilon_{it} \\ &= \phi^2 \xi_{it-2} + \phi \sigma_\varepsilon \varepsilon_{it-1} + \sigma_\varepsilon \varepsilon_{it} \\ &\vdots \\ &= \phi^{t-1} \sigma_{\xi_0} \varepsilon_{i1} + \sum_{s=2}^{t} \phi^{t-s} \sigma_\varepsilon \varepsilon_{is}, \end{aligned}$$

where the innovations to the AR(1) process follow a standard normal distribution, $\varepsilon_{it} \sim N(0, 1)$. The same expression for all ξ_{it} combined can also be written in matrix notation

$$\underbrace{\begin{pmatrix} \xi_{iT} \\ \xi_{iT-1} \\ \vdots \\ \xi_{i2} \\ \xi_{i1} \end{pmatrix}}_{\Xi_i} = \underbrace{\begin{pmatrix} 1 & \phi & \phi^2 & \cdots & \phi^{T-1} \\ 0 & 1 & \phi & \cdots & \phi^{T-2} \\ 0 & 0 & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 1 & \phi \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}}_{\Phi} \cdot \underbrace{\begin{pmatrix} \sigma_\varepsilon & 0 & 0 & 0 & 0 \\ 0 & \sigma_\varepsilon & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & \sigma_\varepsilon & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\xi_0} \end{pmatrix}}_{\Sigma_\varepsilon} \cdot \underbrace{\begin{pmatrix} \varepsilon_{iT} \\ \varepsilon_{iT-1} \\ \vdots \\ \varepsilon_{i2} \\ \varepsilon_{i1} \end{pmatrix}}_{E_i},$$

with the names of the matrices and vectors defined below the curly brackets.

For the MA(1) model we have

$$\zeta_{it} = \theta_1 \nu_{it} + \theta_2 \nu_{it-1},$$

where $\nu_{it} \sim N(0,1)$. Again, this can be written in matrix notation

$$\underbrace{\begin{pmatrix} \zeta_{iT} \\ \zeta_{iT-1} \\ \vdots \\ \zeta_{i2} \\ \zeta_{i1} \end{pmatrix}}_{Z_i} = \underbrace{\begin{pmatrix} \theta_1 & \theta_2 & 0 & 0 & 0 & 0 \\ 0 & \theta_1 & \theta_2 & 0 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 & 0 \\ 0 & 0 & 0 & \theta_1 & \theta_2 & 0 \\ 0 & 0 & 0 & 0 & \theta_1 & \theta_2 \end{pmatrix}}_{\Theta} \cdot \underbrace{\begin{pmatrix} v_{iT} \\ v_{iT-1} \\ v_{iT-2} \\ \vdots \\ v_{i2} \\ v_{i1} \\ v_{i0} \end{pmatrix}}_{V_i}.$$

Combining the AR(1) and MA(1) process in one equation, we can write the vector of unobservable outcomes in (3.4.2) as follows

$$\begin{aligned} \begin{pmatrix} U_{1,i} \\ U_{2,i} \end{pmatrix} &= \begin{pmatrix} \Xi_i + Z_i \\ \frac{\rho}{\sqrt{\theta_1^2+\theta_2^2}}\Xi_i + \frac{\rho}{\sqrt{\theta_1^2+\theta_2^2}} Z_i + \sqrt{1-\rho^2} H_i \end{pmatrix} \\ &= \underbrace{\begin{pmatrix} \Phi\Sigma_\varepsilon & \Theta & 0 \\ \frac{\rho}{\sqrt{\theta_1^2+\theta_2^2}}\Phi\Sigma_\varepsilon & \frac{\rho}{\sqrt{\theta_1^2+\theta_2^2}}\Theta & \sqrt{1-\rho^2}\cdot I \end{pmatrix}}_{A} \cdot \begin{pmatrix} E_i \\ V_i \\ H_i \end{pmatrix}, \end{aligned}$$

where H_i is the vector collecting all time periods of η_{it}, and I is the identity matrix. As in example 1, since E_i, V_i, and H_i are all distributed according to the standard normal distribution, the vector $(U_{1,i}, U_{2,i})$ follows a normal distribution as well, with mean 0, and variance-covariance matrix AA'. Again, these can be plugged into (3.2.2) to obtain the joint distribution of Y_i^* and D_i^*.

3.5 Simulations

In this section I present the results for the estimation of two models on simulated data. The first model is the random effects model from example 1. The second part shows the results for the ARMA model from example 2.

3.5.1 Random effects model

The random effects model that I use in this section, is the same model as the simplest model in chapter 2, without time-varying variances. Two methods to approximate the integral can be compared in this case. Because there is only a single random effect, the high-dimensional integral from section 3.2 can be re-written as a one-dimensional integral (Butler & Moffitt, 1982). In principle, approximating a one-dimensional integral accurately should be straightforward. However for values of the number of integration nodes typically used in applications (e.g. 25), we saw some bias in chapter 2 in the parameters related to the variance. I compare this one-dimensional integration to a second method, where the general solution of approximating a T-dimensional integral is used.

The results consist of two parts. In the first part I estimate parameters by maximum likelihood for simulated datasets and compare the bias in the parameter estimates when using the different methods to approximate the log-likelihood. The same number of integration nodes is used for all variations, but for some of the parameters that are used to create the simulated datasets, the value is changed. These experiments show that approximating the 1D integral using pseudo or quasi Monte Carlo simulations leads to large biases in some of the parameters. One-dimensional Gauss-Hermite integration does not result in bias, except for some of the variance parameters, when the correlation between the unobservables is high. Multi-dimensional sparse grid integration and multi-dimensional Monte Carlo integration do not show any large biases.

In the second part, instead of maximizing the log-likelihood, I approximate the log-likelihood at a fixed value for the parameters. At this point, the log-likelihood is approximated with different numbers of integration nodes and I compare these to the 'true' log-likelihood. This experiment shows that 1D Monte Carlo integration results in the poorest accuracy. One-dimensional Gauss-Hermite integration performs best when the number of integration nodes is increased, for instance to 151 or above for $T = 6$. Multi-dimensional sparse grid integration and multi-dimensional Monte Carlo integration result in reasonably accurate approximations for a small number of gridpoints. Increasing the number of gridpoints improves the accuracy, but not as rapidly compared to one-dimensional Gauss-Hermite integration.

The model in this section is the random effects model presented above.

$$\begin{aligned} Y_{it}^{*} &= \beta_{1,0} + \beta_{1,1} x_{1,it} + \qquad\quad U_{1,it} \\ D_{it}^{*} &= \beta_{2,0} + \beta_{2,1} x_{1,it} + \beta_{2,2} x_{2,it} + U_{2,it}, \end{aligned} \tag{3.5.1}$$

where the unobservables are defined as in (3.4.1). There is one covariate in the continuous outcome equation, $x_{1,it} \sim$ i.i.d. $N(0,1)$. The excluded covariate, $x_{2,it}$, is also created using independent draws from the standard normal distribution. For the simulations I use $(\beta_{1,0}, \beta_{1,1}) = (5,1)$, $(\beta_{2,0}, \beta_{2,1}, \beta_{2,2}) = (1,0,1)$, and $\sigma_\varepsilon = 1$. The standard deviation of the random effect takes three different values, $\sigma_\alpha \in (0.5, 1, 2)$, and ρ takes on four different values, $\rho \in (0, 0.3, 0.6, 0.9)$. The intercept in the participation equation determines how many values of the continuous outcome we observe. The value used here, $\beta_{2,0} = 1$, implies that about 75% of the continuous outcomes are observed. Higher values for σ_α and ρ imply a larger correlation between the latent variables.

Below I give the correlation matrix that corresponds to the variance-covariance matrix for two sets of the parameters. To limit the space occupied by these matrices, I set $T = 3$ for these examples. In the random effect model, the correlation between two different time periods is the same, independent of the number of periods separating them. This makes it easy to imagine what the correlation matrix would look like when we increase T. For the first example, I use $\sigma_\varepsilon = 1$, $\sigma_\alpha = 1$, and $\rho = 0.6$.

$$R^{\text{low}} = \left(\begin{array}{ccc|ccc} 1.00 & 0.50 & 0.50 & 0.73 & 0.36 & 0.36 \\ 0.50 & 1.00 & 0.50 & 0.36 & 0.73 & 0.36 \\ 0.50 & 0.50 & 1.00 & 0.36 & 0.36 & 0.73 \\ \hline 0.73 & 0.36 & 0.36 & 1.00 & 0.26 & 0.26 \\ 0.36 & 0.73 & 0.36 & 0.26 & 1.00 & 0.26 \\ 0.36 & 0.36 & 0.73 & 0.26 & 0.26 & 1.00 \end{array}\right)$$

The second example was created using $\sigma_\varepsilon = 1$, $\sigma_\alpha = 2$, and $\rho = 0.9$.

$$R^{\text{high}} = \left(\begin{array}{ccc|ccc} 1.00 & 0.80 & 0.80 & 0.98 & 0.78 & 0.78 \\ 0.80 & 1.00 & 0.80 & 0.78 & 0.98 & 0.78 \\ 0.80 & 0.80 & 1.00 & 0.78 & 0.78 & 0.98 \\ \hline 0.98 & 0.78 & 0.78 & 1.00 & 0.76 & 0.76 \\ 0.78 & 0.98 & 0.78 & 0.76 & 1.00 & 0.76 \\ 0.78 & 0.78 & 0.98 & 0.76 & 0.76 & 1.00 \end{array}\right)$$

The relative sizes of σ_ε and σ_α determine the values off the diagonal in the upper-left corner. A larger value for σ_α relative to σ_ε increases the auto-correlation in Y_{it}^*, as can be seen in R^{high}. The value of ρ affects the relation between the values in the four blocks. For examples, a value of $\rho = 0$, results in a correlation matrix with zeros in the upper-right and lower-left corner. The lower-right corner is a diagonal matrix in that case.

For each combination of the parameters and for different panel lengths, $T \in (6, 11, 16)$, I created R simulated datasets with $N = 500$ individuals. Estimates of the parameters are used to calculate the bias, defined as $\frac{1}{R}\sum_{r=1}^{R}\left(\hat{\theta}_r - \theta_0\right)$, where $\hat{\theta}_r$ is the estimate we obtain for replication r, and θ_0 is the 'true' value of the parameter, i.e. the one that I use to simulate the data. The number of replications, R, is set to 500.

The results of two different ways to approximate the integral inside the log-likelihood are shown. The first method, 1D integration, uses the approach by Butler and Moffitt (1982), where the integration is defined in terms of the likelihood conditional on α_i and the distribution of α_i. To integrate out the normally distributed α_i, I use either Gauss-Hermite integration nodes and weights (GH), a grid of random nodes from the normal distribution (MC), or a set of Halton sequences (Halton).

The second method, xD integration, approximates the T dimensional integral using the GHK simulator, following the algorithm described in Genz (1992). Four methods are used to determine the integration nodes. The first method uses sparse grids to define the grid of integration nodes (SGI). For the first experiment, I choose an accuracy level of $k = 2$, which means that the integration is exact for polynomials of total order $2(k-1) = 3$. This accuracy level determines the number of integration nodes that are used, which is 11, 21

