USC Dornsife
Dana and David Dornsife College of Letters, Arts and Sciences

Institute for New Economic Thinking

USC Dornsife Institute for New Economic Thinking

Working Paper No. 17-21

Computing Moment Inequality Models Using Constrained Optimization

Baiyu Dong, Yu-Wei Hsieh, and Matthew Shum

June 22, 2017

COMPUTING MOMENT INEQUALITY MODELS USING CONSTRAINED OPTIMIZATION

BAIYU DONG[1], YU-WEI HSIEH[2], AND MATTHEW SHUM[3]

Abstract. Inference for moment inequality models is computationally demanding, and often involves time-consuming grid search. By exploiting the equivalent formulations between unconstrained and constrained optimization, we establish new ways to compute the identified set and its confidence set in moment inequality models which overcome some of these computational hurdles. In simulations, using both linear and nonlinear moment inequality models, we show that our methods can find significantly better solutions and save considerable computing resources relative to conventional grid search. Our methods are user-friendly and can be implemented using a variety of available software packages.

Keywords: Moment Inequality Models; Constrained Optimization; MPEC; MPCC; Partial Identification; Computing Identified Set and Confidence Set

JEL Classification: C61; C63

1. Introduction

Inference in partially identified and moment inequality models is one of the fastest growing subjects in econometrics over the past few years. However, the theoretical work in this area has far outstripped the empirical and applied work utilizing these tools; arguably, this is because the various inference methods suggested in the theoretical literature are often recognized as computationally demanding, and often involve time-consuming grid search. In this paper, we explore ways to overcome the computational hurdles associated with inference in moment inequality models by using constrained optimization techniques.

Date: June 29, 2017.

[1]University of Southern California. E-mail: baiyudon@usc.edu

[2]University of Southern California. E-mail: yuwei.hsieh@usc.edu

[3]California Institute of Technology. E-mail: mshum@caltech.edu

Acknowledgement: We thank Xiaoxia Shi, Christopher Sullivan and Richard Waltz for helpful comments.

There is a large literature in optimization theory exploiting the equivalence between solutions to constrained optimization problems and solutions to a system of equations to design efficient numerical solvers. However, this equivalence is only beginning to be appreciated and applied in the econometrics literature, for computing complex structural models. Su and Judd (2012) and Dube, Fox, and Su (2012) apply these tools to the estimation of (respectively) dynamic discrete-choice and discrete-choice demand models; one main contribution of the present paper, then, can be viewed as extending the applicability of such tools from these point-identified settings to partially-identified models.

Specifically, we use constrained optimization tools to establish new ways to compute the identified set and its confidence set in moment inequality models. Starting with a set of moment inequalities, we show that the identified set can be represented as the solution to a corresponding constrained optimization problem, which also helps in inference—construction of the confidence set. For specificity, we focus on the inference procedure proposed in Chernozhukov, Hong, and Tamer (2007) (hereafter CHT), which is one of the more popular approaches in the literature. The inference procedure in CHT is computationally difficult, involving multiple steps, but in this paper we show how constrained optimization approaches can simplify each step in the procedure.

Inference in moment inequality models involves non-trivial computational issues. First, moment inequality models usually involve optimization of a non-smooth criterion function due to one-sided penalization. Second, the test statistics used for inference in these models are often defined as *suprema* of the criterion function restricted to a *contour set*. Typically, the criterion function is a convex loss function, and maximizing a convex function is therefore a non-convex programming problem. Further, the contour set is generally difficult to characterize, and hence empirical researchers often resort to brute-force grid search.[1] In grid search, to achieve high accuracy, it is necessary to generate large set of points within the contour set, so that grid search becomes the source of difficulties in computing moment inequality models.

In this paper, we provide new solutions to these issues from the perspective of constrained optimization (CO). Our Monte Carlo results show that the CO approach can achieve significantly better results relative to grid search and save considerable computing resources.

[1]Instances include the fixed grid search technique in the 2011 working paper version of Pakes, Porter, Ho, and Ishii (2015) (hereafter PPHI), or stochastic grid generated by MCMC, as in Ciliberto and Tamer (2009).

The CO approach allows us to leverage state-of-the-art solvers based on interior point methods, instead of inefficient and memory-intensive grid search procedures. Within the specific framework of CHT, we identify three computational problems : [a] the problem of computing the identified set; [b] sketching the boundary of the identified set; and [c] computing the confidence set of the identified set. We show that each of these computational problems can be formulated in terms of equivalent smooth constrained optimization problems. Our method is also easy to program for practitioners, and can be implemented using a number of readily available software packages.

The key element of our approach lies in adding additional variables or slackness variables in a way that can produce sparse Jacobian and Hessian of Lagrangian to achieve computational efficiency and smoothness of the problem. This modeling trick also appears in the moment inequality literature but for different purposes. For example, Shi and Shum (2015) study a class of moment inequality models which can be formulated in a CO framework by introducing nuisance parameters. Their motivation, however, is to obtain a simpler asymptotic theory. Recently, Chen, Christensen, O'Hara, and Tamer (2016) propose an MCMC-based approach to construct the confidence set. They transform moment inequalities into moment equalities by adding slackness variables, and jointly sample the structural parameters as well as the additional slackness variables. However, adding slackness variables increases the dimensionality of the problem. For existing procedures used for moment inequality models, then, the introduction of slackness variables leads to a "curse of dimensionality" in computational time and complexity. Fortunately, for our constrained optimization approach, high-dimensionality is less problematic, and in simulations we see that computational time for some algorithms increase only linearly in the number of moment inequalities.

In the following section we first review moment inequality models. Section 3 provides the CO formulation for computing the identified set. Section 4 describes how to sketch the boundary of the identified set. Using the results from both section 3 and 4, we detail the CO formulation of CHT in section 5. Section 6 contains Monte Carlo simulations from two typical moment inequality problems: the static entry game and interval outcome regression. Section 7 concludes.

