RELAXATION METHODS FOR GENERALIZED NASH EQUILIBRIUM PROBLEMS WITH INEXACT LINE SEARCH

Anna von Heusinger and Christian Kanzow

Preprint 282 February 2008

University of Würzburg
Institute of Mathematics
Am Hubland
97074 Würzburg
Germany

e-mail: heusinger@mathematik.uni-wuerzburg.de
kanzow@mathematik.uni-wuerzburg.de

February 21, 2008

Abstract. The generalized Nash equilibrium problem (GNEP) is an extension of the standard Nash game where both the utility functions and the strategy spaces of each player also depend on the strategies chosen by all other players. This problem is rather difficult to solve, and there are only a few methods available in the literature. One of the most popular ones is the so-called relaxation method which is known to be globally convergent under a set of assumptions. Some of these assumptions, however, are rather strong or somewhat difficult to understand. Here we present a modified relaxation method for the solution of a certain class of GNEPs. The convergence analysis is based on completely different arguments and avoids some of the technical conditions for the original relaxation method. Moreover, numerical experiments indicate that the modified relaxation method performs quite well on a number of different examples taken from the literature.

Key Words: Generalized Nash equilibrium problem; Normalized Nash equilibrium; Relaxation method; Regularized Nikaido-Isoda function; Global convergence.

1 Introduction

The generalized Nash equilibrium problem (GNEP for short) has recently attracted much attention. It differs from the standard Nash equilibrium problem (NEP for short) by allowing the strategy spaces of each player to depend on the strategies chosen by the other players. These kind of Nash games occur quite frequently in different models, e.g., when the pollution of a river or the air within a certain area depends on the output of several firms (players) and is not allowed to go beyond a certain limit, when several companies have to share the same set of wires whose capacities are bounded, or when the world's countries have to share a common good like all natural resources. The reader is referred to the survey paper [9] and the references therein for more applications, some theoretical results and an overview of existing methods for the solution of GNEPs.

One of the most popular methods for the solution of GNEPs is the so-called relaxation method that was introduced in [26], see also [19] for a modified version. While basically none of the existing solvers for GNEPs has been tested extensively on a large variety of problems, the relaxation method seems to be the only one that has been applied at least by a small group of different people to a few problems coming from different applications, see [1, 2, 5, 6, 14, 18, 19]. However, the conditions that guarantee convergence of the relaxation method in [26, 19] are very restrictive. Moreover, the rather general inexact stepsize rule given in [26] leads to more or less heuristic implementations of the relaxation method, whereas the exact stepsize rule from [19] is not really implementable, see the comments below for more details.

Our aim is therefore to present a new convergence theory for the relaxation method that allows weaker assumptions and that uses a clear (Armijo-type) rule for the choice of an inexact stepsize that turns out to provide rather good numerical results.

To be more specific, let us now give a formal definition of the Nash and the generalized Nash equilibrium problems (see, e.g., [3, 4, 7, 11, 12, 21] for some standard books on game theory and, in particular, Nash equilibrium problems): Both problems have N players, each player $\nu \in \{1, \ldots, N\}$ controls the variables $x^\nu \in \mathbb{R}^{n_\nu}$. Let $x = (x^1, \ldots, x^N)^T \in \mathbb{R}^n$ be the vector comprised by all these decision variables, where $n := n_1 + \ldots + n_N$. To emphasize the νth player's variables within the vector x, we sometimes write $x = (x^\nu, x^{-\nu})^T$, where $x^{-\nu}$ subsumes all the other players' variables. Moreover, for both NEPs and GNEPs, let $\theta_\nu : \mathbb{R}^n \to \mathbb{R}$ be the νth player's payoff (or loss or utility) function.

For a standard NEP, there is also a separate strategy set $X_\nu \subseteq \mathbb{R}^{n_\nu}$ for each player ν. Then $x^* = (x^{*,1}, \ldots, x^{*,N})^T \in \mathbb{R}^n$ is called a *Nash equilibrium* or a *solution of the NEP* if each block component $x^{*,\nu}$ is a solution of the optimization problem

$$\min_{x^\nu} \ \theta_\nu(x^\nu, x^{*,-\nu}) \quad \text{s.t.} \quad x^\nu \in X_\nu,$$

i.e., x^* is a Nash equilibrium if no player can improve his situation by unilaterally changing his strategy.

On the other hand, in a GNEP, there is a common strategy space $X \subseteq \mathbb{R}^n$ for all players, and a vector $x^* = (x^{*,1}, \ldots, x^{*,N})^T \in \mathbb{R}^n$ is called a *generalized Nash equilibrium*

or a *solution of the GNEP* if each block component $x^{*,\nu}$ is a solution of the optimization problem

$$\min_{x^\nu} \ \theta_\nu(x^\nu, x^{*,-\nu}) \quad \text{s.t.} \quad (x^\nu, x^{*,-\nu}) \in X.$$

Hence, if X is equal to the Cartesian product $X_1 \times \cdots \times X_N$, then the GNEP reduces to a standard NEP, whereas in general the feasible choices of strategies for player ν depend on the strategies chosen by all other players, making the problem much more complicated.

Note that there exist also more general definitions of a GNEP where the feasible set of each player depends on the strategies of all other players (as in the above setting), but is different for each player, see [9, 8] for more details. The majority of papers, however, uses our previous definition of a GNEP.

Throughout this paper, we make the following blanket assumptions.

Assumption 1.1 *(a) The set $X \subseteq \mathbb{R}^n$ is nonempty, closed, and convex.*

(b) The utility functions θ_ν are continuous and, as a function of x^ν alone, convex.

Both assumptions are standard in the context of GNEPs. Nevertheless, we stress that condition (a) is significantly weaker than in many related papers where X, in addition, is assumed to be compact. In particular, this compactness assumption is used (and crucial) in the two papers [26, 19] dealing with the relaxation method.

An important tool for both NEPs and GNEPs is the so-called *Nikaido-Isoda-function*

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

also called the *Ky Fan-function*. In order to overcome some disadvantages of this function, we use a regularized version that was first considered in [13] for standard NEPs and then further investigated for GNEPs in [14, 15], see also [20] for a similar mapping in the context of equilibrium programming problems. To this end, let $\alpha > 0$ be a given parameter that is assumed to be fixed throughout this paper. Then the *regularized Nikaido-Isoda-function* is given by

$$\Psi_\alpha(x, y) := \sum_{\nu=1}^{N} \big[\theta_\nu(x^\nu, x^{-\nu}) - \theta_\nu(y^\nu, x^{-\nu})\big] - \frac{\alpha}{2}\|x - y\|^2. \tag{1}$$

Due to the regularization term, it is easy to see, taking into account Assumption 1.1 (b), that the mapping $\Psi_\alpha(x, \cdot)$ is strongly concave in y (for every fixed x), hence the maximization problem

$$\max_{y} \ \Psi_\alpha(x, y) \quad \text{s.t. } y \in X \tag{2}$$

has a unique solution for each x that we call $y_\alpha(x)$ (note that this statement is not true for the standard Nikaido-Isoda-function), i.e.

$$y_\alpha(x) = \arg\max_{y \in X} \Psi_\alpha(x, y). \tag{3}$$

The corresponding value function is then defined by

$$V_\alpha(x) := \Psi_\alpha\big(x, y_\alpha(x)\big). \tag{4}$$

A vector $x^* \in X$ is then called a *normalized Nash equilibrium* if $V_\alpha(x^*) = 0$ holds. Alternative names are *social equilibrium* or *variational equilibrium*. Note that our definition does not correspond to the original one given in [24], but was shown to be equivalent to one which is nowadays usually called a normalized Nash equilibrium in [14]. It is not difficult to see that every normalized Nash equilibrium is a solution of the GNEP, whereas the converse is not true in general. In fact, it is easy to find examples with infinitely many generalized Nash equilibria which have just a single normalized Nash equilibrium.

