D-gap functions and descent techniques for solving equilibrium problems*

Giancarlo Bigi† Mauro Passacantando†

Abstract: A new algorithm for solving equilibrium problems with differentiable bifunctions is provided. The algorithm is based on descent directions of a suitable family of D-gap functions. Its convergence is proved under assumptions which do not guarantee the equivalence between the stationary points of the D-gap functions and the solutions of the equilibrium problem. Moreover, the algorithm does not require to set parameters according to thresholds which depend on regularity properties of the equilibrium bifunction. The results of preliminary numerical tests on Nash equilibrium problems with quadratic payoffs are reported. Finally, some numerical comparisons with other D-gap algorithms are drawn relying on some further tests on linear equilibrium problems.

Keywords: Equilibrium problem, D-gap function, descent directions, monotonicity.

1 Introduction

In this paper, we consider the following *equilibrium problem*:

$$\text{find } x^* \in C \text{ s.t. } f(x^*, y) \geq 0, \qquad \forall\, y \in C, \tag{EP}$$

where $C \subset \mathbb{R}^n$ is a nonempty, closed and convex set and the equilibrium bifunction $f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ satisfies $f(x, x) = 0$ for all $x \in C$. This format provides a rather general setting which includes several mathematical models such as optimization, multiobjective optimization, variational inequalities, fixed point and complementarity problems, Nash equilibria in noncooperative games and inverse optimization (see e.g. [2, 6, 16]). Throughout all the paper we suppose also that f is continuously differentiable and $f(x, \cdot)$ is convex for all $x \in C$.

Many methods for computing equilibria have been developed, which can be divided into several classes: fixed point and extragradient methods, descent methods, proximal point and Tikhonov-Browder regularization methods (see the recent survey paper [2]). Often these methods extend those originally conceived for optimization or variational inequalities to the more general framework of equilibrium problems, exploiting the underlying common structure provided by *(EP)*.

In this paper we focus on the approach based on descent procedures. In general, descent methods rely on the reformulation of the equilibrium problem as an optimization problem through suitable merit functions. The so-called gap functions yield reformulations as constrained optimization problems (see [1, 3, 4, 9, 10, 12, 14, 15]), while the difference of two appropriate gap functions (D-gap function) leads to reformulations as unconstrained optimization problems (see [8, 13, 23, 24, 25]).

The D-gap function approach was introduced for variational inequalities in [17, 22]. The first methods were developed to solve strongly monotone variational inequalities via unconstrained optimization [11, 18, 19, 21, 22]. Later on, descent methods for monotone variational inequalities have been conceived relying on steps of unconstrained minimization with a sequence of different D-gap functions as objective function [20].

*This paper has been published in Journal of Global Optimization, vol. 62 (1), pp. 183-203, see: `https://link.springer.com/article/10.1007/s10898-014-0223-x`.

†Dipartimento di Informatica, Università di Pisa, Largo B.Pontecorvo 3, 56127 Pisa, Italia

In the framework of the equilibrium problem *(EP)* D-gap functions have been introduced in [13, 23] and solution methods which exploit them have been developed in [8, 13, 24, 25]. These methods need strong assumptions, in fact their convergence requires the strict or uniform strong monotonicity of the gradient mappings $\nabla_x f(x,\cdot)$: this assumption implies that all the stationary points of a D-gap function coincide with its global minima and hence with the solutions of *(EP)* [23]. Furthermore, the parameters of the algorithms in [8, 13] have to be set according to thresholds which depend on the constants of strong monotonicity and Lipschitz continuity of the above gradient mappings. As these values have to be known in advance, it is hard to implement these methods in a general framework.

To overcome these drawbacks, we develop a new solution method for *(EP)* relying on D-gap functions in the same fashion of [20]. In particular, a whole family of D-gap functions is exploited in order to preserve a sufficient decrease condition at each iteration of the algorithm, and this allows to deal with stationarity issues. In fact, the convergence of the method requires just the monotonicity of the mappings $\nabla_x f(x,\cdot)$: as a consequence, there may be stationary points of any given D-gap function which are not global minima and therefore do not solve *(EP)*. Furthermore, the method does not require Lipschitz continuity assumptions and hence no a priori knowledge of constants/thresholds is needed. Thus, the paper aims at providing a method which can be both easily implemented and applied to a wider class of equilibrium problems.

