A Descent Method for Nonsmooth Variational Inequalities via Regularization

BARBARA PANICUCCI
University of Pisa
Department of Applied Mathematics
via Buonarroti 1/c, 56127 Pisa
ITALY
panicucc@mail.dm.unipi.it

MASSIMO PAPPALARDO
University of Pisa
Department of Applied Mathematics
via Buonarroti 1/c, 56127 Pisa
ITALY
pappalardo@dma.unipi.it

MAURO PASSACANTANDO
University of Pisa
Department of Applied Mathematics
via Buonarroti 1/c, 56127 Pisa
ITALY
passacantando@dma.unipi.it

Abstract: in this paper we propose a descent method for solving variational inequality problems where the underlying operator is nonsmooth, locally Lipschitz, and monotone over a closed, convex feasible set. The idea is to combine a descent method for variational inequality problems whose operators are nonsmooth, locally Lipschitz, and strongly monotone, with the Tikonov-Browder regularization technique. Finally, numerical results are presented and discussed.

Key–Words: variational inequality, nonsmooth mapping, gap function, descent method, Tikhonov-Browder regularization.

1 Introduction

Let X a nonempty, closed and convex subset of $\mathbb{R}^n$ and $F : \mathbb{R}^n \to \mathbb{R}^n$ a given map. The variational inequality (VI) problem is to find a point $x^* \in X$ such that

$$\langle F(x^*), x - x^* \rangle \geq 0, \qquad \forall\, x \in X, \qquad \text{(P)}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product in $\mathbb{R}^n$. This problem is becoming an increasingly powerful methodological tool for the study of many problems arising in different fields such us economics, engineering, mechanics and physics. Indeed, this problem contains systems of nonlinear equations, optimization problems and complementarity problems as special cases, and it is also related to fixed point problems. Several methods including projection and its variant forms, auxiliary principle, decomposition and descent methods have been developed for solving VIs. For a complete discussion and history of the VI problem and associated solution methods we refer to [2] and references therein. One of the most popular approaches to solve a VI consists in reformulating it as an equivalent optimization problem. The underlying idea is to use an artificial gap function which is a real valued function whose global minima coincide with the solutions of VI. Gap functions give rise to iterative descent methods for their minimization and, as a consequence, for the solution of the VI problem. Most existing descent methods are shown to converge when the underlying operator F of the VI is continuously differentiable and satisfies suitable monotonicity assumptions. Recently, in [4, 5] two descent methods, with respect to gap functions, for solving a VI with nonsmooth, locally Lipschitz and strongly monotone operator have been proposed. Moreover, in [6] the authors proposed a descent method which is shown to be globally convergent when the operator F is nonsmooth, locally Lipschitz and (not necessarily strongly) monotone, and the feasible set X is bounded.

In this paper, we combine the descent method proposed in [4] and the Tikhonov-Browder regularization technique (see e.g. [1, 7]) to provide a globally convergent algorithm for solving a VI with nonsmooth, locally Lipschitz and (not necessarily strongly) monotone operator, over a not necessarily bounded feasible set. In section 2, we describe the combined method and prove its global convergence. In section 3, some numerical results are reported, which provide useful considerations on the parameter settings.

For the sake of convenience, we first recall some definitions which will be used in our further considerations. Given a symmetric positive definite matrix G, we denote by $\|\cdot\|_G$ the norm in $\mathbb{R}^n$ defined by $\|x\|_G = \sqrt{\langle x, Gx \rangle}$. In particular, $\|\cdot\|$ denotes the classical Euclidean norm induced by unit matrix I. The projection of a point $x \in \mathbb{R}^n$ onto the closed convex set X with respect to $\|\cdot\|_G$, denoted by $\Pi_{X,G}(x)$, is defined as the unique solution of the problem $\min_{y \in X} \|y - x\|_G$. We recall that the mapping

F is said to be *monotone on* X if

$$\langle F(x) - F(y), x - y\rangle \geq 0, \qquad \forall\, x, y \in X,$$

and *strongly monotone on* X *with constant* $\tau > 0$ if

$$\langle F(x) - F(y), x - y\rangle \geq \tau \|x - y\|^2, \qquad \forall\, x, y \in X.$$

In the rest of the paper we utilize the following assumptions.

(A1) *The set* $X \subseteq \mathbb{R}^n$ *is nonempty, closed, and convex.*

(A2) *The mapping* $F : Y \to \mathbb{R}^n$ *is locally Lipschitz at each point of an open convex set* Y *such that* $X \subset Y$*, and* F *is monotone on* Y.

