A survey of GNE computation methods: theory and algorithms

Christophe Dutang

A survey of GNE computation methods: theory and algorithms

Christophe Dutang*

April 15, 2013

This paper deals with optimization methods solving the generalized Nash equilibrium problem (GNEP), which extends the standard Nash problem by allowing constraints. Two cases are considered: general GNEPs where constraint functions are individualized and jointly convex GNEPs where there is a common constraint function. Most recent methods are benchmarked against new methods. Numerical illustrations are proposed with the same software for a fair benchmark.

Keywords: Generalized Nash equilibrium problem; Semismooth equation; Fixed-point methods; Variational Inequality problem

Mathematics Subject Classification (2000): 90C30, 91A10, 91A80, 49M05

We consider a generalized game of N players characterized by their objective function $\theta_i : \mathbb{R}^n \mapsto \mathbb{R}$ and their constraint function $g^i : \mathbb{R}^n \mapsto \mathbb{R}^{m_i}$. The generalized Nash equilibrium problem (GNEP for short) consists in finding $x^\star$ such that for all $i = 1, \ldots, N$, $x_i^\star$ solves the subproblem P_i

$$\min_{x_i \in \mathbb{R}^{n_i}} \theta_i(x_i, x_{-i}^\star) \quad \text{s.t.} \quad g^i(x_i, x_{-i}^\star) \leq 0, \tag{1}$$

where (x_i, x_{-i}) denotes the vector $(x_1, \ldots, x_i, \ldots, x_N) \in \mathbb{R}^n$ with $n = \sum_i n_i$ the total number of variables and $m = \sum_i m_i$ the total number of constraints. Note that the dependency on $x_{-i}^\star$ makes all subproblem P_i interconnected. When players have a common constraint function $g^1 = \cdots = g^N = g$ and the feasible set $\{x, g(x) \leq 0\}$ is convex, the GNEP is called jointly convex GNEP. The GNEP extends standard Nash equilibrium, since ones' player strategy depends on the rival players' strategies. Thus, when each player's constraint function does not depend on the other players' strategies $g^i(x_i, x_{-i}) = g^i(x_i)$, the GNEP reduces to standard Nash equilibrium problem.

GNEP arises from many practical problems, including telecommunications (power allocation), engineering (energy market), economics (market model) and environmental (pollution) applications, see [18] and the references therein for an overview of GNEPs.

This paper aims to make a survey of computational methods to solve general GNEPs and jointly convex GNEPs defined in (1). New computational methods are also identified when doing

*C. Dutang, Institut de Recherche Mathématique Avancée, UMR 7501, Université de Strasbourg et CNRS, 7 rue René Descartes, F-67000 Strasbourg; Tel.: +33 3 68 85 01 86; email: dutang@math.unistra.fr

the survey process. A benchmark of these methods are provided using the R statistical software and the **GNE** package ([13]). The paper is organized as follows: Section 1 present the different reformulations of GNEPs. Section 2 describes the numerous optimization methods solving general GNEPs. Section 3 focuses on optimization methods solving jointly convex GNEPs, before Section 4 concludes.

1 Problem reformulations

As presented in (1), the generalized Nash equilibrium problem is not directly solvable, since each subproblem P_i depends on $x^\star_{-i}$. This section aims to present the different reformulations of the GNEP for which optimization methods have been proposed in the literature.

1.1 The KKT system

The first reformulation uses the Karush-Kuhn-Tucker (KKT) conditions of the N optimization subproblems. Both constraints and objective functions are thus assumed to be twice continuously differentiable C^2. Let $x^\star$ be a solution of the GNEP. If a constraint qualification holds for all players, then for all player i, there exists a Lagrange multiplier $\lambda^{i\star} \in \mathbb{R}^{m_i}$ such that

$$\begin{aligned} &\nabla_{x_i}\theta_i(x^\star) + \sum_{1\leq j\leq m_i} \lambda^{i\star}_j \nabla_{x_i} g^i_j(x^\star) = 0 \qquad (\in \mathbb{R}^{n_i}). \\ &0 \leq \lambda^{i\star},\ -g^i(x^\star) \geq 0,\ g^i(x^\star)^T\lambda^{i\star} = 0 \qquad (\in \mathbb{R}^{m_i}). \end{aligned}$$

Concatening the N subproblems, we get the following "extended" KKT system

$$D_L(x^\star, \lambda^\star) = 0, \quad \lambda^\star \geq 0, \quad g(x^\star) \leq 0, \quad \lambda^{\star T} g(x^\star) = 0, \tag{2}$$

where the functions D, g are defined as

$$D_L(x,\lambda) = \begin{pmatrix} \nabla_{x_1} L_1(x,\lambda^1) \\ \vdots \\ \nabla_{x_N} L_N(x,\lambda^N) \end{pmatrix}, \lambda = \begin{pmatrix} \lambda^1 \\ \vdots \\ \lambda^N \end{pmatrix}, g(x) = \begin{pmatrix} g^1(x) \\ \vdots \\ g^N(x) \end{pmatrix}, \tag{3}$$

and L_i is the Lagrangian function $L_i(x,\lambda^i) = \theta_i(x) + g^i(x)^T\lambda^i$ of the ith subproblem. The following theorem precises the necessary and sufficient conditions between the original GNEP in (1) and the KKT system in (2).

Theorem 1. *Let a GNEP with twice continuity and differentiability of objective and constraint functions.*

(i) If $x^\star$ solves the GNEP at which all the player's subproblems satisfy a constraint qualification, then there exists $\lambda^\star \in \mathbb{R}^m$ such that $x^\star, \lambda^\star$ solve equation 2.

(ii) If $x^\star, \lambda^\star$ solve equation 2 and that the functions θ_i's are player convex and $\{y_i,\ g^i(y_i, x_{-i}) \leq 0\}$ are closed convex sets, then $x^\star$ solves the original GNEP.

The previous theorem is reported in [18] and [16], respectively in Theorem 4.6 and Proposition 1. Using Fritz John conditions, see e.g. [52] or [2], the player convexity of θ_i in item (ii) can be relaxed to player-pseudoconvexity, i.e. $x_i \mapsto \theta_i(x)$ is pseudoconvex. Now, we present two approaches to solve the extended KKT system (2).

1.1.1 The complementarity reformulation

Let us start by the complementarity reformulation, which use complementarity functions as its name suggests.

Definition 1 (complementarity functions)**.** *A complementarity function $\phi : \mathbb{R}^2 \to \mathbb{R}$ is a function verifying the following property*

$$\phi(a, b) = 0 \Leftrightarrow a \geq 0, b \geq 0, ab = 0.$$

Examples of complementarity functions are $\phi_\wedge(a, b) = \min(a, b)$, $\phi_{FB}(a, b) = \sqrt{a^2 + b^2} - (a+b)$, see e.g. [20].

Proposition 1. *The complementarity reformulation of the KKT system (2) is*

$$\Phi(x, \lambda) = 0 \quad \textit{where} \quad \Phi(x, \lambda) = \begin{pmatrix} D_L(x, \lambda) \\ \phi_\circ(-g(x), \lambda) \end{pmatrix}, \tag{4}$$

where $\phi_\circ$ is the component-wise version of the complementarity function ϕ and D defined in (3).

This reformulation of the GNEP is given in [16], [18] and [12]. For a general discussion of semismooth reformulations of optimization problems, see [25].

1.1.2 The constrained equation reformulation

A second approach proposed by [12] in the context of GNEP to solve the KKT system is the reformulation via constrained equations of the type

$$H(z) = 0, z \in \Omega.$$

Proposition 2. *Let $w \in \mathbb{R}^m$ be slack variables (to transform inequality into equality). Let $\circ$ denote the component-wise product, i.e. $x \circ y = (x_1 y_1, \ldots, x_N y_N)$. The constrained equation reformulation of the KKT system (2) is*

$$H(x, \lambda, w) = 0, \quad (x, \lambda, w) \in \Omega, \quad \textit{and} \quad H(x, \lambda, w) = \begin{pmatrix} D_L(x, \lambda) \\ g(x) + w \\ \lambda \circ w \end{pmatrix}, \tag{5}$$

where $\Omega = \{(x, \lambda, w) \in \mathbb{R}^{n+2m}, \lambda \geq 0, w \geq 0\}$.

1.2 The QVI reformulation

The Variational Inequality (VI) and Quasi-Variational Inequality (QVI) problems have to be introduced in order to present the QVI reformulation.

Definition 2 (Variational Inequality)**.** *The Variational Inequality problem consists in finding $x \in K$ such that*

$$\forall y \in K, (y-x)^T F(x) \geq 0,$$

where $F : K \mapsto \mathbb{R}^n$ and is denoted VI(K, F(x)).

VI problems typically arise in minimization problems, see e.g. [19, 20]. Quasi-variational inequality problems are an extension of VI problems where the constraint set K depends on x.

Definition 3 (Quasi-Variational Inequality)**.** *The Quasi-Variational Inequality problem consists in finding x such that*

$$\forall y \in K(x), (y-x)^T F(x) \geq 0,$$

where $F : \mathbb{R}^n \mapsto \mathbb{R}^n$, $K : \mathbb{R}^n \mapsto 2^{\mathbb{R}^n}$ and is denoted by QVI(K(x), F(x)).

Note that a QVI has very complex structure since y must satisfy $y \in K(x)$ for a vector x we are looking for.

Assuming the differentiability of objective functions, the GNEP in (1) can be reformulated as a QVI problem

$$\forall y \in X(x), (y-x)^T F(x) \geq 0, \quad \text{with} \quad F(x) = \begin{pmatrix} \nabla_{x_1} \theta_1(x) \\ \vdots \\ \nabla_{x_N} \theta_N(x) \end{pmatrix}, \tag{6}$$

and a constrained set $X(x) = \{y \in \mathbb{R}^n, \forall i,\ g^i(y_i, x_{-i}) \leq 0\}$. The following theorem states the equivalence between the GNEP and the QVI, see Theorem 3.3 of [18] or Equation (3) [35] for a proof.

Theorem 2. *If objective functions are C^1 and player-convex, the action sets $X_\nu(x_{-\nu})$ are closed convex, then $x^\star$ solves the GNEP if and only if $x^\star$ solves the QVI$(X(x), F(x))$ defined in (6).*

Note that [1] also propose a QVI reformulation of the GNEP based on the convex hull of a normal cone (see Theorems 4.1 and 4.2) and establishe error bounds for the GNEP. For its simplicity, we restrict our survey to the QVI reformulation given in (6).

If the KKT conditions for the QVI problem are expressed, we naturally get back to Equation (2). According to [18], methods to solve general QVI problem arising from GNEP are still missing. We will see in Subsection 1.4, that QVI simplifies to a VI when dealing with jointly convex GNEP. In that case, it makes sense to use that reformulation. For the sake of completeness, we present below two approaches to solve the QVI arising from GNEP.

[24] propose to solve the QVI($X(x), F(x)$) as a sequence of penalized variational inequality problems VI($\tilde{X}, \tilde{F}_k$), where $\tilde{F}_k$ is defined as

$$\tilde{F}_k(x) = \begin{pmatrix} \nabla_{x_1}\theta_1(x) \\ \vdots \\ \nabla_{x_N}\theta_N(x) \end{pmatrix} + \begin{pmatrix} P_1(x) \\ \vdots \\ P_N(x) \end{pmatrix}, \tag{7}$$

with

$$P_\nu(x) = \sum_{i=1}^{m_\nu} \left(u_i^k + \rho_k g_i^\nu(x)\right)_+ \nabla_{x_\nu} g_i^\nu(x).$$

The set $\tilde{X}$ is either $\mathbb{R}^n$ or a box constraint set $[l, u]^n \subset \mathbb{R}^n$, $(\rho_k)_k$ an increasing sequence of penalty parameters and $(u_k)_k$ a bounded sequence. Theorem 3 of [24] shows the convergence of the VIP solutions $x_k^\star$ to a solution of the QVI under some smoothness conditions.

[35] propose to refomulate the QVI problem as a minimization of a (regularized) gap function, which is still as complex as the original GNEP.

Definition 4 (Regularized gap function)**.** *The regularized gap function of the QVI (6) is*

$$V_{QVI}(x) = \sup_{y \in X(x)} \psi_{\alpha VI}(x, y),$$

where $\psi_{\alpha VI}$ is given by

$$\psi_{\alpha VI}(x, y) = \begin{pmatrix} \nabla_{x_1}\theta_1(x) \\ \vdots \\ \nabla_{x_N}\theta_N(x) \end{pmatrix}^T (x - y) - \frac{\alpha}{2}||x - y||^2, \tag{8}$$

for a regularization parameter $\alpha > 0$.

Note that the minimisation problem appearing in the definition of V_{QVI} is a quadratic problem. The following theorem of [35] shows the equivalence a minimizer of V_{QVI} and the GNEP.

Theorem 3. *For each $x \in X(x)$, the regularized gap function V_{QVI} is non-negative $V_{QVI}(x) \geq 0$. If objective functions are continuous, then $x^\star$ solves the GNEP if and only if $x^\star$ is a minimum of V_{QVI} with*

$$V_{QVI}(x^\star) = 0 \text{ and } x^\star \in X(x^\star). \tag{9}$$

1.3 Nikaido-Isoda reformulation

We present a last reformulation of the GNEP called the Nikaido-Isoda (NI) reformulation, which was originally introduced in the context of standard Nash equilibrium problem by [40].

Definition 5 (Nikaido-Isoda function)**.** *For a game with objective functions θ_i, the Nikaido-Isoda function is defined as the function ψ from $\mathbb{R}^{2n}$ to $\mathbb{R}$ by*

$$\psi_{NI}(x, y) = \sum_{\nu=1}^{N} [\theta_\nu(x_\nu, x_{-\nu}) - \theta_\nu(y_\nu, x_{-\nu})]. \tag{10}$$

This function represents the sum of unilateral objective improvements between actions x and y.

In order to emphasize the links with the previous subsection, we introduce the corresponding gap function for the Nikaido-Isoda reformulation.

Definition 6 (gap Nikaido-Isoda function)**.** *The corresponding gap function is defined as*

$$V_{NI}(x) = \sup_{y \in X(x)} \psi_{NI}(x, y).$$

Theorem 3.2 of [18] shows the equivalence between GNEPs and the minimization of the gap Nikaido-Isoda function.

Theorem 4. *The function V_{NI} is such that $\forall x \in X(x), V_{NI}(x) \geq 0$. If objective functions θ_i are continuous, then $x^\star$ solves the GNEP if and only if $x^\star$ is a minimum of V_{NI} with*

$$V_{NI}(x^\star) = 0 \quad \textit{and} \quad x^\star \in X(x^\star), \tag{11}$$

where the set $X(x) = \{y \in \mathbb{R}^n, \forall i,\ g^i(y_i, x_{-i}) \leq 0\}$ and V_{NI} defined in (10).

There is no particular algorithm able to solve this problem for a general constrained set $X(x)$. As for the QVI reformulation, Equations (9) and (11) have a very complex structure due to the presence of $y \in X(x)$ in the definition of the gap function.

1.4 Jointly convex case

In this subsection, we present reformulations for a subclass of GNEP called jointly convex case. The jointly convex case is such that there exists a closed convex subset $X \subset \mathbb{R}^n$ verifying for all i,

$$X(x_{-i}) = \{x_i \in \mathbb{R}^{n_i}, (x_i, x_{-i}) \in X\}.$$

In our context parametrize context, the jointly convex setting requires that the constraint function is common to all players $g^1 = \cdots = g^N = g$ and

$$X = \{x \in \mathbb{R}^n, \forall i = 1, \ldots, N,\ g(x_i, x_{-i}) \leq 0\} \tag{12}$$

is convex. A sufficient condition is to require that g is quasi-convex with respect to all variables. However, it is generally assumed that g is convex with respect to all variables, e.g. in [59] and [60]. Therefore, the joint GNEP consists in finding $x^\star$ such that for all $i = 1, \ldots, N$, $x_i^\star$ solves the subproblem

$$\min_{x_i \in \mathbb{R}^{n_i}} \theta_i(x_i, x_{-i}^\star) \ \text{ s.t. } \ g(x_i, x_{-i}^\star) \leq 0. \tag{13}$$

1.4.1 KKT conditions for the jointly convex case

In the jointly convex case, the KKT conditions (2) become

$$\nabla_{x_i} \theta_i(x^\star) + \nabla_{x_i} g(x^\star) \lambda^{i\star} = 0,\ 0 \leq \lambda^{i\star},\ -g(x^\star) \geq 0,\ g(x^\star)^T \lambda^{i\star} = 0, \tag{14}$$

since the constraint function is common to all players. Note that there are (still) N Lagrange multipliers λ^i. Under the same condition as Subsection 1.1, a solution of this KKT system is also a solution of the original GNEP.

1.4.2 QVI formulation for the jointly convex case

As announced in the jointly convex case, the QVI reformulation (6) simplifies to a variational inequality problem VI(X, F)

$$\forall y \in X, (y-x)^T F(x) \geq 0, \quad \text{with} \quad F(x) = \begin{pmatrix} \nabla_{x_1}\theta_1(x) \\ \vdots \\ \nabla_{x_N}\theta_N(x) \end{pmatrix}, \tag{15}$$

where X is given in (12). But not all solutions of the GNEP are solutions of the $VI(X, F)$. In order to understand this feature, we just have to compare the KKT conditions of the VI (15) and the GNEP (14). This is summarized in the following theorem, see e.g. Theorem 3.9 of [18], Theorem 3.1 of [15].

