Penalty Methods for the Solution of Generalized Nash Equilibrium Problems

Francisco Facchinei
La Sapienza University of Rome

joint work with Christian Kanzow

Zürich, 6 April 2009

Outline

- Definition of the problem
- Review of algorithms
- The new penalty method

PART I

Definition of the problem

Optimization Problem

Optimization Problem

$$\begin{aligned} \min_x \quad & \theta(x) \\ & x \in X \end{aligned}$$

- One “optimizer”
- A single function to be optimized

Nash Equilibrium Problems I

Nash Equilibrium Problems (NEP)

- N players
- Each player controls $x^\nu \in \mathbb{R}^{n_\nu}$
- Set $n := \sum_{\nu=1}^{N} n_\nu, \quad \boldsymbol{x} := \begin{pmatrix} x^1 \\ \vdots \\ x^N \end{pmatrix} \in \Re^n, \quad \boldsymbol{x}^{-\nu} := \begin{pmatrix} x^1 \\ \vdots \\ x^{\nu-1} \\ x^{\nu+1} \\ \vdots \\ x^N \end{pmatrix}$
- $\boldsymbol{x} = (x^\nu, \boldsymbol{x}^{-\nu})$

Nash Equilibrium II

$$\begin{array}{lllllll} \min_{x^1} & \theta_1(x^1, \boldsymbol{x}^{-1}) & & \min_{x^\nu} & \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) & & \min_{x^N} & \theta_N(x^N, \boldsymbol{x}^{-N}) \\ & x^1 \in X_1 & \dots & & x^\nu \in X_\nu & \dots & & x^N \in X_N \end{array}$$

- Several "optimizers (or players)"
- Every player minimizes a different obj. f.
- The obj. f. depend on the variables of the other players
- The feasible sets are independent of the choices of the other players

Nash Equilibrium III

$S_\nu(\boldsymbol{x}^{-\nu})$: Optimal solution set of player ν for a given $\boldsymbol{x}^{-\nu}$ of the other players

$\boldsymbol{x}$ is a Nash equilibrium if
$x^\nu \in S_\nu(\boldsymbol{x}^{-\nu})$ for all players ν

No player can improve by unilaterally deviating from the current situation

Generalized Nash Equilibrium Problem (GNEP)

$$
\begin{array}{ccccc}
\min_{x^1} \;\; \theta_1(x^1, \boldsymbol{x}^{-1}) & & \min_{x^\nu} \;\; \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) & & \min_{x^N} \;\; \theta_N(x^N, \boldsymbol{x}^{-N}) \\
x^1 \in X_1(\boldsymbol{x}^{-1}) & \cdots & x^\nu \in X_\nu(\boldsymbol{x}^{-\nu}) & \cdots & x^N \in X^N(\boldsymbol{x}^{-N})
\end{array}
$$

- Several "optimizers (or players)"
- Every optimizer minimizes a different obj. f.
- The obj. f. depend on the variables of the other players
- Also the feasible sets depend on the other players' variables

Generalized Nash equilibrium II

$S_\nu(\boldsymbol{x}^{-\nu})$: Optimal solution set of player i for a given $\boldsymbol{x}^{-\nu}$ of the other players

$\boldsymbol{x}$ is a Nash equilibrium if

$x^\nu \in S_\nu(\boldsymbol{x}^{-\nu})$ for all players ν

Short History

- Prehistory:
  - Cournot (1838)
  - von Neumann (1928)
  - von Neumann and Morgenstern (1944)
- Nash (1950/1) $\Longrightarrow$ Nash Equilibrium Problem
- $\left.\begin{array}{l}\text{Debreu (1952)}\\ \text{Arrow and Debreu (1954)}\end{array}\right\} \Longrightarrow$ Generalized Nash Eq.
- 1954–beginning of the 1990s: The GNEP is studied mainly within the economic domain
- From the 1990s, modern engineering applications:
  - Robinson (1993, effectiveness in combat models)
  - Scotti (1995, structural design)
  - liberalized markets, telecommunications, web protocols, pollution analysis....

Arrow and Debreu's economic model

The economic equilibrium model: how are commodities produced and exchanged among individuals?

Walras was probably the first author to tackle this issue in a modern mathematical perspective.

Arrow and Debreu considered a general economic system along with a corresponding definition of equilibrium. They then showed that the equilibria of their model are those of a suitably defined GNEP; on this basis, they were able to prove important results on the existence of economic equilibria.

