Noname manuscript No.
(will be inserted by the editor)

A Nonmonotone Trust-Region Method for Generalized Nash Equilibrium Problems with Strong Convergence Properties

Leonardo Galli · Christian Kanzow · Marco Sciandrone

Received: date / Accepted: date

Abstract The generalized Nash equilibrium problem (GNEP) is often difficult to solve by Newton-type methods since the problem tends to have locally nonunique solutions. Here we take an existing trust-region method which is known to be locally fast convergent under a relatively mild error bound condition, and modify this method by a nonmonotone strategy in order to obtain a more reliable and efficient solver. The nonmonotone trust-region method inherits the nice local convergence properties of its monotone counterpart and is also shown to have the same global convergence properties. Numerical results indicate that the nonmonotone trust-region method is significantly better than the monotone version, and is at least competitive to an existing software applied to the same reformulation used within our trust-region framework.

Keywords Generalized Nash Equilibrium Problem · Trust-Region Algorithm · Nonmonotone strategy · Global convergence · Local superlinear convergence

L. Galli
Department of Information Engineering, University of Florence
Via di Santa Marta 3, 50139 Firenze, Italy
E-mail: leonardogalli2@msn.com

C. Kanzow
Institute of Mathematics, University of Würzburg
Am Hubland, 97074 Würzburg, Germany
E-mail: kanzow@mathematik.uni-wuerzburg.de

M. Sciandrone
Department of Information Engineering, University of Florence
Via di Santa Marta 3, 50139 Firenze, Italy
E-mail: marco.sciadrone@unifi.it

1 Introduction

The aim of this paper is to construct an efficient method for the solution of generalized Nash equilibrium problems (GNEPs for short). These problems have a wide range of applications in economics, operations research, computer science, telecommunications etc. The interested reader is referred to the survey papers [9,12] for more details regarding applications, theoretical results and numerical approaches for the solution of GNEPs.

The main difficulty with GNEPs is that these problems tend to have solution sets which are even locally nonunique. This means that standard Newton-type schemes typically do not work very well. This is also illustrated, for example, in the more general context of quasi-variational inequality problems by the semismooth Newton method from [10] which works extremely well for some examples, but fails for quite a few other test problems. On the other hand, there exist some very reliable methods with nice global convergence properties like the interior-point-type scheme from [8] or the augmented Lagrangian-type method from [16], but they are not locally fast convergent, and might even have problems in getting high accuracy of the solutions.

In order to obtain a globally and locally superlinearly convergent method for GNEPs, we are therefore urged to apply suitable methods which also work for nonunique solutions. Fortunately, in the meantime, there exist a few methods for optimization problems and nonlinear systems of equations which have this desired property under an error bound condition, see, e.g., [5,11,13,17,21] for some attempts in this direction. Moreover, there also exist some recent papers that provide error bounds for GNEPs [7,15]. However, these error bounds depend on the particular reformulation of the GNEP. The most prominent reformulations take the KKT conditions of the players, concatenate all KKT conditions into a larger system and apply, e.g., the Fischer-Burmeister function in order to get a semismooth system of equations. Unfortunately, it turns out to be difficult to find Newton-type methods for nonsmooth systems of equations which converge locally superlinearly under an error bound condition without any further assumptions, see, for example, the discussion in [14].

We therefore use a smooth reformulation of the GNEP with some simple bound constraints for which a suitable error bound is available from [7]. Moreover, we take the trust-region method from [20] which works precisely in our situation where we have a smooth constrained system of equations and which is locally superlinearly convergent under an error bound condition. In order to improve the practical convergence of the trust-region method from [20], we introduce a nonmonotone variant of that method in such a way that it is still globally convergent and inherits the local properties of the original method.

The paper is therefore organized in the following way: Section 2 presents our algorithmic scheme. There we first recall the (monotone) trust-region method from [20] and then derive the necessary modifications for a nonmonotone version with the same local convergence properties. The global convergence of the nonmonotone trust-region method is shown in Section 3. The details for the application of this method applied to GNEPs are presented in Section 4. The corresponding numerical results are given in Section 5. We then conclude with some final remarks in Section 6.

Notation: $\mathbb{R}^n$ denotes the n-dimensional Euclidean vector space, $\mathbb{R}^n_+$ is its subset in which vectors have only nonnegative components, the symbol $\|\cdot\|$ is the Euclidean vector norm, G' is the Jacobian of a generic differentiable mapping G, ∇G is its transposed and $\nabla_{z^\nu} G$ is the same, but with respect to the variables z^ν, where z^ν typically indicates a suitable subvector of z. The symbol $B_r(z)$ denotes the (Euclidean) ball of radius $r > 0$ around a given point z, while $P_\Omega(z)$ stands for the (Euclidean) projection of z onto a nonempty, closed and convex set Ω.

2 Trust-Region Methods

Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a given function and $\Omega \subseteq \mathbb{R}^n$ be a nonempty set. We consider the problem of finding a solution of the constrained nonlinear system of equations

$$F(x) = 0, \quad x \in \Omega \tag{1}$$

which is of much interest for its own since problems of this kind arise in many situations. The GNEP discussed in some more detail in Section 4 is only one of the applications. The solution set of (1) will be denoted by X^*.

2.1 The Monotone Trust-Region Method

In this section we report the trust-region method by Tong and Qi [20], formulated in a way such that its generalization to a nonmonotone framework will be easy to state with just some minor modifications.

We first recall or state the assumptions that are assumed to hold for problem (1) in order to get global and local fast convergence of the trust-region method.

Assumption 1 (a) F is continuously differentiable with F' being locally Lipschitzian.
(b) Ω is nonempty, closed, and convex.
(c) The solution set X^* is nonempty.
(d) $\|F(x)\|$ provides a local error bound in a neighbourhood of a solution $x^* \in X^*$, i.e. there exists constants $\delta > 0$ and $\gamma > 0$ such that

$$\text{dist}(x, X^*) \leq \gamma \|F(x)\| \quad \forall x \in \Omega \cap B_\delta(x^*).$$

Note that the central condition in Assumption 1 is part (d) where the usual nonsingularity condition of $F'(x^*)$ is replaced by a (weaker) error bound. This condition is only required in the local analysis in order to prove local fast convergence properties.

In order to describe the trust-region method, let us introduce the merit function

$$\Psi(x) := \frac{1}{2}\|F(x)\|^2$$

associated to (1). Since X^* is nonempty, x^* solves (1) if and only if x^* is a solution of the optimization problem

$$\min_x \ \Psi(x) \quad \text{s.t.} \quad x \in \Omega. \tag{2}$$

We call x^* a *stationary point* of (1) if x^* is a stationary point of the corresponding optimization problem (2), i.e. if

$$\nabla\Psi(x^*)^T(x - x^*) \geq 0 \quad \forall x \in \Omega;$$

recall that this is equivalent to x^* satisfying the fixed-point equation

$$x^* = P_\Omega\big(x^* - \gamma\nabla\Psi(x^*)\big)$$

for an arbitrary constant $\gamma > 0$.

To deal with locally nonunique solutions, the trust-region method considers a subproblem where the objective function involves an additional regularization term. More precisely, the regularized trust-region subproblem at a current iterate x_k is given by

$$\begin{aligned} &\min \Phi_k(d) := \frac{1}{2}\|F(x_k) + F'(x_k)d\|^2 + \frac{1}{2}\mu_k\|d\|^2 \\ &\text{s.t. } \|d\| \leq \Delta, \end{aligned} \tag{3}$$

where $\Delta > 0$ denotes the trust-region radius and $\mu_k > 0$ is a suitable constant depending on the iteration index k. This notation allows us to state the following trust-region method which corresponds to the method from [20], except that we leave the choice of an appropriate scalar $\hat{r}_k$ unspecified for the moment.