The relaxation methods presented in [26, 19] as well as the one to be discussed in this paper find a normalized Nash equilibrium and, therefore, a particular solution of a given GNEP. The relaxation method itself uses the iteration

$$x^{k+1} := x^k + t_k d^k, \quad d^k := y_\alpha(x^k) - x^k, \quad k = 0, 1, 2, \ldots \tag{5}$$

for the particular value $\alpha = 0$ of the parameter α (since this does not guarantee existence and uniqueness of the maximizer $y_\alpha(x)$ in (3), the authors of [26] have to add some assumptions which are not necessary in our case) and a stepsize $t_k \in (0, 1]$ satisfying the conditions

$$t_k \downarrow 0 \quad \text{and} \quad \sum_{k=0}^{\infty} t_k = \infty.$$

These conditions suggest a choice of the form $t_k = \gamma/k$ for some constant $\gamma > 0$, however, in practice this choice leads to very slow convergence, so different heuristics are typically implemented in order to improve the numerical behaviour of the relaxation method, see, e.g., [19, 14]. The version of the relaxation method presented in [19] (again for the case $\alpha = 0$) chooses the stepsize t_k by an exact minimization of the one-dimensional mapping

$$\varphi_k(t) := V_\alpha(x^k + t d^k)$$

over the interval $[0, 1]$. This method was shown to have the same global convergence property as the original relaxation method under the same set of assumptions as in [26], however, since V_α is typically a highly nonlinear function, the computation of t_k by minimizing φ_k is usually not possible. Moreover, its computation is very expensive since each function evaluation of φ_k corresponds to the solution of a constrained optimization problem in order to evaluate the mapping V_α at the intermediate point $x^k + t d^k$.

Note that the iteration (5) of the standard relaxation method (with $\alpha = 0$) can also be applied to the case $\alpha > 0$ considered in this paper, and that the convergence results presented in [26, 19] for each of the above two stepsize rules also hold in this situation under the assumptions stated there. Here, however, we present a completely different convergence analysis motivated by standard descent methods from optimization that uses an inexact Armijo-type line search in order to calculate a suitable stepsize t_k at each iteration k.

The organization of the remaining paper is as follows: Section 2 first recalls some of the basic properties of the regularized Nikaido-Isoda function and then states some new ones. In Section 3, we state our main assumption that is used in this paper and present some sufficient conditions for this assumption to be satisfied. Section 4 then contains a precise statement of our relaxation method and shows that this method is well-defined and globally convergent to a normalized Nash equilibrium under the main assumption introduced in the previous section. This convergence analysis, however, relies on the continuous differentiability of the mappings θ_ν, while the algorithm itself is, in principle, derivative-free. We therefore present an alternative convergence analysis for this algorithm in Section 5 which avoids any smoothness (differentiability) assumption, but which is based on a convexity assumption that is stronger than the one used in Section 4. We then apply our relaxation method to a variety of different examples in Section 6. We conclude with some final remarks and open questions in Section 7.

The notation used in this paper is more or less standard, so we mention only a very few things here: A square matrix A is said to be positive definite if $x^T Ax > 0$ holds for all $x \neq 0$. Note that we do not require symmetry of A here. Furthermore, the symbol $\partial f(x)$ denotes the subdifferential of a convex function f at the given point x, see [16, 23] for its definition and several properties.

2 Properties of the Regularized Nikaido-Isoda-function

Let Ψ_α, y_α, and V_α be defined by (1), (3), and (4), respectively. These mappings have a number of important properties that we summarize in the following result whose proof can be found in [14].

Proposition 2.1 *The following statements hold:*

(a) $V_\alpha(x) \geq 0$ *for all* $x \in X$.

(b) $V_\alpha(x^*) = 0$ *for some* $x^* \in X \iff x^*$ *is a normalized Nash equilibrium of the GNEP.*

(c) x^* *is a fix point of the mapping* $x \mapsto y_\alpha(x) \iff x^*$ *is a normalized Nash equilibrium of the GNEP.*

(d) The mapping $x \mapsto y_\alpha(x)$ *is continuous.*

(e) The function V_α *is continuously differentiable if all* θ_ν *are continuously differentiable, and its gradient is given by* $\nabla V_\alpha(x) = \nabla_x \Psi_\alpha(x, y)\big|_{y=y_\alpha(x)}$.

Note that, for continuously differentiable payoff functions θ_ν, part (e) states that the mapping $V_\alpha(x) = \Psi_\alpha\big(x, y_\alpha(x)\big)$ is continuously differentiable, whereas it does not say anything regarding the smoothness of y_α which, indeed, might be nondifferentiable.

Proposition 2.1 shows that x^* is a normalized Nash equilibrium if and only if it is a global minimum of the constrained minimization problem

$$\min V_\alpha(x) \quad \text{s.t.} \quad x \in X \tag{6}$$

with optimal function value $V_\alpha(x^*) = 0$. The basic idea of our modified relaxation method is to solve this minimization problem by a suitable feasible descent method.

Under certain assumptions, it can be shown that the objective function V_α is (strongly) convex. In view of the definition of V_α, this (strong) convexity depends on similar properties of the regularized mapping $\Psi_\alpha(x, y)$. In order to state a corresponding result, we recall that the function $\Psi_\alpha(\cdot, y)$ (as a function of x alone) is convex on a set $S \subseteq \mathbb{R}^n$ for any given y if the inequality

$$\Psi_\alpha\big(\lambda x + (1-\lambda)z, y\big) \leq \lambda\Psi_\alpha(x, y) + (1-\lambda)\Psi_\alpha(z, y)$$

holds for all $x, z \in S$ and all $\lambda \in (0, 1)$. Moreover, $\Psi_\alpha(\cdot, y)$ (again as a function of x alone) is strongly convex on a set $S \subseteq \mathbb{R}^n$ for any given y if there is a modulus $\mu > 0$ (possibly depending on the particular vector y) such that the inequality

$$\Psi_\alpha\big(\lambda x + (1-\lambda)z, y\big) \leq \lambda\Psi_\alpha(x, y) + (1-\lambda)\Psi_\alpha(z, y) - \mu\lambda(1-\lambda)\|x - z\|^2$$

holds for all $x, z \in S$ and all $\lambda \in (0, 1)$. If the constant $\mu > 0$ can be chosen independently of $y \in S$, then we call $\Psi_\alpha(\cdot, y)$ *uniformly strongly convex* on S. Using this terminology, we have the following result.

Proposition 2.2 *The following statements hold:*

(a) If $\Psi_\alpha(\cdot, y)$ is convex for every $y \in X$, then V_α is also convex on X.

(b) If $\Psi_\alpha(\cdot, y)$ is uniformly strongly convex on X, then V_α is strongly convex on X.

Proof. (a) Exploiting the convexity of $\Psi_\alpha(\cdot, y)$ for any given y, we obtain for every $x, z \in X$ and all $\lambda \in (0, 1)$

$$\begin{aligned} V_\alpha\big(\lambda x + (1-\lambda)z\big) &= \Psi_\alpha\big(\lambda x + (1-\lambda)z, y_\alpha(\lambda x + (1-\lambda)z)\big) \\ &\leq \lambda\Psi_\alpha\big(x, y_\alpha(\lambda x + (1-\lambda)z)\big) + (1-\lambda)\Psi_\alpha\big(z, y_\alpha(\lambda x + (1-\lambda)z)\big) \\ &\leq \lambda\Psi_\alpha\big(x, y_\alpha(x)\big) + (1-\lambda)\Psi_\alpha\big(z, y_\alpha(z)\big) \\ &= \lambda V_\alpha(x) + (1-\lambda)V_\alpha(z), \end{aligned}$$

where the first inequality takes into account that the vector $y_\alpha(\lambda x + (1-\lambda)z)$ belongs to X, whereas the second inequality exploits the definitions of $y_\alpha(x)$ and $y_\alpha(z)$.