2 A combined descent and regularization method

In this section we present a combined descent and regularization approach to solve a VI with nonsmooth, locally Lipschitz and (not necessarily strongly) monotone operator, over a (not necessarily bounded) closed and convex feasible set. The Tikhonov-Browder regularization method was originally proposed in [1, 7] to overcome some ill-posedness related difficulties in the resolution of VIs. Its basic idea is to substitute the original VI with a sequence of "regularized" VIs, obtained by replacing the operator F by $F + \varepsilon\, I$ where ε is a positive parameter. This regularization can be generalized to a nonlinear regularization where F is replaced by the mapping F_ε defined as:

$$F_\varepsilon(x) = F(x) + \varepsilon\, M(x),$$

where $\varepsilon > 0$ and $M : \mathbb{R}^n \to \mathbb{R}^n$ is locally Lipschitz and strongly monotone on Y. The regularization mapping F_ε leads to the following auxiliary VI problem: find a point $x_\varepsilon^* \in X$ such that

$$\langle F_\varepsilon(x_\varepsilon^*), x - x_\varepsilon^*\rangle \geq 0, \qquad \forall\, x \in X. \qquad (\mathrm{P}_\varepsilon)$$

Note that, under assumptions $(A1) - (A2)$, the mapping F_ε is locally Lipschitz continuous (generally nonsmooth) and strongly monotone on Y for each $\varepsilon > 0$, hence each auxiliary problem (P_ε) has a unique solution x_ε^*. Furthermore, if the set $\mathcal{S}$ of solutions of the original problem (P) is nonempty, then it follows from [2, Theorem 12.2.5] that the family of solutions $\{x_\varepsilon^*\}$ converges to a solution x^* of (P), as ε tends to zero, and x^* is uniquely characterized by the following property:

$$\langle M(x^*), x - x^*\rangle \geq 0, \qquad \forall\, x \in \mathcal{S}. \qquad (1)$$

In particular, if M is the identity map (i.e. we consider the Tikhonov-Browder regularization), then x^* is the solution of (P) with least Euclidean norm.

In order to solve each auxiliary problem (P_ε), we can apply the descent method described in [4] which is globally convergent to the solution of a VI with locally Lipschitz and strongly monotone operator. This method is based on the minimization of a gap function with an Armijo-type line search. To this end, we consider the following gap function [3] associated to the problem (P_ε):

$$\begin{aligned}\varphi_\varepsilon(x) &= \max_{y \in X} \left[\langle F_\varepsilon(x), x - y\rangle - \tfrac{\varepsilon}{2}\, \|x - y\|_G^2\right] \\ &= \langle F_\varepsilon(x), x - y_\varepsilon(x)\rangle - \tfrac{\varepsilon}{2}\, \|x - y_\varepsilon(x)\|_G^2,\end{aligned} \qquad (2)$$

where $y_\varepsilon(x) = \Pi_{X,G}\left(x - (\varepsilon\, G)^{-1} F_\varepsilon(x)\right)$ is the unique maximizer. Since $\varphi_\varepsilon(x) \geq 0$ for all $x \in X$ and $\varphi_\varepsilon(x^*) = 0$ if and only if x^* is a solution of (P_ε), the problem (P_ε) is equivalent to the following constrained optimization problem:

$$\min_{x \in X}\ \varphi_\varepsilon(x). \qquad (3)$$

Applying the descent method described in [4] for minimizing the gap function φ_ε, we can find an approximate solution of the problem (P_ε). However, in order to obtain the convergence to a solution of the original problem (P), we need to guarantee that the error for the solution of (P_ε) tends to zero, as ε tends to zero. To this end, we can exploit the following result which establishes an error bound for the solution of the auxiliary problem (P_ε) with the help of the gap function φ_ε.

Proposition 1. *Let assumptions* $(A1) - (A2)$ *be fulfilled and the map* M *be strongly monotone on* Y *with constant* $\tau > 0$*. Then for each* $\varepsilon > 0$ *we have:*

$$\varphi_\varepsilon(x) \geq \varepsilon\, C\, \|x - x_\varepsilon^*\|^2, \qquad \forall\, x \in X, \qquad (4)$$

where

$$C = \begin{cases} \tau - \dfrac{1}{2}\, \lambda_{max}(G) & \text{if } \tau \geq \lambda_{max}(G), \\ \dfrac{\tau^2}{2\, \lambda_{max}(G)} & \text{if } \tau < \lambda_{max}(G), \end{cases}$$

and $\lambda_{max}(G)$ *is the maximum eigenvalue of* G.

Proof. It is well known that

$$\max_{\|x\|=1} \langle x, G\, x\rangle = \lambda_{\max}(G),$$

where $\lambda_{\max}(G)$ is the maximum eigenvalue of G. Hence, for each $x \in \mathbb{R}^n$ one has

$$\|x\|_G^2 = \langle x, G\, x\rangle \leq \lambda_{\max}(G)\, \|x\|^2. \qquad (5)$$