Table 3.1 – Bias in $\beta_{1,0}$ for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	–0.003	–0.002	–0.003	–0.003	–0.003	–0.003	–0.003
	(0.050)	(0.066)	(0.060)	(0.049)	(0.049)	(0.049)	(0.049)
0.6	0.001	0.046	0.034	0.002	0.002	0.001	0.001
	(0.053)	(0.067)	(0.062)	(0.052)	(0.052)	(0.052)	(0.052)
0.9	0.006	**0.052**	0.045	0.003	0.005	0.006	0.006
	(0.050)	(0.065)	(0.057)	(0.050)	(0.049)	(0.049)	(0.049)
$T = 6$, $\sigma_\alpha = 2$							
0	0.003	0.006	0.014	0.005	0.005	0.005	0.005
	(0.185)	(0.178)	(0.176)	(0.088)	(0.088)	(0.088)	(0.088)
0.6	0.019	**0.299**	**0.275**	–0.010	–0.003	–0.002	–0.002
	(0.191)	(0.178)	(0.165)	(0.093)	(0.092)	(0.092)	(0.092)
0.9	0.031	**0.352**	**0.329**	–0.010	–0.001	0.006	0.005
	(0.187)	(0.183)	(0.160)	(0.092)	(0.093)	(0.092)	(0.092)
$T = 16$, $\sigma_\alpha = 1$							
0	0.004	0.004	0.003	0.003	0.003	0.003	0.003
	(0.046)	(0.076)	(0.061)	(0.044)	(0.044)	(0.044)	(0.044)
0.6	–0.003	0.043	0.027	–0.002	–0.002	–0.002	–0.002
	(0.049)	(0.076)	(0.059)	(0.048)	(0.048)	(0.048)	(0.048)
0.9	–0.001	**0.053**	0.036	–0.006	–0.002	–0.002	–0.002
	(0.049)	(0.075)	(0.059)	(0.047)	(0.047)	(0.047)	(0.047)
$T = 16$, $\sigma_\alpha = 2$							
0	0.004	–0.020	–0.017	–0.004	–0.004	–0.004	–0.004
	(0.179)	(0.279)	(0.206)	(0.086)	(0.086)	(0.086)	(0.086)
0.6	0.037	**0.328**	**0.229**	–0.001	0.009	0.009	0.009
	(0.189)	(0.235)	(0.168)	(0.091)	(0.090)	(0.090)	(0.090)
0.9	0.026	**0.344**	**0.239**	**–0.071**	–0.014	–0.009	–0.011
	(0.175)	(0.235)	(0.148)	(0.087)	(0.084)	(0.084)	(0.083)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.2 – Bias in $\beta_{1,1}$ for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.023)	(0.025)	(0.024)	(0.023)	(0.023)	(0.023)	(0.023)
0.6	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)
0.9	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.022)	(0.023)	(0.022)	(0.022)	(0.022)	(0.022)	(0.022)
$T = 6$, $\sigma_\alpha = 2$							
0	–0.001	–0.002	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.024)	(0.027)	(0.024)	(0.023)	(0.023)	(0.023)	(0.023)
0.6	0.002	0.002	0.002	0.002	0.002	0.002	0.002
	(0.025)	(0.027)	(0.024)	(0.023)	(0.023)	(0.023)	(0.023)
0.9	0.000	–0.001	0.000	–0.001	–0.001	–0.001	–0.001
	(0.024)	(0.025)	(0.023)	(0.022)	(0.023)	(0.023)	(0.023)
$T = 16$, $\sigma_\alpha = 1$							
0	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.013)	(0.013)	(0.013)	(0.013)	(0.013)	(0.013)	(0.013)
0.6	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.013)	(0.013)	(0.013)	(0.013)	(0.013)	(0.013)	(0.013)
0.9	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.012)	(0.012)	(0.012)	(0.012)	(0.012)	(0.012)	(0.012)
$T = 16$, $\sigma_\alpha = 2$							
0	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.014)	(0.014)	(0.014)	(0.014)	(0.014)	(0.014)	(0.014)
0.6	0.000	0.000	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.013)	(0.014)	(0.013)	(0.013)	(0.013)	(0.013)	(0.013)
0.9	0.001	0.002	0.002	0.002	0.002	0.002	0.002
	(0.014)	(0.015)	(0.014)	(0.014)	(0.014)	(0.014)	(0.014)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.3 – Bias in $\beta_{2,0}$ for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	0.002	0.002	0.002	0.002	0.002	0.002	0.002
	(0.033)	(0.033)	(0.033)	(0.033)	(0.033)	(0.033)	(0.033)
0.6	0.002	0.015	0.019	0.000	0.002	0.002	0.002
	(0.044)	(0.050)	(0.048)	(0.044)	(0.044)	(0.044)	(0.044)
0.9	0.006	0.012	0.033	0.002	0.005	0.007	0.007
	(0.053)	(0.062)	(0.059)	(0.053)	(0.053)	(0.053)	(0.053)
$T = 6$, $\sigma_\alpha = 2$							
0	0.003	0.003	0.003	0.003	0.003	0.003	0.003
	(0.032)	(0.032)	(0.032)	(0.032)	(0.032)	(0.032)	(0.032)
0.6	0.000	**0.132**	**0.162**	–0.010	–0.004	–0.001	–0.002
	(0.119)	(0.107)	(0.105)	(0.067)	(0.067)	(0.067)	(0.067)
0.9	–0.005	**0.204**	**0.277**	–0.012	–0.004	0.004	0.003
	(0.163)	(0.151)	(0.145)	(0.087)	(0.087)	(0.087)	(0.087)
$T = 16$, $\sigma_\alpha = 1$							
0	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.021)	(0.021)	(0.021)	(0.021)	(0.021)	(0.021)	(0.021)
0.6	–0.002	0.023	0.015	–0.004	–0.002	–0.002	–0.002
	(0.035)	(0.049)	(0.040)	(0.034)	(0.034)	(0.034)	(0.034)
0.9	0.000	0.043	0.033	–0.010	–0.002	–0.001	–0.002
	(0.046)	(0.069)	(0.055)	(0.045)	(0.045)	(0.045)	(0.045)
$T = 16$, $\sigma_\alpha = 2$							
0	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.022)	(0.022)	(0.022)	(0.022)	(0.022)	(0.022)	(0.022)
0.6	0.020	**0.193**	**0.139**	–0.005	0.006	0.006	0.006
	(0.117)	(0.143)	(0.104)	(0.061)	(0.060)	(0.060)	(0.060)
0.9	0.014	**0.297**	**0.215**	**–0.076**	–0.015	–0.009	–0.010
	(0.156)	(0.208)	(0.135)	(0.079)	(0.076)	(0.076)	(0.076)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.4 – Bias in $\beta_{2,1}$ for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	0.003	0.003	0.003	0.003	0.003	0.003	0.003
	(0.030)	(0.030)	(0.030)	(0.030)	(0.030)	(0.030)	(0.030)
0.6	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.031)	(0.030)	(0.030)	(0.031)	(0.031)	(0.031)	(0.031)
0.9	0.000	–0.001	0.000	0.000	0.000	0.000	0.000
	(0.028)	(0.028)	(0.028)	(0.028)	(0.028)	(0.028)	(0.028)
$T = 6$, $\sigma_\alpha = 2$							
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.031)	(0.031)	(0.031)	(0.031)	(0.031)	(0.031)	(0.031)
0.6	0.000	0.000	0.001	0.000	0.000	0.000	0.000
	(0.032)	(0.031)	(0.032)	(0.032)	(0.032)	(0.032)	(0.032)
0.9	0.000	–0.001	0.000	0.000	0.000	0.000	0.000
	(0.031)	(0.029)	(0.031)	(0.031)	(0.031)	(0.031)	(0.031)
$T = 16$, $\sigma_\alpha = 1$							
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)
0.6	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)
0.9	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.018)	(0.017)	(0.018)	(0.017)	(0.018)	(0.018)	(0.018)
$T = 16$, $\sigma_\alpha = 2$							
0	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)
0.6	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.019)	(0.019)	(0.019)	(0.019)	(0.019)	(0.019)	(0.019)
0.9	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.5 – Bias in $\beta_{2,2}$ for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	0.004	0.004	0.004	0.004	0.004	0.004	0.004
	(0.039)	(0.039)	(0.039)	(0.039)	(0.039)	(0.039)	(0.039)
0.6	0.003	–0.016	–0.003	0.003	0.003	0.003	0.003
	(0.040)	(0.039)	(0.039)	(0.040)	(0.040)	(0.040)	(0.040)
0.9	0.000	–0.040	–0.012	0.000	0.000	0.001	0.001
	(0.034)	(0.033)	(0.033)	(0.034)	(0.034)	(0.034)	(0.034)
$T = 6$, $\sigma_\alpha = 2$							
0	0.002	0.002	0.002	0.002	0.002	0.002	0.002
	(0.038)	(0.038)	(0.038)	(0.038)	(0.038)	(0.038)	(0.038)
0.6	–0.017	**–0.056**	–0.014	–0.005	–0.004	–0.002	–0.002
	(0.040)	(0.039)	(0.040)	(0.040)	(0.040)	(0.040)	(0.040)
0.9	–0.033	**–0.100**	–0.023	–0.001	–0.001	0.002	0.002
	(0.031)	(0.032)	(0.033)	(0.033)	(0.033)	(0.033)	(0.033)
$T = 16$, $\sigma_\alpha = 1$							
0	0.002	0.002	0.002	0.002	0.002	0.002	0.002
	(0.025)	(0.025)	(0.025)	(0.025)	(0.025)	(0.025)	(0.025)
0.6	0.003	0.000	0.002	0.002	0.003	0.003	0.003
	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)
0.9	0.002	–0.006	–0.001	0.000	0.001	0.001	0.001
	(0.020)	(0.020)	(0.020)	(0.020)	(0.020)	(0.020)	(0.020)
$T = 16$, $\sigma_\alpha = 2$							
0	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)
0.6	–0.004	–0.007	0.000	–0.003	0.000	0.000	0.000
	(0.024)	(0.025)	(0.024)	(0.024)	(0.024)	(0.024)	(0.024)
0.9	–0.012	–0.014	–0.002	–0.009	–0.002	–0.001	–0.001
	(0.019)	(0.019)	(0.019)	(0.019)	(0.019)	(0.019)	(0.019)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

and 31 nodes for $T = 6$, $T = 11$ and $T = 16$ respectively. As a second method, I again use a grid of random points (MC). The final two methods are based on Halton sequences, where the distinction between the two is whether or not the sequences for different dimensions are shuffled when combining them in a multi-dimensional grid (Halton and Halton shuffled).

Conditional on the number of periods, the four integration methods use the same number of nodes. However, there is a slight difference between the definition of a node for 1D integration and for xD integration. A node is a single number in the case of 1D integration. For xD integration, a node is a vector of $T - 1$ numbers. This implies that for one-dimensional MC integration, for one individual the same random draws are used to approximate the likelihood in each of the separate periods. Whereas different random draws are used in the case of Monte Carlo xD integration. This corresponds to integrating out the distribution of the one-dimensional α_i, or integrating out the distribution of multiple dependent random variables.

Since the same number of nodes is used for all integration methods, the computer run times are very similar. In the approximation of the integral, most time is spent on evaluating $\Phi(\cdot)$, the standard normal CDF. Independent of the approximation method, for a given individual, this function is evaluated T times for every integration node. Increasing the number of individuals, the number of time periods, or the number of integration nodes, all have a similar effect on the total run time. There is no trade-off in computation time between the different methods, so in order to determine which method to use for a given number of nodes, we only have to compare the accuracy of the approximation.

Tables 3.1 to 3.8 show the results from estimating the random effect model. Each table shows the bias in one of the estimated parameters. To economize on space, only a subset of all experiments is shown in the table. Different values for σ_α and ρ were used to simulate datasets. The three sets of rows separated by whitespace, correspond to different values of σ_α, $\sigma_\alpha \in \{1, 2\}$. Within the sets of rows the results for three different values of ρ are shown. The top and bottom half show the difference between two lengths of the panel $T = 6$ and $T = 16$. The left three columns correspond to 1D integration and the four columns on the right correspond to xD integration. The results for $T = 11$ and the results for $\sigma_\alpha = 0.5$ or $\rho = 0.3$ show the same pattern. Biases larger than 0.05 in absolute

value are shown in bold.

Table 3.1 shows the bias in $\beta_{1,0}$, the intercept in the continuous outcome equation. We see that there is substantial bias when the log-likelihood is approximated using 1D Monte Carlo integration or 1D Halton sequences, and the auto-correlation in the error is high, $\sigma_\alpha = 2$. For this type of integration, 11 nodes for $T = 6$, and 31 nodes for $T = 16$ does not result in an accurate approximation of the log-likelihood, leading to biased estimates. When the correlation in the unobservables is largest, $\sigma_\alpha = 2$, and $\rho = 0.9$, there is also a bit of bias for the other approximation methods, most notably for xD integration using sparse grids. There are no large differences in bias when looking at the different panel lengths, $T = 6$ and $T = 16$. This is a consequence of using more nodes to approximate the integral for $T = 16$, than for $T = 6$.

The second difference that we see, is when we compare the standard deviations of the bias for 1D integration and xD integration. For small values of σ_α, there are no apparent differences between the two methods. For $\sigma_\alpha = 2$, the standard deviation for xD integration, both for SGI and MC, is less than half the size of the standard deviation for 1D Gauss-Hermite integration. This suggests that xD integration results in a more accurate approximation of the log-likelihood for this number of nodes, compared to 1D integration. This is confirmed in the experiment described below, where I look directly at the accuracy of the log-likelihood approximation for different numbers of integration nodes.

In table 3.2 we see that there is no bias in the estimates for $\beta_{1,1}$, the coefficient on the covariate in the continuous outcome equation. The only difference that we see, is that the standard deviation of the bias is smaller for $T = 16$ than for $T = 6$. Since the number of individuals is the same for both experiments, and we have observations for all individuals in all time periods, the number of observations is $16/6 = 2\frac{2}{3}$ times larger for $T = 16$, than for $T = 6$. Since the precision of an estimate increases with the square-root of the number of observations, we expect the standard deviation of the bias to be $\sqrt{2\frac{2}{3}} \approx 1.6$ larger for $T = 6$ compared to $T = 16$, which is precisely what we see in the table.

The bias for the intercept of the discrete outcome equation is given in table 3.3. The results are the same as for the intercept in the continuous outcome equation in table

3.1. I.e. if the correlation between the unobservables is high, then using Monte Carlo integration or Halton sequences to approximate the 1D integral leads to large biases. Also, for the high correlation case the approximation of the log-likelihood is more accurate for xD integration, compared to 1D integration, for the number of integration nodes shown in this table. Similar to the coefficient on the covariate in the outcome equation in table 3.2, the coefficient on the covariate in the discrete outcome equation does not exhibit bias, as can be seen in table 3.4. Table 3.5 shows that the coefficient on the instrument, $\beta_{2,2}$ does show bias for 1D Monte Carlo integration when the correlation between the unobservables is high.

Tables 3.6, 3.7, and 3.8 show the bias in σ_ε, σ_α, and ρ respectively. Again, we see that 1D Monte Carlo integration leads to large biases in σ_ε and σ_α. We also see that 1D Gauss-Hermite integration and 1D Halton sequences lead to somewhat biased estimates for σ_ε when $T = 6$ and $\sigma_\alpha = 2$. The largest bias is found for σ_α in table 3.7, where all three 1D integration methods lead to biased estimates when the correlation between unobservables is large. Gauss-Hermite integration results in a downward bias, whereas Monte Carlo integration and Halton sequences result in an upward bias for this parameter.

In a second set of experiments I compare different approximations to the log-likelihood with the 'true' value of the log-likelihood. This experiment is similar to the exercise presented in Heiss (2010). Instead of calculating estimates of the parameters from simulated data by maximizing the log-likelihood function, I evaluate approximations to the log-likelihood at fixed values of the parameters. The same set of parameters as above are used to simulate data for N individuals, where $N = 1000$. I.e. I use $(\beta_{1,0}, \beta_{1,1}) = (5, 1)$, $(\beta_{2,0}, \beta_{2,1}, \beta_{2,2}) = (1, 0, 1)$, and $\sigma_\varepsilon = 1$. Again, σ_α and ρ take on different values, $\sigma_\alpha \in (0.5, 1, 2)$, and $\rho \in (0, 0.3, 0.6, 0.9)$. For each of the 1000 individuals I approximate the individual contribution to the log-likelihood at the same values for the parameters that were used to simulate the data. I use the same parameters that were used to simulate the data, because we are mostly interested in the quality of the approximation to the log-likelihood around the maximal value of the log-likelihood. This results in 1000 approximated log-likelihood contributions for every set of parameters.

We want to compare these approximated individual contributions to the log-likelihood

Table 3.6 – Bias in σ_ε for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	0.000	**0.055**	0.017	–0.001	–0.001	–0.001	–0.001
	(0.016)	(0.019)	(0.017)	(0.016)	(0.016)	(0.016)	(0.016)
0.6	0.001	**0.052**	0.017	–0.001	–0.001	–0.001	–0.001
	(0.019)	(0.021)	(0.019)	(0.019)	(0.019)	(0.019)	(0.019)
0.9	0.000	0.047	0.015	–0.001	–0.001	–0.001	–0.001
	(0.017)	(0.020)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)
$T = 6$, $\sigma_\alpha = 2$							
0	0.041	**0.168**	**0.050**	0.000	0.000	0.000	0.000
	(0.017)	(0.030)	(0.022)	(0.017)	(0.017)	(0.017)	(0.017)
0.6	0.041	**0.141**	0.040	–0.002	–0.001	–0.001	–0.001
	(0.019)	(0.032)	(0.022)	(0.019)	(0.019)	(0.019)	(0.019)
0.9	0.039	**0.131**	0.037	–0.002	–0.001	–0.001	–0.001
	(0.019)	(0.031)	(0.022)	(0.019)	(0.019)	(0.019)	(0.019)
$T = 16$, $\sigma_\alpha = 1$							
0	0.001	0.011	0.004	0.001	0.001	0.001	0.001
	(0.009)	(0.009)	(0.009)	(0.009)	(0.009)	(0.009)	(0.009)
0.6	–0.001	0.008	0.002	–0.002	–0.001	–0.001	–0.001
	(0.010)	(0.011)	(0.010)	(0.010)	(0.010)	(0.010)	(0.010)
0.9	–0.001	0.008	0.002	–0.002	–0.001	–0.001	–0.001
	(0.011)	(0.011)	(0.011)	(0.011)	(0.011)	(0.011)	(0.011)
$T = 16$, $\sigma_\alpha = 2$							
0	0.011	0.024	0.006	0.000	0.000	0.000	0.000
	(0.009)	(0.011)	(0.010)	(0.009)	(0.009)	(0.009)	(0.009)
0.6	0.012	0.021	0.006	–0.001	0.000	0.000	0.000
	(0.010)	(0.012)	(0.011)	(0.010)	(0.010)	(0.010)	(0.010)
0.9	0.013	0.020	0.007	–0.001	0.001	0.001	0.001
	(0.011)	(0.013)	(0.011)	(0.011)	(0.011)	(0.011)	(0.011)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.7 – Bias in σ_α for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	–0.004	**0.067**	**0.083**	0.000	0.000	0.000	0.000
	(0.036)	(0.063)	(0.058)	(0.037)	(0.037)	(0.037)	(0.037)
0.6	–0.009	**0.059**	**0.074**	–0.005	–0.004	–0.004	–0.004
	(0.039)	(0.061)	(0.059)	(0.040)	(0.040)	(0.040)	(0.040)
0.9	–0.008	**0.066**	**0.078**	–0.002	–0.003	–0.002	–0.002
	(0.039)	(0.064)	(0.060)	(0.040)	(0.040)	(0.040)	(0.040)
$T = 6$, $\sigma_\alpha = 2$							
0	**–0.240**	**0.449**	**0.518**	–0.001	–0.001	–0.001	–0.001
	(0.072)	(0.190)	(0.204)	(0.068)	(0.068)	(0.068)	(0.068)
0.6	**–0.258**	**0.356**	**0.410**	0.006	–0.002	–0.001	–0.001
	(0.070)	(0.162)	(0.180)	(0.072)	(0.071)	(0.070)	(0.071)
0.9	**–0.282**	**0.361**	**0.391**	0.021	0.004	0.000	0.001
	(0.071)	(0.169)	(0.190)	(0.074)	(0.075)	(0.075)	(0.075)
$T = 16$, $\sigma_\alpha = 1$							
0	–0.003	**0.135**	**0.081**	–0.002	–0.002	–0.002	–0.002
	(0.034)	(0.070)	(0.059)	(0.034)	(0.034)	(0.034)	(0.034)
0.6	–0.002	**0.127**	**0.075**	–0.001	–0.001	–0.001	–0.001
	(0.034)	(0.064)	(0.056)	(0.034)	(0.034)	(0.034)	(0.034)
0.9	–0.001	**0.131**	**0.074**	0.000	0.000	0.000	0.000
	(0.036)	(0.067)	(0.057)	(0.037)	(0.036)	(0.036)	(0.036)
$T = 16$, $\sigma_\alpha = 2$							
0	**–0.203**	**0.641**	**0.442**	–0.007	–0.007	–0.007	–0.007
	(0.085)	(0.253)	(0.212)	(0.064)	(0.064)	(0.064)	(0.064)
0.6	**–0.211**	**0.486**	**0.317**	0.008	–0.002	–0.002	–0.002
	(0.081)	(0.238)	(0.189)	(0.071)	(0.069)	(0.068)	(0.069)
0.9	**–0.225**	**0.450**	**0.292**	**0.071**	0.007	0.002	0.004
	(0.084)	(0.223)	(0.191)	(0.072)	(0.065)	(0.065)	(0.065)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.8 – Bias in ρ for different values of σ_α and ρ

