Recursivity and the Estimation of Dynamic Games with Continuous Controls

João Granja
Giuseppe Forte

May 9, 2023

Click here for the latest version

Abstract

We revisit the estimation of dynamic games with continuous control variables, such as investments in R&D, quality, and capacity. We show how to use the recursive characterization of Markov Perfect Equilibria to develop estimators that exploit the structure of optimal policies. Our preferred estimator resembles an indirect inference estimator, albeit in a two-step procedure that is common in the estimation of dynamic games. We use Monte Carlo experiments based on an empirically-relevant model of investment in R&D to compare the performance of that estimator with alternatives. We find that our estimator outperforms the commonly-used inequality estimator of Bajari, Benkard, and Levin (2007) and a non-linear least squares estimator based on recursive equilibrium conditions.

First draft: December 4, 2022.
Granja: University College London, joao.granja@ucl.ac.uk.
Forte: University College London and Institute for Fiscal Studies, uctpgfo@ucl.ac.uk.

1 Introduction

Many questions of interest to Industrial Organization economists involve firm choices that have persistent effects on market conditions. Such choices include investments in research and development, the choice of productive capacity, and the choice of product characteristics. Many other examples can be given. Decisions of this type are inherently dynamic and are often taken in industries with few firms. Therefore, their study necessitates the use of dynamic oligopoly models. Furthermore, many of these choices, such as the ones above, are naturally modeled as continuous variables.

This paper revisits the estimation of dynamic oligopoly models with continuous controls. The estimation of such models was made feasible by the seminal contribution of Bajari et al. (2007), henceforth BBL. Here we make the observation that the main estimator proposed by BBL does not use the full structure of the model, in that it does not exploit the structure of equilibrium policies. Exploiting this structure should lead to efficiency gains. We propose estimators that do use the structure of equilibrium policies and conduct Monte Carlo exercises that compare their performance to different implementations of BBL.

Our Monte Carlo exercises are based on Hashmi and van Biesebroeck (2016) – henceforth HvB. HvB propose and estimate an equilibrium model of innovation in the automobile industry. They are interested in the equilibrium relationship between market structure and innovation. In their model, firms engage in R&D – measured by firms' patenting activity – to increase product quality. The cost of R&D depends on the number of patents a firm files for and on an investment cost shock. We base our simulation exercises on the Hashmi and van Biesebroeck (2016) model because it underpins an actual empirical application, and thus accurately represents models used in practice.

The first step in our preferred estimation routine consists of estimating policy functions and state transitions from the data. This step is similar to BBL and estimators of dynamic games with discrete controls, such as the ones proposed by Aguirregabiria and Mira (2007), Pakes, Ostrovsky, and Berry (2007), and Pesendorfer and Schmidt-Dengler (2008). We depart from BBL in the second step. For a given guess of the structural parameters, we use the estimated policy functions and state transitions to form and solve the maximization problem in the right-hand-side of firms' Bellman equations. We do so at the states observed in the data and randomly drawn investment cost shocks. We then project these predicted investment levels onto a space spanned by basis functions of state variables. Finally, we minimize a measure of the distance between the projections of predicted and observed investment levels onto that space. Therefore, our estimator combines elements of the two-step estimators that sprung from Hotz and Miller (1993) with Indirect Inference estimators à la Gourieroux, Monfort, and Renault (1993). We refer to this estimator as the Recursive Indirect Inference (Rec-II) estimator. We also consider a second estimator based on recursive equilibrium conditions. That estimator does not apply the Indirect Inference step; rather, it minimizes a nonlinear least squares objective. We refer to this estimator as the Recursive Nonliner Least Squares (Rec-NLLS) estimator.

The intuition behind estimators based on recursive equilibrium conditions is simple: under the maintained assumption that the estimated policies constitute an equilibrium, solving the right-hand-side of firms' Bellman equations must return the same policy. This intuition underpins the Rec-II and Rec-NLLS estimators.[1]

We find that the Rec-II estimator has desirable properties. In our Monte Carlo exercise, its finite-sample bias is small and the estimator is precise. We compare it to two implementations of BBL. The first one uses additive perturbations to the estimated policy function. The second one uses multiplicative deviations, and in HvB and recommended by Srisuma (2013). In contrast to the Rec-II estimator, we find that both implementations of BBL have very substantial finite sample bias. In fact, the estimates are orders of magnitude away from the true parameters. We relate these findings to the shape of the objective functions that define each estimator. While our estimator's objective function attains its minimum close to the truth and has large curvature, we find the BBL objective to be flat around the true parameters. We find that the Rec-NLLS estimator performs reasonably well, but less so than the Rec-II estimator; it is also more expensive to compute.

The Rec-II estimator enjoys at least one other benefit relative to BBL: it does not require the econometrician to choose policy deviations. This is an advantage, as the performance of the BBL estimator may very well depend on the deviations chosen by the analyst and the literature provides little guidance on how to choose deviations.[2] This concern is substantiated by the different per-

[1]Other estimators could be considered. For instance, one could minimize the distance between estimated and predicted conditional expectations of investment. This would be the continuous-control analog of the Pesendorfer and Schmidt-Dengler (2008) estimator. In results not reported here we find that this estimator performs similarly to the Rec-NLLS estimator.

[2]As noted above, some guidance is provided by Srisuma (2013). He suggests that multiplicative

formance of the two BBL alternatives we consider.

It would be remiss of us not to remind the reader that Bajari et al. (2007) do propose a second estimator based on solving for firms' optimal policies, though in a way very different from what we propose in this paper. Nevertheless, the empirical literature seems to have converged to using the estimator that BBL discuss at greater length, based on value function inequalities. A number of applications, including very recent ones, apply the inequality estimator. These include Ryan (2012), Hashmi and van Biesebroeck (2016), Fowlie, Reguant, and Ryan (2016), and Liu and Siebert (2022). An important paper that does use firms' first-order conditions to estimate a dynamic model is Jofre-Bonet and Pesendorfer (2003). In a dynamic auction model, they show that firms' first-order conditions and the observed distribution of bids identify the distribution of firms' costs. We show that similar ideas extend to the estimation of parameters determining firms' flow profits. Although we consider a model in which firms choose only investment, the ideas in this paper readily extend to models with entry and exit.

The paper most closely related to our is Srisuma (2013). Srisuma also observes that the BBL inequalities may fail to identify the structural parameters and proposes an estimator that makes use of agents' optimization problems in a two-step procedure. The main distinctions between our approach and Srisuma's are practical. Srisuma's estimator is based on minimizing a distance between the observed conditional distributions of agents' actions and the one implied by agents' (pseudo) maximization problems. This has practical drawbacks. The estimation of the implied distribution is done by simulation. A precise estimator may require the solution of many (static) optimization problems – many more than what we have to solve. Moreover, the implied objective is discontinuous in the structural parameters. These two features make the estimator potentially costly and difficult to compute for the models that practitioners take to data. Indeed, the Monte Carlo simulations in Srisuma (2013) are based on simple static models.

We therefore view our contributions as threefold. First, we show how to implement estimators based on recursive equilibrium conditions in an empirically relevant setting, including shocks to firms' marginal costs of investment. Second, we show that these estimators can substantially outperform the commonly-applied inequality-based estimator. Third, we compare the performance of dif-

deviations have more identifying power than additive ones. We implement both and find their performance to be equally poor.

ferent estimators based on recursive equilibrium conditions, including an analog of an estimator known to perform well in dynamic discrete games, and find support for an estimator based on Indirect Inference. That estimator is also easily-adaptable to games with continuous and discrete decisions, such as entry and exit. Along the way we also show that one can do away with value function simulation and can instead solve for these objects, thus eliminating simulation error.

The rest of the paper is organized as follows. In section 2 we discuss the Hashmi and van Biesebroeck (2016) model. In section 3 we discuss estimators based on recursive equilibrium conditions. In section 4 we provide a brief review of the main BBL estimator. In section 5 we discuss the results of our Monte Carlo exercises. Finally, in section 6 we offer concluding remarks.

2 The Economic Model

We base our comparison of estimators on the model of innovation in the automobile industry of Hashmi and van Biesebroeck (2016). As in all dynamic games following the Ericson and Pakes (1995) framework, there is a static part to the model and a dynamic one. In the static part, firms play a Nash-Bertrand pricing game, given demand and marginal cost functions. In the dynamic part, firms invest in R&D. Investments in R&D affect the quality of firms' products, which in turn affects demand and production costs. There is no entry or exit.[3]

The timing of the model is as follows. Firms start the period with given quality levels. They simultaneously set prices, which determine the quantity sold and flow profits. They then privately observe a shock to their investment cost. Next, firms simultaneously decide how much to invest and pay the associated costs. Firm investments affect the distribution of the quality of their products in the beginning of the following period. Firms' random quality levels realize before the start of the new period.

We discuss the static and dynamic parts of the model in turn.

[3]Entry and exit are not observed in the Hashmi and van Biesebroeck (2016) data, which is why those decisions are not included in their model. For our purposes, this simplifies the exposition and allows us to focus on the continuous control, which is our main interest.

2.1 Price Competition Given Demand and Costs

There are N single-product firms in the market, indexed by $j = 1, 2, \ldots, N$.[4] There is a measure M of consumers in the market, and consumers are indexed by i. If consumer i buys firm j's product, she obtains conditional indirect utility

$$u_{ij} = \alpha p_j + \xi_j + \varepsilon_{ij} \ ,$$

where α is the marginal utility of income, p_j is the price of good j, ξ_j is a quality index and ε_{ij} is a random shock that follows a Type 1 Extreme Value distribution. There is also an outside option, indexed by $j = 0$. Its non-random utility is normalized to zero, $u_{i0} = \varepsilon_{i0}$.

As is well-known, this specification of utility yields the market share function

$$s_j(\boldsymbol{p}; \boldsymbol{\xi}) = \frac{\exp(\alpha p_j + \xi_j)}{1 + \sum_k \exp(\alpha p_k + \xi_k)}, \quad j = 1, \ldots, N$$

where $\boldsymbol{p} = (p_1, \ldots, p_N)$ and $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_N)$. Prices are modeled as the outcome of a Bertrand game. Firm j solves

$$\max_{p_j} \ \pi_j(p_j; \boldsymbol{p}_{-j}, \boldsymbol{\xi}) := (p_j - \mu(\xi_j)) s_j(\boldsymbol{p}; \boldsymbol{\xi}) \ ,$$

where $\mu_j(\xi_j)$ is firm j's constant marginal cost of production, which depends on its quality level, and firm j takes $\boldsymbol{p}_{-j} = (p_1, \ldots, p_{j-1}, p_{j+1}, \ldots, p_N)$ as given. The marginal cost function is specified as

$$\mu(\xi_j) = \exp(\theta_{c1} + \theta_{c2}\xi_j) \ .$$

We know from Caplin and Nalebuff (1991) that this pricing game has a unique equilibrium. Moreover, the equilibrium price vector, $\boldsymbol{p}^*(\boldsymbol{\xi}) = (p_1^*(\boldsymbol{\xi}), \ldots, p_N^*(\boldsymbol{\xi}))$, must satisfy the system of first-order conditions:

$$\alpha(1 - s_j(\boldsymbol{p}^*; \boldsymbol{\xi}))(p_j^* - \mu_j(\xi_j)) + 1 = 0, \quad j = 1, \ldots, J \tag{1}$$

The equilibrium $\boldsymbol{p}^*(\boldsymbol{\xi})$ induces profits $\pi_j(\boldsymbol{\xi}) := [p_j^*(\boldsymbol{\xi}) - \mu(\xi_j)] s_j(\boldsymbol{p}^*(\boldsymbol{\xi}), \boldsymbol{\xi})$.

[4] Hashmi and van Biesebroeck (2016) estimate a demand model that accounts for the many products sold by automakers. When setting up their dynamic game of innovation, they (heuristically) aggregate that model to the firm level. This aggregation makes the dynamic model tractable. Here, as our focus is on methods for estimating the dynamic parameters of the model, we start from a model of single-product firms.

2.2 Investment in R&D

2.2.1 Quality Transitions

Firms invest to affect the quality of their product in the next period. Firms' qualities belong to the set $\Xi = \{\xi_{\min}, \xi_{\min}+\delta, \ldots, \xi_{\max}-\delta, \xi_{\max}\}$. If $\xi_{\min} < \xi < \xi_{\max}$, then ξ can decrease by δ, stay unchanged, or increase by δ. These transition probabilities are parameterized as

$$P(\xi'|\xi, x) = \begin{cases} \theta_{t1}[1 - \text{up}(\xi, x)] & \text{if} \quad \xi' = \xi - \delta \\ 1 - \theta_{t1} - \text{up}(\xi, x)(1 - 2\theta_{t1}) & \text{if} \quad \xi' = \xi \\ (1 - \theta_{t1})\text{up}(\xi, x) & \text{if} \quad \xi' = \xi + \delta \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where

$$\text{up}(\xi, x) = e^{-e^{-\theta_{t2}\log(x+1) - \theta_{t3}\xi - \theta_{t4}\xi^2}}.$$

In the expressions above x denotes investment in R&D and $\boldsymbol{\theta}_t = (\theta_{t1}, \theta_{t2}, \theta_{t3}, \theta_{t4})$ is a vector of parameters governing the transition probabilities. To respect the finiteness of Ξ, we adjust these probabilities at the maximum and minimum values of Ξ. When $\xi = \xi_{min}$ we set $P(\xi|\xi, x)$ equal to the sum of the first two cases in equation 2. When $\xi = \xi_{max}$ we set $P(\xi|\xi, x)$ equal to the sum of the second and third cases in equation 2.

