Estimating Discrete-Choice Games of Incomplete Information: A Simple Static Example*

Che-Lin Su†
The University of Chicago
Booth School of Business

March 22, 2012

Abstract

I investigate the problem of estimating discrete-choice games under incomplete information. In these games, multiple equilibria can exist. Also, different values of structural parameters can result in different numbers of equilibria. Consequently, under maximum-likelihood estimation, the likelihood function is a discontinuous function of the structural parameters. I reformulate the maximum-likelihood estimation problem as a constrained optimization problem in the joint space of structural parameters and economic endogenous variables. Under this formulation, the objective function and structural equations are smooth functions. The constrained optimization approach does not require repeatedly solving the game or finding all the equilibria. I use a simple, static-game example to demonstrate this approach, conducting Monte Carlo experiments to evaluate the finite-sample performance of the maximum-likelihood estimator, two-step estimators, and the nested pseudo-likelihood estimator.

Keywords: structural estimation, discrete-choice games of incomplete information, constrained optimization, multiple equilibria

*I have benefited greatly from discussions with Panle Jia Barwick, Kenneth L. Judd, Harry J. Paarsch, Ariel Pakes, Peter E. Rossi, and John Rust, and comments from seminar participants at the 2010 QME Conference, Boston College, and Harvard University. I am very grateful to John Rust for suggesting the static-game example used in this paper. I acknowledge the financial support from The University of Chicago Booth School of Business. All errors are mine.

†Email: Che-Lin.Su@ChicagoBooth.edu

1 Introduction

During the past decade, estimating empirical models of games of incomplete information has become an important and active research area in industrial organization and applied econometrics. An important feature of empirical models of these games is that multiple equilibria can exist. This multiplicity leads to the issue of which equilibria are being played in the data. To date, with the exception of moment inequality estimators (see, for example, Ciliberto and Tamer [2009], and Pakes, Porter, Ho, and Ishii [2011]), it is commonly assumed that only one equilibrium is played in each market in the data. Even under this assumption, however, it is still conceivable that the number of equilibria can change for different values of structural parameters, which raises another potential issue for maximum-likelihood estimation of games: the likelihood function, which is defined as a function of structural parameters, can be discontinuous. When applying the nested fixed-point algorithm, researchers are confronted with two computational tasks: first, for each candidate vector of parameters considered, all Nash equilibria must be found in order to evaluate the corresponding likelihood value; second, the objective (likelihood) function is potentially discontinuous. For the first task, no computational methods can guarantee finding all the equilibria of a game, unless the equilibrium equations form a system of polynomial equations; see Judd, Renner, and Schmedders (2011). Without finding all the equilibria, the likelihood function can be mis-specified. For the second task, no reliable algorithm exists to maximize a discontinuous function. Although some heuristic approaches (for example, grid search or genetic algorithms) can be applied to find an approximated solution, in practice these methods tend to be very slow and, typically, do not find good approximations.

Because of the potential computational costs that make implementing the nested fixed-point algorithm impractical, researchers have proposed two-step estimators for estimating games; see Bajari, Benkard, and Levin (2007); Pakes, Ostrovsky, and Berry (2007); Pesendorfer and Schmidt-Dengler (2008); and Arcidiacono and Miller (2011). The main advantage of two-step estimators is their computational simplicity: they do not require solving for an equilibrium. In a dynamic-game setting, where the cost of computing an equilibrium can increase drastically as the number of firms and/or state space increases, two-step estimators offer an attractive alternative to estimating structural models. However, the performance of two-step estimators depends largely on the accuracy of estimates in the first step and the criterion function used in the second step; see the discussion in Pakes, Ostrovsky, and Berry (2007). For dynamic games with many states, more data will be required to obtain an accurate estimate in the first step. Furthermore, if researchers wish to conduct counterfactual policy analysis after obtaining parameter estimates, then they still need to solve for an equilibrium (or the equilibria) of the game, which amounts to a similar order of magnitude of computational costs when solving the game within an estimation procedure.

Aguirregabiria and Mira (2007) have proposed the nested pseudo-likelihood (NPL) estimator for estimating discrete-choice games. The NPL estimator aims to decrease the potentially large finite-sample biases with two-step estimators. They also proposed the NPL algorithm, a recursive iteration of a two-step pseudo maximum-likelihood estimator, to compute a solution of the NPL estimator. When the NPL algorithm converges, it solves the structural equations and, hence, produces an equilibrium of the game. It has, however, been shown by Pakes, Ostrovsky, and Berry (2007) as well as Pesendorfer and Schmidt-Dengler (2008) that the NPL estimator can perform worse than two-step estimators in finite samples. Moreover, to achieve convergence, the NPL algorithm requires an implicit assumption that the equilibria that generate the observed data are stable under best response iterations or Lyapunov stable. This requirement imposes an undesirable equilibrium selection criterion on the data generating process. Pesendorfer (2010) has argued that best response stability is not a convincing assumption for games of incomplete information. When the best-response stable assumption is not satisfied, the NPL algorithm will either fail to converge or, worse, converge to the wrong equilibria, which leads to incorrect parameter estimates for the NPL estimator; see Pesendorfer and Schmidt-Dengler (2010) for such an example.

Su and Judd (2011) have proposed a constrained optimization approach to estimate structural models. They have demonstrated the use of their approach on the bus-engine replacement model in Rust (1987), and presented a general formulation that can be applied to estimate games with multiple equilibria. Following the insight in Su and Judd (2011), Vitorino (2011) was the first to use the constrained optimization approach to estimate an empirical entry-and-exit games. Assuming that only one equilibrium is played in each market in the data, I present a constrained optimization approach to maximum-likelihood estimation of discrete-choice games of incomplete information. My formulation, defined over the joint space of both structural parameters and economic equilibrium, yields a smooth likelihood function as the objective, and smooth structural equations as the constraints. The constrained optimization approach does not require repeatedly solving for an equilibrium or all the equilibria at each guess of structural parameters. Thus, this approach reduces the perceived computational burden of implementing the maximum-likelihood estimator. While I do not claim that the constrained optimization approach can solve large-scale dynamic games with billions of states, variables, and equations, I believe it offers a valuable alternative for estimating structural models with up to 120,000 variables and constraints, which can easily accommodate most of empirical papers we see in the literature.

Using a simple static game as an organizing example, I conducted Monte Carlo experiments to evaluate the finite-sample properties of the maximum-likelihood (ML), the two-step pseudo maximum-likelihood (2S-PML), the two-step least squares (2S-LS), and the NPL estimators. I examined three data generating processes, in which best-response stable and/or best-response unstable equilibria were used. Conditional on the same observables, I also varied the numbers of repeated observations in the data sets. These Monte Carlo exercises considered here are reasonably general and comparable to those in Aguirregabiria and Mira (2007), Pakes, Ostrovsky, and Berry (2007) as well as Pesendorfer and Schmidt-Dengler (2008).

My Monte Carlo results demonstrate that, in most cases, the ML estimator performs the best and the NPL estimator the worst. For all the estimators, the biases in the estimators decrease as the numbers of repeated observations increase in the data. The ML estimator produces average estimates that are within one standard deviation of the true parameter values, even when only five or ten repeated observations per market exist in the data. The performance of the two-step estimators depends on the type of equilibria used in the data generating process. If the data are generated by best-response stable equilibria, then the 2S-PML estimator produces accurate parameter estimates with moderate numbers of repeated observations, says twenty five or more per market. When the data consist of best-response stable and best-response unstable equilibria, the two-step estimators have high finite-sample biases in almost all cases; the biases decrease considerably with large numbers (100 or 250) of repeated observations. In the experiments, in most cases, the NPL estimator is more biased than either the ML estimator or the two-step estimators. When the data consist of best-response stable and unstable equilibria, the NPL algorithm fails to converge for all data sets. Even when the data are generated by only best-response stable equilibria, the NPL algorithm, surprisingly, often fails to converge with small numbers of repeated observations per market; for example, the NPL algorithm converges in less than five data sets (out of one hundred) with five repeated observations per market in the data. The NPL algorithm converges more frequently with more repeated observations per market. This finding suggests that the NPL algorithm is not a reliable computational strategy unless the underlying equilibria in the data generating process are stable under best-response iterations and large numbers of repeated observations exist in the data.