	1D integration			xD integration			
	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
ρ							
$T = 6$, $\sigma_\alpha = 1$							
0	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.030)	(0.032)	(0.031)	(0.029)	(0.029)	(0.029)	(0.029)
0.6	0.001	0.025	0.010	–0.002	0.000	0.000	0.000
	(0.031)	(0.031)	(0.031)	(0.031)	(0.031)	(0.031)	(0.031)
0.9	0.002	0.011	0.006	0.000	0.001	0.001	0.001
	(0.014)	(0.014)	(0.014)	(0.015)	(0.014)	(0.014)	(0.014)
$T = 6$, $\sigma_\alpha = 2$							
0	–0.002	–0.002	–0.002	–0.001	–0.001	–0.001	–0.001
	(0.016)	(0.018)	(0.016)	(0.015)	(0.015)	(0.015)	(0.015)
0.6	0.021	**0.058**	0.020	–0.001	–0.001	0.000	0.000
	(0.023)	(0.025)	(0.023)	(0.023)	(0.023)	(0.023)	(0.023)
0.9	0.008	0.023	0.010	–0.001	–0.001	0.000	0.000
	(0.013)	(0.012)	(0.013)	(0.014)	(0.014)	(0.014)	(0.014)
$T = 16$, $\sigma_\alpha = 1$							
0	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.018)	(0.019)	(0.018)	(0.018)	(0.018)	(0.018)	(0.018)
0.6	–0.001	0.004	0.001	–0.004	–0.001	–0.001	–0.001
	(0.017)	(0.017)	(0.017)	(0.017)	(0.017)	(0.017)	(0.017)
0.9	0.000	0.002	0.001	–0.002	0.000	0.000	0.000
	(0.008)	(0.008)	(0.008)	(0.009)	(0.008)	(0.008)	(0.008)
$T = 16$, $\sigma_\alpha = 2$							
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.009)	(0.010)	(0.009)	(0.009)	(0.009)	(0.009)	(0.009)
0.6	0.006	0.010	0.003	–0.002	0.000	0.000	0.000
	(0.014)	(0.014)	(0.014)	(0.013)	(0.013)	(0.013)	(0.013)
0.9	0.004	0.006	0.003	–0.002	0.000	0.001	0.001
	(0.008)	(0.008)	(0.008)	(0.008)	(0.008)	(0.008)	(0.008)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$ for xD integration. SGI accuracy is 2, which implies that 11 integration nodes are used for $T = 6$, and 31 integration nodes are used for $T = 16$. Values for the bias larger than 0.05 in absolute value are in bold.

to the 'true' contribution to the log-likelihood for each individual. However, this 'true' contribution is not available. Instead, I use Monte Carlo approximation using xD integration with 1,000,000 integration nodes and refer to this as the truth. This true value can then be used to calculate a measure of the approximation error, averaging over the log-likelihood contributions of the 1000 individuals. The measure that I use, is the mean absolute percentage error (MAPE), defined as

$$\frac{1}{N}\sum_{i=1}^{N}\left|100\cdot\frac{\log\mathcal{L}_i-\log\mathcal{L}_i^0}{\log\mathcal{L}_i^0}\right|,$$

where $\log\mathcal{L}_i$ is the approximated log-likelihood contribution for individual i evaluated at values for the parmeters defined above, and $\log\mathcal{L}_i^0$ is the 'true' log-likelihood for individual i, i.e. without approximation error.

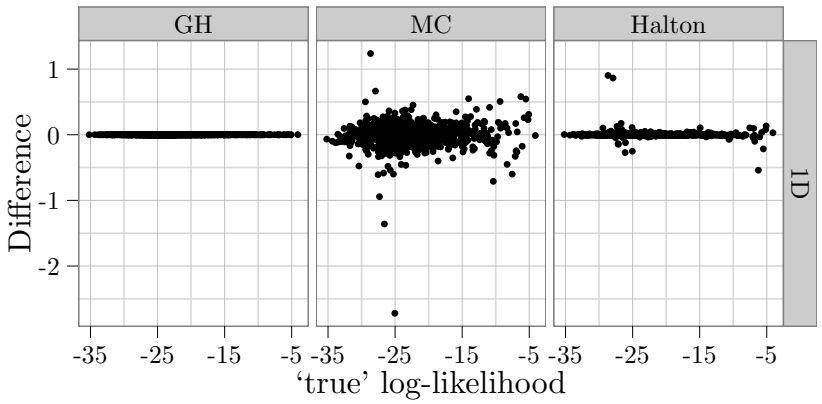

Figure 3.5 – Difference between 'true' and approximated log-likelihood for 1D integration with $T = 16$, $\sigma_\alpha = 2$, and $\rho = 0.6$

Before looking at the mean absolute percentage error, a combined measure, I first show the difference between the 'true' log-likelihood and its approximation for each of the 1000 simulated individuals separately. Figure 3.5 shows these differences for 1D integration, where $T = 16$, $\sigma_\alpha = 2$ and $\rho = 0.6$. The same figures are shown for xD integration in figure 3.6. The approximations in these figures are obtained using 451 integration nodes, which corresponds to sparse grid accuracy, $k = 3$. The patterns that we see in these figures are the same for other values of T, σ_α, and ρ and for different numbers of integration nodes.

From figure 3.5 we see that for 1D integration the error in the approximation is smallest for Gauss-Hermite integration. Monte Carlo integration has the largest variation in

approximation error and integration using Halton sequences is in between the two. They can not be directly compared with 3.6, because of the difference in scale on the y-axis. For xD integration the variation in error seems smallest for Halton sequences, and slightly larger for its shuffled version. Both sparse grid integration and Monte Carlo integration have more variation. The approximation that is obtained using sparse grid integration is always smaller than the true log-likelihood, all the differences are smaller than zero, whereas the other approximation methods are centered around the true-loglikelihood.

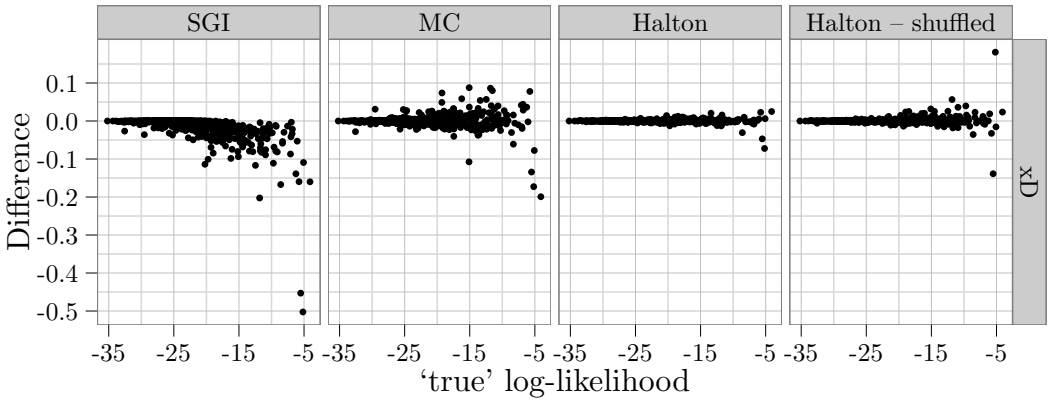

Figure 3.6 – Difference between 'true' and approximated log-likelihood for xD integration with $T = 16$, $\sigma_\alpha = 2$, and $\rho = 0.6$

Tables 3.9 to 3.11 show the MAPE of the log-likelihood approximation for $T = 6$, $T = 11$, and $T = 16$. The effect of different values for ρ are shown in the four horizontal blocks. For σ_α a value of 2.0 is used. Again, there are three columns related to the three variants of 1D integration, and four columns related to the variants of xD integration. Values 2, 3, 4, 5, and 6 are used as SGI accuracies, and the corresponding number of gridpoints used to approximate the integral, is shown in the adjacent column. The conclusion from the results presented below are the same if root mean-squared error (RMSE) is used as a measure to compare the different methods. Tables with RMSE are not included in this paper.

From table 3.9 we can see that all three types of 1D integration lead to a poor approximation of the integral in terms of mean absolute percentage error when using 11 integration nodes. The MAPE is about 10% for Monte Carlo integration and about 5% for Gauss-Hermite integration and Halton sequences. From the previous experiment we

Table 3.9 – MAPE of log-likelihood approximation, $T = 6, \sigma_\alpha = 2.0$

		1D integration			xD integration			
Accuracy	No. of points	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
$\rho = 0$								
2	11	$4.20 \times 10^{+0}$	$9.99 \times 10^{+0}$	$4.04 \times 10^{+0}$	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}
3	51	7.90×10^{-2}	$2.67 \times 10^{+0}$	7.84×10^{-1}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}
4	151	1.48×10^{-5}	$1.50 \times 10^{+0}$	2.26×10^{-1}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}
5	391	1.17×10^{-10}	8.40×10^{-1}	8.00×10^{-2}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}
6	903	1.17×10^{-10}	5.43×10^{-1}	3.79×10^{-2}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}	1.17×10^{-10}
$\rho = 0.3$								
2	11	$4.35 \times 10^{+0}$	$9.82 \times 10^{+0}$	$4.22 \times 10^{+0}$	8.73×10^{-3}	5.25×10^{-2}	2.35×10^{-2}	2.53×10^{-2}
3	51	8.52×10^{-2}	$2.86 \times 10^{+0}$	8.68×10^{-1}	8.75×10^{-3}	1.89×10^{-2}	6.16×10^{-3}	7.20×10^{-3}
4	151	1.71×10^{-4}	$1.50 \times 10^{+0}$	2.25×10^{-1}	1.15×10^{-3}	1.27×10^{-2}	2.44×10^{-3}	2.30×10^{-3}
5	391	1.54×10^{-4}	8.90×10^{-1}	8.44×10^{-2}	1.16×10^{-3}	7.14×10^{-3}	9.44×10^{-4}	1.03×10^{-3}
6	903	1.54×10^{-4}	5.61×10^{-1}	3.71×10^{-2}	1.16×10^{-3}	6.21×10^{-3}	5.17×10^{-4}	5.96×10^{-4}
$\rho = 0.6$								
2	11	$4.49 \times 10^{+0}$	$1.04 \times 10^{+1}$	$4.65 \times 10^{+0}$	1.08×10^{-1}	4.36×10^{-1}	2.03×10^{-1}	2.10×10^{-1}
3	51	1.00×10^{-1}	$3.22 \times 10^{+0}$	9.06×10^{-1}	1.00×10^{-1}	2.14×10^{-1}	5.38×10^{-2}	7.62×10^{-2}
4	151	1.39×10^{-3}	$1.70 \times 10^{+0}$	2.51×10^{-1}	1.09×10^{-2}	1.25×10^{-1}	2.11×10^{-2}	2.80×10^{-2}
5	391	1.37×10^{-3}	$1.01 \times 10^{+0}$	9.25×10^{-2}	1.21×10^{-2}	7.06×10^{-2}	8.85×10^{-3}	1.92×10^{-2}
6	903	1.37×10^{-3}	6.72×10^{-1}	4.20×10^{-2}	1.21×10^{-2}	4.35×10^{-2}	5.03×10^{-3}	8.68×10^{-3}
$\rho = 0.9$								
2	11	$5.08 \times 10^{+0}$	$1.25 \times 10^{+1}$	$5.08 \times 10^{+0}$	2.02×10^{-1}	$1.13 \times 10^{+0}$	4.92×10^{-1}	5.40×10^{-1}
3	51	1.25×10^{-1}	$3.66 \times 10^{+0}$	$1.00 \times 10^{+0}$	1.78×10^{-1}	5.40×10^{-1}	1.24×10^{-1}	1.63×10^{-1}
4	151	3.77×10^{-3}	$2.11 \times 10^{+0}$	2.89×10^{-1}	1.89×10^{-2}	3.25×10^{-1}	4.79×10^{-2}	7.71×10^{-2}
5	391	3.74×10^{-3}	$1.24 \times 10^{+0}$	1.05×10^{-1}	1.64×10^{-2}	1.90×10^{-1}	2.37×10^{-2}	4.40×10^{-2}
6	903	3.74×10^{-3}	8.07×10^{-1}	4.35×10^{-2}	1.61×10^{-2}	1.19×10^{-1}	1.04×10^{-2}	2.48×10^{-2}

The 'true' log-likelihood used to calculate the mean absolute percentage error (MAPE), is based on xD Monte Carlo integration with 1,000,000 integration nodes.