Let us choose an arbitrary point $x \in X$ and $\mu \in (0,1]$, we set $x(\mu) = \mu\, x^*_\varepsilon + (1-\mu)\, x$. Taking into account the strong monotonicity of M and the relation (5), we obtain:

$$\begin{aligned}
\varphi_\varepsilon(x) &\geq \langle F_\varepsilon(x), x - x(\mu)\rangle - \tfrac{\varepsilon}{2}\,\|x - x(\mu)\|^2_G \\
&= \mu\,\langle F_\varepsilon(x), x - x^*_\varepsilon\rangle - \tfrac{\varepsilon}{2}\,\mu^2\,\|x - x^*_\varepsilon\|^2_G \\
&\geq \mu\,\left[\langle F_\varepsilon(x^*_\varepsilon), x - x^*_\varepsilon\rangle + \varepsilon\,\tau\,\|x - x^*_\varepsilon\|^2\right] + \\
&\quad -\tfrac{\varepsilon}{2}\,\mu^2\,\|x - x^*_\varepsilon\|^2_G \\
&\geq \mu\,\varepsilon\,\tau\,\|x - x^*_\varepsilon\|^2 - \tfrac{\varepsilon}{2}\,\mu^2\,\|x - x^*_\varepsilon\|^2_G \\
&\geq \varepsilon\,\left[\mu\,\tau - \tfrac{1}{2}\,\mu^2\,\lambda_{\max}(G)\right]\,\|x - x^*_\varepsilon\|^2.
\end{aligned}$$

Therefore

$$\begin{aligned}
\varphi_\varepsilon(x) &\geq \varepsilon \max_{\mu\in(0,1]}\left[\mu\,\tau - \frac{1}{2}\,\mu^2\,\lambda_{\max}(G)\right]\,\|x - x^*_\varepsilon\|^2 \\
&= \varepsilon\, C\,\|x - x^*_\varepsilon\|^2,
\end{aligned}$$

where

$$C = \begin{cases} \tau - \frac{1}{2}\,\lambda_{\max}(G) & \text{if } \tau \geq \lambda_{\max}(G), \\ \frac{\tau^2}{2\,\lambda_{\max}(G)} & \text{if } \tau < \lambda_{\max}(G). \end{cases}$$

□

Now we state the algorithm for solving the original problem (P), obtained combining the nonlinear regularization with the descent method proposed in [4] applied to the gap function φ_ε to approximate the solution of (P_ε).

Algorithm

0. (Initial step)
   Let G be a symmetric positive definite matrix.
   Let $\{\delta_k\}$ and $\{\varepsilon_k\}$ be sequences decreasing to 0.
   Let $\gamma \in (0,1)$ and $\{\beta_k\}$ be a sequence such that $\beta_k \in (0,1)$ for all $k \in \mathbb{N}$.
   Choose any $x^0 \in X$ and set $k = 1$.
1. (Minimization of φ_{ε_k})
   1a. (Initialization)
   Set $i = 0$ and $z^0 = x^{k-1}$.
   1b. (Stopping criterion)
   If $\varphi_{\varepsilon_k}(z^i) \leq \varepsilon_k\,\delta_k$, then go to step 2.
   1c. (Line search)
   Compute $y^i = \Pi_{X,G}\left(z^i - (\varepsilon_k\, G)^{-1} F_{\varepsilon_k}(z^i)\right)$
   set $d^i = y^i - z^i$
   compute the smallest nonnegative integer m such that:
   $$\varphi_{\varepsilon_k}(z^i + \gamma^m\, d^i) - \varphi_{\varepsilon_k}(z^i) \leq -\beta_k\,\gamma^m\,\|d^i\|^2.$$
   1d. (Update of z^i)
   Set $z^{i+1} = z^i + \gamma^m\, d^i$, $i = i+1$, and return to step 1b.
2. (Update of x^k)
   Set $x^k = z^i$, $k = k+1$, and return to step 1.

Theorem 2. *Let assumptions $(A1)$ – $(A2)$ are fulfilled and that (P) has a solution. Let the map M be locally Lipschitz and strongly monotone on Y with constant $\tau > 0$ and $\beta_k < \tau\,\varepsilon_k$ for all $k \in \mathbb{N}$. Then the sequence $\{x^k\}$ generated by the algorithm converges to the solution x^* of (P) such that (1) holds.*

Proof. For each iteration k, the mapping F_{ε_k} is locally Lipschitz and strongly monotone on Y, that is it satisfies the convergence conditions for the algorithm described in [4], and thus $\lim_{i\to\infty} \varphi_{\varepsilon_k}(z^i) = 0$. Therefore, for each outer iteration k, the stopping criterion at step 1b is verified after a finite number of inner iterations i, and hence the sequence $\{x^k\}$ is well defined.