The remainder of the paper is organized as follows: in Section 2, I describe a simple static discrete-choice game of incomplete information originally proposed by Rust (2008), while in Section 3, I discuss estimating this game using the method of maximum-likelihood under two computational strategies: the nested fixed-point (NFXP) algorithm and the constrained optimization approach. Using a numerical example, I illustrate that the likelihood function of NFXP is discontinuous. Thus, the outer-loop optimization problem in NFXP will be computationally intractable. I then present the equivalent constrained optimization formulation for the maximum-likelihood estimation. I also discuss the two-step estimators as well as the NPL estimator. In Section 4, I present the results of the Monte Carlo experiments, reporting the finite-sample performance of the ML estimator, the two-step estimators and the NPL estimator. I summarize the conclusions in Section 5.

2 Static Discrete-Choice Games of Incomplete Information

The description of the example presented here follows closely the derivation of Rust (2008). For simplicity, I first describe the model for one market and then generalize the model to accommodate data observed in several differentiated markets. Similar models of static discrete-choice games of incomplete information have been proposed and studied by Seim (2006); Sweeting (2009); Bajari, Hong, Krainer, and Nekipelov (2010); Misra and Ellickson (2011) as well as Vitorino (2011).

2.1 Simple Static-Game Model with One Market

Consider a two-player, discrete-choice game of incomplete information with *observed* as well as *unobserved* heterogeneity. Let the firms be labeled a and b, and let y_a and y_b denote the choices of firms a and b, respectively. For simplicity, I assume that each firm has two possible choices; see, for example, the entry-and-exit games in Berry (1992), and Seim (2006) as well as Ciliberto and Tamer (2009).

Let

$$y_a = \begin{cases} 1 & \text{if firm } a \text{ is active,} \\ 0 & \text{if firm } a \text{ is inactive;} \end{cases}$$

and define y_b similarly. The *ex post* utility functions of firms a and b are assumed to be

$$u_a(y_a, y_b, x_a, \epsilon_a) = \begin{cases} [\alpha + y_b(\beta - \alpha)]\, x_a + \varepsilon_{a1}, & \text{if } y_a = 1, \\ 0 + \varepsilon_{a0}, & \text{if } y_a = 0; \end{cases}$$

and

$$u_b(y_a, y_b, x_b, \epsilon_b) = \begin{cases} [\alpha + y_a(\beta - \alpha)]\, x_b + \varepsilon_{b1}, & \text{if } y_b = 1, \\ 0 + \varepsilon_{b0}, & \text{if } y_b = 0; \end{cases}$$

where a scalar x_a is the observed type and a (2×1) vector $\boldsymbol{\varepsilon_a} = (\varepsilon_{a0}, \varepsilon_{a1})$ is the unobserved type for firm a. For firm b, x_b and $\boldsymbol{\varepsilon_b}$ are defined similarly. The structural parameters (to be estimated) (α, β) measure the effect of the observed type x_a and x_b on firm a's and b's utility, respectively. Note that a firm's utility is a function of the joint decision of both firms, (y_a, y_b).

I assume that the observed types (x_a, x_b) are common knowledge among firms; that unobserved types $(\boldsymbol{\varepsilon_a}, \boldsymbol{\varepsilon_b})$ are private information; that $\boldsymbol{\varepsilon_a}$ and $\boldsymbol{\varepsilon_b}$ are independent; and that firm a knows the distribution of $\boldsymbol{\varepsilon_b}$ and firm b knows the distribution of $\boldsymbol{\varepsilon_a}$.

Because of the private information $\boldsymbol{\varepsilon_a}$, firm a's decision will be probabilistic from firm b's point of view. Let p_a denote firm b's belief of the probability that firm a will be active. Similarly, let p_b denote firm a's belief of the probability that firm b will be active. Given firm b's belief p_a, the expected utility of firm b for taking an action y_b is given by

$$\begin{aligned} U_b(y_b, x_b, \boldsymbol{\varepsilon_b}) &= p_a u_b(1, y_b, x_b, \boldsymbol{\varepsilon_b}) + (1 - p_a) u_b(0, y_b, x_b, \boldsymbol{\varepsilon_b}) \\ &= \begin{cases} p_a \beta x_b + (1 - p_a)\alpha x_b + \varepsilon_{b1}, & \text{if } y_b = 1, \\ \varepsilon_{b0}, & \text{if } y_b = 0. \end{cases} \end{aligned}$$

It follows that firm b will be active ($y_b = 1$) if and only if

$$U_b(1, x_b, \boldsymbol{\varepsilon_b}) > U_b(0, x_b, \boldsymbol{\varepsilon_b}).$$

I assume that each component in the error terms $\boldsymbol{\varepsilon_a}$ and $\boldsymbol{\varepsilon_b}$ has a Type 1, extreme-value distribution, so the probability density function is $f(\varepsilon) = \exp(\varepsilon)\exp[-\exp(\varepsilon)]$. Given firm b's belief p_a, the probability that firm b will be active is given by the standard binomial logit formula

$$\begin{aligned}
p_b &= \Pr(y_b = 1) \\
&= \Pr\left[\boldsymbol{\varepsilon_b} | U_b(y_b = 1, x_b, \boldsymbol{\varepsilon_b}) > U_b(y_b = 0, x_b, \boldsymbol{\varepsilon_b})\right] \\
&= \frac{\exp[p_a \beta x_b + (1 - p_a)\alpha x_b]}{1 + \exp[p_a \beta x_b + (1 - p_a)\alpha x_b]} \\
&= \frac{1}{1 + \exp[-x_b \alpha + p_a x_b (\alpha - \beta)]} \\
&\equiv \Psi_b(p_a, p_b, x_b; \alpha, \beta).
\end{aligned} \tag{1}$$

This formula can be thought of as a *best response function* for firm b given b's belief p_a. Similarly, the best response for firm a, given a's belief p_b, is

$$\begin{aligned}
p_a &= \frac{\exp[p_b \beta x_a + (1 - p_b)\alpha x_a]}{1 + \exp[p_b \beta x_a + (1 - p_b)\alpha x_a]} \\
&= \frac{1}{1 + \exp[-x_a \alpha + p_b x_a (\alpha - \beta)]} \\
&\equiv \Psi_a(p_a, p_b, x_a; \alpha, \beta).
\end{aligned} \tag{2}$$

A *Bayes–Nash equilibrium* is a pair of beliefs (p_a^*, p_b^*) that are mutual best responses:

$$\begin{aligned}
p_a^* &= \frac{1}{1 + \exp[-x_a \alpha + p_b^* x_a (\alpha - \beta)]} &= \Psi_a(p_a^*, p_b^*, x_a; \alpha, \beta) \\
p_b^* &= \frac{1}{1 + \exp[-x_b \alpha + p_a^* x_b (\alpha - \beta)]} &= \Psi_b(p_a^*, p_b^*, x_b; \alpha, \beta).
\end{aligned} \tag{3}$$

To simplify the notation, let $\boldsymbol{\theta} = (\alpha, \beta)$, $\boldsymbol{x} = (x_a, x_b)$, $\boldsymbol{p} = (p_a, p_b)$ and $\boldsymbol{\Psi} = (\Psi_a, \Psi_b)$. I rewrite the Bayes–Nash (BN) equilibrium equation (3) as

$$\boldsymbol{p} = \boldsymbol{\Psi}(\boldsymbol{p}, \boldsymbol{x}; \boldsymbol{\theta}). \tag{4}$$

Given the parameters $\boldsymbol{\theta}$ and observed types $\boldsymbol{x}$, there are two unknowns in $\boldsymbol{p}$ in the two BN equilibrium equations defined in (3). Multiple solutions satisfying equation (3) can exist and, hence, multiple BN equilibria can exist. However, one can show that there are at most three equilibria in this model. The example below illustrates this case.

Example 1. Suppose $\boldsymbol{\theta}^0 = (\alpha^0, \beta^0) = (5, -11)$ and firms' observed types $\boldsymbol{x} = (0.52, 0.22)$.

Substituting these values into (3) and solving the following two equations for a BN equilibrium $\boldsymbol{p}^*$:

$$\begin{aligned} p_a &= \frac{1}{1+\exp(-4.48+8.32p_b)} \\ p_b &= \frac{1}{1+\exp(-1.10+3.52p_a)}. \end{aligned} \tag{5}$$

I used the constrained optimization solver, KNITRO, to solve this system of two equations and two unknowns. Since there could be multiple solutions, I tried one hundred different starting points and found the following three BN equilibria:

$$\begin{aligned} &\text{Equilibrium 1:} \quad \boldsymbol{p}_1^* = (0.030100, 0.729886), \\ &\text{Equilibrium 2:} \quad \boldsymbol{p}_2^* = (0.616162, 0.255615), \\ &\text{Equilibrium 3:} \quad \boldsymbol{p}_3^* = (0.773758, 0.164705). \end{aligned}$$

One can verify that Equilibrium 2 is not stable under best-response iterations, which means that best-response iterations will not converge to Equilibrium 2, even if the starting point is very close to that solution.[1]

2.2 Simple Static-Game Model with Multiple Markets

I generalize the model described above to accommodate multiple differentiated markets in the observed data. Assume that there are M markets. The characteristics that differentiate these markets are the firms' observed types. Thus, two vectors of different observed types represent two different markets. I denote by $\boldsymbol{x}^m = (x_a^m, x_b^m)$ the firms' observed types in market m, for $m = 1, \ldots, M$. I assume firms have the same vector of structural parameters $\boldsymbol{\theta} = (\alpha, \beta)$ in all markets, but firms' decisions are independent across the markets.