Theorem 5. *Assuming that θ_i and g are C^1 functions, g is convex and θ_i is player-convex, the subset of generalized Nash equilibria verifying Equation (15) are the solutions of the KKT system (14) with a common multiplier $\lambda^1 = \cdots = \lambda^N = \lambda$.*

Therefore, GNEs verifying the VI problem (15) are subclass of GNE. They are called variational or normalized equilibria, see [36] for a detailed discussion of the VI representation of the QVI reformulation of GNEPs. Furthermore, the regularized gap function also simplifies and becomes

$$V_{\alpha VI}(x) = \sup_{y \in X} \psi_{\alpha VI}(x, y),$$

where $\psi_{\alpha VI}$ is in (8). Constrained equation (9) simplifies to a nonlinear equation $V_{\alpha VI}(x^\star) = 0$ and $x^\star \in X$. Using two regularization parameters $0 < \alpha < \beta$, $x^\star$ is the global minimum of the unconstrained minimization problem

$$\min_{x \in \mathbb{R}^n} V_{\alpha VI}(x) - V_{\beta VI}(x). \tag{16}$$

Furthermore, the VI reformulation leads to a fixed-point problem as shown in the following proposition.

Proposition 3. *Assuming that θ_i and g are C^1 functions, g is convex and θ_i player-convex, then $x^\star$ solves the VI (15) if and only if $x^\star$ is a fixed point of the function*

$$x \mapsto y_{VI}(x) = \arg\max_{y \in X} \psi_{\alpha VI}(x, y). \tag{17}$$

There is a unique maximiser of $\psi_{\alpha VI}$ with respect to y as $y \mapsto \psi_{\alpha VI}(x, y)$ is strictly concave.

Proof. A normalized equilibrium $x^\star$ verifies $V_{VI}(x^\star) = 0$ and $x^\star \in X$. By definition, for all $x \in X$, $V_{VI}(x) = \psi_{\alpha VI}(x, y_{VI}(x))$. Thus, $\psi_{\alpha VI}(x^\star, y_{VI}(x^\star)) = 0$. As $\psi_{\alpha VI}(x^\star, x^\star) = 0$, it follows that $y_{VI}(x^\star) = x^\star$. □

1.4.3 NIF formulation for the jointly convex case

In the jointly convex case, the NI reformulation (11) simplifies to a minimization problem of V_{NI} where

$$V_{NI}(x) = \sup_{y \in X} \psi_{NI}(x, y).$$

As in the previous subsection, solutions of this problem form a subset of GNEs, called normalized equilibrium. From [59], we have the following theorem.

Theorem 6. *Assuming θ_i and g are C^1 functions and g is convex and θ_i player-convex. $x^\star$ is a normalized equilibrium if and only if $x^\star$ solves $V_{NI}(x^\star) = 0$.*

From a mathematical point-of-view, computing V_{NI} is challenging as the supremum may be attained at more than one point. To deal with this problem, we use a regularized version of the Nikaido-Isoda function.

Definition 7 (regularized Nikaido-Isoda function)**.** *For a game with objective functions θ_i, the regularized Nikaido-Isoda function is defined as the function ψ from $\mathbb{R}^{2n}$ to $\mathbb{R}$ by*

$$\psi_{\alpha NI}(x, y) = \sum_{\nu=1}^{N} [\theta_\nu(x_\nu, x_{-\nu}) - \theta_\nu(y_\nu, x_{-\nu})] - \frac{\alpha}{2}||x - y||^2, \tag{18}$$

for a regularization parameter $\alpha > 0$.

The corresponding gap function is now defined as

$$V_{\alpha NI}(x) = \max_{y \in X} \psi_{\alpha NI}(x, y).$$

Since $y \mapsto \psi_{\alpha NI}(x, y)$ is strictly concave as long as objective functions θ_i are player-convex, the supremum is replaced by the maximum. Using two regularization parameters $0 < \alpha < \beta$, the constrained minimization problem can be further simplified to the unconstrained problem

$$\min_{x \in \mathbb{R}^n} V_{\alpha NI}(x) - V_{\beta NI}(x), \tag{19}$$

see [59].

As for the VI reformulation, a normalized equilibrium is also a fixed-point, see Property 3.4 of [59].

Proposition 4. *Assuming θ_i and g are C^1 functions and g is convex and θ_i player-convex. $x^\star$ is a normalized equilibrium if and only if $x^\star$ is a fixed-point of the function*

$$x \mapsto y_{NI}(x) = \arg\max_{y \in X} \psi_{\alpha NI}(x, y). \tag{20}$$

1.5 Scope of the paper

In this section, we present the four main reformulations of the GNEP : the complementarity reformulation and the constrained equation reformulation based on the KKT system, the QVI reformulation and the Nikaido-Isoda reformulation. The latter two are particularly useful for jointly convex GNEP, since in the general setting, no algorithms are yet available to solve optimization problems given in (6) and (11).

In the rest of the paper, we first focus on algorithms solving general GNEP based on the KKT system. Secondly, we turn our attention to algorithms solving jointly convex GNEP based on the QVI and the NIF reformulations. These two situations differs widely, since in the general GNEP, we have to solve a nonlinear equation, while for the jointly convex case, we solve a fixed point equation or a minimization problem.

2 Methods to solve general GNEP

In this section, we study optimization methods solving the KKT system : either the semismooth (nonlinear) equation (4) or the constrained equation (5). Optimization methods solving a smooth equation $F(z) = 0$ are first presented in Subsection 2.1. Then in Subsection 2.2, we consider the necessary extensions solving the semismooth equation (4) of the complementarity reformulation. We continue with specific algorithms solving the constrained equation (5) in Subsection 2.3. Finally, numerical illustrations to compare algorithms for both reformulations in Subsection 2.4 are carried out.

2.1 Classical methods for smooth nonlinear equations

As introduced in many optimization books, see e.g. [5, 11, 41], an optimization method to solve a nonlinear equation is made of two components: a local method and a globalization scheme. Assuming the initial point is not "far" from the root, local methods use a local approximation of the function (generally linear or quadratic approximation based on Taylor expansions) that is easier to solve in order to derive the sequence of iterates. The globalization studies adjustments to be carried out, so that the iterate sequence still converges when algorithms are badly initialized. In this subsection, we focus on local methods and refer to Appendix B for globalization techniques. We also give conditions for convergence to better emphasize the issues with the GNEP.

2.1.1 Local methods

Local methods are sequences $z_{k+1} = z_k + d_k$ where d_k is a root of a certain equation based on an approximation of the root function $F : \mathbb{R}^n \mapsto \mathbb{R}^n$. The first and the most simple method, the Newton method, uses a first-order Taylor expansion

$$F(z + h) = F(z) + J(z)h + o(h)\,,$$

where J denotes the Jacobian[1] $\operatorname{Jac} F$. Let $M_k^N(h) = F(z_k) + J(z_k)h$ be the model function. At the kth iterate, the Newton step consists in solving the system $M_k^N(d_k) = 0$. We get the following equation

$$J(z_k)d_k = -F(z_k). \tag{21}$$

Note that there is no guarantee that $J(z_k)$ is nonsingular.

Another method consists in replacing the Jacobian by an approximate matrix, which will always be invertible. The direction d_k solves

$$H_k d_k = -F(z_k), \tag{22}$$

where H_k is updated by a so-called quasi-Newton scheme. This is equivalent to a model function $M_k^{QN}(h) = F(z_k) + H_k h$. A quasi-Newton scheme needs an iterative process to update the approximate Jacobian from H_k to H_{k+1}. The choice of the matrix is large, since there are n^2 terms. We can assume first that

$$F(z_k) = M_{k+1}^{QN}(z_{k+1} - z_k) \Leftrightarrow H_{k+1}s_k = y_k,$$

where $s_k = z_{k+1} - z_k$ and $y_k = F(z_{k+1}) - F(z_k)$. The latter equation is called the secant equation. We could have considered a scheme to approximate directly the inverse of the Jacobian by W_k with the secant equation $s_k = W_{k+1}y_k$. But still, we have an underdetermined system of n equations for a $n \times n$ matrix.

Having no other property on the Jacobian of F (e.g. symmetry or positiveness), we generally consider the matrix that makes the smallest possible change to the preceding Jacobian according to the Frobenius norm[2], see [41, Chap. 11]. From an implementation point of view, the smallest possible change feature reduces the linear algebra computation work from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$ cost. If we consider minimizing $\min_H ||H - H_k||_F$ for all matrix H verifying the secant equation, we get the (good[3]) Broyden scheme

$$H_{k+1} = H_k + \frac{y_k - H_k s_k}{s_k^T s_k} s_k^T \Leftrightarrow W_{k+1} = W_k + \frac{(s_k - W_k y_k) y_k^T W_k}{s_k^T W_k y_k},$$

using the Sherman-Morrison formula. For a general discussion of quasi-Newton methods, we refer to [10] and for details on quasi-Newton methods tailored to nonlinear equations, see [6] or [41, Chap. 11].

Another way to solve the equation $F(z) = 0$ requires minimizing the norm $f(z) = \frac{1}{2}||F(z)||_2^2$. But not all minima of the function f are roots of F because we must have $f(z) = 0$. A widely known method for this least square problem is the Gauss-Newton method. With this method, we minimize the model function $\frac{1}{2}M_k^N(d_k)^T M_k^N(d_k)$ and get

$$J(z_k)^T J(z_k) d_k = -J(z_k)^T F(z_k). \tag{23}$$

To prevent the right-hand side matrix to be nonsingular, the Levenberg-Marquardt method modifies Equation (23) to

$$\left[J(z_k)^T J(z_k) + \lambda_k I\right] d_k = -J(z_k)^T F(z_k), \tag{24}$$

[1] As usual, $\operatorname{Jac} F$ denotes the Jacobian whereas ∇F the tranpose of the Jacobian.

[2] The Frobenius norm (also called the Euclidean norm) for matrix A is defined as $||A||_F = \sqrt{\sum_{i,j} |a_{ij}|^2}$.

[3] If we minimize $\min_W ||W - W_k||_F$ for all matrix W verifying the secant equation, then we will obtain the (bad) Broyden scheme. According to [6], this method appears often unsatisfactory in practice.

where I denotes the $n \times n$ identity matrix and $\lambda_k \geq 0$. Various choices of Levenberg-Marquart parameter are possible: [22] consider terms $\lambda_k = ||F(z_k)||_2^\delta$ with $\delta \in [1, 2]$. In practice, $\delta = 1$, i.e. $\lambda = ||F(z_k)||_2$, works much better than other δ's. We may also use $||J_k||$ or $\min(||F(z_k)||^2, ||J_k^T F(z_k)||^2)$. The Levenberg-Marquart is sometimes referred to as the modified Gauss-Newton method for its relation to the Gauss-Newton method.

Summarizing this first subsection, local methods consider sequences of the form $z_{k+1} = z_k + d_k$ where the direction d_k is a solution of one of the above equations. Newton direction uses Equation (21), quasi-Newton Equation (22) and Levenberg-Marquardt Equation (24).

2.1.2 Convergence in the differentiable setting

By construction, the Newton and the Levenberg-Marquardt methods require the root function F to be differentiable, while the Broyden method does not. Let us analyze the convergence conditions for these methods. We concatenate all theorems in one, see e.g. Theorems 11.2 and 11.5 of [41] for Newton and Broyden, respectively, Theorem 2.1 of [62] or Theorem 10 of [23] for Levenberg-Marquardt.

Theorem 7. *Suppose that F is continuously differentiable in a open convex set $O \subset\in \mathbb{R}^n$. Let $z^\star$ be a solution of $F(z) = 0$ and let $(z_k^N)_k$ be the sequence generated by the Newton method. If the Jacobian at the solution $J(z^\star)$ is nonsingular, then $(z_k^N)_k$ converges superlinearly to $z^\star$. If in addition, F is Lipschitz continuously differentiable*[4] *near $z^\star$, then the convergence rate is quadratic.*

Let $(z_k^B)_k$ be the sequence generated by the Broyden method. If for the starting points, there exist $\delta, \epsilon > 0$, such that

$$||z_0^B - z^\star|| \leq \delta, ||B_0 - J(z^\star)|| \leq \epsilon,$$

then $(z_k^B)_k$ converges superlinearly to $z^\star$.

Let $(z_k^{LM})_k$ be the sequence generated by the Levenberg-Marquardt method. If F is Lipschitz continuously differentiable on O and $||F(x)||$ provides a local error bound, then $(z_k^{LM})_k$ converges superlinearly to $z^\star$.

In other words, convergence theorems require (i) F is Lipschitz continuously differentiable (LC) [1] in an open convex set around a solution $z^\star$ and (ii) the Jacobian at the solution $J(z^\star)$ is nonsingular for Newton and Broyden method or the function F verifies a local error bound. Moreover, additional conditions may be required to ensure the convergence of globalized sequences, see Appendix B.4. These conditions can be weakened as shown in the next subsection.

2.2 Extension to a semismooth setting of SSR

In the two previous subsections, a blanket assumption of differentiability of the root function F is assumed. We describe the necessary adjustment for the GNEP context.

[4] that is F is continuously differentiable C^1 and the Jacobian is Lipschitzian.

2.2.1 Toward the non-differentiable setting

In the GNEP context (Equation (4) of Subsection 1.1.1), the root function is defined as

$$\Phi(x,\lambda) = \begin{pmatrix} D_L(x,\lambda) \\ \phi_\circ(-g(x),\lambda) \end{pmatrix},$$

where the first component $D_L(x,\lambda)$ is composed of N derivatives of the Lagrangian function $L_i(x,\lambda^i)$ and the second component $\phi_\circ(-g(x),\lambda)$ is the component-wise application of the complementarity function ϕ on the constraint function $g : \mathbb{R}^n \mapsto \mathbb{R}^m$. Firstly, the top component is differentiable when objective and constraint functions are C^2. Secondly, only the bottom component is not everywhere differentiable, because most complementarity functions $\phi(.,.)$ are non-differentiable at $(0,0)$. In this paper, we use the minimum function $\phi_\wedge(a,b) = \min(a,b)$ and the Fischer-Burmeister function $\phi_{FB}(a,b) = \sqrt{a^2+b^2} - (a+b)$, which are not differentiable at $(0,0)$.

To deal with non-differentiability, Clarke ([7]) introduced the generalized gradient. Few laters, its famous book [9] provides a comprehensive presentation of the nonsmooth analysis. We briefly present here some elements necessary to our paper and refer to Appendix C for further details.

Let $G : \mathbb{R}^n \mapsto \mathbb{R}^m$ a function with component G_j. By the Rademacher theorem, a locally Lipschitzian function is almost everywhere differentiable. We define first the limiting Jacobian, also called B(ouligand) subdifferential, denoted by $\partial_B G(x)$.

Definition 8 (limiting Jacobian)**.** *The limiting Jacobian of G at x is defined as*

$$\partial_B G(x) = \{V_x, \exists (x_k)_k \in D_G, x_k \to x, JacG(x_k) \to V_x\},$$

where D_G is the differentiability set of G.

For his desirable properties (such as convexity), the Clarke's generalized Jacobian, based on the limiting Jacobian, is commonly used.

Definition 9 (generalized Jacobian)**.** *The generalized Jacobian $\partial G(x)$ of G at x is the convex hull of the limiting Jacobian $\partial_B G(x)$*[5]*.*

Example 1. *In the Example 7.1.2 of [20], a function $G : \mathbb{R}^2 \mapsto \mathbb{R}^2$ is defined with components $G_1(x) = \min(x_1, x_2)$ and $G_2(x) = |x_1|^3 - x_2$. G is not differentiable at (0,0). By splitting $\mathbb{R}^2$ into 4 parts, we can compute the limiting Jacobian*

$$\partial_B G(0,0) = \left\{ \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix} \right\} \text{ and } \partial G(0,0) = \left\{ \begin{pmatrix} \lambda & 1-\lambda \\ 0 & -1 \end{pmatrix}, \lambda \in [0,1] \right\}.$$

From a computational point of view, using a component-wise version of the generalized Jacobian $\partial G(x) \subset \partial G_1(x) \times \cdots \times \partial G_m(x)$ might be useful, see Appendix C.

Now, we focus on the properties of the generalized Jacobian and a more flexible criterion than continuous differentiability : the semismoothness. Before introducing the semismoothness, we have to present directional derivatives.

[5]Note that if $m = 1$, the generalized Jacobian reduces to the generalized gradient, see Theorem 2.5.1 of [9] for this characterization of the generalized gradient.

Definition 10 (directional derivative)**.** *For a locally Lipschitzian function G, the (classical) directional derivative at x along $h \in \mathbb{R}^n$ is defined as*

$$G'(x;h) = \lim_{t \downarrow 0} \frac{G(x+th) - G(x)}{t}.$$

The Hadamard directional derivative is

$$G'_H(x;h) = \lim_{h' \to h, t \downarrow 0} \frac{G(x+th') - G(x)}{t}.$$

The difference between classical and Hadamard directional derivatives lies the fact that we look at all directions $h' \to h$ and not only at h.