2. Set-Identified Moment Condition Models

We start by introducing a model where the parameter of interest θ is defined by a collection of moment inequalities. Suppose the parameter space Θ is a compact subset of the Euclidean space $\mathbb{R}^d$. The population moment conditions can take either inequality or equality form as follows:

$$
\begin{aligned}
E_P[m_j(W_i,\theta)] &\leq 0, \quad \text{for} \quad j = 1, \ldots, J_{ie} \\
E_P[m_j(W_i,\theta)] &= 0, \quad \text{for} \quad j = J_{ie}+1, \ldots, J_{ie}+J_e,
\end{aligned} \tag{1}
$$

where $m_j(W_i, \theta)$ is the moment function of parameter $\theta \in \Theta$ and random vector W_i. The expectation operator E_P is define in terms of the population probability law P that generates the data W_i. The set of parameters that satisfy Eq. (1), the identified set, is denoted by Θ_I. In the literature, one usually characterizes the identified set as the solution to the following *unconstrained* optimization problem

$$
\Theta_I = \text{argmin}_{\theta \in \Theta} Q(\theta), \tag{2}
$$

where the nonnegative population criterion function is assumed to be

$$
Q(\theta) = \sum_{j=1}^{J_{ie}} \Big(\max \Big\{ E_P[m_j(W_i,\theta)], 0 \Big\} \Big)^2 + \sum_{j=J_{ie}+1}^{J_{ie}+J_e} E_P[m_j(W_i,\theta)]^2. \tag{3}
$$

It is easy to see that the identified set can be written as

$$
\Theta_I = \{\theta \in \Theta, Q(\theta) = 0\}. \tag{4}
$$

The estimation and inference are then carried out by replacing the population moments in Eq. (1) by their sample analog

$$
E_n[m_j(W_i,\theta)] = \frac{1}{n} \textstyle\sum_{i=1}^n m_j(W_i,\theta).
$$

The sample criterion function based on sample moments is denoted by $Q_n(\theta)$.

We investigate three computational problems related to moment inequality models: [a] the problem of computing the identified set; [b] sketching the boundary of the identified set; and [c] computing the confidence set of the identified set. First, Eq. (3) is a *non-smooth* function of parameters due to the one-sided penalty for moment inequalities, even if all moments are *smooth* functions of parameters. In section 3, we show that this problem can be solved by introducing additional variables and reformulating the non-smooth unconstrained optimization problem (2) and (3) into a smooth constrained optimization problem. Second, the solution of Eq. (2) is a set, rather than a singleton. In practice, researchers may wish to report only the extremal points of the identified set. In section 4, we offer a simple numerical procedure to "sketch out" the identified set, and more generally, the convex hull of the identified set.[2] The *third* problem, computing the confidence set as in CHT, is the most challenging one. Constructing the confidence set involves computing a test statistics,

[2]In particular, if all moments are linear in parameters, then problem [a] is a quadratic programming (QP) problem and problem [b] is a quadratically-constrained program (QCP) which are convex optimization

which is itself a non-convex CO problem. We show, in section 5, that by combining our procedures for problems [a] and [b], we are also able to compute the test statistics via efficient CO solvers, instead of applying more computationally and memory intensive grid search. We consider each problem in turn in the following sections.

3. Computing the Identified Set

Finding the sample identified set $\hat{\Theta}_I$ involves minimizing the following criterion function

$$\min Q_n(\theta) = \sum_{j=1}^{J_{ie}} \left(\max \left\{ E_n[m_j(W_i, \theta)], 0 \right\} \right)^2 + \sum_{j=J_{ie}+1}^{J_{ie}+J_e} E_n[m_j(W_i, \theta)]^2. \tag{5}$$

This problem can be stated as the following CO problem:

$$\begin{aligned} \min_{\theta,\epsilon_j} \quad & \sum_{j=1}^{J_{ie}+J_e} \epsilon_j^2 \\ s.t. \quad & E_n[m_j(W_i,\theta)] \le \epsilon_j \quad \text{for} \quad j = 1, \ldots, J_{ie} \\ & E_n[m_j(W_i,\theta)] = \epsilon_j \quad \text{for} \quad j = J_{ie}+1, \ldots, J_{ie}+J_e, \\ & \epsilon_j \ge 0 \quad \text{for} \quad j = 1, \ldots, J_{ie} \\ & \epsilon_j \in \mathbb{R} \quad \text{for} \quad j = J_{ie}+1, \ldots, J_{ie}+J_e. \end{aligned} \tag{6}$$

Notice that Eq. (6) is always feasible. For any $\theta \in \Theta$, one can choose $\epsilon_j, j = 1, \ldots, J_{ie}$ arbitrarily large to satisfy the constraints. For moment equalities, ϵ_j is simply the residual in the usual sense. For moment inequalities, ϵ_j measures the deviation of constraint j. If the moment inequality $E_n[m_j(W_i, \theta)]$ is satisfied, the corresponding ϵ_j is driven down to zero. Otherwise, suppose $E_n[m_j(W_i, \theta)] = \eta_j > 0$, then one can choose $\epsilon_j = \eta_j$ to meet both feasibility and optimality. Next we state the equivalence result:

Proposition 1. *The CO problem of Eq. (6) is equivalent to the unconstrained problem of Eq. (5).*

Proof: For a given θ, there is only one ϵ_j that can satisfy the moment equality $E_n[m_j(W_i, \theta)] = \epsilon_j$. There are many ϵ_js that can satisfy the moment inequality $E_n[m_j(W_i, \theta)] \le \epsilon_j$, but only $\epsilon_j = \max\{E_n[m_j(W_i, \theta)], 0\}$ is optimal. By profiling out ϵ_j as a function of θ, one obtains Eq. (5). □

problems and hence can be easily solved by commercial solvers even with large number of moment conditions. (See Boyd and Vandenberghe (2004) for more discussions on QP and QCP).

In the special case in which all moment conditions are linear in parameters, Eq. (6) is a QP problem. On the other hand, incorporating linear moment conditions into the objective function as in Eq. (5) will result in a non-smooth optimization problem, which is considerably more difficult to solve than QP, because one is unable to fully utilize the gradient information of each moment. One potential concern of the CO formulation is that it includes additional optimizing variables ϵ_js.[3] However, since each ϵ_j is only associated with *one* constraint in an *additive* form, the resulting Jacobian is very sparse, and it vanishes in the Hessian of Lagrangian as well. As we shall see in simulations, such sparsity pattern ensures that one can effectively solve the large scale optimization problem.

4. Sketching the Boundary of the Identified Set

By solving (6), one is able to find a particular solution $\theta \in \hat{\Theta}_I$. Tracing out the entire identified set in the general case inevitably calls for grid search. Alternatively, we provide a numerically lighter procedure to sketch the boundary of the identified set, so the researchers can approximately estimate the location, size, and the shape of the identified set.

First, find the best possible objective value

$$q_n \equiv Q_n(\hat{\theta}) \tag{7}$$

by solving Eq. (6).