The paper is organized as follows. Section 2 provides basic results which play a key role in devising the method. In particular, bounds on the values of the D-gap functions are proved. Section 3 describes the solution method, addressing also possible improvements in the choice of the parameters. Since the convergence result requires the boundedness of the feasible region, conditions which allow to drop it are also addressed. Finally, Section 4 provides preliminary numerical tests to analyse the sensitivity of the algorithm with respect to its parameters and some numerical comparisons with other similar algorithms.

2 Gap and D-gap functions

A gap function for *(EP)* is a real-valued function which is non-negative on C and is 0 in C only at every solution of *(EP)*: its global minima over C coincide with the solution set of the equilibrium problem. The a priori knowledge of the optimal value is a powerful information in devising solution methods.

Auxiliary bifunctions are generally exploited together with f to build gap functions with good regularity properties. With this aim we consider a continuously differentiable bifunction $h : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ satisfying the following conditions:

- $h(x,y) \geq 0$ for all $x, y \in \mathbb{R}^n$ and $h(x,y) = 0$ if and only if $x = y$;
- $h(x,\cdot)$ is strongly convex uniformly in x, i.e., there exists $\tau > 0$ such that

$$h(x,z) \geq h(x,y) + \langle \nabla_y h(x,y), z-y \rangle + \tau ||z-y||^2$$

  holds for any $x, y, z \in \mathbb{R}^n$;
- $\nabla_y h(z,z) = 0$ for all $z \in \mathbb{R}^n$;
- $\nabla_y h(x,\cdot)$ is Lipschitz continuous uniformly in x, i.e., there exists $L > 0$ such that

$$\|\nabla_y h(x,y) - \nabla_y h(x,z)\| \leq L\, ||y-z||$$

  holds for any $x, y, z \in \mathbb{R}^n$;
- $\nabla_x h(x,y) = -\nabla_y h(x,y)$ for all $x, y \in \mathbb{R}^n$.

A bifunction with the above properties can be obtained just taking $h(x,y) = g(y-x)$ for some strongly convex function $g : \mathbb{R}^n \to \mathbb{R}_+$ with a Lipschitz gradient and $g(0) = 0$. The most typical choice is the square of the Euclidean norm.

Given any $\sigma > 0$, the value function

$$\varphi_\sigma(x) = -\min\{\ f(x,y) + \sigma\, h(x,y)\ :\ y \in C\ \} \tag{1}$$

is a gap function for *(EP)* (see, for instance, [15]). Since the objective function $f(x,\cdot) + \sigma h(x,\cdot)$ is strongly convex, the above optimization problem has a unique optimal solution $y_\sigma(x)$ which therefore satisfies the optimality condition

$$\langle \nabla_y f(x, y_\sigma(x)) + \sigma \nabla_y h(x, y_\sigma(x)), z - y_\sigma(x)\rangle \geq 0 \qquad \forall\ z \in C. \tag{2}$$

Moreover, $f(x,x) = h(x,x) = 0$ and the uniqueness of the optimal solution $y_\sigma(x)$ imply that the solution set of *(EP)* coincides with the fixed points of y_σ, i.e., x^* solves *(EP)* if and only if $y_\sigma(x^*) = x^*$. Furthermore, the mapping y_σ is continuous and the gap function φ_σ is continuously differentiable (see [2] and the references therein).

It is possible to reformulate *(EP)* as an unconstrained optimization problem, exploiting the difference of two gap functions. In fact, the so-called D-gap function

$$\varphi_{\alpha,\beta}(x) = \varphi_\alpha(x) - \varphi_\beta(x)$$

with $0 < \alpha < \beta$ is non-negative on $\mathbb{R}^n$ and is 0 only at every solution of *(EP)* (see [13, 23]). Therefore, its global minima on $\mathbb{R}^n$ coincide with the solution set of the equilibrium problem. Obviously, the D-gap function $\varphi_{\alpha,\beta}$ inherits the properties of the gap function (1): in particular, it can be rewritten as

$$\varphi_{\alpha,\beta}(x) = f(x, y_\beta(x)) - f(x, y_\alpha(x)) + \beta h(x, y_\beta(x)) - \alpha h(x, y_\alpha(x)) \tag{3}$$

and it is continuously differentiable with

$$\begin{aligned} \nabla\varphi_{\alpha,\beta}(x) = \quad & \nabla_x f(x, y_\beta(x)) - \nabla_x f(x, y_\alpha(x)) + \\ & +\beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)). \end{aligned} \tag{4}$$