Table 3.10 – MAPE of log-likelihood approximation, $T = 11, \sigma_\alpha = 2.0$

		1D integration			xD integration			
Accuracy	No. of points	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
$\rho = 0$								
2	21	$2.30 \times 10^{+0}$	$5.64 \times 10^{+0}$	$2.42 \times 10^{+0}$	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}
3	201	1.49×10^{-4}	$1.04 \times 10^{+0}$	2.24×10^{-1}	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}
4	1201	6.58×10^{-11}	3.60×10^{-1}	2.29×10^{-2}	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}
5	5281	–	1.62×10^{-1}	4.90×10^{-3}	6.57×10^{-11}	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}
6	19105	–	8.68×10^{-2}	1.61×10^{-3}	6.57×10^{-11}	6.58×10^{-11}	6.58×10^{-11}	6.58×10^{-11}
$\rho = 0.3$								
2	21	$2.28 \times 10^{+0}$	$5.55 \times 10^{+0}$	$2.36 \times 10^{+0}$	1.41×10^{-2}	3.19×10^{-2}	1.56×10^{-2}	1.16×10^{-2}
3	201	3.45×10^{-4}	$1.02 \times 10^{+0}$	2.07×10^{-1}	1.10×10^{-2}	9.09×10^{-3}	2.83×10^{-3}	3.17×10^{-3}
4	1201	1.50×10^{-4}	3.81×10^{-1}	2.26×10^{-2}	2.91×10^{-3}	4.45×10^{-3}	4.91×10^{-4}	7.24×10^{-4}
5	5281	–	1.72×10^{-1}	5.91×10^{-3}	1.86×10^{-3}	2.25×10^{-3}	2.01×10^{-4}	4.06×10^{-4}
6	19105	–	9.11×10^{-2}	1.47×10^{-3}	1.77×10^{-3}	8.73×10^{-4}	1.56×10^{-4}	3.38×10^{-4}
$\rho = 0.6$								
2	21	$2.31 \times 10^{+0}$	$5.82 \times 10^{+0}$	$2.40 \times 10^{+0}$	1.78×10^{-1}	3.16×10^{-1}	1.44×10^{-1}	1.71×10^{-1}
3	201	1.61×10^{-3}	$1.16 \times 10^{+0}$	2.25×10^{-1}	1.33×10^{-1}	8.90×10^{-2}	2.15×10^{-2}	2.80×10^{-2}
4	1201	1.33×10^{-3}	4.52×10^{-1}	2.51×10^{-2}	2.17×10^{-2}	3.60×10^{-2}	4.19×10^{-3}	1.13×10^{-2}
5	5281	–	2.03×10^{-1}	5.84×10^{-3}	1.82×10^{-2}	1.91×10^{-2}	1.81×10^{-3}	4.92×10^{-3}
6	19105	–	1.05×10^{-1}	2.30×10^{-3}	1.86×10^{-2}	8.98×10^{-3}	1.39×10^{-3}	3.79×10^{-3}
$\rho = 0.9$								
2	21	$2.63 \times 10^{+0}$	$6.83 \times 10^{+0}$	$2.66 \times 10^{+0}$	4.93×10^{-1}	$1.03 \times 10^{+0}$	4.14×10^{-1}	4.86×10^{-1}
3	201	4.65×10^{-3}	$1.45 \times 10^{+0}$	2.63×10^{-1}	3.71×10^{-1}	3.26×10^{-1}	5.90×10^{-2}	1.14×10^{-1}
4	1201	4.29×10^{-3}	5.73×10^{-1}	3.09×10^{-2}	6.37×10^{-2}	1.22×10^{-1}	1.31×10^{-2}	4.08×10^{-2}
5	5281	–	2.74×10^{-1}	8.17×10^{-3}	3.93×10^{-2}	6.57×10^{-2}	5.69×10^{-3}	1.92×10^{-2}
6	19105	–	1.38×10^{-1}	5.13×10^{-3}	3.70×10^{-2}	3.14×10^{-2}	4.47×10^{-3}	1.06×10^{-2}

The 'true' log-likelihood used to calculate the mean absolute percentage error (MAPE), is based on xD Monte Carlo integration with 1,000,000 integration nodes. 1D Gauss-Hermite integration with more than 1250 nodes has not been calculated.

Table 3.11 – MAPE of log-likelihood approximation, $T = 16, \sigma_\alpha = 2.0$

		1D integration			xD integration			
Accuracy	No. of points	GH	MC	Halton	SGI	MC	Halton	Halton shuffled
$\rho = 0$								
2	31	$1.50 \times 10^{+0}$	$4.09 \times 10^{+0}$	$1.09 \times 10^{+0}$	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}
3	451	3.25×10^{-7}	4.46×10^{-1}	9.22×10^{-2}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}
4	4151	–	1.58×10^{-1}	5.57×10^{-3}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}
5	27671	–	5.59×10^{-2}	7.21×10^{-4}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}
6	145607	–	2.37×10^{-2}	1.81×10^{-4}	4.88×10^{-11}	4.90×10^{-11}	4.90×10^{-11}	4.90×10^{-11}
$\rho = 0.3$								
2	31	$1.54 \times 10^{+0}$	$4.10 \times 10^{+0}$	$1.02 \times 10^{+0}$	6.77×10^{-3}	1.64×10^{-2}	7.51×10^{-3}	7.63×10^{-3}
3	451	8.05×10^{-5}	4.82×10^{-1}	7.11×10^{-2}	6.42×10^{-3}	4.01×10^{-3}	9.69×10^{-4}	1.03×10^{-3}
4	4151	–	1.59×10^{-1}	6.08×10^{-3}	1.01×10^{-3}	1.28×10^{-3}	1.43×10^{-4}	1.63×10^{-4}
5	27671	–	5.99×10^{-2}	8.14×10^{-4}	8.88×10^{-4}	4.86×10^{-4}	8.24×10^{-5}	8.64×10^{-5}
6	145607	–	2.56×10^{-2}	2.21×10^{-4}	8.86×10^{-4}	2.34×10^{-4}	8.00×10^{-5}	8.20×10^{-5}
$\rho = 0.6$								
2	31	$1.55 \times 10^{+0}$	$4.27 \times 10^{+0}$	9.90×10^{-1}	1.55×10^{-1}	1.71×10^{-1}	9.88×10^{-2}	7.23×10^{-2}
3	451	7.13×10^{-4}	5.52×10^{-1}	7.17×10^{-2}	9.16×10^{-2}	4.34×10^{-2}	9.92×10^{-3}	1.84×10^{-2}
4	4151	–	1.87×10^{-1}	6.97×10^{-3}	2.43×10^{-2}	1.38×10^{-2}	1.41×10^{-3}	4.00×10^{-3}
5	27671	–	7.36×10^{-2}	1.45×10^{-3}	1.23×10^{-2}	4.53×10^{-3}	7.61×10^{-4}	2.04×10^{-3}
6	145607	–	3.09×10^{-2}	8.18×10^{-4}	1.50×10^{-2}	2.80×10^{-3}	7.13×10^{-4}	1.08×10^{-3}
$\rho = 0.9$								
2	31	$1.70 \times 10^{+0}$	$4.68 \times 10^{+0}$	9.76×10^{-1}	5.52×10^{-1}	6.01×10^{-1}	2.49×10^{-1}	3.07×10^{-1}
3	451	3.11×10^{-3}	6.82×10^{-1}	6.91×10^{-2}	3.39×10^{-1}	1.66×10^{-1}	3.24×10^{-2}	6.25×10^{-2}
4	4151	–	2.41×10^{-1}	8.02×10^{-3}	8.71×10^{-2}	5.22×10^{-2}	5.15×10^{-3}	1.86×10^{-2}
5	27671	–	8.98×10^{-2}	3.49×10^{-3}	4.52×10^{-2}	1.80×10^{-2}	3.15×10^{-3}	7.49×10^{-3}
6	145607	–	3.89×10^{-2}	3.19×10^{-3}	4.42×10^{-2}	9.07×10^{-3}	3.14×10^{-3}	4.31×10^{-3}

The 'true' log-likelihood used to calculate the mean absolute percentage error (MAPE), is based on xD Monte Carlo integration with 1,000,000 integration nodes. 1D Gauss-Hermite integration with more than 1250 nodes has not been calculated.

saw that this results in large biases for the estimated σ_α for all three types of integration. The approximation for Monte Carlo integration has the largest MAPE of the three, and as we saw lead to biased estimates for some of the other parameters as well. When the number of integration nodes is increased, the accuracy of the approximation quickly improves for Gauss-Hermite integration. The improvement is more gradual for Monte Carlo integration and Halton sequences. The MAPE when using Halton sequences is a factor ten smaller than the MAPE we estimate when using Monte Carlo integration.

When we look at xD integration, we see that the MAPE is smaller than 1% for all cases but one. In this model the unobservables in the discrete outcome equation are only autocorrelated if ρ is different from 0. If $\rho = 0$, α_i does not enter the latent variable related to the discrete outcome and the matrix Σ_{DD} is equal to the identity matrix. Similarly, the matrices Σ_{DY} and Σ_{DD} are zero. This means that from (3.2.4) we have that $\Sigma_{D|Y}$ is the identity matrix if $\rho = 0$. In that case, the unobservables related to the discrete outcome equation are independent, and the calculation of their joint probability reduces to the multiplication of univariate normal CDFs. These can be approximated very accurately, which results in the negligible size of the mean absolute percentage error that we see for xD integration when $\rho = 0$.

Similar to 1D integration, for xD integration the MAPE for sparse grid integration is smaller than the MAPE for Monte Carlo integration when $T = 6$, but the difference is not of the same magnitude as in the 1D integration case. The MAPE goes down when the number of integration nodes is increased. However, for SGI it looks like the MAPE goes down in steps. There is not much difference between the MAPE for SGI accuracy 2 and 3. The MAPE for accuracy 4 is smaller than the MAPE for accuracy 3, but then stays somewhat the same when the accuracy is increased to 5 or 6. This also holds if we look at root mean-squared error as a measure of approximation error. The plots with the results in Heiss (2010) do not show enough detail to see whether he finds the same results for the dynamic probit model, but on the scale of his plots the differences between these accuracy levels seem to be very small as well.

Halton sequences result in a smaller approximation error than Monte Carlo integration. This difference is especially clear for accuracies 4 and 5, where the MAPE is almost a factor

10 smaller than the MAPE for Monte Carlo integration. The performance of Halton sequences and sparse grid integration is of the same order, where in some cases one is preferred slightly above the other, and vice versa in other cases.

The shuffled Halton sequences do not seem to provide an improvement on regular Halton sequences for this dimension. We expect a benefit from using shuffled Halton sequences for larger dimensions. In tables 3.10 and 3.11 the results are shown for $T = 11$ and $T = 16$, but again we do not see an improvement in MAPE of the shuffled Halton sequences over regular Halton sequences.

The results for $T = 11$ and $T = 16$ in tables 3.10 and 3.11 show a different picture. Sparse grid integration is no longer preferred over Monte Carlo integration in all cases. For $T = 11$ their performance in terms of MAPE is similar, but for $T = 16$ xD integration with Monte Carlo nodes results in a more accurate approximation than xD integration with sparse grids, if the number of integration nodes is increased to 451 or more. In addition, we see that the use of Halton sequences clearly outperforms the other methods.

These results are different from the results described by Heiss (2010). He concludes that Halton sequences result in a better approximation than using Monte Carlo integration, which can also be seen in my results. However, in his case sparse grid integration dominates the other methods, which is only true in my experiments when a small number of gridpoints is used for the integration, and $T = 6$. One potential explanation for this difference is the fact that he uses a different model, with a different correlation structure between the unobservables. Heiss (2010) considers a panel probit model where the errors follow an AR(1) process. Because of the auto-regressive process for the errors, the correlation between the unobservables in two different periods decreases if the number of intermediate periods increases. This is different from my specification, where the correlation is the same between all periods. Also, it appears from his results that the difference between sparse grid integration and Halton sequences becomes smaller if the correlation between unobservables increases or the dimension of the integral increases. For instance, the experiment with the highest correlation between the unobservables, has a coefficient for the AR(1) process of 0.9 and uses $T = 10$. In that case, which is shown in figure 5(f) in Heiss (2010), sparse grid integration has a RMSE which is about 10% lower than using

Halton sequences.

Another way to explain the different conclusions could be in the way Halton sequences are used. For instance, it is not clear whether Heiss (2010) throws away the first set of Halton draws, the so called burn-in. Similarly, the paper does not specify whether the same Halton sequence is used for all individuals, or whether a different sequence is used for every individual, which is what I do. From experiments that I do not report here, the difference in MAPE between re-using the same Halton sequence and using different Halton sequences is a factor five for $T = 16$. Combined with the different correlation structure in Heiss (2010) this could explain the difference in conclusion.

3.5.2 ARMA model

The results in this section are based on the model from example 2 in section 3.4.2. The covariates that I use are the same as in (3.5.1) in the random effect model in the previous section, repeated here for convenience

$$\begin{aligned} Y_{it}^* &= \beta_{1,0} + \beta_{1,1} x_{1,it} + \qquad\qquad U_{1,it} \\ D_{it}^* &= \beta_{2,0} + \beta_{2,1} x_{1,it} + \beta_{2,2} x_{2,it} + U_{2,it}. \end{aligned} \tag{3.5.2}$$

The difference with the model in the previous section is the structure of the unobservables, $U_{1,it}$ and $U_{2,it}$. These follow the ARMA(1,1) process described in (3.4.2). The covariates are simulated similarly to the previous experiment, $x_{1,it} \sim$ i.i.d. $N(0,1)$, and $x_{2,it} \sim$ i.i.d. $N(0,1)$. For the simulations I use $(\beta_{1,0}, \beta_{1,1}) = (5,1)$ as coefficients in the continuous outcome equation. The constant in the discrete outcome equation, $\beta_{2,0}$ takes on three values, $\beta_{2,0} \in \{-1, 0, 2\}$, and the values of the other two coefficients are fixed to $(\beta_{2,1}, \beta_{2,2}) = (0,1)$. The different values of $\beta_{2,0}$ correspond to a proportion of 25%, 50% and 90% of observations respectively for which we observe the continuous outcome.

The standard deviation of the initial value of the auto-regressive part of the unobservable is $\sigma_{\xi_0} = 1$, the standard deviation of the innovations to the auto-regressive process is $\sigma_\varepsilon = 0.5$, and the auto-regressive parameter takes on four different values, $\phi \in \{0, 0.3, 0.6, 0.9\}$. The coefficients for the moving average process are set to $\theta_1 = 0.8$ and $\theta_2 = -0.4$. Finally, the parameter governing the correlation between the unobserv-

ables in the continuous and the discrete outcome equation, ρ, is the same for all variations. I choose an intermediate value, $\rho = 0.6$.

Similar to the previous section, I present two different experiments. In the first experiment, I estimate the parameters of this process on simulated datasets, while keeping the number of gridpoints used to approximate the integral the same. The second experiment keeps the value of the parameters the same, and compares the accuracy of the log-likelihood at that point for different accuracy levels.

For the first experiment, I create $R = 500$ simulated datasets with $N = 500$ individuals. The number of periods for the created panel datasets is set to $T = 6$, $T = 16$, and $T = 26$. I set the accuracy for the sparse grid integration to 2, which implies that 11, 31 and 51 integration nodes are used for $T = 6$, $T = 16$, and $T = 26$ respectively. In contrast to the previous section, only xD integration can be used, because the error structure of the unobservables can not be rewritten using a simple factor structure with one factor in this case.

Tables 3.12 to 3.16 show the bias in the level parameters, $\beta_{1,0}, \beta_{1,1}, \beta_{2,0}, \beta_{2,1}$ and $\beta_{2,2}$. The results for $T = 6$ are shown in the left four columns, and the results for $T = 26$ are shown on the right. Three blocks of rows correspond to the different values of $\beta_{2,0}$ that we used to generate the simulated datasets. Within each of these three blocks, four rows show the results for different ϕ. I compare four ways to create the integration nodes that are used to approximate the integral; sparse grid integration (SGI), Monte Carlo draws (MC), Halton sequences (Halton), and shuffled Halton sequences (Halton shuffled).