Definition 11 (semismooth)**.** *A locally Lipschitzian function G is semismooth at x, if for all $d \in \mathbb{R}^n$, the following limit exists*

$$\lim_{\substack{V \in \partial G(x+t\tilde{d}) \\ \tilde{d} \to d, t \downarrow 0}} V\tilde{d}.$$

In this case, this limit equals to $G'_H(x;h)$.

Therefore, the semismoothness of function H at x requires the Hadamard directional derivative at x to exists along any direction converging to h and not only along h. Finally, we introduce the strong semismoothness, also called 1-order semismoothness, that will be used in convergence theorems.

Definition 12 (strongly semismooth)**.** *A locally Lipschitzian function G is strongly semismooth at x if for all $d \to 0$ and $V \in \partial G(x+d)$, we have*

$$Vd - G'(x,d) = \mathcal{O}\left(||d||^2\right).$$

Examples of semismooth functions are smooth, convex and piecewise linear functions. Furthermore, composite, scalar products, sums, minimum, maximum of semismooth functions are semismooth. Let us study our complementarity functions $\phi_\wedge$ and ϕ_{FB}.

Example 2. *The non-differentiability breakpoint is at $(0,0)$. By standard calculations, the directional derivate of $\phi_\wedge$ at this point is $\phi_\wedge((0,0);(a,b)) = \min(a,b)$. Futhermore, for all nonzero vector (a,b), $\phi_\wedge$ is differentiable and $\nabla\phi_\wedge$ is given by*

$$\nabla\phi_\wedge(a,b) = \begin{pmatrix} \mathbb{1}_{a \leq b} \\ \mathbb{1}_{a > b} \end{pmatrix}.$$

We deduce that

$$\partial_B \phi_\wedge(0,0) = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} \quad \textit{and} \quad \partial\phi_\wedge(0,0) = \left\{ \begin{pmatrix} \lambda \\ 1-\lambda \end{pmatrix}, \lambda \in [0,1] \right\}.$$

Furthermore, for all $V \in \partial\phi_\wedge(c,d)$, such that $(c,d) \to (a,b)$, we have

$$V \begin{pmatrix} a \\ b \end{pmatrix} - \phi_\wedge((0,0);(a,b)) = a\,\mathbb{1}_{c \leq d} + b\,\mathbb{1}_{c > d} - \min(a,b) = o((a,b)). \tag{25}$$

Using Appendix C, we conclude that $\phi_\wedge$ is semismooth at (0,0) but not strongly semismooth.

Example 3. *For ϕ_{FB}, the non-differentiability breakpoint is also at $(0,0)$. In fact, by standard calculations, for all nonzero vector (a,b), we have*

$$\nabla\phi_{FB}(a,b) = \begin{pmatrix} \frac{a}{\sqrt{a^2+b^2}} - 1 \\ \frac{b}{\sqrt{a^2+b^2}} - 1 \end{pmatrix}.$$

We deduce that $\partial_B\phi_{FB}(0,0) = \{\nabla\phi_{FB}(a,b), (a,b) \neq (0,0)\}$ and $\partial\phi_{FB}(0,0) = \bar{B}((-1,-1),1)$, where $\bar{B}(x,\rho)$ denotes the closed ball at x of radius ρ.

Furthermore, for all $V \in \partial\phi_{FB}(c,d)$, such that $(c,d) \to (a,b)$, we have

$$V\begin{pmatrix} a \\ b \end{pmatrix} = \phi_{FB}(a,b) \quad \textit{and} \quad \phi_{FB}((0,0);(a,b)) = \phi_{FB}(a,b).$$

Hence, we get

$$V\begin{pmatrix} a \\ b \end{pmatrix} - \phi_{FB}((0,0);(a,b)) = 0.$$

We conclude that ϕ_{FB} is strongly semismooth at $(0,0)$, as proved in [33].

Now, we can express appropriately the generalized Jacobian of the GNEP. We denote by $J_\Phi(z)$ elements of the generalized Jacobian $\partial\Phi(z)$. Using chain rules and previous definitions, we get

$$J_\Phi(z) = \begin{pmatrix} \mathrm{Jac}_x D_L(x,\lambda) & \mathrm{diag}\left[\left(\nabla_{x_i} g^i(x)\right)_i\right] \\ -D_a(z)\mathrm{Jac}_x g(x) & D_b(z) \end{pmatrix}, \tag{26}$$

where Jac_x denotes the Jacobian with respect to x and diag[...] represents a block diagonal matrix. The diagonal matrices D_a and D_b are given by

$$D_a(z) = \mathrm{diag}[a^1(x,\lambda^1), \ldots, a^N(x,\lambda^N)]\ ,\ D_b(z) = \mathrm{diag}[b^1(x,\lambda^1), \ldots, b^N(x,\lambda^N)],$$

with $a^i(x,\lambda^i), b^i(x,\lambda^i) \in \mathbb{R}^{m_i}$ defined by

$$(a^i_j(x,\lambda^i_j), b^i_j(x,\lambda^i_j)) = \begin{cases} \left(\phi'_a(-g^i_j(x),\lambda^i_j), \phi'_b(-g^i_j(x),\lambda^i_j)\right) & \text{if } (-g^i_j(x),\lambda^i_j) \neq (0,0), \\ (\xi_{ij},\zeta_{ij}) & \text{if } (-g^i_j(x),\lambda^i_j) = (0,0), \end{cases}$$

where ϕ'_a (resp. ϕ'_b) denotes the derivative of ϕ with respect to the first (resp. second) argument a (resp. b) and $(\xi_{ij},\zeta_{ij}) \in \bar{B}(p_\phi, c_\phi)$. See Appendix A for a detailed representation of the generalized Jacobian J_Φ.

Let us specify the parameters p_ϕ and c_ϕ for the two considered complementarity functions: for the minimum function, $p_\phi = (1/2, 1/2)$ and $c_\phi = 1/2$ and for the Fischer-Burmeister function, $p_\phi = (-1,-1)$ and $c_\phi = 1$. By standard chain rules, we get the following proposition.

Proposition 5. *Consider $\phi = \phi_\wedge$ or ϕ_{FB}. If functions θ_i and g_i are C^2, then the root function Φ defined in (26) is C^1 except at points (x,λ) such that $g^i_j(x) = \lambda^i_j = 0$. At these points, when $\phi = \phi_\wedge$, Φ is semismooth, while for $\phi = \phi_{FB}$, Φ is strongly semismooth.*

2.2.2 Local convergence in the semismooth framework

Now, we get back to the general equation $F(z) = 0$. As the Jacobian of the root function is not available, the direction computation of local methods presented in Subsection 2.1.1 must be adapted.

The solution consists in replacing the Jacobian by an element of the generalized Jacobian. Let $J_k \in \partial F(z_k)$. Considering the Newton method (21), we now solve

$$J_k d_k = -F(z_k), \tag{27}$$

whereas for the Levenberg-Marquardt method (24), the direction solves

$$\left[J_k^T J_k + \lambda_k I\right] d_k = -J_k^T F(z_k). \tag{28}$$

Corresponding sequences are called respectively the generalized Newton and the generalized Levenberg-Marquardt methods. For the quasi-Newton framework, there is no modification. Some authors also use the B-subdifferential $\partial_B F(z_k)$ or the component wise B-subdifferential $\partial_B F_1(z_k) \times \cdots \times \partial_B F_n(z_k)$ instead of the generalized Jacobian in (27) and (28).

Now, theorems extending Theorem 7 to semismooth functions are presented. The first extension is due to [47], cf. Theorem 3.2, for the generalized Newton method. We give below a slightly more general version[6] than the original version.

Theorem 8. *Let $z^\star$ a solution of $F(z^\star) = 0$. If F is locally Lipschitzian and semismooth at $z^\star$ and all elements[7] $J^\star \in \partial_B F(z^\star)$ are nonsingular, then the generalized Newton method is well defined and converges superlinearly to $z^\star$. If in addition, F is strongly semismooth at $z^\star$, then the convergence rate is quadratic.*

For the quasi-Newton approach, convergence results have been proposed in the literature, e.g. [27] and [44], where the differentiability is needed at a solution rather than in an open convex. [38] give a minimal condition (lesser than semismoothness) for a general quasi-Newton method to converge linearly.

As in the differentiable setting, e.g. [11], the convergence of quasi-Newton methods for semismooth functions is done in two steps: (i) a first theorem gives conditions of local linear convergence based on the limited difference between approximate Jacobian and elements in the generalized Jacobian, and (ii) a second theorem gives an additional condition for a general quasi-Newton method to converge superlinearly. We report here Theorems 4.1 and 4.2 of [53].

Theorem 9. *Let $z^\star$ a solution of $F(z^\star) = 0$. If F is locally Lipschitzian in the open convex $D \subset \mathbb{R}^n$ such as $z^\star \in D$. Consider the sequence $z_0 \in D$ and $z_{k+1} = z_k - V_k^{-1} F(z_k)$ with V_k a $n \times n$ matrix updated by a quasi-Newton scheme.*
Suppose F is semismooth at $z^\star$ and for all $J^\star \in \partial_b F(x^\star)$ are nonsingular. There exist constant $\epsilon, \Delta > 0$ such that if $||z_0 - z^\star|| \leq \epsilon$ and there exists $W_k \in \partial_b F(z_k)$ such that $||V_k - W_k|| \leq \Delta$, then the quasi-Newton sequence is well defined and converges linearly to $z^\star$.

[6] A version of the previous theorem exists when the generalized Newton method use the limiting Jacobian $J_k \in \partial_B F(z_k)$ (instead of the generalized Jacobian) in Equation (27), see e.g. [53], [32] or [16].

[7] Originally, [47] use the generalized Jacobian and not the limiting Jacobian. But as mentioned in [45] and [43], there is a weaker condition for superlinear convergence to hold, that is all elements in the limiting Jacobian are nonsingular.

Theorem 10. *Let F be locally Lipschitzian in a open convex $D \subset \mathbb{R}^n$. Assume F is semismooth and $\forall J^\star \in \partial_b F(z^\star)$ are nonsingular. Consider a sequence of nonsingular matrices V_k and points $z_{k+1} = z_k - V_k^{-1}F(z_k)$. If $(z_k)_k$ converges to $z^\star$ then $(z_k)_k$ converges superlinearly to $z^\star$ and $F(z^\star) = 0$ is equivalent to $\exists W_k \in \partial_b F(z_k)$,*

$$\lim_{k \to +\infty} \frac{||(V_k - W_k)s_k||}{||s_k||} = 0,$$

with $s_k = z_{k+1} - z_k$.

Local convergence of the generalized Levenberg-Marquardt method is studied in Theorem 6 of [17].

Theorem 11. *Let $z^\star$ a solution of $F(z^\star) = 0$, with F a locally Lipschitzian, semismooth at $z^\star$ and $\forall J^\star \in \partial_B F(z^\star)$ are nonsingular. If the Levenberg-Marquardt parameters $(\lambda_k)_k$ converge to 0, then the (generalized) Levenberg-Marquardt converges superlinearly to $z^\star$. If in addition, F is strongly semismooth and locally directionnally differentiable at $z^\star$, and $\lambda_k = \mathcal{O}(||J_k g_k||)$ or $\lambda_k = \mathcal{O}(||g_k||)$, then the sequence converges quadratically.*

2.2.3 Analysis of assumptions

We analyze in detail the assumptions of preceding theorems. All theorems require the root function to be semismooth at a solution $z^\star$. By Proposition 5, the function Φ defined in Equation (4) is semismooth. If $\phi = \phi_{FB}$, Φ is even strongly semismooth and thus improving the convergence rate.

The nonsingularity condition is required only for elements $J^\star$ of the limiting Jacobian at a solution $z^\star$. As analyzed in [16], the limiting Jacobian (26) might have some identical rows at the bottom part[8]. Let us investigate first this issue[9]. As only terms (ξ_{ij}, ζ_{ij}) in diagonal matrices D^a and D^b change between the generalized and the limiting Jacobian of Φ, we study the nonsingularity condition directly on the generalized Jacobian. In a detailed form, the bottom part has the following structure

$$\left(\begin{array}{ccc|ccc} -D_1^a(x,\lambda^1)\mathrm{Jac}_{x_1} g^1(x) & \dots & -D_1^a(x,\lambda^1)\mathrm{Jac}_{x_N} g^1(x) & D_1^b(x,\lambda^1) & & 0 \\ \vdots & & \vdots & & \ddots & \\ -D_N^a(x,\lambda^N)\mathrm{Jac}_{x_1} g^N(x) & \dots & -D_N^a(x,\lambda^N)\mathrm{Jac}_{x_N} g^N(x) & 0 & & D_N^b(x,\lambda^N) \end{array}\right) \quad (29)$$

where D_i^a and D_i^b are $m_i \times m_i$ diagonal matrices. In the following, we denote by D^a-part and D^b-part the left-hand and right-hand sides of Equation (29).

Proposition 6. *For $\phi = \phi_\wedge$ or ϕ_{FB}, the bottom part of generalized Jacobian of Φ defined in (4) has two identical rows if and only if there are two shared and active constraints.*

Proof. Assume that the generalized Jacobian has two identical rows, say for players i and $\tilde{i}$ and components j_i and $j_{\tilde{i}}$. The D^b-part requires that the j_ith row of D_i^b and the $j_{\tilde{i}}$th row of $D_{\tilde{i}}^b$ equals zero

$$b_j^i(x, \lambda_j^i) = b_{j_{\tilde{i}}}^{\tilde{i}}(x, \lambda_{j_{\tilde{i}}}^{\tilde{i}}) = 0, \quad (30)$$

[8]For the top part, there are less problem in general. As shown in Appendix A, matrices are less sparse: identical rows implies that the constraint functions are linear and two objective derivatives are identical.

[9]For jointly convex GNEP, [28] avoid this issue by considering only smooth parts of (4).

with $b_j^i(x,\lambda_j^i) = \phi_b'(-g_j^i(x),\lambda_j^i)$. Identical rows in the D^a-part is equivalent to the n dimensional equation

$$a_j^i(x,\lambda_j^i)\mathrm{Jac}_x g_j^i(x) = a_{j_{\tilde{i}}}^{\tilde{i}}(x,\lambda_{j_{\tilde{i}}}^{\tilde{i}})\mathrm{Jac}_x g_{j_{\tilde{i}}}^{\tilde{i}}(x). \tag{31}$$

If $\phi = \phi_\wedge$, then Equation (30) leads to

$$g_j^i(x) < \lambda_j^i \text{ and } g_{j_{\tilde{i}}}^{\tilde{i}}(x) < \lambda_{j_{\tilde{i}}}^{\tilde{i}}.$$

Since at a solution $z^\star$ both $g_j^i(x)$ and λ_j^i are nonnegative and (at least) one is zero. Hence, $g_j^i(x) = g_{j_{\tilde{i}}}^{\tilde{i}}(x) = 0$, i.e. active constraints. For the D^a-part using $\phi'_{\wedge a}(a,b) = \mathbb{1}_{a<b}$, Equation (31) leads to $a_j^i(x,\lambda_j^i) = a_{j_{\tilde{i}}}^{\tilde{i}}(x,\lambda_{j_{\tilde{i}}}^{\tilde{i}}) = 1$. Therefore, $\mathrm{Jac}_x g_j^i$ and $\mathrm{Jac}_x g_{j_{\tilde{i}}}^{\tilde{i}}$ are identical. So if the constraints $g_j^i, g_{j_{\tilde{i}}}^{\tilde{i}}$ are active and shared, then both Equations (30) and (31) are satisfied, i.e. the generalized Jacobian has two identical (null) rows.

If $\phi = \phi_{FB}$, then Equation (30) is equivalent to

$$g_j^i(x) = 0 \;\text{ and }\; \lambda_j^i > 0,$$

and the same for $\tilde{j}$. This is equivalent to $g_j^i, g_{j_{\tilde{i}}}^{\tilde{i}}$ are strongly active constraints. These conditions are such that $a_j^i(x,\lambda_j^i)$ and $a_{j_{\tilde{i}}}^{\tilde{i}}(x,\lambda_{j_{\tilde{i}}}^{\tilde{i}})$ are non-zero and expressed as $a_j^i(x,\lambda_j^i) = a_{j_{\tilde{i}}}^{\tilde{i}}(x,\lambda_{j_{\tilde{i}}}^{\tilde{i}}) = -1$. Equation (31) again requires that the constraint functions g_j^i and $g_{j_{\tilde{i}}}^{\tilde{i}}$ have identical derivatives. □

If we exclude the pathological case described above, the nonsingularity of limiting Jacobian[10] (26) requires that $J^\star$ has a non-zero determinant. Using the determinant formula for partitionned matrix and the Schur complement, the nonsingularity condition can be expressed in two different formulations

(i) that $\mathrm{Jac}_x D_L(x,\lambda)$ and

$$D_b(z) + D_a(z)\mathrm{Jac}_x g(x)\mathrm{Jac}_x D_L(x,\lambda)^{-1}\mathrm{diag}\left[\left(\nabla_{x_i} g^i(x)\right)_i\right] \tag{32}$$

are nonsingular,

(ii) $D_b(z)$ and

$$\mathrm{Jac}_x D_L(x,\lambda) + \mathrm{diag}\left[\left(\nabla_{x_i} g^i(x)\right)_i\right] D_b(z)^{-1} D_a(z)\mathrm{Jac}_x g(x) \tag{33}$$

are nonsingular.