The interpretation of this specification is that the quality of a product is subject to a positive and a negative shock. One can think of the negative shock as capturing improvements in the outside option. The positive shock captures successes in the R&D process. The negative shock occurs with probability θ_{t1}, whereas the positive shock occurs with probability $\text{up}(\xi_j, x)$. Quality falls when the product's quality experiences a negative shock without an offsetting positive shock. It grows when quality experiences a positive shock unmatched by a negative shock. In all other cases, quality remains the same. Positive and negative shocks are independent across firms, conditional on $\boldsymbol{\xi}$ and $\boldsymbol{x}$.[5]

The probability of an improvement in the outside option is independent of firms' qualities and investments. The reasoning is that the outside option is outside of the focal market, and the process driving its quality is unresponsive

[5]The independence of negative shocks across firms is at odds with its interpretation as capturing improvements in the outside option. We make this assumption for two reasons. First, it simplifies the exposition somewhat. Second, despite stating otherwise (p. 200), HvB seem to treat the negative shocks as independent at times (see their discussion of simulation in p. 201).

to the conditions of the market being analyzed. The probability of an R&D success, however, does depend on a firm's investment and quality, but not on its competitors'. In particular, if $\theta_{t2} > 0$ and $\theta_{t3} < 0$, the specification of $\text{up}(\xi_j, x)$ implies that the probability of success is increasing in own investment and decreasing in own quality. This last effect captures the notion that it is harder to improve on a high-quality product. The sign of θ_{t4} determines whether an increase in quality reduces the probability of success at an increasing or decreasing rate.

2.2.2 The Firm's Problem

Firms choose their investment to maximize their present-discounted stream of profits. They trade-off better future prospects for the quality of their product against the immediate cost of investment, $c(x, \nu)$. This cost depends on the level of investment x and on a privately observed investment cost shock, ν. The cost shock is assumed to be iid across firms and follows a standard normal distribution. We focus throughout on Markov Perfect Equilibria. Firms' state variables are the vector of qualities in the market, $\boldsymbol{\xi}$, and their privately observed investment cost shock ν. Firm behaviour depends only on the state variables. Denote firm j's policy function by $\sigma_j(\boldsymbol{\xi}, \nu)$ and let $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_N)$. Firm j's problem can then be recursively represented as

$$V_j(\xi_j, \boldsymbol{\xi}_{-j}, \nu) = \max_{x \in \mathbb{R}_+} \left\{ \pi(\xi_j, \boldsymbol{\xi}_{-j}) - c(x, \nu) + \beta \mathbb{E}_{\boldsymbol{\sigma}_{-j}} \left[V_j(\xi_j', \boldsymbol{\xi}_{-j}', \nu' | \xi_j, \boldsymbol{\xi}_{-j}, x) \right] \right\} \quad (3)$$

where

$$\mathbb{E}_{\boldsymbol{\sigma}_{-j}} \left[V_j(\xi_j', \boldsymbol{\xi}_{-j}', \nu') | \xi_j, \boldsymbol{\xi}_{-j}, x \right] = \sum_{\xi_j'} \sum_{\boldsymbol{\xi}_{-j}'} \int_{\nu} V_j(\xi_j', \boldsymbol{\xi}_{-j}', \nu') dF(\nu') P_{-j}(\boldsymbol{\xi}_{-j}' | \boldsymbol{\xi}; \boldsymbol{\sigma}_{-j}) P_j(\xi_j' | \xi_j, x) \quad (4)$$

and

$$P_{-j}(\boldsymbol{\xi}_{-j}' | \boldsymbol{\xi}; \boldsymbol{\sigma}_{-j}) = \prod_{k \neq j} P(\xi_k' | \boldsymbol{\xi}; \sigma_k) = \prod_{k \neq j} \int_{\nu_k} P(\xi_k' | \xi_k, \sigma_k(\boldsymbol{\xi}, \nu_k)) dF(\nu_k) \,. \quad (5)$$

In this last expression, the terms $P(\xi_k' | \xi_k, \sigma_k(\boldsymbol{\xi}, \nu_k))$ are derived from the quality transition model discussed in section 2.2.1.

We proceed as in Doraszelski and Pakes (2007). Define

$$EV_j(\boldsymbol{\xi}) := \int_{\nu_j} V_j(\xi_j, \boldsymbol{\xi}_{-j}, \nu) dF(\nu)$$

and

$$W(\xi'_j|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j}) := \sum_{\xi'_{-j}} EV(\xi'_j, \boldsymbol{\xi}'_{-j}) P_{-j}(\boldsymbol{\xi}'_{-j}|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j}) \ .$$

$EV(\boldsymbol{\xi})$ is the expected present-discounted stream of profits when the vector of qualities is $\boldsymbol{\xi}$. The object $W(\xi'_j|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j})$ is the expected value of landing on quality ξ'_j starting from the vector of qualities $\boldsymbol{\xi}$. With these definitions we can write

$$\mathbb{E}_{\boldsymbol{\sigma}_{-j}}\left[V_j(\xi'_j, \boldsymbol{\xi}'_{-j}, \nu')|\xi_j, \boldsymbol{\xi}_{-j}, x\right] = \sum_{\xi'_j} W(\xi'_j|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j}) P_j(\xi'_j|\xi_j, x)$$

The first-order condition of the maximization problem on the right hand side of the Bellman equation is then

$$\frac{\partial c(x,\nu)}{\partial x} = \beta \sum_{\xi'_j} W(\xi'_j|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j}) \frac{\partial P_j(\xi'_j|\xi_j, x)}{\partial x} \tag{6}$$

At the optimum level of investment, given $\boldsymbol{\xi}$ and ν, the firm equates the marginal cost of investment to its marginal benefit. The marginal cost of investment is an exogenous object. The marginal benefit of investment depends on how investment changes the distribution over quality levels in the following period. It also depends on the value of starting the following period with different levels of quality, ξ'_j. Moreover, these future values take into account the current quality-state. For instance, the gain in present-discounted profits from an increase in quality may be smaller if a firm's competitors all have substantially lower quality, relative to a case in which their qualities are similar to the focal firm's.

It is worth noting for future reference that condition 6 can also be written as

$$\frac{\partial c(x,\nu)}{\partial x} = \beta \frac{\partial \mathrm{up}(\xi_j, x)}{\partial x} \times \Delta W(\boldsymbol{\xi};\boldsymbol{\sigma}_{-j}) \tag{7}$$

where $\Delta W(\boldsymbol{\xi};\boldsymbol{\sigma}_{-j})$ is given by

$$(1-\theta_{t1})[W(\xi_j+\delta|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j}) - W(\xi_j|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j})] + \theta_{t1}[W(\xi_j|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j}) - W(\xi_j-\delta|\boldsymbol{\xi};\boldsymbol{\sigma}_{-j})] \ .$$

Fix a $\boldsymbol{\xi}$ and suppose that there exists ν such that $\sigma(\boldsymbol{\xi}, \nu) > 0$. Assume further that $\partial_x c(x, \nu) > 0$ for all (x, ν).[6] Because $\sigma(\boldsymbol{\xi}, \nu) > 0$, the first-order condition must hold with equality – i.e., equation 7 must hold. Therefore, it must be the case that $\Delta W(\boldsymbol{\xi}; \boldsymbol{\sigma}_{-j}) > 0$. This has two implications. First, this condition is a restriction on the equilibrium value function and by itself provides restrictions on the structural parameters. Second, this implies that the firm's problem is globally concave at that $\boldsymbol{\xi}$ for *all* ν. We will return to these observations when we discuss our estimator.

In our numerical exercise, we parameterize the investment cost function as

$$c(x, \nu) = \theta_{x1} x + \theta_{x2} x^2 + \theta_{x3} \nu x \ .$$

This is as in HvB, except that they also include a cubic term.

2.3 Equilibrium and Computation

We focus on symmetric Markov Perfect Equilibria, i.e., $\sigma_j(\boldsymbol{\xi}, \nu) = \sigma(\boldsymbol{\xi}, \nu)$ for $j = 1, \ldots, N$. As shown by equations (4) and (5), the expectation in the Bellman equation (3) depends on $\sigma(\cdot, \cdot)$. Therefore, one can think of equation (3) as defining an operator that maps policy and value functions into themselves: $T : \Sigma \times \mathcal{V} \to \Sigma \times \mathcal{V}$, where Σ denotes the set of feasible policy functions and $\mathcal{V}$ the set of feasible value functions. This operators returns a firm's best-response to its competitors all playing a given strategy σ (given V). A strategy profile $(\sigma, \ldots, \sigma)$ is a symmetric Markov Perfect Equilibrium if and only if (σ, V_σ) is a fixed-point of T, where V_σ is the expected present-discounted stream of profits when all firms play the strategy σ.

This representation underpins the method we employ to solve for Markov Perfect Equilibria. We start with a guess for value and policy functions. With the policy functions we can compute the implied transition probabilities for firms' qualities. Having transition probabilities and the guess for the value function, we can compute the terms $W(\xi_j' | \boldsymbol{\xi}; \boldsymbol{\sigma}_{-j})$. This allows us to solve firms' first-order conditions, which yields new guesses for policy and value functions. We iterate on these steps until both value and policy functions converge. This is, of course, simply the Pakes and McGuire (1994) algorithm. As in Pakes and McGuire (1994), note that the symmetry assumption implies that the state $\boldsymbol{\xi} =$

[6] This fails to hold in our specification due to the full support of ν. However, in our parameterization the probability that this condition fails is small. We retain the normality assumption for convenience.

$(\xi_j, \boldsymbol{\xi}_{-j})$ is equivalent to $\boldsymbol{\xi} = (\xi_j, \pi(\boldsymbol{\xi}_{-j}))$ for any permutation π of competitors' qualities. Therefore, we need not compute value and policy functions at each $\boldsymbol{\xi}$, but only at a member of the equivalence class. In particular, we retain only $\boldsymbol{\xi}$'s such that competitors' qualities are increasing in their index, i.e., $\xi_2 \leq \ldots \leq \xi_N$. As shown in Pakes and McGuire (1994), this implies that the state space grows in the number of firms not exponentially, but rather as a polynomial of order $|\Xi|$. In our numerical examples below, this reduces the cardinality of the state space from just under 760,000 to 45,900.[7]

3 Estimators Based on Recursive Equilibrium Conditions

In this section we discuss estimators based on the Bellman Equation (3). Firstly, by exploiting (3), these estimators make use of the structure of optimal policy functions, i.e., the fact that optimal policy functions must solve the right-hand side of the Bellman equation. The algorithm proposed in BBL does not make use of this feature. Secondly, the estimators we propose free the researcher from having to commit to a choice of policy deviations, which in practice affect the performance of the BBL estimator. Our estimator will therefore have econometric and practical advantages relative to BBL.[8]

Suppose that we have estimates of the ex-ante value function $EV(\boldsymbol{\xi}; \theta) := \int V(\boldsymbol{\xi}, \nu; \sigma, \sigma, \theta) dF(\nu)$. Denote these estimates by $\widehat{EV}(\boldsymbol{\xi}; \theta)$. These estimates could be obtained by forward simulation, as in Bajari et al. (2007). We discuss an alternative below, but for now let us focus on the main point of differ-

[7]One can alternatively view equation 3 as an operator $\tilde{T} : \Sigma \to \Sigma$ defined in two steps. First, compute the value function implied by σ, i.e.,

$$V(\boldsymbol{\xi}, \nu; \sigma) = \pi(\boldsymbol{\xi}) - c(\sigma(\boldsymbol{\xi}, \nu), \nu) + \beta \mathbb{E}_\sigma \left[V(\boldsymbol{\xi}', \nu') | \boldsymbol{\xi}, \boldsymbol{\sigma}(\boldsymbol{\xi}, \nu) \right] .$$

Next, solve the right-hand-side of equation 3 to obtain $\tilde{T}\sigma$. A symmetric strategy profile $(\sigma, \ldots, \sigma)$ is a symmetric MPE if and only if it is a fixed point of $\tilde{T}$. One can thus solve for an MPE by iterating on these two steps.

[8]Another important point of comparison is computational cost. As will become clear as we discuss our estimator, there is a trade-off. On the one hand, estimator requires simulating or solving for value functions only once, whereas the Bajari et al. (2007) estimator requires doing so for all pairs (σ', σ) that enter the objective function. On the other hand, our estimator requires solving at least as many static optimization problems as observations in the data. In the Monte Carlo experiments we perform in section 5 this computational trade-off is resolved in BBL's favor. The runtimes reported in table 3 are not entirely comparable, though. Our BBL implementation fully exploits the parallelism of the estimation problem. Our current implementation of the recursive estimators does not; doing so is work in progress.

ence between our estimator and BBL. Suppose that we also have estimates of $P(\xi_j' \mid \xi_j, x)$ and $\mathbb{P}(\xi_j' \mid \boldsymbol{\xi}) = \int P(\xi_j' \mid \xi_j, \sigma(\boldsymbol{\xi}, \nu))dF(\nu)$. Let $\hat{\Phi} := (\widehat{EV(\theta)}, \hat{P}, \hat{\mathbb{P}})$ Using $\hat{\Phi}$, we can setup and solve the right-hand side of the Bellman Equation (3):

$$\max_{x\in\mathbb{R}_+} \left\{ \pi(\xi_j, \boldsymbol{\xi}_{-j}) - c(x, \nu; \theta) + \beta \mathbb{E}_{\hat{P},\hat{\mathbb{P}}} \left[\widehat{EV}(\xi_j', \boldsymbol{\xi}'_{-j}; \theta) | \xi_j, \boldsymbol{\xi}_{-j}, x \right] \right\} . \tag{8}$$

Solving problem (8) at some $(\boldsymbol{\xi}, \nu)$ yields a new predicted level of investment at that state, which depends on the parameters θ. We denote these predicted investment levels by $T_\theta(\boldsymbol{\xi}, \nu; \hat{\Phi})$.