- l commodities
- s production units that control $y^j \in \Re^l$
- t consumption units that control $x^i \in \Re^l$
- One (fictious) "market" player that controls the prices $p \in \Re^l$

Arrow and Debreu's economic model II

- The production-player's problem:

$$\begin{array}{rl} \max_{y^j} & p^T y^j \\ s.t. & y^j \in Y^j \end{array}$$

- The consumption-player's problem:

$$\begin{array}{rl} \max_{x^i} & u_i(x^i) \\ s.t. & x^i \in X_i \\ & p^T x^i \le p^T \xi^i + \max\left(0, \sum_{j=1}^{s} \alpha_{ij}(p^T y^j)\right) \end{array}$$

- The market-players problem:

$$\begin{array}{rl} \max_{p} & p^T \left(\sum_{i=1}^{t} x^i - \sum_{j=1}^{s} y^j - \sum_{i=1}^{t} \xi^i\right) \\ s.t. & p \ge 0 \\ & \sum_{h=1}^{l} p_h = 1 \end{array}$$

PART II

Algorithms

A general scheme

Given a GNEP

$$\Downarrow$$

Transform it into another (better understood) problem X
Solve problem X

Two warnings

- In order to solve problem X one must impose some conditions; when these conditions are “brought back” to the original GNEP setting they are often very strong or not very often verified in practice (or both)
- There is too little practical experience in the solution of GNEPs to be able to make a serious comparison of the practical behavior of various methods

Conversion of a NEP to a VI

- $\theta_\nu(x^\nu, \boldsymbol{x}^{-\nu})$ convex in x^ν for every $\boldsymbol{x}^{-\nu}$
- θ_ν continuously differentiable
- X_ν closed and convex

Solve the NEP $\Leftrightarrow$ Solve $VI(K, F)$

$$K := \prod_{\nu=1}^N X_\nu, \quad F(\boldsymbol{x}) := \begin{pmatrix} \nabla_{x^1}\theta_1(\boldsymbol{x}) \\ \vdots \\ \nabla_{x^N}\theta_N(\boldsymbol{x}) \end{pmatrix}$$

Conversion of a GNEP to a QVI

- $\theta_\nu(x^\nu, \boldsymbol{x}^{-\nu})$ convex in x^ν for every $\boldsymbol{x}^{-\nu}$
- θ_ν continuously differentiable
- $X_\nu(\boldsymbol{x}^{-\nu})$ closed and convex

Solve the GNEP $\Leftrightarrow$ Solve $QVI(K, F)$

$$K(\boldsymbol{x}) := \prod_{\nu=1}^N X_\nu(\boldsymbol{x}^{-\nu}), \quad F(\boldsymbol{x}) := \begin{pmatrix} \nabla_{x^1}\theta_1(\boldsymbol{x}) \\ \vdots \\ \nabla_{x^N}\theta_N(\boldsymbol{x}) \end{pmatrix}$$

Solving the $QVI(K, F)$ means finding $\bar{\boldsymbol{x}} \in K(\bar{\boldsymbol{x}})$ s.t. $F(\bar{\boldsymbol{x}})^T(y - \bar{\boldsymbol{x}}) \geq 0, \forall y \in K(\bar{\boldsymbol{x}})$

Using the KKT conditions

Assume the feasible sets are explicitly given by a set of parametric inequalities

$$\begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) \\ & x^\nu \in X_\nu(\boldsymbol{x}^{-\nu}) \end{array} \quad \rightsquigarrow \quad \begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) \\ & g^\nu(x^\nu, \boldsymbol{x}^{-\nu}) \leq 0 \end{array}$$

We can write down the KKT conditions for each player

$$\begin{array}{rcl} \nabla_{x^\nu}\theta_\nu(x^\nu, \bar{\boldsymbol{x}}^{-\nu}) + \nabla_{x^\nu} g^\nu(x^\nu, \bar{\boldsymbol{x}}^{-\nu})^T \lambda^\nu & = & 0, \\ 0 \ \leq \ \lambda^\nu \ \perp \ -g^\nu(x^\nu, \bar{\boldsymbol{x}}^{-\nu}) & \geq & 0 \end{array}$$

- Concatenate the KKT systems of all players and try to solve
- Very recently some theoretical results based on an homothopy approach