The auxiliary bifunction h and the optimal solutions of the inner optimization problems provide the lower and upper bounds for the D-gap function given below.

Lemma 2.1. The inequalities

$$\varphi_{\alpha,\beta}(x) \geq (\beta - \alpha)\, h(x, y_\beta(x)) + \alpha\, \tau\, \|y_\beta(x) - y_\alpha(x)\|^2 \tag{5}$$

and

$$\varphi_{\alpha,\beta}(x) \leq (\beta - \alpha)\, h(x, y_\alpha(x)) - \beta\, \tau\, \|y_\beta(x) - y_\alpha(x)\|^2 \tag{6}$$

hold for any $x \in \mathbb{R}^n$ and $0 < \alpha < \beta$.

Proof. The convexity of $f(x,\cdot)$ and the strong convexity of $h(x,\cdot)$ imply

$$\begin{aligned} f(x, y_\beta(x)) \geq \quad & f(x, y_\alpha(x)) + \langle \nabla_y f(x, y_\alpha(x)), y_\beta(x) - y_\alpha(x)\rangle \\ h(x, y_\beta(x)) \geq \quad & h(x, y_\alpha(x)) + \langle \nabla_y h(x, y_\alpha(x)), y_\beta(x) - y_\alpha(x)\rangle + \\ & +\tau\, \|y_\beta(x) - y_\alpha(x)\|^2, \end{aligned}$$

and the optimality condition satisfied by $y_\alpha(x)$ gives

$$\langle \nabla_y f(x, y_\alpha(x)) + \alpha \nabla_y h(x, y_\alpha(x)), y_\beta(x) - y_\alpha(x)\rangle \geq 0.$$

Therefore, (5) follows from the chain of inequalities and equalities

$$\begin{aligned} 0 \quad & \leq \langle \nabla_y f(x, y_\alpha(x)) + \alpha\, \nabla_y h(x, y_\alpha(x)), y_\beta(x) - y_\alpha(x)\rangle \\ & \leq f(x, y_\beta(x)) - f(x, y_\alpha(x)) + \alpha\, h(x, y_\beta(x)) - \alpha\, h(x, y_\alpha(x)) + \\ & -\alpha\, \tau\, \|y_\beta(x) - y_\alpha(x)\|^2 \\ & = \varphi_{\alpha,\beta}(x) + (\alpha - \beta)\, h(x, y_\beta(x)) - \alpha\, \tau\, \|y_\beta(x) - y_\alpha(x)\|^2 \end{aligned}$$

where the equality holds thanks to (3). Exchanging the roles of $y_\alpha(x)$ and $y_\beta(x)$, the same argument proves (6). $\square$

The inequalities (5) and (6) improve the bounds given in [23, Proposition 3.1] and they extend those given in [20] with $h(x,y) = ||y-x||^2/2$ for variational inequalities to the more general equilibrium problem *(EP)*. Moreover, the reformulation of *(EP)* as an unconstrained optimization problem is a straightforward consequence of these bounds: if x^* solves *(EP)* or equivalently $y_\alpha(x^*) = y_\beta(x^*) = x^*$, then (6) implies $\varphi_{\alpha,\beta}(x^*) = 0$ since $\varphi_{\alpha,\beta}$ is non-negative; vice versa, if $\varphi_{\alpha,\beta}(x^*) = 0$, then (5) implies both $h(x^*, y_\beta(x^*)) = 0$ and $\|y_\beta(x^*) - y_\alpha(x^*)\| = 0$, hence $y_\alpha(x^*) = y_\beta(x^*) = x^*$. It is worth noting that if the D-gap function is 0 at some point, the feasibility of the point itself is guaranteed while this is not necessarily true for the gap function (1).