Note, since $D_b(z)$ is a diagonal matrix, its nonsingularity is equivalent to have non-null terms. We will see in Subsection 2.2 that (32) is a typical condition. Following the above discussion, we conclude that both the generalized Newton and the generalized Levenberg-Marquardt method converge superlinearly to a solution $z^\star$ if one of the two conditions (32) and (33) are satisfied and no constraint functions are shared.

Finally, we focus on the convergence of the Broyden method. Theorems 9 and 10 of [53] give minimal conditions for a quasi-Newton method to converge (superlinearly). In [53] (or also in [45]),

[10] We do not have to require all elements of ∂F but only elements of $\partial_B F$.

the Broyden method is used on smooth parts of the root function F. When applying the Broyden method directly on F, it is hard to show the convergence for a general semismooth function, e.g. [31]. The weakest condition seems to be the Lipschitz continuity of B-derivative of F proved in [27]. In our GNEP setting, the function $F(x,\lambda)$ is not differentiable only when $\lambda_j^i = g_j^i(x) = 0$. Otherwise, F is differentiable for which the convergence is established. So, we expect the generalized Broyden method to work well in our setting and do not prove the convergence. To our knowledge, the Broyden method has been not used in the GNEP context.

2.2.4 Global convergence

A merit function is introduced to globalize local methods, as explained in Appendix B. For a semismooth equation $F(z) = 0$, we use the same function as in the differentiable setting : the residual norm $f(z) = ||\ F(z)||^2/2$. The gradient is given by $\nabla f(z) = V^T F(z)$, where $V \in \partial F(z)$. As mentioned in [46], the merit function may be C^1, even if F is only semismooth.

In our GNEP context, the root function Φ is given in (4) and the merit function is thus defined as

$$f_\Phi(z) = \frac{1}{2}\left\| \begin{pmatrix} D_L(x,\lambda) \\ \phi_\circ(-g(x),\lambda) \end{pmatrix} \right\|_2^2. \tag{34}$$

The gradient of f_Φ can be expressed as

$$\nabla f_\Phi(z) = \begin{pmatrix} \mathrm{Jac}_x D_L(x,\lambda) & \mathrm{diag}\left[\left(\nabla_{x_i} g^i(x)\right)_i\right] \\ -D_a(z)\mathrm{Jac}_x g(x) & D_b(z) \end{pmatrix}^T \begin{pmatrix} D_L(x,\lambda) \\ \phi_\circ(-g(x),\lambda) \end{pmatrix}.$$

using (26) and classic chain rules. As mentioned in [12], the gradient ∇f_Φ is single-valued since only the bottom part of the generalized Jacobian $\partial\Phi(z)$ contains multi-valued expressions (D_a and D_b when $g_j^i(x) = \lambda_j^i = 0$) and the bottom part of $\Phi(x,\lambda)$ has zero entries. Hence, f is C^1 as long as objective and constraint functions are C^2. Therefore, both line-search and trust-region approaches are well defined for our GNEP reformulation (4) and (34).

As given in Appendix B.5 (and as for local convergence), a nonsingularity condition of elements in the limiting Jacobian $\partial_B\Phi$ is required. Under this nonsingularity condition, sequences generated by generalized Newton, generalized Levenberg-Marquardt and generalized Broyden converge to a stationary point of f_Φ. This leads to a new question : does a stationary point of f_Φ solve the original GNEP? Theorem 3 of [12] analyzes the additional conditions for a stationary point of f to be a solution of the nonsmooth equation, hence of the original GNEP. The condition is that $\mathrm{Jac}_x D_L(x,\lambda)$ is nonsingular and the matrix

$$M = \mathrm{Jac}_x g(x)\mathrm{Jac}_x D_L(x,\lambda)^{-1}\mathrm{diag}\left[\left(\nabla_{x_i} g^i(x)\right)_i\right] \tag{35}$$

is a P_0-matrix[11]. If diagonal matrices D_a and D_b have non zero terms, then Condition (35) implies that $D_b(z) + MD_a(z)$ is nonsingular, which is Condition (32). Therefore, if we have the nonsingularity assumption of elements of the limiting Jacobian of Φ, then Condition (35) can be deduced and thus ensuring that a stationary point of the merit function f_Φ (defined in (34)) solves the GNEP.

[11] A $m \times m$ square matrix M is a P_0-matrix if for all subscript set $\alpha \subset \{1,\ldots,m\}$ the determinant $\det(M_{\alpha\alpha}) \geq 0$.

2.3 Tailored optimization algorithms for the CER

This subsection aims to present specific methods solving constrained (nonlinear) equations, first proposed by [12] in the GNEP context. The KKT system can be reformulated as a constrained equation, see Equation (5) of Subsection 1.1.2. Techniques to solve such equation may provide good alternatives to standard optimization procedures. In the context of VI problems, [20] devotes a chapter to interior-point methods for solving such constrained equations (CE). Here, we focus on the method of [39] providing a general framework for CE problems.

Definition 13 (Constrained equation)**.** *A constrained equation is defined as*

$$H(z) = 0, \quad z \in \Omega, \tag{36}$$

where Ω is a closed subset of $\mathbb{R}^n$. Generally, the constraint set Ω has a simple structure, e.g. the nonnegative orthant $\Omega = \mathbb{R}^n_+$ or a hypercube $\Omega = [l, u]$.

As mentioned in [61], in practice, the root function H has also a partitionned form

$$H(z) = \begin{pmatrix} F(z) \\ G(z) \end{pmatrix}.$$

Indeed, the KKT reformulation of the GNEP (2) falls within this framework

Let $\mathring{\Omega}$ denote the interior of Ω. In the contrained equation literature, we assume that (i) Ω is a closed subset with nonempty interior $\mathring{\Omega}$, (ii) there exists a closed convex set $S \subset \mathbb{R}^n$, such that $0 \in S$, $H^{-1}(\mathring{S}) \cap \mathring{\Omega}$ is nonempty but $H^{-1}(\mathring{S}) \cap \text{bd } \Omega$ is empty and (iii) H is C^1 function. The first assumption states that the set S contains zero, so that the set $H^{-1}(\mathring{S})$ contains a solution to Equation (36) and local points around. The second assumption requires that such points should not be on the boundary of Ω. The third assumption is just differentiability. In the following, these three assumptions will be referenced as the constrained equation blanket assumptions.

2.3.1 Potential reduction algorithms with line-search

[39] build a framework with potential functions for solving constrained equations. These potential functions, which play a major role, are first introduced.

Definition 14 (potential function)**.** *A potential function $\mathring{S} \mapsto \mathbb{R}$ satisfies the following properties:*

1. *For all sequences $(u_k)_k$ in $\mathring{S}$ such that either $||u_k||$ tends to infinity or (u_k) tends to a point on the boundary bd $S \setminus \{0\}$, then $p(u_k)$ tends to infinity.*
2. *p is C^1 function on its domain and the curvature condition $u^T \nabla p(u) > 0$ for all nonzero vectors.*
3. *There exists a pair $(a, \bar{\sigma})$ in $\mathbb{R}^n \times]0, 1]$, such that for all $u \in \mathring{S}$, we have $||a||^2 u^T \nabla p(u) \geq \bar{\sigma}(a^T u)(a^T \nabla p(u))$.*

The potential function has the dual objective to keep the sequences $(H(x_k))_k$ away from the set bd $S \setminus \{0\}$ and to help the convergence to the zero vector. The parameter a, known as the central vector, will play a crucial role to generate iterates in the constrained set Ω.

For instance, let us consider $S = \mathbb{R}^n_+$. A typical potential function for S is

$$p(u) = \zeta \log ||u||_2^2 - \sum_{i=1}^{n} \log u_i \quad \text{for} \quad u > 0.$$

[39] prove that this function verifies the above conditions when $\zeta > n/2$ with the pair $(a, \bar{\sigma}) = (\mathbb{1}_n, 1)$ where $\mathbb{1}_n$ being the n-dimensional one vector.

In the GNEP context, we consider the subset $S = \mathbb{R}^{n_\cdot} \times \mathbb{R}^{2m_\cdot}_+$ where $n_\cdot = \sum_{i=1}^N n_i$ is the total number of player variables and $m_\cdot = \sum_{i=1}^N m_i$ is the total number of constraints, i.e. $n = n_\cdot + 2m_\cdot$ in Definition 13. The function H has components F and G given by

$$F(z) = D_L(x, \lambda) \quad \text{and} \quad G(z) = \begin{pmatrix} g(x) + w \\ \lambda \circ w \end{pmatrix},$$

where $z = (x, \lambda, w)$, see, e.g., [61]. [12] propose the following potential function

$$p\,(u) = \zeta \log \left(||u_1||_2^2 + ||u_2||_2^2 \right) - \sum_{i=1}^{2m_\cdot} \log(u_{2i}),$$

where $u = (u_1, u_2) \in \mathbb{R}^{n_\cdot} \times \mathbb{R}^{2m_\cdot}_+$. We choose $\zeta > m_\cdot$ so that p is a potential fucntion with $a = (0_{n_\cdot}, \mathbb{1}_{m_\cdot})$ and $\bar{\sigma} = 1$.

The difficulty of constrained equations, compared to a classical nonlinear equation, is to ensure that all the iterates remains in the constrained set Ω. In [39], this is achieved by considering a modified Newton method globalized with a backtracking line-search. We report below their potential reduction Newton algorithm, which is divided into two parts: (i) computation of the direction using the central vector a and (ii) finding of an appropriate stepsize with a geometric line-search for which the merit function is $\psi(u) = p(H(u))$. Note that $H(u)$ is valued in $\mathbb{R}^{n_\cdot} \times \mathbb{R}^{2m_\cdot}$.

Init $z_0 \in \mathring{\Omega}, 0 < \rho, \alpha < 1$ and choose $\sigma_0 \in [0, \bar{\sigma}[$

Iterate until a termination criterion is satisfied,

- Solve the system (37) to get d_k[12]

$$H(z_k) + \operatorname{Jac} H(z_k) d = \sigma_k \frac{a^T H(z_k)}{||a||_2^2} a. \tag{37}$$

- Find the smallest integer m_k such that

$$\psi(z_k + \rho^{m_k} d_k) \leq \psi(z_k) + \alpha \rho^{m_k} \nabla \psi(z_k)^T d_k \quad \text{and} \quad z_k + \rho^{m_k} d_k \in \mathring{\Omega}.$$

- Set $z_{k+1} = z_k + \rho^{m_k} d_k$.

[12] In [39], they use the directional derivative along d in the left-hand side of Equation (37), which is equivalent to this formulation since H is C^1 under the blanket assumptions.

end Iterate

Due to the presence of a non-zero term in the right-hand side of (37), the modified Newton direction d_k is different from the Newton direction. For a given problem, the special structure of $H = (F, G)$ and a will simplify this term. In this form, the algorithm is defined when the Jacobian $\operatorname{Jac} H$ is nonsingular at $z_k \in \mathring{\Omega}$. Lemma 2 of [39] shows that the direction computed in the first step is a descent direction for the merit function ψ. So, the line-search is well-defined. Theorem 3 of [39] shows the convergence of the potential reduction algorithm.

Theorem 12. *Assume that p is a potential function, the constrained Equation (36) satisfies the constrained equation blanket assumptions, the Jacobian $\operatorname{Jac} H(z)$ is nonsingular for all $z \in \mathring{\Omega}$ and we have $\limsup_k \sigma_k < \bar{\sigma}$. Let $(z_k)_k$ be a sequence generated by the potential reduction Newton algorithm. We have (i) the sequence $(H(z_k))_k$ is bounded and (ii) any accumulation point, if there exists, solves the constrained Equation (36). In particular, if $(z_k)_k$ is bounded, the constrained equation has a solution.*

In the numerical illustrations, the potential reduction algorithm is benchmarked with the affine-scaling algorithm of [3], as in [12].

2.3.2 Application to GNEP

As already mentioned, Equation (5) of the GNEP can be reformulated as a constrained equation. The root function $H : \mathbb{R}^n \times \mathbb{R}^{2m} \mapsto \mathbb{R}^n \times \mathbb{R}^{2m}$ is defined as

$$H(x, \lambda, w) = \begin{pmatrix} D_L(x, \lambda) \\ g(x) + w \\ \lambda \circ w \end{pmatrix},$$

where the dimensions n, m correspond to the GNEP notation and $(a, \bar{\sigma})$ is given by $((0_n, \mathbb{1}_m), 1)$. The potential function is given by

$$p\,(u) = \zeta \log\left(||x||_2^2 + ||\lambda||_2^2 + ||w||_2^2\right) - \sum_{k=1}^{m} \log(\lambda_k) - \sum_{k=1}^{m} \log(w_k),$$

where $u = (x,\ \lambda, w) \in \mathbb{R}^n \times \mathbb{R}^m_{\star+} \times \mathbb{R}^m_{\star+}$ and $\zeta > m$. This reformulation of the potential function emphasizes the three components $u = (x, \lambda, w)$. For the line-search, the gradient ∇p is given by

$$\nabla p(x, \lambda, w) = \begin{pmatrix} \frac{2\zeta}{||x||_2^2 + ||\lambda||_2^2 + ||w||_2^2} x \\ \frac{2\zeta}{||x||_2^2 + ||\lambda||_2^2 + ||w||_2^2} \lambda - \lambda^{-1} \\ \frac{2\zeta}{||x||_2^2 + ||\lambda||_2^2 + ||w||_2^2} w - w^{-1} \end{pmatrix},$$

where λ and w have positive components and terms λ^{-1} and w^{-1} correspond to the component-wise inverse vector. Compared to the semismooth reformulation, the root function H is now C^1. The Jacobian is given by

$$\operatorname{Jac} H(x, \lambda, w) = \begin{pmatrix} \operatorname{Jac}_x D(x, \lambda) & \operatorname{diag}\left[\left(\nabla_{x_i} g^i(x)\right)_i\right] & 0 \\ \operatorname{Jac}_x g(x) & 0 & I \\ 0 & \operatorname{diag}[w] & \operatorname{diag}[\lambda] \end{pmatrix}.$$

As reported in [12], the computation of the direction $d_k = (d_{x,k}, d_{\lambda,k}, d_{w,k})$ in (37) can be simplified due to the special structure of the above Jacobian matrix. The system reduces to a linear system of n equations to find $d_{x,k}$ and the $2m$ components $d_{\lambda,k}, d_{w,k}$ are simple linear algebra. Using the classic chain rule, the gradient of the merit function $\psi(.) = p(H(.))$ is given by

$$\nabla\psi(x,\lambda,w) = \mathrm{Jac}\, H(x,\lambda,w)^T \nabla p(H(x,\lambda,w)).$$

Again the computation of this gradient can be simplified due to the sparse structure of $\mathrm{Jac}\, H$. Theorem 4.3 of [12] is the direct application of the previous theorem in the GNEP context. We do not restate here their theorem, but present their nonsingularity result given in Theorem 4.6. The Jacobian matrix is nonsingular, if the matrix $\mathrm{Jac}_x D_L(x,\lambda)$ is nonsingular and

$$M = \mathrm{Jac}_x g(x) \mathrm{Jac}_x D_L(x,\lambda)^{-1} \mathrm{diag}\left[\left(\nabla_{x_i} g^i(x)\right)_i\right] \tag{38}$$

is a P_0-matrix. This is exactly Condition (35) given in the semismooth setting. So there is no particular gain in terms of assumptions between the two reformulations.

2.4 Numerical illustrations

In this section, we perform numerical illustrations to compared the different methods for computing general GNEP. The implementation is carried out in the R statistical software ([48]) and the package **GNE** ([13]). We start our benchmarking by considering a simple two-player duopoly for which there exists a unique Nash equilibrium. Let $d_1 = 20$, $d_2 = 21$, $\lambda_1 = 4$, $\lambda_2 = 5$, $\rho_1 = 1.1$ and $\rho_2 = 1.25$. The objective function (to be maximized) of Player i is $\theta_i(x) = (d_i - \lambda_i - \rho_i(x_1 + x_2))x_i$ for $i = 1, 2$. One can show that the Nash equilibrium is

$$(x_1^\star, x_2^\star) = \left(\frac{2(d_1-\lambda_1)}{3\rho_1} - \frac{d_2-\lambda_2}{3\rho_2}, \frac{2(d_2-\lambda_2)}{3\rho_2} - \frac{d_1-\lambda_1}{3\rho_1}\right).$$

In Table 1, we benchmark all the root methods of this section : 'Fct. call' is the number of function calls (either Φ or H), 'Jac. Call' is the number of Jacobian calls (either $\partial\Phi$ or $\mathrm{Jac}\, H$), Code is an exit code and $||x_{final} - x^\star||_2$ is the distance between the final iterate and the true NE $x^\star$. Algorithms always stop after a finite number of iterations with an exit code specifying whether the sequence converges or not: (1) convergence is achieved $||F(z)||_\infty < ftol$ with $ftol = 10^{-8}$, (2) algorithm is stuck because two consecutive iterates are too close $||(z_k - z_{k-1})/z_k||_\infty < xtol$ with $xtol = 10^{-8}$, (3) stepsize is too small $t_k < xtol$ or radius is too small $\Delta_k < xtol$, (4) the iteration limit is exceeded $k > k_{max}$ with $k_{max} = 300$; or generalized Jacobian is either ill-conditionned (5) or singular (6); see e.g. Chapter 7 of [11].