Penalization methods

Assume that (as usual) some constraints do not depend on $\boldsymbol{x}^{-\nu}$,

$$\begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) \\ & g^\nu(x^\nu, \boldsymbol{x}^{-\nu}) \leq 0 \\ & h^\nu(x^\nu) \leq 0 \end{array} \quad \rightsquigarrow \quad \begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) + \rho P\left(g^\nu(x^\nu, \boldsymbol{x}^{-\nu})\right) \\ & h^\nu(x^\nu) \leq 0 \end{array}$$

where

$$P\left(g^\nu(x^\nu, \boldsymbol{x}^{-\nu})\right) = \begin{cases} \|\max(0, g^\nu(x^\nu, \boldsymbol{x}^{-\nu}))\|^2 & \text{sequential penalization} \\ \|\max(0, g^\nu(x^\nu, \boldsymbol{x}^{-\nu}))\| & \text{exact penalization} \end{cases}$$

- Solve the resulting NEP

The Jointly Convex Case

For an important class of GNEPs some other possibilities arise

Let $K \subset \Re^n$ be closed and convex

GNEP is jointly convex if $X_\nu(x^{-\nu}) := \{x^\nu : (x^\nu, x^{-\nu}) \in K\}$ or, equivalently, if

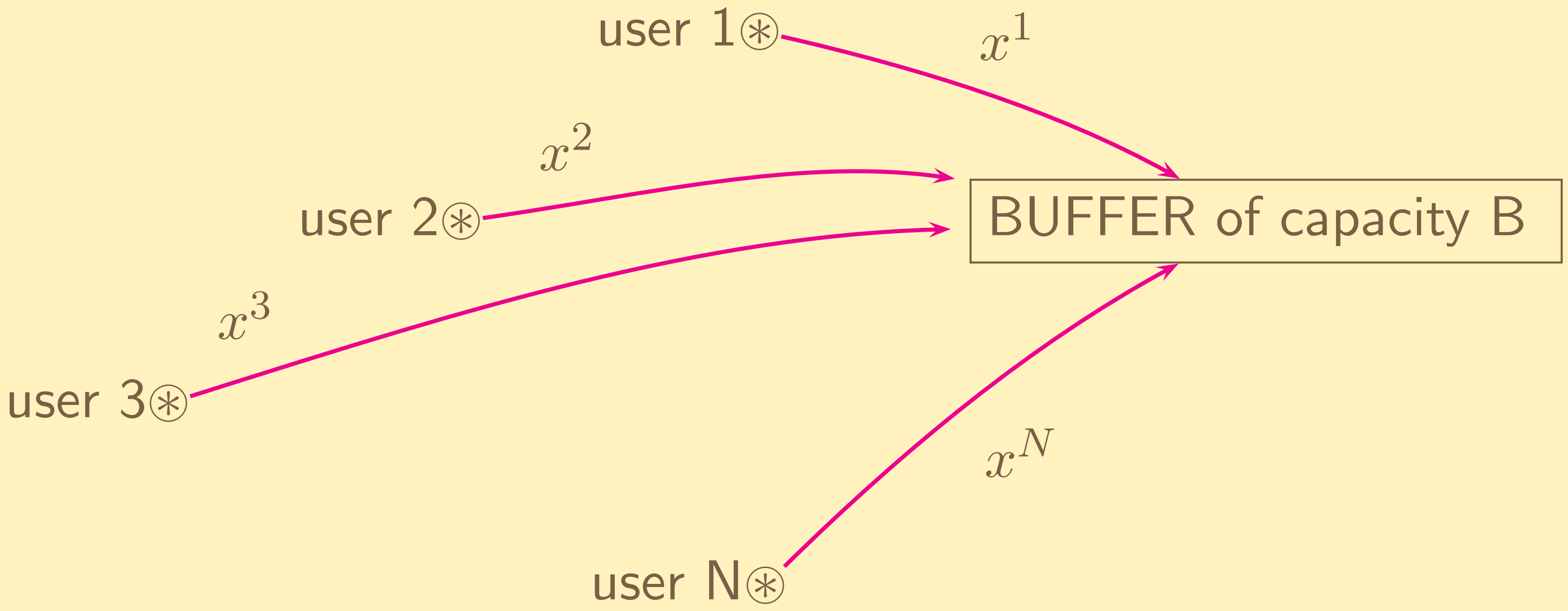
$$X_\nu(x^{-\nu}) = \{x^\nu : g(x^\nu, x^{-\nu}) \leq 0\} \text{ with } g(\cdot,\cdot) \text{ convex}$$