Algorithm 1 (Trust-Region Framework)

(S.0) Choose a starting point $x_0 \in \Omega$ and suitable constants $\alpha_1, \alpha_2, \rho_1, \rho_2, \eta, \sigma$, $\Delta_{min}, \Delta_{max}, \Delta_0$, C such that $0 < \alpha_1 < 1 < \alpha_2$, $0 < \rho_1 < \rho_2 < 1$, $\eta \in (0,1)$, $\sigma \in (0,1)$, $C > 0$, $\Delta_0 > 0$, $\Delta_{max} > \Delta_{min} > 0$. Set $k := 0$.

(S.1) If x_k is a stationary point of the optimization problem (2): STOP. Otherwise set $\Delta_k := \min\big\{\Delta_{max}, \max\{\Delta_{min}, \Delta_k\}\big\}$, $\Delta := \Delta_k$, $\mu_k := C\|F(x_k)\|$.

(S.2) Projected Gradient Direction: Compute

$$d_k^G(\Delta) := -\Big(\frac{\Delta}{\Delta_{max}}\Big)\gamma_k\nabla\Psi(x_k), \tag{4}$$

$$\bar{d}_k^G(\Delta) := P_\Omega\big[x_k + d_k^G(\Delta)\big] - x_k \tag{5}$$

with

$$\gamma_k := \min\bigg\{1, \frac{\Delta_{max}}{\|\nabla\Psi(x_k)\|}, \frac{\eta\Psi(x_k)}{\|\nabla\Psi(x_k)\|^2}\bigg\}.$$

(S.3) Projected Trust-Region Direction: Solve the trust-region subproblem (3) by a suitable algorithm and denote its solution by $d_k^{tr}(\Delta)$. Then compute

$$\bar{d}_k^{tr}(\Delta) := P_\Omega[x_k + d_k^{tr}(\Delta)] - x_k.$$

(S.4) Optimal Combined Direction: Compute

$$\bar{d}_k(\Delta) := t^*(\Delta)\bar{d}_k^G(\Delta) + \big(1 - t^*(\Delta)\big)\bar{d}_k^{tr}(\Delta),$$

where $t^*(\Delta) \in [0,1]$ is an optimal solution of the problem

$$\min_{t\in[0,1]} \; q_\Delta^k(t) := \frac{1}{2}\big\|F(x_k) + F'(x_k)\left(t\bar{d}_k^G(\Delta) + (1-t)\bar{d}_k^{tr}(\Delta)\right)\big\|^2,$$

see below for more details.

(S.5) Updates: Define the actual and predicted reductions by

$$\begin{aligned} Ared_k(\Delta) &:= \Psi\big(x_k + \bar{d}_k(\Delta)\big) - \Psi(x_k), \\ Pred_k(\Delta) &:= \frac{1}{2}\big\|F(x_k) + F'(x_k)\bar{d}_k(\Delta)\big\|^2 - \Psi(x_k), \end{aligned}$$

respectively, and choose a suitable scalar $\hat{r}_k$. If the following two conditions

$$-Pred_k(\Delta) \geq -\sigma \nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) \tag{6}$$

and

$$\hat{r}_k \geq \rho_1 \tag{7}$$

hold, set

$$x_{k+1} := x_k + \bar{d}_k(\Delta), \quad \Delta_{k+1} := \begin{cases} \Delta & \text{if } \rho_1 \leq \hat{r}_k < \rho_2, \\ \alpha_2\Delta & \text{if } \hat{r}_k \geq \rho_2, \end{cases}$$

define $\Delta_k^* := \Delta$ as the trust-region radius that allowed $x_k + \bar{d}_k(\Delta)$ to be accepted, set $k \leftarrow k+1$, and go to (S.1). Otherwise set $\Delta := \alpha_1\Delta$, and go to (S.2).

Let us give a few comments to explain the previous algorithm: in our convergence analysis we always assume implicitly that the termination criterion in (S.1) does not hold after finitely many iterations. Hence we assume in our theoretical analysis of Algorithm 1 that none of the iterates x_k is an exact stationary point of problem (2), so that $P_\Omega\big[x_k - \nabla\Psi(x_k)\big] - x_k \neq 0$ or, equivalently, that $\bar{d}_k^G(\Delta) \neq 0$ holds for all k. The remainder of (S.1) initializes some parameters for the inner iteration starting in (S.2). In particular, we reset the trust-region radius Δ_k by taking its projection onto the interval $[\Delta_{\min}, \Delta_{\max}]$.

Step (S.2) then computes a projected gradient direction which, more or less, is responsible for the global convergence of Algorithm 1. The projected trust-region step from (S.3), on the other hand, is the main ingredient to verify local fast convergence under an error bound assumption. In (S.4), a convex combination of these two directions is taken which yields the smallest objective function of the standard quadratic approximation of the mapping $\varPsi$. Note that the optimal $t^*(\varDelta)$ in this step can be computed analytically. In fact, it is not difficult to see that it has the closed form expression

$$t^*(\varDelta) = \max\bigl\{0, \min\{1, t(\varDelta)\}\bigr\},$$

where $t(\varDelta)$ is the solution of $\nabla q_\Delta^k(t) = 0$ and is given by

$$t(\varDelta) = \begin{cases} \frac{-\left(F(x_k)+F'(x_k)\bar{d}_k^{tr}(\varDelta)\right)^T F'(x_k)\left(\bar{d}_k^G(\varDelta)-\bar{d}_k^{tr}(\varDelta)\right)}{\|F'(x_k)\left(\bar{d}_k^G(\varDelta)-\bar{d}_k^{tr}(\varDelta)\right)\|^2} \\ \qquad\qquad\qquad \text{if } F'(x_k)\bar{d}_k^G(\varDelta) \neq F'(x_k)\bar{d}_k^{tr}(\varDelta), \\ \text{any number in } (-\infty, +\infty) \\ \qquad\qquad\qquad \text{if } F'(x_k)\bar{d}_k^G(\varDelta) = F'(x_k)\bar{d}_k^{tr}(\varDelta). \end{cases}$$

In (S.5), we first compute the actual and predicted reductions at the current point. The subsequent update rule in (S.5) depends on the suitable choice of $\hat{r}_k$. The standard choice corresponding to the monotone trust-region method from [20] is

$$\hat{r}_k := \frac{Ared_k(\varDelta)}{Pred_k(\varDelta)} \tag{8}$$

and completely specifies Algorithm 1. The updates in (S.5) are then similar to a standard trust-region update except that the new point is accepted only if the two conditions (6) and (7) hold. This second condition, which is standard for monotone trust-region method, will be relaxed in our nonmonotone version. The reason for having the additional criterion from (6) comes from the fact that we need to have the predicted reduction to be a negative number whenever we leave the inner iteration. This property is not clear a priori, but will be a consequence of Lemma 1 below.

Note that, by construction, all iterates x_k generated by Algorithm 1 belong to the feasible set $\varOmega$. Furthermore, let us recall from [20] that Algorithm 1 with the standard choice of $\hat{r}_k$ from (8) is

- well-defined, in particular, for each outer iteration k, the number of inner iterations between (S.2) and (S.5) is finite,
- globally convergent in the sense that every accumulation point is a stationary point of (2), and
- locally fast convergent under the relatively weak error bound condition from Assumption 1.