Moreover, from Proposition 1, we have:

$$\varphi_{\varepsilon_k}(x^k) \geq \varepsilon_k\, C\,\|x^k - x^*_{\varepsilon_k}\|^2, \qquad \forall\, k \in \mathbb{N}, \quad (6)$$

where

$$C = \begin{cases} \tau - \dfrac{\lambda_{\max}(G)}{2} & \text{if } \tau \geq \lambda_{\max}(G), \\ \dfrac{\tau^2}{2\,\lambda_{\max}(G)} & \text{if } \tau < \lambda_{\max}(G). \end{cases}$$

¿From (6) and the stopping criterion 1b, we obtain:

$$\|x^k - x^*_{\varepsilon_k}\|^2 \leq \frac{\varphi_{\varepsilon_k}(x^k)}{\varepsilon_k\, C} \leq \frac{\varepsilon_k\, \delta_k}{\varepsilon_k\, C} = \frac{\delta_k}{C} \qquad \forall\, k \in \mathbb{N}.$$

Since $\lim_{k\to\infty} \delta_k = 0$, one has

$$\lim_{k\to\infty} \|x^k - x^*_{\varepsilon_k}\| = 0 \tag{7}$$

Furthermore, on account of [2, Theorem 12.2.5] it follows that

$$\lim_{k\to\infty} x^*_{\varepsilon_k} = x^*, \tag{8}$$

where x^* is the solution of (P) such that (1) holds. Finally, since we have:

$$\|x^k - x^*\| \leq \|x^k - x^*_{\varepsilon_k}\| + \|x^*_{\varepsilon_k} - x^*\|,$$

from (7) and (8) we conclude that $\lim_{k\to\infty} x^k = x^*$. □

Remark 3. *Observe that, instead of using the descent method proposed in [4], based on an Armijo-type line search, it could be analogously considered the combination of the descent method proposed in [5], based on an exact line search rule, with the nonlinear regularization.*

3 Numerical experiments

In this section we show some numerical results for the algorithm proposed in Section 2 on a couple of examples and we provide some comments about the sensitivity of the parameters to the computational effort. The algorithm has been implemented in MATLAB 7.0.4 and the regularization map M has been set as the identity map. As stopping criterion of the algorithm we used the natural residual: $\|x - \Pi_{X,I}(x - F(x))\|_\infty < 10^{-4}$. In both the test problems we chose the feasible set $X = [1, +\infty) \times \cdots \times [1, +\infty)$ and the map $F : \mathbb{R}^n \to \mathbb{R}^n$ as

$$F(x) = A\, x + H(x), \tag{9}$$

where A is a random skew-symmetric matrix and $H : \mathbb{R}^n \to \mathbb{R}^n$ is such that each component H_i is a locally Lipschitz nondecreasing function of the only variable x_i, for all $i = 1, \ldots, n$. Under these conditions it is easy to check that F is a locally Lipschitz and monotone (but not strongly monotone) map on $\mathbb{R}^n$. We set the sequence $\beta_k = \beta\, \varepsilon_k$ where $\beta \in (0, 1)$.

Example 3.1 Consider the VI problem where the map F has the form (9) with

$$A = \begin{pmatrix} 0 & 6 & -2 & -5 & 2 \\ -6 & 0 & -5 & -1 & -4 \\ 2 & 5 & 0 & 0 & 1 \\ 5 & 1 & 0 & 0 & 1 \\ -2 & 4 & -1 & -1 & 0 \end{pmatrix}$$

and $H_i(x) = \max\{x_i^2, 9\}$ for all $i = 1, \ldots, 5$. Preliminary computational results show that setting $G = 100\, I$, $\delta_k = 1/k$, $\varepsilon_k = 1/10^k$, $\gamma = 0.1$, and $\beta_k = 0.5\, \varepsilon_k$ provides a good parameter choice. We applied the algorithm, with such choice of parameters, to solve the considered example starting from 20 points randomly chosen in the box $[1, 10] \times \cdots \times [1, 10]$. Numerical results are summarized in Table 1 containing seven columns: starting point, number of outer iterations, number of inner iterations, number of projections, number of evaluations of the operator, natural residual at x, and approximate solution found x.

Table 1 shows that the algorithm is quite robust with respect to the starting point. In fact, besides converging to the same solution, as proved in Theorem 2, the number of outer iterations is always equal to 5 and the number of inner iterations is always between 10 and 16. The number of projections and of evaluations is stable as well, always between 51 and 64, and between 61 and 80, respectively.

Computational tests have been then carried out to investigate the behavior of the algorithm with different parameter values. First the behavior with respect to different choices of the matrix G is shown in Table 2. The other parameters are set as in Table 1: $\delta_k = 1/k$, $\varepsilon_k = 1/10^k$, $\gamma = 0.1$, and $\beta_k = 0.5\, \varepsilon_k$. Each different choice of the matrix G has been tested over 100 starting points randomly chosen in the box $[1, 10] \times \cdots \times [1, 10]$. In Table 2, for each choice of G, the average and maximum number of outer iterations, inner iterations, projections, and evaluations of F are given. Computational results show that the best behavior is provided by setting $G = 100\, I$. In fact, this choice provides the minimum average number of inner iterations, projections, and evaluations of F. Besides, even the maximum number of projections (and evaluations of F) is smaller than average number provided by other choices of G. However, the number of projections and evaluations at most double even for the worst choices of G. Thus, such parameter seems less significant than others.

In Table 3 results with different values of the sequence δ_k are shown, obtained keeping values of other parameters as in Table 1. For each choice of δ_k, the average and maximum number of outer iterations, inner iterations, projections, and evaluation of F are given. In Table 4 results with different values of the sequence ε_k are given. Results are obtained keeping values of other parameters as in Table 1. For each choice of ε_k, the average and maximum number of outer iterations, inner iterations, projections, and evaluation of F are given.

