Shared Constraints and Implicit Functions in General Nash Equilibria

Youngdae Kim Michael C. Ferris

University of Wisconsin, Madison

DOE-MACS Annual Meeting, Madison, February 16, 2017

Supply function equilibria

$$\begin{array}{lll} \text{OPF}(\alpha): & \min_y & \text{energy dispatch cost } (y, \alpha) \\ & \text{s.t.} & \text{conservation of power flow at nodes} \\ & & \text{Kirchoff's voltage law, and simple bound constraints} \end{array}$$

α are (given) price bids, parametric optimization

Supply function equilibria

OPF(α):	$\min_y$	energy dispatch cost (y, α)
	s.t.	conservation of power flow at nodes
		Kirchoff's voltage law, and simple bound constraints

α are (given) price bids, parametric optimization

Leader($\bar{\alpha}_{-i}$):	$\max_{\alpha_i, y, \lambda}$	firm i's profit (α_i, y, λ)
	s.t.	$0 \le \alpha_i \le \hat{\alpha}_i$
		y solves OPF$(\alpha_i, \bar{\alpha}_{-i})$

Note that objective involves multiplier from OPF problem

Supply function equilibria

$$\begin{array}{lll} \text{OPF}(\alpha): & \min_y & \text{energy dispatch cost } (y, \alpha) \\ & \text{s.t.} & \text{conservation of power flow at nodes} \\ & & \text{Kirchoff's voltage law, and simple bound constraints} \end{array}$$

α are (given) price bids, parametric optimization

$$\begin{array}{lll} \text{Leader}(\bar{\alpha}_{-i}): & \max_{\alpha_i, y, \lambda} & \text{firm } i\text{'s profit } (\alpha_i, y, \lambda) \\ & \text{s.t.} & 0 \le \alpha_i \le \hat{\alpha}_i \\ & & y \text{ solves OPF}(\alpha_i, \bar{\alpha}_{-i}) \end{array}$$

Note that objective involves multiplier from OPF problem

$$\begin{array}{lll} \text{Leader}(\bar{\alpha}_{-i}): & \max_{\alpha_i, y, \lambda} & \text{firm } i\text{'s profit } (y, \lambda, \alpha) \\ & \text{s.t.} & 0 \le \alpha_i \le \hat{\alpha}_i \\ & & y, \lambda \text{ solves KKT(OPF}(\alpha_i, \bar{\alpha}_{-i})) \end{array}$$

This is an example of an MPCC since KKT form complementarity constraints

Generalized Nash Equilibrium Problem

- Strategy sets $K_i(x_{-i}) \subset \mathbb{R}^{n_i}$ depend on other agents actions
- N agents, $x^* = (x_1^*, \ldots, x_N^*)$ solves the GNEP if

$$x_i^* \in \underset{x_i \in K_i(x_{-i}^*)}{\arg\min} \quad f_i(x_i, x_{-i}^*), \quad \text{for } i = 1, \ldots, N \qquad \text{(GNEP)}$$

- $x_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_N)$
- If $K_i(x_{-i}) \equiv K_i$ problem is called a Nash equilibrium problem (NEP)
- Convert to MCP/VI by replacing optimization problems by KKT/optimality conditions

EMP formulation

```
file empinfo / '%emp.info%' /;
put empinfo 'equilibrium';
loop(i,
  put / 'min', obj(i), x(i), deff(i), defg(i) ;
);
putclose;
```

Quasi-Variational Inequality

$$\langle F(x^*), x - x^*\rangle \geq 0, \quad \forall x \in K(x^*), \qquad \text{(QVI)}$$

Theorem

If $f_i(\cdot,\cdot)$ is continuously differentiable, $f_i(\cdot, x_{-i})$ is a convex function, and $K_i(x_{-i})$ is a closed convex set for each given x_{-i}, then x^* is a solution to the equilibrium problem (GNEP) *if and only if it is a solution to the QVI(K, F) where*

$$K(x) = \prod_{i=1}^{N} K_i(x_{-i}),$$
$$F(x) = (\nabla_{x_1} f_i(x_1, x_{-1})^{\mathsf{T}}, \ldots, \nabla_{x_N} f_N(x_N, x_{-N})^{\mathsf{T}})^{\mathsf{T}}.$$

Special cases: $K(x) \equiv K$ is VI(F, K), $K = \mathbb{B}$ is an MCP

Shared constraints

- What if agents have shared knowledge?
- $K_i(x_{-i}) = \{x_i \in \mathbb{R}^{n_i} \mid (x_i, x_{-i}) \in X\}$, X convex

Theorem

If x^* is a solution to the $VI(X, F)$ with F above, then it is a solution to the $QVI(K, F)$, thus it is a solution to (GNEP) with the same assumptions on $f_i(\cdot)$ above. But, the converse may not hold.