Solve the GNEP. $\Longleftarrow$ Solve $VI(K, F)$

Jointly Convex

The solutions of the VI are called normalized equilibria

An example: internet switching problem



$x^\nu =$ packets sent to buffer by player ν

maximize$_{x^\nu}$ $\quad \dfrac{x^\nu}{\sum_{\nu=1}^N x^\nu}\left(1 - \dfrac{\sum_{\nu=1}^N x^\nu}{B}\right)$

subject to $\quad x^\nu \geq 0,$

$$\sum_{\nu=1}^N x^\nu \leq B$$

$$\min_x \quad (x-1)^2 \qquad\qquad \min_y \quad (y-\tfrac{1}{2})^2$$
$$x + y \leq 1 \qquad\qquad x + y \leq 1.$$

Solutions: $(\alpha, 1-\alpha)$ for every $\alpha \in [1/2, 1]$

This is a jointly convex GNEP

$$K = \{(x,y) : x + y \leq 1\}, \quad F = \begin{pmatrix} 2x-2 \\ 2y-1 \end{pmatrix}$$

VI (K, F) is strongly monotone and its unique solution is $(3/4, 1/4)$; this is the normalized equilibrium of the GNEP.

Caution exemplified

Having reduced the (G)NEP to a VI we must pay attention.

The assumptions needed for the convergence of VI algorithms are often very strong (if not meaningless) when "translated" back into the (G)NEP environment
For example, a condition often invoked for the convergence of VI algorithms is that F be monotone. These means assuming that

$$JF(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial^2 \theta_1(\boldsymbol{x})}{\partial x^1 \partial x^1} & \cdots & \frac{\partial^2 \theta_1(\boldsymbol{x})}{\partial x^1 \partial x^\nu} & \cdots & \frac{\partial^2 \theta_1(\boldsymbol{x})}{\partial x^1 \partial x^N} \\ \vdots & & \vdots & & \vdots \\ \frac{\partial \theta_\nu(\boldsymbol{x})}{\partial x^\nu \partial x^1} & \cdots & \frac{\partial^2 \theta_\nu(\boldsymbol{x})}{\partial x^\nu \partial x^\nu} & \cdots & \frac{\partial^2 \theta_\nu(\boldsymbol{x})}{\partial x^\nu \partial x^N} \\ \vdots & & \vdots & & \vdots \\ \frac{\partial \theta_N(\boldsymbol{x})}{\partial x^N \partial x^1} & \cdots & \frac{\partial^2 \theta_N(\boldsymbol{x})}{\partial x^N \partial x^\nu} & \cdots & \frac{\partial^2 \theta_N(\boldsymbol{x})}{\partial x^N \partial x^N} \end{pmatrix}$$

be positive semidefinite. Is this sensible?

Nikaido-Isoda based methods

The mapping

$$\Psi(\boldsymbol{x}, \boldsymbol{y}) := \sum_{\nu=1}^{N} \left[\theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) - \theta_\nu(y^\nu, \boldsymbol{x}^{-\nu}) \right]$$

is called the *Nikaido-Isoda function*. Each summand gives the improvement that the ν-th player gets when changing (unilaterally) from x^ν to y^ν.
Define the nonnegative function

$$V(\boldsymbol{x}) := \sup_{\boldsymbol{y} \in K} \Psi(\boldsymbol{x}, \boldsymbol{y}) \geq 0$$

$\boldsymbol{x} \in K$ is a normalized equilibirum $\quad \Leftrightarrow \quad V(\boldsymbol{x}) = 0$

- Solve the problem $\min_{\boldsymbol{x} \in K} V(\boldsymbol{x})$ in order to get a normalized equilibrium

Note that in general V is nondifferentiable and difficult to compute

Specialized methods can be developed, the most important one being the relaxation method. Also regularized (differentiable) versions exist.

Relaxation method Pros:

- It is probably the method that has been used most in practice
- Conceptually does not require the differentiability of the θ_ν

Relaxation method Cons:

- Assumptions are still strong
- Computationally very intensive

	Ott.	Nash	Gen. Nash	
			"Jointly Convex"	General
Global Alg.	Yes	Conversion to VI	Yes, but ...	Penalties & this talk
Newton	Yes	Conversion to VI	Yes, but ...	Via KKT
Decomp.	Yes	Conversion to VI	Yes, but ...	Facchinei et al. (soon) Specific applications