In Table 1, the first eight rows corresponds to generalized Newton and generalized Broyden methods with a globalization scheme (line search or trust-region). From the ninth to the tenth row, we consider the Levenberg-Marquardt (LM) method with a fixed parameter $\lambda_k = \min(||F(z_k)||^2, ||J_k^T F(z_k)||^2)$ (see Subsection 2.1.1) and an adaptive parameter λ_k (see Appendix B.3). For the complementarity reformulation of Subsection 2.2, we do not have to choose *in this game* a complementarity function as there are no constraints. The last two rows correspond to the constrained equation reformulation : 'PR' stands for potential reduction algorithm and 'AS' is the affine scaling. In this game, we observe that all methods converges in few steps, but the Levenberg-Marquardt iterates are longer to converge.

method	Fct. call	Jac. Call	Code	$\|\|x_{final} - x^\star\|\|_2$
Newton - geom. line search	1	1	1	4.440892e-16
Newton - quad. line search	1	1	1	4.440892e-16
Newton - Powell trust region	1	1	1	4.440892e-16
Newton - dbl. trust region	1	1	1	4.440892e-16
Broyden - geom. line search	1	1	1	4.440892e-16
Broyden - quad. line search	1	1	1	4.440892e-16
Broyden - Powell trust region	1	1	1	4.440892e-16
Broyden - dbl. trust region	1	1	1	4.440892e-16
LM - geom. line search	8	8	1	1.176846e-07
LM - quad. line search	8	8	1	1.176846e-07
LM - adaptive	6	3	1	8.188869e-11
PR - geom. line search	1	1	1	2.979041e-15
AS - Powell trust region	3	3	1	9.930137e-16

Table 1: Benchmark of root methods on the generalized duopoly

Secondly, we consider a more elaborate two-player game for which there are four generalized Nash equilibria. The objective functions (to be minimized) are given by $\theta_1(x) = (x_1 - 2)^2(x_2 - 4)^4$ and $\theta_2(x) = (x_2-3)^2(x_1)^4$, for $x \in \mathbb{R}^2$, while the constraint functions are $g^1(x) = x_1+x_2-1 \leq 0$ and $g^2(x) = 2x_1 + x_2 - 2 \leq 0$. Objective functions are player-strictly convave. This problem is simple but not simplistic, since second-order partial derivatives of objective functions are not constant, as the previous game. Due to the simple form of the objective function, we can manually solve the KKT system for this GNEP. The solutions are listed in Table 2.

	$x_1^\star$	$x_2^\star$	$\lambda_1^\star$	$\lambda_2^\star$
$z^{\star 1}$	2	-2	0	5×2^5
$z^{\star 2}$	-2	3	8	0
$z^{\star 3}$	0	1	4×3^4	0
$z^{\star 4}$	1	0	2^9	6

Table 2: The four GNE

In Table 3, we report the result with the complementarity function ϕ_{FB} and the starting point $z_0 = (5, 5, 0, 0)$, while for the constrained equation method, the starting point is $z_0 = (5, 5, 2, 2, 2, 2)$. Most methods converge to an equilibrium, whose number in Table 2 is indicated the last column $\star i$ of Table 3. Surprisingly, the Newton method with geometric line seach converges to $z^{\star 2}$ whereas the trust-region Newton methods converge to $z^{\star 3}$, despite using the *same* initial points. A similar behavior is found for the Broyden method. Some methods diverge to a local minimum of the merit function $1/2||F(z^\star)||_2^2$ which is not a root of F, for example the potential reduction algorithm and the affine scaling method. In overall, there is a clear advantage for classic semismooth methods solving the extended KKT system on this game. With the minimum complementarity function $\phi_\wedge$, we get similar results.

To further compare these methods, we draw uniformly 1000 random initial points such that $z_0 \in [-10, 10]^2 \times [0, 1]^2$ and run algorithms on each of them. For simplicity, we restrict out comparison to Newton GLS and Broyden PTR methods. Results are summarized in Table 4, the first four columns store the number of sequences converging to a particular GNE, while the last column ∞ contains the number of diverging sequences (under the same termination criterion). On this game based on the number of diverging sequences, the best method seems to be the Broyden method with a Powell trust-region globalization.

	Fct. call	Jac. call	Code	$\|\|F(z^\star)\|\|$	$\star i$
Newton - geom. line search	28	11	1	6.476053e-09	2
Newton - quad. line search	92	31	3	48.30996	
Newton - Powell trust region	63	50	1	5.606129e-07	3
Newton - Dbl. trust region	63	54	1	5.868254e-07	3
Broyden - geom. line search	141	9	1	1.790037e-08	2
Broyden - quad. line search	503	27	3	0.00178784	
Broyden - Powell trust region	261	6	1	8.044891e-07	3
Broyden - Dbl. trust region	68	4	1	7.342636e-08	2
LM min - geom. line search	29	29	1	2.936729e-07	2
LM min - quad. line search	29	29	1	2.936729e-07	2
LM - adaptive	44	22	1	7.249862e-13	2
PR - geom. line search	157	20	3	1830.257	
AS - Powell trust region	6	6	2	14.06935	

Table 3: Benchmark of root methods on the generalized game

	$z^{\star 1}$	$z^{\star 2}$	$z^{\star 3}$	$z^{\star 4}$	∞
Newton - geom. line search (GLS)	0	349	236	0	415
Broyden - Powell trust region (PTR)	120	453	45	87	295

Table 4: Number of GNE found for 1000 random initial points

3 Methods to solve jointly GNEP

As explained in Subsections 1.3 and 1.2, the GNEP can be reformulated as a minimum problem and a fixed-point, respectively, which further simplifies for a jointly convex GNEP. In this section, we start by describing usual methods for solving a minimum problem and a fixed-point problem in Section 3.1 and 3.3. Then, we focus on the application of theses methods to the GNEP, respectively in Sections 3.2 and 3.4

3.1 Classical methods for solving minimization problem

In this subsection, we present natural optimization methods solving smooth minimization problems $\min_x f(x)$. As in Subsection 2.1, we first present local methods and then continue with convergence theorems.

3.1.1 Local methods

Local methods are sequences $z_{k+1} = z_k + d_k$ where d_k is a root of a certain equation based on a local approximation of $f : \mathbb{R}^n \mapsto \mathbb{R}$. Assuming f is C^2, the Newton method uses a second-order Taylor expansion

$$f(z) + \nabla f(z)^T h + \frac{1}{2} h^T \nabla^2 f(z) h + o\big(h^2\big)\,,$$

where $\nabla^2 f$ denotes the Hessian matrix. By finding h minimizing this quadratic (local) approximation at z_k, we get

$$\nabla^2 f(z_k) d_k = -\nabla f(z_k). \tag{39}$$

As for the Jacobian matrix in non-linear equation, there is no guarantee that the Hessian matrix will be always invertible. Quasi-Newton methods, i.e. approximating the Hessian $\nabla^2 f$ by invertible matrices H_k, is also possible for minimum problem. Therefore, for quasi-Newton methods, the direction solves the equation

$$H_k d_k = -\nabla f(z_k). \tag{40}$$

Many schemes can be used to update H_k : we require schemes to verify the secant equation $H_{k+1}s_k = y_k$ with $s_k = z_{k+1} - z_k$ and $y_k = \nabla f(z_{k+1}) - \nabla f(z_k)$. The most classical scheme for updating H_k (BFGS, see e.g. [41, 5]) is a rank-two symmetric scheme that preserves positive definiteness and is defined as

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{y_k^T s_k}.$$

By the Sherman-Morrison formula, this update can be represented in terms of the approximate inverse Hessian, which is how it is implemented in practice. In our GNEP context, the function f is not necessarily twice differentiable. Thus, we present a semismooth generalization of Equation (39) by using the generalized Hessian[13] $\partial^2 f$. The generalized Newton direction solves

$$H_k d_k = -\nabla f(z_k), \tag{41}$$

where $H_k \in \partial^2 f(z_k)$. In the numerical illustrations when solving the minimization problems of Subsection 1.4, we consider only the BFGS scheme (40) as the generalized Newton scheme (41) is heavier to put in place. As for nonlinear equations, a globlalization scheme must be used in addition to a given local methods, cf. Appendix B. In our numerical illustration, we use line-search techniques with the merit function being (directly) f.

We also test two Hessian-free methods, which are particularly attractive in the GNEP context : the conjugate gradient and the Barzilai and Borwein methods, see e.g. [41] and [49], respectively. The latter was already tested in the GNEP context by [59]. Let $g_k = \nabla f(z_k)$. The (nonlinear) conjugate gradient method builds a sequence of directions starting by $d_1 = -g_1$ and

$$d_k = -g_k + \beta_k d_{k-1}$$

for $k > 1$. Generally, β_k are updated according one of the following schemes : the Fletcher-Reeves update $\beta_k = g_k^T g_k / g_{k-1}^T g_{k-1}$ (the default) or the Polak-Ribière $\beta_k = g_k^T(g_k - g_{k-1})/g_{k-1}^T g_{k-1}$. Therefore, there is no equation to solve when computing d_k in the conjugate gradient method. In practice, the conjugate gradient is combined with a backtracking line-search.

The Barzilai and Borwein method was first introduced to deal with large scale optimization methods. This method is particularly relevant in our context, since it uses only the function and the gradient to determine the next iterate. The Barzilai and Borwein method has a direction and a stepsize given by

$$d_k = -g_k,\ t_k = \frac{d_{k-1}^T d_{k-1}}{d_{k-1}^T (g_k - g_{k-1})}.$$

In the literature, the following step size is sometimes considered $t_k = d_{k-1}^T(g_k - g_{k-1})/(g_k - g_{k-1})^T(g_k - g_{k-1})$, but we discard it for numerical accuracy reasons. In practice, the method is also globalized using a nonmonotone line-search, see e.g. [26].

[13] The generalized Hessian is defined as the generalized Jacobian of the function $z \mapsto \nabla f(z)$ for a function f with a locally Lipschitzian gradient $\partial^2 f(x) = \text{co}\{H \in \mathbb{R}^{n\times n}, \exists x_k \to x, \text{ with } \nabla f(x_k) \text{ is differentiable and } \nabla^2 f(x_k) \to H\}$.

3.1.2 Convergence theorems

The convergence theorem for the (classical) Newton method, see e.g. Theorem 3.5 of [41] and Theorem 4.1 of [5], are now recalled.

Theorem 13. *Suppose that f is C^2 in a open convex set $O \subset \mathbb{R}^n$. Let $z^\star$ be a stationary point of f (i.e. $\nabla f(x^\star) = 0$). Let $(z_k^N)_k$ be the sequence generated by the Newton method. If the Hessian $\nabla^2 f$ is Lipschitz continuous around $z^\star$ and z_0 is sufficiently close to $z^\star$, then $(z_k^N)_k$ converges quadratically to $z^\star$.*

Now, we give a generalization by no longer assuming f is C^2. We consider functions with semismooth gradients called SC^1. This theorem is an application of Theorem 3.2 of [47] to the nonsmooth equation $\nabla f(z) = 0$.

Theorem 14. *Let $z^\star$ be a stationary point of a SC^1 function f satisfying $\forall v \neq 0, \forall H \in \partial^2 f(z^\star), v^T H v > 0$. The (generalized) Newton method $z_{k+1} = z_k + d_k$ where d_k solves (41), converges superlinearly to $z^\star$. If in addition ∇f is strongly semismooth at $z^\star$, then the convergence rate is quadratic.*

Note that using the same reasoning as in Subsection 2.2.2, we can replace the nonsingularity of $H \in \partial^2 f(z^\star)$ to the nonsingularity of $H \in \partial_B^2 f(z^\star)$.

When combined with a line-search, the global convergence of the generalized Newton method can be proved. Recalling that for smooth problems, convergence theorems require (at least) the function f to be C^2 with a Lipschitz Hessian in a neighborhood of a solution, see e.g. Theorem 3.5 of [41]. This is similar for the convergence of the generalized Newton method, see e.g. Theorem 4.1 of [29] and Theorems 8.3.19 and 10.4.9 of [20]. We present below a version based on [14] and refer to Appendix B.1 for more details on line-search techniques.

Theorem 15. *Let $x^\star$ be a stationary point of a SC^1 function f satisfying $\forall v \neq 0, \forall H \in \partial^2 f(z^\star), v^T H v > 0$. The (generalized) Newton method with line search $x_{k+1} = x_k + t_k d_k$ where d_k solves (41) and t_k satisfies the Armijo condition, converges superlinearly to $x^\star$. If in addition ∇f is strongly semismooth at $x^\star$, then the convergence is quadratic.*

Proof. By Theorem 3.6 of [14], we have $\exists \rho > 0, \nabla f(x_k)^T d_k \leq -\rho ||d_k||^2$. And by Theorem 3.3 of [14], there exists a $\bar{k}$ such that the Armijo condition is always satisified for a full Newton step $t_k = 1$. Theorem 14 completes the proof. □

Now we turn our attention to the convergence of quasi-Newton methods. In the differentiable setting, e.g. [41] or [5], the convergence of the BFGS method when assuming f is C^2 and a condition on the deviation $||(H_k - \nabla^2 f(z_k))s_k||/||s_k|| \to 0$. Extensions of those convergence theorems when f is only SC^1 have yet to be found. Theorems 9 and 10 applied to the semismooth equation $\nabla f(z) = 0$ provide criteria for a general quasi-Newton method. The resulting condition on the deviation is $||(H_k - V_k)s_k||/||s_k|| \to 0$, where $V_k \in \partial^2 f(z_k)$. It is still an open problem if the BFGS scheme satisfies this condition. However, as the BFGS method does not need the Hessian matrix, we test it without knowing the convergence.

Finally, we give two convergence theorems for the conjugate gradient (with Fletcher-Reeves) and the Barzilai-Borwein method, respectively Theorem 5.7 of [41] and Theorem 2.1 of [49].

Theorem 16. *Let $(z_k^{FR})_k$ generated by the Fletcher-Reeves update. Assume the level set $L = \{z, f(z) \leq f(z_0)\}$ is bounded and f is C^1 with Lipschitz gradient on L. If the line search satisfies the strong Wolfe conditions with $c_1 < c_2 < 1/2$, then the sequence $(z_k^{FR})_k$ either terminates to a stationary point or converges in the sense that $\liminf_{k\to+\infty} ||\nabla f(z_k)|| = 0$.*

Theorem 17. *Consider $(z_k)_k$ the sequence of the globalized Barzilai-Borwein method. Assume the level set $L = \{z, f(z) \leq f(z_0)\}$ is bounded and f is C^1 in some neighborhood of L. Then, the sequence $(z_k)_k$ either attains to a stationary point in a finite number of step or converges to a stationary point if there are in a finite number of steps.*

3.2 Application to the VI and NI reformulations

All necessary tools for the study of the GNEP (13) have been presented : we focus on the two reformulations of Subsection 1.4.

3.2.1 The VI reformulation

Let us start by the VI reformulation of Subsection 1.4.2. We recall that the merit function of the GNEP is $V_{VI}(x) = \max_y \psi_{rVI}(x, y)$ where ψ_{rVI} is defined as

$$\psi_{rVI}(x, y) = F(x)^T(x - y) - \frac{1}{2}(x - y)^T H(x - y),$$

and $F(x) = (\nabla_{x_i}\theta_i(x))_i$. In order to derive the gradient of V_{VI} in terms of ψ_{rVI}, we apply the Dankins theorem, see e.g. [4]. Since ψ_{rVI} is concave in y, we get

$$\nabla V_{VI}(x) = \mathrm{Jac}\, F(x)^T(x - y_H(x)) + F(x) - H(x - y_H(x)),$$

where y_H is the unique maximizer of $y \mapsto \psi_{rVI}(x, y)$. This formulation requires F to be differentiable, i.e. objective functions θ_ν's to be C^2.

3.2.2 The NI reformulation

We continue with the NI reformulation of Subsection 1.4.3. The regularized Nikaido-Isoda function is defined

$$\psi_{rNI}(x, y) = \sum_{\nu=1}^{N} \left[\theta_\nu(x_\nu, x_{-\nu}) - \theta_\nu(y_\nu, x_{-\nu}) - \frac{\alpha}{2}||x_\nu - y_\nu||^2\right],$$

with which the merit function is $V_{rNI}(x) = \max_y \psi_{rNI}(x, y)$. Again by applying the Dankins theorem, we get the gradient V_{rNI}

$$\nabla V_{rNI}(x) = \sum_{\nu=1}^{N} [\nabla_x\theta_\nu(x_\nu, x_{-\nu}) - \nabla_x\theta_\nu(y_{\alpha,\nu}(x), x_{-\nu})] + \begin{pmatrix} \nabla_{x_1}\theta_1(y_{\alpha,1}(x), x_{-1}) \\ \vdots \\ \nabla_{x_N}\theta_N(y_{\alpha,N}(x), x_{-N}) \end{pmatrix} - \alpha(x - y_\alpha(x)),$$

where y_α is the unique maximizer of $y \mapsto \psi_{rNI}(x, y)$. In this case, θ_ν needs only to be C^1.

