Numerical Methods for the Solution of the Generalized Nash Equilibrium Problem

Anna von Heusinger

PhD Thesis

Würzburg, August 2009

1. Gutachter:
Prof. Dr. C. Kanzow
Universität Würzburg
Institut für Mathematik
Lehrstuhl für Angewandte Mathematik II

2. Gutachter:
Prof. M. Fukushima
Kyoto University
Graduate School of Informatics
Department of Applied Mathematics and Physics

Acknowledgements

First of all I thank my PhD advisor Christian Kanzow for constant support throughout the past four years. His ideas, guidance, corrections and additions have contributed essentially. Further, I am grateful that I had the opportunity to study three months at the System Optimization Laboratory of Kyoto University, and I thank Masao Fukushima and the members of the Laboratory for the hospitality and scientific advice.

This work was supported in part by a grant from the International Doctorate Program 'Identification, Optimization and Control with Applications in Modern Technologies' within the Elite Network of Bavaria. The doctorate program also financed the three month stay at the university of Kyoto and the participation at national and international conferences, which, together with the regular seminars and summer schools with the members of the doctorate program, were of great benefit to me.

Also I would like to thank my former teachers Dr. Schreiber, Mack, Angelkort, Dr. Friedrich and Heuser from the Anne-Frank Gymnasium Werne, who promoted my interest in mathematics and physics in the first instance.

At times, working on a PhD thesis can be demanding, and I am very happy about the support and affection from my family and friends.

Contents

Chapter 1

Introduction

The Nash equilibrium, and game theory in general, is nowadays present in various fields of science, most prominently in economics and social science. More recently, also engineering sciences have discovered the Nash equilibrium concept as a means to design technical systems, for instance telecommunication networks. Often not only the formulation of a model is desired but also the actual computation of a Nash equilibrium. This thesis is about the numerical computation of Nash equilibria, more precisely, generalized Nash equilibria. For very simple games, such as two-player games with two strategies a player each, it is possible to calculate a Nash equilibrium analytically, that is, with the help of pencil and paper. Here we aim at the development of numerical methods for the computation of Nash equilibria in a general setting. More precisely, we consider games with finitely many players with continuous cost functions and finite-dimensional strategy sets.

Four different numerical methods are being presented, which are all based on either an optimization reformulation or a fixed point reformulation of the generalized Nash equilibrium problem. These reformulations are introduced in the next chapter. Chapters 3-5 deal with the numerical methods. In chapter 3, descent methods for the solution of constrained and unconstrained optimization reformulations are considered. These methods are designed to be globally convergent, however, local convergence is rarely faster than linear. Therefore, in chapter 4 a Newton-type method is derived through an unconstrained optimization reformulation of the generalized Nash equilibrium problem. Another locally superlinearly convergent method is presented in chapter 5, where a Newton-type method based on a fixed point formulation of the generalized Nash equilibrium problem is considered. Finally some examples of generalized Nash equilibrium problems are described in chapter 6, and numerical results of four numerical methods presented.

In the remainder of this introduction we give a formal definition of the generalized Nash equilibrium problem, and of a particular subclass of these generalized

Nash equilibria called normalized Nash equilibria, on which we focus in this thesis. A popular area where generalized Nash equilibria are applied is in the modelling of the liberalized electricity markets, which is why we present an electricity market model next. The introduction closes with an overview on existing work on the numerical computation of generalized Nash equilibria.

We begin with some terms and results from optimization theory that will be used in this introduction and later on.

1.1 Preliminaries

Here we state some basic facts from optimization theory and clarify our notation. A nonempty set $X \subseteq \mathbb{R}^n$ is said to be *convex*, if for all $x, y \in X$ and $t \in [0, 1]$ we have $tx + (1 - t)y \in X$. A function $f : X \to \mathbb{R}$ is *convex*, if for all $x, y \in X$ and $t \in [0, 1]$ the inequality $f(tx + (1 - t)y) \leq tf(x) + (1 - t)f(y)$ holds.

Given a convex, continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ and convex, continuously differentiable functions $g_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \ldots, m$, we consider the constrained optimization problem

$$\begin{array}{lll} \min & f(x) & \\ \text{subject to} & g_i(x) \leq 0 & \quad \text{for all } i = 1, \ldots, m. \end{array} \tag{1.1}$$

Due to the convexity of the function f, every local minimum of (1.1) is already a global minimum. We say that *Slater's constraint qualification* holds for the convex optimization problem (1.1), if there is a vector $\bar{x}$ such that $g_i(\bar{x}) < 0$ for all $i = 1, \ldots, m$. Slater's constraint qualification implies that for any solution x^* of problem (1.1) there exists a vector of Lagrange multipliers $\lambda = (\lambda_1, \ldots, \lambda_m)^T$ such that the Karush Kuhn Tucker conditions

$$\begin{array}{c} \nabla f(x^*) + \sum_{i=1}^m \lambda_i \nabla g_i(x^*) = 0 \\ 0 \geq g(x^*) \perp \lambda \geq 0. \end{array}$$

hold, where $g(x^*) \perp \lambda$ means that the vector $g(x^*)$ is perpendicular to the vector λ, that is, $\lambda^T g(x^*) = 0$.

Notation: Given a differentiable function $g : \mathbb{R}^n \to \mathbb{R}^m$, $g'(x) \in \mathbb{R}^{m \times n}$ or $Dg(x)$ denotes the Jacobian of g at x, whereas $\nabla g(x) \in \mathbb{R}^{n \times m}$ is the transposed Jacobian. In particular, for $m = 1$, the gradient $\nabla g(x)$ is viewed as a column vector. Several times, we also consider partial derivatives of a real-valued function $f : \mathbb{R}^n \to \mathbb{R}$ with respect to certain block components of x only, and this will be denoted by using suitable subscripts, e.g., $\nabla_{x^\nu} f(x)$ denotes the partial gradient of f at x, where the derivatives are taken with respect to the components of the block vector x^ν only. Second-order partial derivatives with respect to certain block components are written in a similar way as $\nabla^2_{x^\nu x^\mu} f(x)$, for example, meaning that we first differentiate with respect to x^ν and then with respect to x^μ.

For a matrix $A \in \mathbb{R}^{m\times n}$ and a subset $I \subseteq \{1,\dots,n\}$ we denote by A_I the submatrix of A consisting of the columns a_i, $i \in I$. For a vector $d \in \mathbb{R}^n$ we write $d \geq 0$ if $d_i \geq 0$ for all $i = 1,\dots,m$.

1.2 The Generalized Nash Equilibrium Problem

The generalized Nash equilibrium is a solution concept for a particular class of games. Basically a game is described through a number of *players*, their *strategy sets* and their *cost functions*. Let the number of players be N. To refer to a particular player, we use the index $\nu \in \{1,\dots,N\}$. Each player controls a *decision vector* x^ν, where x^ν has to be chosen from a set $X_\nu \subseteq \mathbb{R}^{n_\nu}$. The set X_ν is usually called the *strategy set* of player ν, or the *feasible set* of player ν. Let $n := \sum_{\nu=1}^N n_\nu$, and $x := (x^1, x^2, \dots, x^\nu, \dots, x^N) \in \mathbb{R}^n$ denote the vector that comprises the decision vectors of all players. We write $(x^\nu, x^{-\nu}) := x$ if we want to emphasize the νth player's decision vector within x. Here the vector $x^{-\nu} = (x^1, \dots, x^{\nu-1}, x^{\nu+1}, \dots, x^N)$ is short notation for the vector that consists of all the decision vectors except player νth decision variables.

Each player ν has a *cost function* $\theta_\nu : \mathbb{R}^n \to \mathbb{R}$. Given a decision vector $x = (x^1, x^2, \dots, x^N)$, player ν incurs costs $\theta_\nu(x^1, \dots, x^N)$. Thus, the cost function of player ν does not only depend on player νth decision vector x^ν, but also on all other players decision vectors $x^{-\nu}$. Altogether, a game is fully described by $\Gamma := \{X_\nu, \theta_\nu\}_{\nu=1,\dots,N}$.

We assume that a player acts rational in that, given the decision vector $x^{-\nu}$ of the rival players, he chooses a decision vector x^ν that minimizes his cost function. In other words, given a vector $x^{-\nu}$, player ν solves the optimization problem

$$\min_{x^\nu} \theta_\nu(x^\nu, x^{-\nu}) \qquad \text{subject to } x^\nu \in X_\nu.$$

A *Nash equilibrium* (or *solution to the Nash equilibrium problem*) is a vector x^* with $x^* = (x^{*,1}, x^{*,2}, \dots, x^{*,N}) \in X_1 \times X_2 \times \cdots \times X_N$, such that for each $\nu \in \{1,\dots,N\}$ the vector $x^{*,\nu}$ solves the optimization problem

$$\min_{x^\nu} \theta_\nu(x^\nu, x^{*,-\nu}) \qquad \text{subject to } x^\nu \in X_\nu. \tag{1.2}$$

Literally speaking, at a Nash equilibrium point x^*, no player has an intend to change his own decision vector as long as the other players do not change their decision vector. In the sequel we will use the term *standard Nash equilibrium problem, Nash equilibrium problem* or *standard Nash game,* in short NEP, to refer to the above problem, in contrast to the *generalized Nash equilibrium problem*, GNEP, which we describe next.

In the generalized Nash game not only the cost functions depend on the rival players' decision variables, but also the strategy sets X_ν, $\nu = 1, \dots, N$. Let $X \subseteq \mathbb{R}^n$ be a nonempty, closed and convex set. The set X doesn't necessarily feature a cartesian product structure.

For each $\nu \in \{1, \dots, N\}$ we define the strategy set of player ν,

$$X_\nu(x^{-\nu}) := \{x^\nu \in \mathbb{R}^{n_\nu} \mid (x^\nu, x^{-\nu}) \in X\}, \tag{1.3}$$

which means that the strategy set of player ν is given by a set-valued map $X_\nu : \mathbb{R}^{n-n_\nu} \to \mathbb{R}^{n_\nu}$. In analogy to the definition of the standard Nash equilibrium, we arrive at the following definition of a generalized Nash equilibrium.

Definition 1.2.1 *A vector* $x^* = (x^{*,1}, x^{*,2}, \dots, x^{*,N})$ *is a* generalized Nash equilibrium (GNE) *or a solution to the* generalized Nash equilibrium problem (GNEP), *if for all* $\nu = 1, 2, \dots, N$, *the block component* $x^{*,\nu}$ *of* x^* *solves the optimization problem*

$$\min_{x^\nu} \theta_\nu(x^\nu, x^{*,-\nu}) \qquad \textit{subject to } x^\nu \in X_\nu(x^{*,-\nu}). \tag{1.4}$$

Other terms than *generalized Nash equilibrium* are *coupled constraint Nash equilibrium* [59], [60] *Nash equilibrium problem with shared constraints* [71],[32] and *social equilibrium problem*. Some of these terms refer to the fact that the individual strategy sets of the players are defined through a single convex set X by equation (1.3). In a wider sense, the term 'generalized Nash equilibrium problem' refers to Nash equilibrium problems where the feasible sets $X_\nu(x^{-\nu})$ of the players can not be expressed through a single set X, see [25] for a more detailed description.

From now on, unless otherwise mentioned, we will always assume that the following assumptions concerning the cost functions θ_ν, $\nu = 1, \dots, N$, and the strategy set X hold.

Assumption 1.2.2

(A.1) *For all* $\nu \in \{1, \dots, N\}$ *the cost function* θ_ν *is continuous as a function of* x. *Furthermore,* θ_ν *is convex as a function of the variable* x^ν, *i.e., the function* $\theta_\nu(\cdot, x^{-\nu})$ *is convex for all* $x^{-\nu}$.

(A.2) *The strategy set* X *is nonempty, convex and closed.*

1.3 Normalized Nash Equilibria

An important subclass of the set of generalized Nash equilibria is the class of *normalized Nash equilibria*. It was introduced by J.B. Rosen [92] in a slightly different fashion than described below.

Throughout this section, we assume that in addition to Assumption 1.2.2 the cost functions θ_ν, $\nu = 1, \dots, N$, are continuously differentiable. We consider the function

$$F(x^1, x^2, \dots, x^N) := \begin{pmatrix} \nabla_{x^1}\theta_1(x^1, x^{-1}) \\ \nabla_{x^2}\theta_2(x^2, x^{-2}) \\ \vdots \\ \nabla_{x^N}\theta_N(x^N, x^{-N}) \end{pmatrix},$$

where $\nabla_{x^\nu}\theta_\nu(x^\nu, x^{-\nu})$ denotes the partial gradient of θ_ν with respect to the block component x^ν. Then x^* is a called a *normalized Nash equilibrium* (or *variational equilibrium* in [25]), if and only if $x^* \in X$ satisfies

$$F(x^*)^T(y - x^*) \geq 0 \quad \text{for all } y \in X, \tag{1.5}$$

in other words, x^* is the solution of a variational inequality. In order to clarify the meaning of normalized Nash equilibria, let us consider equation (1.5) with $y := (z^\nu, x^{*,-\nu})$. Since $x^* \in X$ there is z^ν such that $y \in X$, implying that $z^\nu \in X_\nu(x^{*,-\nu})$. Thus (1.5) reduces to the first order necessary conditions for player ν's optimization problem (1.4), that is,

$$\nabla_{x^\nu}\theta_\nu(x^*)^T(z^\nu - x^{*,\nu}) \geq 0 \qquad \forall\, z^\nu \in X_\nu(x^{*,-\nu}).$$

Since $\theta_\nu(\cdot, x^{-\nu})$ is convex by Assumption 1.2.2, the first order condition already implies that $x^{\nu,*}$ solves optimization problem (1.4) of player ν. Therefore, any normalized Nash equilibrium is a generalized Nash equilibrium. The converse is not true in general. This follows from the above considerations, in particular, from the fact that the first order necessary conditions for x^* to be a solution of the GNEP, which are

$$F(x^*)^T(y - x^*) \geq 0 \quad \text{for all } y \in X_1(x^{*,-1}) \times \cdots \times X_N(x^{*,-N}), \tag{1.6}$$

may admit more solutions than the variational inequality (1.5). The above formulation (1.6) of the GNEP is a *quasi-variational inequality* (QVI) [43],[78]. However, in the standard Nash game, the set of Nash equilibria is the same as the set of normalized Nash equilibria, whereas the role of variational inequality problems within generalized Nash games is precisely identified through the notion of normalized Nash equilibria ([6], [43] and [31]).

Since a normalized Nash equilibrium is the solution of the particular variational inequality problem (1.5), one can apply the extensive theory on variational inequalities ([27],[28]) to deduce existence and uniqueness results for normalized Nash equilibria. For instance, it follows that a game satisfying Assumptions 1.2.2 with continuously differentiable cost functions, having the additional property that the strategy set X is compact, has at least one normalized Nash equilibrium. If furthermore, the function F is strictly monotone, that is, for all $x, y \in X$ with $x \neq y$ the inequality

$$(F(x) - F(y))^T (x - y) > 0$$

holds, then the normalized Nash equilibrium is unique. If, in addition, F is strongly monotone, that is, there exists a parameter $\mu > 0$ such that the inequality

$$(F(x) - F(y))^T (x - y) \geq \mu \|x - y\|^2$$

holds for all $x, y \in X$, then there is a unique normalized Nash equilibrium regardless of whether X is compact or not.

The normalized Nash equilibrium has an interesting application in that it provides a solution concept for a particular class of leader-follower games. This connection has been noted by Harker in a remark at the very end of [43], and a somewhat related idea is explored by Krawczyk [60, Section 2.3] in the context of an environmental pollution model.

We consider N players, where each player ν chooses a strategy $x^\nu \in \mathbb{R}^{n_\nu}$ subject to an individual constraint $g_\nu(x^\nu) \leq 0$, $g_\nu : \mathbb{R}^{n_\nu} \to \mathbb{R}^{m_\nu}$. Playing x^ν results in costs $\theta_\nu(x^\nu, x^{-\nu})$ for agent ν, while it requires the use of l differnet scarce resources at level $d_\nu(x^\nu) \in \mathbb{R}^l$, where d_ν is a convex function. We assume that the price for these resources is fixed and that it is equal for all players, that is, prices for resources are given by a single vector $p \in \mathbb{R}^l$. Thus, given the decision vector $x^{-\nu}$ of the rival players, player ν solves the optimization problem

$$\begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, x^{-\nu}) + p^T d_\nu(x^\nu) \\ \text{subject to} & g_\nu(x^\nu) \leq 0. \end{array} \tag{1.7}$$

For $x^{-\nu}$ fixed let $x^{*,\nu}$ be a solution of this optimization problem and suppose that Slater's constraint qualification holds. Moreover, we assume that all functions involved are continuously differentiable. Then the Karush-Kuhn-Tucker conditions are both necessary and sufficient for $x^{*,\nu}$ to be a solution of (1.7), implying that there exists a vector of multipliers $\lambda^{*,\nu} \in \mathbb{R}^{m_\nu}$ such that $(x^{*,\nu}, \lambda^{*,\nu})$ solves the equations

$$\begin{array}{c} \nabla_{x^\nu} \theta_\nu(x^{*,\nu}, x^{-\nu}) + \nabla d_\nu(x^{*,\nu}) \cdot p + \nabla g_\nu(x^{*,\nu}) \cdot \lambda^{*,\nu} = 0 \\ 0 \geq g_\nu(x^{*,\nu}) \perp \lambda^{*,\nu} \geq 0. \end{array} \tag{1.8}$$

Solving optimization problem (1.7) for all players $\nu = 1, \ldots, N$ simultaneously is a standard Nash equilibrium problem. Each player ν has a separate strategy set defined through the function g_ν. Suppose now that there is an additional player who has the power to set prices for the resources. We assume that this player, the leader, is non-discriminating in that he demands the same price from all players, his only intend being that the total resource consumption of all players does not exceed certain upper bounds. This leader might be a governmental authority caring for natural resources like water, minerals, pollutant emissions or land usage.

In order to find a price vector such that aggregated resource consumption of players does not surmount prescribed limits, we formulate a generalized Nash equilibrium problem. In this game, the functions θ_ν are still the cost functions of the players, whose individual strategy set is given by $X_\nu = \{x^\nu \in \mathbb{R}^{n_\nu} \mid g_\nu(x^\nu) \le 0\}$. Different to the standard Nash game above, joint constraints are imposed on the players of the form

$$\bar{X} := \{x \in \mathbb{R}^n \mid \sum_{\nu=1}^{N} d_\nu(x^\nu) \le c\},$$

with $d_\nu : \mathbb{R}^{n_\nu} \to \mathbb{R}^l$ and a given constant $c \in \mathbb{R}^l$.
Consider the optimization problem

$$\begin{array}{ll} \min_{x^\nu} & \theta_\nu(x^\nu, x^{-\nu}) \\ \text{subject to} & g_\nu(x^\nu) \le 0,\ d_\nu(x^\nu) + \sum_{\xi \neq \nu} d_\xi(x^\xi) \le c \end{array} \tag{1.9}$$

for player ν and let $x^{*,\nu}$ be a solution for given $x^{-\nu}$. Assuming that Slater's constraint qualification holds for (1.9), it follows that there are vectors $\lambda^{*,\nu} \in \mathbb{R}^{m_\nu}$ and $\mu^{*,\nu} \in \mathbb{R}^l$ such that the triple $(x^{*,\nu}, \lambda^{*,\nu}, \mu^{*,\nu})$ solves the Karush-Kuhn-Tucker equations

$$\begin{aligned} \nabla_{x^\nu}\theta_\nu(x^{*,\nu}, x^{-\nu}) + \nabla g_\nu(x^{*,\nu}) \cdot \lambda^{*,\nu} + \nabla_{x^\nu} d_\nu(x^{*,\nu}) \cdot \mu^{*,\nu} = 0 \\ 0 \ge g_\nu(x^{*,\nu}) \perp \lambda^{*,\nu} \ge 0 \\ 0 \ge \textstyle\sum_{\xi=1}^{N} d_\xi(x^\xi) - c \perp \mu^{*,\nu} \ge 0. \end{aligned}$$

A comparision with the KKT-condition of optimization problem (1.7) shows that $(x^{*,\nu}, \lambda^{*,\nu})$ solves (1.8) with $p := \mu^{*,\nu}$.

Concatenating these KKT-conditions for all players $\nu = 1, \dots, N$, we see that every solution (x^*, λ^*, μ^*) of the generalized Nash equilibrium problem with the property that $\mu^{*,1} = \mu^{*,2} = \cdots = \mu^{*,N}$ solves the KKT-conditions of problems (1.7) for all players ν with the Lagrange multiplier $p := \mu^{*,1}$. Since $\mu^{*,1} = \mu^{*,2} = \cdots = \mu^{*,N}$, we have a normalized Nash equilibrium. This particular solution of the optimization problems (1.7) has the additional property that $\sum_{\xi=1}^{N} d_\xi(x^\xi) \le c$, where equality holds whenever the corresponding component of p is nonzero. The latter fact allows two economic interpretations, which we will outline in brief. In the scenario of a government facing the decision of whether to impose taxes or not, taxes are not installed whenever the aggregated consumption of the resource (natural resources) does not exceed the targeted amount.

On the other hand, the model allows a quite different interpretation. Suppose that resources, like oil or minerals, are being auctioned on a market with infinitely many sellers. As long as the total demand for a resource is below the available amount, prices for this resource will be equal to the costs it takes to obtain them, for instance costs for oil extraction. (This case is not considered in the model, but the extension is straightforward). Whenever there is a shortfall, prices will rise due to scarcity, which corresponds to a non-zero price in the model above.

1.4 Electricity Market Model

Here we consider an electricity market model with two competitors sharing a power line network that is owned by a third party. The network consists of four nodes with different consumers at each node, see figure 1.1.

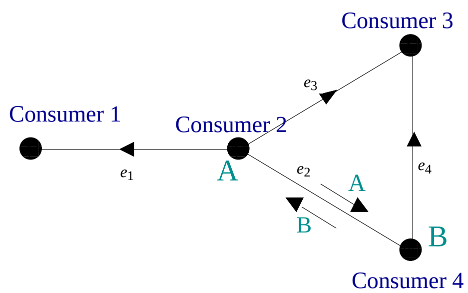

Figure 1.1: Electricity Market Example

Consumer 1 for instance can be interpreted as a remote rural area with low electricity demand, consumers 2 and 4 as big industrial cities and consumer 3 as a smaller city with medium-level electricity demand. Each company owns one power plant, that of company A being at node 2 and that of company B at node 4. Nodes are linked by four power transmission lines with different capacity and different costs (due to different voltage and length). Routing electricity through a power line incurs costs on a company proportional to the current in that particular line. Each company decides about the amount of electricity it sells at each node. Since costs for shipping electricity depend on the paths chosen in the network, the company also has to decide how to route the electricity. This is fully described by one additional decision variable, since the network contains only one loop. In general, power lines can be used in both directions. In this model, however, the direction of flow is prescribed in order to avoid nonsmoothness in the cost functions of the companies.

The model contains the following variables and parameters:

$x_i^A,\ x_i^B$	: electricity company A (B) sells to consumer i
y^A	: current on edge 4 from company A
y^B	: current on edge 4 from company B
$c^A,\ c^B$	: costs for producing one MWh with power plant A (B)
k_i	: capacity limit of link i
e_i	: costs on link i
C_i	: specific constant of consumer i
γ	: parameter of inverse demand function

We suppose that the price for electricity is given by an inverse demand function, that differs in each node due to number and preferences of consumers, that is, electricity price at node i is

$$p_i(x^A, x^B) = C_i^\gamma \cdot \frac{1}{(x_i^A + x_i^B)^\gamma}, \quad i = 1, 2, 3, 4.$$

The cost function of company A is

$$f^A(x^A, y^A) = c^A \sum_{i=1}^{4} x_i^A + e_1 x_1^A + e_2(x_4^A + y^A) + e_3(x_3^A - y^A) + e_4 y^A$$

while company B's costs are

$$f^B(x^B, y^B) = c^B \sum_{i=1}^{4} x_i^B + e_1 x_1^B + e_2(x_1^B + x_2^B + x_3^B - y^B) + e_3(x_3^B - y^B) + e_4 y^B.$$

The profit functions are

$$\pi^A(x^A, y^A, x^B) = p(x^A, x^B)^T \cdot x^A - f^A(x^A, y^A)$$

and

$$\pi^B(x^A, x^B, y^B) = p(x^A, x^B)^T \cdot x^B - f^B(x^B, y^B),$$

respectively. The joint constraints imposed through the capacity limits of the power line are

$$\begin{aligned} x_1^A + x_1^B &\leq k_1, \\ x_4^A + y^A - x_1^B - x_2^B - x_3^B + y^B &\leq k_2, \\ -x_4^A - y^A + x_1^B + x_2^B + x_3^B - y^B &\leq k_2, \\ x_3^A + x_3^B - y^A - y^B &\leq k_3, \\ y^A + y^B &\leq k_4. \end{aligned}$$

Further, for technical reasons we assume that electricity sales are strictly positive at all markets

$$\begin{aligned} x^A &\geq 0.1, \\ x^B &\geq 0.1, \end{aligned}$$

and two additional constraints that prevent negative flow on edge 3,

$$\begin{aligned} y^A &\leq x_3^A, \\ y^B &\leq x_3^B. \end{aligned}$$

Numerical results for this model are presented in chapter 6.

1.5 Previous Work

The first appearance of generalized Nash games, though not termed this way, is probably in the seminal work of Arrow and Debreu [5] on the existence of an equilibrium in abstract economies in 1954. In 1965, Rosen formally introduced the definition of a normalized Nash equilibrium and considered questions of existence and uniqueness. Rosen also proposed a gradient method for computing a normalized Nash equilibrium. However, the generalized Nash equilibrium problem did not attract particular attention for a long time after the work of Rosen. Bensoussan [11] formulated the GNEP as a quasi-variational inequality in 1974, though with infinite-dimensional strategy sets, which Harker [43] further explored for the finite-dimensional case 15 years later. Harker pointed out that the generalized Nash equilibrium problem encompasses a class of Stackelberg-like (Leader-follower) problems.

In the late nineties, generalized Nash games became popular for modelling environmental and energy economic issues, as well as for the design and analysis of telecommunication networks [15], [45], [59] and [61]. Some of the models presented in these papers are described in chapter 6.

Still the numerical solution of the generalized Nash equilibrium problem remained a difficult task. While the standard Nash equilibrium problem 1.2 (NEP) is equivalent to the variational inequality problem 1.5, see [44], as mentioned earlier, the generalized Nash equilibrium problem is not. Nonetheless, numerical methods designed for the solution of the variational inequality problems have constantly inspired approaches towards the solution of the GNEP.

In the eighties, when numerical methods for solving variational inequality problems were yet widely unknown, Dafermos [17] proposed an iterative method for the solution of variational inequalties, which is based on a fixed point formulation. This method resembles somewhat the Jacobi iteration for the solution of a linear system, and is therefore also called Jacobi or Gauss-Seidel method [25]. Başar and Li [64],[9], applied this method in the particular context of NEPs.

Some years later, in 1994, Uryasev and Rubinstein [98] investigated a fixed point iteration for the computation of normalized Nash equilibria, called relaxation method. They improved on existing convergence theory for relaxation methods, in that they did not require differentiability of the cost functions. This relaxation method has been applied by several different authors since then.

Other approaches towards the numerical solution of the GNEP include Penalty methods [78], [39],[29], [26], and solution methods for quasi-variational inequalities [77]. Also the gradient method introduced by Rosen received some further attention. Primal-dual gradient methods are investigated by Flåm [36] and for the NEP by Antipin in [3]. Further relaxation-type methods were analysed in [37] for a very restricted class of generalized Nash equilibrium problems called *convex games*. Quite different from all prior approaches, Nabetani, Fukushima and Tseng [71] compute generalized Nash equilibria through repeated solution of parameterized variational inequalities.

An overview on numerical methods for some Nash games other than the GNEP provides the monograph [70]. These are in particular games defined on graphs and bimatrix games. The book contains one article (chapter 6) about computing an equilibrium in the pure exchange economy, which can be cast as a GNEP.

Chapter 2

Optimization Reformulations

In this chapter we derive constrained and unconstrained optimization reformulations of the generalized Nash equilibrium problem. The first section starts with a constrained optimization reformulation that yields a full characterization of the set of generalized Nash equilibria. The drawback of this optimization reformulation is, however, that the objective function of the optimization problem is in general not differentiable, and therefore the problem itself not easy to solve with existing optimization routines. Thus, in the next section, we present a constrained smooth optimization reformulation. This reformulation characterizes a subset of the set of generalized Nash equilibrium, precisely, it gives the set of normalized Nash equilibria. Alongside with the two constrained optimization reformulations we derive fixed point formulations of the solutions of the the generalized Nash equilibrium problem. The last section deals with an unconstrained smooth optimization reformulation, which again characterizes the set of normalized Nash equilibria. The approach is somewhat related to the work on equilibrium problems in [13] and the recent paper [66].

2.1 A Constrained Optimization Reformulation

An important tool in the theoretical analysis of the generalized Nash equilibrium problem is the so-called *Nikaido-Isoda function*. This function, sometimes also called Ky-Fan function, was introduced originally in order to prove the existence of a Nash equilbrium by means of a fixed point theorem [73]. In the following, however, the Nikaido-Isoda function will be the main tool for the development of numerical methods for the solution of the generalized Nash equilibrium problem. Let θ_ν, $\nu = 1, \dots, N$ be the cost functions as described in the introduction. The

Nikaido-Isoda function is defined through

$$\Psi(x,y) := \sum_{\nu=1}^{N} [\theta_\nu(x^\nu, x^{-\nu}) - \theta_\nu(y^\nu, x^{-\nu})]. \tag{2.1}$$

Let $X_\nu(x^{-\nu})$ be player ν's strategy set as defined in (1.3). For given $x \in \mathbb{R}^n$ we write

$$\Omega(x) := X_1(x^{-1}) \times X_2(x^{-2}) \times \cdots \times X_N(x^N). \tag{2.2}$$

The following Lemma connects the set X with the set-valued map $\Omega(x)$.

Lemma 2.1.1 *We have $x \in \Omega(x)$ if and only if $x \in X$. In particular, $\Omega(x) \neq \emptyset$ for all $x \in X$.*

Proof. Using the definitions of the sets $\Omega(x)$ and $X_\nu(x^{-\nu})$, we immediately obtain

$$\begin{aligned} x \in \Omega(x) \quad &\Longleftrightarrow \quad x^\nu \in X_\nu(x^{-\nu}) \quad \forall \nu = 1, \dots, N \\ &\Longleftrightarrow \quad (x^\nu, x^{-\nu}) \in X \quad \forall \nu = 1, \dots, N \\ &\Longleftrightarrow \quad x = (x^\nu, x^{-\nu}) \in X. \end{aligned}$$

The second part is now obvious. □

Note that, for $x \notin X$, we have either $\Omega(x) = \emptyset$ or $\Omega(x) \neq \emptyset$, but then necessarily $x \notin \Omega(x)$. Furthermore, given any $x \in X$, simple examples show that, in general, neither $\Omega(x)$ is a subset of X nor X is included in $\Omega(x)$.
Using the Nikaido-Isoda-function, we define

$$\hat{V}(x) := \sup_{y \in \Omega(x)} \Psi(x,y), \quad x \in X, \tag{2.3}$$

where, for the moment, we assume implicitly that the supremum is always attained for some $y \in \Omega(x)$. Later, this assumption will not be needed, so we do not state it here explicitly. Then it is not difficult to see that $\hat{V}(x)$ is nonnegative for all $x \in \Omega(x)$, and that x^* is a solution of the GNEP if and only if $x^* \in \Omega(x^*)$ and $\hat{V}(x^*) = 0$, see also the proof of Theorem 2.1.2 below. Therefore, finding a solution of the GNEP is equivalent to computing a global minimum of the optimization problem

$$\min \hat{V}(x) \quad \text{s.t.} \quad x \in \Omega(x). \tag{2.4}$$

Note that this optimization problem has a complicated feasible set since $\Omega(x)$ explicitly depends on x. However, in view of Lemma 2.1.1, the program (2.4) is equivalent to the optimization problem

$$\min \hat{V}(x) \quad \text{s.t.} \quad x \in X.$$

Although the Nikaido-Isoda-function is quite popular (especially for standard Nash games) in the economic and engineering literature, see, for example, [15, 59, 61], it has some disadvantages from a mathematical and practical point of view (also for the standard Nash game): On the one hand, given a vector x, the supremum in (2.3) may not exist unless additional assumptions (like the compactness of X) hold, and on the other hand, this supremum, if it exists, is usually not attained at a single point which, in turn, implies that the mapping $\hat{V}$ and, therefore, also the corresponding optimization reformulation (2.4) is nondifferentiable in general.