For the details, we refer the reader to [18,20].

We close this section by stating two results that were given for a somewhat different active-set-type trust-region method in [18], but whose statements also hold for the above (monotone) trust-region method. These results will be used in our subsequent convergence analysis. For the sake of completeness, we provide the full proofs in an appendix. The first of these results corresponds to [18, Lemma 4.3].

Lemma 1 *Consider the trust-region method from Algorithm 1 with the update* $\hat{r}_k$ *from* (8)*. Then, for all* $k \in \mathbb{N}$ *and all* $\Delta \in (0, \Delta_{max}]$*, it holds that*

$$\nabla \Psi(x_k)^T \bar{d}_k^G(\Delta) \leq -\left(\frac{\Delta}{\Delta_{max}\gamma_k}\right) \left\| \bar{d}_k^G(\Delta_{max}) \right\|^2.$$

Note that this result implies that the predicted reduction is a negative number whenever we leave the inner iteration in Algorithm 1, i.e. we always have

$$Pred_k(\Delta_k^*) < 0, \tag{9}$$

cf. (6).

We next state a technical result which is the counterpart of [18, Proposition 4.1] and whose proof is also given in the appendix.

Proposition 1 *Consider the monotone trust-region method from Algorithm 1 with the update* $\hat{r}_k$ *from* (8)*. Suppose that* x^* *is an accumulation point of a subsequence* $\{x_k\}_{k \in K}$*. If* x^* *is not a stationary point, then there exist an index* $\hat{k} > 0$ *and a constant* $\bar{\Delta} > 0$ *such that, for all* $k \in K$ *with* $k \geq \hat{k}$*,* (6) *and* (7) *hold for all* $\Delta \in (0, \bar{\Delta})$*.*

2.2 The Nonmonotone Trust-Region Method

Here we present a nonmonotone version of Algorithm 1. The main idea is to accept also suitable points which do not necessarily reduce the objective function value. This is achieved by accepting the new step $\bar{d}_k(\Delta)$ more frequently. To this end, we relax condition (7). The strategy is inspired by the work [19] by Toint. We will see in Section 5 that the nonmonotonicity improves the numerical behaviour of the trust-region method.

To give a precise statement of the nonmonotone trust-region method, let us introduce one further parameter $W \in \mathbb{N}$. We then define

$$\Psi_{max}^k := \max_{k-W \leq i \leq k} \Psi(x_i) \tag{10}$$

as the largest function value among the last few iterations, where, formally, we set $x_{-1} := x_{-2} := \ldots := x_{-W} := x_0$ (or, alternatively, we can replace W by $W_k := \min\{k, W\}$). Since we are interested in the behaviour of an infinite sequence, we may assume without loss of generality that we always have $k \geq W$. Furthermore, let us define

$$r(k) \text{ as the (say, largest) iteration index such that } \Psi(x_{r(k)}) = \Psi_{max}^k \tag{11}$$

holds. We then define a modified (nonmonotone) actual reduction at iteration k by

$$Ared_k'(\Delta) := \Psi\big(x_k + \bar{d}_k(\Delta)\big) - \Psi_{max}^k. \tag{12}$$

Using an elementary calculation, we obtain the representation

$$\begin{aligned} Ared_k'(\varDelta) &= \varPsi\big(x_k + \bar{d}_k(\varDelta)\big) - \varPsi_{max}^k \\ &= \varPsi\big(x_k + \bar{d}_k(\varDelta)\big) - \varPsi(x_{r(k)}) \\ &= \varPsi\big(x_k + \bar{d}_k(\varDelta)\big) - \varPsi(x_k) + \sum_{i=r(k)}^{k-1} \big[\varPsi(x_{i+1}) - \varPsi(x_i)\big] \\ &= \varPsi\big(x_k + \bar{d}_k(\varDelta)\big) - \varPsi(x_k) + \sum_{i=r(k)}^{k-1} \big[\varPsi\big(x_i + \bar{d}_i(\varDelta_i^*)\big) - \varPsi(x_i)\big] \\ &= Ared_k(\varDelta) + \sum_{i=r(k)}^{k-1} Ared_i(\varDelta_i^*), \end{aligned}$$

where $\varDelta_i^*$ is defined as in (S.5) of Algorithm 1. This representation motivates to define a corresponding modified (nonmonotone) predicted reduction by

$$Pred_k'(\varDelta) := Pred_k(\varDelta) + \sum_{i=r(k)}^{k-1} Pred_i(\varDelta_i^*). \tag{13}$$

The idea is then to replace (7) by the condition

$$\max\left\{\frac{Ared_k(\varDelta)}{Pred_k(\varDelta)}, \frac{Ared_k'(\varDelta)}{Pred_k'(\varDelta)}\right\} \geq \rho_1. \tag{14}$$

Hence, the nonmonotone trust-region method is fully specified by Algorithm 1 with the choice

$$\hat{r}_k := \max\left\{\frac{Ared_k(\varDelta)}{Pred_k(\varDelta)}, \frac{Ared_k'(\varDelta)}{Pred_k'(\varDelta)}\right\}, \tag{15}$$

in (S.5). Since (14) is a relaxation of the corresponding monotone condition from (7), and all other parts in Algorithm 1 remain unchanged (in particular, we still use the condition from (6)), it follows that the nonmonotone trust-region method from Algorithm 1 with the update of $\hat{r}_k$ from (15) is also well-defined (in particular, the inner loop at each outer iteration k is always finite) and also inherits the local convergence properties from the monotone method and, therefore, the fast convergence rate under the error bound condition from Assumption 1. Hence, it remains to show that our nonmonotone modification does not destroy the global convergence. This is the aim of the next section.

3 Global Convergence

Here we want to show that every accumulation point of a sequence generated by the nonmonotone trust-region method from Algorithm 1 with the specification of $\hat{r}_k$ from (15) is still a stationary point of the corresponding optimization problem (2). A central step in this direction is contained in the following result. Recall that, also in this section, we assume implicitly that our method does not terminate after finitely many iterations with an exact stationary point.

Lemma 2 *Let $\{x_k\}$ be a sequence generated by Algorithm 1 with $\hat{r}_k$ defined by* (15)*. Then the inequality*

$$\Psi(x_0) - \Psi(x_{k+1}) \geq -\rho_1 \sum_{l=0}^{k} Pred_l(\Delta_l^*) \tag{16}$$

holds for every $k \in \mathbb{N}$.

Proof Let $k \geq 0$ be fixed. Then we define an index

$$p(k+1) := \begin{cases} k, & \text{if } \dfrac{Ared_k(\Delta_k^*)}{Pred_k(\Delta_k^*)} \geq \dfrac{Ared_k^{\prime}(\Delta_k^*)}{Pred_k^{\prime}(\Delta_k^*)}, \\ r(k), \text{otherwise}, \end{cases}$$

where $r(k)$ denotes the index defined in (11). We call the corresponding iterate $x_{p(k+1)}$ the “predecessor” of x_{k+1}. We then construct a sequence of strictly increasing iteration counters

$$\{k_0, k_1, k_2, k_3, \ldots\}$$

in such a way that $k_i = p(k_{i+1})$ for all $i = 1, 2, \ldots$, i.e. x_{k_i} is the predecessor of $x_{k_{i+1}}$. Recursively, this means that $k_{i-1} = p(k_i) = p(p(k_{i+1}))$ and so on, until we get down to the index k_1 such that the starting point x_0 is the predecessor of x_{k_1}, i.e. $k_0 := 0 := p(k_1)$. By definition, for each $i \in \mathbb{N}$, we either have $k_i = p(k_{i+1}) = k_{i+1} - 1$ or $k_i = p(k_{i+1}) = r(k_{i+1} - 1)$.