When we look at the results in table 3.12, we see no apparent bias in $\beta_{1,0}$ for any of the methods, except perhaps for SGI when $T = 26$, $\beta_{2,0} = -1$, and $\phi = 0.9$. We do see differences in the standard deviations. There is more variation in the parameter estimates, and thus in the bias, when $\beta_{2,0}$, the constant in the discrete outcome equation, is smaller. If $\beta_{2,0} = -1$, we observe a continuous outcome for only 25% of the observations, compared to 50% and 90% when $\beta_{2,0}$ is 0 or 2 respectively. This implies that there are fewer observations that contain information about $\beta_{1,0}$ if $\beta_{2,0}$ is smaller, hence the loss in precision. Similar reasoning explains the decrease in the standard deviation when comparing $T = 6$ to $T = 26$.

Table 3.12 – Bias in $\beta_{1,0}$ for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.004	0.003	0.002	0.002	–0.005	0.001	0.001	0.001
	(0.069)	(0.069)	(0.069)	(0.069)	(0.030)	(0.031)	(0.031)	(0.031)
0.3	–0.004	–0.001	–0.001	–0.001	–0.003	0.001	0.001	0.001
	(0.070)	(0.070)	(0.070)	(0.070)	(0.033)	(0.033)	(0.033)	(0.033)
0.6	0.008	0.007	0.006	0.006	0.003	0.001	0.001	0.001
	(0.079)	(0.079)	(0.079)	(0.079)	(0.036)	(0.036)	(0.036)	(0.036)
0.9	0.000	0.001	–0.003	–0.003	0.015	0.000	0.000	0.000
	(0.089)	(0.088)	(0.088)	(0.088)	(0.047)	(0.046)	(0.046)	(0.046)
$\beta_{2,0} = 0$								
0	–0.002	0.001	0.000	0.000	–0.002	0.000	0.001	0.001
	(0.033)	(0.033)	(0.033)	(0.033)	(0.015)	(0.015)	(0.015)	(0.015)
0.3	0.000	0.001	0.001	0.001	–0.002	0.000	0.000	0.000
	(0.036)	(0.036)	(0.036)	(0.036)	(0.017)	(0.017)	(0.017)	(0.017)
0.6	0.001	0.000	0.000	0.000	0.001	0.000	0.000	–0.001
	(0.046)	(0.046)	(0.046)	(0.046)	(0.021)	(0.020)	(0.020)	(0.020)
0.9	–0.003	–0.004	–0.005	–0.005	0.005	0.000	0.000	–0.001
	(0.056)	(0.056)	(0.056)	(0.056)	(0.037)	(0.037)	(0.037)	(0.037)
$\beta_{2,0} = 2$								
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.016)	(0.016)	(0.016)	(0.016)	(0.007)	(0.007)	(0.007)	(0.007)
0.3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.019)	(0.019)	(0.019)	(0.019)	(0.009)	(0.009)	(0.009)	(0.009)
0.6	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.024)	(0.024)	(0.024)	(0.024)	(0.013)	(0.013)	(0.013)	(0.013)
0.9	0.001	0.001	0.001	0.001	–0.001	–0.001	–0.001	–0.001
	(0.044)	(0.044)	(0.044)	(0.044)	(0.033)	(0.033)	(0.033)	(0.033)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.13 – Bias in $\beta_{1,1}$ for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.003	–0.003	–0.003	–0.003	0.000	0.000	0.000	0.000
	(0.036)	(0.036)	(0.036)	(0.036)	(0.017)	(0.017)	(0.017)	(0.017)
0.3	–0.004	–0.004	–0.004	–0.004	0.000	0.000	0.000	0.000
	(0.035)	(0.035)	(0.035)	(0.035)	(0.017)	(0.017)	(0.017)	(0.017)
0.6	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.038)	(0.038)	(0.038)	(0.038)	(0.018)	(0.018)	(0.018)	(0.018)
0.9	–0.001	–0.001	–0.001	–0.001	0.000	0.000	0.000	0.000
	(0.038)	(0.038)	(0.038)	(0.038)	(0.018)	(0.018)	(0.018)	(0.018)
$\beta_{2,0} = 0$								
0	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001	–0.001
	(0.025)	(0.025)	(0.025)	(0.025)	(0.012)	(0.012)	(0.012)	(0.012)
0.3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.025)	(0.025)	(0.025)	(0.025)	(0.012)	(0.012)	(0.012)	(0.012)
0.6	0.001	0.001	0.001	0.001	0.000	0.000	0.000	0.000
	(0.027)	(0.027)	(0.027)	(0.027)	(0.013)	(0.013)	(0.013)	(0.013)
0.9	0.002	0.002	0.002	0.002	0.000	0.000	0.000	0.000
	(0.028)	(0.028)	(0.028)	(0.028)	(0.014)	(0.014)	(0.014)	(0.014)
$\beta_{2,0} = 2$								
0	0.001	0.001	0.001	0.001	0.000	0.000	0.000	0.000
	(0.018)	(0.018)	(0.018)	(0.018)	(0.009)	(0.009)	(0.009)	(0.009)
0.3	0.001	0.001	0.001	0.001	0.000	0.000	0.000	0.000
	(0.021)	(0.021)	(0.021)	(0.021)	(0.009)	(0.009)	(0.009)	(0.009)
0.6	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.021)	(0.021)	(0.021)	(0.021)	(0.010)	(0.010)	(0.010)	(0.010)
0.9	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.021)	(0.021)	(0.021)	(0.021)	(0.010)	(0.010)	(0.010)	(0.010)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.14 – Bias in $\beta_{2,0}$ for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	0.004	–0.004	–0.004	–0.004	0.013	0.003	0.003	0.003
	(0.078)	(0.082)	(0.083)	(0.083)	(0.046)	(0.051)	(0.051)	(0.051)
0.3	0.015	0.009	0.009	0.009	0.016	0.009	0.008	0.008
	(0.061)	(0.061)	(0.062)	(0.061)	(0.045)	(0.044)	(0.043)	(0.043)
0.6	0.008	0.006	0.006	0.006	–0.002	0.000	0.000	0.000
	(0.054)	(0.054)	(0.053)	(0.053)	(0.029)	(0.029)	(0.029)	(0.029)
0.9	0.003	0.007	0.007	0.007	–0.022	–0.003	–0.003	–0.002
	(0.053)	(0.053)	(0.053)	(0.053)	(0.032)	(0.031)	(0.031)	(0.031)
$\beta_{2,0} = 0$								
0	0.002	0.001	0.001	0.001	0.000	–0.001	–0.001	–0.001
	(0.026)	(0.026)	(0.026)	(0.026)	(0.013)	(0.013)	(0.013)	(0.013)
0.3	0.001	0.000	0.000	0.000	0.000	–0.001	–0.001	–0.001
	(0.028)	(0.028)	(0.028)	(0.028)	(0.013)	(0.013)	(0.013)	(0.013)
0.6	–0.001	–0.001	–0.001	–0.001	–0.002	–0.001	–0.001	–0.001
	(0.030)	(0.030)	(0.030)	(0.030)	(0.015)	(0.015)	(0.015)	(0.015)
0.9	–0.007	–0.004	–0.003	–0.003	–0.011	–0.003	–0.003	–0.003
	(0.037)	(0.037)	(0.037)	(0.037)	(0.026)	(0.026)	(0.026)	(0.026)
$\beta_{2,0} = 2$								
0	–0.001	–0.001	–0.001	–0.001	–0.002	–0.002	–0.003	–0.003
	(0.097)	(0.097)	(0.097)	(0.097)	(0.073)	(0.072)	(0.073)	(0.073)
0.3	–0.003	–0.003	–0.003	–0.003	0.000	0.000	0.000	0.000
	(0.100)	(0.100)	(0.100)	(0.100)	(0.058)	(0.058)	(0.058)	(0.058)
0.6	0.005	0.005	0.006	0.006	0.003	0.003	0.003	0.003
	(0.084)	(0.084)	(0.084)	(0.084)	(0.040)	(0.040)	(0.040)	(0.040)
0.9	0.007	0.008	0.009	0.009	0.001	0.003	0.003	0.003
	(0.076)	(0.076)	(0.076)	(0.076)	(0.039)	(0.039)	(0.039)	(0.039)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.15 – Bias in $\beta_{2,1}$ for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.002	–0.002	–0.002	–0.002	0.000	0.000	0.000	0.000
	(0.030)	(0.030)	(0.030)	(0.030)	(0.014)	(0.014)	(0.014)	(0.014)
0.3	–0.001	–0.001	–0.001	–0.001	0.000	0.000	0.000	0.000
	(0.029)	(0.030)	(0.030)	(0.030)	(0.014)	(0.014)	(0.015)	(0.015)
0.6	–0.001	–0.001	–0.001	–0.001	0.000	0.000	0.000	0.000
	(0.032)	(0.032)	(0.032)	(0.032)	(0.015)	(0.015)	(0.015)	(0.015)
0.9	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	(0.031)	(0.031)	(0.031)	(0.031)	(0.016)	(0.016)	(0.016)	(0.016)
$\beta_{2,0} = 0$								
0	0.002	0.002	0.002	0.002	0.001	0.001	0.001	0.001
	(0.027)	(0.027)	(0.027)	(0.027)	(0.012)	(0.013)	(0.013)	(0.013)
0.3	–0.001	–0.001	–0.001	–0.001	0.000	0.000	0.000	0.000
	(0.025)	(0.025)	(0.025)	(0.025)	(0.013)	(0.013)	(0.013)	(0.013)
0.6	0.001	0.001	0.001	0.001	0.000	0.000	0.000	0.000
	(0.028)	(0.028)	(0.029)	(0.029)	(0.013)	(0.013)	(0.013)	(0.013)
0.9	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	(0.029)	(0.029)	(0.029)	(0.029)	(0.014)	(0.014)	(0.014)	(0.014)
$\beta_{2,0} = 2$								
0	0.000	0.000	0.000	0.000	–0.001	–0.001	–0.001	–0.001
	(0.040)	(0.040)	(0.040)	(0.040)	(0.020)	(0.020)	(0.020)	(0.020)
0.3	0.000	0.000	0.000	0.000	–0.001	–0.001	–0.001	–0.001
	(0.041)	(0.041)	(0.041)	(0.041)	(0.020)	(0.020)	(0.020)	(0.020)
0.6	0.000	0.000	0.000	0.000	0.001	0.001	0.001	0.001
	(0.041)	(0.041)	(0.041)	(0.041)	(0.020)	(0.020)	(0.020)	(0.020)
0.9	0.001	0.001	0.001	0.001	0.002	0.002	0.002	0.002
	(0.042)	(0.042)	(0.042)	(0.042)	(0.019)	(0.019)	(0.019)	(0.019)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.16 – Bias in $\beta_{2,2}$ for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.003	0.005	0.005	0.005	–0.014	–0.004	–0.004	–0.003
	(0.081)	(0.085)	(0.086)	(0.086)	(0.046)	(0.051)	(0.051)	(0.051)
0.3	–0.015	–0.008	–0.008	–0.008	–0.017	–0.010	–0.009	–0.009
	(0.063)	(0.063)	(0.064)	(0.064)	(0.045)	(0.044)	(0.043)	(0.043)
0.6	–0.006	–0.003	–0.004	–0.004	0.002	0.000	0.000	0.000
	(0.056)	(0.056)	(0.056)	(0.056)	(0.028)	(0.028)	(0.028)	(0.028)
0.9	–0.005	–0.006	–0.005	–0.005	0.003	0.001	0.000	0.000
	(0.050)	(0.050)	(0.050)	(0.050)	(0.021)	(0.021)	(0.021)	(0.021)
$\beta_{2,0} = 0$								
0	–0.008	–0.004	–0.005	–0.005	–0.009	–0.004	–0.003	–0.004
	(0.061)	(0.063)	(0.063)	(0.063)	(0.043)	(0.045)	(0.045)	(0.045)
0.3	–0.007	–0.004	–0.005	–0.005	–0.008	–0.004	–0.004	–0.004
	(0.055)	(0.056)	(0.055)	(0.055)	(0.039)	(0.037)	(0.037)	(0.037)
0.6	0.000	0.000	0.001	0.001	0.002	0.000	0.000	0.000
	(0.044)	(0.045)	(0.044)	(0.044)	(0.023)	(0.023)	(0.023)	(0.023)
0.9	–0.006	–0.006	–0.005	–0.005	–0.003	–0.002	–0.002	–0.002
	(0.040)	(0.040)	(0.040)	(0.040)	(0.017)	(0.017)	(0.017)	(0.017)
$\beta_{2,0} = 2$								
0	0.000	0.000	0.000	0.000	–0.001	–0.001	–0.001	–0.001
	(0.060)	(0.060)	(0.060)	(0.060)	(0.041)	(0.041)	(0.041)	(0.041)
0.3	0.001	0.002	0.002	0.002	0.001	0.001	0.001	0.001
	(0.066)	(0.066)	(0.066)	(0.066)	(0.036)	(0.035)	(0.035)	(0.035)
0.6	0.006	0.006	0.006	0.006	0.002	0.002	0.002	0.002
	(0.058)	(0.058)	(0.058)	(0.058)	(0.029)	(0.029)	(0.029)	(0.029)
0.9	0.005	0.005	0.005	0.005	0.001	0.002	0.002	0.002
	(0.055)	(0.055)	(0.055)	(0.055)	(0.026)	(0.026)	(0.026)	(0.026)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

For the other level parameters we see a similar pattern. There are no large biases and the standard deviations are smaller when $T = 26$ compared to $T = 6$. Increasing $\beta_{2,0}$ leads to a decrease in the standard deviation for $\beta_{1,1}$. A change in $\beta_{2,0}$ does not have any effect on the information we have about the discrete outcome equation, so the standard deviations for $\beta_{2,0}$, $\beta_{2,1}$, and $\beta_{2,2}$ do not show this pattern.

In addition, we see for $\beta_{2,0}$ and $\beta_{2,2}$ in tables 3.14 and 3.16 that the standard deviation for the estimated parameter is smaller when we use sparse grids and $\phi = 0$. This suggests that in that case, sparse grids result in a more accurate approximation of the log-likelihood in terms of variance than the other three methods.

Tables 3.17, 3.18, and 3.19, show the bias in the parameters related to the AR(1) process, σ_{ξ_0}, σ_ε, and ϕ. From these tables we do not see a lot of bias in the estimates for parameter ϕ, but the bias in σ_{ξ_0} and σ_ε is substantial for some combinations of the parameters. The bias is largest when ϕ is small, and the bias decreases if ϕ increases. The bias is also smaller if β_{20} is larger, i.e if there are more continuous outcomes observed. Also, the bias is smaller if $T = 26$, compared to $T = 26$. Finally, the bias is larger when sparse grids are used to approximate the integral.

The parameter σ_ε denotes the standard deviation of the innovations to the auto-regressive process. The difference between the persistence over time for the AR(1) process and the MA(1) process identifies this parameter and θ_1 and θ_2, the parameters in the moving average process. If the persistence of the AR(1) process is low, i.e. when ϕ is small, the difference between these processes is difficult to pick up from a short sample. This problem get worse when the panel is shorter, $T = 6$ versus $T = 26$, or when there are fewer continuous outcomes that add information, β_{20} is smaller.