Second, solve the following problem

$$\begin{aligned} &\max_{\theta,\epsilon_j} && c'\theta \\ &s.t. && E_n[m_j(W_i,\theta)] \le \epsilon_j \quad \text{for} \quad j = 1,\dots,J_{ie} \\ & && E_n[m_j(W_i,\theta)] = \epsilon_j \quad \text{for} \quad j = J_{ie}+1,\dots,J_{ie}+J_e \\ & && \epsilon_j \ge 0 \quad \text{for} \quad j = 1,\dots,J_{ie} \\ & && \epsilon_j \in \mathbb{R} \quad \text{for} \quad j = J_{ie}+1,\dots,J_{ie}+J_e \\ & && \textstyle\sum_{j=1}^{J_{ie}+J_e} \epsilon_j^2 \le q_n + x_{tol}, \end{aligned} \tag{8}$$

where x_{tol} is a small positive number for numerical tolerance purpose. The key step is to include the *objective* function in Eq. (6) as a *constraint* here to restrict the parameter space to be $\hat{\Theta}_I$. We then choose an appropriate linear objective coefficient vector c to

[3]This problem also arises in the recent structural estimation literature; e.g. Dube et al. (2012), and Su and Judd (2012).

trace out the boundary of the identified set. For example, researchers may be interested in the (componentwise) extreme points of the identified set, instead of the entire set. This "bounding box" of $\hat{\Theta}_I$ can be calculated by choosing $c_k = \pm 1$, the k-th coordinate of c, and zero otherwise.

When all moments are linear (and hence the identified set is convex), our procedure corresponds to computing the support function of the convex identified set. In particular, Eq. (8) becomes a QCP problem. For general nonlinear moments, our procedure produces a convex hull of the identified set. The support function is widely used in the moment inequality literature.[4] Our contribution is to provide a simple, explicitly stated CO formulation which can be easily solved by modern solvers. It is worth mentioning that what we propose is the support function defined on the extended parameter space $(\theta, \epsilon_1, \ldots, \epsilon_J)$, not the support function defined on the space of θ as in the other papers. As we shall see next, the presence of residuals ϵs enables us to bound the contour set of the sample criterion function, a crucial element in CHT.

5. Inference: Computing the Confidence Set

In this section we illustrate how CO can be deployed for inference in moment inequality models—computing confidence sets which cover the identified set of the parameters. We start by reviewing CHT's inference procedure.

1. Contour Set

First, define the contour set of the sample criterion at level c

$$C_n(c) = \{\theta \in \Theta, a_n Q_n(\theta) \leq c\}, \tag{9}$$

where $a_n \to \infty$ is a normalizing constant proportional to n. The subscript n in the above definition indicates that the criterion function is computed using the full sample $i = 1, \ldots, n$.

2. Sub-Sampling

The critical value of CHT is obtained via sub-sampling test statistics. Consider sub-samples of size b, $b < n$ and $b/n \to 0$, from the full sample $(W_1, \ldots, W_n)$ without replacement. Repeat L times. We let B_l denote the index set $(l_1, l_2, \ldots, l_b) \subset (1, 2, \ldots, n)$ of the l-th sub-sample.

[4] See the 2011 working paper version of PPHI, Bontemps, Magnac, and Maurin (2012) and Kaido, Molinari, and Stoye (2016).

3. Test Statistics

The test statistic of CHT, computed from the l-th sub-sample is defined as

$$\hat{\mathcal{C}}_{l,b} = \max_{\theta \in C_n(q_n+k_n)} a_b Q_{l,b} \tag{10}$$

The subscripts in $Q_{l,b}$ indicate that the criterion function is computed from the l-th sub-sample of size b, while the subscript in the contour set C_n indicates that constraints involve the full sample of size n. $k_n \propto \log n$ is a user-chosen parameter, and q_n is the best possible objective value define above, which can be obtained by solving Eq. (6). The computational complexity of the CHT's test statistics is twofold: 1. the parameter constraints, and 2. the non-convex objective function even without the presence of moment inequalities.

First, regarding the constraints, in the literature one often resorts to grid search to represent $\theta \in C_n(q_n+k_n)$: by generating many θ_i and checking whether each θ_i belongs to the contour set. This computationally intensive method is a result of the criterion function formulation of the contour set. However, by comparing Eqs. (10) and (8), we recognize that by setting $x_{tol} = k_n$, the condition $\theta \in C_n(q_n + k_n)$ involves *exactly the same constraints* as in Eq. (8). This key observation enables us to explicitly formulate $\theta \in C_n(q_n + k_n)$ as constraints in an optimization problem instead of using memory-intensive grid search. Another advantage of CO is that one can exploit the sparsity of the gradient and Hessian to improve the numerical performance.

Nevertheless, there is a *second* issue, regarding the objective function, that we must *maximize* a statistical loss function and hence the problem is fundamentally non-convex. Whether the moment inequalities are satisfied or not further introduces discontinuities. We propose the following MPCC (Mathematical Programming with Complementarity Constraints) formulation to tackle these issues. (Since the difficulty is mainly due to the presence of moment inequalities, for convenience below we consider models without equalities.)

First, maximizing $Q_{l,b}$ can be equivalently expressed as

$$\begin{aligned} \max \quad & \textstyle\sum_{j=1}^{J} \eta_j^2 \\ s.t. \quad & \max\{E_b[m_j(W_{l_i}, \theta)], 0\} = \eta_j. \end{aligned} \tag{11}$$

The presence of the "max" operation in the constraints leads to a non-linear programming problem with discontinuous derivatives. Standard nonlinear optimization solvers may have problems handling this case. We therefore consider an alternative representation of the max constraint by adding slackness variable s_j to the j-th moment inequality

$$\begin{aligned} & E_b[m_j(W_{l_i}, \theta)] + s_j = \eta_j \\ & s_j \eta_j = 0 \\ & s_j \geq 0 \\ & \eta_j \geq 0 \end{aligned} \tag{12}$$