Now, consider the iterate x_{k+1}, and let x_{k_l} be the corresponding predecessor for some $l \in \mathbb{N}$. Then, we can write

$$\begin{aligned} \psi(x_0) - \psi(x_{k+1}) = \psi(x_0) - \psi(x_{k_1}) \quad &(17) \\ + \sum_{i=1}^{l-1} \left[\psi(x_{k_i}) - \psi(x_{k_{i+1}})\right] \quad &(18) \\ + \psi(x_{k_l}) - \psi(x_{k+1}). \quad &(19) \end{aligned}$$

We next take a closer look at each of the three terms (17)–(19).

First consider the term (18) and choose an arbitrary (but fixed) index $i \in \{1, \ldots, l-1\}$. We distinguish two situations: if $k_i = p(k_{i+1}) = k_{i+1} - 1$, we have $Ared_{k_i}(\Delta_{k_i}^*)/Pred_{k_i}(\Delta_{k_i}^*) \geq \rho_1$. Since the denominator is negative in view of (9), we therefore get

$$\begin{aligned} \psi(x_{k_i}) - \psi(x_{k_{i+1}}) &= \psi(x_{k_i}) - \psi(x_{k_i+1}) \\ &= -Ared_{k_i}(\Delta_{k_i}^*) \\ &\geq -\rho_1 Pred_{k_i}(\Delta_{k_i}^*) \\ &= -\rho_1 \sum_{j=k_i}^{k_{i+1}-1} Pred_j(\Delta_j^*). \end{aligned}$$

On the other hand, if $k_i = p(k_{i+1}) = r(k_{i+1}-1)$, we have $Ared'_{k_{i+1}-1}(\Delta^*_{k_{i+1}-1}) \leq \rho_1 Pred'_{k_{i+1}-1}(\Delta^*_{k_{i+1}-1})$ because of (13) and (9). We then obtain

$$\begin{aligned}
\psi(x_{k_i}) - \psi(x_{k_{i+1}}) &= \psi(x_{r(k_{i+1}-1)}) - \psi(x_{k_{i+1}}) \\
&= \psi_{max}^{k_{i+1}-1} - \psi(x_{k_{i+1}}) \\
&= -Ared'_{k_{i+1}-1}(\Delta^*_{k_{i+1}-1}) \\
&\geq -\rho_1 Pred'_{k_{i+1}-1}(\Delta^*_{k_{i+1}-1}) \\
&= -\rho_1 \sum_{j=r(k_{i+1}-1)}^{k_{i+1}-1} Pred_j(\Delta^*_j) \\
&= -\rho_1 \sum_{j=p(k_{i+1})}^{k_{i+1}-1} Pred_j(\Delta^*_j) \\
&= -\rho_1 \sum_{j=k_i}^{k_{i+1}-1} Pred_j(\Delta^*_j).
\end{aligned}$$

In a similar way, we can deal with the two terms from (17) and (19), respectively, and obtain that

$$\psi(x_0) - \psi(x_{k_1}) \geq -\rho_1 \sum_{j=0}^{k_1-1} Pred_j(\Delta^*_j)$$

and

$$\psi(x_{k_l}) - \psi(x_{k+1}) \geq -\rho_1 \sum_{j=k_l}^{k} Pred_j(\Delta^*_j).$$

The assertion follows by inserting the last three estimates into (17)–(19). □

The previous result allows us to prove global convergence of the nonmonotone trust-region method.

Theorem 1 *Let $\{x_k\}$ be the sequence generated by Algorithm 1 with $\hat{r}_k$ defined by* (15)*. Then every accumulation point of $\{x_k\}$ is a stationary point of* (2)*.*

Proof Let x^* be an accumulation point of $\{x_k\}$ and let

$$\lim_{k \in K,\, k \to \infty} x_k = x^*$$

be a convergent subsequence. By contradiction, suppose that x^* is not a stationary point of problem (2). Then

$$\gamma^* := \lim_{k \in K,\, k \to \infty} \gamma_k = \min\left\{1, \frac{\Delta_{max}}{\|\nabla \Psi(x^*)\|}, \frac{\eta \Psi(x^*)}{\|\nabla \Psi(x^*)\|^2}\right\}$$

is a positive number. The definition of $\bar{d}_k^G(\varDelta)$ via (4) and (5) together with the continuity of the projection operator then implies

$$\left\|\bar{d}_k^G(\varDelta_{max})\right\| \to \left\|P_\varOmega[x^* - \gamma^* \nabla \varPsi(x^*)] - x^*\right\| > 0 \quad \text{as } k \in K, k \to \infty,$$

where the right-hand side is positive since $\gamma^* > 0$ and x^* is not a stationary point. Hence, there exist an index $\tilde{k} > 0$ and a constant $b > 0$ such that

$$\left\|\bar{d}_k^G(\varDelta_{max})\right\| \geq b \quad \forall k \in K, k \geq \tilde{k}. \tag{20}$$

Moreover, using Proposition 1, there exist $\hat{k}$ and $\bar{\varDelta}$ such that, for all $k \in K, k \geq \hat{k}$, and all $\varDelta \in (0, \bar{\varDelta})$, the two conditions (6) and (14) are satisfied. Let us define $\hat{k} := \max\{\hat{k}, \tilde{k}\}$.

Let $k \in K$ and $k \geq \hat{k}$. Using the previous observation together with the updating rule of the trust-region radius in Algorithm 1, it follows that

$$\varDelta_k^* > \alpha_1 \bar{\varDelta}, \tag{21}$$

so we have a uniform lower bound on the size of $\varDelta_k^*$ for all sufficiently large $k \in K$. Taking into account Lemma 2, inequalities (9) and (6) as well as Lemma 1, we obtain for all $k \in K$, $k \geq \hat{k}$ that

$$\begin{aligned}
\varPsi(x_0) - \varPsi(x_{k+1}) &\geq -\rho_1 \sum_{l=0}^{k} Pred_l(\varDelta_l^*) \\
&\geq -\rho_1 \sum_{l \in K,\, l \geq \hat{k}}^{k} Pred_l(\varDelta_l^*) \\
&\geq -\rho_1 \sigma \sum_{l \in K,\, l \geq \hat{k}}^{k} \nabla \varPsi(x_l)^T \bar{d}_l^G(\varDelta_l^*) \\
&\geq \rho_1 \sigma \sum_{l \in K,\, l \geq \hat{k}}^{k} \Big(\frac{\varDelta_l^*}{\varDelta_{max} \gamma_l}\Big) \|\bar{d}_l^G(\varDelta_{max})\|^2 \\
&\geq \rho_1 \sigma \sum_{l \in K,\, l \geq \hat{k}}^{k} \Big(\frac{\varDelta_l^*}{\varDelta_{max}}\Big) b^2.
\end{aligned} \tag{22}$$

where the last inequality follows from (20) and the fact that $\gamma_k \leq 1$. Taking $k \to \infty$ with $k \in K$, the right-hand side of (22) goes to infinity because of (21), while the left-hand side is bounded since $\varPsi(x) \geq 0$ for all $x \in \mathbb{R}^n$. This contradiction completes the proof. □

4 Application to Generalized Nash Equilibrium Problems

As noted in the introduction, a typical class of problems which often have nonunique solutions are the so-called Generalized Nash Equilibrium Problems (GNEPs). Let N be the number of players (or agents) ν of the GNEP ($\nu = 1, ..., N$). Each agent's problem then consists of an optimization problem of the form

$$\min_{z^\nu} \quad \theta_\nu(z^\nu, z^{-\nu}) \quad \text{s.t.} \quad g^\nu(z^\nu, z^{-\nu}) \leq 0, \tag{23}$$

where $z^\nu \in \mathbb{R}^{n_\nu}$ represents the variables which are controlled by the ν-th player, $z^{-\nu} \in \mathbb{R}^{n_{n-n_\nu}}$ express the remaining ones he cannot control, $\theta_\nu : \mathbb{R}^n \to \mathbb{R}$ is his objective (or utility) function, while $g^\nu : \mathbb{R}^n \to \mathbb{R}^{m_\nu}$ describes his constraints and defines the possible strategies of player ν. Note that both θ_ν and g^ν depend on all variables, but that (23) is an optimization problem in z^ν only (parameterized by $z^{-\nu}$).