To check whether the bias that we see is due to a bad approximation of the integral or due to the small sample that is available, I run an additional experiment. In this experiment I use $\beta_{20} = -1$ and $\phi = 0$ to generate data for 500 or 2000 individuals. The number of periods is $T = 6$. I estimate the parameters from these datasets using sparse grid integration and Halton sequences with three different levels of accuracy. The results are shown in table 3.20.

We see from this table that increasing the accuracy does not improve the bias. However,

Table 3.17 – Bias in σ_{ξ_0} for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.026	–0.015	–0.016	–0.015	–0.042	–0.026	–0.026	–0.025
	(0.160)	(0.165)	(0.166)	(0.166)	(0.138)	(0.143)	(0.142)	(0.142)
0.3	**–0.063**	**–0.051**	**–0.051**	**–0.051**	**–0.051**	–0.037	–0.035	–0.035
	(0.148)	(0.150)	(0.150)	(0.150)	(0.143)	(0.140)	(0.139)	(0.139)
0.6	–0.039	–0.034	–0.033	–0.033	0.008	–0.004	–0.004	–0.004
	(0.139)	(0.139)	(0.138)	(0.138)	(0.096)	(0.099)	(0.099)	(0.099)
0.9	–0.023	–0.026	–0.021	–0.022	–0.005	0.000	0.002	0.001
	(0.095)	(0.096)	(0.095)	(0.095)	(0.064)	(0.065)	(0.064)	(0.064)
$\beta_{2,0} = 0$								
0	–0.040	–0.035	–0.036	–0.036	–0.027	–0.018	–0.017	–0.018
	(0.136)	(0.138)	(0.137)	(0.137)	(0.123)	(0.125)	(0.125)	(0.124)
0.3	–0.040	–0.034	–0.035	–0.035	–0.026	–0.020	–0.019	–0.019
	(0.136)	(0.137)	(0.135)	(0.135)	(0.122)	(0.119)	(0.119)	(0.119)
0.6	–0.020	–0.020	–0.018	–0.018	0.007	0.001	0.001	0.001
	(0.111)	(0.111)	(0.110)	(0.110)	(0.075)	(0.076)	(0.076)	(0.076)
0.9	–0.021	–0.020	–0.018	–0.018	–0.005	0.002	0.003	0.003
	(0.073)	(0.073)	(0.073)	(0.073)	(0.051)	(0.051)	(0.051)	(0.051)
$\beta_{2,0} = 2$								
0	–0.019	–0.019	–0.019	–0.019	–0.012	–0.012	–0.012	–0.012
	(0.109)	(0.109)	(0.109)	(0.109)	(0.098)	(0.098)	(0.098)	(0.098)
0.3	–0.022	–0.022	–0.022	–0.022	–0.010	–0.010	–0.010	–0.010
	(0.111)	(0.111)	(0.111)	(0.111)	(0.092)	(0.091)	(0.091)	(0.091)
0.6	–0.013	–0.013	–0.013	–0.013	0.001	0.001	0.001	0.001
	(0.077)	(0.077)	(0.077)	(0.077)	(0.060)	(0.060)	(0.060)	(0.060)
0.9	–0.002	–0.002	–0.001	–0.001	0.000	0.000	0.000	0.000
	(0.054)	(0.054)	(0.054)	(0.054)	(0.042)	(0.042)	(0.042)	(0.042)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.18 – Bias in σ_ε for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	**–0.190**	**–0.163**	**–0.164**	**–0.161**	**–0.168**	**–0.124**	**–0.123**	**–0.121**
	(0.299)	(0.305)	(0.306)	(0.305)	(0.265)	(0.264)	(0.262)	(0.262)
0.3	**–0.213**	**–0.179**	**–0.182**	**–0.181**	**–0.166**	**–0.120**	**–0.113**	**–0.112**
	(0.285)	(0.285)	(0.286)	(0.285)	(0.256)	(0.236)	(0.231)	(0.230)
0.6	**–0.080**	**–0.074**	**–0.076**	**–0.076**	0.009	–0.008	–0.009	–0.009
	(0.205)	(0.203)	(0.202)	(0.202)	(0.089)	(0.089)	(0.089)	(0.089)
0.9	–0.042	–0.045	–0.047	–0.047	0.020	0.001	0.000	0.000
	(0.126)	(0.128)	(0.126)	(0.126)	(0.025)	(0.027)	(0.027)	(0.027)
$\beta_{2,0} = 0$								
0	**–0.152**	**–0.139**	**–0.140**	**–0.140**	**–0.130**	**–0.103**	**–0.101**	**–0.103**
	(0.279)	(0.281)	(0.280)	(0.280)	(0.254)	(0.250)	(0.249)	(0.249)
0.3	**–0.140**	**–0.124**	**–0.126**	**–0.126**	**–0.078**	**–0.057**	**–0.055**	**–0.055**
	(0.265)	(0.263)	(0.260)	(0.260)	(0.194)	(0.177)	(0.175)	(0.175)
0.6	–0.038	–0.038	–0.038	–0.038	0.007	–0.002	–0.003	–0.003
	(0.140)	(0.139)	(0.139)	(0.139)	(0.063)	(0.064)	(0.064)	(0.064)
0.9	–0.025	–0.026	–0.026	–0.026	0.004	0.001	0.000	0.001
	(0.092)	(0.092)	(0.092)	(0.092)	(0.020)	(0.020)	(0.020)	(0.020)
$\beta_{2,0} = 2$								
0	**–0.100**	**–0.100**	**–0.100**	**–0.100**	**–0.075**	**–0.074**	**–0.075**	**–0.075**
	(0.251)	(0.251)	(0.251)	(0.251)	(0.219)	(0.217)	(0.218)	(0.218)
0.3	**–0.099**	**–0.098**	**–0.098**	**–0.098**	–0.024	–0.024	–0.024	–0.024
	(0.224)	(0.223)	(0.224)	(0.224)	(0.118)	(0.117)	(0.117)	(0.117)
0.6	–0.022	–0.022	–0.022	–0.022	0.000	0.000	0.000	0.000
	(0.089)	(0.089)	(0.089)	(0.089)	(0.047)	(0.047)	(0.047)	(0.047)
0.9	–0.005	–0.005	–0.005	–0.005	0.000	0.000	0.000	0.000
	(0.062)	(0.062)	(0.062)	(0.062)	(0.015)	(0.015)	(0.015)	(0.015)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.19 – Bias in ϕ for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.009	–0.004	–0.005	–0.006	–0.005	0.002	0.000	0.000
	(0.211)	(0.217)	(0.217)	(0.217)	(0.210)	(0.221)	(0.220)	(0.220)
0.3	–0.025	–0.011	–0.011	–0.010	–0.009	0.014	0.015	0.016
	(0.192)	(0.191)	(0.190)	(0.189)	(0.176)	(0.169)	(0.169)	(0.167)
0.6	0.004	0.011	0.013	0.013	–0.025	0.004	0.005	0.005
	(0.107)	(0.106)	(0.107)	(0.107)	(0.058)	(0.059)	(0.059)	(0.059)
0.9	0.011	0.012	0.013	0.013	–0.011	–0.001	0.000	0.000
	(0.050)	(0.050)	(0.050)	(0.050)	(0.009)	(0.009)	(0.009)	(0.009)
$\beta_{2,0} = 0$								
0	0.009	0.011	0.011	0.011	0.002	0.005	0.004	0.004
	(0.139)	(0.143)	(0.143)	(0.143)	(0.136)	(0.140)	(0.140)	(0.140)
0.3	–0.001	0.003	0.006	0.006	0.007	0.016	0.016	0.016
	(0.120)	(0.123)	(0.119)	(0.119)	(0.098)	(0.094)	(0.094)	(0.094)
0.6	0.007	0.011	0.012	0.012	–0.011	0.002	0.002	0.003
	(0.067)	(0.067)	(0.067)	(0.067)	(0.040)	(0.041)	(0.041)	(0.041)
0.9	0.007	0.007	0.008	0.008	–0.002	0.000	0.000	0.000
	(0.039)	(0.038)	(0.038)	(0.038)	(0.008)	(0.007)	(0.007)	(0.007)
$\beta_{2,0} = 2$								
0	0.001	0.001	0.001	0.001	0.004	0.004	0.004	0.004
	(0.093)	(0.093)	(0.093)	(0.093)	(0.091)	(0.091)	(0.091)	(0.091)
0.3	0.008	0.008	0.008	0.008	0.011	0.012	0.012	0.012
	(0.069)	(0.069)	(0.069)	(0.069)	(0.057)	(0.057)	(0.057)	(0.057)
0.6	0.008	0.008	0.008	0.008	0.001	0.001	0.001	0.001
	(0.045)	(0.045)	(0.045)	(0.045)	(0.030)	(0.030)	(0.030)	(0.030)
0.9	0.001	0.001	0.001	0.001	0.000	0.000	0.000	0.000
	(0.027)	(0.027)	(0.027)	(0.027)	(0.006)	(0.006)	(0.006)	(0.006)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

when we increase the number of observations from 500 to 2000 individuals, the bias drops by about 50% for sparse grid integration and more than 50% for the Halton sequences. It is therefore likely that the bias that we see is not due to the accuracy of the approximation of the integral, but is instead due to the small number of observations that our synthetic sample has.

Table 3.20 – Bias in σ_ε for different values of N

	No. of	N = 500		N = 2000	
Accuracy	Points	SGI	Halton	SGI	Halton
2	11	**–0.190**	**–0.164**	**–0.100**	**–0.069**
		(0.299)	(0.306)	(0.226)	(0.218)
3	51	**–0.191**	**–0.161**	**–0.100**	**–0.072**
		(0.298)	(0.306)	(0.226)	(0.220)
4	151	**–0.167**	**–0.162**	**–0.080**	**–0.073**
		(0.305)	(0.306)	(0.224)	(0.222)

Mean bias is shown based on 500 replications of simulated datasets with 500 and 2000 individuals, standard deviation in parentheses. Dimension of integration is $T-1$. The number of periods is, $T = 6$. Parameters values used for simulation are $\beta_{20} = -1.0$ and $\phi = 0.0$. Values for the bias larger than 0.05 in absolute value are in bold.

In table 3.21 and 3.22 we see some bias in the parameter estimates for θ_1 and θ_2. For $T = 26$, $\beta_{20} = -1$ and $\beta_{20} = 0$ the bias is larger when ϕ is large and sparse grids are used for the approximation. This suggests that when the correlation between the unobservables is high, sparse grids result in a poorer approximation than the other methods. No large biases are found for ρ in table 3.23, except that sparse grid integration performs slightly worse than the other methods in some cases.

I run a second experiment to directly compare the differences in the approximation between the different methods, similar to the second experiment in the previous section, which considers the random effect model. Again, we simulate outcome data for 1000 individuals, and compare the log-likelihood that we get from different approximation methods with the 'true' log-likelihood. The 'true' log-likelihood is obtained using pseudo Monte Carlo integration with 1,000,000 draws. A subset of the results from the experiments is shown in tables 3.24, 3.25, and 3.26, for $T = 6$, $T = 16$, and $T = 26$ respectively. Results are shown for $\beta_{20} \in \{-1, 0, 2\}$, and $\phi \in \{0.3, 0.9\}$.

In all three tables we see that if we keep the number of integration points and the

Table 3.21 – Bias in θ_1 for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.009	–0.026	–0.026	–0.027	0.025	–0.005	–0.006	–0.008
	(0.206)	(0.216)	(0.218)	(0.218)	(0.148)	(0.164)	(0.165)	(0.166)
0.3	0.028	0.008	0.007	0.007	0.034	0.000	–0.004	–0.004
	(0.209)	(0.206)	(0.207)	(0.207)	(0.181)	(0.181)	(0.180)	(0.179)
0.6	–0.024	–0.030	–0.028	–0.028	**–0.055**	–0.038	–0.036	–0.036
	(0.190)	(0.189)	(0.188)	(0.188)	(0.136)	(0.134)	(0.133)	(0.133)
0.9	–0.030	–0.020	–0.019	–0.019	**–0.092**	–0.021	–0.018	–0.018
	(0.166)	(0.168)	(0.167)	(0.167)	(0.085)	(0.078)	(0.076)	(0.077)
$\beta_{2,0} = 0$								
0	–0.001	–0.013	–0.013	–0.012	0.012	–0.007	–0.009	–0.008
	(0.172)	(0.180)	(0.179)	(0.179)	(0.144)	(0.152)	(0.153)	(0.152)
0.3	–0.007	–0.021	–0.019	–0.019	–0.007	–0.024	–0.025	–0.025
	(0.195)	(0.197)	(0.193)	(0.193)	(0.168)	(0.163)	(0.163)	(0.163)
0.6	–0.035	–0.034	–0.034	–0.034	–0.042	–0.031	–0.030	–0.030
	(0.162)	(0.162)	(0.162)	(0.162)	(0.112)	(0.110)	(0.110)	(0.110)
0.9	–0.020	–0.016	–0.016	–0.016	–0.026	–0.010	–0.009	–0.009
	(0.139)	(0.138)	(0.139)	(0.139)	(0.054)	(0.050)	(0.049)	(0.049)
$\beta_{2,0} = 2$								
0	–0.015	–0.015	–0.015	–0.015	–0.010	–0.010	–0.010	–0.010
	(0.152)	(0.152)	(0.152)	(0.152)	(0.130)	(0.130)	(0.131)	(0.131)
0.3	–0.017	–0.018	–0.018	–0.018	–0.026	–0.027	–0.027	–0.027
	(0.173)	(0.173)	(0.173)	(0.173)	(0.136)	(0.135)	(0.135)	(0.135)
0.6	–0.024	–0.024	–0.024	–0.024	–0.018	–0.017	–0.017	–0.017
	(0.136)	(0.136)	(0.136)	(0.136)	(0.079)	(0.079)	(0.079)	(0.079)
0.9	–0.025	–0.025	–0.025	–0.025	–0.002	–0.002	–0.002	–0.002
	(0.109)	(0.110)	(0.109)	(0.109)	(0.029)	(0.029)	(0.029)	(0.029)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.22 – Bias in θ_2 for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	–0.004	0.010	0.013	0.012	–0.001	–0.002	–0.002	–0.003
	(0.157)	(0.165)	(0.168)	(0.169)	(0.085)	(0.100)	(0.102)	(0.102)
0.3	0.027	0.024	0.021	0.021	0.029	0.006	0.002	0.002
	(0.179)	(0.186)	(0.184)	(0.185)	(0.145)	(0.152)	(0.153)	(0.152)
0.6	–0.007	–0.007	–0.006	–0.006	–0.050	–0.032	–0.030	–0.030
	(0.203)	(0.202)	(0.202)	(0.202)	(0.135)	(0.133)	(0.133)	(0.133)
0.9	–0.023	–0.012	–0.012	–0.012	**–0.080**	–0.018	–0.015	–0.015
	(0.188)	(0.189)	(0.189)	(0.189)	(0.099)	(0.092)	(0.090)	(0.091)
$\beta_{2,0} = 0$								
0	–0.011	–0.011	–0.013	–0.012	–0.001	–0.008	–0.009	–0.008
	(0.103)	(0.108)	(0.110)	(0.110)	(0.078)	(0.084)	(0.086)	(0.085)
0.3	–0.007	–0.014	–0.016	–0.016	–0.007	–0.021	–0.022	–0.022
	(0.161)	(0.163)	(0.162)	(0.162)	(0.136)	(0.134)	(0.134)	(0.134)
0.6	–0.026	–0.025	–0.026	–0.026	–0.038	–0.028	–0.027	–0.027
	(0.162)	(0.162)	(0.162)	(0.162)	(0.111)	(0.109)	(0.109)	(0.109)
0.9	–0.013	–0.008	–0.009	–0.009	–0.019	–0.008	–0.007	–0.007
	(0.150)	(0.150)	(0.150)	(0.150)	(0.063)	(0.058)	(0.058)	(0.058)
$\beta_{2,0} = 2$								
0	–0.021	–0.021	–0.020	–0.020	–0.015	–0.015	–0.015	–0.015
	(0.094)	(0.094)	(0.094)	(0.094)	(0.078)	(0.078)	(0.079)	(0.079)
0.3	–0.021	–0.022	–0.022	–0.022	–0.026	–0.027	–0.027	–0.027
	(0.147)	(0.147)	(0.147)	(0.147)	(0.113)	(0.113)	(0.113)	(0.113)
0.6	–0.024	–0.024	–0.025	–0.025	–0.016	–0.016	–0.016	–0.016
	(0.138)	(0.138)	(0.138)	(0.138)	(0.079)	(0.079)	(0.079)	(0.079)
0.9	–0.022	–0.022	–0.022	–0.022	–0.001	–0.001	–0.001	–0.001
	(0.122)	(0.122)	(0.122)	(0.122)	(0.036)	(0.036)	(0.036)	(0.036)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