(b) Let $\mu > 0$ be the uniform modulus of strong convexity of the mapping $\Psi_\alpha(\cdot, y)$ on the set X. Then, similar to the proof of part (a), we obtain for all $x, z \in X$ and all $\lambda \in (0, 1)$ that

$$\begin{aligned} V_\alpha\big(\lambda x + (1-\lambda)z\big) &= \Psi_\alpha\big(\lambda x + (1-\lambda)z, y_\alpha(\lambda x + (1-\lambda)z)\big) \\ &\leq \lambda\Psi_\alpha\big(x, y_\alpha(\lambda x + (1-\lambda)z)\big) \\ &\quad +(1-\lambda)\Psi_\alpha\big(z, y_\alpha(\lambda x + (1-\lambda)z)\big)\big) - \mu\lambda(1-\lambda)\|x - z\|^2 \end{aligned}$$

$$\begin{aligned} &\leq \lambda\Psi_\alpha\big(x, y_\alpha(x)\big) + (1-\lambda)\Psi_\alpha\big(z, y_\alpha(z)\big) - \mu\lambda(1-\lambda)\|x-z\|^2 \\ &= \lambda V_\alpha(x) + (1-\lambda)V_\alpha(z) - \mu\lambda(1-\lambda)\|x-z\|^2. \end{aligned}$$

Hence V_α is strongly convex on X with modulus $\mu > 0$. □

In order to guarantee the (strong) convexity of V_α, we have to verify the assumptions from Proposition 2.2, namely the (uniform strong) convexity of the mapping $\Psi_\alpha(\cdot, y)$ for all $y \in X$. In general, this requirement is not satisfied under standard convexity assumptions for our payoff functions θ_ν. However, for the case of quadratic payoff functions, we have the following sufficient condition.

Proposition 2.3 *Consider the case where the payoff functions are quadratic, say*

$$\theta_\nu(x) := \frac{1}{2}(x^\nu)^T A_{\nu\nu} x^\nu + \sum_{\substack{\mu=1 \\ \mu\neq\nu}}^{N} (x^\nu)^T A_{\nu\mu} x^\mu \quad \forall \nu = 1, \ldots, N$$

for certain matrices $A_{\nu\mu} \in \mathbb{R}^{n_\nu \times n_\mu}$ such that the diagonal blocks $A_{\nu\nu}$ are (without loss of generality) symmetric. Assume that

$$B := \begin{pmatrix} \frac{1}{2}A_{11} & A_{12} & \cdots & A_{1N} \\ A_{21} & \frac{1}{2}A_{22} & \cdots & A_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \cdots & \frac{1}{2}A_{NN} \end{pmatrix}, \tag{7}$$

is positive definite and let $\lambda_{\min} > 0$ be the smallest eigenvalue of the symmetric matrix $B + B^T$. Then the following statements hold:

(a) The function V_α is convex on $\mathbb{R}^n$ for all $\alpha \in (0, \lambda_{min}]$.

(b) The function V_α is strongly convex on $\mathbb{R}^n$ for all $\alpha \in (0, \lambda_{min})$.

Proof. We show that $\Psi_\alpha(\cdot, y)$ is (uniformly strongly) convex and then apply Proposition 2.2. To this end, first note that the second partial derivatives of Ψ_α with respect to x are given by

$$\nabla^2_{x^\nu x^\mu}\Psi_\alpha(x,y) = \begin{cases} A_{\nu\mu} + A_{\mu\nu}^T, & \text{if } \mu \neq \nu \\ A_{\nu\nu} - \alpha I_{n_\nu}, & \text{if } \mu = \nu. \end{cases} \qquad \forall \nu, \mu = 1, \ldots, N.$$

Hence we have $\nabla^2_{xx}\Psi_\alpha(x,y) = B + B^T - \alpha I$. Consequently, assumption (a) (or (b)) implies that the Hessian $\nabla^2_{xx}\Psi_\alpha(x,y)$ is positive semidefinite (or positive definite). This, in turn, implies that the quadratic function $\Psi_\alpha(\cdot, y)$ itself is convex (or uniformly strongly convex). The statement therefore follows from Proposition 2.2. □

Note that the previous result also holds if the utility functions θ_ν contain additional linear and/or constant terms since they do not change the second-order derivative of Ψ_α used in the proof of that result.

The following example shows that the bounds given in Proposition 2.3 are tight.

Example 2.4 We consider the following Nash equilibrium problem, where player 1 controls the single variable x_1, player 2 controls the single variable x_2, and the corresponding optimization problems are given by

$$\begin{array}{ll|ll} \min_{x_1} & \frac{1}{2}x_1^2 & \min_{x_2} & \frac{1}{2}x_2^2 \\ \text{s.t.} & x_1 \geq 1 & \text{s.t.} & x_2 \geq 1. \end{array}$$

Actually, this is a special case with two separable optimization problems. The unique solution is $x^* = (1,1)^T$, and the matrix $B+B^T$ from Proposition 2.3 has the two eigenvalues $\lambda_1 = \lambda_2 = 1$, hence we have $\lambda_{\min} = 1$.

Given an arbitrary $\alpha > 0$, an elementary calculation shows that the component functions of y_α are given by

$$[y_\alpha(x)]_i = \begin{cases} \frac{\alpha}{1+\alpha}x_i, & \text{if } x_i \geq \frac{1+\alpha}{\alpha}, \\ 1, & \text{else.} \end{cases}$$

Therefore, for all x satisfying $x_i < \frac{1+\alpha}{\alpha}$, we locally have $y_\alpha(x) \equiv \begin{pmatrix} 1 \\ 1 \end{pmatrix}$. Consequently, the Hessian of V_α is this area is given by

$$\nabla^2 V_\alpha(x) = \begin{pmatrix} (1-\alpha) & 0 \\ 0 & (1-\alpha) \end{pmatrix},$$

which implies that V_α is convex in the respective area for all $0 < \alpha \leq 1$ and nonconvex for all $\alpha > 1$. $\Diamond$

The previous results guarantee that (6) is a convex optimization problem, in particular, every stationary point is therefore a global minimum and hence a normalized Nash equilibrium of the GNEP (provided there is at least one such solution of the GNEP). In the next section, we will introduce an assumption which does not necessarily guarantee convexity of the value function V_α, but still implies (among other things) that a stationary point is a global minimum of (6).

3 Main Assumption and Discussion

This section is devoted to a discussion of the main assumption that is used in our subsequent analysis. This assumption was first used in [14] and is as follows:

Assumption 3.1 *(a) The utility functions θ_ν are continuously differentiable.*

(b) For given $x \in X$ with $x \neq y_\alpha(x)$, the inequality

$$\sum_{\nu=1}^{N} \big[\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x), x^{-\nu})\big]^T \big(x - y_\alpha(x)\big) > 0$$

holds.

Note that the smoothness assumption from Assumption 3.1 (a) is necessary, in particular, to formulate part (b). This Assumption 3.1 (b) is crucial for the development and analysis of our relaxation method. On the one hand, it guarantees that the search direction used in the relaxation method is a (feasible) descent direction for the value function V_α, see Lemma 4.2 below (to this end, note that the original relaxation method from [26] does not generate a monotonically decreasing sequence of function values), and, on the other hand, it can be shown that any stationary point of the optimization problem (6) is already a solution of the GNEP under Assumption 3.1. This observation is highly important, and a proof can be found in [14], but it will not be used explicitly within this paper.

The rest of this section is devoted to a discussion of Assumption 3.1 (b). While it looks somewhat strange in the beginning, we will show that it is satisfied under some conditions which are much easier to verify. Further note that these conditions guarantee that Assumption 3.1 holds for an arbitrary $\alpha > 0$. The main criterion is given in the following result.

Theorem 3.2 *Let x^* be a solution GNEP and assume that the utility functions θ_ν are twice continuously differentiable. Suppose that the matrix $A = (A_{\nu\mu})_{\nu,\mu=1}^N$ with $A_{\nu\mu} = \nabla^2_{x^\nu x^\mu}\theta_\nu(x^*)$ is positive definite. Then there is a neighbourhood $N(x^*)$ such that Assumption 3.1 holds for all $x \in U(x^*)$.*

Proof. Given any x, we simplify the notation and write y and y^ν instead of $y_\alpha(x)$ and $y_\alpha^\nu(x)$, respectively. From the intregral mean value theorem it follows that

$$\nabla\theta_\nu(y^\nu, x^{-\nu}) - \nabla\theta_\nu(x^\nu, x^{-\nu}) = \Big(\int_0^1 \nabla^2_{xx^\nu}\theta_\nu\big(x^\nu + \tau(y^\nu - x^\nu), x^{-\nu}\big)\, d\tau\Big)(y^\nu - x^\nu).$$