From the tables the best choice of δ_k seems to be $1/k$, while the best choice of ε_k is $1/10^k$. Results show that the choice of ε_k is more significant than the

choice of δ_k. In fact, choosing a δ_k value different from the best one causes an increase in the number of projections and evaluations of F, but they at most double even for the worst choice. On the other hand, a bad choice of ε_k may cause an increase of the number of projections and evaluations up to 10 times compared to the choice $\varepsilon_k = 1/10^k$. Results suggest a general rule for choosing δ_k and ε_k: they suggest to choose a "quite fast" decreasing sequence for ε_k (although "not too fast"), while they suggest to choose a "slow" decreasing sequence for δ_k.

In Table 5 results with different values of parameter γ (from 0.1 to 0.9) are shown. Results are obtained keeping values of other parameters as in Table 1. For each choice of γ, the average and maximum number of outer iterations, inner iterations, projections, and evaluation of F are given. According to the results the algorithm seems to perform well for $0.1 \leq \gamma \leq 0.2$, while the number of needed iterations, projections, and evaluations increases for $\gamma \geq 0.3$.

Finally, in Table 6 we show results with different values of the sequence β_k. We choose $\beta_k = \beta\,\varepsilon_k$ where β is between 0.1 and 0.9. Results are obtained keeping values of other parameters as in Table 1. The choice of β seems to be less important than the choice of γ: in fact the average number of projections is between 54 and 57 for any chosen value, and the average number of needed evaluations is between 67 and 70.

For the first considered example the most important role seems to be played by ε_k and γ, while G, δ_k, and β seem to be less important. Although γ does not influence as ε_k, its value has a quite significant importance.

Example 3.2 Consider the VI problem where the map F has the form (9) with

$$A = \begin{pmatrix} 0 & 0 & 0 & -2 & -4 & -1 & -3 & 1 & 3 & -4 \\ 0 & 0 & -4 & -3 & 1 & 1 & -2 & -1 & -1 & 1 \\ 0 & 4 & 0 & -2 & 2 & 2 & -3 & 2 & 1 & -1 \\ 2 & 3 & 2 & 0 & -1 & 4 & -2 & -1 & 0 & 1 \\ 4 & -1 & -2 & 1 & 0 & -2 & 0 & -2 & -1 & 1 \\ 1 & -1 & -2 & -4 & 2 & 0 & -3 & 2 & -1 & -2 \\ 3 & 2 & 3 & 2 & 0 & 3 & 0 & 0 & 4 & -3 \\ -1 & 1 & -2 & 1 & 2 & -2 & 0 & 0 & 1 & -3 \\ -3 & 1 & -1 & 0 & 1 & 1 & -4 & -1 & 0 & 1 \\ 4 & -1 & 1 & -1 & -1 & 2 & 3 & 3 & -1 & 0 \end{pmatrix}$$

and $H_i(x) = \max\{e^{x_i}, 6\}$ for all $i = 1, \ldots, 10$. Preliminary computational results show that a good choice of parameters also for this example is given by $G = 100\,I$, $\delta_k = 1/k$, $\varepsilon_k = 1/10^k$, $\gamma = 0.1$, and $\beta_k = 0.5\,\varepsilon_k$. Results for such parameters with 20 starting points randomly chosen in the box $[1, 10] \times \cdots \times [1, 10]$ are shown in Table 7.

In Tables 8-12 the behavior of the algorithm with respect to different choices of the parameters is investigated.

In Table 8 the behavior with respect to different choices of the matrix G is shown. The other parameters are set as in Table 7: $\delta_k = 1/k$, $\varepsilon_k = 1/10^k$, $\gamma = 0.1$, and $\beta_k = 0.5\,\varepsilon_k$. Each different choice of the matrix G has been tested over 100 starting points randomly chosen in the box $[1, 10] \times \cdots \times [1, 10]$. In Table 8, for each choice of G, the average and maximum number of outer iterations, inner iterations, projections, and evaluations of F are given.

In Tables 9 and 10 results for different choices of sequences δ_k and ε_k are given, respectively, keeping fixed the other parameters.

Finally, in Tables 11 and 12 results for different choices of sequences γ and β are given, respectively, keeping fixed the other parameters.

As pointed out by the results on the first example, ε_k and γ seem to play a more significant role than G, δ_k, and β.

4 Conclusions

We proposed an algorithm which combines a well known descent method and the Tikhonov-Browder regularization technique and provides global convergence in solving a VI with nonsmooth, locally Lipschitz and (not necessarily strongly) monotone operator, on a (not necessarily bounded) closed and convex feasible set. We showed some computational experiments which provide insight on the algorithm behavior with respect to the parameters choice. Further, results give useful suggestions for the parameters settings.

References:

[1] F. E. Browder, "Existence and approximation of solutions of nonlinear variational inequalities", Proceedings of the National Academy of Science USA, vol. 56, pp. 1080–1086, 1966.