In order to overcome these deficiencies, we use a simple regularization of the Nikaido-Isoda-function. This idea was used earlier in several contexts, see, for example, Fukushima [38] (for variational inequalities), Gürkan and Pang [42] (for standard Nash games), and Mastroeni [66] (for equilibrium programming problems). Here we apply the regularization idea to GNEPs. To this end, let $\alpha > 0$ be a fixed parameter and define the *regularized Nikaido-Isoda-function* by

$$\Psi_\alpha(x,y) := \sum_{\nu=1}^{N} \left[\theta_\nu(x^\nu, x^{-\nu}) - \theta_\nu(y^\nu, x^{-\nu}) - \frac{\alpha}{2}\|x^\nu - y^\nu\|^2\right]. \tag{2.5}$$

Furthermore, for $x \in X$, let

$$\begin{aligned} \hat{V}_\alpha(x) &:= \max_{y\in\Omega(x)} \Psi_\alpha(x,y) \\ &= \max_{y\in\Omega(x)} \sum_{\nu=1}^{N} \left[\theta_\nu(x^\nu, x^{-\nu}) - \theta_\nu(y^\nu, x^{-\nu}) - \frac{\alpha}{2}\|x^\nu - y^\nu\|^2\right] \qquad (2.6) \\ &= \sum_{\nu=1}^{N} \left\{\theta_\nu(x^\nu, x^{-\nu}) - \min_{y^\nu\in X_\nu(x^{-\nu})} \left[\theta_\nu(y^\nu, x^{-\nu}) + \frac{\alpha}{2}\|x^\nu - y^\nu\|^2\right]\right\}. \end{aligned}$$

be the corresponding value function.

A number of elementary properties of the mapping $\hat{V}_\alpha$ are summarized in the following result.

Theorem 2.1.2 *The regularized function $\hat{V}_\alpha$ has the following properties:*

(a) $\hat{V}_\alpha(x) \geq 0$ *for all* $x \in \Omega(x)$.

(b) x^* *is a generalized Nash equilibrium if and only if* $x^* \in \Omega(x^*)$ *and* $\hat{V}_\alpha(x^*) = 0$.

(c) For every $x \in X$*, there exists a unique vector* $\hat{y}_\alpha(x) = (\hat{y}_\alpha^1(x), \dots, \hat{y}_\alpha^N(x))$ *such that for every* $\nu = 1, \dots, N$,

$$\mathrm{argmin}_{y^\nu\in X_\nu(x^{-\nu})}\left[\theta_\nu(y^\nu, x^{-\nu}) + \frac{\alpha}{2}\|x^\nu - y^\nu\|^2\right] = \hat{y}_\alpha^\nu(x).$$

Proof. (a) For all $x \in \Omega(x)$, we have $\hat{V}_\alpha(x) = \max_{y\in\Omega(x)} \Psi_\alpha(x, y) \geq \Psi_\alpha(x, x) = 0$.

(b) Suppose that x^* is a solution of the GNEP. Then $x^* \in \Omega(x^*)$ and

$$\theta_\nu(x^{*,\nu}, x^{*,-\nu}) \leq \theta_\nu(x^\nu, x^{*,-\nu}) \quad \forall x^\nu \in X_\nu(x^{*,-\nu})$$

for all $\nu = 1, \ldots, N$. Hence

$$\Psi_\alpha(x^*, y) = \sum_{\nu=1}^{N} [\underbrace{\theta_\nu(x^{*,\nu}, x^{*,-\nu}) - \theta_\nu(y^\nu, x^{*,-\nu})}_{\leq 0 \quad \forall y^\nu \in X_\nu(x^{*,-\nu})} - \frac{\alpha}{2}\|x^{*,\nu} - y^\nu\|^2] \leq 0$$

for all $y \in \Omega(x^*)$. This implies

$$\hat{V}_\alpha(x^*) = \max_{y\in\Omega(x^*)} \Psi_\alpha(x^*, y) \leq 0.$$

Together with part (a), we therefore have $\hat{V}_\alpha(x^*) = 0$.

Conversely, assume that $x^* \in \Omega(x^*)$ and $\hat{V}_\alpha(x^*) = 0$. Then $\Psi_\alpha(x^*, y) \leq 0$ holds for all $y \in \Omega(x^*)$. Let us fix a particular player $\nu \in \{1, \ldots, N\}$, and let $x^\nu \in X_\nu(x^{*,\nu})$ and $\lambda \in (0, 1)$ be arbitrary. Then define a vector $y = (y^1, \ldots, y^N) \in \mathbb{R}^n$ blockwise as follows:

$$y^\mu := \begin{cases} x^{*,\mu}, & \text{if } \mu \neq \nu, \\ \lambda x^{*,\nu} + (1-\lambda)x^\nu, & \text{if } \mu = \nu. \end{cases}$$

The convexity of the sets $X_\nu(x^{*,-\nu})$ imply that $y^\mu \in X_\mu(x^{*,-\mu})$ for all $\mu = 1, \ldots, N$, i.e., $y \in \Omega(x^*)$. For this particular y, we therefore obtain

$$\begin{aligned} 0 &\geq \Psi_\alpha(x^*, y) \\ &= \theta_\nu(x^{*,\nu}, x^{*,-\nu}) - \theta_\nu(\lambda x^{*,\nu} + (1-\lambda)x^\nu, x^{*,-\nu}) - \frac{\alpha}{2}(1-\lambda)^2\|x^{*,\nu} - x^\nu\|^2 \\ &\geq (1-\lambda)\theta_\nu(x^{*,\nu}, x^{*,-\nu}) - (1-\lambda)\theta_\nu(x^\nu, x^{*,-\nu}) - \frac{\alpha}{2}(1-\lambda)^2\|x^{*,\nu} - x^\nu\|^2 \end{aligned}$$

from the convexity of θ_ν with respect to x^ν. Dividing both sides by $1 - \lambda$ and then letting $\lambda \to 1^-$ shows that $\theta_\nu(x^{*,\nu}, x^{*,-\nu}) \leq \theta_\nu(x^\nu, x^{*,-\nu})$. Since this holds for all $x^\nu \in X_\nu(x^{*,-\nu})$ and all $\nu = 1, \ldots, N$, it follows that x^* is a solution of the GNEP.

(c) This statement follows immediately from the fact that the mapping $y^\nu \mapsto \theta_\nu(y^\nu, x^{-\nu}) + \frac{\alpha}{2}\|x^\nu - y^\nu\|^2$ is strongly convex (for any given x), also taking into account that $X_\nu(x^{-\nu})$ is a nonempty, closed and convex set. □

Note that the previous result reduces to Proposition 3 in [42] for the standard Nash equilibrium problem. Using the first two statements of Theorem 2.1.2, we see that finding a solution of the GNEP is equivalent to computing a global minimum of the constrained optimization problem

$$\min \hat{V}_\alpha(x) \quad \text{s.t.} \quad x \in \Omega(x), \tag{2.7}$$

which, in turn, can be reformulated as

$$\min \hat{V}_\alpha(x) \quad \text{s.t.} \quad x \in X$$

in view of Lemma 2.1.1. The last statement of Theorem 2.1.2 shows that the new objective function overcomes one of the deficiencies of the mapping $\hat{V}(x)$.

The following result shows that the definition of the mapping $\hat{V}_\alpha$ can also be used in order to get a fixed point characterization of the GNEP.

Proposition 2.1.3 *Let $\hat{y}_\alpha(x)$ be the vector defined in Theorem 2.1.2 (c) as the unique maximizer in the definition of the regularized function $\hat{V}_\alpha(x)$, cf.* (2.6). *Then x^* is a solution of GNEP if and only if x^* is a fixed point of the mapping $x \mapsto \hat{y}_\alpha(x)$, i.e., if and only if $x^* = \hat{y}_\alpha(x^*)$.*

Proof. First assume that x^* is a solution of the GNEP. Then we obtain $x^* \in \Omega(x^*)$ (and, therefore, $x^* \in X$ in view of Lemma 2.1.1) and $\hat{V}_\alpha(x^*) = 0$ from Theorem 2.1.2. In view of the definition of $\hat{y}_\alpha(x^*)$, this implies

$$0 = \hat{V}_\alpha(x^*) = \max_{y \in \Omega(x^*)} \Psi_\alpha(x^*, y) = \Psi_\alpha(x^*, \hat{y}_\alpha(x^*)).$$

On the other hand, we also have $\Psi_\alpha(x^*, x^*) = 0$. Since $x^* \in \Omega(x^*)$ and the maximum $\hat{y}_\alpha(x^*)$ is uniquely defined by Theorem 2.1.2, it follows that $x^* = \hat{y}_\alpha(x^*)$.

Conversely, let x^* be a fixed point of the mapping $\hat{y}_\alpha$. Then $x^* = \hat{y}_\alpha(x^*) \in \Omega(x^*)$ and

$$0 = \Psi_\alpha(x^*, x^*) = \Psi_\alpha(x^*, \hat{y}_\alpha(x^*)) = \hat{V}_\alpha(x^*).$$

Consequently, the statement follows from Theorem 2.1.2. □

We next consider a simple example which shows that, in general, the objective function from (2.7) is nondifferentiable.

Example 2.1.4 Consider the GNEP with $N = 2$ players and the following optimization problems:

$$\begin{array}{ll} \min_{x_1} & \theta_1(x_1, x_2) := -x_1 \\ \text{s.t.} & x_1 + x_2 \leq 1, \\ & 2x_1 + 4x_2 \leq 3, \\ & x_1, x_2 \geq 0, \end{array} \qquad \Bigg| \qquad \begin{array}{ll} \min_{x_2} & \theta_2(x_1, x_2) := 0 \\ \text{s.t.} & x_1 + x_2 \leq 1, \\ & 2x_1 + 4x_2 \leq 3, \\ & x_1, x_2 \geq 0. \end{array}$$

Hence we have $X = \{(x_1, x_2)^T \mid x_1 + x_2 \leq 1, 2x_1 + 4x_2 \leq 3, x_1 \geq 0, x_2 \geq 0\}$. An elementary calculation shows that the solution set is given by

$$\mathcal{S} = \left\{ x^* = (x_1^*, x_2^*) \,\Big|\, x_2^* \in [0, \frac{3}{4}], x_1^* = \left\{ \begin{array}{ll} 1 - x_2^*, & \text{if } x_2^* \in [0, \frac{1}{2}], \\ \frac{3}{2} - 2x_2^*, & \text{if } x_2^* \in [\frac{1}{2}, \frac{3}{4}] \end{array} \right. \right\}.$$

We want to compute $\hat{V}_\alpha(x)$. To this end, we first note that the regularized Nikaido-Isoda-function for this game is

$$\Psi_\alpha(x, y) = -x_1 + y_1 - \frac{\alpha}{2}(x_1 - y_1)^2 - \frac{\alpha}{2}(x_2 - y_2)^2.$$

Moreover, for this example, we have

$$\begin{aligned} X_1(x^{-1}) &= \{x_1 \,|\, x_1 \leq 1 - x_2, x_1 \leq \frac{3}{2} - 2x_2, x_1 \geq 0\} = [0, \min\{1 - x_2, \frac{3}{2} - 2x_2\}] \quad \text{and} \\ X_2(x^{-2}) &= \{x_2 \,|\, x_2 \leq 1 - x_1, x_2 \leq \frac{3}{4} - \frac{1}{2}x_1, x_2 \geq 0\} = [0, \min\{1 - x_1, \frac{3}{4} - \frac{1}{2}x_1\}] \end{aligned}$$

and, therefore

$$\hat{V}_\alpha(x) = -x_1 - \min_{y_1 \in X_1(x^{-1})} [-y_1 + \frac{\alpha}{2}(x_1 - y_1)^2] - \min_{y_2 \in X_2(x^{-2})} [\frac{\alpha}{2}(x_2 - y_2)^2].$$

Given $x = (x_1, x_2) \in \mathbb{R}^2$, the solution of the first minimization problem is given by

$$\hat{y}_\alpha^1(x) = \begin{cases} 0, & \text{if } \frac{1}{\alpha} + x_1 \leq 0, \\ \frac{1}{\alpha} + x_1, & \text{if } \frac{1}{\alpha} + x_1 \in [0, \min\{1 - x_2, \frac{3}{2} - 2x_2\}], \\ \min\{1 - x_2, \frac{3}{2} - 2x_2\}, & \text{if } \frac{1}{\alpha} + x_1 \geq \min\{1 - x_2, \frac{3}{2} - 2x_2\}, \end{cases}$$

and the solution of the second problem is

$$\hat{y}_\alpha^2(x) = \begin{cases} 0, & \text{if } x_2 \leq 0, \\ x_2, & \text{if } x_2 \in [0, \min\{1 - x_1, \frac{3}{4} - \frac{1}{2}x_1\}], \\ \min\{1 - x_1, \frac{3}{4} - \frac{1}{2}x_1\}, & \text{if } x_2 \geq \min\{1 - x_1, \frac{3}{4} - \frac{1}{2}x_1\}. \end{cases}$$

However, since we are only interested in $x \in X$, the above formula simplify to

$$\begin{aligned} \hat{y}_\alpha^1(x) &= \begin{cases} \frac{1}{\alpha} + x_1, & \text{if } \frac{1}{\alpha} + x_1 \in [0, \min\{1 - x_2, \frac{3}{2} - 2x_2\}], \\ \min\{1 - x_2, \frac{3}{2} - 2x_2\}, & \text{if } \frac{1}{\alpha} + x_1 \geq \min\{1 - x_2, \frac{3}{2} - 2x_2\}, \end{cases} \\ &= \min\{\frac{1}{\alpha} + x_1, 1 - x_2, \frac{3}{2} - 2x_2\} \end{aligned}$$

and

$$\hat{y}_\alpha^2(x) = x_2,$$

respectively. Now it is easy to see that the corresponding mapping

$$\hat{V}_\alpha(x) = -x_1 - \left[-\hat{y}^1_\alpha(x) + \frac{\alpha}{2}(x_1 - \hat{y}^1_\alpha(x))^2 \right]$$

is not everywhere differentiable on the feasible set X.

The nondifferentiability of the mapping $\hat{V}_\alpha$ is a major disadvantage if one wants to apply suitable optimization methods to the corresponding reformulation (2.7). The very recent paper [23] considers some further properties of the function $\hat{V}_\alpha$ and a related reformulation approach. In the following section, however, we describe a modification of the current approach which results into a smooth optimization reformulation of the GNEP.

The situation is much more favourable if we specialize our results to the standard NEP. Then it can be shown that the mapping $\hat{V}_\alpha$ is continuously differentiable provided all cost functions θ_ν are smooth. This follows from the observation given in Remark 2.2.7 below.

2.2 A Smooth Constrained Optimization Reformulation

In this section, we modify the idea of the previous one and obtain another constrained optimization reformulation of the GNEP which has significantly different properties than the reformulation discussed in Section 2.1. In particular, the reformulation to be given here is smooth. However, it does not give a complete reformulation of all solutions of the GNEP, but it provides a characterization of the normalized Nash equilibria, which were defined in the introduction. The normalized Nash equilibrium can also be defined through the Nikaido-Isoda function instead of the variational inequality (1.5), thereby avoiding the assumption of differentiability of the cost functions.

Definition 2.2.1 *A vector* $x^* \in X$ *is a called* normalized Nash equilibrium *(NoE) of the GNEP, if* $\sup_{y \in X} \Psi(x^*, y) = 0$ *holds, where* Ψ *denotes the Nikaido-Isoda-function from* (2.1).

The above definition of a normalized Nash equilibrium corresponds to one given in, e.g., [37, 98]. It is slightly different from the original definition of a normalized equilibrium given in [92]. This and other features of the normalized Nash equilibrium were discussed in the introduction in Section 1.3.

We next state a simple property of the Nikaido-Isoda-function which follows immediately from the fact that the cost functions $\theta_\nu(x) = \theta_\nu(x^\nu, x^{-\nu})$ are convex with respect to x^ν.

Lemma 2.2.2 *For any given $x \in X$, the Nikaido-Isoda-function $\Psi(x,y)$ is concave in $y \in X$.*

In order to derive a smooth reformulation of the GNEP, our basic tool is, once again, the regularized Nikaido-Isoda-function $\Psi_\alpha(x,y)$ from (2.5). Based on this mapping, we define

$$\begin{aligned} V_\alpha(x) &:= \max_{y \in X} \Psi_\alpha(x,y) \\ &= \max_{y \in X} \sum_{\nu=1}^{N} \left[\theta_\nu(x^\nu, x^{-\nu}) - \theta_\nu(y^\nu, x^{-\nu}) - \frac{\alpha}{2} \|x^\nu - y^\nu\|^2 \right] \\ &= \max_{y \in X} \left[\Psi(x,y) - \frac{\alpha}{2} \|x - y\|^2 \right]. \end{aligned} \tag{2.8}$$

Note that, due to Lemma 2.2.2, given an arbitrary $x \in X$, we take the maximum of a uniformly concave function in y, hence $V_\alpha(x)$ is well-defined. Comparing the definition of V_α with the one of $\hat{V}_\alpha$ in (2.6), we see that the only difference is that the maximum is taken over all $y \in X$ instead of all $y \in \Omega(x)$.

This minor change has a number of important consequences. We first state the counterpart of Theorem 2.1.2 for the mapping V_α.

Theorem 2.2.3 *The regularized function V_α has the following properties:*

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

(b) x^* *is a normalized Nash equilibrium if and only if* $x^* \in X$ *and* $V_\alpha(x^*) = 0$.

(c) For every $x \in X$, there exists a unique maximizer $y_\alpha(x)$ such that

$$\operatorname{argmax} \left[\Psi(x,y) - \frac{\alpha}{2} \|x - y\|^2 \right] = y_\alpha(x), \tag{2.9}$$

and $y_\alpha(x)$ is continuous in x.

Proof. (a) For any $x \in X$, we have $V_\alpha(x) = \max_{y \in X} \Psi_\alpha(x,y) \geq \Psi_\alpha(x,x) = 0$.

(b) First let x^* be a normalized Nash equilibrium. Then $x^* \in X$ and $\sup_{y \in X} \Psi(x^*,y) \leq 0$. Hence $\Psi(x^*,y) \leq 0$ for all $y \in X$. Since

$$\Psi_\alpha(x^*,y) = \Psi(x^*,y) - \frac{\alpha}{2} \|x^* - y\|^2 \leq \Psi(x^*,y) \leq 0 \quad \forall y \in X,$$

it follows that $V_\alpha(x^*) = \max_{y \in X} \Psi_\alpha(x^*,y) \leq 0$. Together with statement (a), this implies $V_\alpha(x^*) = 0$.

Conversely, let $x^* \in X$ be such that $V_\alpha(x^*) = 0$. Then

$$\Psi_\alpha(x^*, y) \leq 0 \quad \forall y \in X. \tag{2.10}$$

Assume there is a vector $\hat{y} \in X$ such that $\Psi(x^*, \hat{y}) > 0$. Then $\lambda x^* + (1-\lambda)\hat{y} \in X$ for all $\lambda \in (0, 1)$, and Lemma 2.2.2 implies

$$\Psi(x^*, \lambda x^* + (1-\lambda)\hat{y}) \geq \lambda\Psi(x^*, x^*) + (1-\lambda)\Psi(x^*, \hat{y}) = (1-\lambda)\Psi(x^*, \hat{y}) > 0 \quad \forall \lambda \in (0, 1).$$

Therefore, we obtain

$$\begin{aligned}
\Psi_\alpha(x^*, \lambda x^* + (1-\lambda)\hat{y}) &= \Psi(x^*, \lambda x^* + (1-\lambda)\hat{y}) - \frac{\alpha}{2}\|x^* - \lambda x^* - (1-\lambda)\hat{y}\|^2 \\
&= \Psi(x^*, \lambda x^* + (1-\lambda)\hat{y}) - \frac{\alpha}{2}(1-\lambda)^2\|x^* - \hat{y}\|^2 \\
&\geq (1-\lambda)\Psi(x^*, \hat{y}) - \frac{\alpha}{2}(1-\lambda)^2\|x^* - \hat{y}\|^2 \\
&> 0
\end{aligned}$$

for all $\lambda \in (0, 1)$ sufficiently close to 1. This, however, is a contradiction to (2.10).

(c) In view of Lemma 2.2.2, the mapping $y \mapsto \Psi(x, y) - \frac{\alpha}{2}\|x - y\|^2$ is strongly concave (uniformly in x). Hence statement (c) is a consequence of standard sensitivity results, see, for example, [53, Corollaries 8.1 and 9.1]. □

Theorem 2.2.3 shows that we can characterize the normalized Nash equilibria of a GNEP as the global minima of the constrained optimization problem

$$\min V_\alpha(x) \quad \text{s.t.} \quad x \in X. \tag{2.11}$$

In contrast to the corresponding reformulation in (2.7), we do not get a reformulation of all generalized Nash equilibria.

We next state the counterpart of Proposition 2.1.3. Its proof is omitted here since it is essentially the same as the one for Proposition 2.1.3 (using Theorem 2.2.3 instead of Theorem 2.1.2).

Proposition 2.2.4 *Let $y_\alpha(x)$ be the vector defined in Theorem 2.2.3 (c) as the unique maximizer in the definition of the regularized function $V_\alpha(x)$, cf.* (2.6). *Then x^* is a normalized Nash equilibrium of GNEP if and only if x^* is a fixed point of the mapping $x \mapsto y_\alpha(x)$.*

Now we come back to the function V_α and the optimization problem 2.11. Our aim is to show that the regularized function V_α is continuously differentiable, provided that the cost functions θ_ν are continuously differentiable for each player $\nu = 1, \ldots, N$. Based on this result, further properties of the optimization problem 2.11, such as a stationary point result, will be derived in the next chapter.

Theorem 2.2.5 *Suppose that the cost functions θ_ν are continuously differentiable for each player $\nu = 1, \dots, N$. Then the regularized function V_α is continuously differentiable for every $x \in X$, and its gradient is given by*

$$\begin{aligned}\nabla V_\alpha(x) &= \nabla_x \Psi_\alpha(x,y)\big|_{y=y_\alpha(x)} \\ &= \sum_{\nu=1}^{N} \left[\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x), x^{-\nu})\right] \\ &\quad + \begin{pmatrix} \nabla_{x^1}\theta_1(y_\alpha^1(x), x^{-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y_\alpha^N(x), x^{-N}) \end{pmatrix} - \alpha(x - y_\alpha(x)),\end{aligned}$$

where $y_\alpha(x)$ denotes the unique maximizer from Theorem 2.2.3 (c) associated to the given vector x.

Proof. We first recall that the regularized function V_α can be represented as in the last line of (2.6), and that the mapping

$$y \mapsto \Psi_\alpha(x,y) = \Psi(x,y) - \frac{\alpha}{2}\|x-y\|^2$$

is strongly concave for any fixed x in view of Lemma 2.2.2. Hence it follows from Danskin's Theorem (see, for example, [28]) that V_α is differentiable with gradient $\nabla V_\alpha(x) = \nabla_x \Psi_\alpha(x,y)\big|_{y=y_\alpha(x)}$. Using the definition of the mapping Ψ_α, an elementary calculation shows that

$$\begin{aligned}\nabla_x \Psi_\alpha(x,y) &= \sum_{\nu=1}^{N} \left[\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y^\nu, x^{-\nu})\right] \\ &\quad + \begin{pmatrix} \nabla_{x^1}\theta_1(y^1, x^{-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y^N, x^{-N}) \end{pmatrix} - \alpha(x - y),\end{aligned}$$

Inserting $y = y_\alpha(x)$ then gives the desired formula for the gradient of V_α. Since all cost functions θ_ν are continuously differentiable, and since $y_\alpha(x)$ is also a continuous mapping of x in view of Theorem 2.2.3, we finally get that the gradient $\nabla V_\alpha(x) = \nabla_x \Psi_\alpha(x,y)\big|_{y=y_\alpha(x)}$ is continuous, i.e., the regularized function V_α is continuously differentiable. □

The following note shows that no regularization of the Nikaido-Isoda-function is necessary if the cost functions θ_ν have some stronger properties than those mentioned so far.

Remark 2.2.6 Suppose that the functions $\theta_\nu(x) = \theta_\nu(x^\nu, x^{-\nu})$ are strongly convex in x^ν (for any given $x^{-\nu}$). Then the mapping

$$V(x) := \max_{y \in X} \Psi(x, y)$$

is well-defined and gives a reformulation of the GNEP as a smooth optimization problem

$$\min V(x) \quad \text{s.t.} \quad x \in X.$$

This means that there is no need to regularize the function Ψ for strongly convex cost functions. The proof of the above statement follows by simple inspection of the proofs given in this section. Note, however, that the unconstrained optimization reformulation to be presented in Section 2.3 needs a regularized Nikaido-Isoda-function even in the case of strongly convex functions θ_ν.

We close this section with a simple note on the application of our results to the standard Nash equilibrium problem.

Remark 2.2.7 Suppose that the nonempty, closed, and convex set $X \subseteq \mathbb{R}^n$ has a Cartesian product structure, that is,

$$X = X_1 \times X_2 \times \cdots \times X_N$$

with $X_\nu \subseteq \mathbb{R}^{n_\nu}$ fixed. Then $\Omega(x) = X$ for all x, and the GNEP reduces to the standard NEP. Moreover, it follows that

$$\hat{V}_\alpha(x) = \max_{y \in \Omega(x)} \Psi_\alpha(x, y) = \max_{y \in X} \Psi_\alpha(x, y) = V_\alpha(x)$$

for all $x \in X$, i.e., the two functions $\hat{V}_\alpha$ from the previous section and V_α from the current section coincide. In particular, the mapping $\hat{V}_\alpha$ is therefore also continuously differentiable when applied to a standard NEP.

2.3 An Unconstrained Smooth Optimization Reformulation

Here we use the regularized Nikaido-Isoda-function in order to obtain an unconstrained optimization reformulation of the GNEP. To this end, let $0 < \alpha < \beta$ be two given parameters, let

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

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

be the associated regularized Nikaido-Isoda functions, and let

$$\begin{aligned} V_\alpha(x) &:= \max_{y\in X} \Psi_\alpha(x,y) = \Psi_\alpha(x, y_\alpha(x)),\\ V_\beta(x) &:= \max_{y\in X} \Psi_\beta(x,y) = \Psi_\beta(x, y_\beta(x)) \end{aligned}$$

be the corresponding regularized value functions. Formally, these functions are defined only for $x \in X$ in the previous section. However, it is easy to see that they can be defined for any $x \in \mathbb{R}^n$.

Similar to the way the D-gap function was derived from the regularized gap function in the context of variational inequalities, see [82, 99], we then define

$$V_{\alpha\beta}(x) := V_\alpha(x) - V_\beta(x), \quad x \in \mathbb{R}^n. \tag{2.12}$$

In order to show that this difference of two regularized Nikaido-Isoda-functions gives an unconstrained optimization reformulation of the GNEP, we first state the following result.

Lemma 2.3.1 *The inequality*

$$\frac{\beta-\alpha}{2}\|x - y_\beta(x)\|^2 \le V_{\alpha\beta}(x) \le \frac{\beta-\alpha}{2}\|x - y_\alpha(x)\|^2 \tag{2.13}$$

holds for all $x \in \mathbb{R}^n$.

Proof. By definition, we have for any $x \in \mathbb{R}^n$

$$V_\beta(x) = \Psi_\beta(x, y_\beta(x)) = \max_{y\in X} \Psi_\beta(x,y)$$

and, therefore

$$V_\beta(x) \ge \Psi_\beta(x, y_\alpha(x)).$$

This implies

$$\begin{aligned} V_{\alpha\beta}(x) &= V_\alpha(x) - V_\beta(x)\\ &\le \Psi_\alpha(x, y_\alpha(x)) - \Psi_\beta(x, y_\alpha(x))\\ &= \frac{\beta-\alpha}{2}\sum_{\nu=1}^{N} \|x^\nu - y_\alpha^\nu(x)\|^2\\ &= \frac{\beta-\alpha}{2}\|x - y_\alpha(x)\|^2 \end{aligned}$$

for all $x \in \mathbb{R}^n$. This proves the right-hand inequality in (2.13). The other inequality can be verified in a similar way. □

Note that, similar to an observation in [55], Lemma 2.3.1 immediately implies that the level sets of the function $V_{\alpha\beta}$ are compact for compact sets X. This observation guarantees that any sequence $\{x^k\}$ generated by a descent method for $V_{\alpha\beta}$ will remain bounded and, therefore, has at least one accumulation point.

As another consequence of Lemma 2.3.1, we obtain the following result.

Theorem 2.3.2 *The following statements about the function $V_{\alpha\beta}$ hold:*

(a) $V_{\alpha\beta}(x) \geq 0$ *for all* $x \in \mathbb{R}^n$.

(b) x^* *is a normalized Nash equilibrium of the GNEP if and only if* x^* *is a global minimum of* $V_{\alpha\beta}$ *with* $V_{\alpha\beta}(x^*) = 0$.

Proof. (a) Using Proposition 2.2.4, we have

$$V_{\alpha\beta}(x) \geq \frac{\beta - \alpha}{2} \|x - y_\beta(x)\|^2 \geq 0$$

for all $x \in \mathbb{R}^n$.

(b) First assume that x^* is a normalized Nash equilibrium. Then Proposition 2.2.4 implies $x^* = y_\alpha(x^*)$ and $x^* = y_\beta(x^*)$. Hence (2.13) immediately gives $V_{\alpha\beta}(x^*) = 0$.

Conversely, let x^* be such that $V_{\alpha\beta}(x^*) = 0$. Then (2.13) implies $x^* = y_\beta(x^*)$. Hence x^* solves the GNEP in view of Proposition 2.2.4. □

Theorem 2.3.2 shows that the normalized Nash equilibria of GNEP are precisely the global minima of the *unconstrained* optimization problem

$$\min \; V_{\alpha\beta}(x), \quad x \in \mathbb{R}^n. \tag{2.14}$$

We next note that this is a smooth problem. To this end, however, we need to assume, for the remainder of this section, that all cost functions θ_ν are continuously differentiable. Then we have the following result.

Theorem 2.3.3 *The function $V_{\alpha\beta}$ is continuously differentiable for every $x \in \mathbb{R}^n$, and its gradient is given by*

$$\nabla V_{\alpha\beta}(x) \;=\; \sum_{\nu=1}^{N} \left[\nabla\theta_\nu(y_\beta^\nu(x), x^{-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x), x^{-\nu})\right]$$

$$+\begin{pmatrix} \nabla_{x^1}\theta_1(y^1_\alpha(x), x^{-1}) - \nabla_{x^1}\theta_1(y^1_\beta(x), x^{-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y^N_\alpha(x), x^{-N}) - \nabla_{x^N}\theta_N(y^N_\beta(x), x^{-N}) \end{pmatrix}$$
$$-\alpha(x - y_\alpha(x)) + \beta(x - y_\beta(x)).$$

Proof. First recall that $V_\alpha(x)$ and $V_\beta(x)$ are defined for all $x \in \mathbb{R}^n$. Then observe that the formula for the gradients of these two functions, as given in Theorem 2.2.5 for $x \in X$, remain true for all $x \in \mathbb{R}^n$. Since we have $\nabla V_{\alpha\beta}(x) = \nabla V_\alpha(x) - \nabla V_\beta(x)$, the statement follows from Theorem 2.2.5. □

Chapter 3

Descent Methods

In this chapter we consider the smooth constrained optimization reformulation and the smooth unconstrained optimization reformulation from the preceding chapter again. The focus is on properties of the functions V_α and $V_{\alpha\beta}$, respectively. In the first section we derive conditions that imply convexity of the function V_α, as well as stationary point results for both the constrained and unconstrained optimization problems (2.11) and (2.14).