Inequality (5) guarantees also the inequality

$$h(x, y_\beta(x)) \leq \varphi_{\alpha,\beta}(x)/(\beta - \alpha). \tag{7}$$

Managing to make the right-hand side smaller and smaller would drive towards a solution of *(EP)*. To this aim further relationships between the auxiliary bifunction and the D-gap function come in to play.

Lemma 2.2. Let $y_\infty(x) := \arg\min\{\ h(x,y)\ :\ y \in C\ \}$. Then, the relationships

$$\lim_{\beta' \to +\infty} y_{\beta'}(x) = y_\infty(x) \tag{8}$$

and

$$\lim_{\beta' \to +\infty} \varphi_{\alpha,\beta'}(x)/(\beta' - \alpha) = h(x, y_\infty(x)) \leq \varphi_{\alpha,\beta}(x)/(\beta - \alpha) \tag{9}$$

hold for any $x \in \mathbb{R}^n$ and $0 < \alpha < \beta$.

Proof. First, notice that $y_{\beta'}(x) = \arg\min\{\beta'^{-1} f(x,y) + h(x,y)\ :\ y \in C\}$ for any $\beta' > 0$. The strong convexity of $h(x,\cdot)$ implies

$$\begin{aligned} h(x, y_{\beta'}(x)) \geq \quad & h(x, y_\infty(x)) + \langle \nabla_y h(x, y_\infty(x)), y_{\beta'}(x) - y_\infty(x)\rangle + \\ & + \tau\, \|y_{\beta'}(x) - y_\infty(x)\|^2. \end{aligned}$$

Since $y_\infty(x)$ minimizes $h(x,\cdot)$ over C, the first order optimality conditions imply

$$\langle \nabla_y h(x, y_\infty(x)), y_{\beta'}(x) - y_\infty(x)\rangle \geq 0$$

and therefore we get

$$h(x, y_{\beta'}(x)) \geq h(x, y_\infty(x)) + \tau\, \|y_{\beta'}(x) - y_\infty(x)\|^2. \tag{10}$$

On the other hand, we have

$$\beta'^{-1} f(x, y_{\beta'}(x)) + h(x, y_{\beta'}(x)) \leq \beta'^{-1} f(x, y_\infty(x)) + h(x, y_\infty(x)).$$

Thus, the following chain of inequalities hold

$$\begin{aligned} \tau\, \|y_{\beta'}(x) - y_\infty(x)\|^2 \quad & \leq h(x, y_{\beta'}(x)) - h(x, y_\infty(x)) \\ & \leq \beta'^{-1} \left[f(x, y_\infty(x)) - f(x, y_{\beta'}(x)) \right] \\ & \leq \beta'^{-1} \langle \nabla_y f(x, y_\infty(x)), y_\infty(x) - y_{\beta'}(x)\rangle \\ & \leq \beta'^{-1} \|\nabla_y f(x, y_\infty(x))\|\, \|y_{\beta'}(x) - y_\infty(x)\| \end{aligned}$$

taking into account the convexity of $f(x,\cdot)$. As a consequence we have

$$\|y_{\beta'}(x) - y_\infty(x)\| \leq \|\nabla_y f(x, y_\infty(x))\| / \tau\, \beta'.$$

and hence (8) follows just taking the limit as $\beta' \to +\infty$.

Taking into account that f and h are continuous, the equality in (9) follows:

$$\begin{aligned}\lim_{\beta'\to+\infty} \varphi_{\alpha,\beta'}(x)/(\beta'-\alpha) &= \lim_{\beta'\to+\infty} \varphi_\alpha(x) - \varphi_{\beta'}(x)/(\beta'-\alpha)\\ &= \lim_{\beta'\to+\infty} (\varphi_\alpha(x) + f(x, y_{\beta'}(x)) + \beta' h(x, y_{\beta'}(x)))/(\beta'-\alpha)\\ &= h(x, y_\infty(x)).\end{aligned}$$