In these cases caution must be exerted

PART III

The New Penalty Approach

Penalization again

$$\begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) \\ & g^\nu(x^\nu, \boldsymbol{x}^{-\nu}) \leq 0 \end{array} \quad \rightsquigarrow \quad \begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) + \rho_\nu \| g_+^\nu(x^\nu, \boldsymbol{x}^{-\nu}) \|_3 \\ & x^\nu \in \Re^{n_\nu} \end{array}$$

We set $P_\nu(x^\nu, \boldsymbol{x}^{-\nu}, \rho_\nu) = \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) + \rho_\nu \| g_+^\nu(x^\nu, \boldsymbol{x}^{-\nu}) \|_3$

We assume throughout "sufficient" differentiability of all functions involved

Important: the 3-norm is used, so that the penalty term is continuously differentiable at all infeasible points

Two main issues:

- How to choose and/or update the penalty parameter ρ
- How to solve the (unconstrained) penalized problems

We decouple completely these two issues.

How to update ρ I

In order to deal with the updating of the penalty parameter ρ we assume temporarily that an iterative algorithm $\mathcal{A}$ is available that, given a point $\boldsymbol{x}^k$, generates a new point $\boldsymbol{x}^{k+1} := \mathcal{A}[\boldsymbol{x}^k]$. We make the following absolutely natural blanket assumption on $\mathcal{A}$.

For every $\boldsymbol{x}^0$, the sequence $\{\boldsymbol{x}^k\}$ obtained by $\boldsymbol{x}^{k+1} = \mathcal{A}[\boldsymbol{x}^k]$ is such that every limit point (if any) is a solution of the unconstrained penalized problem.

All results on the updating below hold whatever the algorithm $\mathcal{A}$

How to update ρ II

If, for any value of the penalty parameters ρ_ν, we find a solution of the penalized problem that is also feasible for the original constrained problem, then this is a solution of original constrained problem. We try to force feasibility by increasing the penalty parameters.

If a solution $\bar{\boldsymbol{x}}$ of the penalty problem is not feasible for player ν, then $P_\nu(\boldsymbol{x}, \rho_\nu)$ is continuously differentiable at $\bar{\boldsymbol{x}}$ so that

$$\|\nabla_{x^\nu} \theta_\nu(\bar{x}^\nu, \bar{\boldsymbol{x}}^{-\nu})\| = \rho_\nu \left\| \nabla_{x^\nu} \|g_+^\nu(\bar{x}^\nu, \bar{\boldsymbol{x}}^{-\nu})\|_3 \right\|.$$

The idea of the updating scheme is to detect when this “dangerous” situation occurs (see the test at Step 2), and to increase the value of the penalty parameter in this case

How to update ρ III

Updating Scheme

Data: $\boldsymbol{x}^0 \in \mathbb{R}^n$ and $\rho_\nu > 0$ for all $\nu = 1, \ldots, N$. Set $k := 0$.

Step 1: If $\boldsymbol{x}^k$ is a solution of the GNEP: STOP.

Step 2: Let $I^k := \{\nu \mid (x^k)^\nu \notin X_\nu((\boldsymbol{x}^k)^{-\nu})\}$ (violated contraints)

For every $\nu \in I^k$, if

$$\|\nabla_{x^\nu} \theta_\nu((x^k)^\nu, (\boldsymbol{x}^k)^{-\nu})\| > 0.1 \left[\rho_\nu \left\| \nabla_{x^\nu} \|g_+^\nu((x^k)^\nu, (\boldsymbol{x}^k)^{-\nu})\|_3 \right\|\right],$$

then double the penalty parameters ρ_ν.

Step 3: Compute $\boldsymbol{x}^{k+1} = \mathcal{A}[\boldsymbol{x}^k]$, set $k \leftarrow k+1$, and go to Step 1.

How to update ρ IV

Let $\{\boldsymbol{x}^k\}$ be the sequence generated by Updating Scheme. If the penalty parameters are updated a finite number of times only, then every limit point $\bar{\boldsymbol{x}}$ of this sequence is a solution of the GNEP

If instead some penalty parameters grow to infinity and the sequence $\{\boldsymbol{x}^k\}$ is bounded, then, for each ν such that $\rho_\nu \to \infty$, there is a limit point $\bar{\boldsymbol{x}}$ for which one of the following assertions is true:

(a) $\bar{x}^\nu$ is a global minimizer of the constraint violation $\|g_+^\nu(\cdot, \bar{\boldsymbol{x}}^{-\nu})\|_3$ with $\|g_+^\nu(\bar{x}^\nu, \bar{\boldsymbol{x}}^{-\nu})\|_3 > 0$;

(b) $\bar{x}^\nu$ is Fritz John point for the player's problem, but not a solution of it

(c) $\bar{x}^\nu$ is an optimal solution for the player's problem.