The last three conditions, collectively known as the *complementarity constraint*, are sometimes written in shorthand as "$0 \leq s_j \perp \eta_j \geq 0$".[5] By introducing s_j, we effectively transform the indicator for whether the moment inequality is violated into a smooth non-linear constraint. Suppose that the moment inequality is violated at θ, then $s_j = 0$ and $E_b[m_j(W_{l_i}, \theta)] = \eta_j$ is the only solution of Eq. (12). If the moment inequality is satisfied at θ, then $\eta_j = 0$ and $s_j = -E_b[m_j(W_{l_i}, \theta)]$ is the only solution of Eq. (12). Putting everything together, the test statistic in Eq. (10) can be formulated as

$$\begin{aligned} \max_{\theta,\epsilon,\eta,s} \quad & a_b \sum_{j=1}^{J} \eta_j^2 \\ s.t. \quad & E_b[m_j(W_{l_i}, \theta)] + s_j = \eta_j \\ & 0 \leq s_j \perp \eta_j \geq 0 \\ & E_n[m_j(W_i, \theta)] \leq \epsilon_j \quad \text{for} \quad j = 1, \ldots, J \\ & \epsilon_j \geq 0 \quad \text{for} \quad j = 1, \ldots, J_{ie} \\ & a_n \sum_{j=1}^{J} \epsilon_j^2 \leq q_n + k_n. \end{aligned} \tag{13}$$

The cost of MPCC (13) is the increased dimensionality: an additional $2J_{ie}$ parameters are introduced to accommodate the non-smooth objective function, and additional J_{ie} parameters are required to deal with the CO representation of the contour set. But as we noted earlier, modern constrained optimization methods can handle the large number of auxiliary slackness parameters readily (due to the accompanying sparsity). To see it in action, we turn next to several Monte Carlo investigations.

[5]Because many equilibrium conditions in economic models also involve complementarity conditions, the acronyms MPCC and MPEC (Mathematical Programming with Equilibrium Constraints) are used interchangeably in the optimization literature. Note that the MPEC formulations of Dube et al. (2012) and Su and Judd (2012), however, do not contain complementarity constraints.

6. Monte Carlo Experiments

We consider two experiments: the bivariate complete information entry game and the interval outcome regression. The first example is a nonlinear moment inequality model, while the second one is a linear moment inequality model. We focus on the numerical issues only. Specifically, we only report the test statistics computed from different methods, instead of reporting the empirical coverage probability of the confidence set. The latter quantity can be attributed to statistical sampling issues, not necessarily numerical issues. We use either Gurobi or KNITRO[6] to solve various CO problems. As we shall see, they deliver considerable improvement over the grid search and the other solvers available in Matlab.[7]

Example 1: Bivariate Complete Information Entry Game. We consider the 2-player complete information entry game studied by Tamer (2003)

$$y_1 = \mathbf{1}[X_1\beta_1 + y_2\Delta_1 + \epsilon_1 \geq 0]$$
$$y_2 = \mathbf{1}[X_2\beta_2 + y_1\Delta_2 + \epsilon_2 \geq 0].$$

We further assume (ϵ_1, ϵ_2) are bivariate normal with zero means and covariance matrix $\mathbf{\Sigma}$ with unit variances and correlation coefficient ρ.

Suppose these two players play a pure strategy Nash equilibrium (NE). It can be shown that the game has two NE, $(1,0)$ or $(0,1)$, when $(\epsilon_1, \epsilon_2) \in [-X_1\beta_1, -X_1\beta_1 - \Delta_1] \times [-X_2\beta_2, -X_2\beta_2 - \Delta_2]$. Let $\hat{P}_{ij}$ denote the observed frequency of event (i, j), and P_{ij} denote the frequency implied by the model. The CDF of bivariate normal is denoted by Φ. We assume event $(1,0)$ is selected with probability τ in the region of multiplicity. If one remains agnostic about the equilibrium selection, as in Tamer (2003), this model implies two moment equalities and

[6]Recently, KNITRO is proven to be very effective in solving structural models. Examples include Dube et al. (2012), Su and Judd (2012), and Dong, Hsieh, and Zhang (2017). Gurobi has been used to solve large scale matching models; e.g., Galichon, Kominers, and Weber (2016). We use Gurobi 7.0 and KNITRO 10.2.1.

[7]We use Matlab 2017a, with Optimization Toolbox Version 7.6 and Global Optimization Toolbox Version 3.4.2. All optimization solvers run on a Windows 10 desktop with Intel Core i7-5960X over-clocked to 3.3 GHz for 8 cores and 32GB RAM. We do not enable the solvers' parallel computation features.

four moment inequalities for structural parameters $(\beta_1, \beta_2, \Delta_1, \Delta_2, \rho)$:

$$\begin{aligned}
\hat{P}_{00} - P_{00} &= 0, \\
\hat{P}_{10} - P_{10}^U &\leq 0, \\
P_{10}^L - \hat{P}_{10} &\leq 0, \\
\hat{P}_{01} - P_{01}^U &\leq 0, \\
P_{01}^L - \hat{P}_{01} &\leq 0, \\
\hat{P}_{11} - P_{11} &= 0,
\end{aligned}$$

where

$$\begin{aligned}
P_{00} = &\quad Pr(\epsilon_1 \leq -\beta_1 X_1, \epsilon_2 \leq -\beta_2 X_2) \\
= &\quad \Phi(-\beta_1 X_1, -\beta_2 X_2; \boldsymbol{\Sigma}),
\end{aligned}$$

$$\begin{aligned}
P_{10}^U = &\quad Pr(\epsilon_1 \geq -\beta_1 X_1, \epsilon_2 \leq -\beta_2 X_2 - \Delta_2) \\
= &\quad \Phi(\beta_1 X_1, -\beta_2 X_2 - \Delta_2; \boldsymbol{L}_{10} \boldsymbol{\Sigma} \boldsymbol{L}_{10}'),
\end{aligned}$$

$$\begin{aligned}
P_{10}^L = &\quad Pr(\epsilon_1 \geq -\beta_1 X_1, \epsilon_2 \leq -\beta_2 X_2) + Pr(\epsilon_1 \geq -\beta_1 X_1 - \Delta_1, -\beta_2 X_2 \leq \epsilon_2 \leq -\beta_2 X_2 - \Delta_2) \\
= &\quad \Phi(\beta_1 X_1, -\beta_2 X_2; \boldsymbol{L}_{10} \boldsymbol{\Sigma} \boldsymbol{L}_{10}') + \Phi(\beta_1 X_1 + \Delta_1, -\beta_2 X_2 - \Delta_2; \boldsymbol{L}_{10} \boldsymbol{\Sigma} \boldsymbol{L}_{10}') \\
- &\quad \Phi(\beta_1 X_1 + \Delta_1, -\beta_2 X_2; \boldsymbol{L}_{10} \boldsymbol{\Sigma} \boldsymbol{L}_{10}'),
\end{aligned}$$