For each market, there is a set of BN equilibrium equations, parameterized by the observed types. Let $\boldsymbol{p}^m = (p_a^m, p_b^m)$ denote a vector of a BN equilibrium in market m, which satisfies the following equation:

$$\boldsymbol{p}^m = \boldsymbol{\Psi}(\boldsymbol{p}^m, \boldsymbol{x}^m; \boldsymbol{\theta}), \qquad \text{for } m = 1, \ldots, M. \tag{6}$$

[1]Note that best-response stable equilibrium is not among the common notions of stable equilibrium, such as strategic stability of Kohlberg and Mertens (1986), or Mertens (1989, 1991), or the evolutionarily stable strategy (ESS), studied in the game theory literature; see also Chapter 11 in Fudenberg and Tirole (1996).

Let $\boldsymbol{P} = (\boldsymbol{p}^m)_{m=1}^M$ denote the collection of equilibrium probabilities and $\boldsymbol{X} = (\boldsymbol{x}^m)_{m=1}^M$ denote the collection of the firms' observed types for all markets, respectively. With a slight abuse of notation, I simplify the BN equilibrium equations for all markets (6) as

$$\boldsymbol{P} = \boldsymbol{\Psi}(\boldsymbol{P}, \boldsymbol{X}; \boldsymbol{\theta}). \tag{7}$$

From (6), it is clear that markets with different observed types can have different BN equilibria. Also, the numbers of equilibria can be different for markets with different observed types, as illustrated in the following example.

Example 2. I selected the same parameter values $\boldsymbol{\theta}^0 = (\alpha^0, \beta^0) = (5, -11)$ as in Example 1 and considered a case with 256 markets. Discretizing the interval $[0.12, 0.87]$ into sixteen equally spaced grid points yields sixteen different observed types for each firm; jointly, there are $16 \times 16 = 256$ pairs of (x_a^m, x_b^m), for $m = 1, \ldots, 256$. Each of these pairs defines a market. For each market m, I solved for the corresponding BN equilibrium equation (6) with 100 different starting values to find all the BN equilibria $\boldsymbol{p}^{m*} = (p_a^{m*}, p_b^{m*})$ in that market.

In Figure 1, I present the plot of the numbers of equilibria in each market. As one can see, there are three equilibria in most markets. For markets with low x_a and/or x_b, the equilibrium is, however, unique. A small change in the observed types can result in a relatively large change in the number of equilibria; for example, there are three equilibria in the market with observed types $(0.17, 0.87)$, but there is only one equilibrium in the market with the observed types $(0.12, 0.87)$.

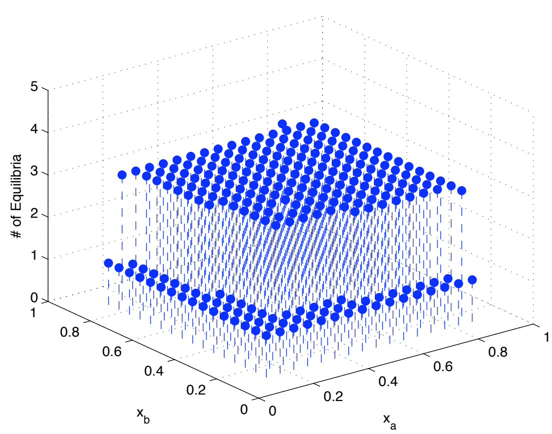


Figure 1: Numbers of equilibria in different markets

3 Estimation

In this section, I describe the data generating process and discuss various estimators used to estimate static discrete-choice games of incomplete information.

3.1 Data Generating Process

Following a common assumption in the literature (see, for example, Aguirregabiria and Mira [2007]; Bajari, Benkard, and Levin [2007]; Pakes, Ostrovsky, and Berry [2007] as well as Pesendorfer and Schmidt-Dengler [2008]), I assume that in each market, only one equilibrium is played in the data. Since equilibrium solutions are different in different markets, as was demonstrated in Example 2, in the data I allow different equilibria to be played in different markets.

Assumption 2: In each market, the firms use the same equilibrium to play independently over the T periods. However, equilibria played across different markets are different.

Researchers observe firms' types $\boldsymbol{x}^m = (x_a^m, x_b^m)$ and decisions $\boldsymbol{y}^m = (y_a^{mt}, y_b^{mt})_{t=1}^T$ in each market m over T periods in the data. Let $\boldsymbol{Z}^m$ denote the data observed in market m:

$$\boldsymbol{Z}^m = \{\boldsymbol{x}^m = (x_a^m, x_b^m),\, \boldsymbol{y}^m = (y_a^{mt}, y_b^{mt})_{t=1}^T\}.$$

Data observed for all M markets over T periods are the collection of $\boldsymbol{Z}^m$ for all m, so

$$\boldsymbol{Z} = \{\boldsymbol{Z}^m, \text{ for } m = 1, \ldots, M\}. \tag{8}$$

3.2 Maximum-Likelihood Estimation

I consider estimating the parameters of the discrete-choice game using the method of maximum-likelihood. I describe two computational algorithms for implementing the ML estimator: the NFXP algorithm and the constrained optimization approach. I demonstrate that these two computational methods solve the same estimation problem; hence, they produce the same estimates.

3.2.1 ML Estimation using the NFXP Approach

Given the parameter vector $\boldsymbol{\theta}$ and firms' observed types $\boldsymbol{x}^m$, let $\bar{\boldsymbol{p}}^m(\boldsymbol{\theta}) = (\bar{p}_a(\boldsymbol{x}^m;\boldsymbol{\theta}), \bar{p}_b(\boldsymbol{x}^m;\boldsymbol{\theta}))$ denote a BN equilibrium that solves equation (3) for market m. Given Assumption 2, if the observed decisions $\boldsymbol{y}^m = (y_a^{mt}, y_b^{mt})_{t=1}^T$ were generated by $\bar{\boldsymbol{p}}^m(\boldsymbol{\theta})$, then the logarithm of the likelihood of the observing data $\boldsymbol{Z}^m$ in market m at the parameters $\boldsymbol{\theta}$ is

$$\begin{aligned} & L_i[\boldsymbol{Z}^m; \bar{\boldsymbol{p}}^m(\boldsymbol{\theta}); \boldsymbol{\theta}] \\ = & \sum_{t=1}^{T} \left\{ y_a^{mt} \times \log[\bar{p}_a(\boldsymbol{x}^m;\boldsymbol{\theta})] + (1 - y_a^{mt}) \times \log[1 - \bar{p}_a(\boldsymbol{x}^m;\boldsymbol{\theta})] \right\} \\ + & \sum_{t=1}^{T} \left\{ y_b^{mt} \times \log[\bar{p}_b(\boldsymbol{x}^m;\boldsymbol{\theta})] + (1 - y_b^{mt}) \times \log[1 - \bar{p}_b(\boldsymbol{x}^m;\boldsymbol{\theta})] \right\}, \end{aligned} \tag{9}$$

Let $\bar{\boldsymbol{P}}(\boldsymbol{\theta}) = (\bar{\boldsymbol{p}}^m(\boldsymbol{\theta}))_{m=1}^M$. Thus, the logarithm of the likelihood of observing the data $\boldsymbol{Z}$ for all markets at the parameters $\boldsymbol{\theta}$ is

$$L[\boldsymbol{\theta}, \bar{\boldsymbol{P}}(\boldsymbol{\theta})] = \sum_{m=1}^{M} L_i[\boldsymbol{Z}^m; \bar{\boldsymbol{p}}^m(\boldsymbol{\theta}), \boldsymbol{\theta}]. \tag{10}$$

Since multiple BN equilibria can exist for each $\boldsymbol{\theta}$, the ML estimator is defined as

$$\boldsymbol{\theta}^{MLE} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ \max_{\bar{\boldsymbol{P}}(\boldsymbol{\theta})} L[\boldsymbol{\theta}, \bar{\boldsymbol{P}}(\boldsymbol{\theta})] \right\}. \tag{11}$$

Calculating $\boldsymbol{\theta}^{MLE}$ using the NFXP algorithm of Rust (1987) is described as follows: in the outer loop, search the structural parameter space over $\boldsymbol{\theta}$ to maximize the objective function $\left\{ \max_{\bar{\boldsymbol{P}}(\boldsymbol{\theta})} L[\boldsymbol{\theta}, \bar{\boldsymbol{P}}(\boldsymbol{\theta})] \right\}$; in the inner loop, for a given $\boldsymbol{\theta}$, find all the BN equilibria, evaluate the corresponding likelihood value at each BN equilibrium, and choose the equilibrium that yields the highest likelihood value. The algorithm then returns to the outer loop and repeats until it converges, or fails.