We can consider different estimators of θ that use the predicted levels of investment $T_\theta(\boldsymbol{\xi}, \nu; \hat{\Phi})$. Our preferred approach is akin to Indirect Inference (Gourieroux et al., 1993).[9] We estimate $\mathbb{E}[\sigma(\boldsymbol{\xi}, \nu) \mid \boldsymbol{\xi}]$ by projecting observed levels of investment onto a set of basis functions $\{\Psi_k\}_{k=1}^B$:

$$x_i = \sum_{k=1}^{B} \gamma_k \Psi_k(\boldsymbol{\xi_i}) + \eta_i \tag{9}$$

Let $\hat{\boldsymbol{\gamma}} := \left[\hat{\gamma}_1, \ldots, \hat{\gamma}_B, \hat{S}_\eta\right]$, where $\hat{S}_\eta = (N - B)^{-1} \sum_{i=1}^N (x_i - \sum_{k=1}^B \hat{\gamma}_k \Psi_k(\boldsymbol{\xi_i}))$ is an estimate of the standard deviation of η_i. We solve the problem (8) for each $\boldsymbol{\xi}$ observed in the data and a randomly drawn $\nu \sim N(0, 1)$, which yields $\{T_\theta(\boldsymbol{\xi}_i, \nu_i; \hat{\Phi})\}_{i=1}^N$.[10] Next, we estimate

$$T_\theta(\boldsymbol{\xi}_i, \nu_i; \hat{\Phi}) = \sum_{k=1}^{B} \lambda_k \Psi_k(\boldsymbol{\xi_i}) + \zeta_i \ , \tag{10}$$

which yields $\hat{\boldsymbol{\lambda}}(\theta; \hat{\Phi}) := \left[\hat{\lambda}_1(\theta; \hat{\Phi}), \ldots, \hat{\lambda}_B(\theta; \hat{\Phi}), \hat{S}_\zeta(\theta; \hat{\Phi})\right]$. Finally, for some positive-definite weight matrix $\boldsymbol{W}$, we define our estimator to be[11]

$$\hat{\theta}_{II} := \arg\min_\theta Q(\theta; \boldsymbol{W}, \hat{\Phi}) = \left[\hat{\boldsymbol{\lambda}}(\theta; \hat{\Phi}) - \hat{\boldsymbol{\gamma}}\right]' \boldsymbol{W} \left[\hat{\boldsymbol{\lambda}}(\theta; \hat{\Phi}) - \hat{\boldsymbol{\gamma}}\right] . \tag{11}$$

[9]We do not claim to be the first to consider applying Indirect Inference to dynamic oligopoly games. Rather, our contribution is to show that its performance in the case of continuous actions can be dramatically superior to that of widely-adopted alternatives. Collard-Wexler (2013) represents a notable example of Indirect Inference applied to a dynamic oligopoly model with discrete actions; Li (2010) also seems to suggest applying Indirect Inference to discrete oligopoly models.

[10]We can also consider multiple draws of ν per observation, which can potentially increase the precision of the estimator.

[11]As always, having the analytical gradient of the objective (11) significantly accelerates the calculation of the optimum. In appendix B we perform the relevant calculations.

We refer to this estimator as the Recursive Indirect Inference estimator, or Rec-II for short.

The intuition for the estimator above is the following.[12] If the policy observed in the data, $\hat{\sigma}$, is a Markov Perfect Equilibrium, it must satisfy the recursive equilibrium conditions (3). Therefore, if we solve the right-hand side of the Bellman Equation using transitions and value functions implied by $\hat{\sigma}$, we must obtain $\hat{\sigma}$ back. One could try to estimate the structural parameters by matching $T_\theta(\cdot;\hat{\Phi})$ and $\hat{\sigma}$. Srisuma (2013) proposes matching the conditional distribution of investment observed in the data and that implied by $T_\theta(\cdot;\hat{\Phi})$, which is a closely related idea. Unfortunately, computing the conditional distribution of investment implied by $T_\theta(\cdot;\hat{\Phi})$ is computationally very costly. We instead match features of those distributions. We find that the resulting estimators are sufficiently cheap to compute and perform better than existing alternatives.

One could use the predicted levels of investment $T_\theta(\boldsymbol{\xi},\nu;\hat{\Phi})$ in other ways. For instance, one can consider a nonlinear least squares estimator:[13]

$$\hat{\theta}_{NLLS} := \arg\min_\theta \sum_{i=1}^{N} \Big(x_i - \mathbb{E}_\nu[T_\theta(\boldsymbol{\xi}_i,\nu;\hat{\Phi}) \mid \boldsymbol{\xi}]\Big)^2 \tag{12}$$

where we compute $\mathbb{E}_\nu[T_\theta(\boldsymbol{\xi}_i,\nu;\hat{\Phi}) \mid \boldsymbol{\xi}]$ by quadrature. Yet another alternative would be to minimize a distance between the estimated expectation of investment conditional on $\boldsymbol{\xi}$ and its model-predicted equivalent. That would amount to substituting the fitted values of equation (9) for the x_i in equation (12). This is be the continuous-control analog of Pesendorfer and Schmidt-Dengler (2008).

Our preferred estimator is $\hat{\theta}_{II}$. There are two reasons for that. First, it is computationally cheaper than $\hat{\theta}_{NLLS}$, as it requires only one evaluation of $T_\theta(\boldsymbol{\xi}_i,\nu_i;\hat{\Phi})$ – i.e., the numerical solution of a static maximization problem – per observation, rather than as many evaluations as quadrature nodes. Second, $\hat{\theta}_{II}$ is readily-adaptable to models with entry and exit decisions. In such models, in addition to (9), we would estimate descriptive models for the entry and exit decisions. We would then compute optimal entry-exit decisions implied by structural parameters θ and re-run the aforementioned models using the simulated decisions as the dependent variable. Finally, we would estimate our structural parameters combining the estimates of (9) and the estimates of the descriptive entry-exit models.

[12] As discussed in the introduction, this intuition underpins not only the Rec-II estimator but all estimators based on the recursive equilibrium conditions.

[13] We thank Dennis Kristensen for the suggestion.

3.1 Estimating EV

Let $EV(\boldsymbol{\xi}) = \int V(\boldsymbol{\xi}, \nu) dF(\nu)$ denote the ex-ante value function and let $\boldsymbol{EV} = [EV(\boldsymbol{\xi}_1), \ldots, EV(\boldsymbol{\xi}_{|\Xi|})]$ be a vector stacking the value function at each state in the state space.[14] We show in appendix A that $\boldsymbol{EV}$ satisfies [15]

$$\boldsymbol{EV} = \boldsymbol{W}(\boldsymbol{\pi}, \sigma)[1, -\boldsymbol{\theta}_{\boldsymbol{x}}']' + \beta \boldsymbol{M}(\boldsymbol{P}) \boldsymbol{EV} \tag{13}$$

or

$$\boldsymbol{EV}(\sigma, \boldsymbol{P}; \boldsymbol{\pi}, \beta, \boldsymbol{\theta}_x) = [I - \beta \boldsymbol{M}(\boldsymbol{P})]^{-1} \boldsymbol{W}(\boldsymbol{\pi}, \sigma)[1, -\boldsymbol{\theta}_{\boldsymbol{x}}']' \ , \tag{14}$$

where the $\boldsymbol{\xi}$-th row of $\boldsymbol{W}(\boldsymbol{\pi}, \sigma)$, which we denote $\boldsymbol{W}(\boldsymbol{\xi}; \boldsymbol{\pi}, \sigma)$, is a vector of the expected flow-profit covariates at state $\boldsymbol{\xi}$ and $\boldsymbol{M}(\boldsymbol{P})$ is the transition matrix implied by the policy function σ. Specifically,

$$\boldsymbol{W}(\boldsymbol{\xi}; \boldsymbol{\pi}, \sigma) = \begin{bmatrix} \pi(\boldsymbol{\xi}) & \mathbb{E}[\sigma(\boldsymbol{\xi}, \nu) | \boldsymbol{\xi}] & \mathbb{E}[\sigma(\boldsymbol{\xi}, \nu)^2 | \boldsymbol{\xi}] & \mathbb{E}[\sigma(\boldsymbol{\xi}, \nu)\nu | \boldsymbol{\xi}] \end{bmatrix} \ , \tag{15}$$

and

$$\boldsymbol{M}(\boldsymbol{P}) = \begin{bmatrix} \mathbb{P}(\boldsymbol{\xi}_1 | \boldsymbol{\xi}_1) & \ldots & \mathbb{P}(\boldsymbol{\xi}_{|\Xi|} | \boldsymbol{\xi}_1) \\ \vdots & \ddots & \vdots \\ \mathbb{P}(\boldsymbol{\xi}_1 | \boldsymbol{\xi}_{|\Xi|}) & \ldots & \mathbb{P}(\boldsymbol{\xi}_{|\Xi|} | \boldsymbol{\xi}_{|\Xi|}) \end{bmatrix} \tag{16}$$

where

$$\mathbb{P}(\boldsymbol{\xi}' | \boldsymbol{\xi}) = \prod_{j=1}^{N} \mathbb{P}(\xi_j' \mid \boldsymbol{\xi}) = \prod_{j=1}^{N} \int \mathbb{P}(\xi_j' \mid \xi_j, \sigma(\boldsymbol{\xi}, \nu)) dF(\nu) \ . \tag{17}$$

Observe that although we write $\boldsymbol{W}(\boldsymbol{\pi}, \sigma)$, $\boldsymbol{W}$ does not depend on the policy function per se, but only on conditional expectations of functions of investment and the covariance between the policy function and the investment cost shock.

Equations (14) to (17) imply that we can estimate EV by estimating the conditional expectations in (15) and $\mathbb{P}(\xi_j' \mid \boldsymbol{\xi})$. The objects $\mathbb{E}[\sigma(\boldsymbol{\xi}, \nu) \mid \boldsymbol{\xi}]$, $\mathbb{E}[\sigma(\boldsymbol{\xi}, \nu)^2 \mid \boldsymbol{\xi}]$, and $P(\xi_j' \mid \boldsymbol{\xi})$ are readily identified from the data. The BBL argument for identification of the policy function implies, in the case in which $\sigma(\boldsymbol{\xi}, \nu)$ is decreasing in ν, that

$$\sigma(\boldsymbol{\xi}, \nu) = F_X^{-1}(1 - \Phi(\nu) \mid \boldsymbol{\xi}) \ , \tag{18}$$

where $F_X(x \mid \boldsymbol{\xi})$ is the distribution of investment conditional on $\boldsymbol{\xi}$, which is identified, and Φ is the standard normal cumulative distribution function. That

[14] More precisely, at each element of the set of equivalence classes induced by the symmetry restrictions.

[15] Related calculations appear e.g. in Jofre-Bonet and Pesendorfer (2003).

is, the policy function is identified as the quantiles of the conditional distribution of investment. Using this, $\mathbb{E}[\sigma(\boldsymbol{\xi},\nu)\nu \mid \boldsymbol{\xi}]$ can be approximated by Gauss Hermite quadrature as $\pi^{-1/2}\sum_{i=1}^{N}\omega_i F_X^{-1}(1-\Phi(\sqrt{2}\nu_i) \mid \boldsymbol{\xi})\sqrt{2}\nu_i$, for judiciously chosen weights ω_i and nodes ν_i. Therefore, all that is needed to estimate $\mathbb{E}[\sigma(\boldsymbol{\xi},\nu)\nu \mid \boldsymbol{\xi}]$ is identified in the data. Note however that in principle the above argument requires estimating investment quantiles at each element of $\boldsymbol{\xi}$ – a daunting task if many such elements are seldom observed. As an alternative, we propose approximating $F_X^{-1}(1-\Phi(x) \mid \boldsymbol{\xi})$ by quantile regression of investment on features of $\boldsymbol{\xi}$.

4 A Brief Review of BBL

In this section we briefly review the main estimator proposed by Bajari et al. (2007), which is the main point of comparison for the estimator we propose in Section 3.

The expected present-discounted stream of profits of a firm when it plays a strategy σ_j and all its competitors play the strategy σ is given by

$$V(\boldsymbol{\xi},\nu;\sigma_j,\sigma) = \mathbb{E}\left\{\sum_{t=0}^{\infty}\beta^t\left[\pi(\xi_{jt},\boldsymbol{\xi}_{-j,t}) - c(\sigma_j(\boldsymbol{\xi}_t,\nu_{jt}),\nu_{jt})\right]\middle|\boldsymbol{\xi}_0=\boldsymbol{\xi},\nu_{j0}=\nu\right\}$$

where, letting $P(\xi,x)$ denote the probability mass function defined in Equation (2), $\xi_{j,t+1}\sim P(\xi_{j,t},\sigma_j(\boldsymbol{\xi}_t,\nu_{jt}))$ and for $k\neq j$, $\xi_{k,t+1}\sim P(\xi_{k,t},\sigma(\boldsymbol{\xi}_t,\nu_{kt}))$.

By definition, a symmetric strategy profile $(\sigma,\ldots,\sigma)$ is a Markov Perfect Equilibrium if

$$V(\boldsymbol{\xi},\nu;\sigma,\sigma) \geq V(\boldsymbol{\xi},\nu;\sigma',\sigma) \quad \forall\, \boldsymbol{\xi},\nu,\sigma' \tag{19}$$

Bajari et al. (2007) base their estimator on the equilibrium conditions (19). Define

$$g(\boldsymbol{\xi},\nu,\sigma';\sigma,\theta) := V(\boldsymbol{\xi},\nu;\sigma,\sigma,\theta) - V(\boldsymbol{\xi},\nu;\sigma',\sigma,\theta)$$

where we have made the dependence on the structural parameters θ explicit. Let H be a distribution over the space of tuples of the form $(\boldsymbol{\xi},\nu,\sigma')$. Define

$$Q(\theta,\sigma) := \int\left(\min\left\{g(\boldsymbol{\xi},\nu,\sigma';\sigma,\theta),0\right\}\right)^2 dH(\boldsymbol{\xi},\nu,\sigma') \tag{20}$$

Let $\mathcal{E}(\theta)$ be the set of MPE when the parameters of the model are given by θ and let θ_0 denote the true parameter value. If $\sigma\in\mathcal{E}(\theta_0)$, the equilibrium conditions above imply that $Q(\theta_0, \sigma) = 0$.