$$\begin{aligned}
P_{01}^U = &\quad Pr(\epsilon_1 \leq -\beta_1 X_1 - \Delta_1, \epsilon_2 \geq -\beta_2 X_2) \\
= &\quad \Phi(-\beta_1 X_1 - \Delta_1, \beta_2 X_2; \boldsymbol{L}_{01} \boldsymbol{\Sigma} \boldsymbol{L}_{01}'),
\end{aligned}$$

$$\begin{aligned}
P_{01}^L = &\quad Pr(\epsilon_1 \leq -\beta_1 X_1 - \Delta_1, \epsilon_2 \geq -\beta_2 X_2 - \Delta_2) + Pr(\epsilon_1 \leq -\beta_1 X_1, -\beta_2 X_2 \leq \epsilon_2 \leq -\beta_2 X_2 - \Delta_2) \\
= &\quad \Phi(-\beta_1 X_1 - \Delta_1, \beta_2 X_2 + \Delta_2; \boldsymbol{L}_{01} \boldsymbol{\Sigma} \boldsymbol{L}_{01}') + \Phi(-\beta_1 X_1, -\beta_2 X_2 - \Delta_2; \boldsymbol{\Sigma}) - \Phi(-\beta_1 X_1, -\beta_2 X_2; \boldsymbol{\Sigma}),
\end{aligned}$$

$$\begin{aligned}
P_{11} = &\quad Pr(\epsilon_1 \geq -\beta_1 X_1 - \Delta_1, \epsilon_2 \geq -\beta_2 X_2 - \Delta_2) \\
= &\quad \Phi(\beta_1 X_1 + \Delta_1, \beta_2 X_2 + \Delta_2; \boldsymbol{\Sigma}),
\end{aligned}$$

$$\boldsymbol{L}_{10} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, \boldsymbol{L}_{01} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

We calculate $\hat{P}_{ij}$ using the population moments to abstract from the sampling error as the purpose here is to demonstrate how to compute the identified set and its extreme points. The population frequencies of 4 events are given by

$$\begin{aligned}
\hat{P}_{00} &= P_{00}, \\
\hat{P}_{11} &= P_{11}, \\
\hat{P}_{10} &= \tau P_{10}^U + (1 - \tau) P_{10}^L, \\
\hat{P}_{01} &= 1 - \hat{P}_{00} - \hat{P}_{10} - \hat{P}_{11}.
\end{aligned}$$

We first use the same design as in Chen et al. (2016). The observed (population) frequencies of 4 events $(\hat{P}_{00}, \hat{P}_{10}, \hat{P}_{01}, \hat{P}_{11})$ are computed by setting $X_1 = X_2 = 1, \beta_1 = \beta_2 = 0.2, \Delta_1 = \Delta_2 = -0.5, \rho = 0.5, \tau = 0.5$. The parameter space is

$$\Theta = \{-1 \leq \beta_1, \beta_2 \leq 2, -2 \leq \Delta_1, \Delta_2 \leq 0, 0 \leq \rho \leq 1\}.$$

As described in section 3 and 4, we proceed in 2 steps. In the first step we solve Eq. (6). Because we use population moments, the best objective value $q_n = 0$. In the second step we solve Eq. (8) to obtain the componentwise bounds for each parameter. We use numerical gradient and BFGS to approximate the Hessian of Lagrangian. The results are summarized in Table 1. We use 50 random starting values for all optimization problems and they all converge to the same solution. Our results coincide with the numbers reported in Chen et al. (2016). In particular, The identified set for Δ_i is approximately $[-1.41, 0]$. Given the fact that the parameter space is $\Delta_i \in [-2, 0]$, the moment conditions only provide little information about the competition effects. On the other hand, the identified set for ρ is $[0, 1]$. The moment conditions do not provide sufficient information to tighten the bound for the correlated payoff shocks. Our method is computationally attractive. It only takes *0.39 seconds* to obtain the minimum criterion function value q_n by solving Eq. (6), and *13.7 seconds* altogether to obtain the upper and lower bounds for 5 structural parameters by solving 10 problems in the form of (8).

We next investigate whether adding regressors can improve the identification power. We assume that $(X_1, X_2) \in \{0, 0.5, 1, 1.5, 2\}^2$. With variations in regressors, there are total $5^2 \times 6$ moment conditions. We also vary the values of the true parameters, labeled by design 1 to 4 in Table 2.[8] Our numerical results indicate that all parameters are near point identification across 4 very different designs. Theoretically, Tamer (2003) establishes the point identification by resorting to identification-at-infinity argument, which implies that the empirical researchers may need large variations in regressors. Our numerical evidence suggests that, in practice, relatively small variations in regressors might suffice to achieve point identification. Table 2 also shows the robustness and efficiency of pairing CO formulations with interior point method. All 50 starting values converge to the same value in both step 1 and step 2. On average, step 1 only takes around 11 seconds. Step 2 involves 10 optimization problems, each of which has 150 nonlinear moments, but it only takes less than *5 minutes* for all designs.

[8]In design 1 we use the same true parameter values as before. In design 2 we use asymmetric competition effect $(\Delta_1, \Delta_2) = (-0.8, -0.2)$; design 3, asymmetric equilibrium selection $\tau = 0.1$; design 4, strong correlated payoff shocks $\rho = 0.8$.

Example 2: Linear Regression Models with Interval Outcomes. We next demonstrate computing the test statistics for the confidence set in the context of linear interval outcome regression. Consider the following linear regression model

$$Y_i = X_i'\theta + \epsilon_i.$$

While X_i is available, only the interval in which Y_i belongs to, i.e. $Y_i \in [Y_{1i}, Y_{2i}]$, is recorded for the dependent variable. The implied conditional moment inequalities are

$$E_P[Y_{1i}|X_i] \leq X_i'\theta \leq E_P[Y_{2i}|X_i].$$

Following CHT and Andrews and Shi (2013), one can convert the conditional moment inequalities into unconditional moment inequalities by using instrumental variable Z_i, which takes the following form:

$$Z_i = \mathbf{1}[X_i \in \mathcal{X}],$$

Z_i is nothing but a dummy variable indicating whether X_i belongs to some pre-specified region $\mathcal{X}$. The unconditional moment inequalities are given by

$$E_P[Z_i Y_{1i}] \leq E_P[Z_i X_i']\theta \leq E_P[Z_i Y_{2i}],$$

which is linear in θ. The data generating process is summarized in Appendix A.