Two important computational difficulties arise when applying NFXP to solve the ML estimator (11): first, researchers must find all the equilibria for any given structural parameters $\boldsymbol{\theta}$; second, the objective function $\left\{ \max_{\bar{\boldsymbol{P}}(\boldsymbol{\theta})} L[\boldsymbol{\theta}, \bar{\boldsymbol{P}}(\boldsymbol{\theta})] \right\}$ can be a discontinuous function of $\boldsymbol{\theta}$. In the example below, I illustrate such a case.

Example 3. Consider the setting in Example 1, in which there are three BN equilibria given $\boldsymbol{\theta}^0 = (\alpha^0, \beta^0) = (5, -11)$ and firms' observed types $\boldsymbol{x} = (0.52, 0.22)$. I first assume that both firms use Equilibrium 1 to play the game 1,000 times and, hence, randomly generate 1,000 pairs of observed decisions $\boldsymbol{y_1} = (y^t_{a1}, y^t_{b1})^{1000}_{t=1}$ independently using Equilibrium 1. Let $\boldsymbol{Z}_1$ denote this data set. Repeat the same procedure, but use Equilibrium 2 and Equilibrium 3 to generate data sets $\boldsymbol{Z}_2$ and $\boldsymbol{Z}_3$, respectively. Fixing $\boldsymbol{x} = (0.52, 0.22)$, I then plotted the numbers of equilibria in the parameter space $\boldsymbol{\theta}$ and the corresponding logarithm of the likelihood function for each of the three data sets in Figure 2. As one can see, the logarithm of the likelihood function for $\boldsymbol{Z}_1$ is continuous in $\boldsymbol{\theta}$, while the logarithms of the likelihood functions for $\boldsymbol{Z}_2$ and $\boldsymbol{Z}_3$ are discontinuous. Recall that Equilibrium 2 is unstable under best-response iterations, while Equilibrium 3 is stable under best response. This example demonstrates that the discontinuity in the likelihood function does not depend on best-response stability of an equilibrium. Instead, the discontinuity arises from the change in the numbers of equilibria for different $\boldsymbol{\theta}$s.

3.2.2 ML Estimation using the Constrained Optimization Approach

I next describe the constrained optimization approach to estimating games of incomplete information under the method of maximum-likelihood.

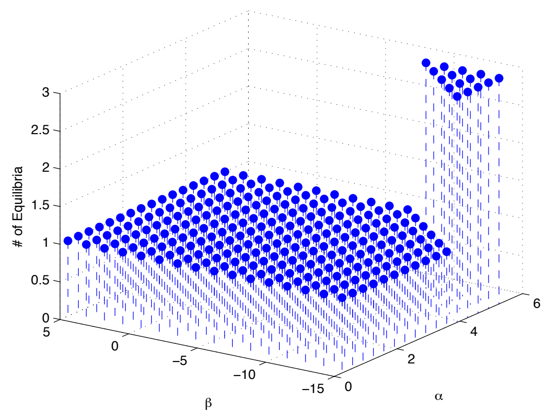
Let $\boldsymbol{p}^m = (p^m_a, p^m_b)$ denote any vector of probabilities in market m and $\boldsymbol{P} = (\boldsymbol{p}^m)^M_{m=1}$. Assuming the observed decisions $\boldsymbol{y}^m = (y^{mt}_a, y^{mt}_b)^T_{t=1}$ were generated by $\boldsymbol{p}^m$, I define the *augmented* logarithm of the likelihood function for observing $\boldsymbol{Z}^m$ in market m as

$$\begin{aligned} \mathcal{L}_i(\boldsymbol{Z}^m; \boldsymbol{p}^m, \boldsymbol{\theta}) &= \sum_{t=1}^{T} \left[y^{mt}_a \times \log(p^m_a) + (1 - y^{mt}_a) \times \log(1 - p^m_a) \right] \\ &+ \sum_{t=1}^{T} \left[y^{mt}_b \times \log(p^m_b) + (1 - y^{mt}_b) \times \log(1 - p^m_b) \right]. \end{aligned} \tag{12}$$

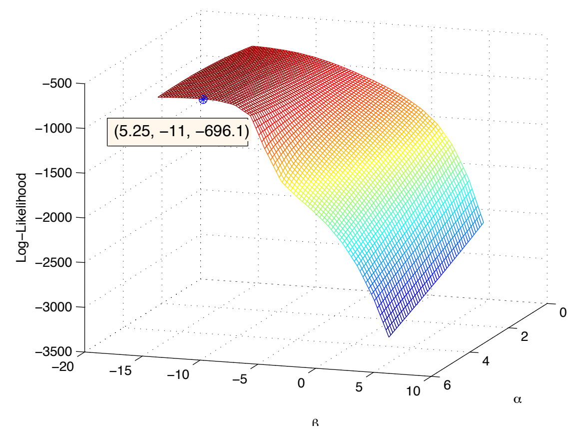
The augmented logarithm of the likelihood of observing the full data $\boldsymbol{Z}$ for all markets is

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{P}) = \sum_{m=1}^{M} \mathcal{L}_i(\boldsymbol{Z}^m; \boldsymbol{p}^m, \boldsymbol{\theta}). \tag{13}$$

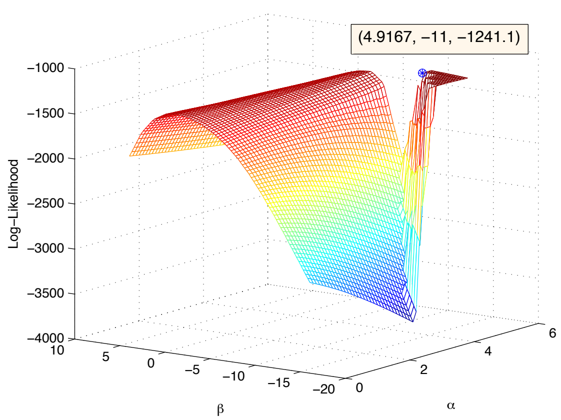
To ensure that, given $\boldsymbol{\theta}$, $\boldsymbol{P}$ is a collection of BN equilibria for all markets, I imposed the



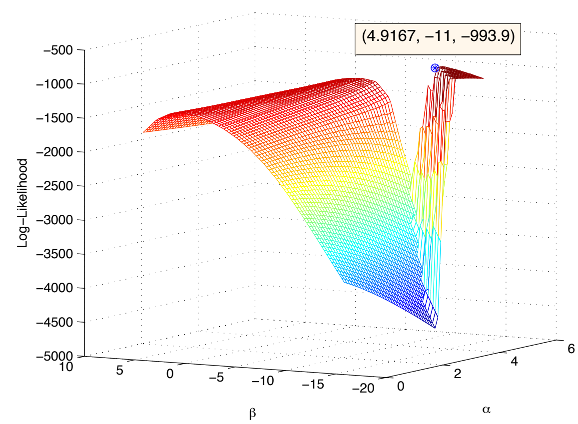
(a) Numbers of Equilibria in the Parameter Space.



(b) Likelihood Function with Equilibrium 1.



(c) Likelihood Function with Equilibrium 2.



(d) Likelihood Function with Equilibrium 3.

Figure 2: Numbers of Equilibria and Examples of the Logarithms of Likelihood Functions. The Circle in (b) – (d) Indicates the Maximizer Obtained from Grid Search.