- In this case, X is called a shared constraint.
- Examples: tragedy of commons example (shared capped channel), river basin pollution (total pollution constraint)

Variational equilibrium

Example

Find $(x_1^*, \ldots, x_N^*)$ satisfying

$$\begin{aligned} x_i^* \in \quad & \arg\min_{x_i} && f_i(x_i, x_{-i}^*), \\ & \text{subject to} && g_i(x_i, x_{-i}^*) \leq 0, \\ & && h(x_i, x_{-i}^*) \leq 0, \quad \text{for } i = 1, \ldots, N. \end{aligned}$$

There are two types of solutions when shared constraints are present. Let μ_i^* be a multiplier associated with the shared constraint $h(x)$ for agent i at the solution x^*. If $\mu_1^* = \cdots = \mu_N^*$, then we call the solution a *variational equilibrium*.

The name stems from the fact that if there are no $g_i(x)$'s, then x^* is a solution to the VI(X, F) and vice versa in theorem above, where $X = \{x \in \mathbb{R}^n \mid h(x) \leq 0\}$.

In other cases, we call a solution a GNEP equilibrium.

EMP formulation

```
loop(i,
  put / 'min', obj(i), x(i), deff(i), defg(i), defh;
);
```

To make it a variational equilibrium:

```
put / 'visol defh';
loop(i,
  put / 'min', obj(i), x(i), deff(i), defg(i), defh;
);
```

Shared variables

Definition

We call variable y an implicit variable if for each x there is at most one y satisfying $(y, x) \in X$. Here the set X is called the defining constraint of variable y.

Definition

In equilibrium problems, variables y_i are shared variables if

- The feasible region of agent $i = 1, \ldots, N$ is given by

$$K_i(x_{-i}) := \{(y_i, x_i) \in \mathbb{R}^{n_y \times n_i} \mid x_i \in X_i(x_{-i}), (y_i, x_i, x_{-i}) \in X\}$$

- y_i's are implicit variables with the same defining constraint X.

Recall

- In the EPEC problem of the multi-leader-follower game discussed above, the leaders share a single follower. The follower's optimal primal and dual variable values, say (y, λ), can be uniquely identified using the KKT conditions.
- From the KKT conditions, represent (y, λ) as functions of leader's variables x such that $y = h(x)$ and $\lambda = g(x)$.
- Replace follower's primal variable y with the function $h(x)$ and reduce the problem into a Nash game among the leaders.
- We may regard variables (y, λ) as shared variables.
- Using the `implicit` keyword, we can then represent the Nash game among leaders neatly without explicitly deriving $h(x)$ and $g(x)$.

EMP implementation

```
file empinfo / '%emp.info%' /;
put empinfo 'equilibrium' /;
put 'implicit y defH' /;
loop(i,
  put 'min', obj(i), x(i), y, deff(i) /;
);
```

MCP size of equilibrium problems containing shared variables by formulation strategy

Strategy	Size of the MCP
replication	$(n + 2mN)$
switching	$(n + mN + m)$
substitution (explicit)	$(n + m)$
substitution (implicit)	$(n + nm + m)$

$$F_i(z) = \begin{bmatrix} \nabla_{x_i} f_i(x,y) - (\nabla_{x_i} H(y,x))\mu_i \\ \nabla_{y_i} f_i(x,y) - (\nabla_{y_i} H(y,x))\mu_i \\ H(y_i, x) \end{bmatrix}, \quad z_i = \begin{bmatrix} x_i \\ y_i \\ \mu_i \end{bmatrix}.$$

EPEC: exchange model

- 23 agents (countries)
- maximize welfare with respect to economic variables and its strategic policy variables in the Nash way while trading goods with other agents subject to the general equilibrium conditions

$$\begin{array}{lll} \text{find} & (w^*, z^*, t^*) & \text{satisfying for } i = 1, \ldots, 23 : \\ (w^*, z^*, t_i^*) \in & \underset{w,z,t_i \in T_i}{\arg\max} & w_i, \\ & \text{subject to} & H(w, z, t) = 0 \end{array}$$