3.2.3 Analysis of assumptions

We turn our attention to the analysis of assumptions of Theorems 14 and 15 for Hessian-based methods, and then Theorems 17 and 16 for Hessian-free methods. For Hessian-based methods, convergence theorems require the function to be SC^1. For the NI reformulation, Theorem 3.6 of [58] (recalled below) shows that the regularized Nikaido-Isoda function is SC^1 in a neighborhood of a GNEP $x^\star$ under certain conditions.

Theorem 18. *Let $x^\star$ be a solution of the GNEP (13). Assume that functions θ_ν are C^2 and player-convex, and function g is C^2 and convex. If for all active constraints (i.e. $i, g_i(y^\star) = 0$) of $\max_y \psi_{rNI}(x, y)$ $\nabla g_i(y^\star)$ are linearly independent with $y^\star = y_\alpha(x^\star)$, then V_{rNI} is an SC^1-function in the neighborhood of $x^\star$.*

The semismoothness of the gradient ∇V_{rNI} is proved by showing that ∇V_{rNI} is a composition of C^1 functions $\nabla_x \theta_\nu$ and a semismooth function y_α. Thus, the difficulty lies on the latter part. Firstly, they show that a semismooth reformulation of the KKT conditions of $\max_y \psi_{rNI}(x, y)$ is locally Lipschitzian and has an explicit form for its generalized Jacobian. Secondly, the condition of linear independence of active gradients and the concavity of $y \mapsto \psi_{rNI}(x, y)$ guarantee the nonsingularity of elements of the generalized Jacobian. Finally, the semismoothness is derived by the implicit function theorem.

For the VI reformulation, we can derive the same result since the regularized gap function $y \mapsto \psi_{rVI}(x, y)$ is also concave but we have to assume F is a C^2 function. We get the following theorem.

Theorem 19. *Let $x^\star$ be a solution of the GNEP (13). Assume that objective functions θ_ν are C^3, and constraint function g is C^2 and convex. If for all active constraints of $\max_y \psi_{rVI}(x, y)$, i.e. $i, g_i(y^\star) = 0$, $\nabla g_i(y^\star)$ are linearly independent with $y^\star = y_H(x^\star)$, then V_{rVI} is an SC^1-function in the neighborhood of $x^\star$.*

Proof. Same proof as Theorem 3.6 of [58] for the problem $\max_y \psi_{rVI}(x, y)$. □

A second assumption for Hessian-based methods in convergence theorems is the nonsingularity of elements in the generalized Hessian : $V \in \partial^2 V_{rNI}(x^\star)$ and $V \in \partial^2 V_{rVI}(x^\star)$. So, a natural question raises about the expression of $\partial^2 V_{rNI}$. Lemma 4.1 and 4.2 of [58] gives an expression $\partial^2 V_{rNI}$, which can be easily adapted to $\partial^2 V_{rVI}$. The expression of the generalized Hessian can be derived using the chain rule and is given by

$$\nabla_{xx}\psi(x, y) + \nabla_{xy}\psi(x, y)\partial y(x) \subset \partial^2 V(x),$$

where V equals to V_{rVI} or V_{rNI}, ψ equals to ψ_{rVI} or ψ_{rNI}, and y equals to y_H or y_α. With the same assumption of nonsingularity (of $H \in \partial^2 V(x^\star)$), we can apply Theorems 14 and 15 to get the global convergence. This is done in Theorem 4.3 of [58] where they apply Theorem 14 and requires the nonsingularity of $V \in \partial^2 V_{rNI}(x^\star)$. Similarly, we can apply those theorems to the function V_{rVI}.

Regarding the Hessian-free methods presented in the previous subsection, the conditions for convergence are lesser than for Hessian-based methods. In Theorems 16 and 17 (for $\min_z f(z)$), they require the differentiability of f and the boundedness of the set $L = \{z, f(z) \leq f(z_0)\}$. We already know that V_{rVI} and V_{rNI} are C^1 functions, while the boundedness of L is generally not a problem.

3.3 Classical methods for solving fixed-point problem

We deal with methods solving the fixed-point problem of Propositions 3 and 4. As seen in Subsections 1.4.2 and 1.4.3, the GNEP can be reformulated to a fixed-point problem. Therefore in this subsection, we present main algorithms to solve the fixed-point equation

$$z = F(z), \tag{42}$$

where $F : \mathbb{R}^n \mapsto \mathbb{R}^n$. By Banach's fixed point theorem, we know that the fixed point is unique if F is contractive, i.e. $\exists 0 < L < 1, \forall z, y, ||F(z) - F(y)|| \leq L||z - y||$. The two main approaches for computing fixed points are polynomial methods and epsilon algorithms, see e.g. [55]. We focus on the former one.

A first method to solve (42) uses the pure fixed-point iterates $z_{n+1} = F(z_n)$, starting from z_0, which is also known as the Picard iteration. In the case of contractive function, the Picard iteration converges to the fixed point from any starting point at a geometric rate. Applying the Newton method to the equation $z - F(z) = 0$ leads to the following sequence of iterate $z_{n+1} = z_n - (I - \mathrm{Jac}\, F(z_n))(z_n - F(z_n))$. In practice, the Newton method is slow to converge and needs the computation of the Jacobian. In the following, we deal with extensions of the Picard method.

Polynomials extrapolation methods aim to accelerate the Picard method by considering polynoms of the function F., Let F^i stands for the ith composition of F (and F^0 the identity function). Polynomial methods are based on the general scheme

$$z_{k+1} = \sum_{i=0}^{d} \gamma_i^k F^i(z_k),$$

where γ_i^k are constants and d is the order of the polynom such that $\sum_{i=0}^{d} \gamma_i^k = 1$. Let $\Delta^{i,j}$ be the forward differences $\Delta^{i,j} z_k = \sum_{l=0}^{j} (-1)^{l-j} C_l^j F^{l+i}(z_k)$ with C_l^j the binomial coefficients for positive integers i, j. For first-order methods, i.e. $d = 1$, the next iterate computation reduces to

$$z_{k+1} = z_k - \gamma_1^k r_k,$$

where $r_k = F(z_k) - z_k$ is known as the residual and γ_1^k is the. We only have to determine γ_1^k since $\gamma_0^k = 1 - \gamma_1^k$. Relaxation methods consider a decreasing stepsize independent of the current iterate z_k, but more advanced methods can be used to determine γ_i^k.

d-order extrapolation methods propose a set of equations for coefficients γ_i^k to satisfy

$$\sum_{i=0}^{d} \beta_{i,j}^k \gamma_i^k = 0, \tag{43}$$

where $\beta_{i,j}^k$ are expressed in terms of forward differences $\Delta^{i,j}z_k$. The reduced rank extrapolation (RRE) method uses $\beta_{i,j}^k = (\Delta^{i,1}z_k)^T\Delta^{j,2}z_k$, whereas the minimal polynomial extrapolation (MPE) method assumes $\beta_{i,j}^k = (\Delta^{i,1}z_k)^T\Delta^{j,1}z_k$. In the special case of $d = 1$, the coefficients γ_i^k have a simple explicit form. From (43), we get

$$\gamma_{1,RRE}^k = \frac{v_k^T r_k}{v_k^T v_k} \text{ and } \gamma_{1,MPE}^k = \frac{r_k^T r_k}{r_k^T v_k}, \tag{44}$$

where $r_k = \Delta^{0,1}z_k = F(z_k) - z_k$ and $v_k = \Delta^{0,2}z_k = F(F(z_k)) - 2F(z_k) + z_k$. The corresponding methods, resp. RRE1 and MPE1, are thus defined as $z_{k+1} = z_k - \gamma_{1,.}^k r_k$. Note that RRE1 method is also called the Richardson method, while the MPE1 is called the Lemaréchal method, see e.g. [50].

For a general d, there is a close-form formula for the system (43) which involves matrix transpose and inverse. For RREd, we have

$$z_{k+1} = z_k - \Delta Z_k(\Delta Z_k^T\Delta^2 Z_k)^{-1}\Delta Z_k^T r_k, \tag{45}$$

and for MPEd,

$$z_{k+1} = z_k - \Delta Z_k(\Delta^2 Z_k^T\Delta^2 Z_k)^{-1}\Delta^2 Z_k^T r_k, \tag{46}$$

where ΔZ_k and $\Delta^2 Z_k$ are $d \times d$ matrices given by

$$\Delta^j Z_k = (\Delta^{1,j}z_k, \ldots, \Delta^{1,j}z_{k+d-1}),$$

for $j = 1, 2$. When d is small, z_{k+1} can be reasonably computed by (45) and (46). But for large values of d, extrapolation methods RREd and MPEd have to be implemented by a recursive algorithm, e.g. [51].

Let $u_i^k = F^i(z_k)$. In the previous polynomial methods, the next iterate is computed as $z_k \to u_0^k, \ldots, u_d^k \to z_{k+1}$. The sequence $(u_i^k)_i$'s, called a cycle or a restart, is used to extrapolate the next point with interpolation point $(u_i^k)_i$. Squaring extrapolation methods consist in applying twice a cycle step to get the next iterate, see e.g. [54] and [50]. Therefore, the next point have the general form

$$z_{k+1} = \sum_{i=0}^{d}\sum_{j=0}^{d}\gamma_i^k\gamma_j^k F^{i+j}(z_k).$$

When $d = 1$, the squaring 1st order method are such that z_{k+1} can be rewritten as $z_{k+1} = z_k - 2\gamma_1^k r_k + (\gamma_1^k)^2 v_k$, where the coefficient γ_1^k is given in (44) depending wether we choose the RRE1 or the MPE1 sequence. The corresponding squared version are denoted SqRRE1 and SqMPE1, respectively. The squared RRE and MPE methods are obtained with the same arbitrary vectors.

Finally, we focus on convergence results for RRE and MPE methods. In some papers, there is a blanket assumption that F is a contraction. We present Theorem 6 of [51] based on their Remark 1. To our knowledge, there is no convergence result for squared methods SqRRE1 and SqMPE1.

Theorem 20. *For a fixed point problem $F(z) = z$, we assume F is differentiable, Jac$F$ is Lipschitz. Let $z^\star$ be a fixed point of F. If Jac$F(z^\star) - I$ is nonsingular and if the stepsize given by (45) or (46) is bounded after a finite number of iterations, then there exists a neighborhood U of $z^\star$ such that for all initial points $z_0 \in U$, the sequence $(z_k)_k$ generated by MPE and RRE converges quadratically to $z^\star$.*

3.4 Application to the VIR and NIR

As given in Subsections 1.4.2 and 1.4.3, generalized Nash equilibria of jointly convex games solve a fixed-point problem

$$y(x) = x,$$

where y denotes either (20) the NI reformulation or (17) for the VI reformulation. Those two fixed-point functions are defined as a solution of a maximization problem

$$y(x) = \arg\max_{y \in X} \psi(x, y),$$

where ψ is either ψ_{rNI} or ψ_{rVI}. Since the function $y \mapsto \psi(x, y)$ is strictly concave in our GNEP setting, maximizers of ψ satisfying the first-order KKT condition $\nabla_y \psi(x, y) = 0$, are global maximizers. Let $F : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}^n$ be the mapping $F(x, y) = \nabla_y \psi(x, y)$: F_{VI} and F_{NI} are defined similarly. By definition, $F(x, y(x)) = 0$. Using the differentiability of objective and constraint functions, we get that F is C^1 with respect to both variables. Furthermore, $\mathrm{Jac}_y F(x, y)$ is a positive definite matrix (i.e. hence invertible), since

$$\mathrm{Jac}_y F_{VI}(x, y) = -H \text{ and } \mathrm{Jac}_y F_{NI}(x, y) = -\mathrm{diag}[\nabla^2_{x_i} \theta_i(y_i, x_{-i})] - \alpha I$$

are positive definite. Therefore, by the implicit function theorem, see e.g. Chapter 8 of [63], we conclude that y is a differentiable function with Jacobian given by $\mathrm{Jac}_x y(x) = -(\mathrm{Jac}_y F(x, y(x)))^{-1} \mathrm{Jac}_x F(x, y(x))$. Hence, we get

$$\mathrm{Jac}_x y(x) = -(\mathrm{Jac}_y (\nabla_y \psi)(x, y(x)))^{-1} \mathrm{Jac}_x (\nabla_y \psi)(x, y(x)).$$

The Lipschitzness of $\mathrm{Jac}\, y$ is guaranted by further differentiability of objective and constraint functions.

3.5 Numerical illustrations

We consider the three-player river basin pollution game of [34], where players engaged in an economic activity must meet environmental constraints. The profit of Player j is assumed to be quadratic in player variable as $\theta_j(x) = (d_1 - d_2(x_1 + x_2 + x_3) - c_{1j} - c_{2j} x_j) x_j$ while the common constraint function is defined as $g(x) = (g_l(x))_{1 \leq l \leq 2}$ with $g_l(x) = u_{l1} e_1 x_1 + u_{l2} e_2 x_2 + u_{l3} e_3 x_3 - K_l \leq 0$ for $l = 1, 2$. Constants of this game are listed in Table 5. The normalized Nash equilibrium has no close-form formula but [59] found a value of $x^\star = (21.1467103, 16.0278207, 2.724244725)$.

Player j	c_{1j}	c_{2j}	e_j	u_{1j}	u_{2j}	Index l	K_l	d_l
1	0.10	0.01	0.50	6.5	4.583	1	100	3
2	0.12	0.05	0.25	5.0	6.250	2	100	0.01
3	0.15	0.01	0.75	5.5	3.750			

Table 5: Constants of the river basin pollution game

As in Subsection 2.4, an exit code of a given algorithm is provided: (1) indicates successfull convergence $||m(x^\star)|| \leq ftol$ with $ftol = 1e^{-6}$, (2) algorithm is stuck and (4) the iteration limit is exceeded $k > kmax$ with $kmax = 100$. For both reformulations, we carry out a double-level optimization since the minimization of the function V or the fixed-point function y require the computation $\max_y \ \psi(x, y)$, i.e. for one outer iteration, there are multiple inner iterations. This

constrained system is solved by an augmented Lagrangian method ([37]) implemented in the **alabama** package ([56]). We set the maximum number of iteration to $kmax$ with the same tolerance $ftol$.

3.5.1 Minimization reformulation

Firstly, results for the minimization reformulation are presented. As explained in Section 1.4, the GNEP is reformulated as an unconstrained minimization problem (19) for NIR and (16) for VIR. As in [59], we set $\alpha = 0.02$ and $\beta = 0.05$. In Table 6, we report the number of calls to the function $V_{\alpha\beta}(.)$, the gradient function $\nabla V_{\alpha\beta}(.)$, the regularized gap function $\psi(.,.)$ and its gradient $\nabla\psi(.,.)$, the termination criterion at final iterate and the code, where the algorithms are denoted as follows: 'BB' for the Barzilai-Borwein method, BFGS for the quasi-Newton BFGS method and 'CG' for the conjugate-gradient method. For both reformulations NIR and VIR, we consider as termination critertion the merit function $m(x) = V_{\alpha\beta}(x)$. The initial point is $x_0 = (1, 1, 1)$.

	$V(.)$ call	$\nabla V(.)$ call	$\psi(.,.)$ call	$\nabla\psi(.,.)$ call	$\|\|m(x^\star)\|\|$	Code
BB-NIR	19	19	61,457	15,504	1.08e-06	2
BB-VIR	25	25	85,826	24,424	1.07e-06	2
BFGS-NIR	74	62	215,628	54,702	8.04e-07	1
BFGS-VIR	40	19	100,128	28,531	1.47e-07	1
CG-NIR	201	101	479,382	120,231	0.0194	4
CG-VIR	123	52	295,865	85,110	0.00194	2

Table 6: Results for the minimization reformulation

As shown in Table 6, only the BFGS method converge, while the BB and the CG methods remains stuck except for the CG-NIR which exhausts the maximum number of iterations. So, there is a clear advantage of the BFGS method. The best estimation of the normalized NE of the river-basin pollution game is $x^\star = (21.46060005, 16.00840374, 2.53209932)$.

3.5.2 Fixed-point reformulation

The results for the fixed-point reformulation are now presented. As described in Section 1.4, the GNEP is reformulated as an unconstrained fixed-point problem (20) for NIR and (17) for VIR (as in [59], we set $\alpha = 0.02$). The different algorithms of Sections 3.3 and 3.4 are denoted as follows: 'pure' for the Picard method, 'relax' for the relaxation method developped in [34], 'grelax' for a line-search-globalized relaxation method developped in [57], 'RRE' and 'MPE' for extrapolation methods RRE1 and MPE1, 'SqRRE' and 'SqMPE' for the corresponding squared version. There are two reformulations either Nikaido-Isoda reformulation (NIR) or Variational Inequality reformulation (VIR).