The ν-th player controls n_ν variables and the total number of variables of the problem, grouped under the name z, is $n := n_1 + \cdots + n_N$. Without loss of generality, we assume that players' options are only constrained by inequalities. The decision of player ν is affected by m_ν inequalities, and the total number of inequalities of the problem is $m := m_1 + \cdots + m_N$.

A solution of a GNEP is called *generalized Nash equilibrium* and is reached when none of the N agents is able to obtain a better value for his utility function by unilaterally changing his strategy. This means that $z^* \in \mathbb{R}^n$ is a generalized Nash equilibrium if, for all $\nu = 1, ..., N$, it holds that

$$g^\nu(z^{*,\nu}, z^{*,-\nu}) \leq 0 \text{ and } \theta_\nu(z^{*,\nu}, z^{*,-\nu}) \leq \theta_\nu(z^\nu, z^{*,-\nu}) \qquad \forall z^\nu \in \mathbb{R}^{n_\nu} : \; g^\nu(z^\nu, z^{*,-\nu}) \leq 0.$$

We assume the following properties to hold for all θ_ν and g^ν.

Property 1 Functions θ_ν and g^ν are twice continuously differentiable with locally Lipschitz continuous second order derivatives for all $\nu = 1, ..., N$.

Property 2 The GNEP is player-convex, i.e. $\theta_\nu(\cdot, z^{-\nu})$ and $g_i^\nu(\cdot, z^{-\nu})$ are convex functions for every $\nu = 1, ..., N$, $i = 1, ..., m_\nu$ and $z^{-\nu}$.

Note that the player-convex case is typically the most general class of GNEPs considered in the literature and, in particular, allows much more freedom than the jointly-convex case, cf. [9] for more details.

The Lagrangian function related to the ν-th optimization problem (23) is

$$L^\nu(z, \lambda^\nu) := \theta_\nu(z^\nu, z^{-\nu}) + \sum_{i=1}^{m_\nu} \lambda_i^\nu g_i^\nu(z^\nu, z^{-\nu}),$$

where $\lambda^\nu \in \mathbb{R}^{m_\nu}$ is the vector of Lagrange multipliers of player ν. If we concatenate all the multipliers, all the constraints, and all the gradients of the Lagrangian, we obtain

$$\lambda := \left(\lambda^\nu\right)_{\nu=1}^N, \qquad H(z, \lambda) := \left(\nabla_{z^\nu} L^\nu(z, \lambda^\nu)\right)_{\nu=1}^N, \qquad g(z) := \left(g^\nu(z)\right)_{\nu=1}^N.$$

Let z be a solution of (23). If we assume any standard constraint qualification to hold for every player $\nu = 1, \dots, N$, it follows that there exists a vector $\lambda \in \mathbb{R}^m$ such that the following (concatenated) KKT-system holds:

$$H(z, \lambda) = 0, \qquad \lambda \geq 0, \qquad g(z) \leq 0, \qquad \lambda^T g(z) = 0. \tag{24}$$

Conversely, any solution of the system yields a solution of the GNEP (without any constraint qualification) due to the assumed player-convexity property.

We, therefore, focus on solving (24) in order to obtain a solution of the original problem (23). The mixed system (24), in turn, might be reformulated in different ways, but for reasons explained in the introduction, we concentrate on a simple, smooth reformulation as a constrained nonlinear system of equations: introducing slack variables $w \in \mathbb{R}^m_+$ and using the Hadamard product (the component-wise product $(w \circ \lambda)_i := w_i \lambda_i$ for all $i = 1, \dots, m$), we obtain the following box-constrained system of equations:

$$F(x) := \begin{pmatrix} H(z, \lambda) \\ g(z) + w \\ w \circ \lambda \end{pmatrix} = 0, \qquad s.t. \quad x := (z, \lambda, w) \in \Omega := \mathbb{R}^n \times \mathbb{R}^m_+ \times \mathbb{R}^m_+. \tag{25}$$

We assume that the corresponding solution set $X^* := \{x \in \Omega : F(x) = 0\}$ is nonempty.

Then we are precisely in the situation described in (1). Since Ω is defined by some nonnegativity constraints only, the projections onto Ω are easy to compute. Assumption 1 (a)-(c) also hold in our GNEP-setting. Furthermore, also the local error bound condition from (d) of this assumption was shown to be satisfied under suitable assumptions in [7]. In particular, these assumptions do not require that the Jacobian of F is nonsingular at a solution. In fact, even nonisolated solutions are allowed. We do not recall the precise conditions here and refer the interested reader to [7] for more details.

5 Numerical Results

In this section we report some numerical results of computational experiments performed in order to evaluate the effectiveness of the proposed nonmonotone algorithm. To this end, we first recall that our method is particularly designed to solve difficult problems with high accuracy under a relatively weak (error bound) condition. For those examples which satisfy stronger assumptions like a nonsingularity condition, one has to expect that other Newton-type schemes are more efficient. Nevertheless, the overall results indicate that our method is very competitive.

Our first aim is to compare the numerical behaviour of our method on a larger set of examples. To this end, we use the following three algorithms:

1. Algorithm 1 with r_k defined in (8): the original Monotone Trust-Region (MTR) framework by Tong and Qi [20];

2. Algorithm 1 with r_k defined in (15): the Nonmonotone Trust-Region (NTR) framework extending MTR;
3. STRSCNE [1,2], a specific solver for nonlinear, box-constrained systems of equations.

MTR and NTR have been implemented in MATLAB, the same programming language used also for the STRSCNE code. The algorithms terminate successfully when

$$\|F(x_k)\|_\infty \leq \epsilon_1, \qquad \text{with } \epsilon_1 = 10^{-4}. \tag{26}$$

For the algorithms, the maximum numbers of function evaluations and Jacobian evaluations have been fixed equal to 100,000. The termination criterion described in Algorithm 1 ($\|\bar{d}_k^G(\Delta)\| < \epsilon_2$) and used in our theoretical investigation has not been employed in our implementation since STRSCNE does not use any similar criterion. Regarding nonmonotone window, above results are obtained with $W = 50$, thus NTR might be considered strongly nonmonotone.

The three methods have been tested on a dataset of 35 different GNEPs [8]. The total number of runs is 57 because multiple starting points have been used for some of the problems (see Table 5.1 of [8] for more details).

We first compare the number of failures obtained by the three algorithms:

– 9 for STRSCNE;
– 12 for MTR;
– 7 for NTR.

From the above comparison, we might see that the adoption of the nonmonotone strategy yields significant advantages in terms of computed global solutions, with respect to the original monotone version of Algorithm 1 (MTR). In fact, MTR seems to converge more frequently to nonoptimal stationary points. Furthermore, Algorithm NTR also outperforms STRSCNE.