Assumption 1 (ID)**.** For any $\theta, \theta' \in \Theta$, $\mathcal{E}(\theta) \cap \mathcal{E}(\theta') = \emptyset$.

If assumption 1 holds, $\sigma \in \mathcal{E}(\theta_0)$, $\theta' \neq \theta_0$, and H is chosen judiciously, then $Q(\theta', \sigma) > 0$: assumption 1 implies that $\sigma \notin \mathcal{E}(\theta')$, so that $g(\boldsymbol{\xi}, \nu, \sigma'; \sigma, \theta') < 0$ for some $(\boldsymbol{\xi}, \nu, \sigma', \sigma)$.[16]

Bajari et al. (2007) propose estimating the structural parameters of the model by minimizing a sample analog of (20). In particular, for some set of $(\boldsymbol{\xi}, \nu, \sigma')$ tuples, indexed by $i = 1, \dots, n_I$, they propose minimizing

$$\hat{Q}(\theta, \hat{\sigma}) := \frac{1}{n_I} \sum_{i=1}^{n_I} \left(\min \left\{ g(\boldsymbol{\xi}_i, \nu_i, \sigma_i'; \hat{\sigma}, \theta), 0 \right\} \right)^2$$

where $\hat{\sigma}$ is the strategy profile estimated in a first step. Evaluating this objective requires estimates of the value function V under the estimated policies and under the deviations. Bajari et al. (2007) propose obtaining these estimates by forward simulation. As they note, linearity of flow profits significantly reduces the computational burden of forward simulation. Linearity implies that there exists a function $\Lambda(\boldsymbol{\xi}, \nu; \sigma', \sigma)$ such that $V(\boldsymbol{\xi}, \nu; \sigma', \sigma, \theta) = \Lambda(\boldsymbol{\xi}, \nu; \sigma', \sigma)\theta$. Therefore, the forward simulation needs to be performed only once for each pair (σ', σ) to obtain $\Lambda(\boldsymbol{\xi}, \nu; \sigma', \sigma)$. Linearity of flow profits does apply to the Hashmi and van Biesebroeck (2016) model.

5 Monte Carlo Simulations

We now discuss the performance of the estimators discussed in Sections 3 and 4 for the model presented in Section 2.[17] We consider a rather data-rich environment, with data from 100 separate markets recorded for 40 periods. 5 firms compete in each market.[18] Firms have marginal cost parameters θ_{c1} =

[16] To be more precise, we require that $g(\boldsymbol{\xi}, \nu, \sigma'; \sigma, \theta') < 0$ on a set of positive H-measure for all $\theta' \neq \theta_0$. We can attach this condition to our definition of MPE. Given a measure μ on the set of tuples $(\boldsymbol{\xi}, \nu, \sigma')$, say that $(\sigma, \dots, \sigma)$ is a symmetric MPE if $g(\boldsymbol{\xi}, \nu, \sigma'; \sigma, \theta_0) < 0$ with zero μ-measure. Then choose H such that μ is absolutely continuous with respect to H. If $\theta' \neq \theta_0$, assumption 1 implies that $g(\boldsymbol{\xi}, \nu, \sigma'; \sigma, \theta') < 0$ with positive μ-measure. This implies that $g(\boldsymbol{\xi}, \nu, \sigma'; \sigma, \theta') < 0$ with positive H-measure, otherwise absolute continuity of μ with respect to H would be violated.

[17] While we eventually intend to consider multiple parameterisations of the model, for the moment we only discuss one such parameterisation.

[18] For comparison, Hashmi and van Biesebroeck (2016) aggregate data to have a single market with 14 firms for the 1982-2006 period. Ryan (2012) collected a total of 517 market-year pairs

$2.47, \theta_{c2} = 0$ – i.e. constant marginal cost $mc = \exp(2.47)$; unobserved product quality goes from -1.4 to 1.4 in increments of .2. There is a substantial probability of quality downgrade shocks ($\theta_{t1} = 0.547$), and the probability of a quality upgrading shock is increasing in own investment ($\theta_{t2} = 0.062$) but decreasing (at a decreasing rate) in own quality level ($\theta_{t3}, \theta_{t4} < 0$). Investment costs are convex ($\theta_{x1}, \theta_{x2} > 0$) in investment and increasing in the 'shock' term ($\theta_{x3} > 0$). We outline all parameters in Table 1.

We compare the performance of four estimators:

1. The II estimator proposed in Section 3, Equation 11;
2. A BBL estimator that uses multiplicative shifts of the candidate policy function to compute deviations: $\sigma'(\boldsymbol{\xi}, \nu) = \iota\sigma(\boldsymbol{\xi}, \nu)$ for $\iota \in \{.90, .95, 1.05, 1.10\}$ (Hashmi and van Biesebroeck (2016)'s estimator);
3. A BBL estimator computing $\sigma'(\boldsymbol{\xi}, \nu) = \sum_{k=1}^{B} \tilde{\gamma}_k \Psi_k(\boldsymbol{\xi_i})$, where $\tilde{\boldsymbol{\gamma}} \sim \mathrm{N}(\hat{\gamma}, \hat{\boldsymbol{\Sigma}}_\gamma)$ and $\hat{\gamma}$, $\hat{\boldsymbol{\Sigma}}_\gamma$ are estimated from the data (BBL estimator);
4. The NLLS estimator defined in Equation 12.

Table 2 describes the performance of the four estimators. It lists true parameter values, along with estimate average and confidence intervals for the four estimators over 125 simulations.

Because the BBL and HvB estimators are defined by inequalities, it is in principle possible for them to return parameter interval estimates rather than point estimates.[19] We allow for this possibility and present average lower and upper bound for each estimated set.[20] In practice, we find these estimators always return point estimates.

In order to present confidence intervals that are sensible for both point- and set-identified parameters, we display the shortest intervals containing 95 percent of

(an unbalanced panel of 27 markets over 19 years), with the number of firms in a market-year ranging from 1 to 20. We repeat the estimator comparison for a data structure akin to the one in Ryan (2012) in Appendix C.

[19] Aguirregabiria, Collard-Wexler, and Ryan (2021): "[...] in most applications of the BBL method, the relatively small set of alternative CCPs selected by the researcher does not provide enough moment inequalities to achieve point identification such that the BBL method provides set estimation of the structural parameters."

[20] Because each inequality are linear in θ, the entire list of inequalities defines a polyhedron. We first check whether such polyhedron is non-empty. If it is not, we compute parameter sets as the projections of the polyhedron on each parameter's axis. If it is, we minimise the BBL objective.

estimated intervals/points, as defined for example in Manski and Tamer (2002).

Of the four estimators, the II estimator evidently performs the best: averages are close to the true parameters, with small confidence intervals. BBL estimators and the NLLS estimator display substantial bias and very large confidence intervals. Figures 1, 2, 3, and 4 present the distribution of parameter estimates for each algorithm; vertical dotted lines represent true parameter values. II estimates are correctly centered around the true values, displaying minor bias and little skewness in the distribution of the estimates. On the other hand, HvB and BBL estimators are centered around wrong values, displaying a bias that can be orders of magnitude larger than the parameter to be estimated. The finite sample bias of the NLLS estimator is smaller than that of the BBL implementations, but it is still substantial compared to that of the II estimator. It is interesting to note that all estimators but II present substantial skewness in the estimates of θ_{x3}, in our experience the more complicated parameter to pin down.

Table 1: Exercise Parameters

Parameter	Value
Data Structure	
Number of Firms	5
Number of Markets	100
Number of Periods	40
Number of Households	1.00E+08
Model	
State Space	-1.4:0.2:1.4
Discount Factor	0.925
Marginal Utility of Income	-0.222
Marginal Cost Parameters	[2.47 0.0]
Investment Cost Parameters	[2.625 1.624 0.5096]
Transition Probability Parameters	[0.547 0.072 -0.884 -0.285]
II Estimator	
Number of II Simulations	5
BBL Estimators	
BBL: Number of Inequalities	1000
BBL: Number of Simulated Paths	500
BBL: Simulation Horizon	80
BBL: Vector of Scalar Deviations	[0.9 0.95 1.05 1.1]

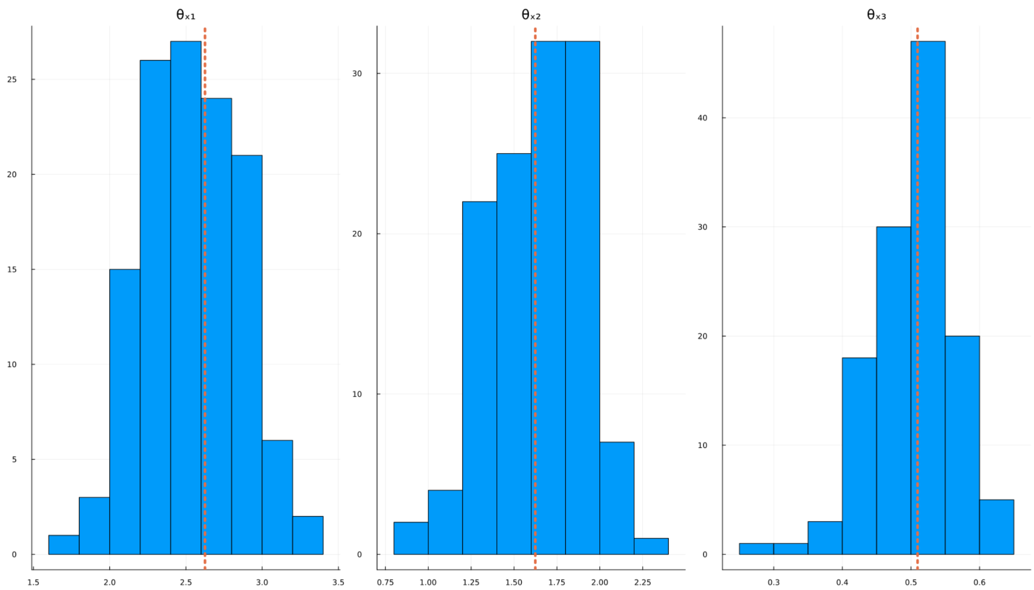
This table reports parameters used in our Monte Carlo exercise. The parameters labeled BBL relate to our implementation of the BBL estimator. Number of inequalities is the number of equilibrium conditions included in the BBL objective. Number of simulated paths is the number of paths used in estimating value functions via simulation. Simulation horizon is how far the paths are simulated. The vector of scalar deviations defines the HvB deviations discussed in the text. The following set of parameters relate to the HvB model discussed above. Number of II simulations is the number of ν draws per observation in the data used in the II estimator. The number of markets and periods pertain to the size of the dataset used in the Monte Carlo simulations. State space is the set of possible qualities for a firm; the notation $-1.4 : 0.2 : 1.4$ indicates that quality can be as low as -1.4 and as high as 1.4, growing in increments of 0.2.

Table 2: Summary of Parameter Estimates

	Value	II	BBL	HvB	NLLS
θ_{x1}	2.625	2.533	[38.589, 38.589]	[0.0, 0.0]	3.007
		(1.944, 3.197)	(0.0, 119.943)	(0.0, 0.0)	(1.659, 5.451)
θ_{x2}	1.624	1.642	[46.225, 46.225]	[3.613, 3.613]	1.538
		(1.143, 2.18)	(0.0, 72.004)	(2.274, 4.769)	(0.713, 2.33)
θ_{x3}	0.5096	0.503	[4.858, 4.858]	[2.926, 2.926]	1.214
		(0.379, 0.609)	(0.0, 17.633)	(0.0, 15.06)	(0.0, 3.784)

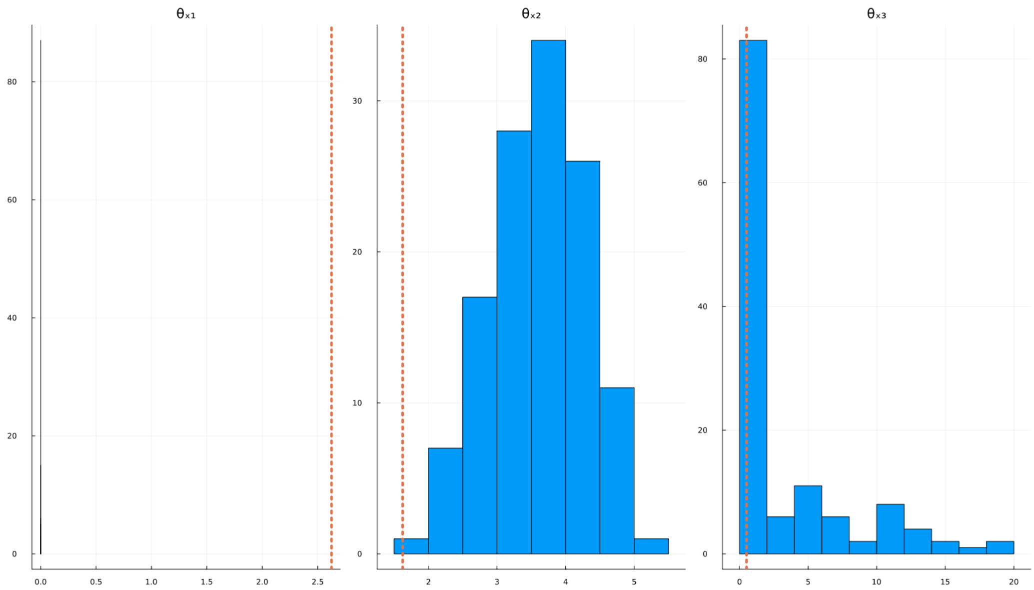
This table summarizes the results of our Monte Carlo experiment. The first column shows the value of the parameters of the investment cost function in the data generating process. Each subsequent column shows the mean and shortest 95% interval for estimates across Monte Carlo replications. The column labeled "II" shows the results of the estimator we propose in this paper. The column labeled "BBL" shows the results of the BBL estimator with random deviations. The column labeled "HvB" shows the results of the BBL estimator with HvB deviations. The column labeled "NLLS" shows the results of the estimator defined by the program **??**.