Let n be the sample size and M be the number of instruments. There are total $2M$ moment inequalities. We let $\boldsymbol{X}$ denote the design matrix, $(\boldsymbol{Y}_1, \boldsymbol{Y}_2)$ denote the dependent variables, $\boldsymbol{Z}$ denote the IV matrix and $|\boldsymbol{\theta}|$ denote the cardinality of structural parameter θ. Let $\boldsymbol{x}' = (\theta_1, \ldots, \theta_{|\boldsymbol{\theta}|}, \epsilon_1, \ldots, \epsilon_M, \epsilon_{M+1}, \ldots, \epsilon_{2M})$ be the decision variables in the following mathematical programming problem, and x_j be the j-th element of $\boldsymbol{x}$. In this case the normalizing constant is $a_n = n$.

The first step of CHT involves computing the minimum sample criterion function value q_n, and the second step involves computing the test statistics, restricted to the contour set at a level determined by the first step estimate of q_n.

Step 1

In the literature one usually obtains q_n by solving the unconstrained problem (5), which we implement using Nelder-Mead, Pattern Search and Simulated Annealing (SA) in Matlab.[9] Meanwhile, we propose computing q_n by solving CO. In this case, Eq. (6) takes the following

[9] Manski and Tamer (2002) use simulated annealing.

form of QP, which we solve by Gurobi:

$$\begin{aligned} q_n = \min \quad & \boldsymbol{x}'\boldsymbol{Q}\boldsymbol{x} \\ s.t. \quad & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b} \\ & x_j \geq 0 \quad \forall j > |\boldsymbol{\theta}|, \end{aligned} \tag{14}$$

where

$$\mathbf{Q} = \left(\begin{array}{c|c} \mathbf{0}_{|\boldsymbol{\theta}|\times|\boldsymbol{\theta}|} & \mathbf{0} \\ \hline \mathbf{0} & \mathbb{I}_{2M\times 2M} \end{array}\right), \boldsymbol{A} = \left(\begin{array}{c|cc} -\boldsymbol{Z}'\boldsymbol{X} & -\mathbb{I}_{M\times M} & \mathbf{0} \\ \hline \boldsymbol{Z}'\boldsymbol{X} & \mathbf{0} & -\mathbb{I}_{M\times M} \end{array}\right), \boldsymbol{b} = \left(\begin{array}{c} -\boldsymbol{Z}'\boldsymbol{Y}_1 \\ \hline \boldsymbol{Z}'\boldsymbol{Y}_2 \end{array}\right)$$

The results are gathered in Table 3. The model contains 10 structural parameters and 1000 linear moment inequalities. We use 10 common starting points for all the algorithms. In this case, Eq. (14) is a convex programming problem and hence all starting points converge to the same solution with $q_n = 0$. Both Nelder-Mead and Pattern Search converge to points which are far from the identified set. They also exhibit large variations across starting points as reflected by the standard deviation of the estimated objective values. SA does slightly better job, with 4 starting points converging to $q_n = 0$. However, its runtime is substantially longer than the interior point method of Gurobi, by a factor of 3124. This dramatic performance difference comes from the fact that the modern solver can effectively utilize the sparsity of both $\boldsymbol{A}$ and $\boldsymbol{Q}$.

Step 2

We next detail how to compute test statistics $\hat{\mathcal{C}}_{l,b}$ defined in Eq. (10), and then compare the performance of MPCC v.s. grid search. This is the most important practical issue, as a poor estimate of $\hat{\mathcal{C}}_{l,b}$ will lead to inaccurate critical value and hence wrong statistical decision. To highlight the source of numerical issue, we focus on comparing $\hat{\mathcal{C}}_{l,b}$ directly, instead of the implied critical value.

Let $(\boldsymbol{Y}_{1l}, \boldsymbol{Y}_{2l}, \boldsymbol{X}_l, \boldsymbol{Z}_l)$ denote the l-th subsample of size $b = 400$ from the full sample $(\boldsymbol{Y}_1, \boldsymbol{Y}_2, \boldsymbol{X}, \boldsymbol{Z})$. There are total $L = 100$ sub-samples. We choose the tuning parameter $k_n = \log(n)$ as in CHT. Let

$$\boldsymbol{x} = (\theta_1, \ldots, \theta_{|\boldsymbol{\theta}|}, \epsilon_1, \ldots, \epsilon_M, \ldots, \epsilon_{2M}, \eta_1, \ldots, \eta_{2M}, s_1, \ldots, s_{2M}).$$

First, we solve Eq. (8) to obtain the bounding box of the contour set $C_n(q_n + k_n)$, and then supply the upper and lower bounds on the structural parameters to MPCC and grid search. In this case, Eq. (8) is a QCP problem and hence can be solved in less than 1 second.

The MPCC formulation (13) takes the following form that can be solved by KNITRO:

$$\begin{aligned} \hat{\mathcal{C}}_{l,b} = \max \quad & \tfrac{1}{b}\boldsymbol{x}'\boldsymbol{Q}_o\boldsymbol{x} \\ s.t. \quad & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b} \\ & \boldsymbol{A}_{eq}\boldsymbol{x} = \boldsymbol{b}_{eq} \\ & \tfrac{1}{n}\boldsymbol{x}'\boldsymbol{Q}_c\boldsymbol{x} \leq q_n + k_n \\ & \eta_j \perp s_j \\ & x_j \geq 0 \quad \forall j > |\boldsymbol{\theta}|, \end{aligned}$$

$$\boldsymbol{A} = \left(\begin{array}{c|cccccc} -\boldsymbol{Z}'\boldsymbol{X} & -\mathbb{I}_{M\times M} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \boldsymbol{Z}'\boldsymbol{X} & \mathbf{0} & -\mathbb{I}_{M\times M} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right), \boldsymbol{b} = \left(\begin{array}{c} -\boldsymbol{Z}'\boldsymbol{Y}_1 \\ \hline \boldsymbol{Z}'\boldsymbol{Y}_2 \end{array}\right),$$

$$\boldsymbol{A}_{eq} = \left(\begin{array}{c|cccccc} -\boldsymbol{Z}_l'\boldsymbol{X}_l & \mathbf{0} & \mathbf{0} & -\mathbb{I}_{M\times M} & \mathbf{0} & \mathbb{I}_{M\times M} & \mathbf{0} \\ \hline \boldsymbol{Z}_l'\boldsymbol{X}_l & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbb{I}_{M\times M} & \mathbf{0} & \mathbb{I}_{M\times M} \end{array}\right), \boldsymbol{b}_{eq} = \left(\begin{array}{c} -\boldsymbol{Z}_l'\boldsymbol{Y}_{1l} \\ \hline \boldsymbol{Z}_l'\boldsymbol{Y}_{2l} \end{array}\right),$$

$$\boldsymbol{Q}_o = \begin{pmatrix} \mathbf{0}_{|\boldsymbol{\theta}|\times|\boldsymbol{\theta}|} & & & \\ & \mathbf{0}_{2M\times 2M} & & \\ & & \mathbb{I}_{2M\times 2M} & \\ & & & \mathbf{0}_{2M\times 2M} \end{pmatrix}, \boldsymbol{Q}_c = \begin{pmatrix} \mathbf{0}_{|\boldsymbol{\theta}|\times|\boldsymbol{\theta}|} & & & \\ & \mathbb{I}_{2M\times 2M} & & \\ & & \mathbf{0}_{2M\times 2M} & \\ & & & \mathbf{0}_{2M\times 2M} \end{pmatrix}.$$