BN equilibrium equation (7) as constraints. The ML estimation problem is then

$$\begin{aligned} \max_{(\boldsymbol{\theta},\boldsymbol{P})} \quad & \mathcal{L}(\boldsymbol{\theta},\boldsymbol{P}) \\ \text{subject to} \quad & \boldsymbol{P} = \boldsymbol{\Psi}(\boldsymbol{P},\boldsymbol{x},\boldsymbol{\theta}). \end{aligned} \tag{14}$$

The decision variables in my formulation are $\boldsymbol{\theta}$ and $\boldsymbol{P}$. Note, too, that the structural parameters $\boldsymbol{\theta}$ do not directly enter either the augmented function (12) or the objective function (13). Instead of defining the likelihood function as a (potentially) discontinuous function of θ in (11), the objective function $\mathcal{L}(\boldsymbol{\theta},\boldsymbol{P})$ is a smooth function of the equilibrium probabilities $\boldsymbol{P}$. Thus, the constrained optimization approach yields a smooth objective function as well as smooth constraints. Consequently, one can use state-of-the-art constrained optimization solvers to compute a solution to the ML estimation problem (14).

Following Proposition 1 in Su and Judd (2011), I state the equivalence of the ML estimators formulated in (11) and (14).

Proposition 1. Let $\bar{\boldsymbol{\theta}}$ be a solution of the ML estimation problem defined in (11). Denote $\bar{\boldsymbol{P}}^*(\boldsymbol{\theta}) = \underset{\bar{\boldsymbol{P}}(\boldsymbol{\theta})}{\operatorname{argmax}}\, L[\boldsymbol{\theta},\bar{\boldsymbol{P}}(\boldsymbol{\theta})]$. Let $(\boldsymbol{\theta}^*,\boldsymbol{P}^*)$ be a solution of the constrained optimization problem (14). Then $L[\bar{\boldsymbol{\theta}},\bar{\boldsymbol{P}}^*(\bar{\boldsymbol{\theta}})] = \mathcal{L}(\boldsymbol{\theta}^*,\boldsymbol{P}^*)$. If the model is identified, then $\bar{\boldsymbol{\theta}} = \boldsymbol{\theta}^*$.

Proof. See Proposition 1 in Su and Judd (2011).

Aitchison and Silvey (1958) as well as Silvey (1975) have demonstrated that the ML estimator formulated in (14) is consistent as well as asymptotically normal. I refer the reader to Aitchison and Silvey (1958) for detailed derivation and proof, and Silvey (1975) for a concise discussion.

3.3 Two-Step Estimators

Hotz and Miller (1993) pioneered using two-step estimators to estimate single-agent dynamic discrete-choice models. One attractive feature of two-step estimators is computational simplicity, at least when compared to the NFXP algorithm or the constrained optimization approach. In this section, I describe the 2S-PML estimator and the 2S-LS estimator for estimating the discrete-choice game.

For a given vector of probabilities $\widehat{\boldsymbol{P}} = (\hat{\boldsymbol{p}}^m)_{m=1}^M$, the pseudo likelihood function of observing the data $\boldsymbol{Z}^m$ in market m is defined as

$$
\begin{aligned}
& L_i^{PML}(\boldsymbol{Z}^m, \hat{\boldsymbol{p}}^m; \boldsymbol{\theta}) \\
= & \sum_{t=1}^{T} \left\{ y_a^{mt} \times \log[\Psi_a(\hat{p}_a^m, \hat{p}_b^m, \boldsymbol{x}^m, \boldsymbol{\theta})] + (1 - y_a^{mt}) \times \log[1 - \Psi_a(\hat{p}_a^m, \hat{p}_b^m, \boldsymbol{x}^m, \boldsymbol{\theta})] \right\} \\
+ & \sum_{t=1}^{T} \left\{ y_b^{mt} \times \log[\Psi_b(\hat{p}_a^m, \hat{p}_b^m, \boldsymbol{x}^m, \boldsymbol{\theta})] + (1 - y_b^{mt}) \times \log[1 - \Psi_b(\hat{p}_a^m, \hat{p}_b^m, \boldsymbol{x}^m, \boldsymbol{\theta})] \right\}.
\end{aligned}
$$

The pseudo likelihood of observing the full data $\boldsymbol{Z}$ for all markets is then

$$
L^{PML}(\boldsymbol{\theta}, \widehat{\boldsymbol{P}}) = \sum_{m=1}^{M} L_i^{PML}(\boldsymbol{Z}^m, \hat{\boldsymbol{p}}^m; \boldsymbol{\theta}). \tag{15}
$$

Notice that when defining the pseudo likelihood function (15), the component $\hat{\boldsymbol{p}}^m$ in $\widehat{\boldsymbol{P}}$ need not be a BN equilibrium. The pesudo likelihood function is well defined for any probabilities $\widehat{\boldsymbol{P}}$.

The 2S-PML estimator is as follows:

Step 1: Find $\widehat{\boldsymbol{P}}_0$, a consistent estimate of the true equilibrium probabilities $\boldsymbol{P}^0$.

Step 2: Fix $\widehat{\boldsymbol{P}}_0$. Solve the pseudo maximum-likelihood estimator: (16)

$$
\boldsymbol{\theta}^{2S-PML} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \; L^{PML}(\boldsymbol{\theta}, \widehat{\boldsymbol{P}}_0).
$$

One can also use the 2S-LS estimator of Pesendorfer and Schimdt-Dengler (2008). Instead of maximizing the pseudo likelihood function, one chooses structural parameters $\boldsymbol{\theta}$ to minimize the ℓ_2 norm of the errors in the BN equilibrium equation (7) in the 2S-LS estimator:[2]

Step 1: Find $\widehat{\boldsymbol{P}}_0$, a consistent estimate of the true equilibrium probabilities $\boldsymbol{P}^0$.

Step 2: Fix $\widehat{\boldsymbol{P}}_0$. Solve the least square problem: (17)

$$
\boldsymbol{\theta}^{2S-LS} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \, \|\widehat{\boldsymbol{P}}_0 - \boldsymbol{\Psi}(\widehat{\boldsymbol{P}}_0, \boldsymbol{x}; \boldsymbol{\theta})\|_2^2.
$$

[2]Pesendorfer and Schimdt-Dengler (2008) proposed an asymptotic least-squares estimator and derived the asymptotically optimal weight matrix. Here, I do not derive the optimal weight matrix and use the identity matrix as the weight matrix in this example.

For two-step estimators, the optimization problem in the second step involves only structural parameters. Also, the BN equilibrium equations are not imposed in the second step and, hence, may not be satisfied. Because two-step estimators do not solve the BN equilibrium equations, they are computationally light. However, two-step estimators can perform poorly in finite samples when the first-step estimates are imprecise due to insufficient amounts of data, or when researchers do not choose suitable criterion functions in the second step; see the discussion in Pakes, Ostrovsky, and Berry (2007). Furthermore, if researchers seek to conduct counterfactual analysis after obtaining parameter estimates, they still need to solve the game for an equilibrium (or the equilibria), which amounts to a similar order of magnitude of computational burden when using the constrained optimization approach for maximum-likelihood estimation.

3.4 The Nested Pseudo-Likelihood Estimator

Recognizing the limitations of two-step estimators, Aguirregabiria and Mira (2007) have proposed the NPL estimator for estimating dynamic discrete games, aiming to reduce the finite-sample biases of the 2S-PML. They also proposed the NPL algorithm, a recursive iteration of the 2S-PML estimator, to compute a solution of the NPL estimator.

The NPL algorithm is described as follows: first, find an initial guess of equilibrium probabilities $\widetilde{\boldsymbol{P}}_0$. For $K \geq 1$, fix $\widetilde{\boldsymbol{P}}_{K-1}$ and solve the pseudo maximum-likelihood estimator for $\tilde{\boldsymbol{\theta}}_K$:

$$\tilde{\boldsymbol{\theta}}_K = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, L^{PML}(\boldsymbol{\theta}, \widetilde{\boldsymbol{P}}_{K-1}). \tag{18}$$

Given $\tilde{\boldsymbol{\theta}}_K$, obtain $\widetilde{\boldsymbol{P}}_K$ by one best-response iteration on the equilibrium equation (7):

$$\widetilde{\boldsymbol{P}}_K = \boldsymbol{\Psi}(\widetilde{\boldsymbol{P}}_{K-1}, \boldsymbol{x}; \tilde{\boldsymbol{\theta}}_K). \tag{19}$$

Increase K by 1 and repeat the above procedure until convergence or if the maximum number of iterations $\bar{K}$ is reached, then declare a failed run and restart with a new initial guess.