Figure 1: II Estimator Parameter Estimates



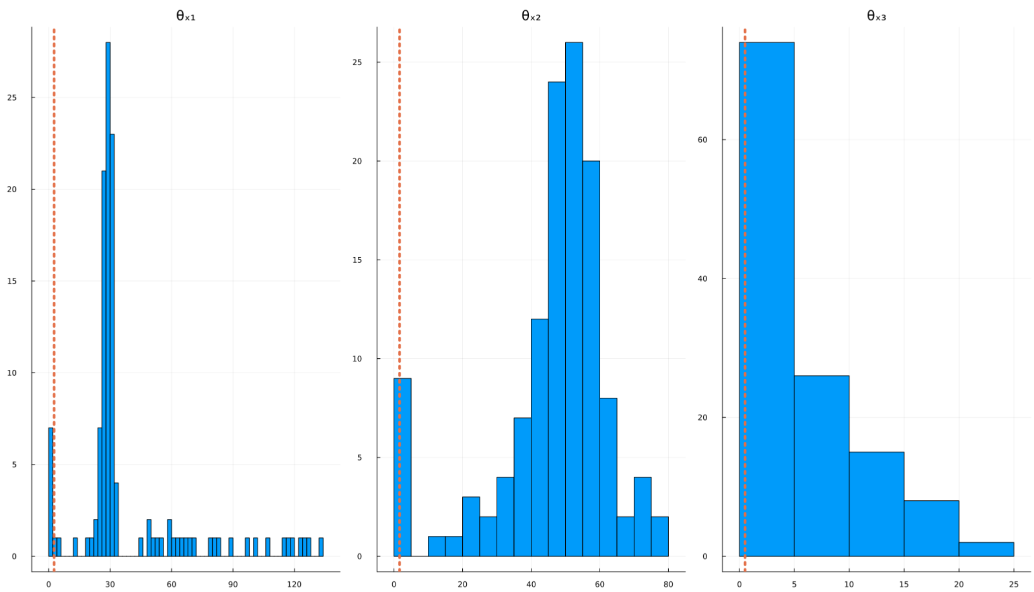
This figure plots the distribution of parameter estimates obtained using the indirect inference estimator defined in Equation 11 over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

Figure 2: HvB Estimator Parameter Estimates



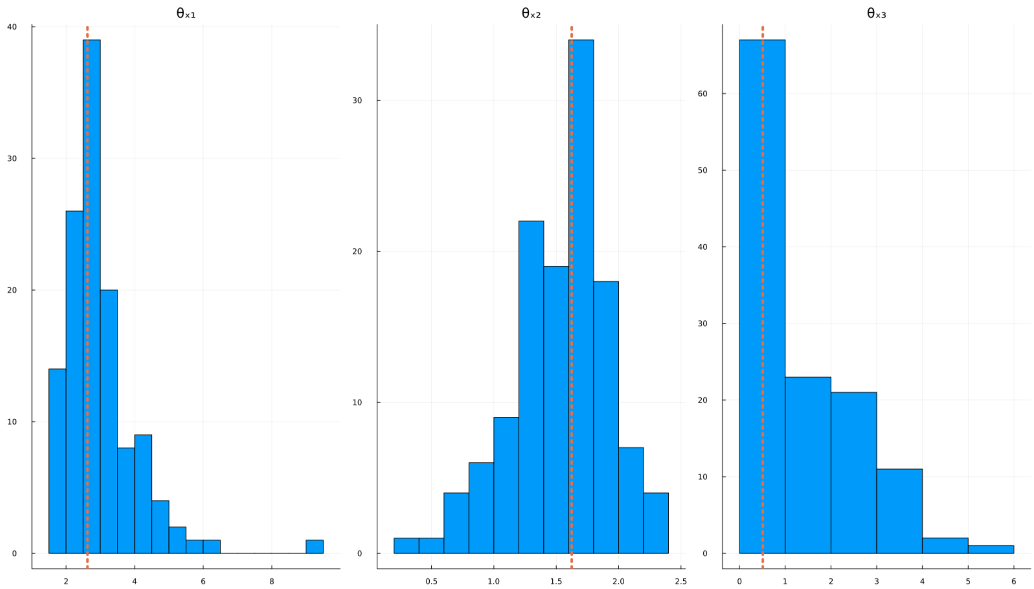
This figure plots the distribution of parameter estimates obtained using the BBL estimator with HvB deviations over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

Figure 3: BBL Estimator Parameter Estimates



This figure plots the distribution of parameter estimates obtained using the BBL estimator with random deviations over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

Figure 4: NLLS Parameter Estimates



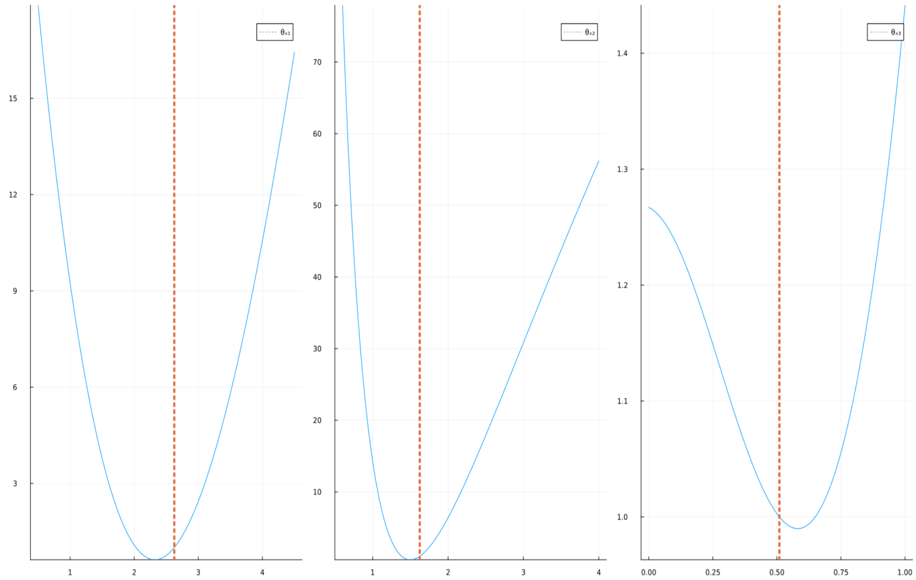
This figure plots the distribution of parameter estimates obtained using the estimator defined by the problem **??** over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

The relative performance of the different estimators is a consequence of the differences in the objective functions that define them. We illustrate this by analysing the shape of each objective function in a neighbourhood of the true parameters for a particular simulated dataset. To do so, we plot two-dimensional slices of the objective function by varying one parameter at a time while holding the others fixed at their true values. To render the shape of different objective functions comparable, we normalise objective values on the parameter grid by dividing each by the objective value at the true parameter.
Figures 5 to 8 display the results. Vertical lines represent true parameter values.

Figure 5 shows that the objective function of the II estimator features pronounced local convexity around each true parameter, with minimal distance between the local objective minimum and the objective value at the true parameter. This is reflected in the good performance of the estimator. The picture is very different for the HvB and BBL estimators: they are approximately flat around the true parameter, and zooming in reveals that they display no convexity at all for the considered grid – they are almost linear around the correct parameter values. This is consistent with the large bias displayed in Figures 2 and 3: the objective is convex around parameter values that are wrong by orders of magnitude.

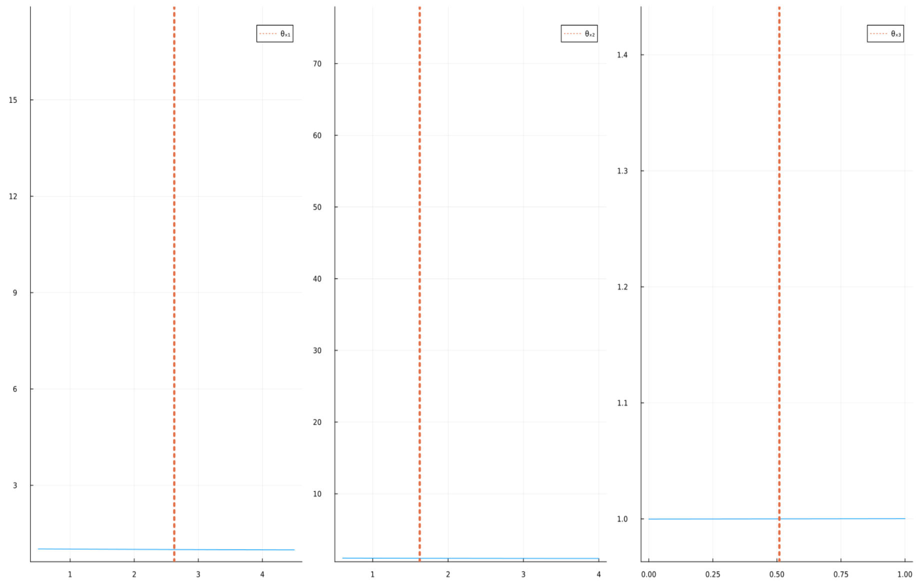
Finally, objective plots for the NLLS estimator are informative about its imprecision. Objective function minima are not far from the minima at the true parameter value (lower bias) for θ_{x1}, θ_{x2}, but the NLLS objective function is much less convex around local minima than the II objective function is. On the other hand, the objective function is evidently not centered around the true parameter objective value for θ_{x3}, which again speaks to the complications arising in precisely estimating the parameter.

Figure 5: II Estimator Objective



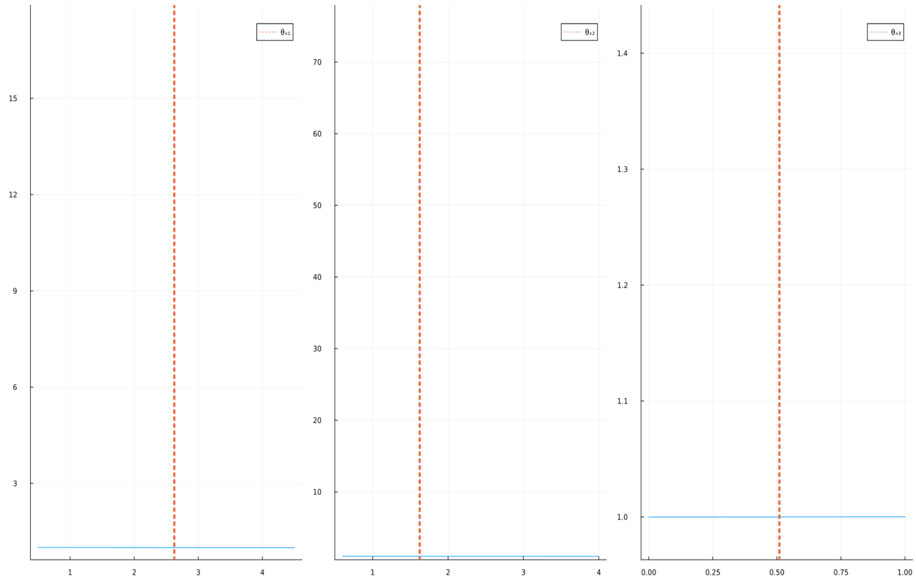
This figure plots the value of the II estimator objective function varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

Figure 6: HvB Estimator Objective



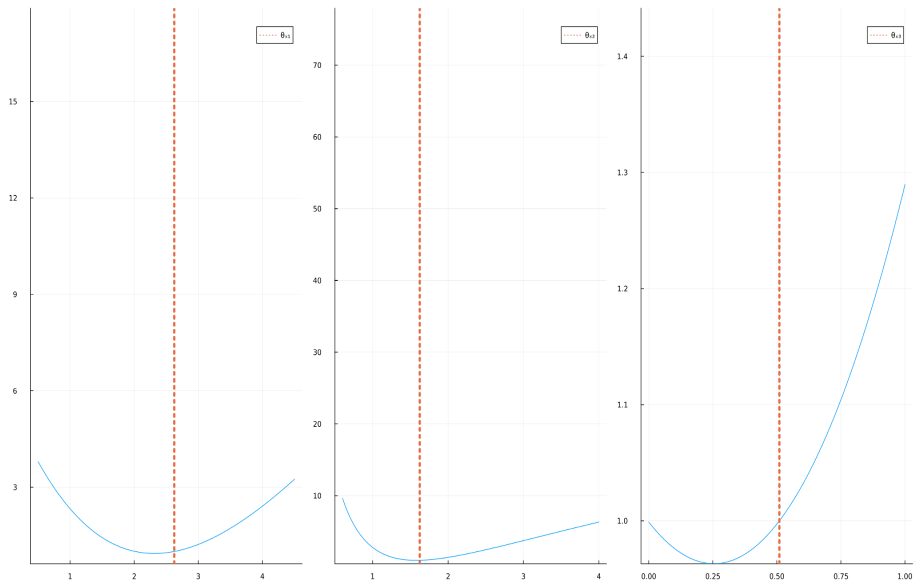
This figure plots the value of the objective function of the BBL estimator with HvB deviations varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

Figure 7: BBL Estimator Objective



This figure plots the value of the objective function of the BBL estimator with random deviations varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

Figure 8: NLLS Estimator Objective



This figure plots the value of the objective function of the estimator defined by program **??** varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

Table 3: Exercise Timings

Call	Average Time	Average Allocation
Overall		
Preliminaries	30.7s	36.8GiB
Simulation	124ms	26.2MiB
Step 1: Estimating Transition Parameters	132ms	2.66MiB
Computing $\mathbb{E}$VF Covariates	196ms	4.43MiB
Computing Projection Objects	19.1ms	21.3MiB
Step 2: II Estimator	55.9s	23.4GiB
Step 2: NLLS Estimator	226s	86.3GiB
Step 2: BBL Policy Covariates	15.1ms	14.4MiB
Step 2: BBL Estimator	1.69s	15.3MiB
Simulation	1.63s	2.60MiB
Optimisation	57.5ms	12.7MiB
Step 2: HvB Estimator	1.59s	12.5MiB
Simulation	1.58s	1.82MiB
Optimisation	10.7ms	10.6MiB

This table reports the computation time and memory allocation associated with each step in the computation of the four estimators discussed above. Note that the memory allocation reported here is not the amount of memory required to compute these estimators, but rather the total size of objects that are written to memory throughout the course of estimation. "Preliminaries" include the calculation of descriptors of flow profits, objects describing the state space of the game, and solving for a MPE. "Simulation" is the simulation of the data. "Computing $\mathbb{E}$VF Covariates" refers to the calculation of $\boldsymbol{EV}(\cdot)$ in Equation (14). "Computing Projection Objects" refers to the calculation of objects used in the implementation of the II estimator that need to be computed only once. The other items refer to the different estimators.