The next section deals with the relaxation method proposed in [98], which is one of the most popular methods for computing normalized Nash equilibria. We show that it is possible to interprete the relaxation method, which is essentially a fixed point iteration, as a feasible descent method for the constrained optimization problem (2.11). This viewpoint, in particular the application of a line search, leads to improved theoretical and numerical results.

Finally we extend the relaxation method to the non-differentiable case, that is, we do not require the assumption that the cost functions are smooth. Proving convergence for this nonsmooth method however requires stronger assumptions regarding convexity than for the differentiable case. Throughout this chapter, we assume that Assumption 1.2.2 holds.

3.1 Properties of the Optimization Reformulation

Let Ψ_α, V_α, and y_α be defined by (2.5), (2.8), and (2.9), respectively. Theorem 2.2.3 shows that x^* is a normalized Nash equilibrium if and only if it is a global minimum of the constrained minimization problem

$$\min\ V_\alpha(x) \quad \text{s.t.} \quad x \in X \tag{3.1}$$

with optimal function value $V_\alpha(x^*) = 0$. Moreover, from Theorem 2.2.5 it follows that V_α is differentiable, if the cost functions θ_ν are all differentiable.

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

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

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

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

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 3.1.1 *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 $\lambda \in (0, 1)$

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

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

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

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

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 3.1.1, 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 cost functions θ_ν. However, for the case of quadratic cost functions, we have the following sufficient condition.

Proposition 3.1.2 *Consider the case where the cost 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

$$
B := \begin{pmatrix} \frac{1}{2}A_{11} & A_{12} & \cdots & A_{1N} \\ A_{21} & \frac{1}{2}A_{22} & \cdots & A_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \cdots & \frac{1}{2}A_{NN} \end{pmatrix}, \tag{3.2}
$$

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 3.1.1. 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 3.1.1. □

Note that the previous result also holds if the cost 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 3.1.2 are tight.

Example 3.1.3 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

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

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 3.1.2 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 $\alpha > 1$. ◇

The previous results guarantee that (3.1) 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). Next we 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 (3.1).

Assumption 3.1.4

(a) *The cost functions θ_ν are continuously differentiable.*

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

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

holds.

Note that the smoothness assumption from Assumption 3.1.4 (a) is necessary, in particular, to formulate part (b). This Assumption 3.1.4 (b) is crucial for the development and analysis of the following descent method. On the one hand, it can be shown that any stationary point of the optimization problem (3.1) is a solution of the GNEP provided that Assumption 3.1.4 holds, see below. On the other hand, it implies that the search direction used in the relaxation method from the next section is a (feasible) descent direction for the value function V_α, see Lemma 3.2.2 below.

We postpone a discussion of Assumption 3.1.4 to the end of this section. The following result first shows that Assumption 3.1.4 provides a sufficient condition for a stationary point to be a global minimum and, therefore, to be a normalized Nash equilibrium.

Theorem 3.1.5 *Let $x^* \in X$ be a stationary point of* (2.11) *in the sense that*

$$\nabla V_\alpha(x^*)^T (x - x^*) \geq 0 \quad \forall x \in X. \tag{3.3}$$

If Assumption 3.1.4 holds at $x = x^*$, then x^* is a normalized Nash equilibrium of the GNEP.

Proof. Using (3.3) and the representation of the gradient $\nabla V_\alpha(x^*)$ from Theorem 2.2.5, we obtain

$$\begin{aligned}
0 &\leq \nabla V_\alpha(x^*)^T (x - x^*) \\
&= \sum_{\nu=1}^{N} \left[\nabla\theta_\nu(x^{*,\nu}, x^{*,-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})\right]^T (x - x^*) \\
&\quad + \sum_{\nu=1}^{N} \nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})^T (x^\nu - x^{*,\nu}) - \alpha(x^* - y_\alpha(x^*))^T (x - x^*) \\
&= \sum_{\nu=1}^{N} \left[\nabla\theta_\nu(x^{*,\nu}, x^{*,-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})\right]^T (x - x^*)
\end{aligned}$$

$$+\sum_{\nu=1}^{N}\left[\nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu}) - \alpha(x^{*,\nu} - y_\alpha^\nu(x^*))\right]^T(x^\nu - x^{*,\nu})$$

for all $x \in X$. Choosing $x = y_\alpha(x^*)$, we therefore get

$$\begin{array}{rcl} 0 & \leq & \sum_{\nu=1}^{N}\left[\nabla\theta_\nu(x^{*,\nu}, x^{*,-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})\right]^T(y_\alpha(x^*) - x^*) \\ & & + \sum_{\nu=1}^{N}\left[\nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu}) - \alpha(x^{*,\nu} - y_\alpha^\nu(x^*))\right]^T(y_\alpha^\nu(x^*) - x^{*,\nu}). \end{array} \tag{3.4}$$

Now recall that $y_\alpha(x^*)$ is the unique solution of the optimization problem

$$\max \sum_{\nu=1}^{N}\left[\theta_\nu(x^{*,\nu}, x^{*,-\nu}) - \theta_\nu(y^\nu, x^{*,-\nu}) - \frac{\alpha}{2}\|x^{*,\nu} - y^\nu\|^2\right] \quad \text{s.t.} \quad y \in X.$$

Consequently, $y_\alpha(x^*)$ satisfies the corresponding optimality conditions

$$\begin{pmatrix} \nabla_{x^1}\theta_1(y_\alpha^1(x^*), x^{*,-1}) - \alpha(x^{*,1} - y_\alpha^1(x^*)) \\ \vdots \\ \nabla_{x^N}\theta_N(y_\alpha^N(x^*), x^{*,-N}) - \alpha(x^{*,N} - y_\alpha^N(x^*)) \end{pmatrix}^T (z - y_\alpha(x^*)) \geq 0 \quad \forall z \in X.$$

Using $z = x^*$, we therefore obtain

$$\sum_{\nu=1}^{N}\left[\nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu}) - \alpha(x^{*,\nu} - y_\alpha^\nu(x^*))\right]^T(x^{*,\nu} - y_\alpha^\nu(x^*)) \geq 0.$$

Taking this into account, we get

$$0 \leq \sum_{\nu=1}^{N}\left[\nabla\theta_\nu(x^{*,\nu}, x^{*,-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})\right]^T(y_\alpha(x^*) - x^*) \tag{3.5}$$

from (3.4). Now assume that $x^* \neq y_\alpha(x^*)$. Then (3.5) and Assumption 3.1.4 together imply $0 < 0$. This contradiction shows that $x^* = y_\alpha(x^*)$. Hence x^* is a normalized Nash equilibrium of the GNEP because of Theorem 2.2.3 c). □

The rest of this section is devoted to a discussion of Assumption 3.1.4 (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 3.1.4 holds for an arbitrary $\alpha > 0$. The main criterion is given in the following result.

Theorem 3.1.6 *Let x^* be a normalized Nash equilibrium and assume that the cost 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 3.1.4 holds for all $x \in N(x^*)$.*

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

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

Hence we get

$$\begin{aligned}
&\sum_{\nu=1}^N [\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y^\nu, x^{-\nu})] \\
&= \sum_{\nu=1}^N \Big[\Big(\int_0^1 \nabla^2_{xx^\nu}\theta_\nu(x^\nu + \tau(y^\nu - x^\nu), x^{-\nu})\, d\tau\Big)(x^\nu - y^\nu)\Big] \qquad (3.6)\\
&= \Big(\int_0^1 \nabla^2_{xx^1}\theta_1(x^1 + \tau(y^1 - x^1), x^{-1})\, d\tau, \dots, \int_0^1 \nabla^2_{xx^N}\theta_N(x^N + \tau(y^N - x^N), x^{-N})\, d\tau\Big)(x - y) \\
&= \Big(\int_0^1 \Big[\nabla^2_{xx^1}\theta_1(x^1 + \tau(y^1 - x^1), x^{-1}), \dots, \nabla^2_{xx^N}\theta_N(x^N + \tau(y^N - x^N), x^{-N})\Big] d\tau\Big)(x - y) \\
&= \int_0^1 \Big[\nabla^2_{xx^1}\theta_1(x^1 + \tau(y^1 - x^1), x^{-1}), \dots, \nabla^2_{xx^N}\theta_N(x^N + \tau(y^N - x^N), x^{-N})\Big](x - y) d\tau.
\end{aligned}$$

Since the functions θ_ν are twice continuously differentiable, and since x^* is a fix point of $y_\alpha(\cdot)$ in view of Theorem 2.2.3 c), the assumption that A is positive definite implies that there exists a neighbourhood $N(x^*)$ such that the slightly perturbed matrix

$$\Big(\nabla^2_{xx^1}\theta_1(x^1 + \tau(y_\alpha^1(x) - x^1), x^{-1}), \dots, \nabla^2_{xx^N}\theta_N(x^N + \tau(y_\alpha^N(x) - x^N), x^{-N})\Big)$$

is positive definite for all $x \in N(x^*)$ and $\tau \in [0, 1]$. Together with (3.6) this implies that Assumption 3.1.4 holds for all $x \in N(x^*)$ with $x \neq y_\alpha(x)$. □

The following two corollaries are consequences of Theorem 3.1.6 and provide some simplified sufficient conditions for Assumption 3.1.4 to be satisfied.

Corollary 3.1.7 *Consider the case where the cost functions θ_ν are quadratic, say*

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

for $\nu = 1, \dots, N$. Suppose that the matrix $A = (A_{\nu\mu})_{\nu,\mu=1}^N$ is positive definite. Then Assumption 3.1.4 is satisfied at an arbitrary point $x \in X$ with $x \neq y_\alpha(x)$.

Proof. The statement follows immediately from Theorem 3.1.6 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 (3.2). In fact, B being positive definite implies that the diagonal block matrix $D := \frac{1}{2}\text{diag}(A_{11}, \dots, A_{NN})$ 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 3.1.8 *Suppose that the cost functions θ_ν are twice continuously differentiable and that the matrix $B(x, y) = (B_{\mu\nu}(x, y))_{\mu,\nu=1}^N$ with*

$$B_{\mu\nu}(x, y) = \nabla^2_{x^\mu x^\nu}\theta_\nu(y^\nu, x^{-\nu}) \tag{3.7}$$

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

$$B(x, y) = -\nabla^2_{xy}\Psi_\alpha(x, y) - \nabla^2_{yy}\Psi_\alpha(x, y) \tag{3.8}$$

are positive definite for all $x, y \in X$. Then Assumption 3.1.4 holds for all $x \in X$ with $x \neq y_\alpha(x)$.

Proof. By taking a look at the proof of Theorem 3.1.6, we immediately see that the assumed positive definiteness of the matrices $B(x, y)$ with the block components given by (3.7) implies that Assumption 3.1.4 holds.

Hence we only have to show that the mapping B has the alternative representation given in (3.8). 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., [47]. □

The following example shows that the condition given in (3.7) is not sufficient for the convexity of the function V_α. In particular, it follows that Assumption 3.1.4 guarantees that stationary points are global minima for a class of nonconvex problems.

Example 3.1.9 Consider a two-person game where each player controls a single variable, and where the corresponding optimization problems are given by

$\min_{x_1}$	$\frac{1}{2}x_1^2 + \frac{3}{4}x_1 x_2$	$\min_{x_2}$	$\frac{1}{2}x_2^2 + \frac{3}{4}x_1 x_2$
s.t.	$x_1 \geq 1$	s.t.	$x_2 \geq 1$.

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 (3.7) 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 3.1.4 holds for all $x \in X$. ◇

In the remainder of this section we consider the unconstrained optimization reformulation from chapter 2.3 again. We know that (2.14) is a smooth unconstrained optimization reformulation of the GNEP. Thus, if we want to develop a numerical method based on this reformulation, we need to compute the global minimum of $V_{\alpha\beta}$. However, standard optimization software is usually only able to find a stationary point, therefore we next want to give a result saying that such a stationary point is already a normalized Nash equilibrium under certain conditions. To this end, we first state the following preliminary result.

Lemma 3.1.10 *The inequality*

$$\sum_{\nu=1}^{N} \left[\nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x), x^{-\nu}) - \nabla_{x^\nu}\theta_\nu(y_\beta^\nu(x), x^{-\nu}) - \alpha(x^\nu - y_\alpha^\nu(x)) + \beta(x^\nu - y_\beta^\nu(x))\right]^T (y_\beta^\nu(x) - y_\alpha^\nu(x)) \geq 0$$

holds for any $x \in \mathbb{R}^n$.

Proof. As noted in the proof of Theorem 3.1.5, $y_\alpha^\nu(x)$ satisfies the optimality condition

$$\sum_{\nu=1}^{N} \left[\nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x), x^{-\nu}) - \alpha(x^\nu - y_\alpha^\nu(x))\right]^T (z^\nu - y_\alpha^\nu(x)) \geq 0 \quad \forall z \in X.$$

In a similar way, it follows that $y_\beta^\nu(x)$ satisfies

$$\sum_{\nu=1}^{N} \left[\nabla_{x^\nu}\theta_\nu(y_\beta^\nu(x), x^{-\nu}) - \beta(x^\nu - y_\beta^\nu(x))\right]^T (z^\nu - y_\beta^\nu(x)) \geq 0 \quad \forall z \in X.$$

Using $z = y_\beta(x)$ in the first inequality and $z = y_\alpha(x)$ in the second inequality, we get

$$\sum_{\nu=1}^{N} \left[\nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x), x^{-\nu}) - \alpha(x^\nu - y_\alpha^\nu(x))\right]^T (y_\beta^\nu(x) - y_\alpha^\nu(x)) \geq 0$$

and

$$\sum_{\nu=1}^{N} \left[\nabla_{x^\nu}\theta_\nu(y_\beta^\nu(x), x^{-\nu}) - \beta(x^\nu - y_\beta^\nu(x))\right]^T (y_\alpha^\nu(x) - y_\beta^\nu(x)) \geq 0,$$

respectively. Adding these two inequalities gives the desired result. □

In order to state a result that a stationary point is, automatically, a global minimum of $V_{\alpha\beta}$, we need a certain condition which is quite similar to the one stated in Assumption 3.1.4.

Assumption 3.1.11 *For given $x \in \mathbb{R}^n$ with $y_\alpha(x) \neq y_\beta(x)$, the inequality*

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

holds.

Using Assumption 3.1.11, we are now able to state the following result.

Theorem 3.1.12 *Let x^* be a stationary point of $V_{\alpha\beta}$. If Assumption 3.1.11 holds at $x = x^*$, then x^* is a normalized Nash equilibrium of the GNEP.*

Proof. Since x^* is a stationary point of $V_{\alpha\beta}$, we obtain from Theorem 2.3.3

$$\begin{aligned} 0 &= \nabla V_{\alpha\beta}(x^*) \\ &= \sum_{\nu=1}^{N} \left[\nabla\theta_\nu(y_\beta^\nu(x^*), x^{*,-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})\right] \\ &\quad + \begin{pmatrix} \nabla_{x^1}\theta_1(y_\alpha^1(x^*), x^{*,-1}) - \nabla_{x^1}\theta_1(y_\beta^1(x^*), x^{*,-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y_\alpha^N(x^*), x^{*,-N}) - \nabla_{x^N}\theta_N(y_\beta^N(x^*), x^{*,-N}) \end{pmatrix} \\ &\quad -\alpha(x^* - y_\alpha(x^*)) + \beta(x^* - y_\beta(x^*)). \end{aligned} \tag{3.9}$$

Multiplication with $(y_\beta(x^*) - y_\alpha(x^*))^T$ and using Lemma 3.1.10, we therefore get

$$\begin{aligned} 0 = &\sum_{\nu=1}^{N} \left[\nabla\theta_\nu(y_\beta^\nu(x^*), x^{*,-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})\right]^T (y_\beta(x^*) - y_\alpha(x^*)) \\ &+ \sum_{\nu=1}^{N} \left[\nabla_{x^\nu}\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu}) - \nabla_{x^\nu}\theta_\nu(y_\beta^\nu(x^*), x^{*,-\nu})\right. \\ &\qquad \left. -\alpha(x^{*,\nu} - y_\alpha^\nu(x^*)) + \beta(x^{*,\nu} - y_\beta^\nu(x^*))\right]^T (y_\beta^\nu(x^*) - y_\alpha^\nu(x^*)) \end{aligned}$$

$$\geq \sum_{\nu=1}^{N} [\nabla\theta_\nu(y_\beta^\nu(x^*), x^{*,-\nu}) - \nabla\theta_\nu(y_\alpha^\nu(x^*), x^{*,-\nu})](y_\beta(x^*) - y_\alpha(x^*)).$$

Assume that $y_\beta(x^*) - y_\alpha(x^*) \neq 0$. Then the previous chain of inequalities together with Assumption 3.1.11 gives the contradiction $0 > 0$. Hence $y_\alpha(x^*) = y_\beta(x^*)$. But then (3.9) simplifies to $(\beta - \alpha)(x^* - y_\alpha(x^*)) = 0$. Since $\alpha < \beta$, this implies $x^* = y_\alpha(x^*)$. Consequently, x^* is a normalized Nash equilibrium in view of Proposition 2.2.4. □

3.2 A Relaxation Method with Inexact Line Search

The so-called relaxation method is a fixed point iteration based on the result 2.2.4 and computes normalized Nash equilibria. 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 [12, 15, 46, 59, 61]. However, the conditions that guarantee convergence of the relaxation method in [98, 61] are very restrictive. Moreover, the rather general inexact stepsize rule given in [98] leads to more or less heuristic implementations of the relaxation method, whereas the exact stepsize rule from [61] is not really implementable, see the comments below for more details.

Here we 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.

The relaxation methods presented in [98, 61] as well as the one to be discussed in the following find a normalized Nash equilibrium and, therefore, a particular solution of a given GNEP. The relaxation method itself uses the iteration

$$x^{k+1} := x^k + t_k d^k, \quad d^k := y_\alpha(x^k) - x^k, \quad k = 0, 1, 2, \dots \tag{3.10}$$

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 (2.9), the authors of [98] have to add some assumptions which are not necessary in our case, and convergence of the method is guaranteed, if the stepsize $t_k \in (0, 1]$ satisfies 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., [61, 46]. The version of the relaxation method presented in [61] chooses the stepsize t_k by an exact minimization of the one-dimensional mapping

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

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 [98], 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 $x^k + td^k$.

Note that the iteration (3.10) 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 [98, 61] 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.

Throughout this section we suppose that Assumption 3.1.4 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 3.2.1 *(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 (for instance $V_\alpha(x^k) \leq \varepsilon$ *for some* $\varepsilon > 0$*, or* $\|y_\alpha(x^k) - x^k\| < \varepsilon$*).*

(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*

$$V_\alpha(x^k + t_k d^k) \leq V_\alpha(x^k) - \sigma t_k^2 \|d^k\|. \tag{3.11}$$

(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 cost functions θ_ν, cf. Assumption 3.1.4. 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 3.2.1 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 cost 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 3.1.4 used within this section.

Our first aim is to show that Algorithm 3.2.1 is well-defined. To this end, we note that d^k is always a direction of descent for the merit function V_α.

Lemma 3.2.2 *Let $x^k \in X$ be the current iterate and d^k be the vector computed in Step (S.2) of Algorithm 3.2.1. 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 Theorem 2.2.5 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. [46]), we then obtain

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

The first term of this equality is negative by Assumption 3.1.4, 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 3.1.4 was crucial in proving the descent property. Based on the previous result, we are now in the position to show that Algorithm 3.2.1 is well–defined.

Lemma 3.2.3 *Algorithm 3.2.1 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 3.2.1 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 3.2.2 that $\nabla V_\alpha(x^k)^T d^k < 0$ since x^k is not a solution of our GNEP (otherwise the algorithm would have stopped in (S.1)). This contradiction completes the proof. □

We next give a global convergence result for Algorithm 3.2.1.

Theorem 3.2.4 *Every accumulation point of a sequence generated by Algorithm 3.2.1 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 2.2.4, 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 (e.g., by $V_\alpha(x^*)$), 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

$$\lim_{k \in K} t_k = 0. \tag{3.12}$$

Let $l_k \in \mathbb{N}$ be the unique exponent such that $t_k = \beta^{l_k}$ in (S.3) of Algorithm 3.2.1. In view of (3.12), 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 3.2.1. 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 3.2.2 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 3.2.1. This observation is formally stated in the following remark.

Remark 3.2.5 It is not difficult to see that all our previous results remain true if we replace the line search rule (3.11) in Algorithm 3.2.1 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 (3.11) 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 3.2.1. It turns out that one should not expect local quadratic convergence of the iteration $x^{k+1} = y_\alpha(x^k)$, even under very favourable assumptions. This is illustrated by the following simple example.

Example 3.2.6 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

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

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 (2.8) 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 $\alpha > 0$. $\diamond$

3.3 A Nonsmooth Descent Method

In this section, we consider Algorithm 3.2.1 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 the following slightly stronger assumption that we assume to hold throughout this section.

Assumption 3.3.1 *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 3.1.1 and its proof, a direct consequence of Assumption 3.3.1 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 3.2.2.

Lemma 3.3.2 *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

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

$$
\begin{aligned}
&\leq t\Psi_\alpha(y_\alpha(x), y_\alpha(x+td)) + (1-t)\Psi_\alpha(x, y_\alpha(x+td)) \qquad (3.13)\\
&\leq t\Psi_\alpha(y_\alpha(x), y_\alpha(x+td)) + (1-t)\Psi_\alpha(x, y_\alpha(x))\\
&= t\Psi_\alpha(y_\alpha(x), y_\alpha(x+td)) + (1-t)V_\alpha(x)\\
&= t[\Psi_\alpha(y_\alpha(x), y_\alpha(x+td)) - V_\alpha(x)] + V_\alpha(x)
\end{aligned}
$$

or, equivalently,

$$\frac{V_\alpha(x+td) - V_\alpha(x)}{t} \leq \Psi_\alpha(y_\alpha(x), y_\alpha(x+td)) - V_\alpha(x). \tag{3.14}$$

Since the function y_α is continuous by Theorem 2.2.3 c), we have $y_\alpha(x+td) \to y_\alpha(x)$ for $t \to 0$ and, therefore, $\Psi_\alpha(y_\alpha(x), y_\alpha(x+td)) \to \Psi_\alpha(y_\alpha(x), y_\alpha(x)) = 0$. Hence it follows from (3.14) 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

$$\frac{V_\alpha(x+td) - V_\alpha(x)}{t} \leq -\varepsilon \quad \forall t \in (0, \bar{t}]. \tag{3.15}$$

This completes the proof. □

We next show that Algorithm 3.3.1 is well-defined under Assumption 3.3.1.

Lemma 3.3.3 *Algorithm 3.2.1 is well–defined and generates a sequence $\{x^k\}$ belonging to the feasible set X.*

Proof. Similar to the proof of Lemma 3.2.3, 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

$$V'_\alpha(x^k; d^k) \geq 0. \tag{3.16}$$

On the other hand, we immediately obtain from (3.15) that $V'_\alpha(x^k; d^k) \leq -\varepsilon$ for some sufficiently small $\varepsilon = \varepsilon(x^k) > 0$, a contradiction to (3.16). □

We now come to the main global convergence result of Algorithm 3.2.1 under Assumption 3.3.1.

Theorem 3.3.4 *Every accumulation point of a sequence generated by Algorithm 3.2.1 is a normalized Nash equilibrium of our GNEP.*

Proof. We try to copy the proof of Theorem 3.2.4. Basically, this is possible since V_α is a convex function, hence we can exploit suitable properties of the convex subdifferential, see, e.g., [50, 90] 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. Theorem 2.2.3 c)) 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 Theorem 2.2.4, we only need to show that $d^* = 0$.

Assume that $d^* \neq 0$. Similar to the proof of Theorem 3.2.4, 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

$$\frac{V_\alpha(x^k + \beta^{l_k-1}d^k) - V_\alpha(x^k)}{\beta^{l_k-1}} > -\sigma\beta^{l_k-1}\|d^k\| \quad \forall\, k \in K. \tag{3.17}$$

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 (3.17) 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 (3.17) 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 (3.17), 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 (3.15) that

$V'_\alpha(x^*; d^*) < 0$. This contradiction shows that $d^* = 0$ and therefore completes the proof. □

Chapter 4

Newton's Method based on an Optimization Reformulation

In chapter 2 we introduced optimization reformulations of the generalized Nash equilibrium problem. Two of these reformulations have a differentiable objective function, namely the constrained optimization reformulation (2.11) and the unconstrained optimization reformulation (2.14). However, the objective functions of the latter optimization reformulations are, in general, not twice differentiable. Here we investigate some further properties of these reformulations and, in particular, show that they are sufficiently smooth so that locally superlinearly convergent Newton-type methods can be applied in order to solve the underlying GNEP.

As in the preceding chapters, let θ_ν, $\nu = 1, \dots, N$ be the cost function and X the joint strategy set. In particular, throughout this chapter, we assume that the set X is represented by inequalities, that is,

$$X = \{x \in \mathbb{R}^n \mid g(x) \leq 0\} \tag{4.1}$$

with some function $g : \mathbb{R}^n \to \mathbb{R}^m$. Additional equality constraints are also allowed, but for notational simplicity, we prefer not to include them explicitly. In many cases, a player ν might have some additional constraints of the form $h^\nu(x^\nu) \leq 0$ depending on his decision variables only. However, these additional constraints may simply be viewed as part of the joint constraints $g(x) \leq 0$, with some of the component functions g_i of g depending on the block component x^ν of x only.

Stronger than Assumptions 1.2.2, we impose the following conditions throughout this chapter.

Assumption 4.0.5

(a) The cost functions θ_ν, $\nu = 1, \dots, N$ are twice continuously differentiable, and convex with respect to the variable x^ν, i.e., the function $\theta_\nu(\cdot, x^{-\nu})$ is convex, uniformly for all $x^{-\nu}$;

(b) *The function g is twice continuously differentiable, its components g_i are convex (in x), and the corresponding strategy space X defined by* (4.1) *is nonempty.*

The smoothness assumptions are natural since our aim is to develop locally fast convergent methods for the solution of GNEPs. Note that Assumption 4.0.5 (b) implies that the strategy space X is nonempty, closed, and convex.

In the first section, we recall some basic facts and recent results from nonsmooth analysis. Then, in the next section, we show that the optimization reformulations from chapter 2 are SC^1 reformulations of the GNEP, i.e., the objective function is continuously differentiable with semismooth gradient. In the last section we consider a Newton-type method based on the unconstrained optimization reformulation of the GNEP.

4.1 Semismooth Functions

In this section, we first recall some basic definitions and results from nonsmooth analysis in this section, and then state some preliminary results that will be used in our subsequent analysis. To this end, let $F : \mathbb{R}^n \to \mathbb{R}^m$ be a locally Lipschitzian mapping. According to Rademacher's theorem (see [86]), it follows that F is almost everywhere differentiable. Let D_F denote the set of all differentiable points of F. Then we call

$$\partial_B F(x) := \{H \in \mathbb{R}^{m\times n} \,\big|\, \exists\{x^k\} \subseteq D_F : x^k \to x, F'(x^k) \to H\}$$

the *B-subdifferential* of F at x. Its convex hull

$$\partial F(x) := \mathrm{conv}\partial_B F(x)$$

is Clarke's *generalized Jacobian* of F at x, see [14]. In case of $m = 1$, we call this set also the *generalized gradient* of F at x which, therefore, is a set of row vectors. Furthermore, we call the set

$$\partial_C F(x) := (\partial F_1(x)^T \times \ldots \times \partial F_m(x)^T)^T$$

the *C-subdifferential* of F at x, i.e., the C-subdifferential is the set of matrices whose ith rows consist of the elements of the generalized gradient of the ith component functions F_i. According to [14, Proposition 2.6.2], the following inclusions hold:

$$\partial_B F(x) \subseteq \partial F(x) \subseteq \partial_C F(x). \tag{4.2}$$

Based on the generalized Jacobian, we next recall the definition of a semismooth function.

Definition 4.1.1 *Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be locally Lipschitz continuous. Then F is called* semismooth *at x if F is directionally differentiable at x and*

$$\|Hd - F'(x; d)\| = o(\|d\|)$$

holds for all $d \to 0$ and all $H \in \partial F(x + d)$.

In the following, we often call a mapping $F : \mathbb{R}^n \to \mathbb{R}^m$ semismooth if it is semismooth at every point $x \in \mathbb{R}^n$. The notion of a semismooth function was originally introduced by Mifflin [68] for functionals, and later extended by Qi and Sun [85] to vector-valued mappings.

Note that there are many different notions of semismooth functions available in the literature, and we would like to give some comments here. First of all, our definition of a semismooth function is not the original one from [85], however, it follows from [85, Theorem 2.3] that it is equivalent to the original definition (note that the assumption of directional differentiability is missing in that result). Another very popular reformulation of the semismoothness of a locally Lipschitz and directionally differentiable function is that it satisfies

$$\|F(x + d) - F(x) - Hd\| = o(\|d\|) \tag{4.3}$$

for all $d \to 0$ and all $H \in \partial F(x + d)$. Sun [95] calls this the *superlinear approximation property* of F at x since it is central in order to prove local superlinear convergence of certain Newton-type methods, see also the general scheme in Kummer [62, 63]. The equivalence of this superlinear approximation property to our definition of semismoothness can be found, e.g., in [28, Theorem 7.4.3] and is based on the fact that a locally Lipschitz and directionally differentiable function is automatically B-differentiable, see [94] for details. On the other hand, property (4.3) can be defined also for mappings that are not necessarily directionally differentiable. In fact, Gowda [40] takes this property of a locally Lipschitz function as the definition of semismoothness. In order to avoid confusion with the existing definition of semismoothness, Pang et al. [83] suggested the name *G-semismoothness* (with the 'G' referring to Gowda).

We stress that the previous discussion on semismoothness is somewhat crucial for our later analysis since we want to apply a suitable implicit function theorem for semismooth functions. However, there are different implicit function theorems available in the literature, and they are based on different notions of a semismooth (or related) function, see, [95, 40] and, in particular, the corresponding discussion in [83].

We next state a simple result that will play an important role in later sections, in particular, the equivalence between statements (a) and (d).

Lemma 4.1.2 *Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be locally Lipschitz continuous and directionally differentiable, and let $x \in \mathbb{R}^n$ be an arbitrary point. Then the following statements are equivalent:*

(a) *F is semismooth at x, i.e., $\|Hd - F'(x;d)\| = o(\|d\|)$ for all $d \to 0$ and all $H \in \partial F(x+d)$.*

(b) *$\|Hd - F'(x;d)\| = o(\|d\|)$ for all $d \to 0$ and all $H \in \partial_B F(x+d)$.*

(c) *$\|Hd - F'(x;d)\| = o(\|d\|)$ for all $d \to 0$ and all $H \in \partial_C F(x+d)$.*

(d) *F_i is semismooth for all components $i = 1, \dots, m$, i.e., $\|h_i d - F_i'(x;d)\| = o(\|d\|)$ for all $d \to 0$, all $h_i \in \partial F_i(x+d)$, and all $i = 1, \dots, m$.*

Proof. The implications $(c) \implies (a) \implies (b)$ follow directly from the fact that $\partial_B F(x+d) \subseteq \partial F(x+d) \subseteq \partial_C F(x+d)$, cf. (4.2).

The implication $(b) \implies (a)$ is a consequence of Carathéodory's theorem. To see this, let $d^k \to 0$ and $H^k \in \partial F(x+d^k)$ be given arbitrarily. Then, for all $k \in \mathbb{N}$, we can find at most $r := nm + 1$ matrices $H_j^k \in \partial_B F(x+d^k)$ and numbers $\lambda_j^k \geq 0$ satisfying

$$\sum_{j=1}^r \lambda_j^k = 1 \quad \text{and} \quad H^k = \sum_{j=1}^r \lambda_j^k H_j^k.$$

Using (b), we therefore obtain

$$\begin{aligned} \|H^k d^k - F'(x;d^k)\| &= \Big\| \sum_{j=1}^r \lambda_j^k H_j^k d^k - F'(x;d^k) \Big\| \\ &\leq \sum_{j=1}^r \lambda_j^k \|H_j^k d^k - F'(x;d^k)\| = o(\|d^k\|) \end{aligned}$$

in view of the boundedness of λ_j^k.