When the NPL algorithm converges, a solution $(\boldsymbol{\theta}^{NPL}, \boldsymbol{P}^{NPL})$ satisfies the BN equilibrium equation (7). However, there are serious drawbacks with the NPL algorithm and, hence, the NPL estimator. First, since the NPL algorithm performs one best-response iteration (19) to update the equilibrium probabilities $\widetilde{\boldsymbol{P}}_K$ in the K^{th} recursive iteration, it will only converge to equilibria that are stable under best response. If the data were generated by equilibria that are unstable under best response iterations, then the NPL algorithm will either fail to converge or, worse, converge to wrong equilibria, which then leads to incorrect parameter estimates for the NPL estimator; see Pesendorfer and Schmidt-Dengler (2010) for such an example. Second, even when the data are generated by best-response stable equilibria, the NPL algorithm can generate cycling iterations and, consequently, fail to converge. Third, the NPL algorithm can take many recursive iterations before it converges and, computationally, it is not as efficient as the constrained optimization approach. I illustrate these points using the Monte Carlo experiments below.

4 Monte Carlo

I used Example 2 as the econometric model in the Monte Carlo experiments to study finite-sample performance of four estimators: the ML (with the constrained optimization approach), the 2S-PML, the 2S-LS, and the NPL estimators.[3] Recall that, in Example 2, the true parameter values are $\boldsymbol{\theta}^0 = (5, -11)$ and $M = 256$ differentiated markets exist.

4.1 Experiment Specifications

In generating the data, I maintained the assumption that in each market only one equilibrium is played; however, different equilibria can be played in different markets. I describe the types of equilibria used in the data generating process below.

Scenario 1: The best-response stable equilibrium with lowest probability of being active for firm a is played in each market for markets with multiple equilibria.
Scenario 2: Randomly choose one of the best-response stable equilibria to be played in each market for markets with multiple equilibria.
Scenario 3: Randomly choose one equilibrium to be played in each market. The equilibrium chosen can be stable or unstable under best response iterations.

For each data set, I simulated the static game for T periods to generate T pairs of observed decisions, $\boldsymbol{y}^m = (y_a^{mt}, y_b^{mt})_{t=1}^T$ in each market $m = 1, \ldots, 256$. Conditional on

[3] As noted by Pakes, Ostrovsky, and Berry (2007), the pseudo likelihood function is an inappropriate criterion function to use in the second step. The reason that I still use the 2S-PML estimator in my Monte Carlo experiments is to provide a direct comparison on the performance of the 2S-PML to that of the ML estimator and the NPL estimator.

the observables $\boldsymbol{x}^m$, the number of periods T (ranging from 5 to 250) gives the number of repeated observations per market in the data. For each scenario, I constructed one hundred data sets for each T.

Since the firms' decisions are observed in every market, I used a frequency estimator to estimate the first-step equilibrium probabilities for two-step estimators. The estimated equilibrium probabilities $\widehat{\boldsymbol{P}}_{\text{freq}} = (\hat{\boldsymbol{p}}^m_{\text{freq}})_{m=1}^{256}$ from the frequency estimator solve

$$\hat{p}^m_{a,\text{freq}} = \frac{1}{T}\sum_{t=1}^{T} y_a^{mt}, \quad \hat{p}^m_{b,\text{freq}} = \frac{1}{T}\sum_{t=1}^{T} y_b^{mt}, \text{ for } m = 1, \ldots, 256.$$

For the constrained optimization approach, there are $2 + 2 \times 256 = 514$ decision variables and 512 equality constraints (two equilibrium probabilities (p_a^m, p_b^m) as well as two BN equilibrium equations in each market). To find the parameter estimates for each data set, I used one hundred different starting points for $\boldsymbol{\theta}$ and the frequency estimates $\widehat{\boldsymbol{P}}_{\text{freq}}$ as the starting value for $\boldsymbol{P}$. For the NPL estimator, I used $\widehat{\boldsymbol{P}}_{\text{freq}}$, $\frac{3\widehat{\boldsymbol{P}}_{\text{freq}}}{4}$, and $\frac{\widehat{\boldsymbol{P}}_{\text{freq}}}{2}$ as the three initial guesses of equilibrium probabilities for the NPL algorithm for each data set. In addition, I also used logit probabilities as an initial guess for the NPL algorithm. I set the maximum number of NPL iterations to $\bar{K} = 1000$. If the number of NPL iterations reaches $\bar{K}$, then I declared that the NPL algorithm failed to converge in that run. If the NPL algorithm failed to converge from all four starting values, I declared that the NPL algorithm failed to converge to a solution for the NPL estimator for that data set.

I coded the optimization problem for each estimator in AMPL and called the nonlinear optimization solver KNITRO to solve the problem. I chose AMPL as the programing platform because AMPL uses automatic differentiation to compute exact first-order and second-order derivatives efficiently and passes the derivative information together with sparsity structure of the constrained Jacobian and Hessian matrices to optimization solvers. The derivatives as well as the sparsity structure information are necessary for KNITRO (and other optimization solvers) to perform well, especially on large-scale problems. With this choice of software for numerical implementation, I hope to provide a fair comparison on the numerical performance of each estimator.

4.2 Monte Carlo Results

The Monte Carlo results are reported in Tables 1 to 3. Overall, my results indicate that the ML estimator performs the best and the NPL estimator the worst. For the two-step estimators, in general, the 2S-PML estimator performs better than the 2S-LS estimator.[4] For the NPL estimator, Aguirregabiria and Mira (2007) found that the NPL algorithm with logit estimates as an initial guess converged faster than that with frequency estimates. On the contrary, I find the NPL algorithm with frequency estimates as initial guesses performs better.[5] Except for $T = 5$ in Scenario 1, NPL with logit estimates either often fails to converge (in Scenario 1) or converges to wrong parameter estimates (in Scenario 2). Hence, I focus on the performance of NPL with frequency estimates as initial guesses in the discussion below.

For many experiments in Scenarios 1 and 2, the ML and the 2S-PML estimators do quite well at recovering the true parameter values. Even with small numbers of repeated observations ($T = 5$ or 10) per market, the mean of the ML estimator is within one standard deviation of the true parameters values; on the contrary, the mean of parameter estimates on α of two-step estimators are one standard devision away from the true value. The biases and the root mean square error (RMSE) in the ML estimator are significantly smaller than those of the two-step estimators when the numbers of repeated observations per market are small. The 2S-PML estimator produces accurate parameter estimates with twenty-five or more repeated observations per market in the data. Given sufficiently many repeated observations, the two-step estimators can provide good parameter estimates with little computational efforts. The NPL estimator, surprisingly, fails frequently for $T = 50$ or smaller; for example, the NPL algorithm converges in only two data sets for $T = 5$ and in 29 data sets for $T = 10$ in Scenario 1. In those failed runs, the NPL algorithm generates cycling iterations without any indication of achieving convergence. This finding suggests that the convergence of the NPL algorithm is not as robust as perceived in Aguirregabiria and Mira (2007), even data are generated by best-response stable equilibria. In cases where the NPL algorithm does converge, the NPL estimator is more biased than the ML estimator or the two-step estimators. The NPL estimator requires large numbers of repeated observations ($T = 100$ for Scenario 1 and $T = 250$ for Scenario 2) to obtain estimates that are comparable to those

[4] One can improve the performance of the 2S-LS estimator by using the optimal weighting matrix, which has been suggested by Pesendorfer and Schimdt-Dengler (2008).

[5] In my experiment, the NPL algorithm usually converges to the same parameter estimates from starting values $\widehat{\boldsymbol{P}}_{\text{freq}}$, $\frac{3\widehat{\boldsymbol{P}}_{\text{freq}}}{4}$, and $\frac{\widehat{\boldsymbol{P}}_{\text{freq}}}{2}$.

of the ML and the two-step estimators.

In terms of computation time, the two-step estimators are fast, perhaps not a surprising result because the two-step estimators do not require solving a BN equilibrium. Both the constrained optimization approach and the NPL estimator impose the equilibrium constraints to be satisfied and, hence, require more computational time than the two-step estimators. As one can see from Tables 1 and 2, the constrained optimization approach requires only one to two seconds of computing time per starting point. Aguirregabiria and Mira (2007) suggested that the NPL estimator requires a relatively small additional computational cost over the 2S-PML estimator. In my experiments, however, the NPL algorithm requires on average more than 160 NPL iterations (i.e., iterating over the 2S-PML for more than 160 times) and around thirty seconds to converge per starting point for $T = 1000$ or sixty seconds for $T = 250$. Hence, for Scenarios 1 and 2, the constrained optimization approach is about thirty times faster than the NPL algorithm in computing time for $T = 100$ and $T = 250$.