Hence we get

$$\begin{aligned}
&\sum_{\nu=1}^{N} \big[\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y^\nu, x^{-\nu})\big] \\
&= \sum_{\nu=1}^{N} \Big[\Big(\int_0^1 \nabla^2_{xx^\nu}\theta_\nu\big(x^\nu + \tau(y^\nu - x^\nu), x^{-\nu}\big)\, d\tau\Big)(x^\nu - y^\nu)\Big] \qquad (8)\\
&= \Big(\int_0^1 \nabla^2_{xx^1}\theta_1\big(x^1 + \tau(y^1 - x^1), x^{-1}\big)\, d\tau, \ldots, \int_0^1 \nabla^2_{xx^N}\theta_N\big(x^N + \tau(y^N - x^N), x^{-N}\big)\, d\tau\Big)(x - y) \\
&= \Big(\int_0^1 \Big[\nabla^2_{xx^1}\theta_1\big(x^1 + \tau(y^1 - x^1), x^{-1}\big), \ldots, \nabla^2_{xx^N}\theta_N\big(x^N + \tau(y^N - x^N), x^{-N}\big)\Big] d\tau\Big)\,(x - y)
\end{aligned}$$

$$= \int_0^1 \Big[\nabla^2_{xx^1}\theta_1\big(x^1+\tau(y^1-x^1),x^{-1}\big),\ldots,\nabla^2_{xx^N}\theta_N\big(x^N+\tau(y^N-x^N),x^{-N}\big)\Big]\big(x-y\big)d\tau.$$

Since the functions θ_ν are twice continuously differentiable, and since x^* is a fix point of $y_\alpha(\cdot)$ in view of Proposition 2.1, the assumption that A is positive definite implies that there exists a neighbourhood $N(x^*)$ such that the slightly perturbed matrix

$$\Big(\nabla^2_{xx^1}\theta_1\big(x^1+\tau(y^1_\alpha(x)-x^1),x^{-1}\big),\ldots,\nabla^2_{xx^N}\theta_N\big(x^N+\tau(y^N_\alpha(x)-x^N),x^{-N}\big)\Big)$$

is positive definite for all $x\in N(x^*)$ and $\tau\in[0,1]$. Together with (8) this implies that Assumption 3.1 holds for all $x\in N(x^*)$ with $x\neq y_\alpha(x)$. □

The following two corollaries are consequences of Theorem 3.2 and provide some simplified sufficient conditions for Assumption 3.1 to be satisfied.

Corollary 3.3 *Consider the case where the payoff functions θ_ν are quadratic, say*

$$\theta_\nu(x)=\frac{1}{2}(x^\nu)^TA_{\nu\nu}x^\nu+\sum_{\substack{\mu=1\\ \mu\neq\nu}}^N(x^\nu)^TA_{\nu\mu}x^\mu$$

for $\nu=1,\ldots,N$. Suppose that the matrix $A=(A_{\nu\mu})_{\nu,\mu=1}^N$ is positive definite. Then Assumption 3.1 is satisfied at an arbitrary point $x\in X$ with $x\neq y_\alpha(x)$.

Proof. The statement follows immediately from Theorem 3.2 by noting that the second-order partial derivatives of our quadratic functions θ_ν are given by $\nabla^2_{x^\nu x^\mu}\theta_\nu(x)=A_{\nu\mu}$ for all $x\in\mathbb{R}^n$. □

Note that the assumption of the matrix $A=(A_{\nu\mu})$ being positive definite is weaker than the corresponding condition on the matrix B defined in (7). In fact, B being positive definite implies that the diagonal block matrix $D:=\frac{1}{2}\mathrm{diag}\big(A_{11},\ldots,A_{NN}\big)$ is also positive definite, which, in turn, gives the positive definiteness of A since this matrix is simply the sum of B and D.

Corollary 3.4 *Suppose that the utility functions θ_ν are twice continuously differentiable and that the matrix $B(x,y)=\big(B_{\mu\nu}(x,y)\big)_{\mu,\nu=1}^N$ with*

$$B_{\mu\nu}(x,y)=\nabla^2_{x^\mu x^\nu}\theta_\nu(y^\nu,x^{-\nu}) \tag{9}$$

is positive definite for all $x,y\in X$ or equivalently, that the matrices

$$B(x,y)=-\nabla^2_{xy}\Psi_\alpha(x,y)-\nabla^2_{yy}\Psi_\alpha(x,y) \tag{10}$$

are positive definite for all $x,y\in X$. Then Assumption 3.1 holds for all $x\in X$ with $x\neq y_\alpha(x)$.

Proof. By taking a look at the proof of Theorem 3.2, we immediately see that the assumed positive definiteness of the matrices $B(x,y)$ with the block components given by (9) implies that Assumption 3.1 holds.

Hence we only have to show that the mapping B has the alternative representation given in (10). This, however, follows directly from the expression of the second-order derivatives $\nabla^2_{xy}\Psi_\alpha(x,y)$ and $\nabla^2_{yy}\Psi_\alpha(x,y)$, see, e.g., [15]. □

The following example shows that the condition given in (9) is not sufficient for the convexity of the function V_α. In particular, it follows that Assumption 3.1 guarantees that stationary points are global minima for a class of nonconvex problems.

Example 3.5 Consider a two-person game where each player controls a single variable, and where the corresponding optimization problems are given by

$$\begin{array}{ll|ll} \min_{x_1} & \frac{1}{2}x_1^2 + \frac{3}{4}x_1x_2 & \min_{x_2} & \frac{1}{2}x_2^2 + \frac{3}{4}x_1x_2 \\ \text{s.t.} & x_1 \geq 1 & \text{s.t.} & x_2 \geq 1. \end{array}$$

The unique Nash equilibrium is $x^* = (1,1)^T$. Elementary calculations show that, for all $x \in X := [1,\infty) \times [1,\infty)$ sufficiently close to x^*, we have $y_\alpha(x) \equiv \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ and, therefore,

$$\nabla^2 V_\alpha(x) = \begin{pmatrix} 1-\alpha & \frac{3}{2} \\ \frac{3}{2} & 1-\alpha \end{pmatrix}.$$

Obviously, there is no $\alpha > 0$ such that this matrix is positive semidefinite. In particular, the function V_α is not convex on X. Nevertheless, the matrix $B(x,y)$ from (9) is equal to

$$B(x,y) = \begin{pmatrix} 1 & \frac{3}{4} \\ \frac{3}{4} & 1 \end{pmatrix}$$

and therefore positive definite for all $\alpha \in (0,\infty)$ and all $x, y \in X$, which implies that Assumption 3.1 holds for all $x \in X$. ◊

4 Algorithm and Convergence

Here we present our algorithm and prove a global convergence result. To this end, we suppose throughout this section that Assumption 3.1 holds at every point $x \in X$ or at least at every iterate $x^k \in X$ that is generated by the following algorithm.

Algorithm 4.1 *(Relaxation method with inexact line search)*

(S.0) Choose $x^0 \in X, \beta, \sigma \in (0,1)$, and set $k := 0$.

(S.1) Check a suitable termination criterion $\big($like $V_\alpha(x^k) \leq \varepsilon$ for some $\varepsilon > 0\big)$.

(S.2) Compute $y_\alpha(x^k)$ and set $d^k := y_\alpha(x^k) - x^k$.

(S.3) Compute $t_k = \max\{\beta^l \mid l = 0, 1, 2, \ldots\}$ *such that*

$$V_\alpha(x^k + t_k d^k) \leq V_\alpha(x^k) - \sigma t_k^2 \|d^k\|. \tag{11}$$

(S.4) Set $x^{k+1} := x^k + t_k d^k, k \longleftarrow k+1$*, and go to (S.1).*

Recall that we assume continuous differentiability of all utility functions θ_ν, cf. Assumption 3.1. This assumption is crucial for the subsequent convergence analysis presented in this section. Nevertheless, we would like to point out that, at least in principle, Algorithm 4.1 is a derivative-free method. In practice, the situation is somewhat different since we have to be able to compute the function values of V_α which corresponds to the solution of a constrained optimization problem, and this is typically done by suitable methods that exploit the differentiability of the utility functions θ_ν. While this section is therefore devoted to a convergence analysis using derivatives, we present a completely derivative-free analysis in the next section which, however, is based on a convexity-type assumption which is stronger than the central Assumption 3.1 used within this section.

Our first aim is to show that Algorithm 4.1 is well-defined. To this end, we first note that d^k is always a direction of descent for the merit function V_α.