The implication $(a) \implies (d)$ can be verified in the following way: Using the chain rule from [14, Theorem 2.6.6], the composite mapping $f := g \circ F$ with the continuously differentiable function $g(z) := z_i$ has the generalized gradient

$$\begin{aligned} \partial F_i(x) &= \partial f(x) = \partial g(F(x)) \partial F(x) = e_i^T \partial F(x) \\ &= \{h_i \mid h_i \text{ is the } i\text{th row of some } H \in \partial F(x)\}. \end{aligned}$$

Therefore, if we assume that (a) holds, and if we take an arbitrary $d \in \mathbb{R}^n$ as well as any component $i \in \{1, \dots, m\}$, it follows that for any $h_i \in \partial F_i(x+d)$, we can choose an element $H \in \partial F(x+d)$ such that its ith row is equal to h_i. Then we get

$$\left| F_i'(x;d) - h_i d \right| = \left| e_i^T (F'(x;d) - Hd) \right| \leq \|F'(x;d) - Hd\| = o(\|d\|),$$

hence F_i is semismooth at x.

Finally, (d) $\Longrightarrow$ (c) is an immediate consequence of the definition of the C-subdifferential.

Altogether, we have shown that $(c) \Longrightarrow (a) \Longrightarrow (d) \Longrightarrow (c)$ and $(a) \Longleftrightarrow (b)$, implying that all four statements are indeed equivalent. □

Some parts of the previous result are known, for example, [85, Corollay 2.4] shows that the semismoothness of all component functions implies the semismoothness of F itself. The fact that the converse also holds seems to be around in the community, but we were not able to find an explicit reference. Furthermore, [40, page 447] already observed the equivalence of statements (a) and (b) in Lemma 4.1.2, albeit in the slightly different context of G-semismoothness.

We next want to state an implicit function theorem for semismooth mappings that will be used in order to show local fast convergence of our Newton-type method for generalized Nash equilibrium problems. To this end, consider a mapping $H : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^n$, $(x, y) \mapsto H(x, y)$. Then $\pi_y \partial H(x, y)$ denotes the set of all $n \times n$ matrices M such that, for some $n \times m$ matrix N, the $n \times (m+n)$ matrix $[N, M]$ belongs to $\partial H(x, y)$. The set $\pi_x \partial H(x, y)$ is defined in a similar way.

Theorem 4.1.3 *Suppose that $H : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^n$ is locally Lipschitz and semismooth in a neighbourhood of a point $(\bar{x}, \bar{y})$ satisfying $H(\bar{x}, \bar{y}) = 0$, and assume that all matrices in $\pi_y \partial H(\bar{x}, \bar{y})$ are nonsingular. Then there exists an open neighborhood X of $\bar{x}$ and a function $g : X \to \mathbb{R}^n$ which is Lipschitz and semismooth on X such that $g(\bar{x}) = \bar{y}$ and $H(x, g(x)) = 0$ for all $x \in X$.*

Proof. Since this particular implicit function theorem does not seem to be available in the literature, we derive it from a suitable inverse function theorem. To this end, consider the mapping $F : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^m \times \mathbb{R}^n$ defined by

$$F(x, y) := \begin{pmatrix} x - \bar{x} \\ H(x, y) \end{pmatrix}.$$

Then

$$\partial F(\bar{x}, \bar{y}) \subseteq \begin{pmatrix} I_m & 0 \\ \pi_x \partial H(\bar{x}, \bar{y}) & \pi_y \partial H(\bar{x}, \bar{y}) \end{pmatrix},$$

and our assumptions imply that all elements from the generalized Jacobian $\partial F(\bar{x}, \bar{y})$ are nonsingular. Noting that a continuously differentiable function is always semismooth and recalling that the mapping H is semismooth by assumption, it follows from Lemma 4.1.2 that F is also semismooth. Hence we can apply the inverse function theorem from [83, Theorem 6] and obtain open neighbourhoods U of $(\bar{x}, \bar{y})$ and W of $(0, 0) = F(\bar{x}, \bar{y})$ such that $F : U \to W$ is a homeomorphism and has a locally Lipschitz and semismooth inverse $G : W \to U$. Since W is open, the set

$$X := \{x \in \mathbb{R}^m \mid (x - \bar{x}, 0) \in W\}$$

is also open as a subset of $\mathbb{R}^m$. We now show that there is a locally Lipschitz and semismooth function $g : X \to \mathbb{R}^n$ such that $g(\bar{x}) = \bar{y}$ and $H(x, g(x)) = 0$ for all $x \in X$.

To this end, let $x \in X$ be arbitrarily given. Then $(x - \bar{x}, 0) \in W$, and because $F : U \to W$ is a homeomorphism, the definition of the mapping F implies that there is a unique vector y such that $(x, y) \in U$ and $F(x, y) = (x - \bar{x}, 0)$. Consequently, we have $H(x, y) = 0$. Note that this unique vector y depends on x. Setting $g(x) := y$ then gives us a mapping $g : X \to \mathbb{R}^n$ such that $H(x, g(x)) = 0$ for each $x \in X$. This implies

$$F(x, g(x)) = \begin{pmatrix} x - \bar{x} \\ H(x, g(x)) \end{pmatrix} = \begin{pmatrix} x - \bar{x} \\ 0 \end{pmatrix} \quad \forall x \in X.$$

Applying the inverse mapping G on both sides gives

$$\begin{pmatrix} x \\ g(x) \end{pmatrix} = G(x - \bar{x}, 0) \quad \forall x \in X.$$

This shows that g coincides with certain component functions of G. Since the inverse function G is semismooth, it therefore follows from Lemma 4.1.2 that g is also semismooth. This completes the proof of our implicit function theorem. □

A related implicit function theorem was stated in Sun [95]. However, he only assumes that H has the local superlinear approximation property, and states that the implicit function has the superlinear approximation property, too. A similar result was also stated by Gowda [40] in the framework of H-differentiable functions. Note also that the assumption on the nonsingularity of all elements from $\pi_y \partial H(\bar{x}, \bar{y})$ (corresponding to the strongest possible condition in the inverse function theorem from [83]) can be weakened, but that this (relatively strong) condition will be satisfied in our context.

We close this section with the definition of an SC^1-function that will become important in the next section.

Definition 4.1.4 *A mapping $f : \mathbb{R}^n \to \mathbb{R}$ is called an SC^1-*function *if it is continuously differentiable and its gradient ∇f is semismooth.*

4.2 SC^1-Optimization Reformulations

Consider the GNEP with cost functions $\theta_\nu, \nu = 1, \dots, N$, and a joint strategy set X satisfying the requirements from Assumptions 4.0.5. Our aim is to show that the GNEP can then be reformulated as both constrained and unconstrained SC^1 optimization problems. This SC^1-reformulation is based on the smooth constrained and unconstrained optimization reformulations of chapter 2.
We briefly restate the essential statements from chapter 2. In equation 2.8, for $\gamma > 0$ (γ instead of α) we defined the function

$$V_\gamma(x) = \Psi_\gamma(x, y_\gamma(x)), \tag{4.4}$$

with

$$y_\gamma(x) = \arg\max_{y \in X} \Psi_\gamma(x, y), \tag{4.5}$$

where Ψ_γ is the regularized Nikaido-Isoda function,

$$\Psi_\gamma(x, y) = \sum_{\nu=1}^{N} [\theta_\nu(x^\nu, x^{-\nu}) - \theta_\nu(y^\nu, x^{-\nu}) - \frac{\gamma}{2}\|x^\nu - y^\nu\|^2]. \tag{4.6}$$

Theorem 2.2.3 shows that x^* is a normalized Nash equilibrium if and only if x^* solves the optimization problem

$$\min_x V_\gamma(x) \quad \text{subject to } x \in X.$$

Furthermore, Theorem 2.2.5 implies that the function V_γ is continuously differentiable with gradient

$$\nabla V_\gamma(x) = \nabla_x \Psi_\gamma(x, y)\big|_{y = y_\gamma(x)}. \tag{4.7}$$

Unfortunately, V_γ is, in general, not twice continuously differentiable. However, in view of Assumption 4.0.5, we see that the regularized Nikaido-Isoda-function $\Psi_\gamma(x, y)$ is twice continuously differentiable. Using the fact that the composition of semismooth functions is again semismooth, see [33], it therefore follows immediately from the representation (4.7) of the gradient ∇V_γ that V_γ is an SC^1-function if the mapping $x \mapsto y_\gamma(x)$ is semismooth. Our aim in this section is therefore to prove the semismoothness of this mapping.

To this end, we first consider a more general parameterized optimization problem of the form

$$\min_y \; f(x, y) \quad \text{s.t.} \quad y \in X \tag{4.8}$$

where $f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable and uniformly convex with respect to the variable y (for every fixed x). The feasible set X is given by a number of inequalities as in (4.1) such that Assumption 4.0.5 (b) holds. Then the Lagrangian of the optimization problem (4.8) is given by

$$L(x, y, \lambda) = f(x, y) + \sum_{i=1}^{m} \lambda_i g_i(y),$$

where, again, $x \in \mathbb{R}^n$ is supposed to be fixed. Let $y = y(x)$ be the unique solution of the optimization problem (4.8). Then, under a suitable constraint qualification, like the Slater condition, it follows that there exists a Lagrange multiplier $\lambda = \lambda(x) \in \mathbb{R}^m$ such that (y, λ) together with the fixed x solves the KKT system

$$\nabla_y L(x, y, \lambda) = \nabla_y f(x, y) + \nabla g(y)\lambda = 0, \quad 0 \leq \lambda \perp -g(y) \geq 0. \tag{4.9}$$

Using the minimum function $\varphi : \mathbb{R} \times \mathbb{R} \to \mathbb{R}, \varphi(a, b) := \min\{a, b\}$, we can reformulate the KKT system (4.9) as a system of nonlinear equations $\Phi(x, y, \lambda) = 0$ via the function

$$\Phi(x, y, \lambda) := \begin{pmatrix} \nabla_y L(x, y, \lambda) \\ \phi(-g(y), \lambda) \end{pmatrix} \tag{4.10}$$

with

$$\phi(-g(y), \lambda) := (\varphi(-g_1(y), \lambda_1), \ldots, \varphi(-g_m(y), \lambda_m))^T \in \mathbb{R}^m.$$

Our first result gives a representation of the B-subdifferential and the generalized Jacobian of the mapping Φ.

Lemma 4.2.1 *Suppose that f and g are C^2-functions. Let $w = (x, y, \lambda) \in \mathbb{R}^{n+n+m}$. Then, each element $H \in \partial\Phi(w)^T$ can be represented as follows:*

$$H = \begin{pmatrix} \nabla^2_{yx} L(x, y, \lambda)^T & 0 \\ \nabla^2_{yy} L(x, y, \lambda) & -\nabla g(y) D_a(y, \lambda) \\ \nabla g(y)^T & D_b(y, \lambda) \end{pmatrix}$$

where $D_a(y, \lambda) := \mathrm{diag}(a_1(y, \lambda), \ldots, a_m(y, \lambda))$, $D_b(y, \lambda) = \mathrm{diag}(b_1(y, \lambda), \ldots, b_m(y, \lambda)) \in \mathbb{R}^{m \times m}$ are diagonal matrices whose ith diagonal elements are given by

$$a_i(y, \lambda) = \begin{cases} 1, & \text{if } -g_i(y) < \lambda_i, \\ 0, & \text{if } -g_i(y) > \lambda_i, \\ \mu_i, & \text{if } -g_i(y) = \lambda_i, \end{cases} \quad \text{and} \quad b_i(y, \lambda) = \begin{cases} 0, & \text{if } -g_i(y) < \lambda_i, \\ 1, & \text{if } -g_i(y) > \lambda_i, \\ 1 - \mu_i, & \text{if } -g_i(y) = \lambda_i, \end{cases}$$

for any $\mu_i \in [0, 1]$. The elements $H \in \partial_B \Phi(w)^T$ are obtained by choosing $\mu_i \in \{0, 1\}$.

Proof. The first n components of the vector function Φ are continuously differentiable and Φ is continuously differentiable with respect to x, so the expression for the first n rows and columns of H readily follows. To compute the remaining entries of H, we use the fact that each element of the generalized Jacobian of ϕ can be represented by an element of the C-subdifferential of ϕ, that is

$$\partial\phi(-g(y),\lambda)^T \subseteq \partial\varphi(-g_1(y),\lambda_1)^T \times\cdots\times \partial\varphi(-g_m(y),\lambda_m)^T.$$

If i is such that $-g_i(y) \neq \lambda_i$, then φ is continuously differentiable at $(-g_i(y),\lambda_i)$ and the expression for the $(n+i)$th column of H follows. If instead $-g_i(y) = \lambda_i$, then, using the definition of the B-subdifferential, it follows that

$$\partial_B\varphi(-g_i(y),\lambda_i)^T = \{(-\nabla g_i(y)^T, 0), (0, e_i^T)\}.$$

Taking the convex hull, we therefore get

$$\partial\varphi(-g_i(y),\lambda_i)^T = \{(-\mu_i\nabla g_i(y)^T, (1-\mu_i)e_i^T) \mid \mu_i \in [0,1]\}.$$

(Note that this representation cannot be obtained by directly applying [14, Theorem 2.3.9 (iii)] since the min-function is not regular in the sense of [14, Definition 2.3.4].) This gives the representation of $H \in \partial\Phi(w)^T$. □

Our next aim is to establish conditions for the nonsingularity of all elements in $\pi_{(y,\lambda)}\partial\Phi(w)^T$ at a point $w = (x, y, \lambda)$ satisfying $\Phi(w) = 0$. By definition, taking the continuous differentiability of Φ with respect to x into account, the elements $V \in \pi_{(y,\lambda)}\partial\Phi(w)^T$ can be obtained by deleting the first n rows of the matrices H from Lemma 4.2.1. In order to get a more detailed description of the matrices $V \in \pi_{(y,\lambda)}\partial\Phi(w)^T$, let us partition the index set $\{1,\dots,m\}$ into

$$I_0 := \{i \mid g_i(y) = 0\} \quad \text{and} \quad I_< := \{i \mid g_i(y) < 0\},$$

where both the set of active constraints I_0 and the set of inactive constraints $I_<$ depend on the current vector y. The set of active constraints can be further divided into

$$I_{00} := \{i \in I_0 \mid \lambda_i = 0\} \quad \text{and} \quad I_+ := \{i \in I_0 \mid \lambda_i > 0\},$$

with both sets depending on y and λ. The set I_{00} will further be partitioned into

$$I_{01} := \{i \in I_{00} \mid \mu_i = 1\}, \quad I_{02} := \{i \in I_{00} \mid \mu_i \in (0,1)\}, \quad I_{03} := \{i \in I_{00} \mid \mu_i = 0\}.$$

Note that these index sets also depend (via μ_i) on the particular element taken from the generalized Jacobian of $\Phi(w)$.

With these index sets, and using a suitable reordering of the constraints, every element $V \in \pi_{(y,\lambda)}\partial\Phi(x,y,\lambda)^T$ has the following structure (the dependence on $w =$

(x, y, λ) is suppressed for simplicity):

$$V = \begin{pmatrix} \nabla^2_{yy} L & -\nabla g_+ & -\nabla g_{01} & -\nabla g_{02}(D_a)_{02} & 0 & 0 \\ \nabla g_+^T & 0 & 0 & 0 & 0 & 0 \\ \nabla g_{01}^T & 0 & 0 & 0 & 0 & 0 \\ \nabla g_{02}^T & 0 & 0 & (D_b)_{02} & 0 & 0 \\ \nabla g_{03}^T & 0 & 0 & 0 & I & 0 \\ \nabla g_<^T & 0 & 0 & 0 & 0 & I \end{pmatrix}, \qquad (4.11)$$

where $(D_a)_{02}$ and $(D_b)_{02}$ are positive definite diagonal matrices, and where we used the abbreviations $\nabla g_+, \nabla g_{01}$ etc. for the matrices $\nabla g_{I_+}, \nabla g_{I_{01}}$ etc.

In order to obtain a suitable nonsingularity result, let us introduce the matrices

$$M(J) := \begin{pmatrix} \nabla^2_{yy} L & -\nabla g_+ & -\nabla g_J \\ \nabla g_+^T & 0 & 0 \\ \nabla g_J^T & 0 & 0 \end{pmatrix},$$

where J is any subset of I_{00}. Using these matrices, we next define the concept of strong regularity for the parameterized optimization problem (4.8). This name comes from the fact that our condition corresponds to Robinson's strong regularity assumption (see [89]) in the context of ordinary nonlinear programs, cf. [65, 30].

Definition 4.2.2 *A triple* $w^* = (x^*, y^*, \lambda^*)$ *satisfying* $\Phi(w^*) = 0$ *is called* strongly regular *for the optimization problem* (4.8) *if the matrices* $M(J)$ *have the same nonzero orientation for all* $J \subseteq I_{00}$.

According to Robinson [89], strong regularity holds if the strong second order sufficiency condition and the linear independence constraint qualification (LICQ for short) hold, where LICQ means that the gradients $\nabla g_i(x^*)$ $(i : g_i(x^*) = 0)$ of the active inequality constraints are linearly independent (note that LICQ is also a necessary condition for strong regularity). In particular, it therefore follows that all matrices $M(J)$ have the same nonzero orientation if $\nabla^2_{yy} L$ is positive definite and LICQ holds. This is the situation we are particularly interested in. In fact, in this case, there is an easy way to see that strong regularity holds at $w^* = (x^*, y^*, \lambda^*)$. To this end, write

$$M(J) = \begin{pmatrix} H & -A_J \\ A_J^T & 0 \end{pmatrix} \quad \text{with} \quad H := \nabla^2_{yy} L \quad \text{and} \quad A_J := (\nabla g_+, \nabla g_J).$$

Using block Gaussian elimination, it follows that

$$M(J) = \begin{pmatrix} I & 0 \\ A_J^T H^{-1} & I \end{pmatrix} \begin{pmatrix} H & -A_J \\ 0 & A_J^T H^{-1} A_J \end{pmatrix}.$$

Consequently, we get

$$\det(M(J)) = \det\begin{pmatrix} H & -A_J \\ 0 & A_J^T H^{-1} A_J \end{pmatrix} = \det(H)\det(A_J^T H^{-1} A_J) > 0 \quad \forall J \subseteq I_{00}$$

since H is positive definite and A_J has full column rank for all $J \subseteq I_{00}$.

We next state our main result on the nonsingularity of the elements of the projected generalized Jacobian $\pi_{(y,\lambda)}\partial\Phi(x^*, y^*, \lambda^*)$. Its proof is similar to one given in [30] which, however, uses a different reformulation of the KKT system arising from variational inequalities.

Theorem 4.2.3 *Consider the optimization problem* (4.8) *with* $f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ *and* $g : \mathbb{R}^n \to \mathbb{R}^m$ *being twice continuously differentiable. Let* $w^* = (x^*, y^*, \lambda^*) \in \mathbb{R}^{n+n+m}$ *be a solution of the system* $\Phi(w) = 0$, *and suppose that the strong regularity condition holds at* w^*. *Then all elements* $V \in \pi_{(y,\lambda)}\partial\Phi(w^*)$ *are nonsingular.*

Proof. Consider an arbitrary but fixed element in $\pi_{(y,\lambda)}\partial\Phi(w^*)^T$. This element has the structure indicated in (4.11) and is obviously nonsingular if and only if the following matrix is nonsingular:

$$V = \begin{pmatrix} \nabla^2_{yy}L & -\nabla g_+ & -\nabla g_{01} & -\nabla g_{02} \\ \nabla g_+^T & 0 & 0 & 0 \\ \nabla g_{01}^T & 0 & 0 & 0 \\ \nabla g_{02}^T & 0 & 0 & (D_b)_{02}(D_a)_{02}^{-1} \end{pmatrix}. \tag{4.12}$$

The matrix (4.12) can be written as the sum of the matrix $M(J)$, with $J = I_{01} \cup I_{02}$, and the diagonal matrix

$$D := \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (D_b)_{02}(D_a)_{02}^{-1} \end{pmatrix}.$$

Given a square matrix $\bar{A}$ of dimension r and a diagonal matrix $\bar{D}$ of the same dimension, it follows from [16, p. 60] that

$$\det(\bar{D} + \bar{A}) = \sum_{\alpha} \det\bar{D}_{\alpha\alpha}\det\bar{A}_{\bar{\alpha}\bar{\alpha}}, \tag{4.13}$$

where the summation ranges over all subsets α of $\{1, \dots, r\}$ (with complements $\bar{\alpha} = \{1, \dots, r\}\setminus\alpha$), and where it is assumed that the determinant of an "empty" matrix is equal to 1. Exploiting this formula, the determinant of (4.12) can be written as

$$\det M(J) + \sum_{\emptyset \neq \alpha \subseteq I_{02}} \det D_{\alpha\alpha}\det M(J)_{\bar{\alpha}\bar{\alpha}}, \tag{4.14}$$

where the first term corresponds to $\alpha = \emptyset$. Moreover, we have taken into account that if α contains an element which does not belong to I_{02}, then the determinant of $D_{\alpha\alpha}$ is 0. Since the nonzero diagonal elements of the matrix D are all positive, it follows that the determinants of $D_{\alpha\alpha}$ in (4.14) are all positive. Then to show that the determinant of (4.12) is nonzero and hence to conclude the proof, it will now be sufficient to show that the determinants of $M(J)$ and of all $M(J)_{\bar{\alpha}\bar{\alpha}}$ in (4.14) never have opposite signs, and that at least one of them is nonzero. But this is a direct consequence of Definition 4.2.2. □

Now we are able to apply Theorem 4.1.3 to the optimization problem (4.8).

Corollary 4.2.4 *Let the assumptions of Theorem 4.2.3 be satisfied. Then there exists a neighbourhood U of x^* and a semismooth function $G : U \to \mathbb{R}^{n+m}, x \mapsto (y(x), \lambda(x))$ such that $\Phi(x, G(x)) = 0$ holds for all $x \in U$. In particular, the mapping $x \mapsto y(x)$ is semismooth.*

Proof. The existence and semismoothness of the implicit function $x \mapsto G(x) = (y(x), \lambda(x))$ is an immediate consequence of Theorems 4.1.3 and 4.2.3. Using Lemma 4.1.2, this, in particular, implies the local semismoothness of the mapping $x \mapsto y(x)$. □

We now get back to our GNEP and the mapping V_γ defined in (4.4). The following is the main result of this section.

Theorem 4.2.5 *Let $x^* \in X$ and assume that LICQ holds at $y_\gamma(x^*)$. Then V_γ is an SC^1-function in a neighbourhood of $y_\gamma(x^*)$.*

Proof. In view of the introductory remarks of this section, we have to show that the mapping $x \mapsto y_\gamma(x)$ is semismooth in a neighbourhood of x^*. By definition, $y_\gamma(x)$ is the solution of the optimization problem

$$\max_y \ \Psi_\gamma(x, y) \quad \text{s.t.} \quad y \in X := \{y \in \mathbb{R}^n \mid g(y) \le 0\}, \tag{4.15}$$

cf. (4.5). This is an optimization problem of the form (4.8) with $f(x, y) := -\Psi_\gamma(x, y)$. Here, the mapping f is uniformly convex with respect to y due to the regularization term in the definition of the regularized Nikaido-Isoda-function and the assumed convexity of the mappings θ_ν with respect to the variables x^ν. Corollary 4.2.4 therefore gives the semismoothness of the mapping $x \mapsto y_\gamma(x)$ provided that the strong regularity assumption holds at $(x^*, y_\gamma(x^*), \lambda_\gamma(x^*))$, where $y_\gamma(x^*)$ denotes the solution of problem (4.15) with $x = x^*$ and $\lambda_\gamma(x^*)$ is the corresponding unique (due to LICQ) multiplier.

Since LICQ holds at $y_\gamma(x^*)$, it suffices to show that the Hessian (with respect to y) of the corresponding Lagrangian

$$L_\gamma(x, y, \lambda) = -\Psi_\gamma(x, y) + \sum_{i=1}^{m} \lambda_i g_i(y)$$

is positive definite at $(x, y, \lambda) = (x^*, y_\gamma(x^*), \lambda_\gamma(x^*))$, see the comments after Definition 4.2.2. However, we already observed that $-\Psi_\gamma(x, y)$ is uniformly convex with respect to y, hence its Hessian is (uniformly) positive definite. Furthermore, $\nabla^2 g_i(y_\gamma(x^*))$ is positive semidefinite due to the assumed convexity of the functions g_i. Hence the assertion follows from the fact that $\lambda = \lambda_\gamma(x^*)$ is nonnegative (as a multiplier corresponding to an inequality constraint). □

Note that, if, in addition to the assumptions of Theorem 4.2.5, strict complementarity holds at $y_\gamma(x^*)$, then y_γ is continuously differentiable and V_γ is a C^2-function in a neighbourhood of x^*. This follows directly from the previous derivation by using the standard implicit function theorem in place of Theorem 4.1.3.

Furthermore, we would like to point out that the assertion of Theorem 4.2.5 holds at all $x \in \mathbb{R}^n$ such that LICQ is satisfied at $y_\gamma(x)$.

4.3 Newton's method

In view of Theorem 4.2.5, both the constrained optimization reformulation (2.11) and the unconstrained reformulation (2.14) of the GNEP are SC^1 optimization problems. Hence it is reasonable to believe that locally superlinearly convergent Newton-type methods can be derived for the solution of GNEPs via the solution of these optimization problems. Here we focus on the unconstrained reformulation (2.14) and show that one can indeed expect local fast convergence of a nonsmooth Newton-type method under suitable assumptions.

The nonsmooth Newton-type method from [85, 84] for the minimization of the unconstrained function $V_{\alpha\beta}$ from (2.12) is an iterative procedure of the form

$$x^{k+1} := x^k + d^k, \quad k = 0, 1, 2, \dots, \tag{4.16}$$

where $x^0 \in \mathbb{R}^n$ is a starting point and d^k is a solution of the linear system

$$H_k d = -\nabla V_{\alpha\beta}(x^k) \quad \text{for some} \quad H_k \in \partial^2 V_{\alpha\beta}(x^k), \tag{4.17}$$

where $\partial^2 V_{\alpha\beta}(x^k)$ denotes the *generalized Hessian* of $V_{\alpha\beta}$ at x^k in the sense of [51], i.e., $\partial^2 V_{\alpha\beta}(x^k)$ is the generalized Jacobian in the sense of Clarke [14] of the locally Lipschitz mapping $F := \nabla V_{\alpha\beta}$.

In order to compute the gradient and (generalized) Hessian matrix of the mapping $V_{\alpha\beta}$, we need several (partial) derivatives of the mapping Ψ_γ from (4.6). These derivatives are summarized in the following result whose proof is omitted since it follows from standard calculus rules.

Lemma 4.3.1 *The mapping* Ψ_γ *from* (4.6) *is twice continuously differentiable with (partial) derivatives*

$$
\begin{aligned}
\nabla_x \Psi_\gamma(x,y) &= \sum_{\nu=1}^{N} \left[\nabla\theta_\nu(x^\nu, x^{-\nu}) - \nabla\theta_\nu(y^\nu, x^{-\nu})\right] + \begin{pmatrix} \nabla_{x^1}\theta_1(y^1, x^{-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y^N, x^{-N}) \end{pmatrix} - \gamma(x-y), \\
\nabla_y \Psi_\gamma(x,y) &= -\begin{pmatrix} \nabla_{x^1}\theta_1(y^1, x^{-1}) \\ \vdots \\ \nabla_{x^N}\theta_N(y^N, x^{-N}) \end{pmatrix} + \gamma(x-y), \\
\nabla^2_{xx} \Psi_\gamma(x,y) &= \sum_{\nu=1}^{N} \left[\nabla^2\theta_\nu(x^\nu, x^{-\nu}) - \nabla^2\theta_\nu(y^\nu, x^{-\nu})\right] \\
&\quad + \begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^1x^N}\theta_N(y^N, x^{-N}) \\ \vdots & \ddots & \vdots \\ \nabla^2_{x^Nx^1}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^Nx^N}\theta_N(y^N, x^{-N}) \end{pmatrix} \\
&\quad + \begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^1x^N}\theta_1(y^1, x^{-1}) \\ \vdots & \ddots & \vdots \\ \nabla^2_{x^Nx^1}\theta_N(y^N, x^{-N}) & \cdots & \nabla^2_{x^Nx^N}\theta_N(y^N, x^{-N}) \end{pmatrix} \\
&\quad - diag\begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1, x^{-1}) & & \\ & \ddots & \\ & & \nabla^2_{x^Nx^N}\theta_N(y^N, x^{-N}) \end{pmatrix} - \gamma I, \\
\nabla^2_{xy} \Psi_\gamma(x,y) &= -\begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^1x^N}\theta_N(y^N, x^{-N}) \\ \vdots & \ddots & \vdots \\ \nabla^2_{x^Nx^1}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^Nx^N}\theta_N(y^N, x^{-N}) \end{pmatrix} \\
&\quad + diag\begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1, x^{-1}) & & \\ & \ddots & \\ & & \nabla^2_{x^Nx^N}\theta_N(y^N, x^{-N}) \end{pmatrix} + \gamma I, \\
\nabla^2_{yx} \Psi_\gamma(x,y) &= \nabla^2_{xy}\Psi_\gamma(x,y)^T \\
&= -\begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^1x^N}\theta_1(y^1, x^{-1}) \\ \vdots & \ddots & \vdots \\ \nabla^2_{x^Nx^1}\theta_N(y^N, x^{-N}) & \cdots & \nabla^2_{x^Nx^N}\theta_N(y^N, x^{-N}) \end{pmatrix}
\end{aligned}
$$

$$+diag\begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1,x^{-1}) & & \\ & \ddots & \\ & & \nabla^2_{x^Nx^N}\theta_N(y^N,x^{-N}) \end{pmatrix} + \gamma I,$$

$$\nabla^2_{yy}\Psi_\gamma(x,y) = -diag\begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(y^1,x^{-1}) & & \\ & \ddots & \\ & & \nabla^2_{x^Nx^N}\theta_N(y^N,x^{-N}) \end{pmatrix} - \gamma I.$$

We next consider the problem of how to implement the nonsmooth Newton-type method. To this end, we have to compute, at each iterate x^k, an element $H_k \in \partial^2 V_{\alpha\beta}(x^k)$. Since this is not an easy task, we first assume that V_α and V_β are both twice continuously differentiable at x^k, hence $V_{\alpha\beta}$ is twice continuously differentiable at $x := x^k$ with Hessian

$$\nabla^2 V_{\alpha\beta}(x) = \nabla^2 V_\alpha(x) - \nabla^2 V_\beta(x). \tag{4.18}$$

Hence we need to calculate the Hessians $\nabla^2 V_\gamma(x)$ for $\gamma \in \{\alpha,\beta\}$. Therefore, let $\gamma \in \{\alpha,\beta\}$ be fixed and recall that V_γ is given by (4.4) with gradient $\nabla V_\gamma(x) = \nabla_x \Psi_\gamma(x, y_\gamma(x))$, cf. (4.7), where $y_\gamma(x)$ denotes the solution of the optimization problem (4.5). Using the chain rule, we therefore get

$$\nabla^2 V_\gamma(x) = \nabla^2_{xx}\Psi_\gamma(x,y_\gamma(x)) + \nabla^2_{xy}\Psi_\gamma(x,y_\gamma(x))Dy_\gamma(x), \tag{4.19}$$

where $Dy_\gamma(x) \in \mathbb{R}^{n\times n}$ denotes the usual Jacobian (with respect to x) of the mapping y_γ. Expressions for the matrices $\nabla^2_{xx}\Psi_\gamma(x,y_\gamma(x))$ and $\nabla^2_{xy}\Psi_\gamma(x,y_\gamma(x))$ are given in Lemma 4.3.1. At a nondifferentiable point, we have the following result.