For Scenario 3, the ML estimator is the only estimator that recovers the true parameter values. Even though some equilibria used to generate data are unstable under best response, the mean of the ML estimator is within one standard derivation of the true values.This finding should not be surprising because constrained optimization algorithms do not rely on best-response iterations. Thus, the presence of best-response unstable equilibria in the data should not affect the performance of the ML estimator under the constrained optimization approach. For the two-step estimators, both the 2S-PML and the 2S-LS estimators reject the true parameter values $\boldsymbol{\theta}^0 = (5, -11)$. The mean of parameter estimates are at least one standard deviation away from the true values in all experiments; the biases are significantly higher than those of Scenarios 1 and 2. In general, the 2S-PML estimator performs better than the 2S-LS estimator; however, 250 repeated observations are required for the 2S-PML estimator to produce reasonable parameter estimates. The NPL algorithm fails to converge and cannot compute a solution of the NPL estimator in all one hundred data sets in this scenario. Pesendorfer and Schmidt-Dengler (2010) have provided a counterexample that illustrates that the NPL algorithm produces wrong parameter values. By using the one best-response iteration update in equation (19), the NPL algorithm and, as a result, the NPL estimator implicitly requires that equilibria in the data are stable under best response. Obviously, this assumption is violated in the data generating process for Scenario 3.

5 Conclusion

I have proposed a constrained optimization formulation of the ML estimation problem for static games of incomplete information and conducted Monte Carlo experiments to examine the finite-sample performance of the ML estimator, the two-step estimators, and the NPL estimator. My Monte Carlo results demonstrate that the finite-sample performance of the ML estimator is superior to those of the two-step estimators as well as the NPL estimator, particularly when small numbers of repeated observations exist in the data. The two-step estimators perform well when there are sufficiently many repeated observations to estimate equilibrium probabilities accurately in the first step; however, they can be biased when only small numbers of repeated observations are available or when the data are generated by best-response unstable equilibria. The NPL estimator frequently fails to converge when only small numbers of repeated observations are available and generates higher biases than the ML estimator and the two-step estimators. Also, when it does converge, the NPL estimator requires many more recursive iterations than the 2S-PML estimator to converge. My findings are important and valuable because researchers need to be aware of the trade-off between computational costs and the accuracy of parameter estimates. To avoid unintended consequences, researchers also need to understand the implications of choosing specific computational procedures to implement specific estimation algorithms.

Table 1: Best-Response Stable Equilibrium with Lowest Probabilities of Entry for firm a.

T	Estimator	Estimates		RMSE	CPU Time Per Run (sec.)	# of Data Sets Converged	Avg. NPL Iter.
		α	β				
	Truth	5	-11	–	–	–	–
5	ML (Cons. Opt.)	5.234 (0.278)	-11.238 (0.506)	0.665	0.692	100	–
5	2-Step PML	4.459 (0.276)	-10.646 (0.796)	1.058	0.040	100	–
5	2-Step LS	4.514 (0.347)	-11.369 (1.100)	1.300	0.053	100	–
5	NPL (freq. prob.)	4.863 (0.241)	-10.019 (1.830)	1.639	36.051	2	987
5	NPL (logit prob.)	5.105 (0.193)	-10.173 (0.629)	1.057	28.477	39	762
10	ML (Cons. Opt.)	5.065 (0.143)	-11.111 (0.345)	0.393	0.441	100	–
10	2-Step PML	4.787 (0.165)	-10.886 (0.529)	0.602	0.043	100	–
10	2-Step LS	4.914 (0.238)	-11.473 (0.852)	1.002	0.055	100	–
10	NPL (freq. prob.)	5.054 (0.241)	-10.411 (1.830)	0.958	33.153	29	808
10	NPL (logit prob.)	5.096 (0.136)	-10.219 (0.482)	0.928	35.840	28	847
25	ML (Cons. Opt.)	5.018 (0.076)	-11.022 (0.181)	0.197	0.417	100	–
25	2-Step PML	4.926 (0.114)	-11.040 (0.264)	0.298	0.058	100	–
25	2-Step LS	5.014 (0.147)	-11.387 (0.479)	0.632	0.057	100	–
25	NPL (freq. prob.)	4.995 (0.081)	-10.607 (0.563)	0.688	29.122	71	543
25	NPL (logit prob.)	5.076 (0.096)	-10.164 (0.280)	0.888	50.674	28	856
50	ML (Cons. Opt.)	5.000 (0.061)	-11.000 (0.133)	0.146	0.398	100	–
50	2-Step PML	4.956 (0.080)	-10.983 (0.198)	0.218	0.090	100	–
50	2-Step LS	5.007 (0.109)	-11.119 (0.329)	0.365	0.056	100	–
50	NPL (freq. prob.)	4.998 (0.070)	-10.665 (0.472)	0.581	32.133	86	409
50	NPL (logit prob.)	5.119 (0.093)	-10.226 (0.238)	0.821	80.510	16	913
100	ML (Cons. Opt.)	5.005 (0.046)	-10.996 (0.103)	0.112	0.858	100	–
100	2-Step PML	4.985 (0.060)	-11.011 (0.164)	0.175	0.164	100	–
100	2-Step LS	5.011 (0.077)	-11.090 (0.238)	0.265	0.056	100	–
100	NPL (freq. prob.)	5.005 (0.051)	-10.908 (0.283)	0.301	34.516	96	242
100	NPL (logit prob.)	5.061 (0.048)	-10.130 (0.249)	0.906	155.480	15	942
250	ML (Cons. Opt.)	5.000 (0.031)	-10.995 (0.062)	0.069	1.798	100	–
250	2-Step PML	4.994 (0.037)	-11.002 (0.092)	0.099	0.379	100	–
250	2-Step LS	5.005 (0.042)	-11.025 (0.152)	0.160	0.057	100	–
250	NPL (freq. prob.)	5.002 (0.051)	-10.955 (0.283)	0.198	57.083	100	174
250	NPL (logit prob.)	5.138 (0.022)	-10.276 (0.045)	0.739	383.240	3	985

Standard deviations are reported in parentheses.

Table 2: Best-Response Stable Equilibrium in Each Market

T	Estimator	Estimates		RMSE	CPU Time Per Run (sec.)	# of Data Sets Converged	Avg. NPL Iter.
		α	β				
	Truth	5	-11	–	–	–	–
5	ML (Cons. Opt.)	5.197 (0.245)	-11.189 (0.463)	0.588	0.803	100	–
5	2-Step PML	4.380 (0.263)	-10.427 (0.711)	1.132	0.040	100	–
5	2-Step LS	4.395 (0.318)	-11.131 (1.078)	1.278	0.053	100	–
5	NPL (freq. prob.)	4.707 (0.241)	-8.534 (1.830)	2.574	34.847	4	975
5	NPL (logit prob.)	1.738 (0.026)	-3.318 (0.046)	8.346	28.712	3	988
10	ML (Cons. Opt.)	5.104 (0.149)	-11.038 (0.304)	0.354	0.472	100	–
10	2-Step PML	4.787 (0.181)	-10.831 (0.523)	0.615	0.043	100	–
10	2-Step LS	4.893 (0.243)	-11.418 (0.805)	0.942	0.055	100	–
10	NPL (freq. prob.)	5.019 (0.148)	-9.732 (0.753)	1.534	28.135	46	682
10	NPL (logit prob.)	N/A (N/A)	N/A (N/A)	N/A	33.175	0	1000
25	ML (Cons. Opt.)	5.040 (0.085)	-10.992 (0.193)	0.214	0.245	100	–
25	2-Step PML	4.945 (0.113)	-10.943 (0.307)	0.335	0.059	100	–
25	2-Step LS	5.022 (0.135)	-11.175 (0.509)	0.553	0.057	100	–
25	NPL (freq. prob.)	5.032 (0.088)	-10.087 (0.824)	1.229	25.661	75	469
25	NPL (logit prob.)	1.781 (0.000)	-3.364 (0.000)	8.287	46.136	1	1000
50	ML (Cons. Opt.)	5.009 (0.060)	-10.999 (0.149)	0.160	0.380	100	–
50	2-Step PML	4.967 (0.089)	-10.990 (0.203)	0.223	0.091	100	–
50	2-Step LS	5.016 (0.111)	-11.106 (0.343)	0.374	0.058	100	–
50	NPL (freq. prob.)	5.018 (0.071)	-10.243 (0.780)	1.087	30.148	86	384
50	NPL (logit prob.)	1.763 (0.000)	-3.386 (0.000)	8.274	69.820	1	998
100	ML (Cons. Opt.)	5.011 (0.046)	-10.982 (0.095)	0.107	0.821	100	–
100	2-Step PML	4.995 (0.060)	-11.011 (0.164)	0.176	0.164	100	–
100	2-Step LS	5.022 (0.077)	-11.090 (0.249)	0.275	0.059	100	–
100	NPL (freq. prob.)	5.024 (0.060)	-10.661 (0.650)	0.733	30.406	99	225
100	NPL (logit prob.)	1.775 (0.000)	-3.379 (0.000)	8.276	123.580	1	999
250	ML (Cons. Opt.)	5.003 (0.025)	-10.993 (0.057)	0.062	1.838	100	–
250	2-Step PML	4.9957 (0.034)	-11.000 (0.103)	0.108	0.377	100	–
250	2-Step LS	5.008 (0.040)	-11.025 (0.171)	0.176	0.060	100	–
250	NPL (freq. prob.)	5.010 (0.060)	-10.854 (0.650)	0.470	53.572	100	168
250	NPL (logit prob.)	1.774 (0.003)	-3.374 (0.000)	8.281	281.110	2	997

Standard deviations are reported in parentheses.