The grid search consists of 3 steps.[10] First, we generate 10^9 uniform random numbers within the bounding box of the contour set $C_n(q_n + k_n)$.[11] Second, we keep those points within $C_n(q_n + k_n)$, ending up with 282,284 points representing the contour set in the case of 100 instruments, and 797,971 points in the case of 500 instruments. Lastly, we find the point that produces the largest objective value.

The Monte Carlo results are summarized in Table 4. We use 100 starting points for MPCC and report the best solution as well as the accumulated runtime for 100 starting points. We compute the mean and median values of test statistics $\hat{\mathcal{C}}_{l,b}$ found by MPCC across 100 sub-samples. To ease the comparisons, the values of test statistics found by grid search are further divided by those of MPCC. For example, if the ratio of grid search/MPCC is less

[10]The grid search consumes around 100GB RAM in the case of 500 instruments, and hence is executed on a HP Z840 workstation with two E5-2687W v3 running at 3.1 GHz and 128GB RAM.

[11]We also try using 10^8 Sobol sequences. The results are similar and hence are not reported here.

than one, it means that grid search is dominated by MPCC. In the case of 100 instruments, the third column of Table 4 shows that MPCC dominates grid search across all 100 subsamples. The objective values obtained from grid search are on average 45% of MPCC, and the runtime is 261 times longer than that of MPCC. We further consider the case with 500 instruments (1000 inequalities). We find that grid search is both computationally and memory intensive. In this case it consumes about 100GB RAM, and it takes 20 cores running for more than two days to find grids representing the contour set $C_n(q_n + k_n)$. By contrast, MPCC takes only 1 core running for less than 1 hour, and the achieved objective values are about *twice* larger than those of grid search. Lastly, we also find that the computing time of MPCC increases in an approximate linear way as the number of moment inequality increases, which makes MPCC an attractive method for computing CHT test statistics.

7. Conclusion

In this paper we propose using constrained optimization to improve the test statistics estimate of CHT, which is a critical step in empirical research. The cost-benefit analysis from our Monte Carlo results suggests that empirical researchers can use constrained optimization formulation to achieve better estimates of the test statistics with much cheaper computing resources, as compared to the typical grid search approach.

At the same time, the Monte Carlo experiments in this paper have considered models with linear moments or twice continuously differentiable nonlinear moments, as required by the interior point method which underlies the solvers used in the simulations. Theoretically our formulations also apply to general moment inequality models. However, in some cases constrained optimization may not always be the best alternative. Specifically, when indicator functions appear in the moments (e.g. Khan and Tamer (2009)) or the criterion function (e.g. Bugni (2010)), it generally leads to mixed integer programming (MIP) problems under our formulation. In ongoing work we are accessing how to alleviate the computational burden for doing inference in such models.

References

Andrews, D. W. K. and X. Shi (2013): "Inference Based On Conditional Moment Inequalities," *Econometrica*, 21, 609–666.

Bontemps, C., T. Magnac, and E. Maurin (2012): "Set Identified Linear Models," *Econometrica*, 80, 1129–1155.

Boyd, S. and L. Vandenberghe (2004): *Convex Optimization*, Cambridge University Press, United Kingdom.

Bugni, F. A. (2010): "Bootstrap Inference In Partially Identified Models Defined By Moment Inequalities: Coverage Of The Identified Set," *Econometrica*, 78, 735–753.

Chen, X., T. M. Christensen, K. O'Hara, and E. Tamer (2016): "MCMC Confidence Sets for Identified Sets," *Working Paper*.

Chernozhukov, V., H. Hong, and E. Tamer (2007): "Estimation And Confidence Regions For Parameter Sets In Econometric Models," *Econometrica*, 75, 1243–1284.

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

Dong, B., Y.-W. Hsieh, and X. Zhang (2017): "Implementing Maximum Likelihood Estimation of Empirical Matching Models: MPEC v.s. NFXP," *Working Paper*.

Dube, J.-P., J. Fox, and C.-L. Su (2012): "Improving the Numerical Performance of BLP Static and Dynamic Discrete Choice Random Coefficients Demand Estimation," *Econometrica*, 80, 2231–2267.

Galichon, A., S. Kominers, and S. Weber (2016): "Costly Concessions: An Empirical Framework for Matching with Imperfectly Transferable Utility," *Working Paper*.

Kaido, H., F. Molinari, and J. Stoye (2016): "Confidence Intervals For Projections Of Partially Identified Parameters," *Working Paper*.

Khan, S. and E. Tamer (2009): "Inference On Endogenously Censored Regression Models Using Conditional Moment Inequalities," *Journal of Econometrics*, 152, 104–119.

Manski, C. F. and E. Tamer (2002): "Inference On Regressions With Interval Data On A Regressor Or Outcome," *Econometrica*, 70, 519–546.

Pakes, A., J. Porter, K. Ho, and J. Ishii (2015): "Notes And Comments Moment Inequalities And Their Application," *Econometrica*, 83, 315–334.

Shi, X. and M. Shum (2015): "Simple Two-stage Inference For A Class Of Partially Identified Models," *Econometric Theory*, 31, 493–520.

Su, C.-L. and K. L. Judd (2012): "Constrained Optimization Approaches to Estimation of Structural Models," *Econometrica*, 80, 2213–2230.

Tamer, E. (2003): "Incomplete Simultaneous Discrete Response Model with Multiple Equilibria," *The Review of Economic Studies*, 70, 147–165.

Appendix A: DGP of Interval Outcome Regression

(1) Initialization

  (a) Sample size $n = 2000$.