Lemma 4.3.2 *The following inclusion holds at an arbitrary point $x \in \mathbb{R}^n$:*

$$\nabla^2_{xx}\Psi_\gamma(x,y_\gamma(x)) + \nabla^2_{xy}\Psi_\gamma(x,y_\gamma(x))\partial_B y_\gamma(x) \subseteq \partial^2_B V_\gamma(x),$$

where $\partial^2_B V_\gamma$ denotes the Bouligand subdifferential of the function ∇V_γ.

Proof. Let $x \in \mathbb{R}^n$ be arbitrarily given, and let $Y \in \partial_B y_\gamma(x)$. Then there is a sequence $\{\xi^k\} \to x$ such that y_γ is differentiable at each ξ^k and $Dy_\gamma(\xi^k) \to Y$ for $k \to \infty$. Then the representation (4.7) of ∇V_γ shows that V_γ is twice differentiable at each ξ^k, and we therefore obtain, taking the continuity of y_γ and the twice continuous differentiability of Ψ_γ into account:

$$\begin{aligned}\nabla^2 V_\gamma(\xi^k) &= \nabla^2_{xx}\Psi_\gamma(\xi^k,y_\gamma(\xi^k)) + \nabla^2_{xy}\Psi_\gamma(\xi^k,y_\gamma(\xi^k))Dy_\gamma(\xi^k) \\ &\to \nabla^2_{xx}\Psi_\gamma(x,y_\gamma(x)) + \nabla^2_{xy}\Psi_\gamma(x,y(x))Y.\end{aligned}$$

This shows that the right-hand side belongs to $\partial^2_B V_\gamma(x)$. □

Hence we need to consider the computation of $\partial_B y_\gamma(x)$. By definition, $y_\gamma(x)$ is the unique solution of the optimization problem

$$\min_y -\Psi_\gamma(x,y) \quad \text{s.t.} \quad y \in X := \{y \in \mathbb{R}^n \mid g(y) \leq 0\}.$$

Assume that LICQ holds at $y_\gamma(x)$, and let

$$L_\gamma(x,y,\lambda) := -\Psi_\gamma(x,y) + \sum_{i=1}^m \lambda_i g_i(y)$$

be the Lagrangian of this optimization problem. Since LICQ holds at $y_\gamma(x)$, it follows that there exist unique multipliers $\lambda_\gamma(x)$ such that the following KKT conditions hold at $(x,y,\lambda) = (x, y_\gamma(x), \lambda_\gamma(x))$:

$$\nabla_y L_\gamma(x,y,\lambda) = 0, \quad \lambda \geq 0, \;\; g(y) \leq 0, \;\; \lambda^T g(y) = 0.$$

Here we have

$$\nabla_y L_\gamma(x,y,\lambda) = -\nabla_y \Psi_\gamma(x,y) + \sum_{i=1}^m \lambda_i \nabla g_i(y).$$

Therefore, assuming, for the moment, that strict complementarity holds, we then obtain from the standard implicit function theorem that the implicit function $G(x) := (y_\gamma(x), \lambda_\gamma(x))$ satisfies (locally) the system of equations

$$\Phi_\gamma(x, G(x)) = 0, \quad \text{where} \quad \Phi_\gamma(x,y,\lambda) := \begin{pmatrix} \nabla_y L_\gamma(x,y,\lambda) \\ \min\{-g(y), \lambda\} \end{pmatrix}. \tag{4.20}$$

Differentiating this system therefore gives

$$0 = D_x \Phi_\gamma(x, G(x)) = D_x \Phi_\gamma(x, G(x)) + D_{(y,\lambda)} \Phi_\gamma(x, G(x)) D_x G(x),$$

from which we obtain $D_x G(x) = (D_x y_\gamma(x), D_x \lambda_\gamma(x))$ by solving the linear system

$$D_{(y,\lambda)} \Phi_\gamma(x, G(x)) D_x G(x) = -D_x \Phi_\gamma(x, G(x)). \tag{4.21}$$

Some computational effort leads to the following formula for the Jacobian of the function y_γ at x :

$$\nabla y_\gamma(x)^T = C^{-1}A - C^{-1}D(D^T C^{-1} D)^{-1} D^T C^{-1} A, \tag{4.22}$$

with

$$A \;=\; A(x) \;:=\; \nabla^2_{yx} \Psi_\gamma(x, y_\gamma(x)),$$

$$
\begin{aligned}
C &= C(x) := -\nabla^2_{yy}\Psi_\gamma(x, y_\gamma(x)) + \sum_{i=1}^m \lambda_i \nabla^2 g_i(y_\gamma(x)), \\
D &= D(x) := \nabla g_{I_0}(y_\gamma(x)),
\end{aligned}
$$

where I_0 denotes the index set of active constraints at $y_\gamma(x)$. Still this formula does not provide an element of the Bouligand subdifferential $\partial_B y_\gamma(x)$. Yet it can be shown, that under the linear independence constraint qualification, all elements of $\partial_B y_\gamma(x)$ can be expressed by a formula of the type of (4.23), see Proposition 5.1.6 in the next chapter.

Even if we had an element of $\partial_B y_\gamma(x)$, it would not be easy to calculate an element of $\partial^2_B V_{\alpha\beta}(x)$. Knowledge of $\partial_B y_\gamma(x)$ allows us to compute elements from $\partial^2_B V_\alpha(x)$ and $\partial^2_B V_\beta(x)$, respectively, as Lemma 4.3.2 shows, however, this does not suffice, since it is very unlikely that the difference of any two elements in $\partial^2_B V_\beta(x)$ and $\partial^2_B V_\alpha(x)$ is an element of $\partial^2_B V_{\alpha\beta}(x)$.

On the other hand, all examples from the literature presented in chapter 6 except Rosen's example satisfy strict complementarity at the solution. Therefore, applying Newton's method to the unconstrained optimization reformulation is not difficult to implement in these cases.

The following is the central local convergence result for our nonsmooth Newton-type method for the solution of the GNEP.

Theorem 4.3.3 *Let x^* be a normalized Nash equilibrium of the GNEP such that all elements $V \in \partial^2 V_{\alpha\beta}(x^*)$ are nonsingular. Then the nonsmooth Newton-type method from* (4.16), (4.17) *is locally superlinearly convergent to x^*.*

Proof. Since $V_{\alpha\beta}$ is an SC^1-function in view of Theorem 4.2.5, the result follows immediately from [85]. □

There are a number of comments that we would like to add in the following remark.

Remark 4.3.4 (a) Theorem 4.3.3 remains true if we replace the assumption that all elements of $\partial^2 V_{\alpha\beta}(x^*)$ are nonsingular by the weaker condition that all elements from the smaller set $\partial^2_B V_{\alpha\beta}(x^*)$ are nonsingular. This follows immediately from a result in [84].

(b) Theorem 4.3.3 is a local convergence result only. However, since $V_{\alpha\beta}$ is continuously differentiable, it is easy to globalize this method by either a line search or a trust-region strategy. These globalized methods typically find a stationary point of $V_{\alpha\beta}$ only, and a sufficient condition for such a stationary point to be a normalized Nash equilibrium of the GNEP is given in Assumption 3.1.11.

(c) Theorem 4.3.3 gives a local superlinear rate of convergence. It is also possible to get a locally quadratically convergent method by our approach. To this end, we have to strengthen Assumption 1.2.2 to some extend and assume that, in addition, the Hessian matrices $\nabla^2\theta_\nu$ and $\nabla^2 g_i$ are locally Lipschitz around a solution x^* of the GNEP. Moreover, one has to use another implicit function theorem like the one from Sun [95] in order to guarantee a local quadratic approximation property (as defined in [95]) or to modify Theorem 4.1.3 in a suitable way.

(d) A simple sufficient condition for the nonsingularity of all elements from $\partial^2 V_{\alpha\beta}(x^*)$ (or $\partial_B^2 V_{\alpha\beta}(x^*)$) exists for linear-quadratic games as defined in Proposition 3.1.2. Suppose that the solution x^* from Theorem 4.3.3 is locally unique (for which there exist simple conditions in the case of linear-quadratic games) and satisfies strict complementarity, i.e., the corresponding vectors $y_\gamma(x^*)$ and $\lambda_\gamma(x^*)$ satisfy the strict complementarity condition. Then the standard implicit function theorem guarantees that $V_{\alpha\beta}$ is twice continuously differentiable around x^*. Therefore, the local uniqueness of the normalized solution x^*, together with the quadratic nature of the costs functions, implies that the Hessian $\nabla^2 V_{\alpha\beta}(x^*)$ is positive definite.

(e) The Newton method presented in this chapter is, under favourable conditions, locally superlinearly or even quadratically convergent. Yet, there are two major drawbacks of this method: First, there are some difficulties in the calculation of elements of the generalized Jacobian of $\nabla V_{\alpha\beta}$, and second, we have to assume that the linear independence constraint qualification holds at the solution x^*. In the next chapter we therefore present an approach that disposes with both problems: It allows for a simple formula for the elements of the generalized Jacobian, and it does not require LICQ, but instead the slightly weaker constant rank constraint qualification.

Chapter 5

Newton's Method through Fixed Point Formulation

In the preceding chapter we presented a Newton method in order to solve the unconstrained optimization problem

$$\min V_{\alpha\beta}(x) \quad x \in \mathbb{R}^n,$$

where the function $V_{\alpha\beta}$ (defined in (2.12)) has the property that every global minimum of $V_{\alpha\beta}$ is a normalized Nash equilibrium. In this chapter, we also develop a Newton method for the computation of a normalized Nash equilibrium, however, the approach is completely different. Basically, the Newton method presented in this chapter solves a fixed point equation, opposed to the Newton method in the preceding chapter which solves the necessary first-order condition for an unconstrained optimization problem.

Throughout this chapter, we assume that the feasible set X is given by inequalities,

$$X := \{x \in \mathbb{R}^n \mid g(x) \leq 0\}, \tag{5.1}$$

with a function $g : \mathbb{R}^n \to \mathbb{R}^m$. Both g and the cost functions θ_ν, $\nu = 1, \dots, N$, shall satisfy the following assumptions.

Assumption 5.0.5

(a) *The cost functions θ_ν, $\nu = 1, \dots, N$ are twice continuously differentiable, and convex with respect to the variable x^ν, i.e., the function $\theta_\nu(\cdot, x^{-\nu})$ is convex, uniformly for all $x^{-\nu}$;*

(b) *The function g is twice continuously differentiable, its components g_i are convex functions (in x), and the corresponding strategy space X defined by* (5.1) *is nonempty.*

This is exactly the same assumption as Assumption (4.0.5) in the preceding chapter.
For a fixed parameter $\alpha > 0$ we define the function

$$\psi_\alpha(x, y) := \sum_{\nu=1}^{N} \left[\theta_\nu(y^\nu, x^{-\nu}) + \frac{\alpha}{2}\|y^\nu - x^\nu\|^2\right] \tag{5.2}$$

and consider the optimization problem

$$\min_y \psi_\alpha(x, y) \quad \text{s.t. } y \in X. \tag{5.3}$$

Due to the convexity assumptions on X and $\theta_\nu(\cdot, x^{-\nu})$, this minimization problem has a unique solution for every x which we denote by

$$y_\alpha(x) := \arg\min_{y \in X} \psi_\alpha(x, y).$$

It is not difficult to see that the function y_α is essentially the same function as the one defined in equation (2.9). Therefore, Theorem 2.2.3 c) implies that y_α is continuous. Moreover, from Proposition 2.2.4 we have that x^* is a normalized Nash equilibrium if and only if x^* is a fixed point of the mapping y_α, i.e., $x^* = y_\alpha(x^*)$. This puts into consideration Newton methods applied to the equation

$$y_\alpha(x) - x = 0 \tag{5.4}$$

in order to develop an algorithm for the computation of normalized Nash equilibria.

In general, the function y_α is not differentiable which causes difficulties. In the preceding chapter it was shown that y_α is semismooth, if, in addition to the above assumptions, the linear independence constraint qualification holds, see the proof of Theorem 4.2.5. Using this result, one can define a nonsmooth Newton method for the solution of the nonlinear equation (5.4) replacing the first derivative of y_α by generalized Jacobians. Yet, using the approach in the preceding chapter, it is difficult to calculate an element of the generalized Jacobian and, as a consequence, to show convergence of the Newton method. Therefore, we concern here an alternative approach which yields explicit formulas for a suitable substitute of the derivative of the function y_α. Additionally, we dispose with the linear independence constraint qualification and replace it by the somewhat weaker constant rank constraint qualification.

5.1 The Computable Generalized Jacobian

In this section we define a kind of replacement for the Jacobian of the function $y_\alpha(\cdot)$, which is related to the B-subdifferential, but easier to compute in practice.

The concept is similar to the computable generalized Jacobian introduced in [96] for the projection operator onto a closed convex set. The extension here for the function $y_\alpha(\cdot)$ is actually motivated to a large extent by the ideas from [96].

By definition, $y_\alpha(x)$ is the unique solution of the parameterized optimization problem

$$\begin{array}{ll} \text{minimize}_y \ \psi_\alpha(x,y) & \\ \text{subject to} \quad g_i(y) \le 0, \quad i = 1,\dots,m, \end{array} \tag{5.5}$$

where $\psi_\alpha : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is defined by

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

Then the KKT conditions for problem (5.5) can be written as

$$\begin{array}{l} \nabla_y \psi_\alpha(x,y) + \sum_{i=1}^{m} \lambda_i \nabla g_i(y) = 0, \\ \lambda_i \ge 0, \quad g_i(y) \le 0, \quad \lambda_i \cdot g_i(y) = 0, \quad i = 1,\dots,m. \end{array} \tag{5.6}$$

Note that $\nabla_y \psi_\alpha(x,y)$ is given by

$$\nabla_y \psi_\alpha(x,y) = \begin{pmatrix} \nabla_{x^1}\theta_1(y^1, x^{-1}) + \alpha(y^1 - x^1) \\ \vdots \\ \nabla_{x^N}\theta_N(y^N, x^{-N}) + \alpha(y^N - x^N) \end{pmatrix} \in \mathbb{R}^n. \tag{5.7}$$

Let

$$I_0(x) := \{ i \mid g_i(y_\alpha(x)) = 0 \} \tag{5.8}$$

be the index set of active constraints at $y = y_\alpha(x)$.

We adopt one of the main assumptions used in [96] which also appear in the context of piecewise differentiable functions.

Assumption 5.1.1 *The constant rank constraint qualification (CRCQ) holds at $y_\alpha(x)$, i.e., there exists a neighbourhood $N(y_\alpha(x))$ of $y_\alpha(x)$ such that for every set $J \subseteq I_0(x)$, the set of gradient vectors*

$$\{ \nabla g_i(y) \mid i \in J \}$$

has the same rank (which depends on J) for all $y \in N(y_\alpha(x))$.

The CRCQ is weaker than the linear independence constraint qualification. Moreover, it is always fulfilled in the case of linear constraints. Furthermore, due to a result by Janin [54], it is known that the CRCQ is a suitable constraint qualification in the sense that the satisfaction of CRCQ at the minimizer $y_\alpha(x)$ of problem (5.5) guarantees the existence (not necessarily uniqueness) of corresponding Lagrange multipliers λ such that the KKT conditions (5.6) hold. Hence the set

$$\mathcal{M}(x) := \{ \lambda \in \mathbb{R}^m \mid (y_\alpha(x), \lambda) \text{ satisfies (5.6)} \} \tag{5.9}$$

is always nonempty under Assumption 5.1.1.

There is a family of index sets that will play a crucial role in our analysis. For each $x \in \mathbb{R}^n$, define

$$\begin{aligned} \mathcal{B}(x) \quad := \quad \{J \subseteq I_0(x) \mid \nabla g_i(y_\alpha(x))\, (i \in J) &\text{ are linearly independent and} \\ \operatorname{supp}(\lambda) \subseteq J &\text{ for some } \lambda \in \mathcal{M}(x)\}, \end{aligned} \tag{5.10}$$

where $\operatorname{supp}(\lambda)$ denotes the support of the nonnegative vector $\lambda \in \mathbb{R}^m$, i.e.,

$$\operatorname{supp}(\lambda) := \{i \in \{1, \ldots, m\} \mid \lambda_i > 0\}.$$

We first claim that the family $\mathcal{B}(x)$ is always nonempty.

Lemma 5.1.2 *Suppose $\mathcal{M}(x) \neq \emptyset$. Then $\mathcal{B}(x) \neq \emptyset$.*

Proof. Let us choose a multiplier $\lambda \in \mathcal{M}(x)$ with minimal support. If $\operatorname{supp}(\lambda) = \emptyset$, we take $J := \emptyset$ and immediately see that $J \in \mathcal{B}(x)$. Now suppose $\operatorname{supp}(\lambda) \neq \emptyset$. We claim that $J := \operatorname{supp}(\lambda)$ belongs to $\mathcal{B}(x)$. Obviously, we have $\operatorname{supp}(\lambda) \subseteq J \subseteq I_0(x)$. Hence it remains to show that $\nabla g_i(y_\alpha(x))$ $(i \in J)$ are linearly independent. Suppose this is not true. Then there is a nonzero vector $\beta_J = (\beta_i)_{i \in J}$ such that

$$\sum_{i \in J} \beta_i \nabla g_i(y_\alpha(x)) = 0.$$

Replacing β_J by $-\beta_J$ if necessary, we may assume without loss of generality that at least one component β_i $(i \in J)$ is positive. Let $\tilde{t} := \min\{\lambda_i/\beta_i \mid \beta_i > 0\}$. Then we have $\lambda_i - \tilde{t}\beta_i \geq 0$ for all $i \in J$ and $\lambda_{i_0} - \tilde{t}\beta_{i_0} = 0$ for at least one index $i_0 \in J$. Now define

$$\tilde{\lambda}_i := \begin{cases} \lambda_i - \tilde{t}\beta_i, & i \in J, \\ \lambda_i, & i \notin J. \end{cases}$$

Then it follows immediately that the vector $\tilde{\lambda} = (\tilde{\lambda}_1, \ldots, \tilde{\lambda}_m)^T$ belongs to $\mathcal{M}(x)$. However, by construction, the support of $\tilde{\lambda}$ is strictly contained in the support of λ, a contradiction to our choice of λ. □

Recall that $\mathcal{M}(x) \neq \emptyset$ under Assumption 5.1.1, and hence the statement of Lemma 5.1.2 holds, in particular, in this situation.

For an index set $J \subseteq \{1,\dots,m\}$ with complement $\hat{J} := \{1,\dots,m\}\setminus J$, we now consider the function $\phi_\alpha(\cdot,\cdot,\cdot\,; J) : \mathbb{R}^{n+n+m} \to \mathbb{R}^{n+m}$ defined by

$$\phi_\alpha(x,y,\lambda;J) := \begin{pmatrix} \nabla_y \psi_\alpha(x,y) + \sum_{i\in J} \lambda_i \nabla g_i(y) \\ g_J(y) \\ \lambda_{\hat{J}} \end{pmatrix}, \tag{5.11}$$

where the partition $(J,\hat{J})$ of $\{1,\dots,m\}$ is used to split the vectors λ and $g(y)$ into $\lambda = (\lambda_J, \lambda_{\hat{J}})$ and $g(y) = (g_J(y), g_{\hat{J}}(y))$, respectively.

Lemma 5.1.3 *Let $x \in X$ and suppose that Assumption 5.1.1 holds. Furthermore, let $\mathcal{M}(x)$ be defined by* (5.9)*. Then, for any $J \in \mathcal{B}(x)$, there exists a unique vector $\lambda \in \mathcal{M}(x)$ such that $\phi_\alpha(x, y_\alpha(x), \lambda; J) = 0$.*

Proof. Let $J \in \mathcal{B}(x)$ and let $\lambda \in \mathcal{M}(x)$ be such that $\operatorname{supp}(\lambda) \subseteq J$. Then we have $\lambda = (\lambda_J, \lambda_{\hat{J}})$ with $\lambda_J \geq 0$ and $\lambda_{\hat{J}} = 0$. Since $(x, y_\alpha(x), \lambda)$ satisfies the KKT conditions (5.6), we have $\nabla_y \psi_\alpha(x, y_\alpha(x)) + \sum_{i\in J} \lambda_i \nabla g_i(y_\alpha(x)) = 0$ and $g_J(y_\alpha(x)) = 0$ (since $J \subseteq I_0(x)$). Hence $\phi_\alpha(x, y_\alpha(x), \lambda; J) = 0$ holds. Furthermore, the gradients $\nabla g_i(y_\alpha(x))$ $(i \in J)$ are linearly independent, which implies that λ is uniquely determined. □

We next show that, under certain assumptions, for any fixed x and $J \in \mathcal{B}(x)$, the system of equations $\phi_\alpha(x,y,\lambda;J) = 0$ has a locally unique solution $(y(x;J), \lambda(x;J))$.

Lemma 5.1.4 *Let $\bar{x} \in X$ be given, and suppose that Assumption 5.1.1 holds at $\bar{y} := y_\alpha(\bar{x})$. Let $J \in \mathcal{B}(\bar{x})$ be a fixed index set and $\bar{\lambda} \in \mathcal{M}(\bar{x})$ be the corresponding unique multiplier from Lemma 5.1.3 such that $\phi_\alpha(\bar{x}, \bar{y}, \bar{\lambda}; J) = 0$. Then the following statements hold:*

(a) There exist open neighbourhoods $N^J(\bar{x})$ of $\bar{x}$ and $N^J(\bar{y}, \bar{\lambda})$ of $(\bar{y}, \bar{\lambda})$, and a C^1-diffeomorphism $(y(\cdot\,;J), \lambda(\cdot\,;J)) : N^J(\bar{x}) \to N^J(\bar{y}, \bar{\lambda})$ such that $y(\bar{x};J) = \bar{y}$, $\lambda(\bar{x};J) = \bar{\lambda}$ and

$$\phi_\alpha(x, y(x;J), \lambda(x;J); J) = 0 \tag{5.12}$$

holds for all $x \in N^J(\bar{x})$.

(b) The transposed Jacobian of the function $y(\cdot\,;J)$ is given by the formula

$$\nabla y(x;J) = A^T C^{-1} - A^T C^{-1} D (D^T C^{-1} D)^{-1} D^T C^{-1}, \tag{5.13}$$

where

$$\begin{aligned} A = A(x;J) &:= -\nabla^2_{yx} \psi_\alpha(x, y(x;J)), \\ C = C(x;J) &:= \nabla^2_{yy} \psi_\alpha(x, y(x;J)) + \sum_{i\in J} \lambda_i(x;J) \nabla^2 g_i(y(x;J)), \\ D = D(x;J) &:= \nabla g_J(y(x;J)). \end{aligned}$$

Proof. (a) First note that, by Lemma 5.1.3, the pair $(\bar{y}, \bar{\lambda})$ is determined uniquely for any given $\bar{x}$ and $J \in \mathcal{B}(\bar{x})$. The Jacobian of $\phi_\alpha(\cdot\,; J)$ with respect to the variables (y, λ) is given by (after some reordering)

$$\nabla_{(y,\lambda)}\phi_\alpha(x, y, \lambda; J) = \begin{pmatrix} \nabla^2_{yy}\psi_\alpha(x,y) + \sum_{i\in J}\lambda_i\nabla^2 g_i(y) & \nabla g_J(y) & 0 \\ \nabla g_J(y)^T & 0 & 0 \\ 0 & 0 & I_{|\hat{J}|} \end{pmatrix}. \tag{5.14}$$

We claim that this matrix is nonsingular at $(x, y, \lambda) = (\bar{x}, \bar{y}, \bar{\lambda})$. Statement (a) is then an immediate consequence of the standard implicit function theorem. In fact, the nonsingularity follows from the observation that the Jacobian $\nabla g_J(\bar{y})$ has full rank by the choice of $J \in \mathcal{B}(\bar{x})$ together with the observation that Assumption 5.0.5 (a) implies the positive definiteness of the matrix $\nabla^2_{yy}\psi_\alpha(\bar{x}, \bar{y})$, whereas Assumption 5.0.5 (b) guarantees that the terms $\bar{\lambda}_i\nabla^2 g_i(\bar{y})$ are at least positive semidefinite for all $i \in J$.

(b) Differentiating equation (5.12) with respect to x and using some algebraic manipulations, it is not difficult to obtain the desired formula for the derivatives of the function $y(\cdot\,; J)$. The details are left to the reader. □

Our aim is to give a relation between the functions $y(\cdot\,; J)$ as defined in Lemma 5.1.4 and the function $y_\alpha(\cdot)$ that is the solution map of the parameterized optimization problem (5.5). More precisely, we will show that, under the same assumptions as in Lemma 5.1.4, there is a neighbourhood of the point $\bar{x}$ such that, for every x in this neighbourhood, there is an index set J (depending on the point x) such that $y_\alpha(x) = y(x; J)$ holds. This is made precise in the next lemma.

Lemma 5.1.5 *Let $\bar{x} \in X$ be given, and suppose that Assumption 5.1.1 holds at $\bar{y} := y_\alpha(\bar{x})$. Then there exists a neighbourhood $N(\bar{x})$ of $\bar{x}$ such that for all $x \in N(\bar{x})$, the following statements hold:*

(a) The CRCQ holds at $y_\alpha(x)$;

(b) $\mathcal{B}(x) \subseteq \mathcal{B}(\bar{x})$;

(c) at any given point $x \in N(\bar{x})$, the equality $y_\alpha(x) = y(x; J)$ holds for any index set $J \in \mathcal{B}(x)$, where $y(\cdot\,; J)$ is the function defined in Lemma 5.1.4.

Proof. (a) This follows from the definition of the CRCQ and the continuity of the function $y_\alpha(\cdot)$, cf. Proposition 2.2.3 (c).

(b) The proof is essentially the same as the one in [79] for the projection operator. Assume there exists no neighbourhood $N(\bar{x})$ of $\bar{x}$ such that $\mathcal{B}(x) \subseteq \mathcal{B}(\bar{x})$ for all

$x \in N(\bar{x})$. Then there is a sequence $\{x^k\}$ converging to $\bar{x}$ such that for each k, there is an index set $J_k \in \mathcal{B}(x^k) \setminus \mathcal{B}(\bar{x})$. Since there are only finitely many such index sets, by working with a subsequence if necessary, we may assume that these index sets J_k are the same for all k. Let this common index set be J.

According to the definition of $\mathcal{B}(x^k)$, the vectors $\nabla g_i(y_\alpha(x^k))$ $(i \in J)$ are linearly independent and there exists $\lambda^k \in \mathcal{M}(x^k)$ such that $\mathrm{supp}(\lambda^k) \subseteq J \subseteq I_0(x^k)$, but $J \notin \mathcal{B}(\bar{x})$. Due to the continuity of the functions g_i and y_α, it holds that $I_0(x^k) \subseteq I_0(\bar{x})$, hence we have $J \subseteq I_0(\bar{x})$ for all k sufficiently large. Furthermore, the assumed CRCQ condition guarantees that the vectors $\nabla g_i(y_\alpha(\bar{x}))$ $(i \in J)$ are also linearly independent. Hence we have $J \notin \mathcal{B}(\bar{x})$ only if there is no $\lambda \in \mathcal{M}(\bar{x})$ such that $\mathrm{supp}(\lambda) \subseteq J$. However, the KKT conditions imply that

$$\nabla_y \psi_\alpha(x^k, y_\alpha(x^k)) + \sum_{i \in J} \lambda_i^k \nabla g_i(y_\alpha(x^k)) = 0 \quad \text{for all } k. \tag{5.15}$$

Since the functions y_α and ∇g_i are continuous, we have $\nabla g_i(y_\alpha(x^k)) \to \nabla g_i(y_\alpha(\bar{x}))$. Taking into account the linear independence of $\{\nabla g_i(y_\alpha(\bar{x}))\}_{i \in J}$, we see that the sequence $\{\lambda^k\}$ is convergent, say $\lambda_i^k \to \check{\lambda}_i$ for all $i \in J$. Taking the limit in (5.15) and setting $\check{\lambda}_i = 0$ for $i \in \hat{J}$, we can easily verify that the vector $\check{\lambda} := (\check{\lambda}_J, \check{\lambda}_{\hat{J}})$ belongs to $\mathcal{M}(\bar{x})$. Moreover, the definition of $\check{\lambda}$ guarantees that $\mathrm{supp}(\check{\lambda}) \subseteq J$, and hence $J \in \mathcal{B}(\bar{x})$. This contradicts our assumption.

(c) From (a) and (b) it follows that there is a neighbourhood $N(\bar{x})$ such that for any $x \in N(\bar{x})$, the CRCQ holds at $y_\alpha(x)$ and $\mathcal{B}(x) \subseteq \mathcal{B}(\bar{x})$. Furthermore, for each $J \in \mathcal{B}(\bar{x})$, let $N^J(\bar{x})$ and $N^J(\bar{y}, \bar{\lambda})$ be the neighbourhoods defined in Lemma 5.1.4, where $\bar{\lambda}$ is the vector also defined there. We then define the neighbourhood

$$V(\bar{x}) := \bigcap_{J \in \mathcal{B}(\bar{x})} N^J(\bar{x}) \cap N(\bar{x}),$$

which is open since there are only finitely many J's.

For any given vector $x \in V(\bar{x})$, the optimization problem (5.5) has a unique solution $y_\alpha(x)$. Moreover, Lemma 5.1.3 implies that for every fixed $J \in \mathcal{B}(x)$, there exists a unique Lagrange multiplier $\lambda = \lambda^J(x) \in \mathcal{M}(x)$ such that $(x, y_\alpha(x), \lambda^J(x))$ satisfies

$$\phi_\alpha(x, y_\alpha(x), \lambda^J(x); J) = 0.$$

In particular, for the Lagrange multiplier associated with $(\bar{x}, \bar{y})$, we write $\lambda^J(\bar{x})$ as $\bar{\lambda}^J$.

On the other hand, Lemma 5.1.4 implies that there is a continuously differentiable function $(y(\cdot\,; J), \lambda(\cdot\,; J)) : N^J(\bar{x}) \to N^J(\bar{y}, \bar{\lambda}^J)$ such that, for every $x \in N^J(\bar{x})$, the pair $(y(x; J), \lambda(x; J))$ is the unique solution of

$$\phi_\alpha(x, y, \lambda; J) = 0 \tag{5.16}$$

in the set $N^J(\bar{y}, \bar{\lambda}^J)$. Hence, if we can show that there exists an open neighbourhood $U(\bar{x}) \subseteq V(\bar{x})$ such that for every $x \in U(\bar{x})$ and every $J \in \mathcal{B}(x)$ we have

$$(y_\alpha(x), \lambda^J(x)) \in N^J(\bar{y}, \bar{\lambda}^J),$$

then the uniqueness implies that

$$(y_\alpha(x), \lambda^J(x)) = (y(x; J), \lambda(x; J))$$

for all $x \in U(\bar{x})$ and $J \in \mathcal{B}(x)$, and this would conclude the proof.

Suppose there exists no such open neighbourhood $U(\bar{x}) \subseteq V(\bar{x})$. Then there exists a sequence $\{x^k\}$ with $x^k \to \bar{x}$ and $J_k \in \mathcal{B}(x^k)$ such that

$$(y_\alpha(x^k), \lambda^{J_k}(x^k)) \notin N^{J_k}(\bar{y}, \bar{\lambda}^{J_k}) \quad \text{for all } k. \tag{5.17}$$

By working with a subsequence, we may assume that J_k is the same index set for all k. Denote this index set by J. Furthermore, choose open neighbourhoods $N^J(\bar{y})$ of $\bar{y}$ and $N^J(\bar{\lambda}^J)$ of $\bar{\lambda}^J$ such that $N^J(\bar{y}) \times N^J(\bar{\lambda}^J) \subseteq N^J(\bar{y}, \bar{\lambda}^J)$.