For both reformulations, we consider as termination criterion either the merit functions $m(x) = \psi(x, y(x))$ (either NI or VI) evaluated at current iterate x_k or no merit function (FP) so that the termination criterion is $||x_k - x_{k-1}||$. The initial point is chosen as $x_0 = (1, 1, 1)$ (similar results are obtained $x_0 = (25, 25, 25)$). Table 7 report the number of calls to the function $y(.)$, the merit function $m(.)$, the regularized gap function $\psi(.,.)$ and its gradient $\nabla\psi(.,.)$, the termination criterion at final iterate and the code.

	$y(.)$ call	$m(.)$ call	$\psi(.,.)$ call	$\nabla\psi(.,.)$ call	$\|\|m(x^\star)\|\|$	Code
pure-NIR-NI	29	29	23,490	5,355	6.41e-07	1
pure-NIR-FP	61	0	48,881	11,278	21.1	2
pure-VIR-VI	20	20	16,239	4,132	9.86e-07	1
pure-VIR-FP	100	0	79,542	20,435	21.1	4
relax-NIR-NI	101	101	158,369	36,838	0.172	4
relax-NIR-FP	101	101	79,126	18,474	0.00395	4
relax-VIR-VI	101	101	160,869	41,195	0.131	4
relax-VIR-FP	101	101	80,855	20,677	0.00363	4
grelax-NIR-NI	101	580	600,597	142,395	0.0907	4
grelax-VIR-VI	101	3,042	2,609,630	666,968	0.0443	4
RRE-NIR-NI	41	21	49,925	11,473	-1.47e-07	1
RRE-NIR-FP	29	0	23,555	5,368	21.1	2
RRE-VIR-VI	13	7	16,245	4,138	5.52e-07	1
RRE-VIR-FP	101	0	81,503	20,882	21.1	4
MPE-NIR-NI	43	22	52,048	11,980	2.61e-08	1
MPE-NIR-FP	38	0	30,557	7,031	21.1	2
MPE-VIR-VI	21	11	25,905	6,617	1.56e-08	1
MPE-VIR-FP	101	0	81,331	20,819	21.1	4
SqRRE-NIR-NI	15	8	18,565	4,229	5.14e-07	1
SqRRE-NIR-FP	74	0	59,015	13,639	21.1	2
SqRRE-VIR-VI	15	8	18,572	4,765	1.94e-07	1
SqRRE-VIR-FP	101	0	81,567	20,848	21.1	4
SqMPE-NIR-NI	15	8	18,710	4,259	-6.21e-07	1
SqMPE-NIR-FP	71	0	56,809	13,108	21.1	2
SqMPE-VIR-VI	19	10	23,666	6,015	9.38e-07	1
SqMPE-VIR-FP	104	0	83,959	21,494	21.1	4

Table 7: Results for the FP reformulation

As given in Table 7, relaxation methods 'relax' and 'grelax' do not converge within 100 (outer) iterations, leading to a high number of calls to ψ and $\nabla\psi$. The pure fixed-point iteration performs relatively well compared to its simplicity, then comes the extrapolation methods MPE and RRE, and finally their squared version. As described in [50], squared methods outperform the simple extrapolation methods whatever the problem is when the merit function is specified (i.e. not for FP). Our best estimation of the normalized NE is $x^\star = (21.12806751, 16.03599004, 2.736678647)$.

4 Conclusion

The generalized Nash equilibrium problem (GNEP) is of particular importance when modelling concrete applications, e.g. in economics, in computer science and in biology. The demand for effective computational methods of the GNEP in general form is increasing. This survey paper provided a large panel of optimization methods available for the general GNEP and the jointly convex GNEP. Our numerical experiments showed an advantage for the KKT reformulation of the GNEP compared to the constrained equation reformulation, yet, in [12], the constrained equation reformulation was better. A method working for any general GNEP has yet to be found and its convergence to be proved. Regarding jointly convex GNEP, there is a clear advantage to use the fixed-point reformulation especially solved with extrapolation techniques. In this paper, we also propose new optimization methods solving the GNEP by considering quasi-Newton methods (the

Broyden method for general games and the BFGS method for jointly convex games), as well as extrapolation methods when solving the GNEP as a fixed-point.

This work was partially funded by the AXA research fund and the Swiss National Science Foundation Project 200021-124635/1.

References

[1] D. Aussel, R. Correa, and M. Marechal. Gap functions for quasivariational inequalities and generalized Nash equilibrium problems. *Journal of Optimization Theory and Applications*, 151(3):474–488, 2011.

[2] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. Wiley interscience, 2006.

[3] S. Bellavia, M. Macconi, and B. Morini. An affine scaling trust-region approach to bound-constrained nonlinear systems. *Applied Numerical Mathematics*, 44(3):257–280, 2003.

[4] D.P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, Massachusetts, 1999.

[5] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal. *Numerical Optimization: Theoretical and Practical Aspects, Second edition*. Springer-Verlag, 2006.

[6] C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, 19(92):577–593, 1965.

[7] F. H. Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205(1):247–262, 1975.

[8] F. H. Clarke and D. N. Bessis. Partial subdifferentials, derivates and Rademacher's theorem. *Transactions of the American Mathematical Society*, 351(7):2899–2926, 1999.

[9] Frank H. Clarke. *Optimization and Nonsmooth Analysis*. SIAM, 1990.

[10] J. E. Dennis and Jorge J. Morée. Quasi-newton methods, motivation and theory. *SIAM Review*, 19(1), 1977.

[11] J. E. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. SIAM, 1996.

[12] A. Dreves, Francisco Facchinei, Christian Kanzow, and S. Sagratella. On the solutions of the KKT conditions of generalized Nash equilibrium problems. *SIAM Journal on Optimization*, 21(3):1082–1108, 2011.

[13] C. Dutang. *GNE: computation of Generalized Nash Equilibria*, 2013. R package version 0.97.

[14] Francisco Facchinei. Minimization of SC1 functions and the Maratos effect. *Operations Researchs Letters*, 17(3):131–137, 1995.

[15] Francisco Facchinei, Andreas Fischer, and Veronica Piccialli. On generalized Nash games and variational inequalities. *Operations Research Letters*, 35(2):159–164, 2007.

[16] Francisco Facchinei, Andreas Fischer, and Veronica Piccialli. Generalized Nash equilibrium problems and Newton methods. *Math. Program., Ser. B*, 117(1-2):163–194, 2009.

[17] Francisco Facchinei and Christian Kanzow. A nonsmooth inexact Newton method for the solution of large-scale nonlinear complementarity problems. *Mathematical Programming*, 76(3):493–512, 1997.

[18] Francisco Facchinei and Christian Kanzow. Generalized Nash equilibrium problems. Updated version of the 'quaterly journal of operations research' version, 2009.

[19] Francisco Facchinei and Jong-Shi Pang. *Finite-Dimensional Variational Inequalities and Complementary Problems. Volume I.* In Operations Research. Springer-Verlag, New York, Berlin, Heidelberg, 2003.

[20] Francisco Facchinei and Jong-Shi Pang. *Finite-Dimensional Variational Inequalities and Complementary Problems. Volume II.* In Operations Research. Springer-Verlag, New York, Berlin, Heidelberg, 2003.

[21] Jin-Yan Fan. A modified Levenberg-Marquardt algorithm for singular system of nonlinear equations. *Journal of Computational Mathematics*, 21(5):625–636, 2003.

[22] Jin-Yan Fan and Ya-Xiang Yuan. On the quadratic convergence of the Levenberg-Marquardt Method without nonsingularity assumption. *Computing*, 74(1):23–39, 2005.

[23] Andreas Fischer. Local behavior of an iterative framework for generalized equations with nonisolated solutions. *Math. Program., Ser. A*, 94(1):91–124, 2002.

[24] Masao Fukushima and Jong-Shi Pang. Quasi-variational inequalities, generalized Nash equilibria, and multi-leader-follower games. *Comput. Manag. Sci.*, 2:21–56, 2005.

[25] Masao Fukushima and Liqun Qi, editors. *Reformulation - Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods.* Kluwer Academic Publishers, 1999.

[26] L. Grippo, F. Lampariello, and S. Lucidi. A nonmonotone line search technique for Newton's method. *SIAM Journal on Numerical Analysis*, 23(4):707–716, 1986.

[27] C.M. Ip and J. Kyparisis. Local convergence of quasi-Newton methods for B-differentiable equations. *Mathematical Programming*, 56(1-3):71–89, 1992.

[28] A.F. Izmailov and M.V. Solodov. On error bounds and Newton-type methods for generalized Nash equilibrium problems. Moscow State University and Instituto de Matemática Pura e Applicada, 2012.

[29] Houyan Jiang. Global convergence analysis of the generalized Newton and Gauss-Newton methods for the Fischer-Burmeister equation for the complementarity problem. *Mathematics of Operations Research*, 24(3):529–543, 1999.

[30] Houyan Jiang, Masao Fukushima, Liqun Qi, and Defeng Sun. A Trust Region Method for Solving Generalized Complementarity Problem. *SIAM Journal on Optimization*, 8(1), 1998.

[31] Houyan Jiang, Liqun Qi, Xiaojun Chen, and Defeng Sun. Semismoothness and superlinear convergence in nonsmooth optimization and nonsmooth equations. In *Nonlinear Optimization and Applications.* Plenum Press, 1996.

[32] Houyan Jiang and D. Ralph. Global and local superlinear convergence analysis of Newton-type methods for semismooth equations with smooth least squares. In Masao Fukushima and Liqun Qi, editors, *Reformulation - nonsmooth, piecewise smooth, semismooth and smoothing methods*. Boston MA: Kluwer Academic Publishers, 1998.

[33] Christian Kanzow and Helmut Kleinmichel. A new class of semismooth Newton-type methods for nonlinear complementarity problems. *Computational Optimization and Applications*, 11(3):227–251, 1998.

[34] J.B. Krawczyk and S. Uryasev. Relaxation algorithms to find Nash equilibria with economic applications. *Environmental Modeling and Assessment*, 5(1):63–73, 2000.

[35] Katsunori Kubota and Masao Fukushima. Gap function approach to the generalized Nash equilibrium problem. *Journal of Optimization Theory and Applications*, 144(3):511–531, 2010.

[36] A. A. Kulkarni and U. V. Shanbhag. Revisiting generalized Nash games and variational inequalities. preprint, 2010.

[37] K. Lange. *Optimization*. Springer, 2004.

[38] V. L. R. Lopes and J. M. Martinez. On the convergence of quasi-Newton methods for nonsmooth problems. preprint, 1999.

[39] R.D.C. Monteiro and Jong-Shi Pang. A Potential Reduction Newton Method for Constrained equations. *SIAM Journal on Optimization*, 9(3):729–754, 1999.

[40] Hukukane Nikaido and Kazuo Isoda. Note on non-cooperative convex games. *Pacific J. Math.*, 5(1):807–815, 1955.

[41] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Science+Business Media, 2006.

[42] M.J.D. Powell. A hybrid method for nonlinear algebraic equations. In P. Rabinowitz, editor, *Numerical Methods for Nonlinear Algebraic Equations*, chapter 6. Gordon & Breach, 1970.

[43] Liqun Qi. Convergence analysis of some algorithms for solving nonsmooth equations. *Mathematics of Operations Research*, 18(1):227–244, 1993.

[44] Liqun Qi. On superlinear convergence of quasi-Newton methods for nonsmooth equations. *Operations Researchs Letters*, 20(5):223–228, 1997.

[45] Liqun Qi and Houyan Jiang. Semismooth KKT equations and convergence analysis of Newton and Quasi-Newton methods for solving these equations. *Mathematics of Operations Research*, 22(2):301–325, 1997.

[46] Liqun Qi and Defeng Sun. A Survey of Some Nonsmooth Equations and Smoothing Newton Methods. Technical report, Applied Mathematics Report AMR 98/10, School of Mathematics, the University of New South Wales, 1998.

[47] Liqun Qi and J. Sun. A nonsmooth version of Newton's method. *Mathematical Programming*, 58(1-3):353–367, 1993.

[48] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013.

[49] M. Raydan. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. *SIAM Journal on Optimization*, 7(1):26–33, 1997.

[50] C. Roland, R. Varadhan, and C. E. Frangakis. Squared polynomial extrapolation methods with cycling: an application to the positron emission tomography problem. *Numerical Algorithms*, 44(2):159–172, 2007.

[51] A. Sidi. Efficient implementation of minimal polynomial and reduced rank extrapolation methods. *Journal of Computational and Applied Mathematics*, 36:305–337, 1991.

[52] Leo Simon. Mathematical methods. Technical report, Berkeley, Lecture notes, 2011.

[53] Defeng Sun and Jiye Han. Newton and Quasi-Newton methods for a class of nonsmooth equations and related problems. *SIAM Journal on Optimization*, 7(2):463–480, 1997.

[54] R. Varadhan. Squared Extrapolation methods (SQUAREM): a new class of simple and efficient numerical schemes for accelerating the convergence of the EM algorithm. working paper, 2004.

[55] R. Varadhan. Simple and globally convergent methods for accelerating the convergence of any em algorithm. *Scandinavian Journal of Statistics*, 35(2):335–353, 2008.

[56] R. Varadhan. *alabama: Constrained nonlinear optimization*, 2011. R package version 2011.9-1.

[57] Anna von Heusinger. *Numerical methods for the solution of the generalized Nash equilibrium problem*. PhD thesis, Universität Würzburg, 2009.

[58] Anna von Heusinger and Christian Kanzow. SC1 optimization reformulations of the generalized Nash equilibrium problem. *Optimization Methods and Software*, 23(6):953–973, 2008.

[59] Anna von Heusinger and Christian Kanzow. Optimization reformulations of the generalized Nash equilibrium problem using the Nikaido-Isoda type functions. *Computational Optimization and Applications*, 43(3), 2009.

[60] Anna von Heusinger, Christian Kanzow, and Masao Fukushima. Newton's method for computing a normalized equilibrium in the generalized Nash game through fixed point formulation. *Mathematical Programming*, 123(1):1–25, 2010.

[61] T. Wang, R.D.C. Monteiro, and Jong-Shi Pang. An interior point potential reduction method for constrained equations. *Mathematical Programming*, 74(2):159–195, 1996.

[62] Nobuo Yamashita and Masao Fukushima. On the rate of convergence of the Levenberg-Marquardt method. Technical report, Kyoto University, 2000.

[63] V.A. Zorich. *Mathematical Analysis I*, volume 1. Universitext, Springer, 2000.

A The KKT system

The generalized Jacobian $J(z)$ of the complementarity formulation has the following form

$$\left(\begin{array}{ccc|cc} \mathrm{Jac}_{x_1} L_1(x,\lambda^1) & \dots & \mathrm{Jac}_{x_N} L_1(x,\lambda^1) & \mathrm{Jac}_{x_1} g^1(x)^T & 0 \\ \vdots & & \vdots & & \ddots \\ \mathrm{Jac}_{x_1} L_N(x,\lambda^N) & \dots & \mathrm{Jac}_{x_N} L_N(x,\lambda^N) & 0 & \mathrm{Jac}_{x_N} g^N(x)^T \\ \hline -D_1^a(x,\lambda^1)\mathrm{Jac}_{x_1} g^1(x) & \dots & -D_1^a(x,\lambda^1)\mathrm{Jac}_{x_N} g^1(x) & D_1^b(x,\lambda^1) & 0 \\ \vdots & & \vdots & & \ddots \\ -D_N^a(x,\lambda^N)\mathrm{Jac}_{x_1} g^N(x) & \dots & -D_N^a(x,\lambda^N)\mathrm{Jac}_{x_N} g^N(x) & 0 & D_N^b(x,\lambda^N) \end{array}\right).$$

B Globalization techniques

Globalization schemes are briefly presented. There are mainly two frameworks: line search and trust-region methods.

B.1 Line-search techniques

Line-search techniques are a refinement of the local sequence by considering the sequence $z_{k+1} = z_k + t_k d_k$ where $t_k \in]0,1]$ is the stepsize[14] in direction d_k at the current iterate z_k. Overall, an (outer) iteration is done in two steps : the computation of the direction d_k and the computation of the stepsize t_k. Line-search techniques propose criteria to choose t_k in a certain number of iterations called inner iterations. As the stepsize may reduces the (full) step from z_k to z_{k+1}, line-search version of an algorithm is sometimes called the damped version of that algorithm.

Let f be a merit function. We define the function $t \mapsto \phi_k(t) = f(z_k + t d_k)$. We want to find a good minimizer of ϕ_k. However, it is useless to find the global minimizer $\arg\min\ \phi_k(t)$, because we want to solve the outer problem $F(z) = 0$, and not the inner problem $\min\ \phi_k(t)$. In the following, we assume we have a descent direction d_k for the merit function f, as a minimal condtion to choose t_k is $f(z_{k+1}) < f(z_k)$. This descent direction condition translates to $\phi_k'(0) < 0$. We are focused on two things, t_k should be big enougth to ensure a sufficient decrease of ϕ, and also t_k should not be too small to guarantee a sufficient big step.