  (b) True parameter $\theta = (0.32, 2.51, 0.47, 2.74, 1.58, 0.95, 2.43, 1.91, 1.42, 1.29)$.

(2) Generate data

  (a) Generate X_i from independent normal with mean vector 50 and variance 100.

  (b) $Y_i = X_i'\theta + \epsilon_i$, where ϵ_i is generated from normal with mean 0 and variance 100.

(3) Convert Y_i to $[Y_{1i}, Y_{2i}]$

  (a) The simulated Y_i is censored at 970.

  (b) We use the following intervals with length 10: $[600, 610], [612, 620], \ldots, [960, 970]$.

(4) Generate IV

  (a) We first randomly pick up two indices j, k.

  (b) We generate the (random) endpoints X_{j1} and X_{j2} from the uniform distribution defined on the range of $X(j)$. Repeat the same procedure to obtain X_{k1} and X_{k2}.

  (c) $Z_i = 1$ if $X_i(j) \in [X_{j1}, X_{j2}]$ and $X_i(k) \in [X_{k1}, X_{k2}]$.

  (d) Repeat 4.(a). 10000 times to generate 10000 IVs.

  (e) The simulated IV may contain too many zeros or ones. We further select 100 IVs by the following step.

  (f) Sort IVs by the number of ones in decreasing order. Pick up from the 501-th to the 600-th IVs, from the sorted IVs to form the IV matrix $\boldsymbol{Z}$ in design 1, and 501-th to the 1000-th IVs to form the IV matrix $\boldsymbol{Z}$ in design 2.

Tables

Table 1. Simulation Results of Bivariate Entry Game[a](No Regressors)[b]

Para.	True Values	Est. Bounds		Convergence		Avg Run Time[c]	
		lb	ub	1st Step[d]	2nd Step[e]	1st step	2nd step[f]
β_1	0.2	-0.042900	0.650800		100%		
β_2	0.2	-0.042900	0.650600		100%		
Δ_1	-0.5	-1.411200	0.000000	100%	100%	0.3948	13.7629
Δ_2	-0.5	-1.411100	0.000000		100%		
ρ	0.5	0.000000	0.990000		100%		
τ	0.5	(for simulating data)					

[a]We solve all optimization problems using 50 random starting values, and they all converge to the same value.
[b]Both X_1 and $X_2 = 1$.
[c]Run time is measured in seconds. We report the average time across 50 random starting points.
[d]The first step minimizes the criterion function and find a point belonging to the identified set as by-product.
[e]The second step solves the componentwise bounds for each parameter. It consists of 10 optimization problems.
[f]The reported time for the 2nd step is the sum of the run time for all the 10 optimizations.

Table 2. Simulation Results of Bivariate Entry Game[a](with Regressors)[b]

Para.	True Values	Est. Bounds		Convergence		Avg Run Time[c]	
		lb	ub	1st Step[d]	2nd Step[e]	1st step	2nd step[f]
		Design #1					
β_1	0.2	0.199990	0.200023	100%	100%	10.609	277.195
β_2	0.2	0.199994	0.200008		100%		
Δ_1	-0.5	-0.500089	-0.499983		100%		
Δ_2	-0.5	-0.500042	-0.499982		100%		
ρ	0.5	0.499989	0.500000		100%		
τ	0.5	(for simulating data)					
		Design #2					
β_1	0.2	0.199985	0.200019	100%	100%	11.157	275.473
β_2	0.2	0.199987	0.200043		100%		
Δ_1	-0.8	-0.800032	-0.799989		100%		
Δ_2	-0.2	-0.200080	-0.199323		100%		
ρ	0.5	0.499988	0.500000		100%		
τ	0.5	(for simulating data)					
		Design #3					
β_1	0.2	0.199983	0.200009	100%	100%	11.410	282.626
β_2	0.2	0.199989	0.200079		100%		
Δ_1	-0.8	-0.800119	-0.799979		100%		
Δ_2	-0.2	-0.200277	-0.199689		100%		
ρ	0.5	0.499990	0.500000		100%		
τ	0.1	(for simulating data)					
		Design #4					
β_1	0.2	0.199997	0.200009	100%	100%	11.090	240.913
β_2	0.2	0.199995	0.200008		100%		
Δ_1	-0.5	-0.500016	-0.499992		100%		
Δ_2	-0.5	-0.500077	-0.499992		100%		
ρ	0.8	0.799994	0.800000		100%		
τ	0.5	(for simulating data)					

[a]We solve all optimization problems using 50 random starting values, and they all converge to the same value.
[b]Both X_1 and $X_2 \in \{0, 0.5, 1, 1.5, 2\}$.
[c]Run time is measured in seconds. We report the average time across 50 random starting points.
[d]The first step minimizes the criterion function and find a point belonging to the identified set as by-product.
[e]The second step solves the componentwise bounds for each parameter. It consists of 10 optimization problems.
[f]The reported time for the 2nd step is the sum of the run time for all the 10 optimizations.

Table 3. First-Step Sample Criterion Minimization: Interval Outcome Regression[a]

	Objective Value			Run Time
	Mean	S.D.	Converges to IdS[b]	
Constr. Opt.				
Interior Point[c]	0	0	100%	0.0197
Un-Constr. Opt.[d]				
Nelder-Mead	1.47×10^{10}	1.07×10^{10}	0	0.4201
Pattern Search	1.47×10^{10}	1.07×10^{10}	0	0.4272
Simulated Annealing	9.55×10^{8}	1.79×10^{09}	40%	61.5469

[a]Sample Size $N = 2000$ with 500 instruments.
[b]We use 10 random starting value and report the percentage of $q_n = 0$.
[c]We use gurobi to solved the QP problem.
[d]We use the default settings of fminsearch, patternsearch, and simulannealbnd in Matlab for the unconstrained optimization.

Table 4. Test Statistics of Interval Outcome Regression[a]

	Obj. Values and Ratios[b]			Run Time[c]
	Mean	Median	$(ratio < 1)$%	
Design #1: 100 IVs				
MPCC	1.30×10^{5}	1.16×10^{5}		579.89
Grid Search/MPCC	0.4519	0.4439	100%	151730.00
Design #2: 500 IVs				
MPCC	1.14×10^{5}	1.02×10^{5}		2962.40
Grid Search/MPCC	0.5219	0.4698	98%	206350.00

[a]Simulated data use 100 sub-samples of size 400.
[b]For MPCC, we report the original function values; for grid search, we report the ratios of objective values divided by those of MPCC.
[c]Run time is measured in seconds. We report the average total run time of 100 starting points.