[2] F. Facchinei and J.-S. Pang, Finite-dimensional variational inequalities and complementarity problems, Springer: Berlin, 2003.

[3] M. Fukushima, "Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems", Mathematical Programming, vol. 53, pp. 99–110, 1992.

[4] I. Konnov, "Descent methods for nonsmooth variational inequalities", Computational Mathematics and Mathematical Physics, vol. 46, pp. 1186–1192, 2001.

[5] I. Konnov, B. Panicucci, M. Passacantando, "A derivative-free descent method for nonsmooth

variational inequalities", International Journal of Pure and Applied Mathematics, to appear.

[6] B. Panicucci, M. Passacantando, M. Pappalardo, "A globally convergent descent method for non-smooth variational inequalities", Computational Optimization and Applications, to appear.

[7] A. N. Tikhonov, V. Y. Arsenin, Solution of ill-posed problems, Wiley, New York, 1977.

[8] H. Xu, "Regularized gap function and D-gap function for nonsmooth variational inequalities", in "Optimization and Related topics" edited by A. Rubinov and B. Glover, Kluwer Academic Publishers: Dordrecht, pp. 153–176, 2001.

Table 1: Numerical results for Example 3.1 with $G = 100\,I$, $\delta_k = 1/k$, $\varepsilon_k = 1/10^k$, $\gamma = 0.1$, and $\beta_k = 0.5\,\varepsilon_k$.

starting point	outer iter.	inner iter.	proj.	eval. of F	natural residual	approximate solution
$(10, 2, 2, 7, 1)$	5	11	53	64	6.55E-05	$(1, 3.999992, 1, 1, 1)$
$(8, 9, 8, 7, 4)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(4, 7, 6, 3, 8)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(4, 3, 10, 3, 8)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(6, 3, 3, 5, 8)$	5	14	59	73	6.47E-05	$(1, 3.999992, 1, 1, 1)$
$(2, 10, 4, 2, 9)$	5	11	53	64	6.49E-05	$(1, 3.999992, 1, 1, 1)$
$(3, 7, 10, 4, 5)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(4, 4, 10, 10, 6)$	5	15	61	76	6.44E-05	$(1, 3.999992, 1, 1, 1)$
$(10, 9, 1, 6, 2)$	5	12	55	67	6.75E-05	$(1, 3.999992, 1, 1, 1)$
$(2, 9, 10, 3, 9)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(9, 8, 7, 3, 9)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(10, 7, 7, 7, 7)$	5	16	64	80	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(6, 5, 3, 4, 2)$	5	12	55	67	6.53E-05	$(1, 3.999992, 1, 1, 1)$
$(5, 1, 6, 8, 6)$	5	11	53	64	6.61E-05	$(1, 3.999992, 1, 1, 1)$
$(1, 10, 10, 8, 6)$	5	12	55	67	6.74E-05	$(1, 3.999992, 1, 1, 1)$
$(3, 5, 8, 3, 8)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(5, 2, 5, 3, 9)$	5	14	59	73	6.44E-05	$(1, 3.999992, 1, 1, 1)$
$(2, 9, 7, 3, 9)$	5	13	57	70	6.66E-05	$(1, 3.999992, 1, 1, 1)$
$(9, 7, 2, 3, 9)$	5	15	61	76	6.48E-05	$(1, 3.999992, 1, 1, 1)$
$(5, 6, 2, 6, 2)$	5	10	51	61	6.75E-05	$(1, 3.999992, 1, 1, 1)$

Table 2: Behavior of the algorithm with respect to different choices of the matrix G.

	outer iterations		inner iterations		projections		evaluations of F	
G	avg.	max	avg.	max	avg.	max	avg.	max
I	5	5	20.1	21	119.2	123	139.3	144
$10\,I$	5	5	16.9	18	83.6	87	100.5	105
$100\,I$	5	5	13.3	18	57.2	70	70.5	88
$200\,I$	5	5	22.6	23	94.6	98	117.2	121
$500\,I$	5	5	35.8	46	112.4	146	148.2	192

Table 3: Behavior of the algorithm with respect to different δ_k choice.

	outer iterations		inner iterations		projections		evaluations of F	
δ_k	avg.	max	avg.	max	avg.	max	avg.	max
$10/k$	5.9	6	12.7	18	62.8	79	75.4	97
$1/k$	5	5	12.6	17	55.1	67	67.7	84
$1/k^2$	5	5	12.9	18	56.1	70	69.0	88
$1/k^3$	5	5	14.9	19	67.0	77	81.8	96
$1/2^k$	5	5	13.3	17	58.1	67	71.4	84
$1/10^k$	5	5	21.4	25	94.8	104	116.3	129
$1/100^k$	5	5	30.5	35	139.1	151	169.6	186

Table 4: Behavior of the algorithm with respect to different ε_k choice.