Table 3: Randomly Chosen Equilibrium in Each Market

T	Estimator	Estimates		RMSE	CPU Time Per Run (sec.)	# of Data Sets Converged	Avg. NPL Iter.
		α	β				
	Truth	5	−11	–	–	–	–
5	ML (Cons. Opt.)	5.027 (0.179)	−10.743 (0.585)	0.661	1.346	100	–
5	2-Step PML	3.068 (0.208)	−7.279 (0.512)	4.228	0.043	100	–
5	2-Step LS	2.918 (0.203)	−7.597 (0.654)	4.047	0.048	100	–
5	NPL (freq. prob.)	N/A (N/A)	N/A (N/A)	N/A	31.527	0	1000
5	NPL (logit prob.)	N/A (N/A)	N/A (N/A)	N/A	33.748	0	1000
10	ML (Cons. Opt.)	5.029 (0.126)	−10.816 (0.326)	0.394	0.641	100	–
10	2-Step PML	3.719 (0.165)	−8.535 (0.403)	2.812	0.042	100	–
10	2-Step LS	3.459 (0.164)	−8.499 (0.531)	2.990	0.049	100	–
10	NPL (freq. prob.)	N/A (N/A)	N/A (N/A)	N/A	35.756	0	1000
10	NPL (logit prob.)	N/A (N/A)	N/A (N/A)	N/A	37.786	0	1000
25	ML (Cons. Opt.)	5.018 (0.084)	−10.964 (0.166)	0.189	0.512	100	–
25	2-Step PML	4.302 (0.122)	−9.663 (0.268)	1.537	0.060	100	–
25	2-Step LS	3.959 (0.134)	−9.311 (0.354)	2.019	0.050	100	–
25	NPL (freq. prob.)	N/A (N/A)	N/A (N/A)	N/A	52.268	0	1000
25	NPL (logit prob.)	N/A (N/A)	N/A (N/A)	N/A	54.315	0	1000
50	ML (Cons. Opt.)	5.005 (0.056)	−11.007 (0.139)	0.150	0.669	100	–
50	2-Step PML	4.590 (0.099)	−10.280 (0.230)	0.865	0.093	100	–
50	2-Step LS	4.279 (0.109)	−9.895 (0.283)	1.354	0.052	100	–
50	NPL (freq. prob.)	N/A (N/A)	N/A (N/A)	N/A	82.390	0	1000
50	NPL (logit prob.)	N/A (N/A)	N/A (N/A)	N/A	84.415	0	1000
100	ML (Cons. Opt.)	5.006 (0.045)	−10.997 (0.092)	0.102	1.252	100	–
100	2-Step PML	4.773 (0.067)	−10.607 (0.165)	0.487	0.174	100	–
100	2-Step LS	4.533 (0.084)	−10.285 (0.200)	0.881	0.053	100	–
100	NPL (freq. prob.)	N/A (N/A)	N/A (N/A)	N/A	150.220	0	1000
100	NPL (logit prob.)	N/A (N/A)	N/A (N/A)	N/A	152.560	0	1000
250	ML (Cons. Opt.)	5.000 (0.028)	−10.999 (0.057)	0.063	2.512	100	–
250	2-Step PML	4.905 (0.043)	−10.828 (0.114)	0.231	0.410	100	–
250	2-Step LS	4.905 (0.051)	−10.624 (0.157)	0.472	0.054	100	–
250	NPL (freq. prob.)	N/A (N/A)	N/A (N/A)	N/A	351.990	0	1000
250	NPL (logit prob.)	N/A (N/A)	N/A (N/A)	N/A	354.470	0	1000

Standard deviations are reported in parentheses.

References

[1] Arcidiacono, P., and R. A. Miller (2011): "CCP Estimation of Dynamic Discrete Choice Models with Unobserved Heterogeneity," *Econometrica*, 79, 1823–1868.

[2] Aguirregabiria, V., and P. Mira (2007): "Sequential Estimation of Dynamic Discrete Games," *Econometrica*, 75, 1–53.

[3] Aitchison, J., and S. D. Silvey (1958): "Maximum Likelihood Estimation of Parameters subject to Restraints," *Annals of Mathematical Statistics*, 29, 813–828.

[4] Bajari, P., C. L. Benkard, and J. Levin (2007): "Estimating Dynamic Games of Imperfect Competition," *Econometrica*, 75, 1331–1370.

[5] Bajari, P., H. Hong, J. Krainer and D. Nekipelov (2010): "Estimating Static Models of Strategic Interactions," *Journal of Business and Economic Statistics*, 28, 469–482.

[6] Berry, S. (1992): "Estimation of a Model of Entry in the Airline Industry," *Econometrica*, 60, 889–917.

[7] Ciliberto, F., and E. Tamer (2009): "Market Structure and Multiple Equilibria in Airline Markets," *Econometrica*, 77, 1791–1828.

[8] Fudenberg, D., and J. Tirole (1991): *Game Theory.* Cambridge: MIT Press.

[9] Hotz, J., and R. A. Miller (1993): "Conditional Choice Probabilities and the Estimation of Dynamic Models," *Review of Economic Studies*, 60, 497–529.

[10] Judd, K., P. Renner, and K. Schmedders (2011): "Finding All Pure Strategy Equilibria in Dynamic and Static Games with Continuous Strategies," *Quantitative Economics*, forthcoming.

[11] Kohlberg, E., and J.-F. Mertens (1986): "On the Strategic Stability of Equilibria," *Econometrica*, 54, 1003–1038.

[12] Mertens, J.-F. (1989): "Stable Equilibria–a Reformulation: Part I. Definition and Basic Properties," *Mathematics of Operations Research*, 14, 575–625.

[13] Mertens, J.-F. (1991): "Stable Equilibria–a Reformulation: Part II. Discussion of the Definition, and Further Results," *Mathematics of Operations Research*, 16, 694–753.

[14] Misra, S., and P. Ellickson (2011): "Estimating Discrete Games," *Marketing Science*, forthcoming.

[15] Pakes, A., M. Ostrovsky, and S. Berry (2007): "Simple Estimators for the Parameters of Discrete Dynamic Games, with Entry/Exit Examples," *RAND Journal of Economics*, 38, 373–399.

[16] Pakes, A., J. Porter, K. Ho, and J. Ishii (2011): "Moment Inequalities and Their Applications," Working paper, Harvard University.

[17] Pesendorfer, M., and P. Schmidt-Dengler (2008): "Asymptotic Least Squares Estimators for Dynamic Games," *Review of Economic Studies*, 75, 901–928.

[18] Pesendorfer, M., and P. Schmidt-Dengler (2010): "Sequential Estimation of Dynamic Discrete Games: A Comment," *Econometrica*, 78, 833–842.

[19] Rust, J. (1987): "Optimal Replacement of GMC Bus Engines: An Empirical Model of Harold Zurcher," *Econometrica*, 55, 999–1033.

[20] Rust, J. (2008): Private communication.

[21] Seim, K. (2006): "An Empirical Model of Firm Entry with Endogenous Product-Type Choices," *RAND Journal of Economics*, 37, 619–640.

[22] Su, C.-L. and K. L. Judd (2011): "Constrained Optimization Approaches to Estimation of Structural Models," *Econometrica,* forthcoming.

[23] Sweeting, A. (2009): "The Strategic Timing of Radio Commercials: An Empirical Analysis Using Multiple Equilibria," *RAND Journal of Economics*, 40, 710–742.

[24] Silvey, S. D. (1975): *Statistical Inference*. Chapman & Hall, London.

[25] Vitorino, M. A. (2011): "Empirical Entry Games with Complementarities: An Application to the Shopping Center Industry", *Journal of Marketing Research*, forthcoming