Table 3.23 – Bias in ρ for different values of $\beta_{2,0}$ and ϕ

	T = 6, xD integration				T = 26, xD integration			
ϕ	SGI	MC	Halton	Halton shuffled	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$								
0	0.000	–0.015	–0.015	–0.015	0.014	–0.001	–0.002	–0.002
	(0.111)	(0.118)	(0.120)	(0.120)	(0.053)	(0.061)	(0.061)	(0.061)
0.3	0.014	0.006	0.006	0.006	0.016	0.006	0.005	0.005
	(0.087)	(0.082)	(0.082)	(0.081)	(0.055)	(0.053)	(0.052)	(0.052)
0.6	0.003	0.002	0.003	0.003	–0.005	–0.001	–0.001	–0.001
	(0.061)	(0.060)	(0.060)	(0.060)	(0.031)	(0.031)	(0.031)	(0.031)
0.9	0.005	0.007	0.010	0.010	–0.029	–0.003	–0.002	–0.002
	(0.050)	(0.051)	(0.050)	(0.050)	(0.020)	(0.021)	(0.021)	(0.021)
$\beta_{2,0} = 0$								
0	0.006	–0.002	–0.001	–0.001	0.009	0.000	0.000	0.000
	(0.070)	(0.076)	(0.075)	(0.075)	(0.046)	(0.049)	(0.050)	(0.049)
0.3	0.009	0.003	0.005	0.005	0.008	0.002	0.002	0.002
	(0.065)	(0.069)	(0.063)	(0.063)	(0.041)	(0.040)	(0.040)	(0.040)
0.6	0.002	0.003	0.003	0.003	–0.006	–0.002	–0.001	–0.001
	(0.049)	(0.049)	(0.049)	(0.049)	(0.024)	(0.024)	(0.025)	(0.025)
0.9	0.002	0.005	0.007	0.007	–0.015	–0.002	–0.002	–0.002
	(0.038)	(0.038)	(0.038)	(0.038)	(0.015)	(0.015)	(0.015)	(0.015)
$\beta_{2,0} = 2$								
0	0.006	0.006	0.006	0.006	0.002	0.001	0.001	0.001
	(0.062)	(0.062)	(0.062)	(0.062)	(0.043)	(0.043)	(0.043)	(0.043)
0.3	0.010	0.009	0.010	0.010	0.001	0.001	0.001	0.001
	(0.061)	(0.061)	(0.061)	(0.061)	(0.038)	(0.038)	(0.038)	(0.038)
0.6	0.008	0.008	0.009	0.009	–0.002	–0.002	–0.002	–0.002
	(0.058)	(0.058)	(0.058)	(0.058)	(0.030)	(0.030)	(0.030)	(0.030)
0.9	0.000	0.001	0.001	0.001	–0.001	0.000	0.000	0.000
	(0.041)	(0.041)	(0.041)	(0.041)	(0.018)	(0.018)	(0.018)	(0.018)

Mean bias is shown based on 500 replications of simulated datasets with 500 individuals, standard deviation in parentheses. Dimension of integration is $T - 1$. SGI accuracy is 2, which implies that 11 and 51 integration nodes are used for $T = 6$, and $T = 26$ respectively. Values for the bias larger than 0.05 in absolute value are in bold.

approximation method fixed, the approximation has the least error when $\beta_{20} = 2$. The difference between the approximation errors is smaller between $\beta_{20} = 0$ and $\beta_{20} = -1$, but the errors are somewhat smaller when we observe more continuous outcomes, i.e. when $\beta_{20} = 0$. Also, a higher correlation between the unobservables, $\phi = 0.9$ compared to $\phi = 0.3$, corresponds to an increase in the approximation error.

When looking at the effect of an increase in accuracy we see again that the approximation error for sparse grid integration decreases in steps. This is especially clear when $T = 6$ or $T = 16$. For accuracy 2 and 3, the approximation error is virtually the same. Then the approximation error improves by almost a factor ten, when we go from accuracy 3 to 4. Increasing the accuracy further, to 5 and 6, does hardly show any improvement over accuracy 4.

If we compare between approximation methods, we see that sparse grid integration has the lowest approximation error for $T = 6$ and $T = 16$ when a small number of nodes is used, accuracy is 2. In some cases this method has a factor 10 lower approximation error. However, when we increase the number of integration points, Halton sequences often perform better, especially in the more difficult cases, e.g when $\beta_{20} = -1$ or $\phi = 0.9$. Again, shuffled Halton sequences do not show an improvement over regular Halton sequences.

3.6 Conclusion

In this paper I formulated a model with multiple continuous and discrete outcomes and I gave some examples of the dependence structure that could generate the underlying latent variables. If the unobservable elements in the model are random draws from the normal distribution, the multi-dimensional integral entering the log-likelihood, can be approximated using the GHK simulator.

Instead of using pseudo-random draws from the uniform distribution to use in combination with the GHK simulator, I follow Heiss (2010) and use sparse grids and Halton sequences. In two models with different error structures I compare the performance of these methods to pseudo-random draws. Similar to Heiss (2010), I compare the approximations of the log-likelihood that we obtain using different numbers of integration nodes to the 'true' log-likelihood at a fixed value for the parameters. My conclusion is somewhat

Table 3.24 – MAPE of log-likelihood approximation, $T = 6$

Accuracy	No. of points	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$, $\phi = 0.3$					
2	11	3.93×10^{-2}	3.45×10^{-1}	1.90×10^{-1}	1.88×10^{-1}
3	51	3.96×10^{-2}	1.60×10^{-1}	5.82×10^{-2}	5.87×10^{-2}
4	151	5.92×10^{-3}	9.25×10^{-2}	2.15×10^{-2}	2.19×10^{-2}
5	391	6.06×10^{-3}	5.94×10^{-2}	9.55×10^{-3}	9.84×10^{-3}
6	903	6.06×10^{-3}	3.90×10^{-2}	5.32×10^{-3}	5.42×10^{-3}
$\beta_{2,0} = -1$, $\phi = 0.9$					
2	11	2.18×10^{-1}	$1.90 \times 10^{+0}$	8.65×10^{-1}	8.54×10^{-1}
3	51	2.54×10^{-1}	8.33×10^{-1}	2.05×10^{-1}	2.16×10^{-1}
4	151	2.43×10^{-2}	4.88×10^{-1}	7.90×10^{-2}	9.36×10^{-2}
5	391	3.53×10^{-2}	3.16×10^{-1}	3.49×10^{-2}	4.54×10^{-2}
6	903	3.48×10^{-2}	2.11×10^{-1}	1.75×10^{-2}	2.52×10^{-2}
$\beta_{2,0} = 0$, $\phi = 0.3$					
2	11	1.56×10^{-2}	1.38×10^{-1}	6.46×10^{-2}	6.44×10^{-2}
3	51	1.58×10^{-2}	6.50×10^{-2}	2.35×10^{-2}	2.38×10^{-2}
4	151	2.09×10^{-3}	3.72×10^{-2}	7.97×10^{-3}	7.94×10^{-3}
5	391	2.16×10^{-3}	2.28×10^{-2}	3.52×10^{-3}	3.58×10^{-3}
6	903	2.15×10^{-3}	1.56×10^{-2}	1.96×10^{-3}	2.05×10^{-3}
$\beta_{2,0} = 0$, $\phi = 0.9$					
2	11	1.21×10^{-1}	6.70×10^{-1}	2.86×10^{-1}	2.91×10^{-1}
3	51	1.23×10^{-1}	2.60×10^{-1}	7.73×10^{-2}	8.00×10^{-2}
4	151	1.64×10^{-2}	1.70×10^{-1}	2.92×10^{-2}	3.10×10^{-2}
5	391	1.68×10^{-2}	1.03×10^{-1}	1.37×10^{-2}	1.66×10^{-2}
6	903	1.68×10^{-2}	7.01×10^{-2}	6.18×10^{-3}	9.57×10^{-3}
$\beta_{2,0} = 2$, $\phi = 0.3$					
2	11	5.56×10^{-4}	4.03×10^{-3}	2.25×10^{-3}	2.29×10^{-3}
3	51	5.56×10^{-4}	2.17×10^{-3}	6.71×10^{-4}	6.77×10^{-4}
4	151	6.71×10^{-5}	1.33×10^{-3}	2.65×10^{-4}	2.72×10^{-4}
5	391	6.71×10^{-5}	6.85×10^{-4}	1.03×10^{-4}	1.06×10^{-4}
6	903	6.71×10^{-5}	4.16×10^{-4}	8.27×10^{-5}	9.22×10^{-5}
$\beta_{2,0} = 2$, $\phi = 0.9$					
2	11	4.92×10^{-3}	2.68×10^{-2}	1.39×10^{-2}	1.41×10^{-2}
3	51	4.78×10^{-3}	9.98×10^{-3}	3.29×10^{-3}	3.35×10^{-3}
4	151	7.24×10^{-4}	6.76×10^{-3}	1.42×10^{-3}	1.44×10^{-3}
5	391	6.80×10^{-4}	4.12×10^{-3}	5.19×10^{-4}	5.39×10^{-4}
6	903	6.80×10^{-4}	3.29×10^{-3}	3.22×10^{-4}	3.80×10^{-4}

The 'true' log-likelihood used to calculate the mean absolute percentage error (MAPE), is based on xD Monte Carlo integration with 1,000,000 integration nodes.

Table 3.25 – MAPE of log-likelihood approximation, $T = 16$

Accuracy	No. of points	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$, $\phi = 0.3$					
2	31	3.68×10^{-2}	1.17×10^{-1}	1.01×10^{-1}	1.01×10^{-1}
3	451	3.79×10^{-2}	2.97×10^{-2}	1.43×10^{-2}	1.41×10^{-2}
4	4151	5.13×10^{-3}	1.02×10^{-2}	1.71×10^{-3}	1.75×10^{-3}
5	27671	5.61×10^{-3}	3.99×10^{-3}	7.36×10^{-4}	7.90×10^{-4}
6	145607	5.61×10^{-3}	1.79×10^{-3}	6.59×10^{-4}	6.70×10^{-4}
$\beta_{2,0} = -1$, $\phi = 0.9$					
2	31	3.56×10^{-1}	6.87×10^{-1}	4.43×10^{-1}	4.48×10^{-1}
3	451	3.44×10^{-1}	1.87×10^{-1}	5.79×10^{-2}	7.67×10^{-2}
4	4151	6.06×10^{-2}	6.18×10^{-2}	8.22×10^{-3}	1.91×10^{-2}
5	27671	4.88×10^{-2}	2.30×10^{-2}	3.98×10^{-3}	7.43×10^{-3}
6	145607	5.16×10^{-2}	1.11×10^{-2}	3.94×10^{-3}	4.99×10^{-3}
$\beta_{2,0} = 0$, $\phi = 0.3$					
2	31	1.77×10^{-2}	6.17×10^{-2}	5.55×10^{-2}	5.56×10^{-2}
3	451	1.84×10^{-2}	1.64×10^{-2}	7.26×10^{-3}	7.11×10^{-3}
4	4151	2.20×10^{-3}	5.38×10^{-3}	8.78×10^{-4}	9.34×10^{-4}
5	27671	2.46×10^{-3}	2.12×10^{-3}	3.81×10^{-4}	3.97×10^{-4}
6	145607	2.46×10^{-3}	9.73×10^{-4}	3.42×10^{-4}	3.45×10^{-4}
$\beta_{2,0} = 0$, $\phi = 0.9$					
2	31	1.75×10^{-1}	2.35×10^{-1}	1.67×10^{-1}	1.73×10^{-1}
3	451	1.36×10^{-1}	6.41×10^{-2}	2.25×10^{-2}	2.79×10^{-2}
4	4151	3.42×10^{-2}	2.18×10^{-2}	3.07×10^{-3}	6.24×10^{-3}
5	27671	2.09×10^{-2}	8.25×10^{-3}	1.45×10^{-3}	2.50×10^{-3}
6	145607	2.06×10^{-2}	3.75×10^{-3}	1.31×10^{-3}	1.71×10^{-3}
$\beta_{2,0} = 2$, $\phi = 0.3$					
2	31	7.47×10^{-4}	4.08×10^{-3}	2.57×10^{-3}	2.56×10^{-3}
3	451	7.50×10^{-4}	9.90×10^{-4}	3.47×10^{-4}	3.46×10^{-4}
4	4151	9.16×10^{-5}	3.29×10^{-4}	3.93×10^{-5}	3.97×10^{-5}
5	27671	9.24×10^{-5}	1.33×10^{-4}	2.20×10^{-5}	2.21×10^{-5}
6	145607	9.24×10^{-5}	6.21×10^{-5}	2.00×10^{-5}	1.99×10^{-5}
$\beta_{2,0} = 2$, $\phi = 0.9$					
2	31	7.78×10^{-3}	1.73×10^{-2}	1.21×10^{-2}	1.19×10^{-2}
3	451	7.03×10^{-3}	4.69×10^{-3}	1.52×10^{-3}	1.63×10^{-3}
4	4151	1.26×10^{-3}	1.44×10^{-3}	2.26×10^{-4}	2.71×10^{-4}
5	27671	1.02×10^{-3}	5.77×10^{-4}	1.09×10^{-4}	1.30×10^{-4}
6	145607	1.01×10^{-3}	2.68×10^{-4}	9.55×10^{-5}	9.86×10^{-5}

The 'true' log-likelihood used to calculate the mean absolute percentage error (MAPE), is based on xD Monte Carlo integration with 1,000,000 integration nodes.