One could think that requiring $f(z_{k+1}) < f(z_k)$ is enough to show convergence, but unfortunately not. In literature, see, e.g., [11, 41, 5], two typical conditions are used to determine an appropriate stepsize. Let $0 < c_1 < 1/2 < c_2 < 1$. The Armijo condition ensures a decrease of f

$$\phi_k(t) \leq \phi_k(0) + t c_1 \phi_k'(0) \Leftrightarrow f(x_k + t d_k) \leq f(x_k) + t c_1 \nabla f(x_k)^T d_k.$$

The curvature condition ensures an increase of $\nabla\phi$, implying a decrease of f,

$$\phi_k'(t) \geq c_2 \phi_k'(0) \Leftrightarrow \nabla f(x_k + t d_k)^T d_k \geq c_2 \nabla f(x_k)^T d_k.$$

[14] Considering the stepsize t_k to $]0,1]$ is not too restrictive since the direction d_k is not unitary, i.e. $||d_k|| >> 1$.

These two conditions are referred to the Wolfe conditions. In this paper, we use a backtracking algorithm, for which the curvature condition is always satisfied. Let $t_{k,0} = 1$ be the initial guess of the stepsize. The backtracking algorithm is defined as follows

Repeat until $f(x_k + td_k) \leq f(x_k) + t_{k,i} c_1 \nabla f(x_k)^T d_k$ satisfied,

- propose a new $t_{k,i+1}$ using $t_{k,i}, \dots t_{k,0}$.

end Repeat

This algorithm always tests a full step with $t_{k,0} = 1$. For the backtracking line search, a classic result shows that the full step will be eventually satisfied as z_k tends to a solution. In practice, we test two stepsize proposal algorithms. The geometric line search uses

$$t_{k,i+1} = \rho \times t_{k,i},$$

with $0 < \rho < 1$, whereas the quadratic line search uses a quadratic approximation of ϕ using the information $\phi_k(t_{k,i})$, $\phi_k(t_{k,i-1})$, $\phi'_k(t_{k,i-1})$. We get

$$t_{k,i+1} = -\frac{1}{2} \frac{\phi'_k(t_{k,i-1}) t_{k,i}^2}{\phi_k(t_{k,i}) - \phi_k(t_{k,i-1}) - \phi'_k(t_{k,i-1}) t_{k,i}}.$$

Other proposal, such as cubic approximation, are e.g. given in [5] and [11].

Until now, we do not specify the merit function f. For nonlinear equation, we generally choose $f(z) = \frac{1}{2}||F(z)||_2^2$, sometimes referred to the residual function. This merit function has some defficiencies, since a local minimum is not necessarily a root of the function F. We will see later in the GNEP context, that f has still some interesting properties. Line-search methods require to a tractable formula for the gradient $\nabla f(z) = \mathrm{Jac} F(z)^T F(z)$, when testing the Armijo condition. However, in a quasi-Newton framework, we do not necessarily have a tractable Jacobian. A typical way to deal with this is to use a numerical Jacobian, e.g., based on the forward difference. We use [11]'s algorithm A5.4.1 defined by

$$D(F)(z) = (D_1, \dots, D_n), \quad \text{with} \quad D_j = \frac{F(z + h_j e_j) - F(z)}{h_j} \in \mathbb{R}^n,$$

where e_j is the jth unit vector and h_j a small step, typically, $h_j = \sqrt{\epsilon} z_j$ where ϵ is the epsilon machine ($\epsilon = 1e^{-16}$).

B.2 Trust-region approach

Trust-region strategies relaxe the constraint that d_k is a descent direction. Line search assumes the "best" point from z_k lies on the half-line $z_k + \mathbb{R}_+ d_n$. Quoting [5], "what is magic in this half line? answer: nothing". Trust-region approach will look for appropriate steps h_k in a "small" region around z_k. Such regions are not the half-line as in line search.

To find the root of $F(x) = 0$, trust-region methods minimizes a local quadratic approximation m_k of a merit function $f : \mathbb{R}^n \mapsto \mathbb{R}$ on a bounded subset: the trust region $\{z, ||z - z_k|| \leq \Delta_k\}$. The

name comes from viewing Δ_k as providing a region in which we can trust m_k to adequately model f. Classic trust region methods consider the following model function

$$m_k(h) = f(z_k) + g(x_k)^T h + \frac{1}{2} h^T \tilde{H}_k h,$$

with $g(x_k)$ the approximate gradient of f and $\tilde{H}_k$ a (ideally) positive definite matrix approximating the Hessian of f.

To adapt the trust region radius Δ_k, we define the following ratio

$$\rho_k(h) = \frac{f(x_k) - f(x_k + h)}{m_k(0) - m_k(h)}.$$

$\rho_k(h)$ is the ratio of the actual reduction and the predicted reduction of the merit function for a step h. The higher is $\rho_k(h)$, the higher is the reduction of the merit function f for a given h. Now, we can define a generic algorithm for a model function m_k, see e.g. [41] for a recent survey.

Init $\Delta_0 = 1, \eta_0 > 0, 0 < \eta_1 < \eta_2 < 1$ and $0 < \gamma_1 < 1 < \gamma_2$.

Iterate until a termination criterion is satisfied,

- get $h_k = \underset{||h|| < \Delta_k}{\arg\min}\ m_k(h)$ (approximately),
- compute $\rho_k(h_k)$,
  - if $\rho_k(h_k) < \eta_1$ then $\Delta_{k+1} = \gamma_1 \Delta_k$ (unsuccessful),
  - else if $\rho_k(h_k) > \eta_2$ and $||h_k|| = \Delta_k$ then $\Delta_{k+1} = \min(\gamma_2 \Delta_k, \Delta_{max})$ (very successful),
  - else $\Delta_{k+1} = \Delta_k$.
- next iterate
  - if $\rho_k(h_k) > \eta_0$ then $x_{k+1} = x_k + h_k$,
  - else $x_{k+1} = x_k$.

end Iterate

Typical values of parameters are $\Delta_0 = 1$ or $\frac{||g_0||}{10}$, $\Delta_{max} = 10^{10}$ for radius bounds, $\eta_0 = 10^{-4}$, $\eta_1 = \frac{1}{4}$, $\eta_2 = \frac{3}{4}$ for ratio threshold, $\gamma_1 = \frac{1}{2}$ and $\gamma_2 = 2$ for radius expansion coefficients.

If readers have been attentive, then they should have noticed that the algorithm cannot be used directly. In fact, we have to determine how to compute the approximate solution h_k of the following minimization problem

$$\min_{||h|| < \Delta_k} m_k(h).$$

As for line search techniques, this problem has to be solved approximately as this problem is not our primary concern. There are two heuristic methods to achieve this: Powell's dogleg and double dogleg methods. The Powell dogleg method uses a linear approximation of the model function $m_k(h)$, see e.g. Chapter 6 of [42]. Let p_k^S be the scaled steepest descent direction and p_k^N be the Newton point defined as

$$p_k^S = -\frac{g_k^T g_k}{g_k^T \tilde{H}_k g_k} g_k \quad \text{and} \quad p_k^N = -\tilde{H}_k^{-1} g_k.$$

The Powell dogleg method is as follows.

- **If** $||p^N|| \leq \Delta_k$, then $h^\star = p^N$.
- **Else if** $||p^S|| \geq \Delta_k$, then $h^\star = \Delta_k / ||p^S|| \times p^S$.
- **Else**, we choose a convex combination between the two points p^S and p^N. That is we find a $\lambda \in [0,1]$ such that $||p^S + \lambda(p^N - p^S)|| = \Delta_k$. We get $h^\star = \lambda^\star p^N + (1-\lambda^\star)p^S$ with

$$\lambda^\star = \frac{- < p^S, p^N - p^S > + \sqrt{< p^S, p^N - p^S >^2 - ||p^N - p^S||^2 (||p^S||^2 - \Delta_k)}}{||p^N - p^S||^2},$$

  where $< .,. >$ denotes the scalar product and $||.||$ denotes the Euclidean norm.

[41] also propose a "simple" dogleg method which remove the step $||p^S|| \geq \Delta_k$, see Algorithm 11.6. The double dogleg method finds an approximate solution of the model function $m_k(h)$ assuming the Hessian matrix is $\tilde{H}_k = L^T L$. The double dogleg method is a variant Powell dogleg method using a forcing parameter η_k. [11] propose the following procedure.

- **If** $||p^N|| \leq \Delta_k$, then $h^\star = p^N$.
- **Else if** $\eta_k ||p^N|| \leq \Delta_k$, then $h^\star = p^N \times \eta_k / \Delta_k$.
- **Else if** $||p^S|| \geq \Delta_k$ then $h^\star = p^S \times \Delta_k / ||p^S||$.
- **Else**, we choose λ such that $||p^S + \lambda(\eta_k p^N - p^S)|| = \Delta_k$. We get back to the Powell dogleg case with $\eta_k p^N$ instead of p^N.

where the parameter $\eta_k \leq 1$ is defined as $\eta_k = 0.2 + 0.8\alpha^2/(\beta |g_k^T d^N|)$, with $\alpha = ||g_k||^2, \beta = ||Lg_k||^2$. As for line-search techniques, we use the merit function $f(z) = \frac{1}{2}||F(z)||_2^2$. We recall that the gradient is given by

$$g(z) = \mathrm{Jac}\, F(z)^T F(z).$$

Therefore, the approximated Hessian $\tilde{H}_k$ is $\mathrm{Jac}\, F(z_k)^T \mathrm{Jac}\, F(z_k)$ as in the Gauss-Newton model. Hence, the steepest descent point and the Newton point have the following expression

$$p_k^S = -\frac{g_k^T g_k}{g_k^T J_k^T J_k g_k} g_k \quad \text{and} \quad p_k^N = -J_k^T g_k.$$

As in the previous subsection, when working a quasi-Newton method, the Jacobian is numerically approximated by a forward difference.

B.3 The special case of the Levenberg-Marquardt method

Until now, all globalization methods are adapted for the Newton or the Broyden direction defined in Equations (21) and (22). We need to precise how to globalize the Levenberg-Marquardt direction. This method was introduced in the context of the least-square problem $\min \frac{1}{2}||F(z)||_2^2$. In fact, there is a relation between the trust-region approach and the Levenberg-Marquardt method. The solution to the quadratic problem

$$\min_{||h|| < \Delta_k} f(z_k) + g(x_k)^T h + \frac{1}{2} h^T \tilde{H}_k h$$

is equivalent to the problem of finding $\lambda^\star, h^\star$ such that

$$(\tilde{H}_k + \lambda^\star I)h^\star = -g(x_k) \quad \text{and} \quad \lambda^\star(\Delta_k - ||h^\star||) = 0, \tag{47}$$

with the condition that $\tilde{H}_k + \lambda^\star I$ is a positive semidefinite matrix. We note in Equation (47) that variable $\lambda^\star$ has the same role as in the Levenberg-Marquardt method.

We can easily interpret Equation (47). If $h^\star$ lies strictly inside the trust-region then parameter $\lambda^\star$ is zero. Otherwise, $h^\star$ hits the radius Δ_k, and then parameter $\lambda^\star$ is set to a positive value. With this interpretation, the use of a trust-region approach with the Levenberg-Marquardt method is redundant.

However, we still consider a globalization strategy for the Levenberg-Marquardt method. Firstly, we test the geometric line-search strategy, defined in Subsection B.1, which is proposed in [22], [62]. Secondly, we use the adaptive Levenberg-Marquardt method discussed in [21]. The method consists in adjusting the parameter λ_k based on $\lambda_k = \mu_k||F(z_k)||_2$, where μ_k is udpated at each iteration depending on the value of the ratio ρ_k. Their Algorithm 2.2 updates μ_k as in the generic algorithm of the previous Subsection B.2. So, we do not restate here the updating scheme.

B.4 Convergence criteria in the differentiable setting

Let f be the merit function. The global convergence of line-search techniques is guaranteed when we have the following conditions: (i) the set $\mathcal{L} = \{z, f(z) \leq f(z_0)\}$ is bounded, (ii) the Jacobian is bounded in an open convex set around a solution $z^\star$ and (iii) line-search always satisfies the Wolfe conditions with a descent direction, see, e.g. Theorem 11.6 of [41]. We have seen that the backtracking algorithm satisfies the Wolfe conditions at each step.

The convergence of trust-region strategies is proved with similar conditions and requires also that the set $\mathcal{L}$ and the Jacobian are bounded. Furthermore, the approximated solution of the quadratic local problem $\min\ m_k(h)$ such that $||h|| < \Delta_k$ must verify two conditions: (i)

$$m_k(0) - m_k(h^\star) \geq c_1||J_k^T F_k|| \min(\Delta_k, ||J_k^T F_k||/||J_k^T J_k||)$$

for some $c_1 > 0$ and (ii) $||h^\star\ || \leq \gamma\Delta_k$ for some $\gamma \geq 1$.

B.5 Global convergence in the semismooth setting

Let us study the global convergence starting by line-search techniques. The merit function used is the same function as for the differentiable setting the residual norm.

$$f(z) = \frac{1}{2}||\ F(z)||^2.$$

The gradient is given by $\nabla f(z) = V^T F(z)$, where $V \in \partial F(z)$. As mentioned in [46], the merit function may be C^1, even if F is only semismooth.

[32] and [46] show the convergence of the Newton method globalized with a backtracking (geometric) line search. When using the generalized Jacobian, [29] shows the convergence of the

corresponding algorithm. All proofs rely on the fact that after a large number of iteration, the full Newton step is accepted, i.e. we get back to local convergence.

Theorem 21. *Suppose that the root function F is a semismooth function and the merit function f is C^1. Then any accumulation points $z^\star$ of the line-search generalized Newton method is a stationary point of f, i.e. $\nabla f(z^\star) = 0$. If $z^\star$ is a solution of $F(z) = 0$ and all matrices in $\partial_B F(z^\star)$ are nonsingular, then the whole sequence converges superlinearly (resp. quadratically) to $z^\star$ (if F is strongly semismooth at $z^\star$).*

To our knowledge, the global convergence of general quasi-Newton methods for nonsmooth equation is not established. However, for the Levenberg-Marquardt with a backtracking line-search technique, [32] and [29] show the convergence. We present below a light version of their theorem.

Theorem 22. *Let $(z_k)_k$ be a sequence of the generalized LM method globalized with a backtracking line search to solve $F(x) = 0$ for a semismooth function F and a C^1 merit function f. Assuming the direction step is solvable at each iteration, we denote by $z^\star$ an accumulation point. If $\lambda_k = \min(f(z_k), ||\nabla f(z_k)||)$ and elements in $\partial_B F(z^\star)$ are nonsingular, then $(z_k)_k$ converges superlinearly. If in addition F is strongly semismooth at $z^\star$, then it converges quadratically.*

Now, we focus on the second globalization strategy: the trust-region approach. We present first a convergence result of [32], based on a result of [30] in the context of complementarity problems.

Theorem 23. *Consider $F(z) = 0$ with F semismooth and a C^1 merit function f? Let (z_k) be generated by the trust-region algorithm with the submodel given $m_k(h) = J_k^T F(x_k)h + \frac{1}{2}h^T J_k^T J_k h$. If $z^\star$ is an accumulation point of the sequence, then $z^\star$ is a stationary point of f. If for all element of $\partial_B F(z^\star)$ are nonsingular, then the entire sequence converges to $x^\star$ superlinearly. If in addition, F is strongly semismooth at $z^\star$, then the convergence rate is quadratic.*

C Nonsmooth analysis

Definition 15 (locally Lipschitzian)**.** *G is locally Lipschitzian (on $\mathbb{R}^n$) if $\forall x \in \mathbb{R}^n, \exists U \in \mathcal{N}(x), \forall y, z \in U, \exists k_x > 0, ||G(y) - G(z)|| \leq k_x||y - z||$.*

Theorem 24 (from [8])**.** *Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ be a locally Lipschitz function. Then f is almost everywhere differentiable.*

From [9, Cor 2.2.4, Chap. 2], for a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ locally Lipschitzian at x, we have that the generalized gradient $\partial f(y)$ is a singleton for all $y \in B(x, \epsilon)$ is equivalent to f is C^1 on $B(x, \epsilon)$. From [9, Prop 2.6.2, Chap. 2], we have the following properties of the generalized Jacobian.

Proposition 7.

- *$\partial G(x)$ is a nonempty, convex, compact subset of $\mathbb{R}^{m\times n}$, while $\partial_B G(x)$ is nonempty and compact.*
- *∂G is upper semicontinuous and closed at x and $\partial_B G$ is upper semicontinuous.*
- *$\partial G(x) \subset \partial G_1(x) \times \cdots \times \partial G_m(x)$, where the right-hand side is a matrix set where the ith row is the generalized gradient.*

The term $\partial G_1(x) \times \cdots \times \partial G_m(x)$ is sometimes denoted by $\partial_C G(x)$. But it is not Clarke's subdifferential, which seems to refer only to real-valued function, i.e. $\partial G(x) = \partial_C G(x)$.

Proposition 8 (From Theorem 2.3 of [47])**.**

- *G is semismooth at x.*
- $\forall V \in \partial G(x+h), h \to 0, Vh - G'(x;h) = o(||h||)$.
- $\forall x \in D_G, G'(x+h;h) - G'(x;h) = o(||h||)$.

From Lemma 2.2 of [47] and Lemma 2.1 of [53], we have the following properties

Proposition 9. *If G is semismooth at x, then $d \mapsto G'(x;d)$ is a Lipschitz function; $\forall h, \exists V \in G(x), Vh = G'(x;h)$ and $\forall h \to 0, G(x+h) - G(x) - G'(x;h) = o(||h||)$.*