Our next aim is giving conditions ensuring that (a), (b) and (c) cannot occur, so that the ρ_ν remain finite and every limit point is a solution of the GNEP

How to update ρ V

We need some constraint qualifications

- $\partial^*_{x^\nu} \|g^\nu_+(x^\nu, \boldsymbol{x}^{-\nu})\|_3 \quad := \quad \Big\{\xi \in \mathbb{R}^{n_\nu} \,\Big|\, \exists\{\boldsymbol{y}^k\}$ with $(y^k)^\nu$ not feasible for player ν such that $\{\boldsymbol{y}^k\} \to \boldsymbol{x}$ and $\nabla_{x^\nu} \|g^\nu_+((y^k)^\nu, (\boldsymbol{y}^k)^{-\nu})\|_3 \to \xi \Big\}$

We say that the GNEP satisfies the constraint qualification CQ$_3$ at a point $\bar{\boldsymbol{x}}$ if

$$0 \notin \partial^*_{x^\nu} \|g^\nu_+(\bar{x}^\nu, \bar{\boldsymbol{x}}^{-\nu})\|_3, \qquad \forall \nu = 1, \ldots, N$$

- We say that the GNEP satisfies the EMFCQ at a point $\bar{\boldsymbol{x}}$ if, for every player $\nu = 1, \ldots, N$, there exists a vector d^ν such that

$$\nabla_{x^\nu} g^\nu_i(\bar{x}^\nu, \bar{\boldsymbol{x}}^{-\nu})^T d^\nu < 0 \quad \forall i \in I^\nu_+(\bar{\boldsymbol{x}}),$$

where $I^\nu_+(\bar{\boldsymbol{x}}) := \Big\{i \in \{1, \ldots, m_\nu\} \,\Big|\, g^\nu_i(\bar{x}^\nu, \bar{\boldsymbol{x}}^{-\nu}) \geq 0\Big\}$

How to update ρ VI

Assume that the sequence $\{\boldsymbol{x}^k\}$ generated by Updating Scheme is bounded. Consider the following assertions:

(a) The EMFCQ holds at every limit point $\bar{\boldsymbol{x}}$ of $\{\boldsymbol{x}^k\}$;
(b) The CQ$_3$ condition holds at every limit point $\bar{\boldsymbol{x}}$ of $\{\boldsymbol{x}^k\}$;
(c) The penalty parameters are updated a finite number of times only

Then the following implications hold:

$$(a) \quad \Rightarrow \quad (b) \quad \Rightarrow \quad (c).$$

The algorithm $\mathcal{A}$ I

Our algorithm $\mathcal{A}$ is based on smoothing. Recall that the objectives functions of the penalized game are

$$
\begin{aligned}
P_\nu(\boldsymbol{x}, \rho_\nu) &= P_\nu(x^\nu, \boldsymbol{x}^{-\nu}, \rho_\nu) \\
&= \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) + \rho_\nu \| g_+^\nu(x^\nu, \boldsymbol{x}^{-\nu}) \|_3 \\
&= \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) + \rho_\nu \left(\sum_{i=1}^{m_\nu} \max\{0, g_i^\nu(x^\nu, \boldsymbol{x}^{-\nu})\}^3 \right)^{1/3}.
\end{aligned}
$$

We approximate these functions by the smooth mappings

$$
\begin{aligned}
\tilde{P}_\nu(\boldsymbol{x}, \rho_\nu, \varepsilon) &:= \tilde{P}_\nu(x^\nu, \boldsymbol{x}^{-\nu}, \rho_\nu, \varepsilon) \\
&:= \theta_\nu(x^\nu, \boldsymbol{x}^{-\nu}) + \rho_\nu \left(\sum_{i=1}^{m_\nu} \max\{0, g_i^\nu(x^\nu, \boldsymbol{x}^{-\nu})\}^3 + \varepsilon \right)^{1/3} \\
&\quad + \frac{\varepsilon}{2} \|x^\nu\|^2,
\end{aligned}
$$

where $\varepsilon > 0$

The algorithm $\mathcal{A}$ II

We approximate the nonsmooth penalized game by the unconstrained smooth (actually C^2) game where each player's problem is $\tilde{P}_\nu$ instead of P_ν. The solutions of this smoothed game are the solutions of the following system of equations:

$$F_\varepsilon(\boldsymbol{x}) := \begin{pmatrix} \nabla_{x^1} \tilde{P}_1(x^1, \boldsymbol{x}^{-1}, \rho_1, \varepsilon) \\ \vdots \\ \nabla_{x^N} \tilde{P}_N(x^N, \boldsymbol{x}^{-N}, \rho_N, \varepsilon) \end{pmatrix} = 0. \tag{1}$$