Lemma 4.2 *Let $x^k \in X$ be the current iterate and d^k be the vector computed in Step (S.2) of Algorithm 4.1. Then $\nabla V_\alpha(x^k)^T d^k < 0$, i.e. d^k is a direction of descent at x^k (as long as x^k is not a normalized Nash equilibrium of the GNEP).*

Proof. For simplicity of notation, we write y_α instead of $y_\alpha(x)$ and omit the iteration index k. Recall from Proposition 2.1 that $\nabla V_\alpha(x) = \nabla_x \Psi_\alpha(x, y)\big|_{y=y_\alpha(x)}$. Calculating the partial derivative of Ψ_α with respect to x (cf. [14]), we then obtain

$$\begin{aligned}
\nabla V_\alpha(x)^T d &= \Big(\sum_{\nu=1}^N \big[\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y_\alpha^\nu, x^{-\nu})\big] + \ldots \\
&\qquad \begin{pmatrix} \nabla_{x^1}\theta_1(y_\alpha^1, x^{-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y_\alpha^N, x^{-N}) \end{pmatrix} - \alpha(x - y_\alpha)\Big)^T (y_\alpha - x) \\
&= \Big(\sum_{\nu=1}^N \big[\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y_\alpha^\nu, x^{-\nu})\big]\Big)^T (y_\alpha - x) \\
&\qquad + \Big(\begin{pmatrix} \nabla_{x^1}\theta_1(y_\alpha^1, x^{-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y_\alpha^N, x^{-N}) \end{pmatrix} - \alpha(x - y_\alpha)\Big)^T (y_\alpha - x).
\end{aligned}$$

The first term of this equality is negative by Assumption 3.1, while the second term is nonpositive due to the first order optimality condition for $y_\alpha(x) := \arg\max_{y \in X} \Psi_\alpha(x, y)$. Altogether, we conclude that $\nabla V_\alpha(x)^T d < 0$, hence d is a descent direction. □

Note that Assumption 3.1 was crucial in proving the descent property. – Based on the previous result, we are now in the position to show that Algorithm 4.1 is well–defined.

Lemma 4.3 *Algorithm 4.1 is well–defined and generates a sequence $\{x^k\}$ belonging to the feasible set X.*

Proof. The fact that $\{x^k\}$ belongs to X follows by induction: We have $x^0 \in X$ by our choice of the starting point. Moreover, if $x^k \in X$, we also have

$$x^{k+1} = x^k + t_k d^k = (1 - t_k)x^k + t_k y_\alpha(x^k) \in X$$

since $x^k, y_\alpha(x^k) \in X$, $t_k \in (0, 1]$ and X is convex by assumption. In order to show that Algorithm 4.1 is well–defined, we only need to verify that the inner loop in (S.3) is finite at each iteration k. To this end, let the iteration number k be fixed, and assume that the calculation of t_k is an infinite loop. Then we have

$$V_\alpha(x^k + \beta^l d^k) > V_\alpha(x^k) - \sigma\beta^{2l}\|d^k\| \quad \forall\, l \in \mathbb{N}$$

or, equivalently,

$$\frac{V_\alpha(x^k + \beta^l d^k) - V_\alpha(x^k)}{\beta^l} > -\sigma\beta^l\|d^k\| \quad \forall\, l \in \mathbb{N}.$$

Taking the limit $l \longrightarrow +\infty$ and using the fact that V_α is continuously differentiable, we obtain $\nabla V_\alpha(x^k)^T d^k \geq 0$. On the other hand, we know from Lemma 4.2 that $\nabla V_\alpha(x^k)^T d^k < 0$ since x^k is not a solution of our GNEP $\big($otherwise the algorithm would have stopped in (S.1)$\big)$. This contradiction completes the proof. □

We next give a global convergence result for Algorithm 4.1.

Theorem 4.4 *Every accumulation point of a sequence generated by Algorithm 4.1 is a normalized Nash equilibrium of our GNEP.*

Proof. Let x^* be such an accumulation point, and let $\{x^k\}_K$ be a corresponding subsequence converging to x^*. The continuity of the solution operator $x \longmapsto y_\alpha(x)$ then implies $\{y_\alpha(x^k)\}_K \longrightarrow y_\alpha(x^*)$. Hence we have $\{d^k\}_K \longrightarrow y_\alpha(x^*) - x^* =: d^*$. In view of Proposition 2.1, we only need to show that $d^* = 0$.

Assume we have $d^* \neq 0$. Since the entire sequence $\{V_\alpha(x^k)\}$ is monotonically decreasing (by construction) and bounded from below $\big($e.g., by $V_\alpha(x^*)\big)$, it follows that the entire sequence $\{V_\alpha(x^k)\}$ converges. From our line search rule, we therefore get

$$0 \longleftarrow V_\alpha(x^{k+1}) - V_\alpha(x^k) \leq -\sigma t_k^2\|d^k\| \leq 0 \quad \forall\, k \in \mathbb{N}.$$

This implies

$$\lim_{k \longrightarrow \infty} t_k^2\|d^k\| = 0.$$

Since $d^* \neq 0$ by assumption, we therefore have

$$\lim_{k \in K} t_k = 0. \tag{12}$$

Let $l_k \in \mathbb{N}$ be the unique exponent such that $t_k = \beta^{l_k}$ in (S.3) of Algorithm 4.1. In view of (12), we can assume without loss of generality that $t_k < 1$ for all $k \in K$, hence the stepsize $\frac{t_k}{\beta} = \beta^{l_k - 1}$ does not satisfy the inequality from (S.3) of Algorithm 4.1. Hence we have

$$V_\alpha(x^k + \beta^{l_k - 1} d^k) > V_\alpha(x^k) - \sigma(\beta^{l_k - 1})^2 \|d^k\| \quad \forall\, k \in K.$$

This can be written as

$$\frac{V_\alpha(x^k + \beta^{l_k - 1} d^k) - V_\alpha(x^k)}{\beta^{l_k - 1}} > -\sigma \beta^{l_k - 1} \|d^k\| \quad \forall\, k \in K.$$

Taking the limit $k \longrightarrow \infty$ on K, using the fact that $\beta^{l_k - 1} \longrightarrow 0$ and exploiting the continuous differentiability of V_α, we therefore obtain from the mean value theorem that

$$\nabla V_\alpha(x^*)^T d^* \geq 0.$$

On the other hand, since $d^* = y_\alpha(x^*) - x^* \neq 0$, it follows from Lemma 4.2 that $\nabla V_\alpha(x^*)^T d^* < 0$. This contradiction shows that $d^* = 0$ and, therefore, x^* is indeed a normalized Nash equilibrium of our GNEP. □

The previous convergence result also holds for a minor modification of Algorithm 4.1. This observation is formally stated in the following remark.

Remark 4.5 It is not difficult to see that all our previous results remain true if we replace the line search rule (11) in Algorithm 4.1 by the slightly modified condition

$$V_\alpha(x^k + t_k d^k) \leq V_\alpha(x^k) - \sigma t_k^2 \|d^k\|^2$$

where the only difference to the original condition (11) is that we now take the square of $\|d^k\|$ rather than $\|d^k\|$ itself.

We close this section with a simple example discussing the rate of convergence of Algorithm 4.1. In fact, motivated by a somewhat similar idea for the solution of variational inequalities in [25], one might expect local quadratic convergence for the iteration $x^{k+1} := y_\alpha(x^k)$ corresponding to the stepsize $t_k = 1$ in Algorithm 4.1. However, it turns out that this is not true even under very favourable assumptions. This is illustrated by the following simple example.