	outer iterations		inner iterations		projections		evaluations of F	
ε_k	avg.	max	avg.	max	avg.	max	avg.	max
$1/k^2$	214.8	218	44.7	53	587.5	623	632.1	674
$1/k^3$	39.2	40	36.7	44	196.9	228	233.6	272
$1/2^k$	14.1	17	21.7	28	90.0	120	111.7	148
$1/10^k$	5	5	12.9	18	56.0	70	69.0	88
$1/100^k$	3	3	14.7	16	69.8	76	84.5	92

Table 5: Behavior of the algorithm with respect to different γ choice.

	outer iterations		inner iterations		projections		evaluations of F	
γ	avg.	max	avg.	max	avg.	max	avg.	max
0.1	5	5	13.3	19	56.9	73	70.1	92
0.2	5	6	16.1	22	76.1	111	92.3	133
0.3	5	5	58	66	397.8	459	455.8	525
0.4	5	5	12.8	15	81.3	90	94.1	105
0.5	5.4	6	86.8	229	867.5	961	954.3	1190
0.6	5	6	21	34	186.2	444	207.2	478
0.7	5	5	52.9	66	604.3	706	657.1	772
0.8	5	5	65.3	74	1619.3	1806	1684.7	1880
0.9	5.2	6	134.6	202	7950.6	13404	8085.3	13606

Table 6: Behavior of the algorithm with respect to different β choice.

	outer iterations		inner iterations		projections		evaluations of F	
β	avg.	max	avg.	max	avg.	max	avg.	max
0.1	5	5	12.9	18	56.1	70	68.9	88
0.2	5	6	12.9	17	56.2	67	69.1	84
0.3	5	5	12.9	18	56.2	70	69.1	88
0.4	5	5	13.1	18	56.5	70	69.6	88
0.5	5	5	12.3	16	54.5	64	66.8	80
0.6	5	5	12.4	16	54.6	64	67.0	80
0.7	5	5	13.1	18	56.7	70	69.8	88
0.8	5	5	12.8	18	55.4	70	68.2	88
0.9	5	5	12.8	18	55.9	70	68.6	88

Table 7: Numerical results for Example 3.2 with $G = 100\,I$, $\delta_k = 1/k$, $\varepsilon_k = 1/10^k$, $\gamma = 0.1$, and $\beta_k = 0.5\,\varepsilon_k$.

starting point	outer iter.	inner iter.	proj.	eval. of F	natural residual	approximate solution
$(2,8,1,8,3,2,3,3,8,9)$	5	41	159	200	3.01E-05	$(2.158317, 2.037456, 1, 1, 1, 2.165077, 1, 1, 1.836163, 1)$
$(10,3,2,7,9,2,4,7,4,7)$	5	34	124	158	3.07E-05	$(2.158317, 2.037456, 1, 1, 1, 2.165077, 1, 1, 1.836163, 1)$
$(6,2,2,2,7,5,7,1,8,7)$	5	24	92	116	6.00E-05	$(2.158319, 2.037455, 1, 1, 1, 2.165073, 1, 1, 1.836165, 1)$
$(5,6,9,7,7,5,8,6,5,7)$	5	23	88	111	5.31E-05	$(2.158318, 2.037455, 1, 1, 1, 2.165074, 1, 1, 1.836165, 1)$
$(3,9,4,2,3,5,3,4,6,6)$	5	40	140	180	6.43E-05	$(2.158320, 2.037454, 1, 1, 1, 2.165073, 1, 1, 1.836165, 1)$
$(3,3,8,4,3,2,5,9,8,4)$	5	42	145	187	5.57E-05	$(2.158319, 2.037455, 1, 1, 1, 2.165074, 1, 1, 1.836164, 1)$
$(3,9,6,3,1,1,3,6,10,8)$	5	19	76	95	4.60E-05	$(2.158322, 2.037454, 1, 1, 1, 2.165074, 1, 1, 1.836162, 1)$
$(7,3,9,9,10,5,5,8,2,2)$	5	22	83	105	3.72E-05	$(2.158318, 2.037456, 1, 1, 1, 2.165078, 1, 1, 1.836158, 1)$
$(2,5,5,8,6,9,4,3,7,8)$	5	28	107	135	3.13E-05	$(2.158317, 2.037456, 1, 1, 1, 2.165077, 1, 1, 1.836164, 1)$
$(9,2,3,3,8,2,2,3,1,4)$	5	32	124	156	3.02E-05	$(2.158317, 2.037456, 1, 1, 1, 2.165077, 1, 1, 1.836163, 1)$
$(2,2,4,6,4,2,5,10,10,6)$	5	24	91	115	3.03E-05	$(2.158317, 2.037456, 1, 1, 1, 2.165077, 1, 1, 1.836163, 1)$
$(4,8,5,9,3,2,4,5,5,8)$	5	24	88	112	5.59E-05	$(2.158319, 2.037455, 1, 1, 1, 2.165074, 1, 1, 1.836164, 1)$
$(4,2,2,6,2,10,6,9,5,7)$	5	32	118	150	3.18E-05	$(2.158318, 2.037456, 1, 1, 1, 2.165077, 1, 1, 1.836163, 1)$
$(8,7,5,8,2,6,5,4,2,7)$	5	25	98	123	2.99E-05	$(2.158317, 2.037456, 1, 1, 1, 2.165077, 1, 1, 1.836163, 1)$
$(8,8,6,2,1,4,5,4,3,4)$	5	27	100	127	5.35E-05	$(2.158318, 2.037455, 1, 1, 1, 2.165074, 1, 1, 1.836165, 1)$
$(7,8,8,1,8,7,2,7,6,3)$	5	23	88	111	5.31E-05	$(2.158318, 2.037455, 1, 1, 1, 2.165074, 1, 1, 1.836165, 1)$
$(10,2,7,3,3,9,7,3,5,4)$	5	36	126	162	4.57E-05	$(2.158323, 2.037454, 1, 1, 1, 2.165074, 1, 1, 1.836162, 1)$
$(7,2,2,9,2,7,6,5,3,6)$	5	33	116	149	4.30E-05	$(2.158315, 2.037456, 1, 1, 1, 2.165076, 1, 1, 1.836164, 1)$
$(4,8,8,6,4,3,2,4,9,7)$	5	22	85	107	4.97E-05	$(2.158322, 2.037454, 1, 1, 1, 2.165074, 1, 1, 1.836163, 1)$
$(1,3,4,9,9,9,1,8,1,5)$	5	34	124	158	6.25E-05	$(2.158319, 2.037455, 1, 1, 1, 2.165073, 1, 1, 1.836166, 1)$