Let $\{\varepsilon_k\}$ and $\{\eta_k\}$ be two sequences of positive numbers converging to 0 and, for every k, let $\boldsymbol{x}(\varepsilon_k)$ be a point such that

$$\| F_{\varepsilon_k}(\boldsymbol{x}(\varepsilon_k)) \| \leq \eta_k.$$

Then every limit point of the sequence $\boldsymbol{x}(\varepsilon_k)$ is a solution of the nonsmooth penalized game

The algorithm $\mathcal{A}$ III

On this basis we can, roughly speaking, use any standard method for the solution of a smooth system of equations as algorithm $\mathcal{A}$.

The one theoretical problem we are left to deal with is "Can we ensure that we are able to solve the smooth systems $F_\varepsilon(\boldsymbol{x}) = 0$?"

We do not actually need to solve these systems at each iteration but eventually we want to get more and more accurate solutions. While the approach seems to work very well in practice, this theoretical issue is still under investigation.

The overall algorithm

Data: $\boldsymbol{x}^0 \in \mathbb{R}^n$, $\rho_\nu > 0$, $\varepsilon_0 > 0$, $S > 0$ and integer. Set $k := 0$.

Step 1: If $\boldsymbol{x}^k$ is such that

$$1.\ \|\max\{0, g(\boldsymbol{x}^k\}\| \leq 10^{-6}; \quad 2.\ \varepsilon^k \leq 10^{-6}; \qquad 3.\ \|F_{\varepsilon_k}(\boldsymbol{x})\| \leq 10^{-6};$$

then STOP.

Step 2: Let $I^k := \{\nu \,|\, (x^k)^\nu \notin X_\nu((\boldsymbol{x}^k)^{-\nu})\}$. For every $\nu \in I^k$, if

$$\|\nabla_{x^\nu}\theta_\nu((x^k)^\nu, (\boldsymbol{x}^k)^{-\nu})\| > 0.1\left[\rho_\nu \left\| \nabla_{x^\nu}\|g_+^\nu((x^k)^\nu, (\boldsymbol{x}^k)^{-\nu})\|_3 \right\|\right],$$

then double the penalty parameter ρ_ν.

Step 3: Perform at most S steps of an equation solver to the nonlinear system of equations $F_{\varepsilon_k}(\boldsymbol{x}) = 0$. Let $\boldsymbol{x}^{k+1}$ be the final iterate of this equation solver.
If $\|F_{\varepsilon_k}(\boldsymbol{x}^{k+1})\| \leq 1000\varepsilon_k$, set $\varepsilon_{k+1} = 0.1\varepsilon_k$, otherwise let $\varepsilon_{k+1} := \varepsilon_k$.
Set $k \leftarrow k+1$, and go to Step 1.

Numerical results: non jointly convex

Ex.	N	n	St.p.	k	i_{total}	ε_f	$\|F_{\varepsilon_f}(\boldsymbol{x}^f)\|$	$\|g_+(\boldsymbol{x}^f)\|$	$\rho^f_{\max}$
A1	10	10	0.01	5	34	1e-7	3.6e-13	7.6e-4	10
			0.1	5	33	1e-7	3.6e-13	7.6e-4	10
			1	5	30	1e-7	3.6e-13	7.6e-4	10
A2	10	10	0.01	6	46	1e-8	2.4e-12	6.2e-4	10
			0.1	6	44	1e-08	7.6e-13	9.1e-4	10
			1	F					
A3	3	7	0	3	4	1e-5	1.7e-11	0	1
			1	3	5	1e-5	2.3e-11	0	1
			10	3	8	1e-5	1.5e-11	0	10
A4	3	7	0	16	247	1e-10	1.4e-10	5.6e-4	1e+5
			1	29	510	1e-08	1.4e-10	5.6e-4	1e+5
			10	22	351	1e-9	1.6e-10	3.8e-4	1e+4
A5	3	7	0	3	43	1e-5	6.2e-6	4.3e-4	100
			1	3	43	1e-5	6.2e-6	4.3e-4	100
			10	5	62	1e-5	8.9e-16	4.3e-4	100
A6	3	7	0	10	88	1e-10	5.3e-9	3.5e-4	1e+4
			1	9	121	1e-9	4.7e-10	6.1e-4	1e+4
			10	10	99	1e-10	5.4e-9	3.5e-4	1e+4
A7	4	20	0	12	148	1e-9	3.6-10	0.001	1e+4
			1	12	151	1e-9	3.6e-10	0.001	1e+4
			10	F					
A8	3	3	0	5	81	1e-5	4.6e-13	6.3e-5	100
			1	4	31	1e-6	9.4e-14	2.0e-4	10
			10	F					
A9 a	7	56	0	6	169	1e-8	2.8e-11	3.7e-3	100
A9b	7	112	0	22	1034	1e-10	5.9e-8	9.6e-4	100
A10a	8	24		11	246	1e-9	6.4e-10	1.5e-3	1000
A10b	25	125		8	203	1e-8	3.1e-11	8.4e-3	100
A10c	37	222		101	14637	1e-8	4.8e-9	0.01	1e+4
A10d	37	370		11	1282	1e-7	2.3e-11	0.03	1000
A10e	48	576		12	1392	1e-7	1.6e-10	0.04	1000