Example 4.6 Consider the GNEP (which is actually an unconstrained NEP) with two players, where each player controls only a single variable and where the corresponding optimization problems are given by

$$\begin{array}{ll|ll} \min_{x_1} & \frac{1}{2} x_1^2 & \min_{x_2} & \frac{1}{2} x_2^2 \\ \text{s.t.} & (x_1, x_2) \in \mathbb{R}^2 & \text{s.t.} & (x_1, x_2) \in \mathbb{R}^2. \end{array}$$

The solution of this GNEP is obviously the origin $x^* = (0,0)^T$. Given any $x \in \mathbb{R}^2$, an easy calculation shows that the maximizer $y_\alpha(x)$ of the corresponding optimization problem (2) is given by

$$y_\alpha(x) = \frac{\alpha}{1+\alpha} x.$$

Consequently, for the stepsize $t_k = 1$ in our relaxation method, we obtain

$$x^{k+1} = x^k + t_k d^k = y_\alpha(x^k) = \frac{\alpha}{1+\alpha} x^k.$$

Clearly, this shows that the rate of convergence is neither superlinear nor quadratic although the example is very simple and has very nice properties. On the other hand, it shows that we have a fast linear rate of convergence for small $\alpha > 0$. $\Diamond$

5 Derivative-free Analysis

In this section, we consider Algorithm 4.1 once again. To this end, recall that the method does not use any derivative information. The previous analysis, however, assumes differentiability of all functions θ_ν. Here we present a completely derivative-free analysis using, however, the following slightly stronger assumption that we assume to hold throughout this section.

Assumption 5.1 *The function $\Psi_\alpha(\cdot, y)$ is convex for every y taken from an open convex neighbourhood of the set X.*

In view of Proposition 2.2 and its proof, a direct consequence of Assumption 5.1 is the convexity of the mapping V_α on the open convex neighbourhood of X. In particular, the function V_α is therefore both directionally differentiable and locally Lipschitzian on this set. These observations will be exploited in our subsequent analysis.

We begin our analysis with the following counterpart of Lemma 4.2.

Lemma 5.2 *Let $x \in X$ be any given point, and let $d := y_\alpha(x) - x$. Then there is a constant $\bar{t} > 0$ (depending on x) such that $V_\alpha(x + td) < V_\alpha(x)$ for all $t \in (0, \bar{t}]$ (provided that x is not a normalized Nash equilibrium of the GNEP).*

Proof. For arbitrary $t \in (0,1)$, the convexity of $\Psi_\alpha(\cdot, y)$ implies

$$\begin{aligned}
V_\alpha(x+td) &= \Psi_\alpha\big(x+td, y_\alpha(x+td)\big) \\
&= \Psi_\alpha\big(x+t(y_\alpha(x)-x), y_\alpha(x+td)\big) \\
&= \Psi_\alpha\big(ty_\alpha(x)+(1-t)x, y_\alpha(x+td)\big) \\
&\le t\Psi_\alpha\big(y_\alpha(x), y_\alpha(x+td)\big) + (1-t)\Psi_\alpha\big(x, y_\alpha(x+td)\big) \qquad (13) \\
&\le t\Psi_\alpha\big(y_\alpha(x), y_\alpha(x+td)\big) + (1-t)\Psi_\alpha\big(x, y_\alpha(x)\big) \\
&= t\Psi_\alpha\big(y_\alpha(x), y_\alpha(x+td)\big) + (1-t)V_\alpha(x)
\end{aligned}$$

$$= \quad t\big[\Psi_\alpha\big(y_\alpha(x), y_\alpha(x+td)\big) - V_\alpha(x)\big] + V_\alpha(x)$$

or, equivalently,

$$\frac{V_\alpha(x+td) - V_\alpha(x)}{t} \leq \Psi_\alpha\big(y_\alpha(x), y_\alpha(x+td)\big) - V_\alpha(x). \tag{14}$$

Since the function y_α is continuous by Proposition 2.1, we have $y_\alpha(x+td) \to y_\alpha(x)$ for $t \to 0$ and, therefore, $\Psi_\alpha\big(y_\alpha(x), y_\alpha(x+td)\big) \to \Psi_\alpha\big(y_\alpha(x), y_\alpha(x)\big) = 0$. Hence it follows from (14) that there is an $\varepsilon = \varepsilon(x) > 0$ (e.g., $\varepsilon := \frac{1}{2}V_\alpha(x)$) and a $\bar{t} = \bar{t}(x) > 0$ such that

$$\frac{V_\alpha(x+td) - V_\alpha(x)}{t} \leq -\varepsilon \quad \forall t \in (0, \bar{t}]. \tag{15}$$

This completes the proof. □

We next show that Algorithm 5.1 is well-defined under Assumption 5.1.

Lemma 5.3 *Algorithm 4.1 is well–defined and generates a sequence $\{x^k\}$ belonging to the feasible set X.*

Proof. Similar to the proof of Lemma 5.3, we only have to show that the stepsize selection in (S.3) is a finite procedure at each iteration k. To this end, we fix the iteration counter k and assume that the calculation of t_k is an infinite loop. Then

$$\frac{V_\alpha(x^k + \beta^l d^k) - V_\alpha(x^k)}{\beta^l} > -\sigma\beta^l\|d^k\| \quad \forall\, l \in \mathbb{N}.$$

Taking the limit $l \longrightarrow +\infty$ and using the fact that V_α is convex and, therefore, directionally differentiable at the current iterate $x^k \in X$, we get

$$V'_\alpha(x^k; d^k) \geq 0. \tag{16}$$

On the other hand, we immediately obtain from (15) that $V'_\alpha(x^k; d^k) \leq -\varepsilon$ for some sufficiently small $\varepsilon = \varepsilon(x^k) > 0$, a contradiction to (16). □

We now come to the main global convergence result of Algorithm 4.1 under Assumption 5.1.

Theorem 5.4 *Every accumulation point of a sequence generated by Algorithm 4.1 is a normalized Nash equilibrium of our GNEP.*

Proof. We try to copy the proof of Theorem 4.4. Basically, this is possible since V_α is a convex function, hence we can exploit suitable properties of the convex subdifferential, see, e.g., [16, 23] for more details.

Let x^* be an accumulation point, and let $\{x^k\}_K$ be a corresponding subsequence converging to x^*. The continuity of the solution operator $x \longmapsto y_\alpha(x)$ (cf. Proposition 2.1) then implies $\{y_\alpha(x^k)\}_K \longrightarrow y_\alpha(x^*)$. Hence we have $\{d^k\}_K \longrightarrow y_\alpha(x^*) - x^* =: d^*$. In view of Proposition 2.1, we only need to show that $d^* = 0$.

Assume that $d^* \neq 0$. Similar to the proof of Theorem 4.4, we know that the entire sequence $\{V_\alpha(x^k)\}$ converges and, since $d^* \neq 0$, that $\lim_{k\in K} t_k = 0$. Let us write $t_k = \beta^{l_k}$ for some exponent $l_k \in \mathbb{N}$. Then the line search rule is not satisfied for β^{l_k-1} for all $k \in K$ (sufficiently large), giving

$$\frac{V_\alpha(x^k + \beta^{l_k-1}d^k) - V_\alpha(x^k)}{\beta^{l_k-1}} > -\sigma\beta^{l_k-1}\|d^k\| \quad \forall\, k \in K. \tag{17}$$

Taking the limit $k \longrightarrow \infty$ on K, the right-hand side converges to zero. In order to get the limit of the left-hand side, we first note that the mean value theorem for convex functions shows that, for each $k \in K$, there is a vector ξ^k on the line segment between x^k and $x^k + \beta^{l_k-1}d^k$ and an element $g^k \in \partial V_\alpha(\xi^k)$ such that

$$V_\alpha(x^k + \beta^{l_k-1}d^k) - V_\alpha(x^k) = \beta^{l_k-1}(g^k)^T d^k.$$

Hence the left-hand side of (17) simply becomes

$$\frac{V_\alpha(x^k + \beta^{l_k-1}d^k) - V_\alpha(x^k)}{\beta^{l_k-1}} = (g^k)^T d^k$$

Now, on the subset $K \subseteq \mathbb{N}$, we have $x^k \to x^*$, $\beta^{l_k-1} \to 0$, and $d^k \to d^* = y_\alpha(x^*) - x^*$. This implies $x^k + \beta^{l_k-1}d^k \to x^*$ and, therefore, also $\xi^k \to x^*$. Since the mapping $x \mapsto \partial V_\alpha(x)$ is locally bounded, the sequence $\{g^k\}_K$ is bounded. Without loss of generality, we can therefore assume that the entire subsequence $\{g^k\}_K$ converges to some vector g^*. Taking into account that the mapping $x \mapsto \partial V_\alpha(x)$ is also closed, it follows that $g^* \in \partial V_\alpha(x^*)$. Exploiting the fact that the directional derivative is the support function of the convex subdifferential, we obtain from (17) that

$$\frac{V_\alpha(x^k + \beta^{l_k-1}d^k) - V_\alpha(x^k)}{\beta^{l_k-1}} = (g^k)^T d^k \to (g^*)^T d^* \leq \max_{g\in\partial V_\alpha(x^*)} g^T d^* = V'_\alpha(x^*; d^*).$$

In view of (17), we have $(g^*)^T d^* \geq 0$, in particular, it therefore follows that $V'_\alpha(x^*; d^*) \geq 0$. On the other hand, since $d^* \neq 0$, it follows from (15) that $V'_\alpha(x^*; d^*) < 0$. This contradiction shows that $d^* = 0$ and therefore completes the proof. □

6 Numerical Results

We implemented Algorithm 4.1 with the modified stepsize rule from Remark 4.5. The method is terminated whenever $V_\alpha(x^k) \leq \varepsilon$ with $\varepsilon := 10^{-12}$ and uses the parameters

$\alpha = 10^{-4}, \beta = 0.5, \sigma = 10^{-4}$. The implementation is done in MATLAB, using the build-in function `SNOPT` from the TOMLAB package in order to calculate $y_\alpha(x^k)$ at each iteration k.