- w_i is a welfare index variable of agent i
- z is a vector of variables such as prices, quantities, and so on
- t_i is a vector of strategic variables of agent i (import tariffs)
- $H(\cdot) : \mathbb{R}^{253 \times 506} \to \mathbb{R}^{253}$ represents the general equilibrium conditions
- Applications: Brexit, modified GATT, Russian Sanctions

Model statistics and performance comparison of the EPEC

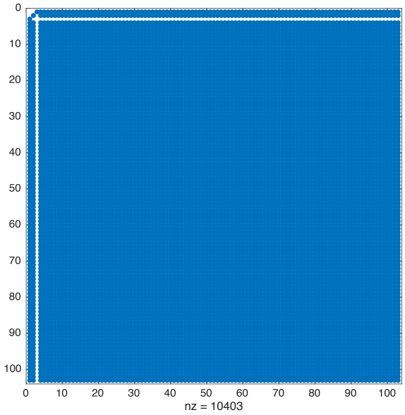
MCP statistics according to the shared variable formulation		
Replication	Switching	Substitution
12,144 rows/cols 544,019 non-zeros 0.37% dense	6,578 rows/cols 444,243 non-zeros 1.03% dense	129,030 rows/cols 3,561,521 non-zeros 0.02% dense

PATH configuration	Shared variable formulation (major, time)		
	Replication	Switching	Substitution
crash on, spacer step off prox on for switching	7 iters 8 secs	20 iters 22 secs	20 iters 406 secs
crash off, spacer step off prox on for switching	24 iters 376 secs	22 iters 19 secs	21 iters 395 secs
crash off, spacer step on prox off for all	8 iters 28 secs	8 iters 18 secs	8 iters 219 secs

Cournot Model: inverse demand function

- Cournot model: $|\mathcal{A}| = 5$
- Size $n = |\mathcal{A}| * N_a$

Size (n)	Time (secs)
1,000	35.4
2,500	294.8
5,000	1024.6

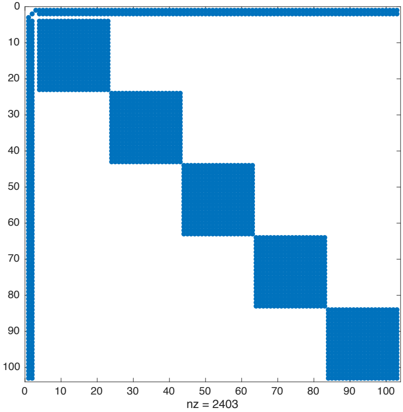


Jacobian nonzero pattern
$n = 100,\ N_a = 20$

Computation: implicit functions

- Use implicit fn: $z(x) = \sum_j x_j$
- Generalization to $F(z, x) = 0$ (via adjoints)
- empinfo: implicit z F

Size (n)	Time (secs)
1,000	2.0
2,500	8.7
5,000	38.8
10,000	$>$ 1080

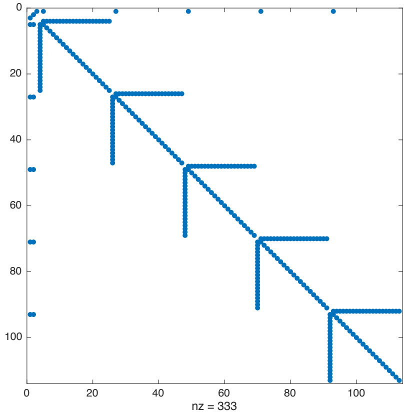


Jacobian nonzero pattern
$n = 100,\ N_a = 20$

Computation: implicit functions and local variables

- Use implicit fn: $z(x) = \sum_j x_j$ (and local aggregation)
- Generalization to $F(z, x) = 0$ (via adjoints)
- empinfo: implicit z F

Size (n)	Time (secs)
1,000	0.5
2,500	0.8
5,000	1.6
10,000	3.9
25,000	17.7
50,000	52.3



Jacobian nonzero pattern
$n = 100,\ N_a = 20$

Conclusions

- All results generalize to MOPEC (multiple optimization problems with equilibrium constraints)
- It's available (in GAMS)
- Enables modelers to convey structure to algorithms and allows algorithms to exploit this (who knows what and when)
- It's good
- Can use adjoint calculations
- New algorithms needed to further exploit structure

Spacer steps

- Given (x, y, μ) during iterations
- Compute a unique feasible pair $(\tilde{y}, \tilde{\mu})$
- Evaluate the residual at $(x, \tilde{y}, \tilde{\mu})$
- Choose the point if it has less residual than the one of (x, y, μ)