We conclude this section with an overview of the computational costs of each estimator. Table 3 displays run time and memory allocation for the different steps of our simulation, averaged over 125 runs.[21] As expected Step 1 – shared by all the algorithms considered – is fast, as it only entails estimating transition parameters by maximum likelihood and policy function parameters by linear regression.

On the other hand, there is substantial variation in the time and memory costs of various Step 2 algorithms. BBL-type estimators are the fastest, with both the HvB and BBL estimator taking less than 2 seconds between simula-

[21]'Preliminaries' are objects that only need to be computed once to set up multiple simulations.

tion and optimisation.[22] The II estimator is substantially slower: taking approximately 56s to return an estimate. Estimation time will be increasing in the number of datasets generated for indirect inference, which we set to 5. Finally, the NLLS estimator takes the most time of all, clocking at approximately 4 minutes (with very substantial dispersion, not documented in the table). The reason for this substantial cost is that evaluation of its objective and gradient requires solving a large number of static optimisation problems, whereas the gradient of the II estimator is readily computed analytically (see Appendix B).

6 Conclusion

We have revisited the estimation of dynamic games with continuous controls. We note that the commonly applied inequality estimator of Bajari et al. (2007) does not fully exploit the structure of optimal policies and propose an estimator that does so. Our estimator combines two-step methods that are common in the estimation of dynamic models with indirect inference ideas. We conduct a Monte Carlo experiment based on an empirically-relevant model and find that the estimator we propose significantly outperforms available alternatives in terms of precision. Bajari et al. (2007) themselves propose an estimator that does use the structure of optimal policies. However, the empirical literature has converged to applying exclusively their inequality estimator. We hope that by providing clear implementation details and documenting the substantial advantages of fully exploiting the structure of the model, our contribution will steer the literature towards doing so whenever feasible.

[22] In earlier drafts, we had found BBL-type estimators to be much slower. Substantial code optimization exponentially cut computation time.

References

Aguirregabiria, V., Collard-Wexler, A., & Ryan, S. P. (2021). Dynamic games in empirical industrial organization. In *Handbook of industrial organization* (Vol. 4, pp. 225–343).

Aguirregabiria, V., & Mira, P. (2007). Sequential estimation of dynamic discrete games. *Econometrica*, *75*, 1-53.

Bajari, P., Benkard, C. L., & Levin, J. (2007). Estimating dynamic models of imperfect competition. *Econometrica*, *75*, 1331-1370.

Caplin, A., & Nalebuff, B. (1991). Aggregation and imperfect competition: On the existence of equilibrium. *Econometrica*, *59*, 25.

Collard-Wexler, A. (2013). Demand fluctuations in the ready-mix concrete industry. *Econometrica*, *81*(3), 1003–1037.

Doraszelski, U., & Pakes, A. (2007). A framework for applied dynamic analysis in io. In M. P. R. Armstrong (Ed.), .

Ericson, R., & Pakes, A. (1995). Markov-perfect industry dynamics: A framework for empirical work. *The Review of Economic Studies*, *62*, 53-82.

Fowlie, M., Reguant, M., & Ryan, S. P. (2016). Market-based emissions regulation and industry dynamics. *Journal of Political Economy*.

Gourieroux, C., Monfort, A., & Renault, E. (1993). Indirect inference. *Journal of Applied Econometrics*, *8*, S85-S118.

Hashmi, A. R., & van Biesebroeck, J. (2016). The relationship between market structure and innovation in industry equilibrium: A case study of the global automobile industry. *Review of Economics and Statistics*, *98*, 192-208.

Hotz, V. J., & Miller, R. A. (1993). Conditional choice probabilities and the estimation of dynamic models. *The Review of Economic Studies*, *60*, 497-529.

Jofre-Bonet, M., & Pesendorfer, M. (2003). Estimation of a dynamic auction game. *Econometrica*, *71*(5), 1443–1489.

Li, T. (2010, July). Indirect inference in structural econometric models. *Journal of Econometrics*, *157*(1), 120–128. doi: 10.1016/j.jeconom.2009.10.027

Liu, A. H., & Siebert, R. B. (2022, 1). The competitive effects of declining entry costs over time: Evidence from the static random access memory market. *International Journal of Industrial Organization*, *80*.

Manski, C. F., & Tamer, E. (2002). Inference on regressions with interval data on a regressor or outcome. *Econometrica*, *70*(2), 519–546.

Pakes, A., & McGuire, P. (1994). Computing markov-perfect nash equilibria :

Numerical implications of a dynamic differentiated product model. *The RAND Journal of Economics*, *25*, 555-589.

Pakes, A., Ostrovsky, M., & Berry, S. (2007). Simple estimators for the parameters of discrete dynamic games (with entry/exit examples). *RAND Journal of Economics*, *38*, 373-399.

Pesendorfer, M., & Schmidt-Dengler, P. (2008). Asymptotic least squares estimators for dynamic games. *The Review of Economic Studies*, *75*(3), 901–928.

Ryan, S. P. (2012). The costs of environmental regulation in a concentrated industry. *Econometrica*, *80*, 1019-1061.

Srisuma, S. (2013). Minimum distance estimators for dynamic games. *Quantitative Economics*, *4*(3), 549–583.

Appendices

Appendix A Characterizing EV

This section derives equation (14). We start from equations (3) to (5), repeated here for the reader's convenience:

$$V_j(\xi_j, \boldsymbol{\xi}_{-j}, \nu) = \max_{x \in \mathbb{R}_+} \left\{ \pi(\xi_j, \boldsymbol{\xi}_{-j}) - c(x, \nu) + \beta \mathbb{E}_{\boldsymbol{\sigma}_{-j}} \left[V_j(\xi'_j, \boldsymbol{\xi}'_{-j}, \nu' | \xi_j, \boldsymbol{\xi}_{-j}, x) \right] \right\} \quad (3)$$

where

$$\mathbb{E}_{\boldsymbol{\sigma}_{-j}} \left[V_j(\xi'_j, \boldsymbol{\xi}'_{-j}, \nu') | \xi_j, \boldsymbol{\xi}_{-j}, x \right] = \sum_{\xi'_j} \sum_{\boldsymbol{\xi}'_{-j}} \int_{\nu} V_j(\xi'_j, \boldsymbol{\xi}'_{-j}, \nu') dF(\nu') P_{-j}(\boldsymbol{\xi}'_{-j} | \boldsymbol{\xi}; \boldsymbol{\sigma}_{-j}) P_j(\xi'_j | \xi_j, x) \quad (4)$$

and

$$P_{-j}(\boldsymbol{\xi}'_{-j} | \boldsymbol{\xi}; \boldsymbol{\sigma}_{-j}) = \prod_{k \neq j} P(\xi'_k | \boldsymbol{\xi}; \sigma_k) = \prod_{k \neq j} \int_{\nu_k} P(\xi'_k | \xi_k, \sigma_k(\boldsymbol{\xi}, \nu_k)) dF(\nu_k) \,. \quad (5)$$

Plugging in the equilibrium policy into the right-hand side of (3) and using (4), (5), and the definition of $EV(\boldsymbol{\xi})$, we have

$$V(\boldsymbol{\xi}, \nu) = \pi(\boldsymbol{\xi}) - c(\sigma(\boldsymbol{\xi}, \nu); \boldsymbol{\theta}_x) + \beta \sum_{\boldsymbol{\xi}'} EV(\boldsymbol{\xi}') \prod_{k \neq j} \mathbb{P}(\xi'_k | \boldsymbol{\xi}) \mathbb{P}(\xi'_j \mid \xi_j, \sigma(\boldsymbol{\xi}, \nu)) \,.$$

Integrating both sides of this equation yields

$$\begin{aligned} EV(\boldsymbol{\xi}) &= \pi(\boldsymbol{\xi}) - \int c(\sigma(\boldsymbol{\xi}, \nu); \boldsymbol{\theta}_x) dF(\nu) + \beta \sum_{\boldsymbol{\xi}'} EV(\boldsymbol{\xi}') \prod_{k \neq j} \mathbb{P}(\xi'_k | \boldsymbol{\xi}) \int \mathbb{P}(\xi'_j \mid \xi_j, \sigma(\boldsymbol{\xi}, \nu)) dF(\nu) \\ &= \pi(\boldsymbol{\xi}) - \boldsymbol{\theta}_x \cdot \int w_x(\sigma(\boldsymbol{\xi}, \nu), \nu) dF(\nu) + \beta \sum_{\boldsymbol{\xi}'} EV(\boldsymbol{\xi}') \prod_{k=1}^{N} \mathbb{P}(\xi'_k | \boldsymbol{\xi}) \,, \end{aligned}$$

where $w_x(x, \nu) = \begin{bmatrix} x & x^2 & \nu x \end{bmatrix}$ is the vector of cost covariates, and the second equality uses the linearity of the cost of investment and the definition of $\mathbb{P}(\xi'_j \mid \boldsymbol{\xi}) = \int \mathbb{P}(\xi'_j \mid \xi_j, \sigma(\boldsymbol{\xi}, \nu)) dF(\nu)$. Therefore,

$$EV(\boldsymbol{\xi}) = \boldsymbol{W}(\boldsymbol{\xi}; \boldsymbol{\pi}, \sigma) \cdot [1, -\boldsymbol{\theta}'_{\boldsymbol{x}}]' + \beta M(\boldsymbol{\xi}, \boldsymbol{P}) \boldsymbol{EV}$$

where $\boldsymbol{W}(\boldsymbol{\xi};\boldsymbol{\pi},\sigma)$ is the expectation of flow-profit covariates given $\boldsymbol{\xi}$, as defined in equation (15), and $M(\boldsymbol{\xi},\boldsymbol{P})$ is the $\boldsymbol{\xi}$-th row of the matrix $\boldsymbol{M}(\boldsymbol{P})$ defined in equation (16).

Stacking the last equation for all $\boldsymbol{\xi}$ and solving for $\boldsymbol{EV}$ gives equation (14). To see that $I-\beta\boldsymbol{M}(\boldsymbol{P})$ is indeed invertible, assume it is not. Then there exists $x\in\mathbb{R}^{|\Xi|}$ such that $[I-\beta\boldsymbol{M}(\boldsymbol{P})]x=0$, or $x=\beta\boldsymbol{M}x$. This implies $\|x\|_\infty=\|\beta\boldsymbol{M}x\|_\infty$. However, letting $(\boldsymbol{M}x)_i$ denote the i-th coordinate of $\boldsymbol{M}x$, we have

$$|(\boldsymbol{M}x)_i|=\left|\sum_{j=1}^{|\Xi|}M_{ij}x_j\right|\leq\sum_{j=1}^{|\Xi|}M_{ij}|x_j|\leq\|x\|_\infty\sum_{j=1}^{|\Xi|}M_{ij}=\|x\|_\infty\ ,$$

where we have used that $\boldsymbol{M}$ is a stochastic matrix. Therefore $\|x\|_\infty=\|\beta\boldsymbol{M}x\|_\infty=\beta\|\boldsymbol{M}x\|_\infty\leq\beta\|x\|_\infty$, a contradiction.

Appendix B The Gradient of $Q(\theta)$

We have

$$\nabla Q(\theta)=2\underset{1\times(k+1)}{[\hat{\boldsymbol{\lambda}}(\theta)-\hat{\gamma}]'}\underset{(k+1)\times(k+1)}{W}\underset{(k+1)\times J}{D_\theta\hat{\boldsymbol{\lambda}}(\theta)}$$

where $D_\theta\hat{\boldsymbol{\lambda}}(\theta)$ is the derivative (i.e., the Jacobian) of $\hat{\boldsymbol{\lambda}}(\theta)$ and we have made the dimensions explicit for clarity. The symbols k and J represent, respectively, the number of covariates in the empirical policy function and the number of structural parameters to be estimated. These objects have dimension $(k+1)$ rather than k because we include the estimated standard deviation in our objective function.