Example 6.1 This test problem is the river basin pollution game taken from [19]. The corresponding numerical results are given in Table 1. ◊

k	x_1^k	x_2^k	x_3^k	$V_\alpha(x^k)$	stepsize
0	0.000000	0.000000	0.000000	90.878301693511	0.000
1	19.325863	17.174698	3.811533	0.118402581670	1.000
2	20.704303	16.105378	3.049526	0.003663469196	1.000
3	21.036699	16.036757	2.808432	0.000213429907	1.000
4	21.118197	16.029540	2.746408	0.000012918766	1.000
5	21.138222	16.028243	2.731024	0.000000789309	1.000
6	21.143173	16.027948	2.727213	0.000000047954	1.000
7	21.144471	16.027877	2.726212	0.000000001927	1.000
8	21.144714	16.027858	2.726025	0.000000000000	1.000

Table 1: Numerical results for Example 6.1

Example 6.2 This test problem is an internet switching model introduced by Kesselman et al. [17] and also analysed by Facchinei et al. [8]. We modify this example slightly and add the additional constraint $x^\nu \geq 0.01, \nu = 1, \ldots, N$ in order to avoid possible domain violations of the utility functions (which contain logarithmic terms). This does not alter the solution. We set $N = 10$ (and $B = 1$ in the description of the model in [8]) and use the starting point $x^0 = (0.1, \ldots, 0.1)^T$. The exact solution of this problem is $x^* = (0.9, \ldots, 0.9)^T$. We only state the first two components of the iteration vector in Table 2. This is the only example where the full stepsize $t_k = 1$ is never accepted. A straightforward computation yields that in the case of two players, Assumption 3.1 does not hold whenever $x_1 = x_2$. Because of the symmetry of the game, this is probably also true for the case of $N > 2$ players. Setting $t_k = 1$ by force yields an iteration sequence which, more or less, alternates between two vectors, cf. Table 3. Using our line search globalization, however, we observe very fast linear convergence (see Table 2).

The situation changes if the starting point for the players is chosen unequally. Taking $x^0 = (0.1, 0.11, 0.12, 0.13 \ldots)^T \in \mathbb{R}^{10}$, for example, we obtain the results from Table 4. Here $V_\alpha(x^k)$ still decreases linearly, but only with a moderate rate. ◊

Example 6.3 Here we consider a simple two-player game originally suggested by Rosen [24]. The solution violates strict complementarity. More precisely, our example has the two pay-off functions

$$\theta_1(x_1, x_2) = \frac{1}{2}x_1^2 - x_1 x_2 \quad \text{and} \quad \theta_2(x_1, x_2) = x_2^2 + x_1 x_2$$

k	x_1^k	x_2^k	$V_\alpha(x^k)$	stepsize
0	0.100000	0.100000	0.026332722333	0.000
1	0.087172	0.087172	0.002241194298	0.250
2	0.090379	0.090379	0.000039775125	0.250
3	0.089905	0.089905	0.000002517609	0.250
4	0.090024	0.090024	0.000000156756	0.250
5	0.089994	0.089994	0.000000010751	0.250
6	0.090002	0.090002	0.000000000671	0.250
7	0.090000	0.090000	0.000000000000	0.250

Table 2: Numerical results for Example 6.2 using $x^0 = (0.1, 0.1, \ldots, 0.1)^T$

k	x_1^k	x_2^k	$V_\alpha(x^k)$	stepsize
0	0.100000	0.100000	0.026332722333	0.000
1	0.048687	0.048687	0.344972828471	1.000
2	0.100000	0.100000	0.026332722333	1.000
3	0.048687	0.048687	0.344972828471	1.000
4	0.100000	0.100000	0.026332722333	1.000
5	0.048687	0.048687	0.344972828471	1.000
6	0.100000	0.100000	0.026332722333	1.000
7	0.048687	0.048687	0.344972828471	1.000

Table 3: Numerical results for Example 6.2 using $x^0 = (0.1, 0.1, \ldots, 0.1)^T$ and forcing $t_k \equiv 1$

and the joint constraints given by

$$X := \{x \in \mathbb{R}^2 \mid x_1 \geq 0, x_2 \geq 0, x_1 + x_2 \geq 1\}.$$

The unique normalized Nash equilibrium of this GNEP is $x^* = (1, 0)^T$ and does not satisfy strict complementarity since an easy calculation shows that $y_\gamma(x^*) = (1, 0)^T$ and $\lambda_\gamma(x^*) = (0, 0, 1)^T$, hence strict complementarity does not hold in the second component. Table 5 shows our corresponding numerical results. $\Diamond$

Example 6.4 This test problem is the Cournot oligopoly problem with shared constraints and nonlinear payoff functions as described in Outrata et al. [22, p. 233]. Our results (using different values for the parameter P from [22]) are given in Table 6. $\Diamond$

7 Final Remarks

We presented a slightly modified relaxation method for finding solutions of generalized Nash equilibrium problems. The basic difference to the standard relaxation method is that we replace the Nikaido-Isoda-function by its regularized counterpart (thus avoiding the

k	x_1^k	x_2^k	x_3^k	$V_\alpha(x^k)$	stepsize
0	0.100000	0.110000	0.120000	0.426072413930	0.000
1	0.055000	0.060000	0.065000	0.029845357936	0.500
2	0.090304	0.092485	0.094657	0.026443474030	1.000
3	0.078749	0.080644	0.082531	0.002346586690	0.250
4	0.083141	0.084771	0.086393	0.000111209217	0.250
5	0.083670	0.085074	0.086472	0.000056181123	0.250
6	0.087607	0.088233	0.088856	0.000050545959	1.000
7	0.087515	0.088055	0.088592	0.000010394061	0.250
8	0.088418	0.088807	0.089195	0.000009758951	0.500
9	0.088478	0.088814	0.089148	0.000003392002	0.250
10	0.088981	0.089223	0.089465	0.000002383523	0.500
11	0.089062	0.089271	0.089479	0.000001223374	0.250
12	0.089643	0.089736	0.089829	0.000001039496	1.000
13	0.089633	0.089712	0.089792	0.000000222366	0.250
14	0.089765	0.089822	0.089880	0.000000203240	0.500
15	0.089775	0.089825	0.089874	0.000000073548	0.250
16	0.089849	0.089885	0.089920	0.000000050643	0.500
17	0.089861	0.089892	0.089923	0.000000026695	0.250
18	0.089947	0.089960	0.089974	0.000000021076	1.000
19	0.089946	0.089958	0.089969	0.000000004764	0.250
20	0.089965	0.089974	0.089982	0.000000004212	0.500
21	0.089967	0.089974	0.089981	0.000000001596	0.250
22	0.089978	0.089983	0.089988	0.000000001075	0.500
23	0.089979	0.089984	0.089989	0.000000000582	0.250
24	0.089992	0.089994	0.089996	0.000000000356	1.000
25	0.089992	0.089994	0.089995	0.000000000101	0.250
26	0.089994	0.089996	0.089997	0.000000000000	0.500

Table 4: Numerical results for Example 6.2 using $x^0 = (0.1, 0.11, \ldots, 0.19)^T$

assumptions that X is compact and the maximizer $y_\alpha(x)$ is unique from [26, 19]) and that we replace the line search rules from [26, 19] by an Armijo-type condition. The numerical results indicate that the method has a relatively fast linear rate of convergence. A proof of this statement is missing and part of our future research. Closely related to this project is the question under which conditions the mapping V_α (or its square root) provides a local or global error bound for GNEPs. So far, not much is known about error bounds for GNEPs (in the moment, we are only aware of the preliminary error bound result from [8]), but this topic certainly deserves further investigations.