Finally, (5) and (10) imply

$$\begin{aligned}\varphi_{\alpha,\beta}(x)/(\beta-\alpha) &\geq h(x, y_\beta(x))\\ &\geq h(x, y_\infty(x)) + \tau \, \|y_\beta(x) - y_\infty(x)\|^2\\ &\geq h(x, y_\infty(x)),\end{aligned}$$

i.e., the inequality in (9) holds. □

If h is an actual (squared) distance between points, then $y_\infty(x)$ is the corresponding projection of x onto C. In any case, the properties of h guarantee $y_\infty(x) = x$ for any $x \in C$: whenever a feasible point is taken, the limit in (9) is 0 and choosing β large enough allows to make the right-hand side of (7) as small as desired. Anyway, this is not enough to devise an algorithm: the above lemma requires $\beta \to +\infty$ and (7) would simply provide the obvious statement $h(x, y_\infty(x)) = 0$ for a feasible x. A key tool to overcome these issues is controlling the decrease of the D-gap function along search directions by the value of the right-hand side of (7) at the current iterate within a descent type method (see Theorem 3.1(a) and condition (22) in the next section).

3 Solution method

Methods based on D-gap functions generally require the strict or strong monotonicity of the gradient map $\nabla_x f(x,\cdot)$ for any $x \in \mathbb{R}^n$ [8, 13, 24, 25]. Under this strict (strong) monotonicity assumption any stationary point of $\varphi_{\alpha,\beta}$ is actually a global minimum and therefore solves *(EP)* (see [23, 24]) though $\varphi_{\alpha,\beta}$ is not necessarily convex: therefore, in principle, any local minimization algorithm could be exploited.

We aim at developing a solution method under assumptions which do not guarantee the above property. The method of this section requires just that $\nabla_x f(x,\cdot)$ is monotone on C for any $x \in \mathbb{R}^n$, i.e.,

$$\langle \nabla_x f(x,y) - \nabla_x f(x,z), y - z\rangle \geq 0, \qquad \forall\, x \in \mathbb{R}^n, \forall\, y, z \in C. \tag{11}$$

Indeed, condition (11) does not guarantee that stationary points are global minima. If *(EP)* is actually a variational inequality, i.e. $f(x,y) = \langle F(x), y - x\rangle$ for some $F : \mathbb{R}^n \to \mathbb{R}^n$, then (11) is equivalent to the monotonicity of F.

Example 3.1. Consider *(EP)* with $n = 2$, $f(x,y) = x_1 - y_1 + x_2 - y_2$ and the ball $B(0,1)$ of center 0 and unitary radius as the feasible region C. It is easy to check that $x^* = (\sqrt{2}/2, \sqrt{2}/2)$ is the unique solution of *(EP)*. Notice that $\nabla_x f(x,\cdot)$ is monotone but not strictly monotone since $\nabla_x f(x,y) = (1,1)$ for any $x, y \in \mathbb{R}^2$.

Considering $h(x,y) = \|y - x\|_2^2/2$, the gap function (1) reads

$$\varphi_\sigma(x) = \max\{y_1 + y_2 - \sigma[(y_1 - x_1)^2 + (y_2 - x_2)^2)]/2 \ : \ y \in C\} - x_1 - x_2.$$

Since $\hat{y}_\sigma(x) = (x_1+1/\sigma, x_2+1/\sigma)$ maximizes the objective function over the whole $\mathbb{R}^2$, $y_\sigma(x) = \hat{y}_\sigma(x)$ and therefore $\varphi_\sigma(x) = 1/\sigma$ hold if $\hat{y}_\sigma(x)$ is feasible, i.e., if $x \in B(z_\sigma, 1)$ for $z_\sigma = (-1/\sigma, -1/\sigma)$. Consequently, $\varphi_{\alpha,\beta}(x) = 1/\alpha - 1/\beta$ holds whenever $x \in B(z_\alpha, 1) \cap B(z_\beta, 1)$ for any $0 < \alpha < \beta$. Taking any $\alpha \geq \sqrt{2}$ and $\beta > \alpha$ or any $\alpha < \sqrt{2}$ and $\alpha < \beta < \sqrt{2}\alpha/(\sqrt{2} - \alpha)$, this intersection is not empty and has also a nonempty interior. Therefore, any point x in the interior of $B(z_\alpha, 1) \cap B(z_\beta, 1)$ is stationary for $\varphi_{\alpha,\beta}$ though it does not solve *(EP)*.