Table 3.26 – MAPE of log-likelihood approximation, $T = 26$

Accuracy	No. of points	SGI	MC	Halton	Halton shuffled
$\beta_{2,0} = -1$, $\phi = 0.3$					
2	51	3.55×10^{-2}	7.18×10^{-2}	9.54×10^{-2}	9.61×10^{-2}
3	1251	3.74×10^{-2}	1.36×10^{-2}	6.93×10^{-3}	6.78×10^{-3}
4	19751	4.65×10^{-3}	3.51×10^{-3}	6.94×10^{-4}	7.47×10^{-4}
5	226951	5.49×10^{-3}	1.19×10^{-3}	5.00×10^{-4}	5.07×10^{-4}
$\beta_{2,0} = -1$, $\phi = 0.9$					
2	51	4.66×10^{-1}	4.05×10^{-1}	4.13×10^{-1}	3.94×10^{-1}
3	1251	3.39×10^{-1}	8.03×10^{-2}	3.96×10^{-2}	4.56×10^{-2}
4	19751	1.01×10^{-1}	2.09×10^{-2}	4.69×10^{-3}	9.41×10^{-3}
5	226951	5.38×10^{-2}	6.85×10^{-3}	3.08×10^{-3}	3.94×10^{-3}
$\beta_{2,0} = 0$, $\phi = 0.3$					
2	51	1.72×10^{-2}	3.93×10^{-2}	5.29×10^{-2}	5.27×10^{-2}
3	1251	1.83×10^{-2}	8.29×10^{-3}	3.68×10^{-3}	3.68×10^{-3}
4	19751	2.01×10^{-3}	2.04×10^{-3}	3.72×10^{-4}	4.04×10^{-4}
5	226951	2.44×10^{-3}	6.62×10^{-4}	2.87×10^{-4}	2.90×10^{-4}
$\beta_{2,0} = 0$, $\phi = 0.9$					
2	51	1.95×10^{-1}	1.59×10^{-1}	1.71×10^{-1}	1.64×10^{-1}
3	1251	1.37×10^{-1}	3.20×10^{-2}	1.58×10^{-2}	1.67×10^{-2}
4	19751	4.27×10^{-2}	8.24×10^{-3}	1.79×10^{-3}	2.86×10^{-3}
5	226951	2.32×10^{-2}	2.64×10^{-3}	1.17×10^{-3}	1.41×10^{-3}
$\beta_{2,0} = 2$, $\phi = 0.3$					
2	51	8.15×10^{-4}	3.13×10^{-3}	3.56×10^{-3}	3.55×10^{-3}
3	1251	8.18×10^{-4}	6.39×10^{-4}	2.57×10^{-4}	2.53×10^{-4}
4	19751	1.00×10^{-4}	1.80×10^{-4}	3.24×10^{-5}	3.16×10^{-5}
5	226951	1.01×10^{-4}	5.36×10^{-5}	2.29×10^{-5}	2.28×10^{-5}
$\beta_{2,0} = 2$, $\phi = 0.9$					
2	51	6.48×10^{-3}	1.11×10^{-2}	1.14×10^{-2}	1.15×10^{-2}
3	1251	5.86×10^{-3}	2.23×10^{-3}	1.01×10^{-3}	1.04×10^{-3}
4	19751	1.04×10^{-3}	5.84×10^{-4}	1.09×10^{-4}	1.23×10^{-4}
5	226951	8.44×10^{-4}	1.77×10^{-4}	8.02×10^{-5}	7.88×10^{-5}

The 'true' log-likelihood used to calculate the mean absolute percentage error (MAPE), is based on xD Monte Carlo integration with 1,000,000 integration nodes.

different from Heiss (2010). He sees strong benefits of using sparse grid integration in all cases, where my experiments show that sparse grid integration works better than the other methods if only a few number of nodes are used. When more nodes are used or when the correlation between the unobservables is high, Halton sequences are preferred.

In a separate experiment, I find the parameters that maximize the log-likelihood for different sets of simulated data. In those experiments I keep the number of integration nodes that I use to approximate the integral the same. These experiments show that for the random effects model 1D approximation of the integrals does not work well if a small number of nodes is used. This results in biased estimates for the variance parameters, even when Gauss-Hermite quadrature is used. With the same number of integration nodes, xD integration does not result in biased estimates. When we increase the number of nodes, the approximation error for 1D integration decreases much more rapidly than the approximation error for xD integration. Care has to be taken in choosing the number of nodes when using 1D integration to approximate the log-likelihood in a random effects model. If a higher-dimensional integral has to be approximated, because data from more periods is available, a higher number of nodes is required to ensure a high enough accuracy for the approximation.

For the xD integration methods we do not find substantial differences in the bias of the parameters between the different approximation methods. Even though Halton sequences and sparse grid integration provide more accurate approximations to the log-likelihood than pseudo Monte Carlo integration with the same number of nodes, on average only very small differences can be found in the actual values for the parameters that maximize the likelihood.

There are two possible explanations for this. The mean average percentage error is for almost all approximation methods smaller than one percent. The differences in approximation error could be too small to have a noticeable effect on the set of parameters that maximize the log-likelihood. As a second reason, this could be the result of how the synthetic datasets are constructed. The observable variables, $x_{1,it}$ and $x_{2,it}$, are generated from the normal distribution. Perhaps, when summing over the log-likelihood contributions of every individual, we are averaging out approximation errors. This averaging out

would potentially not happen when the covariates that enter the model follow a different distribution, which is almost always the case in practice. In practice, it is a good idea to run a similar experiment assessing the performance of the different approximation methods using the same data that will be used for the actual estimation. This will give a good sense for which method works better for that particular dataset and model.

Bibliography

Abowd, J. M., & Card, D. (1989, March). On the Covariance Structure of Earnings and Hours Changes. *Econometrica*, *57*(2), 411–445.

Altonji, J. G., Smith, A., & Vidangos, I. (2009, February). *Modeling Earnings Dynamics* (NBER Working Papers No. 14743). National Bureau of Economic Research, Inc.

Amemiya, T. (1984). Tobit models: A survey. *Journal of Econometrics*, *24*(1-2), 3–61.

An, M. Y., & Liu, M. (2000, November). Using indirect inference to solve the initial-conditions problem. *The Review of Economics and Statistics*, *82*(4), 656–667.

Baker, M., & Solon, G. (2003, April). Earnings Dynamics and Inequality among Canadian Men, 1976–1992: Evidence from Longitudinal Income Tax Records. *Journal of Labor Economics*, *21*(2), 289–321.

Bhat, C. R. (2003, November). Simulation estimation of mixed discrete choice models using randomized and scrambled Halton sequences. *Transportation Research Part B: Methodological*, *37*(9), 837–855.

Blundell, R., & Etheridge, B. (2010, January). Consumption, income and earnings inequality in Britain. *Review of Economic Dynamics*, *13*(1), 76–102.

Blundell, R., Pistaferri, L., & Preston, I. (2008, December). Consumption Inequality and Partial Insurance. *American Economic Review*, *98*(5), 1887–1921.

Blundell, R., & Preston, I. (1998, May). Consumption Inequality and Income Uncertainty. *Quarterly Journal of Economics*, *113*(2), 603–640.

Blundell, R., Reed, H., & Stoker, T. M. (2003, November). Iterpreting Aggregate Wage Growth: The Role of Labor Market Participation. *The American Economic Review*, *93*(4), 1114–1131.

Börsch-Supan, A., & Hajivassiliou, V. A. (1993). Smooth unbiased multivariate probability

simulators for maximum likelihood estimation of limited dependent variable models. *Journal of Econometrics*, *58*(3), 347–368.

Browning, M., Ejrnæs, M., & Alvarez, J. (2010). Modelling Income Processes with Lots of Heterogeneity. *Review of Economic Studies*, *77*(4), 1353–1381.

Butler, J. S., & Moffitt, R. (1982, May). A Computationally Efficient Quadrature Procedure for the One-Factor Multinomial Probit Model. *Econometrica*, *50*(3), 761–764.

Byrd, R. H., Nocedal, J., & Waltz, R. A. (2006). KNITRO: An integrated package for nonlinear optimization. In *Large scale nonlinear optimization, 35-59, 2006* (pp. 35–59). Springer Verlag.

Dickens, R. (2000, January). The Evolution of Individual Male Earnings in Great Britain: 1975–95. *The Economic Journal*, *110*(460), 27–49.

Dubé, J.-P. H., Fox, J. T., & Su, C.-L. (2009). *Improving the numerical performance of BLP static and dynamic discrete choice random coefficients demand estimation.* (May 2009. NBER Working Paper No. w14991. Available at SSRN: http://ssrn.com/abstract=1408911)

Duffee, G. R., & Stanton, R. H. (2008). Evidence on simulation inference for near unit-root processes with implications for term structure estimation. *Journal of Financial Econometrics*, *6*(1), 108–142.

ESRC Research Centre on Micro-social Change. (1991–2006). British Household Panel Survey, Data files and associated documentation. Colchester: The Data Archive.

Fuleky, P., & Zivot, E. (2010a). *Further evidence on simulation inference for near unit-root processes with implications for term structure estimation.* (Working paper)

Fuleky, P., & Zivot, E. (2010b). *Indirect inference based on the score.* (Working paper)

Gallant, A. R., & Tauchen, G. (1996, October). Which moments to match? *Econometric Theory*, *12*(04), 657–681.

Genz, A. (1992, June). Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, *1*(2), 141–149.

Genz, A. (2004). Numerical computation of rectangular bivariate and trivariate normal and t probabilities. *Statistics and Computing*, *14*(3), 251–260.

Geweke, J. (1996). Chapter 15 Monte carlo simulation and numerical integration. In

H. M. Amman, D. A. Kendrick, & J. Rust (Eds.), *Handbook of computational economics* (Vol. 1, pp. 731–800). Elsevier.

Gill, P. E., Murray, W., & Saunders, M. A. (2002). SNOPT: An SQP Algorithm For Large-Scale Constrained Optimization. *SIAM Journal on Optimization*, *12*(4), 979–1006.

Gottschalk, P., & Moffitt, R. A. (1994). The Growth of Earnings Instability in the U.S. Labor Market. *Brookings Papers on Economic Activity*, *25*(2), 217–272.

Gouriéroux, C. S., Monfort, A., & Renault, E. (1993). Indirect inference. *Journal of Applied Econometrics*, *8*(S1), S85–S118.

Gouriéroux, C. S., Phillips, P. C. B., & Yu, J. (2010, July). Indirect inference for dynamic panel models. *Journal of Econometrics*, *157*(1), 68–77.

Guvenen, F. (2007, June). Learning Your Earning: Are Labor Income Shocks Really Very Persistent? *American Economic Review*, *97*(3), 687–712.

Guvenen, F. (2009, January). An Empirical Investigation of Labor Income Processes. *Review of Economic Dynamics*, *12*(1), 58–79.

Haider, S. J. (2001). Earnings Instability and Earnings Inequality of Males in the United States: 1967–1991. *Journal of Labor Economics*, *19*(4), 799–836.

Hajivassiliou, V., McFadden, D., & Ruud, P. (1996). Simulation of multivariate normal rectangle probabilities and their derivatives: theoretical and computational results. *Journal of econometrics*, *72*(1–2), 85–134.

Heckman, J. J. (1979, January). Sample selection bias as a specification error. *Econometrica*, *47*(1), 153–161.

Heckman, J. J. (1981). The Incidental Parameters Problem and the Problem of Initial Conditions in Estimating a Discrete Time-Discrete Data Stochastic Process. In C. F. Manski & D. McFadden (Eds.), *Structural Analysis of Discrete Data.* Cambridge, MA: MIT Press.

Heiss, F. (2010). The panel probit model: adaptive integration on sparse grids. In W. Greene & R. Carter Hill (Eds.), *Advances in Econometrics, Vol 26: Maximum simulated likelihood methods and applications* (pp. 41–64). Emerald Group Publishing Limited.

Heiss, F., & Winschel, V. (2008, May). Likelihood approximation by numerical integration on sparse grids. *Journal of Econometrics*, *144*(1), 62–80.

Hess, S., Polak, J. W., & Daly, A. (2003). On the performance of the shuffled halton sequence in the estimation of discrete choice models. Presented at the *European Transport Conference 2003*, Strasbourg.

Hryshko, D. (2012). Labor income profiles are not heterogeneous: Evidence from income growth rates. *Quantitative Economics*, *3*(2), 177–209.

Hyslop, D. R. (1999, November). State Dependence, Serial Correlation and Heterogeneity in Intertemporal Labor Force Participation of Married Women. *Econometrica*, *67*(6), 1255–1294.

Jenkins, S. P., & Lambert, P. J. (2011, September). Robert Moffitt and Peter Gottschalk's 1995 paper 'Trends in the covariance structure of earnings in the U.S.: 1969–1987'. *Journal of Economic Inequality*, *9*(3), 433–437.

Jiang, W., & Turnbull, B. (2004). The indirect method: Inference based on intermediate statistics: A synthesis and examples. *Statistical Science*, *19*(2), 239–263.

Judd, K. L. (1998). *Numerical methods in economics* (Vol. 1) (No. 0262100711). The MIT Press.

Keane, M. P. (1994). A Computationally Practical Simulation Estimator for Panel Data. *Econometrica*, *62*(1), 95–116.

Keane, M. P., & Smith, A. A. (2004, August). *Generalized Indirect Inference for Discrete Choice Models* (Econometric Society 2004 North American Winter Meetings No. 512). Econometric Society.

Lillard, L. A., & Willis, R. J. (1978, September). Dynamic Aspects of Earning Mobility. *Econometrica*, *46*(5), 985–1012.

Low, H., Meghir, C., & Pistaferri, L. (2010, September). Wage Risk and Employment Risk over the Life Cycle. *American Economic Review*, *100*(4), 1432–67.

MaCurdy, T. E. (1982, January). The use of time series processes to model the error structure of earnings in a longitudinal data analysis. *Journal of Econometrics*, *18*(1), 83–114.

Magnac, T., Robin, J.-M., & Visser, M. (1995). Analysing incomplete individual employ-

ment histories using indirect inference. *Journal of Applied Econometrics*, *10*(S1), S153–S169.

Meghir, C., & Pistaferri, L. (2004, January). Income Variance Dynamics and Heterogeneity. *Econometrica*, *72*(1), 1–32.

Moffitt, R. A., & Gottschalk, P. (1995). *Trends in the covariance structure of earnings in the U.S.: 1969–1987.* (Brown University Working Paper)

Moffitt, R. A., & Gottschalk, P. (2011, September). Trends in the covariance structure of earnings in the U.S.: 1969–1987. *Journal of Economic Inequality*, *9*(3), 439–459.

Nickell, S. (1981, November). Biases in dynamic models with fixed effects. *Econometrica*, *49*(6), 1417–1426.

Ramos, X. (2003, May). The Covariance Structure of Earnings in Great Britain, 1991–1999. *Economica*, *70*(278), 353–374.

Smith, A. A., Jr. (1993). Estimating nonlinear time-series models using simulated vector autoregressions. *Journal of Applied Econometrics*, *8*(S1), S63–S84.

Smith, A. A., Jr. (2008). Indirect inference. In S. N. Durlauf & L. E. Blume (Eds.), *The new palgrave dictionary of economics.* Basingstoke: Palgrave Macmillan.

Stewart, M. B. (2007). The interrelated dynamics of unemployment and low-wage employment. *Journal of Applied Econometrics*, *22*(3), 511–531.

Su, C.-L., & Judd, K. L. (2010). *Constrained optimization approaches to estimation of structural models.* (Working paper)

Train, K. (2003). *Discrete Choice Methods With Simulation.* Cambridge University Press.

Wächter, A., & Biegler, L. T. (2006). On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, *106*(1), 25–57.

Yen, S. T. (2005, May). A Multivariate Sample-Selection Model: Estimating Cigarette and Alcohol Demands with Zero Observations. *American Journal of Agricultural Economics*, *87*(2), 453–466.