Numerical results: jointly convex

Ex.	N	n	St.p.	k	i_{total}	ε_f	$\|F_{\varepsilon_f}(\boldsymbol{x}^f)\|$	$\|g_+(\boldsymbol{x}^f)\|$	$\rho^f_{\max}$
B1	2	2	0	9	60	1e-11	9.9e-12	9.6e-5	10
B2	2	2	(2, 0)	3	5	1e-5	3.6e-12	0	1
B3	3	3	0	10	84	1e-11	1.7e-9	1.5e-4	10
B4	10	10	0	3	16	1e-5	2.1e-10	0	100
B5	3	6	0	3	9	1e-5	4.7e-10	0	1
B6 (P=75)	5	5	10	10	102	1e-10	1.3e-7	3.9e-4	1000
B6 (P=100)	5	5	10	9	74	1e-10	1.7e-8	3.1e-4	1000
B6 (P=150)	5	5	10	10	101	1e-11	3.3e-8	2.9e-4	100
B6 (P=200)	5	5	10	10	73	1e-11	2.8e-8	2.3e-4	10
B7	2	3	0	9	71	1e-10	1.5e-10	1.9e-4	100
B8	2	12	0	19	346	1e-7	1.1e-7	0.001	1e+4
			1	12	182	1e-7	4.1e-7	5.2e-4	10e+4
			10	11	16	1e-7	4.4e-8	5.7e-4	1e+4

Numerical results for Problem B7

k	ρ_k	$(x_1^1)^k$	$(x_2^1)^k$	$(x_1^2)^k$	$\lVert \max\{0, g(\boldsymbol{x}^k)\}\rVert$	ε_k	$\lVert F_{\varepsilon_k}(\boldsymbol{x}^k)\rVert$	i_{total}
0	1	0.0000000	0.0000000	0.0000000	0.0000000	10^{-3}		
1	1	1.9482194	14.5776644	4.4138139	31.6005704	10^{-3}	0.000	7
2	10	0.3147472	10.6578689	7.5978619	0.0652462	10^{-4}	3.607	27
3	100	0.0012125	11.0019638	7.9970862	0.0186015	10^{-4}	0.000	36
4	100	0.0003897	11.0010847	7.9988202	0.0086351	10^{-5}	0.000	41
5	100	0.0001635	11.0005208	7.9994696	0.0040083	10^{-6}	0.000	47
6	100	0.0000742	11.0002434	7.9997555	0.0018605	10^{-7}	0.000	53
7	100	0.0000342	11.0001132	7.9998867	0.0008636	10^{-8}	0.000	59
8	100	0.0000158	11.0000525	7.9999474	0.0004008	10^{-9}	0.000	65
9	100	0.0000073	11.0000244	7.9999756	0.0001860	10^{-10}	0.000	71

Conclusions

The algorithm compares favourably to existing penalty methods (Fukushima Pang (2005), Facchinei Pang (2006), Fukushima (2008))

We have proposed, studied and tested a penalty method for the solution of GNEPs

The algorithms seems to perform very well in practice

We believe that what we report is by far the largest numerical testing in the literature (for general problems)

Some theoretical questions on the algorithm $\mathcal{A}$ require further study

Many variants are possible and we plan to explore them