Since the function y_α is continuous, we have $y_\alpha(x^k) \to y_\alpha(\bar{x}) = \bar{y}$. Hence $y_\alpha(x^k) \in N^J(\bar{y})$ for all k sufficiently large. On the other hand, for every x^k with associated $y_\alpha(x^k)$ and $\lambda^J(x^k)$, we have from (5.11)

$$\nabla_y \psi_\alpha(x^k, y_\alpha(x^k)) + \sum_{i \in J} \lambda_i^J(x^k) \nabla g_i(y_\alpha(x^k)) = 0, \tag{5.18}$$
$$\lambda_i^J(x^k) = 0, \quad i \in \hat{J}$$

for all k. The continuity of the functions $\nabla_y \psi_\alpha$, y_α and ∇g_i, together with the linear independence of the vectors $\nabla g_i(\bar{y})$ $(i \in J)$, which is again a consequence of the CRCQ, implies that the sequence $\{\lambda^J(x^k)\}$ is convergent. Let $\tilde{\lambda}^J$ be the corresponding limit point. Taking the limit in (5.18) therefore gives

$$\nabla_y \psi_\alpha(\bar{x}, \bar{y}) + \sum_{i \in J} \tilde{\lambda}_i^J \nabla g_i(\bar{y}) = 0$$

as well as $\tilde{\lambda}_i^J = 0$ for all $i \in \hat{J}$. Then the CRCQ implies that $\tilde{\lambda}^J$ is the only vector satisfying these equations. However, by definition, $\bar{\lambda}^J$ also satisfies these equations, so it follows that $\tilde{\lambda}^J = \bar{\lambda}^J$.

Hence $\lambda^J(x^k)$ converges to $\bar{\lambda}^J$, meaning that $\lambda^J(x^k) \in N^J(\bar{\lambda}^J)$ for all k sufficiently large. Therefore we have

$$(y_\alpha(x^k), \lambda^J(x^k)) \in N^J(\bar{y}) \times N^J(\bar{\lambda}^J) \subseteq N^J(\bar{y}, \bar{\lambda}^J)$$

for all k sufficiently large, a contradiction to (5.17). □

Since there are only finitely many possible index sets $J \subseteq \{1, \ldots, m\}$, it follows from Lemma 5.1.5 that, given any point x in a sufficiently small neighbourhood of $\bar{x}$, the function $y_\alpha(\cdot)$ is equal to one of the finitely many functions $y(\cdot\,; J)$ and, therefore, piecewise smooth. However it is not necessarily easy to compute an element of the B-subdifferential of y_α at x, which is defined by

$$\partial_B y_\alpha(x) := \{G \in \mathbb{R}^{n\times n} \mid G = \lim_{x^k \to x} \nabla y_\alpha(x^k)^T,\ \{x^k\} \subseteq \Omega\},$$

where $\Omega := \{x \in \mathbb{R}^n \mid y_\alpha(\cdot) \text{ is differentiable at } x\}$. Lemma 5.1.5 then suggests to use, in place of the B-subdifferential, the following modification of a generalized Jacobian, which we call the *computable generalized Jacobian* of $y_\alpha(\cdot)$ at x:

$$\partial_C y_\alpha(x) := \{\nabla y(x; J)^T \mid J \in \mathcal{B}(x)\}, \tag{5.19}$$

where $\mathcal{B}(x)$ is defined by (5.10). Note that computing an element of $\partial_C y_\alpha(x)$ amounts to finding an index set $J \in \mathcal{B}(x)$, which is implementable in practice. While the inclusion $\partial_B y_\alpha(x) \subseteq \partial_C y_\alpha(x)$ holds at any x, the converse is not true in general, see [22, Example 11]. Under additional assumptions, however, it can be shown that these two sets coincide; see, in particular, [**?**, Corollary 3.2.2] or [21, Theorem 3.3].

Nevertheless, we have the following result, which indicates that there is a good chance that the two sets actually coincide.

Proposition 5.1.6 *Let $\bar{x} \in \mathbb{R}^n$ be fixed and let the linear independence constraint qualification (LICQ) hold at $\bar{y} = y_\alpha(\bar{x})$, that is, the vectors $\nabla g_i(\bar{y})$ $(i \in I_0(\bar{x}))$ are linearly independent. Then for all but at most n values of α, the equality $\partial_B y_\alpha(\bar{x}) = \partial_C y_\alpha(\bar{x})$ holds.*

Proof. The inclusion $\partial_C y_\alpha(\bar{x}) \subseteq \partial_B y_\alpha(\bar{x})$ holds provided that the matrix $\nabla^2_{yx}\psi_\alpha(\bar{x}, \bar{y})$ is nonsingular, see [76, Corollary 3.2.2] or [21, Theorem 3.3].

Now, an elementary calculation shows that $\nabla^2_{yx}\psi_\alpha(\bar{x}, \bar{y})$ is written as $\nabla^2_{yx}\psi_\alpha(\bar{x}, \bar{y}) = W - \alpha I$, where the matrix $W \in \mathbb{R}^{n\times n}$ is given by

$$W = \begin{pmatrix} 0 & \nabla^2_{x^1 x^2}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^1 x^N}\theta_1(y^1, x^{-1}) \\ \nabla^2_{x^2 x^1}\theta_2(y^2, x^{-2}) & 0 & \cdots & \nabla^2_{x^2 x^N}\theta_2(y^2, x^{-2}) \\ \vdots & \vdots & \ddots & \vdots \\ \nabla^2_{x^N x^1}\theta_N(y^N, x^{-N}) & \nabla^2_{x^N x^2}\theta_N(y^N, x^{-N}) & \cdots & 0 \end{pmatrix}.$$

Since the matrix W has at most n different eigenvalues, the equality

$$\det(\nabla^2_{yx}\psi_\alpha(\bar{x}, \bar{y})) = \det(W - \alpha I) = 0$$

holds for at most n different values of α. This implies that $\nabla^2_{yx}\psi_\alpha(\bar{x}, \bar{y})$ is nonsingular for all but at most n values of α. □

Note that the previous result is the only one where we need the LICQ condition. Neither LICQ nor this result will be used in our subsequent analysis. It is stated here to give partial evidence that the gap between the computable generalized Jacobian and the B-differential is not so significant.

To conclude this section, we consider the special case of a GNEP with quadratic cost functions and linear constraints. Here the function $y_\alpha(\cdot)$ turns out to be piecewise linear (or piecewise affine, to be more precise).

Proposition 5.1.7 *Consider the case where the cost functions θ_ν are quadratic, i.e.,*

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

for $\nu = 1, \dots, N$. Suppose that the feasible set X is given by linear inequalities, i.e., $X := \{x \in \mathbb{R}^n \mid Bx \leq b\}$ for some matrix $B \in \mathbb{R}^{m\times n}$ and vector $b \in \mathbb{R}^m$. Let $\bar{x} \in X$ be arbitrarily given. Then there exists a neighbourhood $U(\bar{x})$ of $\bar{x}$ such that for every $x \in U(\bar{x})$ and every $J \in \mathcal{B}(x)$, there exist a matrix $V^J \in \mathbb{R}^{n\times n}$ and a vector $w^J \in \mathbb{R}^n$ such that $y_\alpha(x) = y(x; J) = V^J x + w^J$.

Proof. Since X is polyhedral, the CRCQ holds at every point $x \in X$. By Lemma 5.1.5, there exists a neighbourhood $N(\bar{x})$ of $\bar{x}$ such that for all $x \in N(\bar{x})$, we have $\mathcal{B}(x) \subseteq \mathcal{B}(\bar{x})$ and $y_\alpha(x) = y(x; J)$ for all $J \in \mathcal{B}(x)$, where $y(\cdot\,; J)$ is the function defined in Lemma 5.1.4.

Now consider an arbitrary index set $J \in \mathcal{B}(\bar{x})$, and let $y(\cdot\,; J)$ be the corresponding function. Furthermore, let $\bar{A}$ denote the $n \times n$ matrix $\bar{A} = (A_{\nu\mu})_{\nu,\mu=1}^N$ and $\mathrm{diag}(A_{\nu\nu})$ denote the block-diagonal matrix with block component matrices $A_{\nu\nu}$, $\nu = 1, \dots, N$. From Lemma 5.1.4, $y(\cdot\,; J)$ is a continuously differentiable function on $N(\bar{x})$ with Jacobian

$$V^J := \nabla y(x; J)^T = C^{-1}A - C^{-1}D(D^T C^{-1} D)^{-1} D^T C^{-1} A,$$

where $A := -\bar{A} + \mathrm{diag}(A_{\nu\nu}) + \alpha I$, $C := \mathrm{diag}(A_{\nu\nu}) + \alpha I$ and $D := B_J^T$. The assumptions on the cost functions θ_ν and the set X imply that the matrix V^J is constant. Consequently, $y(\cdot\,; J)$ is an affine function, i.e., there is a vector w^J such that $y(x; J) = V^J x + w^J$. □

Note that it follows from the above proof that we have

$$y_\alpha(x) \in \{V^J x + w^J \mid J \in \mathcal{B}(\bar{x})\}$$

for all x in a sufficiently small neighbourhood of $\bar{x}$, i.e., $y_\alpha(\cdot)$ is a piecewise affine function.

5.2 Newton's Method

For the computation of a normalized Nash equilibrium, we use the nonsmooth Newton method from [57] and apply it to the system of equations

$$F(x) := y_\alpha(x) - x = 0.$$

From the current iterate x^k, the next iterate x^{k+1} is computed by

$$x^{k+1} = x^k - H_k^{-1} F(x^k), \tag{5.20}$$

where H_k is an element of the nonempty computable generalized Jacobian

$$\partial_C F(x^k) = \partial_C y_\alpha(x^k) - I = \{\nabla y(x^k; J)^T - I \mid J \in \mathcal{B}(x^k)\}. \tag{5.21}$$

In this section, we give sufficient conditions for the matrices H_k to be nonsingular and show local superlinear/quadratic convergence of this nonsmooth Newton method.

For convenience, we write

$$M(x,y) := \begin{pmatrix} \nabla^2_{x^1 x^1}\theta_1(y^1, x^{-1}) & \cdots & \nabla^2_{x^1 x^N}\theta_1(y^1, x^{-1}) \\ \vdots & \ddots & \vdots \\ \nabla^2_{x^N x^1}\theta_N(y^N, x^{-N}) & \cdots & \nabla^2_{x^N x^N}\theta_N(y^N, x^{-N}) \end{pmatrix}.$$

This notation also facilitates the comparison with Newton methods from [32] which are based on a variational inequality formulation of the GNEP.

The following assumption will be needed to establish fast local convergence of the nonsmooth Newton method (5.20).

Assumption 5.2.1 *For each $J \in \mathcal{B}(x)$ and $\lambda \in \mathcal{M}(x)$, we have*

$$d^T\Big(M(x, y_\alpha(x)) + \sum_{i\in J} \lambda_i \nabla^2 g_i(y_\alpha(x))\Big)\, d > 0 \quad \forall d \in \mathcal{T}^J(x),\ d \neq 0, \tag{5.22}$$

where $\mathcal{T}^J(x)$ is defined by

$$\mathcal{T}^J(x) := \{d \in \mathbb{R}^n \mid \nabla g_i(y_\alpha(x))^T d = 0\ \forall i \in J\}. \tag{5.23}$$

The condition (5.22) is a kind of second order sufficiency condition. We will revisit this condition after showing the following nonsingularity result.

Lemma 5.2.2 *Let $\bar{x} \in X$ and $\bar{y} := y_\alpha(\bar{x})$. Suppose that Assumptions 5.1.1 and 5.2.1 hold at $\bar{x}$. Then the matrix $\nabla y(\bar{x}; J)^T - I$ is nonsingular for any index set $J \in \mathcal{B}(\bar{x})$.*

Proof. Assume that there exists an index set $J \in \mathcal{B}(\bar{x})$ such that the matrix $\nabla y(\bar{x}; J)^T - I$ is singular. Let $\bar{\lambda} \in \mathcal{M}(\bar{x})$ be the corresponding Lagrange multiplier, which is unique by Lemma 5.1.3 under Assumption 5.1.1, such that $\phi_\alpha(\bar{x}, \bar{y}, \bar{\lambda}; J) = 0$ holds. Furthermore, let $y(\cdot\,; J)$ and $\lambda(\cdot\,; J)$ be the functions defined in Lemma 5.1.4; in particular, recall that we have $y(\bar{x}; J) = \bar{y}$ and $\lambda(\bar{x}; J) = \bar{\lambda}$.

Since $\nabla y(\bar{x}; J)^T - I$ is singular, there exists a nonzero vector $v \in \mathbb{R}^n$ such that $(\nabla y(\bar{x}; J)^T - I)v = 0$, which is equivalent to saying that $\nabla y(\bar{x}; J)^T$ has an eigenvalue equal to one with eigenvector v. From Lemma 5.1.4, along with the fact that $y(\bar{x}; J) = \bar{y}$ and $\lambda(\bar{x}; J) = \bar{\lambda}$, we have the formula

$$\nabla y(\bar{x}; J)^T = C^{-1}A - C^{-1}D(D^T C^{-1} D)^{-1} D^T C^{-1} A, \tag{5.24}$$

with

$$\begin{aligned} A &= A(\bar{x}; J) := -\nabla^2_{yx}\psi_\alpha(\bar{x}, \bar{y}), \\ C &= C(\bar{x}; J) := \nabla^2_{yy}\psi_\alpha(\bar{x}, \bar{y}) + \sum_{i \in J} \bar{\lambda}_i \nabla^2 g_i(\bar{y}), \\ D &= D(\bar{x}; J) := \nabla g_J(\bar{y}). \end{aligned}$$

This expression of $\nabla y(\bar{x}; J)^T$ reveals immediately that $D^T \nabla y(\bar{x}; J)^T = 0_{m \times n}$, which implies that

$$0 = D^T \nabla y(\bar{x}; J)^T v = D^T v = \nabla g_J(\bar{y})^T v$$

holds, and thus,

$$v \in \mathcal{T}^J(\bar{x}), \tag{5.25}$$

where $\mathcal{T}^J(\bar{x})$ is given by (5.23) with $x = \bar{x}$. Therefore, multiplication of equation (5.24) from the left side with $v^T C$ and from the right side with v gives, using the fact that v is an eigenvector of the matrix $\nabla y(\bar{x}; J)^T$ with eigenvalue 1 once again,

$$v^T C v = v^T A v. \tag{5.26}$$

Note that the matrices C and A are expressed as

$$\begin{aligned} C &= \nabla^2_{yy}\psi_\alpha(\bar{x}, \bar{y}) + \sum_{i \in J} \bar{\lambda}_i \nabla^2 g_i(\bar{y}) \\ &= \begin{pmatrix} \nabla^2_{x^1 x^1}\theta_1(\bar{y}^1, \bar{x}^{-1}) & & \\ & \ddots & \\ & & \nabla^2_{x^N x^N}\theta_N(\bar{y}^N, \bar{x}^{-N}) \end{pmatrix} \\ &\quad + \alpha I + \sum_{i \in J} \bar{\lambda}_i \nabla^2 g_i(\bar{y}) \end{aligned}$$

and

$$A = -\nabla^2_{yx}\psi_\alpha(\bar{x}, \bar{y})$$

$$= -\begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(\bar{y}^1,\bar{x}^{-1}) & \cdots & \nabla^2_{x^1x^N}\theta_1(\bar{y}^1,\bar{x}^{-1}) \\ \vdots & \ddots & \vdots \\ \nabla^2_{x^Nx^1}\theta_N(\bar{y}^N,\bar{x}^{-N}) & \cdots & \nabla^2_{x^Nx^N}\theta_N(\bar{y}^N,\bar{x}^{-N}) \end{pmatrix}$$
$$+ \begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(\bar{y}^1,\bar{x}^{-1}) & & \\ & \ddots & \\ & & \nabla^2_{x^Nx^N}\theta_N(\bar{y}^N,\bar{x}^{-N}) \end{pmatrix} + \alpha I.$$

Hence we have

$$C - A = M(\bar{x},\bar{y}) + \sum_{i\in J} \bar{\lambda}_i \nabla^2 g_i(\bar{y}). \tag{5.27}$$

On the other hand, by (5.22) in Assumption 5.2.1 and (5.25), we have

$$v^T\left(M(\bar{x},\bar{y}) + \sum_{i\in J} \bar{\lambda}_i \nabla^2 g_i(\bar{y})\right) v > 0.$$

This together with (5.27) contradicts (5.26). □

The following example shows that Assumption 5.2.1 may hold even if the matrix $M(x, y_\alpha(x))$ is not positive semidefinite. Furthermore, it shows that Assumption 5.2.1 does not imply uniqueness of the normalized Nash equilibrium.

Example 5.2.3 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

$\min_{x_1}$	$\frac{1}{2}x_1^2 - 2x_1x_2 + x_1$	$\min_{x_2}$	$\frac{1}{2}x_2^2 - 2x_1x_2 + x_2$
s.t.	$x_1 + x_2 \geq 0$	s.t.	$x_1 + x_2 \geq 0.$

The cost functions are convex with respect to x^ν each, and the game satisfies Assumptions 5.0.5. The normalized Nash equilibria are $x^* = (x_1^*, x_2^*) = (1, 1)$ with Lagrange multiplier $\lambda^* = 0$, and $x^* = (0, 0)$ with $\lambda^* = 1$. We have

$$M(x, y_\alpha(x)) = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix},$$

and, for $x^* = (0, 0)$,

$$T(x^*) = \{d \in \mathbb{R}^2 \mid d = t\begin{pmatrix} 1 \\ -1 \end{pmatrix}, t \in \mathbb{R}\}.$$

This yields

$$d^T M d = 6t^2 > 0$$

for all $d \in T(x^*)$.

In the case of quadratic cost functions, there is a simple sufficient condition for Assumption 5.2.1 to hold.

Corollary 5.2.4 *Suppose that the cost functions θ_ν are given by*

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

for $\nu = 1, \dots, N$. Then Assumption 5.2.1 holds provided that the matrix $A := (A_{\nu\mu})_{\nu,\mu=1}^N$ is positive definite.

Next we prove that the matrices H_k provide a superlinear approximation for the function F.

Lemma 5.2.5 *Let x^* be a NoE. Suppose that Assumption 5.1.1 holds at x^*. Then we have for any $H \in \partial_C F(x)$*

$$F(x) - F(x^*) - H(x - x^*) = o(\|x - x^*\|). \tag{5.28}$$

Furthermore if the second derivatives of all θ_ν and all g_i are Lipschitz continuous around x^*, then

$$F(x) - F(x^*) - H(x - x^*) = O(\|x - x^*\|^2). \tag{5.29}$$

Proof. By Lemma 5.1.4, for each $J \in \mathcal{B}(x^*)$, there is a neighbourhood $N^J(x^*)$ of x^* and a continuously differentiable function $y(\cdot\,; J)$ defined on $N^J(x^*)$ such that $y(x^*; J) = y_\alpha(x^*) = x^*$. Let $\varepsilon > 0$ be arbitrarily given. Then the continuous differentiability of $y(\cdot\,; J)$ on $N^J(x^*)$ ensures the existence of a $\delta(\varepsilon, J) > 0$ such that

$$\frac{\|y(x; J) - y(x^*; J) - \nabla y(x; J)^T(x - x^*)\|}{\|x - x^*\|} < \varepsilon \tag{5.30}$$

holds whenever $\|x - x^*\| < \delta(\varepsilon, J)$. Let $\bar{\delta}(\varepsilon) := \min_{J\in\mathcal{B}(x^*)} \delta(\varepsilon, J) > 0$. Then (5.30) holds for any x such that $\|x - x^*\| < \bar{\delta}(\varepsilon)$ and any $J \in \mathcal{B}(x^*)$.

Now consider an arbitrary sequence $\{x^k\}$ converging to x^* and pick any $H_k \in \partial_C F(x^k)$. By the definition (5.21) of $\partial_C F(x)$, H_k can be written as $H_k = \nabla y(x^k, J_k)^T - I$ for some $J_k \in \mathcal{B}(x^k) \subseteq \mathcal{B}(x^*)$. Hence, from the preceding argument, we have

$$\begin{aligned} &\frac{\|F(x^k) - F(x^*) - H_k(x^k - x^*)\|}{\|x^k - x^*\|} \\ = \;& \frac{\|y_\alpha(x^k) - y_\alpha(x^*) - \nabla y(x^k; J_k)^T(x^k - x^*)\|}{\|x^k - x^*\|} \\ = \;& \frac{\|y(x^k; J_k) - y(x^*; J_k) - \nabla y(x^k; J_k)^T(x^k - x^*)\|}{\|x^k - x^*\|} < \varepsilon \end{aligned}$$

for all k such that $\|x^k - x^*\| < \bar{\delta}(\varepsilon)$. Since $\{x^k\}$ and ε are arbitrary, we may conclude that (5.28) holds.

If all functions θ_ν and g_i have Lipschitz continuous second derivatives, then for all $J \in \mathcal{B}(x^*)$ the function $\nabla y(\cdot\,; J)$ is locally Lipschitz continuous. This follows from formula (5.13) and the fact that the sum and the product of locally Lipschitz continuous functions again lead to a locally Lipschitz continuous function. Then it is not difficult to derive (5.29) in a similar manner to the above. □

Summarizing the above arguments, we are now in a position to state the main local convergence result which shows that our method is locally superlinearly/quadratically convergent. Note that this result holds under the CRCQ condition which is weaker than the linear independence constraint qualification.

Theorem 5.2.6 *Let x^* be a NoE and suppose that Assumptions 5.1.1 and 5.2.1 hold at x^*. Then there is a neighbourhood $N(x^*)$ of x^* such that for an arbitrary initial point $x^0 \in N(x^*)$, the sequence generated by the nonsmooth Newton method* (5.20) *converges to x^* superlinearly. Furthermore, if all the functions θ_ν and g_i have Lipschitz continuous second derivatives, then the convergence rate is quadratic.*

Proof. By Lemma 5.2.2, each $H \in \partial_C F(x^*) = \{\nabla y(x^*; J)^T - I \mid J \in \mathcal{B}(x^*)\}$ is nonsingular. Since the functions $\nabla y(\cdot\,; J)$ $(J \in \mathcal{B}(x^*))$ are continuous, there exists a neighbourhood $N(x^*)$ of x^* such that $\mathcal{B}(x^k) \subseteq \mathcal{B}(x^*)$ and hence the matrices $H_k \in \partial_C F(x^k) = \{\nabla y(x^k; J)^T - I \mid J \in \mathcal{B}(x^k)\}$ are nonsingular for all $x^k \in N(x^*)$. The rest of the proof consists of standard arguments based on Lemma 5.2.5 and the definition of the nonsmooth Newton method (5.20). □

Our final result shows that the nonsmooth Newton method enjoys a local one-step convergence property if the GNEP is described by quadratic cost functions and linear constraints.

Proposition 5.2.7 *Suppose that the cost functions and the constraints are given as in Proposition 5.1.7 and that the matrix $A := (A_{\mu\nu})_{\mu,\nu=1}^N$ is positive definite. Let x^* be a NoE. Then there is a neighbourhood $N(x^*)$ of x^* such that, once x^k enters $N(x^*)$, the next iterate x^{k+1} coincides with x^*.*

Proof. By Lemma 5.1.5, there exists a neighbourhood $N(x^*)$ of x^* such that for every $x \in N(x^*)$ and every $J \in \mathcal{B}(x)$, we have $y_\alpha(x) = y(x; J)$ and $\mathcal{B}(x) \subseteq \mathcal{B}(x^*)$. Moreover, from Proposition 5.1.7, we have $y(x; J) = V^J x + w^J$ for all $x \in N(x^*)$ with V^J and w^J being some constant matrix and vector, respectively. Define the function $F(\cdot\,; J)$ on $N(x^*)$ by $F(x; J) := y(x; J) - x$.

Let $x^k \in N(x^*)$ and $J_k \in \mathcal{B}(x^k)$. Since $y(\cdot\,; J_k)$ is an affine function on $N(x^*)$, Taylor's formula yields

$$F(x^*; J_k) = F(x^k; J_k) + F'(x^k; J_k)(x^* - x^k), \tag{5.31}$$

where $F'(\cdot\,; J_k) = V^{J_k} - I$ is the Jacobian of $F(\cdot\,; J_k)$, which is nonsingular from Lemma 5.2.2 and Corollary 5.2.4. Since $\mathcal{B}(x^k) \subseteq \mathcal{B}(x^*)$, we have $F(x^*; J_k) = y(x^*; J_k) - x^* = y_\alpha(x^*) - x^* = F(x^*) = 0$ by Lemma 5.1.5 (c) and Proposition 2.2.4. Exploiting the nonsingularity of $F'(x^k; J)$, we then obtain from (5.31) that

$$x^* = x^k - F'(x^k; J_k)^{-1} F(x^k; J_k).$$

The right-hand side is precisely the Newton iteration at x^k, and hence x^{k+1} coincides with the NoE x^*. □

Chapter 6

Applications and Numerical Results

This chapter is concerned with four implementations of the numerical methods developed in the preceding chapters and their application to some generalized Nash games. In particular, we consider a Barzilai-Borwein method for the solution of the unconstrained optimization reformulation (2.14), the relaxation method defined in Algorithm 3.2.1, a Newton method for solving the unconstrained optimization problem (2.14), and a Newton method (5.20) that solves a fixed point formulation of the GNEP. We illustrate the numerical performance of these methods with five examples of generalized Nash games taken from the literature and the electricity market example from the introduction.

6.1 Implementations

All implementations are done in MATLAB using the solver `SNOPT` from the TOMLAB package in order to calculate the values of $y_\alpha(x)$ and $y_\beta(x)$, respectively. In order to compare the performance of the different methods, we use a similar stopping criterion for all algorithms. Namely, we require that $\|y_\alpha(x) - x\| < \varepsilon$ with $\varepsilon := 10^{-8}$ for the first order methods, that is, the Barzilai-Borwein method and the relaxation method, and $\varepsilon := 10^{-12}$ for the Newton-type methods, where the parameter α is set to 10^{-4} for all methods except the Newton method solving the unconstrained optimization reformulation, where it is set to 10^{-2}. Since the examples have a rather simple structure, all algorithms are terminated by force if a maximum of 15 iterations is reached.

6.1.1 Barzilai Borwein Method

Here we illustrate the performance of a first-order numerical method for the solution of the unconstrained optimization problem

$$\min V_{\alpha\beta}(x) \quad \text{s.t. } x \in \mathbb{R}^n,$$

in order to compute a normalized Nash equilibrium, see the definition of $V_{\alpha\beta}$ in (2.12) and some further theoretical properties of $V_{\alpha\beta}$ at the end of chapter 3.1.

We choose the Barzilai-Borwein (BB) gradient method [8], see also [87, 88, 35, 18, 41] for some subsequent modifications and investigations of this method, for the unconstrained minimization of the objective function $V_{\alpha\beta}$. This method uses the iterative procedure

$$x^{k+1} := x^k - \alpha_k \nabla V_{\alpha\beta}(x^k), \quad k = 0, 1, 2, \dots$$

with the stepsize

$$\alpha_k := \frac{y^T s}{y^T y},$$

where

$$s := x^k - x^{k-1}, \quad y := \nabla V_{\alpha\beta}(x^k) - \nabla V_{\alpha\beta}(x^{k-1}).$$

Hence the BB method has the advantage of using an explicit formula for the stepsize. So no extra line search is required which would be very expensive in our case since this would require further evaluations of the mapping $V_{\alpha\beta}$. Each function evaluation of $V_{\alpha\beta}$, however, needs the solution of two constrained optimization problems in order to compute $y_\alpha(x)$ and $y_\beta(x)$.

We use the parameters $\alpha = 10^{-4}$ and $\beta = 5 \cdot 10^{-4}$ for all test examples, and terminate the iteration if either $\|y_\alpha(x^k) - x^k\| < 10^{-8}$ or the iteration number exceeds 15. We state the value of $V_{\alpha\beta}(x^k)$, which shows that the objective function value decreases in nearly every iteration despite the lack of a line search. At first view, it seems odd not to use the value of $V_{\alpha\beta}$ for termination criterion. However, the value of $V_{\alpha\beta}$ depends strongly on the choice of α and β, so choosing α and β close to equality might cause the algorithm to stop earlier than for other values of α and β, even if the distance to solution is greater. In any case, it is easier to compare the Barzilai-Borwein method with the other methods using the stopping criterion based on $\|y_\alpha(x^k) - x^k\|$.

6.1.2 Relaxation Method

Here we implemented Algorithm 3.2.1 with the modified stepsize rule from Remark 3.2.5. The method is terminated whenever $\|y_\alpha(x^k) - x^k\| < \varepsilon$ with $\varepsilon := 10^{-8}$

and uses the parameters $\alpha = 10^{-4}$, $\beta = 0.5$ and $\sigma = 10^{-4}$. For the electricity market example we also tested the method setting $\sigma = 1$.

6.1.3 Newton's Method based on Optimization Reformulation

This is the Newton-type method described through equations (4.16) and (4.17) for the solution of the unconstrained optimization reformulation of the GNEP (2.14).

To this end, we need to calculate elements from $\partial_B^2 V_{\alpha\beta}(x^k)$. This is not an easy task, however, all examples except Rosen's example satisfy, in addition to the linear independence constraint qualification, strict complementarity at the solution, which results in differentiability of the function $\nabla V_{\alpha\beta}$, so we simply compute the Hessian of $V_{\alpha\beta}(x^k)$ in each iteration. We use an Armijo-type line search in order to globalize this method. Moreover, we switch to the steepest descent direction whenever the generalized Newton direction is not computable or does not satisfy a sufficient decrease condition. In our experiments, however, we were always able to take the generalized Newton direction. The method is terminated whenever $\|y_\alpha(x^k) - x^k\| < \varepsilon$ with $\varepsilon := 10^{-12}$ and uses the two parameters $\alpha = 0.01$ and $\beta = 0.05$ for the definition of $V_{\alpha\beta}$.

6.1.4 Newton's Method through Fixed Point Formulation

Here we implemented the Newton method according to the iterative scheme (5.20). The parameter α is set to 10^{-4} for all test runs and the iteration is stopped whenever $\|y_\alpha(x^k) - x^k\| < 10^{-12}$.

To calculate an element of the computable generalized Jacobian of y_α at x^k, we need to find an index set $J \in \mathcal{B}(x^k)$ together with a corresponding multiplier λ^k. This is an easy task if the linear independence constraint qualification (LICQ) holds at the minimum $y_\alpha(x^k)$. In this case, we can take, for example, $J := I_0(x^k)$, where $I_0(x)$ is defined by (5.8). However, since LICQ is not needed in the convergence theory of this method, we have to find a way to compute J and λ^k under the weaker CRCQ assumption. To this end, consider the linear program

$$\begin{aligned} &\min_\lambda \sum_{i \in I_0} \lambda_i \\ &\text{s.t. } \nabla g(y_\alpha(x^k))\,\lambda = -\nabla_y \psi_\alpha(x^k, y_\alpha(x^k)), \\ &\qquad \lambda_i \geq 0 \quad \forall\, i \in I_0, \\ &\qquad \lambda_i = 0 \quad \forall\, i \in \{1, \ldots, m\} \setminus I_0, \end{aligned} \tag{6.1}$$

where $I_0 := I_0(x^k)$. Since CRCQ holds at $y_\alpha(x^k)$, it follows that $\mathcal{M}(x^k)$ is nonempty, and hence (6.1) has at least one feasible point. Moreover, the objective function is obviously bounded from below on the feasible set. Standard linear programming theory then shows that (6.1) is solvable; moreover, at least one of the vertices of the polyhedron defined by the feasible set of (6.1) is also a solution. Now, let λ^k be such a vertex solution of (6.1). Then, again by standard results for linear programs, it follows that the gradients $\nabla g_i(y_\alpha(x^k))$ corresponding to the positive components $\lambda_i^k > 0$ are linearly independent. This proves the following result.

Lemma 6.1.1 *Suppose that the CRCQ (or any other constraint qualification) holds at $y_\alpha(x^k)$. Let λ^k be a vertex solution of the linear program* (6.1) *and define $J := \{i \in I_0 \mid \lambda_i^k > 0\}$. Then J belongs to $\mathcal{B}(x^k)$.*

Note that, in principle, a vertex solution of the linear program (6.1) can be calculated by the simplex method. It should be noted, however, that the linear program (6.1) is not given in standard form since the rows of the constraint matrix may be linearly dependent. Typically, implementations of the simplex method deal with this problem automatically. Alternatively, one could modify (6.1) like in the Big-M method to get an equivalent linear program which satisfies the full row rank condition.