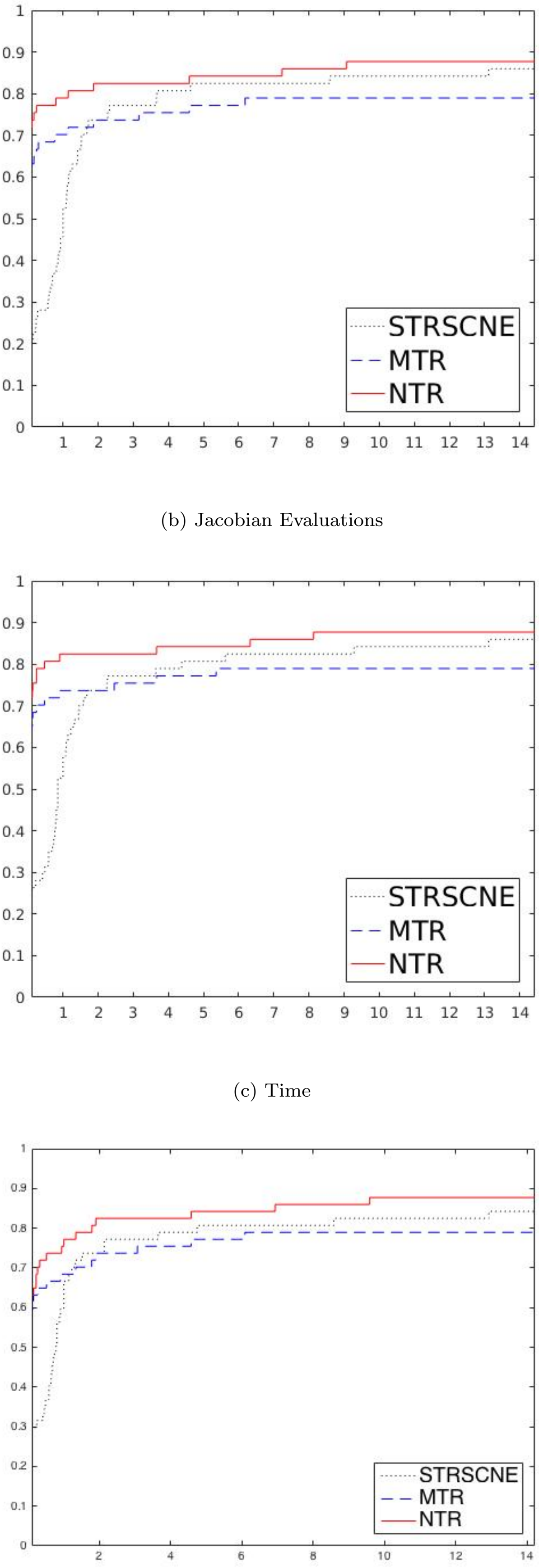
The algorithms have been numerically compared using performance profiles [6]. In Figure 1 we show results obtained employing the following measures:

- total number of function evaluations (Figure 1a);
- total number of Jacobian evaluations (Figure 1b);
- actual computational time, estimated by `tic toc` MATLAB function (Figure 1c).

Note that in order to obtain reliable measurements on computational time, experiments have been repeated several times, choosing the minimum obtained results as the final one. In addition, algorithms are considered to be equivalent on this metric, if the difference between their measures is smaller than $10^{-3}\,s$. Moreover, note that scale used for all the metrics in Figure 1 is logarithmic since results are sometimes pretty close.

Figure 1 clearly shows the good performances of the proposed algorithm with respect to all the employed metrics. Again, it may be observed that the employment of the nonmonotone strategy leads to a significant improvement of the performances of the original monotone version of the algorithm. The proposed algorithm may be considered at least competitive with a sound and efficient code as STRSCNE.

Fig. 1: Performance profiles between STRSCNE (black dotted line), MTR (blue dashed line) and NTR (red solid line).



(a) Function Evaluations

(b) Jacobian Evaluations

(c) Time

Table 1: Detailed numerical results for problem A11.

k	$\|F(x)\|_\infty$	$\|\bar{d}_k^G(\Delta_k^*)\|_2$	Δ_k
0	1.00e+02	1.1e-01	5.00
1	5.65e+01	2.0e-01	10.00
2	1.56e+01	2.6e-01	20.00
3	3.85e+00	1.5e-01	40.00
4	9.48e-01	1.5e-01	80.00
5	2.22e-01	4.2e-02	160.00
6	4.33e-02	1.1e-02	320.00
7	4.44e-03	2.0e-03	640.00
8	7.38e-05	5.2e-05	1000.00
Final	2.20e-08	3.1e-08	

An illustrative example

In the remainder of this section we are going to show performances obtained by NTR on a specific GNEP from [8]: example A11. As presented in *Example 1* from [7], this is a peculiar problem because the error bound condition for (25) is proven to be valid for the whole solution set, while the Jacobian of F is singular at all solutions. This means that local fast convergence for classical Newton-like methods cannot be expected here, while MTR and NTR are proven to be locally superlinearly convergent.

In Table 1 we present the iteration history obtained by NTR where the columns show, for each iteration, $\{\|F(x_k)\|_\infty\}$, $\{\|\bar{d}_k^G(\Delta_k^*)\|_2\}$ and the initial trust region radius Δ_k.

From Table 1 we may observe the two phases of the algorithm: the globalization strategy might be considered active between iterations 0 and 4, as the measure of stationarity $\|\bar{d}_k^G(\Delta_k^*)\|$ is not sufficiently small and not always decreasing. Starting with iteration 5, the norm of the projected gradient is decreasing rapidly, following the typical behaviour of superlinear convergence.

6 Conclusions

This paper modifies a (monotone) trust-region method for constrained systems of equations by introducing a suitable nonmonotonicity criterion. The corresponding nonmonotone trust-region method is shown to be globally convergent to stationary points and locally fast convergent under an error bound condition. Numerical results obtained for a suitable reformulation of generalized Nash equilibrium problems indicate that the nonmonotone method is both more reliable and more efficient than its monotone counterpart. A preliminary testing shows that the numerical behaviour can still be improved if one simply skips the additional criterion (6) from Algorithm 1. Part of our future research is, therefore, to investigate whether this condition can (at least) be relaxed in an appropriate way.

A Proofs of Lemma 1 and Proposition 1

We first recall some elementary properties of the projection operator.

Lemma 3 *The following statements hold:*
(a) $\big(P_\Omega(z) - z\big)^T \big(P_\Omega(z) - x\big) \leq 0 \quad \forall x \in \Omega,\ \forall z \in \mathbb{R}^n$*;*
(b) $\|P_\Omega(x_2) - P_\Omega(x_1)\| \leq \|x_2 - x_1\| \quad \forall x_1, x_2 \in \mathbb{R}^n$*.*
(c) Given $x, d \in \mathbb{R}^n$*, the function*

$$\theta(t) := \frac{\big\|P_\Omega(x + td) - x\big\|}{t}, \quad t > 0,$$

is nonincreasing.

The first two properties in Lemma 3 are a well-known characterization of the projection, whereas the third property was shown, e.g., in [3] in the context of a suitable globalization of a projected gradient method.