Table 8: Behavior of the algorithm with respect to different choices of the matrix G.

	outer iterations		inner iterations		projections		evaluations of F	
G	avg.	max	avg.	max	avg.	max	avg.	max
I	5	5	75.2	80	444.5	464	519.7	543
$10\,I$	5	5	29	46	130.8	196	159.9	242
$100\,I$	5	5	27.4	43	103.4	160	130.8	202
$200\,I$	5.1	6	47.6	183	182.1	726	229.6	909
$500\,I$	5.7	6	81	155	337.9	693	418.9	848

Table 9: Behavior of the algorithm with respect to different δ_k choice.

	outer iterations		inner iterations		projections		evaluations of F	
δ_k	avg.	max	avg.	max	avg.	max	avg.	max
$10/k$	5.2	6	156.3	7429	839.6	41486	995.8	48915
$1/k$	5	5	25.9	44	98.2	168	124.1	212
$1/k^2$	5	5	27.2	46	104.0	173	131.2	219
$1/k^3$	5	5	29.2	43	112.2	165	141.4	208
$1/2^k$	5	5	28.0	44	105.8	164	133.8	207
$1/10^k$	5	5	39.5	61	156.2	228	195.7	289
$1/100^k$	5	5	60.4	82	249.7	312	310	394

Table 10: Behavior of the algorithm with respect to different ε_k choice.

	outer iterations		inner iterations		projections		evaluations of F	
ε_k	avg.	max	avg.	max	avg.	max	avg.	max
$1/k^2$	151.1	167	49.2	72	461.9	537	511.1	596
$1/k^3$	29.8	34	40.8	59	190.7	242	231.5	297
$1/2^k$	15.6	16	41.3	67	166.3	246	207.7	313
$1/10^k$	5	5	27.4	43	103.1	160	130.6	202
$1/100^k$	3	3	39.5	47	172.6	197	212.1	244

Table 11: Behavior of the algorithm with respect to different γ choice.

	outer iterations		inner iterations		projections		evaluations of F	
γ	avg.	max	avg.	max	avg.	max	avg.	max
0.1	5	5	27.2	43	103.3	158	130.5	200
0.2	5	5	31.1	46	140.8	200	172.0	244
0.3	5	5	28.9	44	171.6	235	200.5	279
0.4	5	5	25.4	36	153.8	209	179.2	242
0.5	5	5	29.6	39	231.8	314	261.4	350
0.6	5	5	36.6	42	343.9	427	380.5	467
0.7	5	5	40.9	51	574.2	710	615.1	759
0.8	5	5	63.0	74	1367.9	1619	1430.9	1693
0.9	5	5	112.3	130	5242.9	5975	5355.2	6105

Table 12: Behavior of the algorithm with respect to different β choice.

	outer iterations		inner iterations		projections		evaluations of F	
β	avg.	max	avg.	max	avg.	max	avg.	max
0.1	5	5	26.3	47	100.2	170	126.5	217
0.2	5	5	26.7	43	100.6	169	127.2	209
0.3	5	5	26.0	42	99.1	159	125.1	201
0.4	5	5	26.7	44	101.3	175	128.0	219
0.5	5	5	25.5	40	96.4	152	121.9	192
0.6	5	5	29.0	46	108.7	173	137.7	219
0.7	5	5	27.9	86	105.4	331	133.3	417
0.8	5	5	38.3	123	147.2	488	185.5	611
0.9	5	5	47.5	237	185.0	948	232.6	1185