6.2 Examples of Generalized Nash Equilibrium Problems

In order to highlight some distinguishing properties of the numerical methods described before, we selected five Nash games from the literature as test examples in addition to the electricity market model from the introduction. These examples can be classified according to the structure of cost functions and constraints. Examples 6.2.2, 6.2.4 and 6.2.6 are linear-quadratic games, that is, the cost functions are quadratic with linear terms. Such games have been discussed in Proposition 3.1.2, Corollary 3.1.7 and Proposition 5.1.7, though without linear terms. It is not difficult to see that the conclusions of these propositions still hold with additional linear terms in the cost function. Examples 6.2.3 and 6.2.5, as well as the electricity market example from the introduction, have in common that the cost functions include a particular function that is subject of the following Lemma.

Lemma 6.2.1 *Let $x^\mu \in \mathbb{R}$ for $\mu \in \{1,\dots,N\}$ and consider for fixed $\nu \in \{1,\dots,N\}$ the function*

$$f_\nu(x^\nu, x^{-\nu}) := \delta_\nu x^\nu - \frac{x^\nu}{(\sum_{\mu=1}^N x^\mu)^\gamma},$$

where δ_ν and γ are some real parameters. Assume that $x^\mu \geq 0$ for all $\mu \in \{1,\dots,N\}$ and $\sum_{\mu=1}^N x^\mu > 0$. Then f_ν is well-defined and for $0 < \gamma < 1$, the function f_ν is strictly convex with respect to the variable x^ν and convex if $\gamma = 1$.

Proof. We calculate the partial derivatives of f_ν. This yields

$$\begin{aligned}
\nabla_{x^\nu} f_\nu(x) &= \delta - (\sum_{\mu=1}^N x^\mu)^{-\gamma} + \gamma x^\nu (\sum_{\mu=1}^N x^\mu)^{-\gamma-1},\\
\nabla^2_{x^\nu x^\nu} f_\nu(x) &= 2\gamma(\sum_{\mu=1}^N x^\mu)^{-\gamma-1} - (\gamma^2+\gamma)x^\nu(\sum_{\mu=1}^N x^\mu)^{-\gamma-2},\\
\nabla^2_{x^\nu x^\xi} f_\nu(x) &= \gamma(\sum_{\mu=1}^N x^\mu)^{-\gamma-1} - (\gamma^2+\gamma)x^\nu(\sum_{\mu=1}^N x^\mu)^{-\gamma-2} \quad \text{for } \xi \neq \nu.
\end{aligned}$$

In order to prove strict convexity of the function f_ν with respect to x^ν it is sufficient to show that $\nabla^2_{x^\nu x^\nu} f_\nu(x) > 0$. We arrive at

$$\begin{array}{rrl}
& 0 < \gamma < 1 & \\
\Rightarrow & 2\gamma & > \gamma^2 + \gamma \\
\Rightarrow & 2\gamma(\sum_{\mu=1}^N x^\mu)(\sum_{\mu=1}^N x^\mu)^{-1-\gamma} & > (\gamma^2+\gamma)x^\nu(\sum_{\mu=1}^N x^\mu)^{-1-\gamma} \\
\Rightarrow & \nabla^2_{x^\nu x^\nu} f_\nu(x) & > 0
\end{array}$$

where we used $\sum_{\mu=1}^N x^\mu > 0$ and $x^\mu \geq 0 \; \forall \mu = 1, \ldots, N$, in the last but one inequality. The case $\gamma = 1$ follows analogue with '≥' instead of '>'.

□

The numerical results indicate that the local convergence performance of all methods is connected to the properties of the matrix

$$M(x^*) := \begin{pmatrix} \nabla^2_{x^1x^1}\theta_1(x^*) & \ldots & \nabla^2_{x^1x^N}\theta_1(x^*) \\ \nabla^2_{x^2x^1}\theta_2(x^*) & \ldots & \nabla^2_{x^2x^N}\theta_2(x^*) \\ \vdots & \ddots & \vdots \\ \nabla^2_{x^Nx^1}\theta_N(x^*) & \ldots & \nabla^2_{x^Nx^N}\theta_N(x^*) \end{pmatrix}. \tag{6.2}$$

From a theoretical point of view, if $M(x^*)$ is positive definite, then Assumption 3.1.4 holds in a neighbourhood of a normalized Nash equilibrium by Theorem 3.1.6, which is essential for the relaxation method. Furthermore, this assumption implies that the Newton method based on the fixed point formulation has local superlinear convergence, see Lemma 5.2.2. In practice, the condition number of the matrix $M(x^*)$ seems to influence the efficiency of the first-order methods in a similar way the Hessian of the objective function does in optimization problems. The details are discussed with in the description of the respective examples.

Example 6.2.2 This test problem is the river basin pollution game taken from [45] and [61]. The game consists of three players with cost functions

$$\theta_\nu(x) = (d_2 \sum_{\mu=1}^{3} x^\mu + c_{1\nu} + c_{2\nu} x^\nu - d_1) x^\nu,$$

where $x^\nu \in \mathbb{R}_+$ for $\nu = 1, 2, 3$. Furthermore, the players face two joint constraints, $l = 1, 2$, of the form

$$q_l(x) = \sum_{\mu=1}^{3} u_{\mu l} e_\mu x^\mu \leq K_l.$$

All constants are given in Table 6.1.

Player ν	$c_{1\nu}$	$c_{2\nu}$	e_ν	$u_{\nu 1}$	$u_{\nu 2}$	d_1	d_2	K_1	K_2
1	0.10	0.01	0.50	6.5	4.583	3.0	0.01	100	100
2	0.12	0.05	0.25	5.0	6.250				
3	0.15	0.01	0.75	5.5	3.750				

Table 6.1: Constants for the river basin pollution game

This game is a Cournot-type model. In [61], the authors describe the following economic interpretation: Each player ν represents a company engaged in some economic activity, for instance paper pulp production, at a level x^ν. The companies sell their product on the same market and the price on that market depends on the total output on the market. Thus, the revenue function, and therefore the profit function of each agent, which equates to the negative cost function defined above, depends on the production level of the rival companies.

The joint constraints q_l arise from a limitation on environmental damage induced by the players' activities. In this particular example, the companies are located along a river and expel pollutions from production into the river. The pollution level of the river is monitored by two measurement stations along the river. At each station, the local authority sets a limit on pollutant concentration, which results in the joint constraints q_l, $l = 1, 2$.

Since this game has quadratic cost functions with linear terms, the matrix $M(x^*)$ defined in equation (6.2) is

$$M(x^*) = \begin{pmatrix} 0.04 & 0.01 & 0.01 \\ 0.01 & 0.12 & 0.01 \\ 0.01 & 0.01 & 0.04 \end{pmatrix},$$

which is positive definite with condition number $\text{cond}(M) = 4.1$. Thus, in view of Corollary 3.1.7 the sufficient conditions for convergence of the relaxation method hold. Moreover, the matrix B defined in Proposition 3.1.2 is positive definite, implying that for small α the function V_α is convex. Thus, we may assume that the relaxation method works quite well.

The local convergence of the Newton method based on the fixed point formulation is ensured by Lemma 5.2.2. Moreover, Proposition 5.2.7 implies that the Newton method based on the fixed point formulation terminates after a finite number of iterations for suitable starting points. We actually observe convergence in just one step for both Newton methods. The numerical results can be found in Table 6.5.

Example 6.2.3 This test problem is the internet switching model introduced by Kesselman et al. [56] and further analyzed by Facchinei et al. [32]. The cost function of each user is given by

$$\theta_\nu(x) = \frac{x^\nu}{B} - \frac{x^\nu}{\sum_{\mu=1}^N x^\mu},$$

with constraints $x^\nu \geq 0.01$, $\nu = 1, \ldots, N$, and a joint constraint of the form

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

The constraints have been slightly modified from those in [56] to ensure that the cost functions θ_ν are defined on the whole feasible set. The exact solution of this problem is $x^* = (0.9, \ldots, 0.9)^T$.

This is the only example where the full step size $t_k = 1$ was never accepted in the relaxation method. Using our line search globalization, however, we observe very fast linear convergence (see Table 6.6). 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 Tables 6.7 and 6.8. Both relaxation method and Barzilai-Borwein method have difficulties here, opposed to the Newton methods.

◇

Example 6.2.4 Here we consider a simple two-player game originally suggested by Rosen [92]. The example has the two cost functions

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

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 $y_\alpha(x^*) = (1,0)^T$ and $\lambda_\alpha(x^*) = (0,0,1)^T$ for arbitrary parameter $\alpha > 0$, hence strict complementarity does not hold in the second component. Table 6.9 shows our corresponding numerical results. ◇

Example 6.2.5 This test problem is a Cournot oligopoly problem with shared constraints and nonlinear cost functions, which was first described in [69] as a standard Nash game and later in [75, p. 233] with additional joint constraints.

The model considers a number of N firms competing on the same market. Each company chooses a production output $x^\nu \in \mathbb{R}_+$ so as to maximize her profit function

$$\theta_\nu(x^\nu, x^{-\nu}) = p(x^\nu, x^{-\nu}) \cdot x^\nu - f_\nu(x^\nu),$$

where the market price p is given by the inverse demand function

$$p(x^\nu, x^{-\nu}) = 5000^\gamma \cdot (\sum_{\mu=1}^{N} x^\mu)^{-\gamma}$$

and the cost function of firm ν is

$$f_\nu(x^\nu) = c_\nu \cdot x^\nu + \frac{\beta_\nu}{\beta_\nu + 1} K_\nu^{-\frac{1}{\beta_\nu}} \cdot (x^\nu)^{\frac{\beta_\nu+1}{\beta_\nu}}.$$

The shared constraint imposed on joint production is

$$\sum_{\mu=1}^{N} x^\mu \leq P.$$

In fact, in order to guarantee that all functions are well-defined, one should add an additional constraint $\sum_{\mu=1}^{N} x^\mu \geq \varepsilon$ with some small constant ε. However, since this does not change the solution and no difficulties arise in the numerical solution of the problem, we omit this.

In accordance with [69] and [75] we consider five players and choose $\gamma = \frac{1}{1.1}$. The remaining constants are given in Table 6.2, except for the parameter P, which we vary.

It is not difficult to see that the negative of the profit function θ_ν can be written as the sum of a strictly convex function and a function of the type defined in Lemma 6.2.1. Lemma 6.2.1 i) then implies that the negated profit function θ_ν is strictly convex with respect to player ν's variable. The numerical results can be found in Tables 6.10, 6.11, 6.12 and Table 6.13.

◇

Firm	c_ν	K_ν	β_ν
1	10	5	1.2
2	8	5	1.1
3	6	5	1.0
4	4	5	0.9
5	2	5	0.8

Table 6.2: Parameter specifications for the Cournot Oligopoly

Example 6.2.6 We solve the electricity market problem suggested by Contreras et al. [15], case study 1. This model involves three power generating companies owning one, two, and three power plants, respectively. It is different to the electricity market model from the introduction in that it does not involve a power line network, in particular, there is only one type of consumers with linear demand function. To avoid confusion, we refer to the present example as 'electricity market example I', while 'electricity market example II' refers to the next example.

The decision variable x_i^ν of company ν determines the electricity generated with power plant i. Let n_ν be the number of power plants owned by company ν. Each power plant has a capacity limit $P_{\nu i}^{\max}$ incurring a production limit $0 \leq x_i^\nu \leq P_{\nu i}^{\max}$

The profit function of company ν is given by

$$\theta_\nu(x^\nu, x^{-\nu}) = p(x^\nu, x^{-\nu}) \cdot \sum_{i=1}^{n_\nu} x_i^\nu - \sum_{i=1}^{n_\nu} f_{\nu i}(x_i^\nu),$$

where the inverse demand function p is given by

$$p(x^\nu, x^{-\nu}) = 378.4 - 2 \sum_{\nu=1}^{3} \sum_{i=1}^{n_\nu} x_i^\nu,$$

and the cost function $f_{\nu i}$ of power plant i owned by player ν is

$$f_{\nu i}(x^\nu, x^{-\nu}) = \frac{1}{2} c_{\nu i} \cdot (x_i^\nu)^2 + d_{i\nu} x_i^\nu + e_{i\nu}.$$

The constants are specified in Table 6.3.

In this game the restriction is only imposed on electricity generation of a particular company, thus, this is a standard Nash equilibrium problem. For the numerical solution of the problem we take once again the negated profit function of a player as cost function. While the matrix $M(x^*)$ is positive definite, its condition number is quite bad (around 610), which results in slow convergence of the first order methods. In fact, the relaxation method was extremly slow with the choice

company ν	generator i	$c_{\nu i}$	$d_{\nu i}$	$e_{i\nu}$	$P_{\nu i}^{\min}$	$P_{\nu i}^{\max}$
1	1	0.04	2	0	0	80
2	1	0.035	1.75	0	0	80
	2	0.125	1	0	0	50
3	1	0.0166	3.25	0	0	55
	2	0.05	3	0	0	30
	3	0.05	3	0	0	40

Table 6.3: Parameter specifications for the electricity market problem

of $\sigma = 10^{-4}$, see Table 6.14. Choosing $\sigma = 1$ gives better results, cf. Table 6.15. The numerical results also show that the Barzilai-Borwein method works suprisingly well. The Newton methods have no problem with this example, as the results in Table 6.15 show.

Example 6.2.7 Here we solve the electricity market example from the introduction, which we call 'electricity market example II' in the numerical results section. The game has two players, A and B, each controlling 5 variables. Player A controls $x^A \in \mathbb{R}^4$ and $y^A \in \mathbb{R}$, while player B controls x^B and y^B. In order to implement the method we change the names of the variables, in that we define

$$x := (x^A, y^A, x^B, y^B),$$

in particular, $x_5 = y^A$ and $x_{10} = y^B$. Parameters are chosen as follows:

i	e_i	k_i	C_i	c^A	c^B	γ
1	5	300	30000	26	28	$\frac{1}{1.1}$
2	2	300	50000			
3	2	300	40000			
4	2	300	30000			

Table 6.4: Parameter specification for the electricity market model II

Since the profit functions and constraints have been defined in the introduction already, we do not state them here. Instead of the profit function we consider of course the negative of the profit function so as to have minimization problems. We proceed with some considerations about convergence conditions for this particular example.

For this example, the matrix M from equation (6.2) has the structure

$$M(x) = \begin{pmatrix} & & 0 & & 0 \\ & A(x) & \vdots & B(x) & \vdots \\ 0 & \dots & 0 & \dots & 0 \\ & C(x) & \vdots & D(x) & \vdots \\ 0 & \dots & 0 & \dots & 0 \end{pmatrix},$$

where $A(x), B(x), C(x)$ and $D(x) \in \mathbb{R}^{4\times 4}$ are diagonal matrices with entries

$$\begin{aligned} A_{ii}(x) &= 2\gamma p_i(x)(x_i + x_{i+5})^{-1} - (\gamma^2 + \gamma)p_i(x)(x_i + x_{i+5})^{-2}x_i, \\ B_{ii}(x) &= \gamma p_i(x)(x_i + x_{i+5})^{-1} - (\gamma^2 + \gamma)p_i(x)(x_i + x_{i+5})^{-2}x_i, \\ C_{ii}(x) &= \gamma p_i(x)(x_i + x_{i+5})^{-1} - (\gamma^2 + \gamma)p_i(x)(x_i + x_{i+5})^{-2}x_{i+5}, \\ D_{ii}(x) &= 2\gamma p_i(x)(x_i + x_{i+5})^{-1} - (\gamma^2 + \gamma)p_i(x)(x_i + x_{i+5})^{-2}x_{i+5}. \end{aligned}$$

This immediately shows that the matrix $M(x)$ is not positive definite for any x, not even nonsingular. However, there is a possibility to derive a condition which implies that Assumption 5.22 holds, the latter being important to show convergence of the Newton method based on the fixed point reformulation. To this end, we formulate the following Lemma.

Lemma 6.2.8 *Suppose that $0 < \gamma < 1$, $x \in \mathbb{R}^{10}$, and $x > 0$ such that $x_i \leq 3x_{i+5}$ and $x_{i+5} \leq 3x_i$ for all $i = 1, 2, 3, 4$. Then the matrix*

$$\begin{pmatrix} A(x) & B(x) \\ C(x) & D(x) \end{pmatrix}$$

is positive definite.

Proof. With the Theorem of Gerschgorin, it follows easily that all eigenvalues of the matrix in question are positive. However, this does not imply that the matrix itself is positive definite, since the matrix is not symmetric.

We write $a_i := A_{ii}(x)$, $b_i = B_{ii}(x), c_i = C_{ii}(x)$ and $d_i = D_{ii}(x)$. With this notation we have, for $z \neq 0$ arbitrary,

$$\begin{aligned} z^T \begin{pmatrix} A(x) & B(x) \\ C(x) & D(x) \end{pmatrix} z &= \sum_{i=1}^{4} [z_i(a_i z_i + b_i z_{i+4}) + z_{i+4}(c_i z_i + d_i z_{i+4})] \\ &= \sum_{i=1}^{4} [a_i z_i^2 + (b_i + c_i) z_i z_{i+4} + d_i z_{i+4}^2] \\ &\geq \sum_{i=1}^{4} \min\{a_i, d_i\}(z_i^2 + z_{i+4}^2) - |b_i + c_i||z_i z_{i+4}| \\ &\geq \sum_{i=1}^{4} \big(\min\{a_i, d_i\} - \max\{|b_i|, |c_i|\}\big)(z_i^2 + z_{i+4}^2). \end{aligned}$$

In the remainder we show that $\min\{a_i, d_i\} - \max\{|b_i|, |c_i|\} > 0$ for all γ and x satisfying the assumptions. For simplicity, we write

$$\delta := p_i(x)(x_i + x_{i+5})^{-1}.$$

Note that $\delta > 0$ for all $x > 0$. It follows that

$$a_i - b_i = \gamma\delta > 0$$

and

$$\begin{aligned} a_i + b_i &= 3\gamma\delta - (\gamma^2 + \gamma)\delta \cdot 2 \cdot \tfrac{x_i}{x_i + x_{i+5}} \\ &\geq 3\gamma\delta - (\gamma^2 + \gamma)\delta \cdot 2 \cdot \tfrac{3}{4} \\ &> 0, \end{aligned}$$

where we used the assumption that $x_i \leq 3x_{i+5}$ and $\gamma < 1$. Thus it follows that $a_i - |b_i| > 0$ holds for all $i = 1, 2, 3, 4$. Further we get

$$\begin{aligned} a_i - c_i = &\ \gamma\delta - (\gamma^2 + \gamma)\delta \cdot \tfrac{x_i - x_{i+5}}{x_i + x_{i+5}} \\ \geq &\ \gamma\delta - (\gamma^2 + \gamma)\delta \cdot \tfrac{2}{3} \cdot \tfrac{x_i}{x_i + x_{i+5}} \\ \geq &\ \gamma\delta - (\gamma^2 + \gamma)\delta \cdot \tfrac{2}{3} \cdot \tfrac{3}{4} \\ > &\ 0, \end{aligned}$$

and

$$\begin{aligned} a_i + c_i &= 3\gamma\delta - (\gamma^2 + \gamma)\delta \cdot \tfrac{x_i + x_{i+5}}{x_i + x_{i+5}} \\ &> 0, \end{aligned}$$

from which we conclude that $a_i - |c_i| > 0$ for all $i = 1, 2, 3, 4$. The inequalities $d_i - |b_i| > 0$ and $d_i - |c_i| > 0$ follow analogously. All in all, this proves the assertion. □

To show that Assumption (5.22) holds at the solution x^*, we have to analyse the strongly active constraints at x^*. In particular, it is not difficult to verify that whenever the constraints $x^5 \leq x^3$ and $x^5 + x^{10} \leq k_5$ are strongly active for all $\lambda \in \mathcal{M}(x^*)$, which is the case at the numerical solution presented in the next section, then any vector $d \in \mathcal{T}(x^*, \lambda)$ has at least one component $i \neq 5$, $i \neq 10$ with $d_i \neq 0$. According to the previous Lemma this implies immediately that $d^T M(x^*) d > 0$, meaning that Assumption (5.22) holds.

We solved this example with starting point $(100, 100, \ldots, 100)$, however, both the Barzilai-Borwein method as well as the Newton method that is based on the unconstrained optimization reformulation failed for this starting point. So for the latter two methods a starting point very close to a solution was chosen. Moreover, for the Newton method we changed the parameters α and β, since it failed otherwise. Interestingly, the relaxation method was able to find the solution even from remote starting points such as $(1000, 1000, 1000, 1000, 10, 10, 10, 10, 10, 10)$ and $(10, 10, \ldots, 10)$ in only few iterations, while the Newton method based on the fixed point formulation failed for these starting points.

6.3 Numerical Results

We present the numerical results arranged according to the examples. The column *InnerIt* refers to the number of iterations necessary for the computation of $y_\alpha(x^k)$, or $y_\alpha(x^k)$ and $y_\beta(x^k)$, with the solver `SNOPT` in each iteration.

Barzilai-Borwein method

k	x_1^k	x_2^k	x_3^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.000000	0.000000	0.000000	0.1361518082979245	0
1	0.007261	0.006910	0.002941	0.1360427094475938	14
2	17.628067	16.776627	7.138440	0.0036885797244235	14
3	19.273268	16.174578	4.492779	0.0007416785230008	14
4	21.026541	16.074125	2.836986	0.0000032596075654	12
5	21.138745	16.004907	2.722435	0.0000001156482999	8
6	21.151220	16.042181	2.731168	0.0000000570958838	8
7	21.144064	16.028078	2.725161	0.0000000002262616	8
8	21.144692	16.027979	2.725905	0.0000000000047775	4
9	21.144762	16.027865	2.725976	0.0000000000000127	4
10	21.144768	16.027865	2.725982	0.0000000000383554	2
11	21.144767	16.027865	2.725982	0.0000000000000000	2

Relaxation method

k	x_1^k	x_2^k	x_3^k	$V_\alpha(x^k)$	stepsize	InnerIt
0	0.000000	0.000000	0.000000	90.8783016935181820	0.0000	0
1	19.325861	17.174694	3.811536	0.1184021869776447	1.0000	7
2	20.704322	16.105367	3.049513	0.0036631052660658	1.0000	6
3	21.036702	16.036753	2.808431	0.0002134195425450	1.0000	4
4	21.118192	16.029539	2.746413	0.0000129244012658	1.0000	3
5	21.138243	16.028243	2.731008	0.0000007843314006	1.0000	3
6	21.143180	16.027947	2.727207	0.0000000476926749	1.0000	3
7	21.144396	16.027876	2.726271	0.0000000029210997	1.0000	3
8	21.144696	16.027859	2.726040	0.0000000000000000	1.0000	3

Newton's method based on optimization reformulation

k	x_1^k	x_2^k	x_3^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.000000	0.000000	0.000000	10.2971171700988862	0
1	21.144786	16.027881	2.725962	0.0000000000000000	14

Newton's method based on fixed point formulation

k	x_1^k	x_2^k	x_3^k	$\|y_\alpha(x^k) - x^k\|$	InnerIt
0	0.000000	0.000000	0.000000	26.1340202062020914	0
1	21.144800	16.027868	2.725956	0.0000000000000000	7

Table 6.5: Numerical results for the river basin pollution game

Barzilai-Borwein method

k	x_1^k	x_2^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.100000	0.100000	0.0000052653091851	0
1	0.099892	0.099892	0.0000051496217887	8
2	0.090643	0.090643	0.0000000207265598	8
3	0.090046	0.090046	0.0000000001068226	6
4	0.090000	0.090000	0.0000000000000000	6

Relaxation method

k	x_1^k	x_2^k	$V_{\alpha}(x^k)$	stepsize	InnerIt
0	0.100000	0.100000	0.0263327223481960	0.0000	0
1	0.087171	0.087171	0.0022418335124145	0.2500	4
2	0.090378	0.090378	0.0000397082293171	0.2500	1
3	0.089905	0.089905	0.0000025173339792	0.2500	3
4	0.090024	0.090024	0.0000001572108120	0.2500	2
5	0.089994	0.089994	0.0000000093048805	0.2500	2
6	0.090001	0.090001	0.0000000005214422	0.2500	3
7	0.090000	0.090000	0.0000000000000000	0.2500	1

Newton's method based on unconstrained optimization reformulation

k	x_1^k	x_2^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.100000	0.100000	0.0005120314348019	0
1	0.090631	0.090631	0.0000019453839386	8
2	0.090003	0.090003	0.0000000000474542	6
3	0.090000	0.090000	0.0000000000000000	2

Newton's method based on fixed point formulation

k	x_1^k	x_2^k	x_3^k	$\|y_\alpha(x^k) - x^k\|$	InnerIt
0	0.100000	0.100000	0.100000	0.1622704430677725	0
1	0.090238	0.090238	0.090238	0.0037588135778690	4
2	0.090000	0.090000	0.090000	0.0000000000000000	3

Table 6.6: Numerical results for the Internet switching example using starting point $x^0 = (0.1, \ldots, 0.1)$

Barzilai-Borwein method, terminated by force

k	x_1^k	x_2^k	x_3^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.100000	0.110000	0.120000	0.0000381000000002	0
1	0.099964	0.109960	0.119956	0.0000380695260959	2
2	0.010000	0.010000	0.010000	0.0000161999993167	2
3	0.064943	0.064943	0.064943	0.0000024579357576	2
4	0.100000	0.100000	0.100000	0.0000052653091811	2
5	0.068976	0.068976	0.068976	0.0000019249391410	8
6	0.072177	0.072177	0.072177	0.0000015482299654	2
7	0.100000	0.100000	0.100000	0.0000052653091654	2
8	0.074779	0.074779	0.074779	0.0000012721816533	8
9	0.076936	0.076936	0.076936	0.0000010638817308	2
10	0.100000	0.100000	0.100000	0.0000052653091788	2
11	0.078753	0.078753	0.078753	0.0000009028469890	8
12	0.080305	0.080305	0.080305	0.0000007757884890	2
13	0.100000	0.100000	0.100000	0.0000052653090356	2
14	0.081646	0.081646	0.081646	0.0000006737793841	8
15	0.082815	0.082815	0.082815	0.0000005906430434	4

Table 6.7: Numerical results for the Internet switching example using starting point $x^0 = (0.1, 0.12, \ldots, 0.19)$

Relaxation method, terminated by force

k	x_1^k	x_2^k	x_3^k	$V_\alpha(x^k)$	stepsize	InnerIt
0	0.100000	0.110000	0.120000	0.4260724139309636	0.0000	0
1	0.055000	0.060000	0.065000	0.0298453579367650	0.5000	1
2	0.090304	0.092485	0.094657	0.0264434723093843	1.0000	5
3	0.078749	0.080644	0.082531	0.0023467024352681	0.2500	5
4	0.083142	0.084771	0.086392	0.0001111978318160	0.2500	5
5	0.083670	0.085074	0.086472	0.0000561849531093	0.2500	4
6	0.087607	0.088233	0.088856	0.0000505374595352	1.0000	4
7	0.087515	0.088055	0.088592	0.0000103974935023	0.2500	4
8	0.088418	0.088807	0.089196	0.0000097726000347	0.5000	3
9	0.088479	0.088813	0.089148	0.0000033929490075	0.2500	3
10	0.088981	0.089223	0.089465	0.0000023853246782	0.5000	3
11	0.089063	0.089271	0.089480	0.0000012232352152	0.2500	3
12	0.089644	0.089736	0.089829	0.0000010429882894	1.0000	3
13	0.089633	0.089712	0.089793	0.0000002229457964	0.2500	3
14	0.089765	0.089822	0.089880	0.0000002032869300	0.5000	3
15	0.089775	0.089824	0.089874	0.0000000736802234	0.2500	3

Newton's method based on unconstrained optimization reformulation

k	x_1^k	x_2^k	x_3^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.100000	0.110000	0.120000	0.0038100000000220	0
1	0.010000	0.010000	0.010000	0.0016200000002407	2
2	0.100000	0.100000	0.100000	0.0005120314355159	2
3	0.090631	0.090631	0.090631	0.0000019453837216	8
4	0.090003	0.090003	0.090003	0.0000000000474544	6
5	0.090000	0.090000	0.090000	0.0000000000000000	2

Newton's method based on fixed point formulation

k	x_1^k	x_2^k	x_3^k	$\|y_\alpha(x^k) - x^k\|$	InnerIt
0	0.100000	0.110000	0.120000	0.4364630568571630	0
1	0.010000	0.010000	0.010000	0.2846049894364130	1
2	0.100000	0.100000	0.100000	0.1622713515888371	1
3	0.090238	0.090238	0.090238	0.0037589338084981	4
4	0.090000	0.090000	0.090000	0.0000000000000000	3

Table 6.8: Numerical results for the Internet switching example using starting point $x^0 = (0.1, 0.12, \ldots, 0.19)$

Barzilai-Borwein method

k	x_1^k	x_2^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	1.000000	1.000000	0.0001999999999742	0
1	1.000000	0.999600	0.0001998400319845	2
2	1.000000	-0.000000	0.0000000000000000	2

Relaxation method

k	x_1^k	x_2^k	$V_{\alpha}(x^k)$	stepsize	InnerIt
0	1.000000	1.000000	1.9999499999354007	0.0000	0
1	1.000000	0.000000	0.0000000000000000	1.0000	1

Newton's method based on optimization reformulation

k	x_1^k	x_2^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.000000	0.000000	0.0110681478315567	0
1	1.000000	-0.000000	0.0000000000000000	10

Newton's method based on fixed point formulation

k	x_1^k	x_2^k	$\|y_\alpha(x^k) - x^k\|$	InnerIt
0	1.000000	1.000000	0.9999999999353941	0
1	1.000000	0.000000	0.0000000000000000	1

Table 6.9: Numerical results for Rosen's example

Barzilai-Borwein method

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$V_{\alpha\beta}(x^k)$	InnerIt
$P = 75$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	0.0268480539534721	0
1	10.001070	10.001579	10.002074	10.002531	10.002904	0.0268272037401402	14
2	12.817462	14.156111	15.461249	16.663323	17.645778	0.0009845862149476	14
3	10.689551	13.066910	15.324900	17.297873	18.745320	0.0000092732708712	10
4	10.430158	13.023471	15.393945	17.378413	18.772063	0.0000000864525910	8
5	10.406391	13.035304	15.407584	17.381497	18.770792	0.0000000007169719	6
6	10.403678	13.035667	15.407236	17.381502	18.771533	0.0000000000200473	4
7	10.403886	13.035907	15.407414	17.381566	18.771292	0.0000000000001722	4
8	10.403870	13.035891	15.407399	17.381551	18.771278	0.0000000000000139	2
9	10.403874	13.035891	15.407399	17.381551	18.771285	0.0000000000000000	2
$P = 100$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	0.1020219378488036	0
1	10.002988	10.003580	10.004129	10.004584	10.004872	0.1019411654908708	14
2	17.637915	19.151786	20.554940	21.720377	22.455028	0.0021474746283765	14
3	14.600540	17.886589	20.784894	22.959137	24.032952	0.0000382036002087	12
4	14.075578	17.773689	20.889808	23.111326	24.137870	0.0000001514347596	10
5	14.053873	17.798219	20.907911	23.111158	24.131859	0.0000000019138247	8
6	14.049869	17.798091	20.906969	23.111440	24.133067	0.0000000000324460	6
7	14.050123	17.798397	20.907212	23.111459	24.132877	0.0000000000001882	4
8	14.050107	17.798382	20.907196	23.111444	24.132863	0.0000000000000107	2
9	14.050115	17.798382	20.907196	23.111444	24.132863	0.0000000000000000	2
10	14.050115	17.798382	20.907196	23.111444	24.132863	0.0000000000000000	2
$P = 150$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	0.4015502352185649	0
1	10.007001	10.007725	10.008291	10.008588	10.008490	0.4012296537875955	18
2	27.581167	29.400812	30.821726	31.568544	31.321645	0.0026245998283327	18
3	24.142654	28.700749	31.861325	33.139026	32.342113	0.0000426540202867	12
4	23.600032	28.667245	32.016883	33.289873	32.418046	0.0000000516033283	12
5	23.589835	28.684688	32.021791	33.286980	32.418230	0.0000000002427575	10
6	23.588618	28.684232	32.021454	33.287291	32.418180	0.0000000000031846	6
7	23.588697	28.684322	32.021511	33.287265	32.418218	0.0000000000000178	4
8	23.588690	28.684322	32.021504	33.287265	32.418218	0.0000000000000000	2
$P = 200$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	0.9008965096018073	0
1	10.011344	10.012106	10.012512	10.012407	10.011675	0.9001761099880241	20
2	38.371089	40.278778	41.294023	41.030151	39.198877	0.0013693429954593	20
3	36.065684	40.686198	42.687317	41.907698	38.713861	0.0000130260916454	16
4	35.788937	40.745185	42.803534	41.966728	38.693647	0.0000000053306220	12
5	35.785448	40.749010	42.802408	41.966346	38.697054	0.0000000000110361	8
6	35.785340	40.748952	42.802469	41.966375	38.696840	0.0000000000000718	4
7	35.785347	40.748952	42.802488	41.966375	38.696840	0.0000000000000013	2
8	35.785344	40.748952	42.802488	41.966375	38.696840	0.0000000000000000	2
9	35.785345	40.748952	42.802488	41.966375	38.696840	0.0000000000000000	2