Proof of Lemma 1: Let $k \in \mathbb{N}$ be fixed, choose $\Delta > 0$, and define

$$z_k := x_k + d_k^G(\Delta) = x_k - \frac{\Delta}{\Delta_{\max}} \gamma_k \nabla\Psi(x_k)$$

for the sake of notational convenience. Then an elementary calculation yields

$$\begin{aligned}
\nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) &= \nabla\Psi(x_k)^T \big(P_\Omega(z_k) - x_k\big) \\
&= \frac{\Delta_{\max}}{\Delta \cdot \gamma_k} \big(x_k - z_k\big)^T \big(P_\Omega(z_k) - x_k\big) \\
&= \frac{\Delta_{\max}}{\Delta \cdot \gamma_k} \big(P_\Omega(z_k) - z_k\big)^T \big(P_\Omega(z_k) - x_k\big) \\
&\quad + \frac{\Delta_{\max}}{\Delta \cdot \gamma_k} \big(x_k - P_\Omega(z_k)\big)^T \big(P_\Omega(z_k) - x_k\big) \\
&\leq -\frac{\Delta_{\max}}{\Delta \cdot \gamma_k} \big\|\bar{d}_k^G(\Delta)\big\|^2,
\end{aligned}$$

where the inequality follows from Lemma 3 (a), the definition of $\bar{d}_k^G(\Delta)$, and the feasibility of x_k. On the other hand, Lemma 3 (c) with $d := -\nabla\Psi(x_k)$ implies that

$$\begin{aligned}
\frac{\big\|\bar{d}_k^G(\Delta)\big\|}{\Delta} &= \frac{\big\|P_\Omega\big(x_k - \frac{\Delta}{\Delta_{\max}} \gamma_k \nabla\Psi(x_k)\big) - x_k\big\|}{\Delta} \\
&\geq \frac{\big\|P_\Omega\big(x_k - \gamma_k \nabla\Psi(x_k)\big) - x_k\big\|}{\Delta_{\max}} \\
&= \frac{\big\|\bar{d}_k^G(\Delta_{\max})\big\|}{\Delta_{\max}}
\end{aligned}$$

holds for all $0 < \Delta \leq \Delta_{\max}$. Combining the last two inequalities yields the assertion. □

Proof of Proposition 1: Since $x_k \to x^*$ for $k \in K$ and $k \to \infty$, the continuity of F' implies that there is a constant b_1 such that $\|F'(x_k)\| \leq b_1$ for all $k \in K$. Using (4), (5), and Lemma 3 (b), we therefore obtain for all $k \in K$

$$\begin{aligned}
\big\|F'(x_k)\bar{d}_k^G(\Delta)\big\| &= \big\|F'(x_k)(P_\Omega[x_k + d_k^G(\Delta)] - x_k)\big\| \\
&\leq \big\|F'(x_k)\big\|\big\|x_k + d_k^G(\Delta) - x_k\big\| \\
&\leq \frac{\Delta\gamma_k}{\Delta_{max}} \big\|F'(x_k)\big\|\big\|\nabla\Psi(x_k)\big\| \\
&\leq b_1 \Delta,
\end{aligned} \tag{27}$$

where the last inequality follows from the definition of γ_k in (S.2).

Since x^* is not a stationary point by assumption, we can follow the argument from the first part of the proof of Theorem 1 in order to see that there is a constant $b > 0$ such that

$$\left\|\bar{d}_k^G(\Delta_{max})\right\| \geq b \quad \forall k \in K, k \geq \hat{k}. \tag{28}$$

Let

$$\tilde{\Delta} = \min\left\{\Delta_{max}, \frac{(1-\sigma)b^2}{b_1^2 \Delta_{max}}\right\}. \tag{29}$$

We first prove that (6) holds for all $k \in K, k \geq \hat{k}$ and all $\Delta \in (0, \tilde{\Delta}]$. From the definition of $\bar{d}_k(\Delta)$, we get that

$$\begin{aligned}
Pred_k(\Delta) &= \frac{1}{2}\left\|F(x_k) + F'(x_k)\bar{d}_k(\Delta)\right\|^2 - \Psi(x_k) \\
&\leq \frac{1}{2}\left\|F(x_k) + F'(x_k)\bar{d}_k^G(\Delta)\right\|^2 - \Psi(x_k) \\
&= \nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) + \frac{1}{2}\left\|F'(x_k)\bar{d}_k^G(\Delta)\right\|^2 \\
&= \sigma\nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) + (1-\sigma)\nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) + \frac{1}{2}\left\|F'(x)\bar{d}_k^G(\Delta)\right\|^2 \\
&\leq \sigma\nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) - (1-\sigma)\Big(\frac{\Delta}{\Delta_{max}\gamma_k}\Big)\left\|\bar{d}_k^G(\Delta_{max})\right\|^2 + \frac{1}{2}b_1^2\Delta^2 \\
&\leq \sigma\nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) - b_1^2\Delta\tilde{\Delta} + \frac{1}{2}b_1^2\Delta^2 \\
&\leq \sigma\nabla\Psi(x_k)^T \bar{d}_k^G(\Delta),
\end{aligned}$$

where the second inequality follows directly from Lemma 1 and (27), the third inequality follows from (28) and (29) and recalling that $0 < \gamma_k \leq 1$, while the last inequality holds since $\Delta \leq \tilde{\Delta}$.

In order to prove that (7) holds for $k \in K$ and k sufficiently large and for Δ belonging to an interval $(0, \bar{\Delta}]$, we will first show that

$$-Pred_k(\Delta) \geq \beta\Delta, \tag{30}$$

and

$$Ared_k(\Delta) - Pred_k(\Delta) \leq c_1\Delta^2 \tag{31}$$

hold for suitable constants $\beta > 0$ and $c_1 > 0$

First we show that (30) holds. To this aim, taking $\Delta \in (0, \tilde{\Delta}]$, using Lemma 1 and (27), we can write

$$\begin{aligned}
\frac{1}{2}\left\|F(x_k) + F'(x_k)\bar{d}_k^G(\Delta)\right\|^2 &= \frac{1}{2}\|F(x_k)\|^2 + \nabla\Psi(x_k)^T \bar{d}_k^G(\Delta) + \frac{1}{2}\left\|F'(x_k)\bar{d}_k^G(\Delta)\right\|^2 \\
&\leq \Psi(x_k) - \Big(\frac{\Delta}{\gamma_k\Delta_{max}}\Big)\left\|\bar{d}_k^G(\Delta_{max})\right\|^2 + \frac{1}{2}b_1^2\Delta^2 \\
&\leq \Psi(x_k) - \Big(\frac{\Delta}{\gamma_k\Delta_{max}}\Big)\left\|\bar{d}_k^G(\Delta_{max})\right\|^2 + \frac{1}{2}\Delta\frac{b^2(1-\sigma)}{\Delta_{max}} \\
&\leq \Psi(x_k) - \Big(\frac{\Delta}{\gamma_k\Delta_{max}}\Big)\left\|\bar{d}_k^G(\Delta_{max})\right\|^2 + \frac{1}{2}\Delta\frac{\left\|\bar{d}_k^G(\Delta_{max})\right\|^2}{\gamma_k\Delta_{max}} \\
&= \Psi(x_k) - \Big(\frac{\Delta}{2\gamma_k\Delta_{max}}\Big)\left\|\bar{d}_k^G(\Delta_{max})\right\|^2,
\end{aligned}$$

where the second inequality follows from (29), and the third holds recalling that $(1-\sigma) < 1, \gamma_k \leq 1$, and (28). Consequently, we have

$$Pred_k(\Delta) \leq \frac{1}{2}\left\|F(x_k) + F'(x_k)\bar{d}_k^G(\Delta)\right\|^2 - \Psi(x_k) \leq -\Big(\frac{\Delta}{2\gamma_k\Delta_{max}}\Big)\left\|\bar{d}_k^G(\Delta_{max})\right\|^2 < 0,$$

where the first inequality follows from the definitions of $\bar{d}_k$ and $t^*(\Delta)$ in (S.4). Thus, using (28) and recalling again that $\gamma_k \leq 1$, we obtain that there exists $\beta > 0$ such that (30) is satisfied for all $k \in K, k \geq \hat{k}$ and all $\Delta \in (0, \tilde{\Delta}]$.