From the above, all that is left to calculate is $D_\theta\hat{\boldsymbol{\lambda}}(\theta)$. Remember that $\hat{\boldsymbol{\lambda}}(\theta)=(\hat{\lambda}_1(\theta),\ldots,\hat{\lambda}_k(\theta),S_\varsigma(\theta))$. Therefore

$$\underset{(k+1)\times J}{D_\theta\hat{\boldsymbol{\lambda}}(\theta)}=\begin{pmatrix}\underset{k\times J}{D_\theta\hat{\lambda}(\theta)}\\ \underset{1\times J}{\nabla_\theta S_\varsigma(\theta)}\end{pmatrix}$$

The object $\hat{\lambda}(\theta)$ is an OLS estimate, and thus satisfies

$$(X'X)\hat{\lambda}(\theta)=X'T_\theta(\hat{\sigma})$$

where X is the matrix of features

$$X = \begin{pmatrix} \psi_1(\xi_1) & \dots & \psi_k(\xi_1) \\ \vdots & \ddots & \vdots \\ \psi_1(\xi_N) & \dots & \psi_k(\xi_N) \end{pmatrix}$$

and[23]

$$T_\theta(\hat{\sigma}) = \begin{pmatrix} T_\theta(\hat{\sigma})(\xi_1, \nu_1) \\ \vdots \\ T_\theta(\hat{\sigma})(\xi_N, \nu_N) \end{pmatrix},$$

where $T_\theta(\hat{\sigma})(\xi_i, \nu_i)$ is the optimal level of investment when the parameters are θ, future behavior is given by $\hat{\sigma}$, and the state is (ξ_i, ν_i). Therefore,

$$(X'X)D_\theta\hat{\lambda}(\theta) = X'D_\theta T_\theta(\hat{\sigma}) \tag{21}$$

where

$$D_\theta T_\theta(\hat{\sigma}) = \begin{pmatrix} \nabla_\theta T_\theta(\hat{\sigma})(\xi_1, \nu_i) \\ \vdots \\ \nabla_\theta T_\theta(\hat{\sigma})(\xi_N, \nu_N) \end{pmatrix}$$

By the Implicit Function Theorem, the gradients in this matrix are given by

$$\nabla_\theta T_\theta(\hat{\sigma})(\xi_i, \nu_i) = -\left[\frac{\partial f}{\partial x}(x^*, \xi_i, \nu_i; \theta, \hat{\sigma})\right]^{-1} \nabla_\theta f(x^*, \xi_i, \nu_i; \theta, , \hat{\sigma})$$

where $x^* = T_\theta(\hat{\sigma})(\xi, \nu)$ and $f(x, \xi, \nu; \theta, \hat{\sigma})$ is the investment first-order condition. We can then solve for $D_\theta\hat{\lambda}$ from equation 21.

Next, we need $\nabla_\theta S_\varsigma(\theta)$. We have

$$S(\theta) = \left\{\frac{1}{n-k}\sum_{i=1}^{n}[T_\theta(\hat{\sigma})(\xi_i, \nu_i) - x(\xi_i)'\hat{\lambda}(\theta)]^2\right\}^{\frac{1}{2}}$$

where $x(\xi_i) := (\psi_1(\xi_i) \dots \psi_k(\xi_i))'$. Therefore,

$$\begin{aligned} \nabla_\theta S_\varsigma(\theta) &= \frac{1}{2}\{\cdot\}^{-\frac{1}{2}} \times \frac{2}{n-k}\sum_{i=1}^{n}[T_\theta(\hat{\sigma})(\xi_i, \nu_i) - x(\xi_i)'\hat{\lambda}(\theta)]\big\{\nabla_\theta T_\theta(\hat{\sigma})(\xi_i, \nu_i) - \nabla_\theta[x(\xi_i)'\hat{\lambda}(\theta)]\big\} \\ &= \frac{1}{S_\varsigma(\theta)(n-k)}\sum_{i=1}^{n}[T_\theta(\hat{\sigma})(\xi_i, \nu_i) - x(\xi_i)'\hat{\lambda}(\theta)]\big\{\nabla_\theta T_\theta(\hat{\sigma})(\xi_i, \nu_i) - \nabla_\theta[x(\xi_i)'\hat{\lambda}(\theta)]\big\} \end{aligned}$$

[23] We draw the ν_i shocks once and store them in memory so that all parameter values use the same ν_i shocks.

The first gradient in these expressions has already been characterized. The second gradient is

$$\nabla_\theta[x(\xi_i)'\hat{\lambda}(\theta)] = x(\xi_i)' D_\theta \hat{\lambda}(\theta)$$

and $D_\theta \hat{\lambda}(\theta)$ has just been characterized.

Appendix C Simulating data as in Ryan (2012)

In this section we compare performance of the estimators presented in Section 5 for simulated datasets with sample size in line with Ryan (2012). We consider data for 27 separate markets recorded for 20 periods. 5 firms compete in each market.[24]

Firms have marginal cost parameters $\theta_{c1} = 2.47, \theta_{c2} = 0$ – i.e. constant marginal cost $mc = \exp(2.47)$; unobserved product quality goes from -1.4 to 1.4 in increments of .2. There is a substantial probability of quality downgrade shocks ($\theta_{t1} = 0.547$), and the probability of a quality upgrading shock is increasing in own investment ($\theta_{t2} = 0.062$) but decreasing (at a decreasing rate) in own quality level ($\theta_{t3}, \theta_{t4} < 0$). Investment costs are convex ($\theta_{x1}, \theta_{x2} > 0$) in investment and increasing in the 'shock' term ($\theta_{x3} > 0$). We outline all parameters in Table 4.

We compare the performance of the four estimators presented in Section 5. Table 2 lists true parameter values along with estimate average and standard deviation for the four estimators over 125 simulations. Of the four estimators, the II estimator evidently performs the best, though with more finite sample bias and larger uncertainty than in Section 5. The NLLS estimator and, in particular, BBL estimators display substantial bias and larger standard deviation than the II estimator. Figures 9, 10, 11, and 12 present the distribution of parameter estimates for each algorithm; vertical dotted lines represent true parameter values. II estimates are correctly centered around the true values and present low skewness. HvB and BBL estimators are centered around wrong values, displaying a bias that can be orders of magnitude larger than the parameter to be estimated. The finite sample bias of the NLLS estimator is smaller than that of the BBL implementations, but it is still substantial.

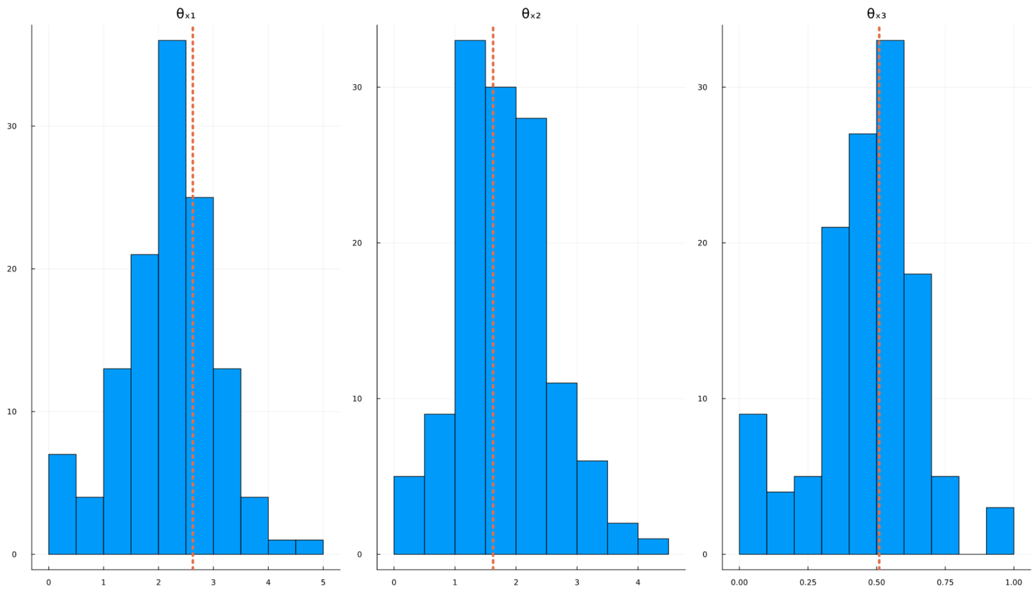
[24] Ryan (2012) collected a total of 517 observations (an unbalanced panel of 27 markets over 19 years), with the number of firms per observation ranging from 1 to 20 (mean 4.7).

Table 4: Exercise Parameters

Parameter	Value
Data Structure	
Number of Firms	5
Number of Markets	27
Number of Periods	20
Number of Households	1.0e8
Model	
State Space	-1.4:0.2:1.4
Discount Factor	0.925
Marginal Utility of Income	-0.222
Marginal Cost Parameters	[2.47 0.0]
Investment Cost Parameters	[2.625 1.624 0.5096]
Transition Probability Parameters	[0.547 0.072 -0.884 -0.285]
II Estimator	
Number of II Simulations	5
BBL Estimators	
BBL: Number of Inequalities	1000
BBL: Number of Simulated Paths	500
BBL: Simulation Horizon	80
BBL: Vector of Scalar Deviations	[0.9 0.95 1.05 1.1]

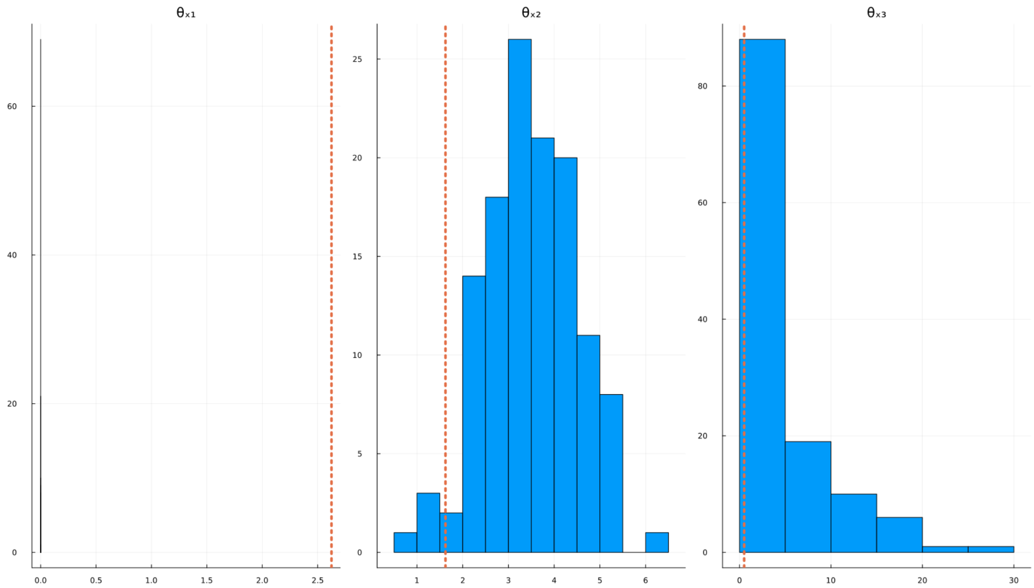
This table reports parameters used in our Monte Carlo exercise. The parameters labeled BBL relate to our implementation of the BBL estimator. Number of inequalities is the number of equilibrium conditions included in the BBL objective. Number of simulated paths is the number of paths used in estimating value functions via simulation. Simulation horizon is how far the paths are simulated. The vector of scalar deviations defines the HvB deviations discussed in the text. The following set of parameters relate to the HvB model discussed above. Number of II simulations is the number of ν draws per observed observation used in the II estimator. The number of markets and periods pertain to the size of the dataset used in the Monte Carlo simulations. State space is the set of possible qualities for a firm; the notation $-1.4 : 0.2 : 1.4$ indicates that quality can be as low as -1.4 and as high as 1.4, growing in increments of 0.2.

Figure 9: II Estimator Parameter Estimates



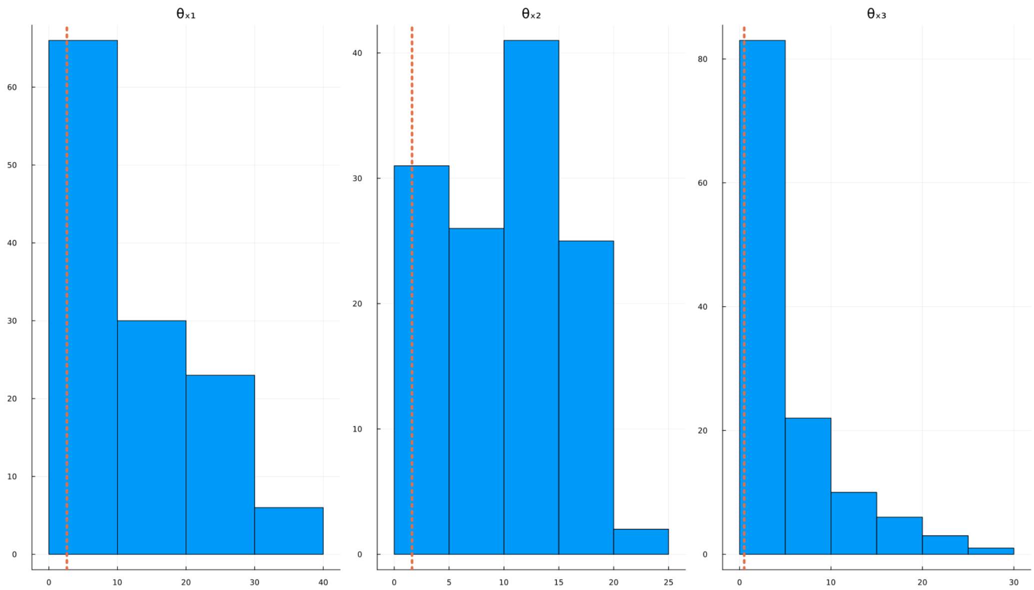
This figure plots the distribution of parameter estimates obtained using the II estimator described in Equation 11 over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

Figure 10: HvB Estimator Parameter Estimates



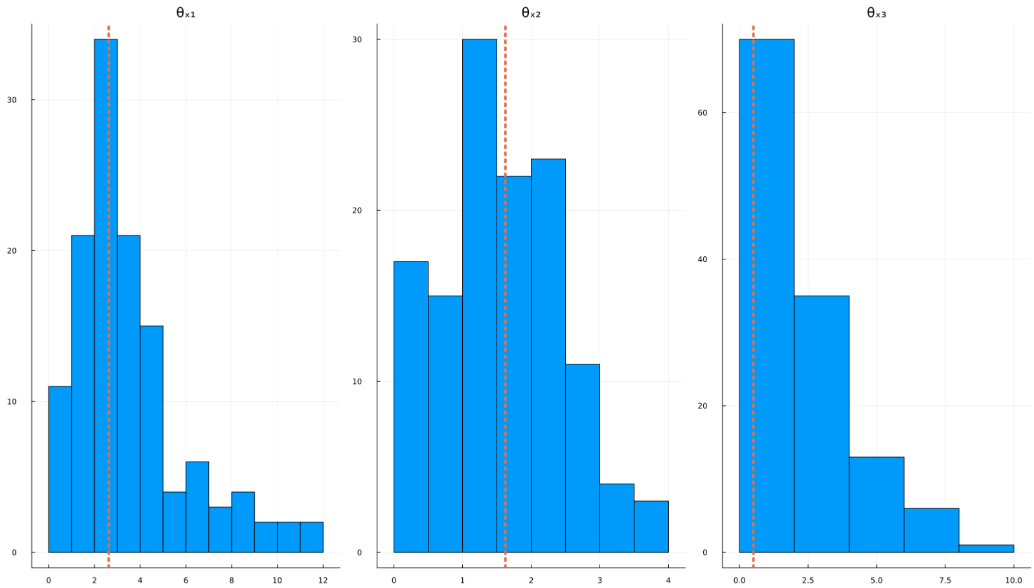
This figure plots the distribution of parameter estimates obtained using the BBL estimator with HvB deviations over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

Figure 11: BBL Estimator Parameter Estimates



This figure plots the distribution of parameter estimates obtained using the BBL estimator with random deviations over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

Figure 12: NLLS Estimator Parameter Estimates



This figure plots the distribution of parameter estimates obtained using the NLLS estimator over 125 Monte Carlo replications. The vertical dashed red line indicates the value of the corresponding parameter in the data generating process.