References

[1] E. Adida and G. Perakis: *Dynamic pricing and inventory control: Uncertainty and competition. Part A: Existence of Nash equilibrium.* Operations Research, sub-

k	x_1^k	x_2^k	$V_\alpha(x^k)$	stepsize
0	1.000000	1.000000	1.999950000000	0.000
1	1.000000	0.000000	0.000000000000	1.000

Table 5: Numerical results for Example 6.3 using $x^0 = (1, 1)^T$

mitted.

[2] E. Adida and G. Perakis: *Dynamic pricing and inventory control: Uncertainty and competition. Part B: An algorithm for the normalized Nash equilibrium.* Operations Research, submitted.

[3] J.-P. Aubin: *Optima and Equilibria. An Introduction to Nonlinear Analysis.* Springer, Berlin, second edition 1998.

[4] T. Başar and G.J. Olsder: *Dynamic Noncooperative Game Theory.* Academic Press, New York, second edition 1995 (reprinted by SIAM, Philadelphia, 1999).

[5] S. Berridge and J.B. Krawczyk: *Relaxation algorithms in finding Nash equilibria.* Economic working papers archives, 1997, http://econwpa.wustl.edu/eprints/comp/papers/9707/9707002.abs

[6] J. Contreras, M. Klusch, and J.B. Krawczyk: *Numerical solutions to Nash-Cournot equilibria in coupled constraint electricity markets.* IEEE Transactions on Power Systems 19, 2004, pp. 195–206.

[7] E. van Damme: *Stability and Perfection of Nash Equilibria.* Springer, Berlin, Germany, second edition 1996.

[8] F. Facchinei, A. Fischer, and V. Piccialli: *Generalized Nash equilibrium problems and Newton methods.* Mathematical Programming, to appear.

[9] F. Facchinei and C. Kanzow: *Generalized Nash equilibrium problems.* 4OR – A Quarterly Journal of Operations Research 5, 2007, pp. 173–210.

[10] S.D. Flåm and A. Ruszczyński: *Noncooperative convex games: Computing equilibria by partial regularization.* IIASA Working Paper 94-42, Laxenburg, Austria, May 1994.

[11] F. Forgó, J. Szép, and F. Szidarovszky: *Introduction to the Theory of Games. Concepts, Methods, Applications.* Kluwer Academic Publishers, Dordrecht, Netherlands, 1999.

[12] D. Fudenberg and J. Tirole: *Game Theory.* MIT Press, Cambridge, MA, 1991.

[13] G. Gürkan and J.-S. Pang: *Approximations of Nash equilibria.* Mathematical Programming, to appear.

[14] A. von Heusinger and C. Kanzow: *Optimization reformulations of the generalized Nash equilibrium problem using Nikaido-Isoda-type functions.* Computational Optimization and Applications, to appear.

[15] A. von Heusinger and C. Kanzow: *SC^1 optimization reformulations of the generalized Nash equilibrium problem.* Technical Report, University of Würzburg, Würzburg, Germany, May 2007 (revised January 2008).

[16] J.-B. Hiriart-Urruty and C. Lemaréchal: *Convex Analysis and Minimization Algorithms I.* Springer, Berlin, Germany, 1993.

[17] A. Kesselman, S. Leonardi, and V. Bonifaci: *Game-theoretic analysis of internet switching with selfish users.* Lecture Notes in Computer Science 3828, 2005, pp. 236–245.

[18] J.B. Krawczyk: *Coupled constraint Nash equilibria in environmental games.* Resource and Energy Economics 27, 2005, pp. 157–181.

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

[20] G. Mastroeni: *Gap functions for equilibrium problems.* Journal of Global Optimization 27, 2003, pp. 411–426.

[21] R.B. Myerson: *Game Theory. Analysis of Conflict.* Harvard University Press, Cambridge, MA, 1991.

[22] J. Outrata, M. Kocvara and J. Zowe *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints.* Kluwer Academic Publishers, Dordrecht, Netherlands, 1998.

[23] R.T. Rockafellar: *Convex Analysis.* Princeton University Press, Princeton, NJ, 1970.

[24] J.B. Rosen: *Existence and uniqueness of equilibrium points for concave N-person games.* Econometrica 33, 1965, pp. 520–534.

[25] K. Taji, M. Fukushima, and T. Ibaraki: *A globally convergent Newton method for solving strongly monotone variational inequalities.* Mathematical Programming 58, 1993, pp. 369–383.

[26] S. Uryasev and R.Y. Rubinstein: *On relaxation algorithms in computation of noncooperative equilibria.* IEEE Transactions on Automatic Control 39, 1994, pp. 1263–1267.

k	x_1^k	x_2^k	x_3^k	$V_\alpha(x^k)$	stepsize
$P = 75$					
0	10.000000	10.000000	10.000000	1028.878642907024	0.000
1	13.012778	14.054536	15.077163	3.586375155786	1.000
2	11.285845	13.311206	15.235897	0.354069959857	1.000
3	10.704937	13.106986	15.332160	0.036616039323	1.000
4	10.507359	13.052947	15.377112	0.003948786754	1.000
5	10.439599	13.039442	15.395812	0.000440170522	1.000
6	10.416257	13.036372	15.403098	0.000050438731	1.000
7	10.408172	13.035815	15.405832	0.000005904036	1.000
8	10.405351	13.035815	15.406824	0.000000702526	1.000
9	10.404380	13.035815	15.407195	0.000000084802	1.000
10	10.404035	13.035852	15.407322	0.000000000000	1.000
$P = 100$					
0	10.000000	10.000000	10.000000	1836.050150600377	0.000
1	17.833057	19.050570	20.189450	4.898567426891	1.000
2	15.207025	18.069382	20.605731	0.389727842587	1.000
3	14.408253	17.849904	20.795588	0.033154445717	1.000
4	14.161948	17.805303	20.868540	0.002976203103	1.000
5	14.085260	17.797975	20.894315	0.000278156683	1.000
6	14.061205	17.797524	20.903000	0.000026779751	1.000
7	14.053616	17.797912	20.905860	0.000002633959	1.000
8	14.051210	17.798178	20.906771	0.000000263170	1.000
9	14.050445	17.798303	20.907059	0.000000026572	1.000
10	14.050201	17.798354	20.907149	0.000000000000	1.000
$P = 150$					
0	10.000000	10.000000	10.000000	2960.339138269361	0.000
1	27.861563	29.366476	30.558893	3.398159424100	1.000
2	24.632651	28.734752	31.626021	0.180425444160	1.000
3	23.846733	28.671683	31.919700	0.010223508454	1.000
4	23.653032	28.675775	31.995979	0.000607189161	1.000
5	23.604841	28.681019	32.015205	0.000037283006	1.000
6	23.592766	28.683240	32.019964	0.000002343399	1.000
7	23.589725	28.684005	32.021141	0.000000149536	1.000
8	23.588952	28.684235	32.021422	0.000000009467	1.000
9	23.588757	28.684300	32.021486	0.000000000000	1.000
$P = 200$					
0	10.000000	10.000000	10.000000	3592.920967502584	0.000
1	38.595611	40.204250	41.080180	1.286880698864	1.000
2	36.344520	40.610613	42.510038	0.041767272152	1.000
3	35.896150	40.715632	42.751509	0.001512281298	1.000
4	35.807282	40.741277	42.793422	0.000057432013	1.000
5	35.789684	40.747242	42.800846	0.000002231508	1.000
6	35.786198	40.748592	42.802181	0.000000087882	1.000
7	35.785505	40.748881	42.802426	0.000000003479	1.000
8	35.785366	40.748943	42.802473	0.000000000124	1.000
9	35.785345	40.748954	42.802477	0.000000000000	1.000

Table 6: Numerical results for Example 6.4