Table 6.10: Barzilai-Borwein method, Cournot oligopoly

Relaxation method

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$V_\alpha(x^k)$	stepsize	InnerIt
$P = 75$								
0	10.000000	10.000000	10.000000	10.000000	10.000000	1028.8786429070255508	0	0
1	13.012779	14.054536	15.077163	16.029954	16.825568	3.5863763599505036	1	7
2	11.285846	13.311206	15.235897	16.937369	18.229682	0.3540710625785035	1	5
3	10.704955	13.106954	15.332147	17.236881	18.619062	0.0366161338762231	1	4
4	10.507367	13.052936	15.377109	17.334696	18.727893	0.0039489433586007	1	4
5	10.439626	13.039433	15.395804	17.366456	18.758681	0.0004407116225397	1	3
6	10.416270	13.036369	15.403095	17.376688	18.767579	0.0000505346883797	1	3
7	10.408178	13.035815	15.405831	17.379981	18.770196	0.0000059183085803	1	3
8	10.405378	13.035815	15.406784	17.381016	18.771007	0.0000007309734651	1	2
9	10.404398	13.035815	15.407169	17.381377	18.771241	0.0000000918735889	1	2
10	10.404046	13.035851	15.407310	17.381493	18.771301	0.0000000000000000	1	2
$P = 100$								
0	10.000000	10.000000	10.000000	10.000000	10.000000	1836.0501506003590748	0	0
1	17.833057	19.050570	20.189450	21.150398	21.776524	4.8985691329028089	1	7
2	15.207009	18.069357	20.605740	22.548004	23.569889	0.3897086827633425	1	6
3	14.408248	17.849890	20.795596	22.950904	23.995363	0.0331524073207579	1	5
4	14.161944	17.805297	20.868563	23.065798	24.098398	0.0029760163785664	1	4
5	14.085258	17.797975	20.894328	23.098440	24.123998	0.0002781506763527	1	4
6	14.061206	17.797529	20.903009	23.107717	24.130539	0.0000267855976024	1	4
7	14.053618	17.797915	20.905866	23.110376	24.132225	0.0000026365305624	1	3
8	14.051213	17.798180	20.906774	23.111128	24.132705	0.0000002639463250	1	3
9	14.050453	17.798289	20.907047	23.111345	24.132866	0.0000000281294894	1	2
10	14.050207	17.798347	20.907143	23.111409	24.132894	0.0000000000000000	1	2
$P = 150$								
0	10.000000	10.000000	10.000000	10.000000	10.000000	2960.3391382693284868	0	0
1	27.861564	29.366477	30.558893	31.203707	31.009359	3.3981613265572332	1	9
2	24.632641	28.734751	31.626029	32.850552	32.156026	0.1804220099046543	1	6
3	23.846734	28.671683	31.919700	33.194867	32.367016	0.0102235564269385	1	6
4	23.653031	28.675783	31.996005	33.267441	32.407740	0.0006072128116255	1	6
5	23.604847	28.681014	32.015203	33.282960	32.415976	0.0000373064440210	1	5
6	23.592767	28.683251	32.019964	33.286283	32.417735	0.0000023469891688	1	4
7	23.589726	28.684002	32.021128	33.287038	32.418106	0.0000001503470039	1	4
8	23.588960	28.684201	32.021408	33.287226	32.418205	0.0000000105297725	1	2
9	23.588765	28.684288	32.021485	33.287257	32.418205	0.0000000000000000	1	2
$P = 200$								
0	10.000000	10.000000	10.000000	10.000000	10.000000	3592.9209675026095283	0	0
1	38.595613	40.204251	41.080180	40.865307	39.254648	1.2868808680857118	1	10
2	36.344525	40.610616	42.510038	41.804641	38.730180	0.0417678220639321	1	8
3	35.896154	40.715637	42.751504	41.940932	38.695772	0.0015123953742686	1	7
4	35.807288	40.741280	42.793422	41.962130	38.695880	0.0000574531909636	1	6
5	35.789690	40.747242	42.800846	41.965643	38.696579	0.0000022363864192	1	5
6	35.786204	40.748594	42.802181	41.966242	38.696779	0.0000000888394029	1	3
7	35.785510	40.748872	42.802431	41.966359	38.696829	0.0000000036951777	1	2
8	35.785374	40.748937	42.802468	41.966379	38.696842	0.0000000001307215	1	2
9	35.785360	40.748951	42.802468	41.966379	38.696842	0.0000000000000000	1	1

Table 6.11: Relaxation method, Cournot oligopoly

Newton's method based on optimization reformulation

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$\|y_\alpha(x^k) - x^k\|$	InnerIt
$P = 75$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	2.6786204605270996	0
1	11.014554	13.105738	15.130358	16.959172	18.384208	0.0064462863037402	14
2	10.405898	13.034733	15.406324	17.381103	18.770855	0.0000000558757351	8
3	10.403849	13.035879	15.407384	17.381549	18.771333	0.0000000000004384	4
4	10.403859	13.035877	15.407382	17.381548	18.771332	0.0000000000000535	2
5	10.403863	13.035877	15.407382	17.381547	18.771332	0.0000000000000000	2
$P = 100$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	10.1939346917138209	0
1	15.431454	17.979156	20.324112	22.223645	23.334286	0.0330866131224870	14
2	14.055011	17.795377	20.905247	23.111411	24.132776	0.0000003126881686	10
3	14.050101	17.798372	20.907203	23.111424	24.132903	0.0000000000000851	4
4	14.050097	17.798373	20.907204	23.111424	24.132903	0.0000000000000000	2
$P = 150$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	40.1459577329819695	0
1	26.213388	28.810884	30.821384	31.844606	31.465176	0.1144536600658421	18
2	23.594821	28.681442	32.022050	33.289099	32.419084	0.0000005886128564	12
3	23.588697	28.684325	32.021503	33.287262	32.418211	0.0000000000000000	8
$P = 200$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	90.0826592112757680	0
1	38.411841	40.116481	41.037314	40.813982	39.081464	0.1305591329655253	20
2	35.788495	40.748594	42.804137	41.967241	38.697104	0.0000002088390430	15
3	35.785339	40.748957	42.802481	41.966382	38.696840	0.0000000000000000	8

Table 6.12: Newton's method based on optimization reformulation, Cournot oligopoly

Newton's method based on fixed point formulation

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$\|y_\alpha(x^k) - x^k\|$	InnerIt
$P = 75$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	11.5863027186178531	0
1	10.727996	13.099087	15.304209	17.218265	18.650443	0.2667545777621945	7
2	10.403967	13.035818	15.407354	17.381555	18.771306	0.0000000000000000	4
$P = 100$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	22.5856681233344716	0
1	14.742243	17.889842	20.649363	22.776440	23.942112	0.5830566903965523	7
2	14.050339	17.798223	20.907147	23.111451	24.132840	0.0002091129151843	5
3	14.050091	17.798381	20.907187	23.111428	24.132914	0.0000000000000000	2
$P = 150$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	44.8079718213763485	0
1	24.666020	28.638950	31.530397	32.884666	32.279967	0.9504256360932131	9
2	23.588783	28.684250	32.021532	33.287256	32.418178	0.0000000000000000	7
$P = 200$							
0	10.000000	10.000000	10.000000	10.000000	10.000000	67.1154610852267837	0
1	36.770882	40.503658	42.325655	41.769703	38.630101	0.9181893886394852	10
2	35.785304	40.748979	42.802485	41.966390	38.696842	0.0000305348293455	7
3	35.785335	40.748961	42.802484	41.966378	38.696841	0.0000000000000000	2

Table 6.13: Newton's method based on fixed point formulation, Cournot oligopoly

Barzilai-Borwein method, termination forced

k	x_1^k	x_2^k	x_3^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.000000	0.000000	0.000000	3.0311662234453252	0
1	0.051406	0.035012	0.017841	3.0251475139957620	64
2	47.368574	32.250330	16.433010	0.0715357894412136	64
3	41.663602	26.906336	12.447389	0.0157110871790564	62
4	43.637895	28.816875	13.786874	0.0022529927912274	48
5	43.777938	28.987535	13.850859	0.0019860267947251	50
6	45.134373	30.619596	14.451977	0.0006981427206973	52
7	45.828495	31.384372	14.509053	0.0004955949266563	48
8	46.246198	31.699690	15.088737	0.0001743351052406	62
9	45.975878	31.537081	14.632948	0.0003304410778542	42
10	46.470766	31.998881	14.980723	0.0001725436483664	46
11	46.296809	31.831463	14.859763	0.0000254077725542	46
12	46.314363	31.846087	14.870673	0.0000230479308367	34
13	46.616490	32.102620	15.042368	0.0000009312260811	34
14	46.661529	32.243389	14.829838	0.0000090882841041	38
15	46.652374	32.142600	15.006356	0.0000003026499943	42

Relaxation method, termination forced

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$V_\alpha(x^k)$	InnerIt	Stepsize
0	0.000000	0.000000	0.000000	0.000000	0.000000	52371.6094234325355501	0	0.000
1	80.000000	68.380846	25.181592	50.158649	21.708525	42744.2808159713604255	32	1.000
2	0.526463	1.097102	6.268441	0.000000	3.512643	42072.0794469271204434	27	1.000
3	80.000000	65.449158	24.363140	47.788213	20.926774	37689.0644626562789199	33	1.000
4	4.331442	2.622106	6.698038	0.000000	4.444352	37147.0035835723028868	27	1.000
5	80.000000	63.251739	23.745700	46.062377	20.353441	34157.4454841218612273	32	1.000
6	7.146937	3.735045	7.011569	0.246908	5.020995	33693.7588451127958251	25	1.000
7	80.000000	61.617456	23.286507	44.792163	19.930745	31658.2507151712889026	32	1.000
8	9.230623	4.554719	7.242499	0.871427	5.230787	31241.6921930087737564	25	1.000
9	80.000000	60.404915	22.945817	43.855209	19.617534	29870.5953501933618099	31	1.000
10	10.773381	5.160365	7.413115	1.334715	5.386468	29483.8909876110883488	26	1.000
11	80.000000	59.506641	22.693269	43.162108	19.385895	28581.4010472223126271	23	1.000
12	11.915649	5.608469	7.539364	1.677942	5.501838	28214.0366338659441681	26	1.000
13	80.000000	58.841307	22.506357	42.648948	19.214398	27645.7632874234222982	25	1.000
14	12.761360	5.940151	7.632825	1.932217	5.587226	27291.2047398758331838	25	1.000
15	79.958887	58.348674	22.367940	42.269055	19.087431	26941.2895882350239845	25	1.000

Table 6.14: Numerical results for the electricity market example I

Relaxation method, $\sigma = 1$, termination forced

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$V_\alpha(x^k)$	InnerIt	Stepsize
0	0.000000	0.000000	0.000000	0.000000	0.000000	52371.6094234325355501	0	0.000
1	80.000000	68.380846	25.181592	50.158649	21.708525	42744.2808159713604255	32	1.000
2	40.263232	34.738974	15.725016	25.079324	12.610584	33.3390410686054182	27	0.500
3	43.284224	33.273137	15.315784	23.038547	12.644888	8.5324916592153031	26	1.000
4	45.189424	32.866192	15.202456	22.690663	12.533050	2.4301259557224184	21	1.000
5	45.729968	32.348829	15.057528	22.275656	12.395186	0.9249245813960247	24	1.000
6	46.399782	32.406838	15.074095	22.311452	12.407640	0.5504151289254504	24	1.000
7	46.332824	32.123566	14.994591	22.088695	12.333389	0.4542232682575659	22	1.000
8	46.696199	32.293364	15.042407	22.217135	12.376326	0.4263998492369000	18	1.000
9	46.589291	32.181398	15.010933	22.130224	12.347342	0.0040787136268533	18	0.500
10	46.625111	32.167024	15.006902	22.118069	12.343418	0.0010068732068689	15	1.000
11	46.644134	32.160958	15.005130	22.112926	12.341637	0.0002509667549085	21	1.000
12	46.652319	32.156890	15.004052	22.109556	12.340536	0.0000650451317720	20	1.000
13	46.657626	32.155917	15.003742	22.108712	12.340203	0.0000190872658195	18	1.000
14	46.659011	32.154411	15.003360	22.107535	12.339785	0.0000079338418535	18	1.000
15	46.660938	32.154646	15.003435	22.107580	12.339886	0.0000049292382130	12	1.000

Newton's method based on optimization reformulation

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$V_{\alpha\beta}(x^k)$	InnerIt
0	0.000000	0.000000	0.000000	0.000000	0.000000	285.8006379540747730	0
1	80.000067	23.536227	12.590189	15.420989	10.120063	9.8613630619447576	61
2	46.661637	32.153992	15.003106	22.107226	12.339598	0.0000000000159522	24
3	46.661622	32.154032	15.003126	22.107200	12.339582	0.0000000000000000	8

Newton's method based on fixed point formulation

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	$\|y_\alpha(x^k) - x^k\|$	InnerIt
0	0.000000	0.000000	0.000000	0.000000	0.000000	123.1610182063412395	0
1	80.000000	23.536242	12.590164	15.421419	10.118248	22.3691037849047198	32
2	46.661622	32.154050	15.003109	22.107198	12.339584	0.0000000000000000	14

Table 6.15: Numerical results for the electricity market example I

Barzilai-Borwein method

k	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$V_{\alpha\beta}(x^k)$	InnerIt
0	117.00	356.00	176.00	328.00	123.00	101.00	300.00	176.00	328.00	176.00	0.0005441749813340	0
1	117.00	356.00	176.00	328.00	123.00	101.00	300.00	176.00	328.00	176.00	0.0005437172349780	28
2	117.66	356.62	176.19	328.49	123.74	101.70	300.35	176.22	328.49	176.21	0.0000008561955186	28
3	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000069543561	18
4	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000000565859	10
5	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000000000020	8
6	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000000000000	2

Newton's method based on unconstrained optimization reformulation, $\alpha = 10^{-4}, \beta = 5 \cdot 10^{-4}$

k	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$V_{\alpha\beta}(x^k)$	InnerIt
0	117.00	356.00	176.00	328.00	123.00	101.00	300.00	176.00	328.00	176.00	0.0005441749923731	0
1	117.70	356.66	176.18	328.52	123.77	101.74	300.36	176.23	328.51	176.23	0.0000030893444696	31
2	117.64	356.62	176.20	328.47	123.79	101.68	300.32	176.21	328.47	176.21	0.0000000293252044	18
3	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000004279991	14
4	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000000662262	12
5	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	-0.0000000000053261	6
6	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000000000000	4

Relaxation method

k	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$V_{\alpha}(x^k)$	stepsize	InnerIt
0	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	15298.0520053010204720	0	0
1	117.59	296.42	169.77	278.82	130.25	102.89	263.22	169.75	278.82	169.75	434.2703592396658792	1	29
2	117.67	353.94	176.43	328.72	123.58	101.70	304.72	176.42	328.72	176.42	0.9584084392662747	1	12
3	117.64	356.77	176.20	328.46	123.80	101.69	300.62	176.20	328.46	176.20	0.0047460645900628	1	10
4	117.64	356.64	176.21	328.47	123.79	101.69	300.30	176.21	328.47	176.21	0.0000138890875876	1	9
5	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000912400164	1	6
6	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000000000000	1	3

Newton's method based on fixed point formulation

k	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$\|y_{\alpha}(x^k) - x^k\|$	InnerIt
0	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	380.8002984381889746	0
1	117.72	1076.69	197.54	1012.10	90.04	104.01	1144.65	209.96	1176.45	209.96	2031.1264201718749973	29
2	117.65	486.47	177.58	438.35	124.87	101.68	323.17	175.13	366.18	175.13	184.0779777475790979	19
3	117.64	357.02	176.22	329.46	123.92	101.69	300.27	176.08	327.29	176.08	1.5412694384380414	14
4	117.64	356.63	176.21	328.48	123.79	101.69	300.29	176.21	328.38	176.21	0.0966162993432361	8
5	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0006802778139822	3
6	117.64	356.63	176.21	328.47	123.79	101.69	300.32	176.21	328.47	176.21	0.0000000000000000	2

Table 6.16: Numerical results for the electricity market example II

Bibliography

[1] E. ALLGOWER AND K. GEORG: *Simplicial and continuation methods for approximating fixed points and solutions to systems of equations.* SIAM Review 22, 1980, pp. 28–85.

[2] A. ANTIPIN: *Extra-proximal methods for solving two-person nonzero-sum games.* Mathematical Programming, Series B, published online 2007

[3] A. ANTIPIN: *Equilibrium programming: Gradient methods* Automation and Remote Control 58, 1997, p. 1337–1347.

[4] A. ANTIPIN AND S.J. FLÅM: *Equilibrium programming using proximal-like algorithms.* Mathematical Programming 78, 1997, 29–41.

[5] K.J. ARROW AND G. DEBREU: *Existence of an equilibrium for a competitive economy.* Econometrica 22, 1954, pp. 265–290.

[6] J.-P. AUBIN: *Mathematical Methods of Game and Economic Theory.* North-Holland, Amsterdam, 1979.

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

[8] J. BARZILAI AND J.M. BORWEIN: *Two point step size gradient method.* IMA Journal on Numerical Analysis 8, 1988, pp. 141–148.

[9] T. BAŞAR: *Relaxation techniques and asynchronous algorithms for on-line computation of non-cooperative equilibria.* Journal of Economic Dynamics and Control 11, 1987, pp. 531–549.

[10] T. BAŞAR AND G.J. OLSDER: *Dynamic Noncooperative Game Theory.* Academic Press, New York, second edition 1995 (reprinted by SIAM, Philadelphia, 1999).

[11] A. Bensoussan: *Points de Nash dans le cas de fonctionelles quadratiques et jeux differentiels lineaires a N personnes.* SIAM Journal on Control 12, 1974, pp. 460–499.

[12] 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

[13] E. Blum and W. Oettli: *From optimization and variational inequalities to equilibrium problems.* The Mathematics Student 63, 1994, pp.123–145.

[14] F.H. Clarke: *Optimization and Nonsmooth Analysis.* John Wiley, New York, 1983 (reprinted by SIAM, Philadelphia, 1990).

[15] 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.

[16] R.W. Cottle, J.-S. Pang, and R.E. Stone: *The Linear Complementarity Problem.* Academic Press, Boston, 1992.

[17] S. Dafermos: *An iterative scheme for variational inequalities.* Mathematical Programming 26, 1983, pp. 40–47.

[18] Y.H. Dai and L.Z. Liao: *R-linear convergence of the Barzilai and Borwein gradient method.* IMA Journal on Numerical Analysis 22, 2002, pp. 1–10.

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

[20] G. Debreu: *A social equilibrium existence theorem.* Proceedings of the National Academy of Sciences 38 (10), 1952, pp. 886–893.

[21] S. Dempe and D. Pallaschke: *Quasidifferentiability of optimal solutions in parametric nonlinear optimization.* Optimization 40, 1997, pp. 1-24.

[22] S. Dempe and S. Vogel: *The generalized Jacobian of the optimal solution in parametric optimization.* Optimization 50, 2001, pp. 387–405.

[23] A. Dreves and C. Kanzow: *Nonsmooth optimization reformulations characterizing all solutions of jointly convex generalized Nash equilibrium problems.* Preprint 287, Institute of Mathematics, University of Würzburg, Würzburg, April 2009.

[24] L. Drouet et. al. *An oracle based method to compute a coupled equilibrium in a model of international climate policy.* Computational Management Science 5, 2008, p. 119–140.

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

[26] F. Facchinei and C. Kanzow: *Penalty methods for the solution of generalized Nash equilibrium problems.* Preprint 285, Institute of Mathematics, University of Würzburg, Würzburg, January 2009.

[27] F. Facchinei and J.-S. Pang: *Finite-Dimensional Variational Inequalities and Complementarity Problems, Volume I.* Springer, New York, NY, 2003.

[28] F. Facchinei and J.-S. Pang: *Finite-Dimensional Variational Inequalities and Complementarity Problems, Volume II.* Springer, New York, NY, 2003.

[29] F. Facchinei and J.-S. Pang: *Exact penalty functions for generalized Nash problems.* appeared in: Large-Scale Nonlinear Optimization, Springer, 2006.

[30] F. Facchinei, A. Fischer, and C. Kanzow: *Regularity properties of a semismooth reformulation of variational inequalities.* SIAM Journal on Optimization 8, 1998, pp. 850–869.

[31] F. Facchinei, A. Fischer, and V. Piccialli: *On generalized Nash games and VIs.* Technical Report, Department of Computer and System Sciences "A. Ruberti", Università di Roma "La Sapienza", Rome, Italy, January 2006.

[32] F. Facchinei, A. Fischer, and V. Piccialli: *Generalized Nash equilibrium problems and Newton methods.* Mathematical Programming 117, 2009, pp. 163–194.

[33] A. Fischer: *Solution of monotone complementarity problems with locally Lipschitzian mappings.* Mathematical Programming 76, 1997, pp. 513–532.

[34] S.D. Flåm and A.S. Antipin: *Equilibrium programming using proximal-like algorithms.* Mathematical Programming 78, 1997, pp. 29–41.

[35] R. Fletcher: *On the Barzilai-Borwein method.* Numerical Analysis Report NA/207, Department of Mathematics, University of Dundee, Dundee, United Kingdom, 2001.

[36] S.D. Flåm *Paths to constrained Nash equilibria.* Applied Mathematics and Optimization 27, 1993, pp. 275–289.

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

[38] M. Fukushima: *Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems.* Mathematical Programming 53, 1992, pp. 99–110.

[39] M. Fukushima: *Restricted generalized Nash equilibria and controlled penalty algorithm.* Technical Report 2008-007, Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, July, 2008.

[40] M.S. Gowda: *Inverse and implicit function theorems for H-differentiable and semismooth functions.* Optimization Methods and Software 19, 2004, pp. 443–461.

[41] L. Grippo and M. Sciandrone: *Nonmonotone globalization techniques for the Barzilai-Borwein gradient method.* Computational Optimization and Applications 23, 2002, pp. 143–169.

[42] G. Gürkan and J.-S. Pang: *Approximations of Nash equilibria.* Mathematical Programming 117, 2009, pp. 223–253.

[43] P.T. Harker: *Generalized Nash games and quasivariational inequalities.* European Journal of Operations Research 54, 1991, pp. 81–94.

[44] P.T. Harker and J.-S. Pang: *Finite-Dimensional Variational Inequality and Nonlinear Complementarity Problems: A Survey of Theory, Algorithms and Applications.* Mathematical Programming 48, 1990, pp. 161–220.

[45] A. Haurie and J.B. Krawczyk: *Optimal charges on river effluent from lumped and distributed sources.* Environmental Modeling and Assessment 2, 1997, pp. 93–106.

[46] A. von Heusinger and C. Kanzow: *Optimization reformulations of the generalized Nash equilibrium problem using Nikaido-Isoda-type functions.* Computational Optimization and Applications, available online under DOI 10.1007/s10589-007-9145-6.

[47] A. von Heusinger and C. Kanzow: *SC^1 optimization reformulations of the generalized Nash equilibrium problem.* Optimization Methods and Software, Volume 23 Issue 6, 2008.

[48] A. von Heusinger and C. Kanzow: *Relaxation methods for generalized Nash equilibrium problems with inexact line search.* Journal of Optimization Theory and Applications, available online under DOI 10.1007/s10957-009-9553-0.

[49] A. von Heusinger, C. Kanzow, and M. Fukushima: *Newton's method for computing a normalized equilibrium in the generalized Nash game through fixed point formulation.* Preprint 286, Institute of Mathematics, University of Würzburg, Würzburg, January 2009.

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

[51] J.-B. Hiriart-Urruty, J.-J. Strodiot, and V.H. Nguyen: *Generalized Hessian matrix and second-order optimality conditions for problems with $C^{1,1}$ data.* Applied Mathematics and Optimization 11, 1984, pp. 43–56.

[52] B.F. Hobbs: *Linear complementarity models of Nash-Cournot competition in bilateral and POOLCO power markets.* IEEE Transactions on Power Systems 16, 2001, pp. 194–202.

[53] W.W. Hogan: *Point-to-set maps in mathematical programming.* SIAM Review 15, 1973, pp. 591–603.

[54] R. Janin: *Directional derivative of the marginal function in nonlinear programming.* Mathematical Programming Study 21, 1984, pp. 110–126.

[55] C. Kanzow and M. Fukushima: *Theoretical and numerical investigation of the D-gap function for box constrained variational inequalities.* Mathematical Programming 83, 1998, pp. 55–87.

[56] 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.

[57] M. Kojima and S. Shindo: *Extension of Newton and Quasi-Newton methods to systems of PC^1 equations.* Journal of the Operations Research Society of Japan 29, 1986, pp. 352–372.

[58] J.B. Krawczyk: *An Open-Loop Nash Equilibrium in an Environmental Game with Coupled Constraints.* Proceedings of the Ninth International Symposium on Dynamic Games and Applications, December 18-21, 2000, Adelaide, pp. 325-339.

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

[60] J.B. Krawczyk: *Numerical solutions to coupled-constraint (or generalised Nash) equilibrium problems.* Computational Management Science 4, 2007, p. 183–204.

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

[62] B. Kummer: *Newton's method for non-differentiable functions.* In J. Guddat et al. (eds.): *Advances in Mathematical Optimization.* Akademie-Verlag, Berlin, 1988, pp. 114–125.

[63] B. Kummer: *Newton's method based on generalized derivatives for nonsmooth functions: Convergence analysis.* In W. Oettli and D. Pallaschke (eds.): *Advances in Optimization.* Lecture Notes in Economics and Mathematical Systems 382, Springer-Verlag, Berlin, 1992, pp. 171–194.

[64] S. Li and T. Başar: *Distributed algorithms for the computation of noncooperative equilibria.* Automatica 23, 1987, pp. 523–533.

[65] J. Liu: *Strong stability in variational inequalities.* SIAM Journal on Control and Optimization 33, 1995, pp. 725–749.

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

[67] L.W. McKenzie: *On the existence of general equilibrium for a competetive market.* Econometrica 27, 1959, pp. 54–71.

[68] R. Mifflin: *Semismooth and semiconvex functions in constrained optimization.* SIAM Journal on Control and Optimization 15, 1977, pp. 957–972.

[69] F. Murphy, H.D. Sherali and A.B. Soyster: *A mathematical programming approach for determining oligopolistic market equilibrium* Mathematical Programming 24, 1982, pp. 92–100.

[70] N. Nisan et.al. (eds.) *Algorithmic Game Theory.* Cambridge University Press, Cambridge, 2007.

[71] K. Nabetani, P. Tseng and M. Fukushima: *Parametrized variational inequality approaches to generalized Nash equilibrium problems with shared constraints.* Technical Report 2008-011, Department of Applied Mathematics and Physics, Kyoto University, October 2008.

[72] J. Nash: *Non-cooperative games.* Annals of Mathematics 54, 1951, pp. 286–295.

[73] H. Nikaido and K. Isoda: *Note on noncooperative convex games.* Pacific Journal of Mathematics 5, 1955, pp. 807–815.

[74] J.M. Ortega and W.C. Rheinboldt: *Iterative Slution of Nonlinear Equations in Several Variables.* Academic Press, New York, 1970.

[75] J. Outrata, M. Kocvara, and J. Zowe: *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints.* Kluwer Academic Press, Dordrecht, 1998.

[76] J. Outrata and J. Zowe: *A numerical approach to optimization problems with variational inequality constraints.* Mathematical Programming 68, 1995, pp. 105–130.

[77] J. Outrata and J. Zowe: *A Newton method for a class of quasi-variational inequalities.* Computational Optimization and Applications 4, 1995, pp. 5–21.

[78] J.-S. Pang and M. Fukushima: *Quasi-variational inequalities, generalized Nash equilibria, and multi-leader-follower games.* Computational Management Science 2, 2005, pp. 21–56.

[79] J.-S. Pang and D. Ralph: *Piecewise smoothness, local invertibility, and parametric analysis of normal maps.* Mathematics of Operations Research 21, 1996, pp. 401–426.

[80] J.-S. Pang and J. Sun: *Nash equilibria with piecewise quadratic costs.* Pacific Journal of Optimization 2, 2006, pp. 679–692.

[81] J.-S. Pang and L. Qi: *Nonsmooth equations: motivation and algorithms.* SIAM Journal on Optimization 3, 1993, pp. 443–465.

[82] J.-M. Peng: *Equivalence of variational inequality problems to unconstrained optimization.* Mathematical Programming 78, 1997, pp. 347–356.

[83] J.-S. Pang, D. Sun, and J. Sun: *Semismooth homeomorphisms and strong stability of semidefinite and Lorentz complementarity problems.* Mathematics of Operations Research 28, 2003, pp. 39–63.

[84] L. Qi: *Convergence analysis of some algorithms for solving nonsmooth equations.* Mathematics of Operations Research 18, 1993, pp. 227–244.

[85] L. Qi and J. Sun: *A nonsmooth version of Newton's method.* Mathematical Programming 58, 1993, pp. 353–367.

[86] H. Rademacher: *Über partielle und totale Differenzierbarkeit I.* Mathematical Annals 89, 1919, pp. 340–359.

[87] M. Raydan: *On the Barzilai and Borwein choice of the steplength for the gradient method.* IMA Journal on Numerical Analysis 13, 1993, pp. 321–326.

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

[89] S.M. Robinson: *Strongly regular generalized equations.* Mathematics of Operations Research 5, 1980, pp. 43–62.

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

[91] R.T. Rockafellar and R.J.-B. Wets: *Variational Analysis.* Springer, Berlin, 2004.

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

[93] T. Sandholm, A. Gilpin and V. Conitzer *Mixed-Integer programming methods for finding Nash equilibria.* Proceedings of the national conference on artificial intelligence 20, part 2, 2005, pages 495-501.

[94] A. Shapiro: *On concepts of directional differentiability.* Journal of Optimization Theory and Applications 66, 1990, pp. 477–487.

[95] D. Sun: *A further result on an implicit function theorem for locally Lipschitz functions.* Operations Research Letters 28, 2001, pp. 193–198.

[96] D. Sun, M. Fukushima, and L. Qi: *A computable generalized Hessian of the D-gap function and Newton-type methods for variational inequality problems.* Complementarity and Variational Problems: State of the Art, M.C. Ferris and J.-S. Pang (eds.), SIAM , Philadelphia , 1997, pp. 452-472.

[97] 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.

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

[99] N. Yamashita, K. Taji, and M. Fukushima: *Unconstrained optimization reformulations of variational inequality problems.* Journal of Optimization Theory and Applications 92, 1997, pp. 439–456.

[100] L. Zhang and J. Han: *Unconstrained optimization reformulation of equilibrium problems.* Technical Report, Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing, China, 2006.