Table 5: Summary of Parameter Estimates

Parameter	Value	II	BBL	HvB	NLLS
θ_{x1}	2.625	2.171	[11.404, 11.404]	[0.0, 0.0]	3.51
		(0.0, 3.514)	(0.0, 30.195)	(0.0, 0.0)	(0.0, 9.992)
θ_{x2}	1.624	1.838	[9.877, 9.877]	[3.521, 3.521]	1.584
		(0.339, 3.328)	(0.0, 19.308)	(1.382, 5.288)	(0.072, 3.266)
θ_{x3}	0.5096	0.459	[4.447, 4.447]	[3.757, 3.757]	1.885
		(0.0, 0.737)	(0.0, 19.666)	(0.0, 17.224)	(0.0, 6.786)

This table summarizes the results of our Monte Carlo experiment. The first column shows the value of the parameters of the investment cost function in the data generating process. Each subsequent column shows the mean and standard error of estimates across Monte Carlo replications. The column labeled "II" shows the results of the estimator we propose in this paper. The column labeled "BBL" shows the results of the BBL estimator with random deviations. The column labeled "HvB" shows the results of the BBL estimator with HvB deviations. The column labeled "NLLS" shows the results of the estimator defined by the program **??**.

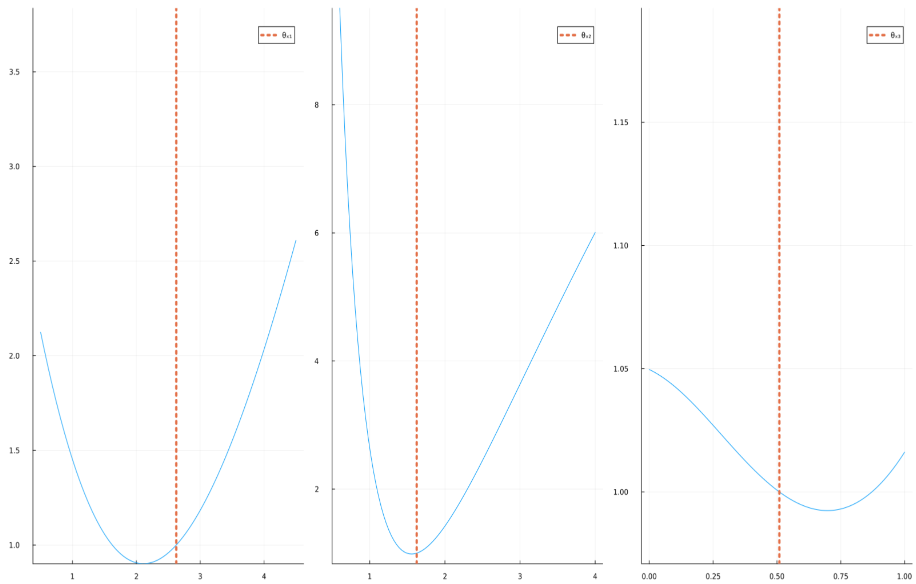
The relative performance of the different estimators is a consequence of the differences in the objective functions that define them. We illustrate this by analysing the shape of each objective function in a neighbourhood of the true parameters for a particular simulated dataset. To do so, we plot two-dimensional slices of the objective function by varying one parameter at a time while holding the others fixed at their true values. To render the shape of different objective functions comparable, we normalise objective values on the parameter grid by dividing each by the objective value at the true parameter. Figures 13 to 16 display the results. Vertical lines represent true parameter values.

Figure 13 shows that the objective function of the II estimator features pronounced local convexity around each true parameter, with minimal distance between the local objective minimum and the objective value at the true parameter. This is reflected in the good performance of the estimator, both in terms of short 95% intervals and small bias.

The picture is very different for the HvB and BBL estimators. They are approximately flat around the true parameter, and zooming in reveals that they display no convexity at all for the considered grid – they are almost linear around the correct parameter values. This is consistent with the large bias displayed in Figures 10 and 11: the objective is convex around parameter values that are wrong by orders of magnitude.

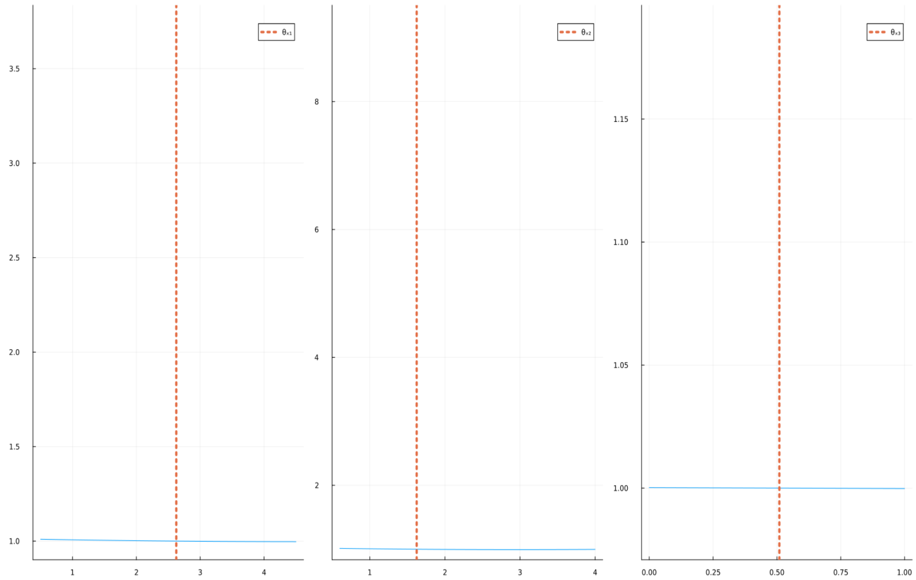
Finally, objective plots for the NLLS estimator are informative about its imprecision. Objective function minima are not far from the minima at the true parameter value (low bias) for θ_{x1}, θ_{x2}, but the NLLS objective function is much less convex around local minima than the II objective function is. On the other hand, the objective function is not centered around the true parameter objective value for θ_{x3}.

Figure 13: II Estimator Objective



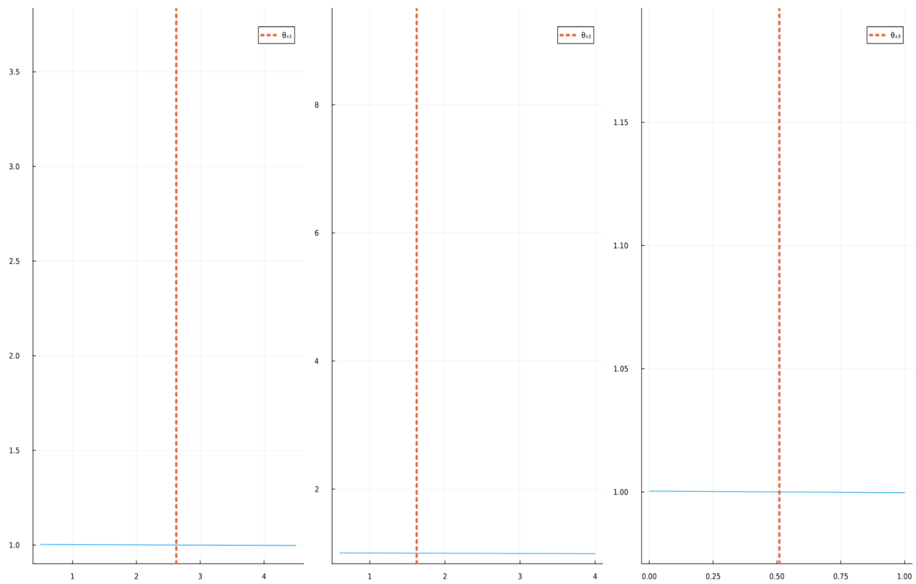
This figure plots the value of the II estimator objective function varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

Figure 14: HvB Estimator Objective



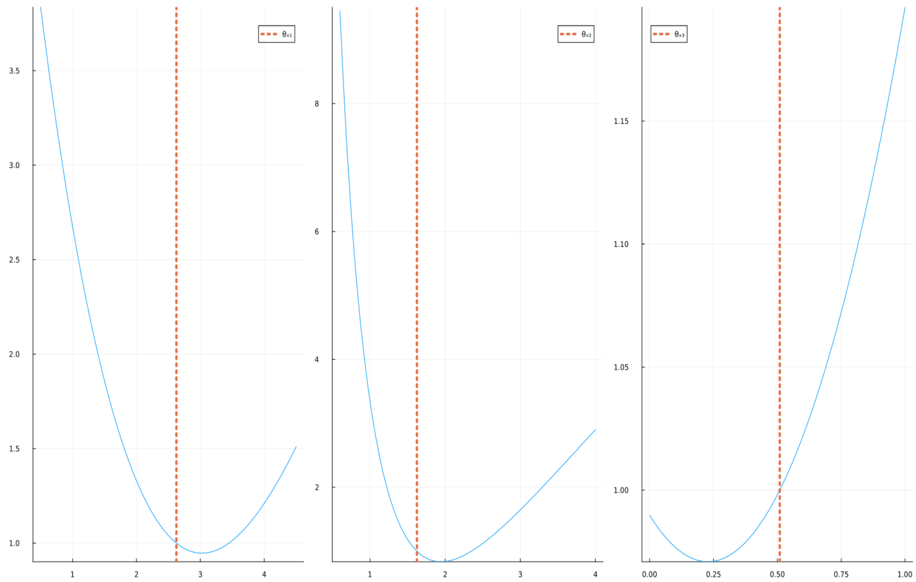
This figure plots the value of the objective function of the BBL estimator with HvB deviations varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

Figure 15: BBL Estimator Objective



This figure plots the value of the objective function of the BBL estimator with random deviations varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

Figure 16: NLLS Estimator Objective



This figure plots the value of the objective function of the estimator defined by program **??** varying one parameter at a time while holding the other two parameters fixed at their true values. Values are scaled by the value of the objective at the true parameters.

We end this section by again providing an overview of the computational costs of each estimator. Table 6 displays run time and memory allocation for the different steps of our simulation, averaged over 125 runs. As expected Step 1 – shared by all the algorithms considered – is fast, as it only entails estimating transition parameters by maximum likelihood and policy function parameters by linear regression.

On the other hand, there is substantial variation in the time and memory costs of various Step 2 algorithms. BBL-type estimators are the fastest, with both the HvB and BBL estimator taking less than 2 seconds between simulation and optimisation. Note that simulation time is remarkably similar for this smaller dataset and the larger dataset in Section 5, so BBL costs do not scale with sample size in the same way as costs for recursive estimators do. This suggests a trade-off between precision and computational time that could lead to favor BBL in larger data sets.

The II estimator is substantially slower, taking approximately 10s to return an estimate. Estimation time will be increasing in the number of datasets generated for indirect inference, which again we set to 5. Finally, the NLLS estimator takes the most time of all, clocking at approximately 25 seconds.

The reason for this substantial cost is that evaluation of its objective and gra-

Table 6: Exercise Timings

Call	Average Time	Average Allocation
Overall		
Preliminaries	30.7s	36.8GiB
Simulation	84.7ms	6.16MiB
Step 1: Estimating Transition Parameters	35.7ms	2.52MiB
Computing $\mathbb{E}$VF Covariates	197ms	4.43MiB
Computing Projection Objects	7.96ms	3.40MiB
Step 2: II Estimator	10.0s	3.81GiB
Step 2: NLLS Estimator	25.7s	9.66GiB
Step 2: BBL Policy Covariates	13.8ms	14.4MiB
Step 2: BBL Estimator	1.78s	16.2MiB
Simulation	1.72s	2.60MiB
Optimisation	59.0ms	13.6MiB
Step 2: HvB Estimator	1.74s	12.0MiB
Simulation	1.73s	1.82MiB
Optimisation	11.7ms	10.2MiB

This table reports the computation time and memory allocation associated with each step in the computation of the four estimators discussed above. Note that the memory allocation reported here is not the amount of memory required to compute these estimators, but rather the total size of objects that are written to memory throughout the course of estimation. "Preliminaries" include the calculation of descriptors of flow profits, objects describing the state space of the game, and solving for a MPE. "Simulation" is the simulation of the data. "Computing VF Covariates" refers to the calculation of $\boldsymbol{X}^{EV}(\boldsymbol{\xi};\sigma)$ discussed in section **??**. "Computing Projection Objects" refers to the calculation of objects used in the implementation of the II and NLLS estimators that need to be computed only once. The other items refer to the different estimators.

dient requires solving a large number of static optimisation problems, whereas the gradient of the II estimator is readily computed analytically (see Appendix B).