Now we prove (31). From Lemma 3 (b), recalling the definitions of $d_k^G(\Delta)$, $\bar{d}_k^G(\Delta)$ and γ_k, we have

$$\|\bar{d}_k^G(\Delta)\| \leq \|d_k^G(\Delta)\| \leq \Delta, \quad \forall \Delta \in (0, \Delta_{max}].$$

From the definition of $\bar{d}_k^{tr}(\Delta)$, using Lemma 3 (b) again, and recalling that $d_k^{tr}(\Delta)$ is the trust-region solution, we have

$$\|\bar{d}_k^{tr}(\Delta)\| \leq \|d_k^{tr}(\Delta)\| \leq \Delta, \quad \forall \Delta \in (0, \Delta_{max}].$$

Consequently, from the last two inequalities we get $\|\bar{d}_k(\Delta)\| \leq \Delta$. Since F' is locally Lipschitzian, it is globally Lipschitz on compact sets. Consequently, $\nabla\Psi$ is also globally Lipschitz on compact sets. Since $x_k \to x^*$ for $k \in K$ and $\bar{d}_k(\Delta)$ is bounded for all $\Delta \in (0, \Delta_{\max}]$, we can apply the Mean Value Theorem and obtain the existence of suitable numbers $\theta_k \in (0,1)$ and a Lipschitz constant $L > 0$ such that

$$\begin{aligned}\Psi\big(x_k + \bar{d}_k(\Delta)\big) - \Psi(x_k) - \nabla\Psi(x_k)^T\bar{d}_k(\Delta) &= \nabla\Psi\big(x_k + \theta_k\bar{d}_k(\Delta)\big)^T\bar{d}_k(\Delta) - \nabla\Psi(x_k)^T\bar{d}_k(\Delta)\\ &\leq L\Delta\|\bar{d}_k(\Delta)\|\end{aligned}$$

for all $k \in K$ and all $\Delta \in (0, \Delta_{\max}]$, where the last inequality takes into account the Cauchy-Schwarz inequality. Hence, we can write

$$\begin{aligned}Ared_k(\Delta) - Pred_k(\Delta) &= \Psi\big(x_k + \bar{d}_k(\Delta)\big) - \frac{1}{2}\big\|F(x_k) + F'(x_k)\bar{d}_k(\Delta)\big\|^2\\ &= \Psi\big(x_k + \bar{d}_k(\Delta)\big) - \Psi(x_k) - \nabla\Psi(x_k)^T\bar{d}_k(\Delta)\\ &\quad - \frac{1}{2}\bar{d}_k(\Delta)^T F'(x_k)^T F'(x_k)\bar{d}_k(\Delta)\\ &\leq L\Delta\big\|\bar{d}_k(\Delta)\big\| - \frac{1}{2}\bar{d}_k(\Delta)^T F'(x_k)^T F'(x_k)\bar{d}_k(\Delta)\\ &\leq L\Delta^2 + c_2\big\|\bar{d}_k(\Delta)\big\|^2\\ &\leq c_1\Delta^2\end{aligned}$$

for suitable constants $c_1, c_2 > 0$.

Finally, exploiting (30) and (31), it follows that there exists $\bar{\Delta} > 0$ such that

$$\hat{r}_k = 1 - \frac{Ared_k(\Delta) - Pred_k(\Delta)}{-Pred_k(\Delta)} \geq \rho_1, \quad \forall k \in K, k \geq \hat{k} \text{ and } \forall \Delta \in (0, \bar{\Delta}].$$

This concludes the proof. □

References

1. Bellavia S., Macconi M., Morini B., An affine scaling trust-region approach to bound-constrained nonlinear systems. Applied Numerical Mathematics 44, 257–180 (2003)
2. Bellavia S., Macconi M., Morini B., STRSCNE: A Scaled Trust Region Solver for Constrained Nonlinear Equations. Computational Optimization and Applications 28, 31–50 (2004)
3. Bertsekas D. P.: Nonlinear Programming. Athena Scientific, Belmont, second edition (1999)
4. Calamai P. H., Moré J. J.: Projected Gradient Methods for Linear Constrained Problems. Mathematical Programming 39, 93–116 (1987)
5. Dan H., Yamashita N., Fukushima M.: Convergence properties of the inexact Levenberg-Marquardt method under local error bound conditions. Optimization Methods and Software 17, 605-626 (2002)

6. Dolan E. D., Moré J. J.: Benchmarking optimization software with performance profiles. Mathematical Programming 91, 201–213 (2002)
7. Dreves A., Facchinei F., Fischer A., Herrich M.: A new error bound result for Generalized Nash Equilibrium Problems and its algorithmic application. Computational Optimization and Applications 59, 63–84 (2014)
8. Dreves A., Facchinei F., Kanzow C., Sagratella S.: On the solution of the KKT conditions of Generalized Nash Equilibrium Problems. SIAM Journal on Optimization 21, 1082–1108 (2011)
9. Facchinei F., Kanzow C.: Generalized Nash equilibrium problems. Annals of Operations Research 175, 177-211 (2010)
10. Facchinei F., Kanzow C., Karl S., Sagratella S.: The semismooth Newton method for the solution of quasi-variational inequalities. Computational Optimization and Applications 62, 85-109 (2015)
11. Fan J., Yuan Y.: On the quadratic convergence of the Levenberg-Marquardt method without nonsingularity assumption. Computing 74, 23-39 (2005).
12. Fischer A., Herrich M., Schönefeld K.: Generalized Nash equilibrium problems – Recent advances and challenges. Pesquisa Operacional 34, 521-558 (2014)
13. Fischer A., Shukla P.K., Wang, M.: On the inexactness level of robust Levenberg-Marquardt methods. Optimization 59, 273-287 (2010).
14. Herrich M.: Local convergence of Newton-type methods for nonsmooth constrained equations and applications. PhD Thesis, Institute of Mathematics, Technical University of Dresden, Germany, 2014
15. Izmailov A.F., Solodov M.V.: On error bounds and Newton-type methods for generalized Nash equilibrium problems. Computational Optimization and Applications 59, 201-218 (2014)
16. Kanzow C., Steck D.: Augmented Lagrangian methods for the solution of generalized Nash equilibrium problems. SIAM Journal on Optimization 26, 2034-2058 (2016)
17. Kanzow C., Yamashita N., Fukushima M.: Levenberg–Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints. Journal of Computational and Applied Mathematics 172, 375–397 (2004)
18. Qi L., Tong X. J., Li D. H.: Active-set projected trust-region algorithm for box-constrained nonsmooth equations. Journal of Optimization Theory and Applications 120, 601–625 (2004)
19. Toint P. L.: Non-monotone trust-region algorithms for nonlinear optimization subject to convex constraints. Mathematical Programming 77, 69–94 (1997)
20. Tong X. J., Qi L.: On the convergence of a trust-region method for solving constrained nonlinear equations with degenerate solutions. Journal of Optimization Theory and Applications 123, 187–211 (2004)
21. Yamashita N., Fukushima M.: On the rate of convergence of the Levenberg-Marquardt method. Computing 15, 239-249 (2001)