Figure 1 shows the graph of $\varphi_{\alpha,\beta}$ for $\alpha = \sqrt{2}$ and $\beta = 2$.

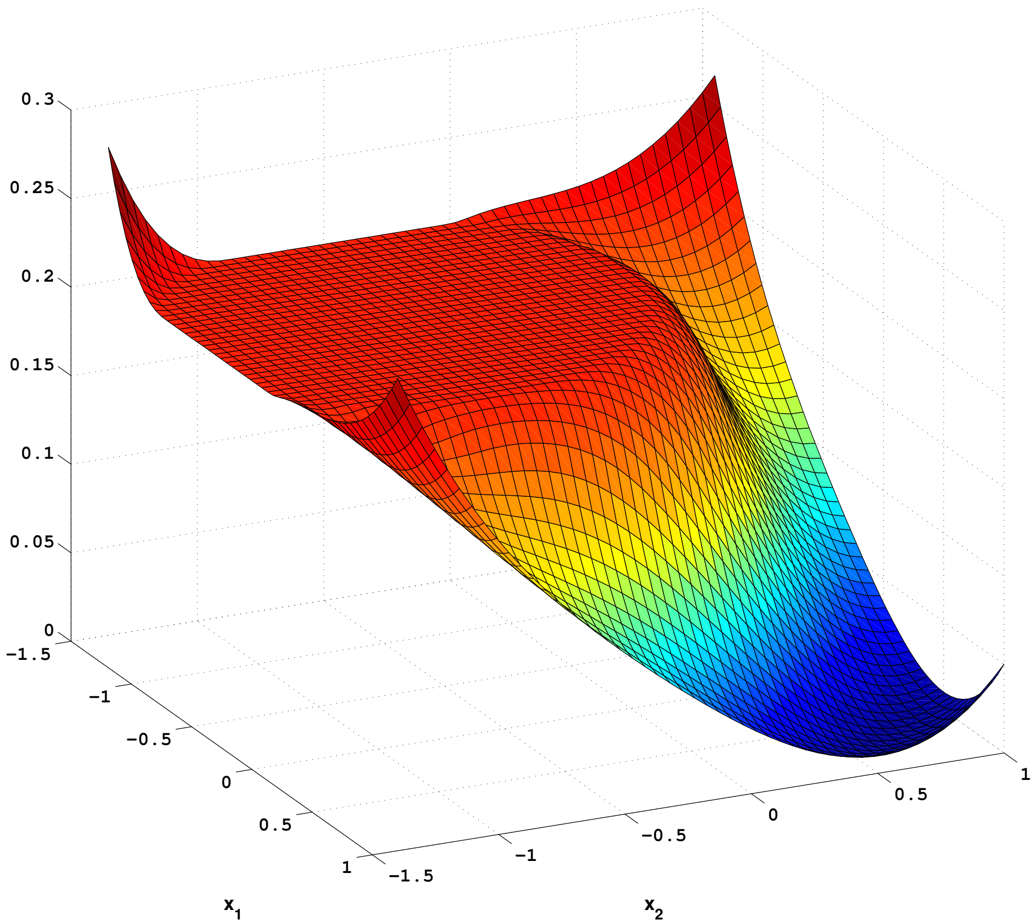

Figure 1: The D-gap function $\varphi_{\alpha,\beta}$ with $\alpha = \sqrt{2}$ and $\beta = 2$ in Example 3.1.

The following theorem provide the key tool for devising a descent method which does not get trapped into stationary points not solving *(EP)*.

Theorem 3.1. Suppose (11) holds. Then,

(a) the inequalities

$$\begin{aligned}&\langle \nabla \varphi_{\alpha,\beta}(x), y_\alpha(x) - y_\beta(x)\rangle \leq \\ &\leq \langle \beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle \leq 0\end{aligned} \tag{12}$$

hold for any $x \in \mathbb{R}^n$ and $0 < \alpha < \beta$.

(b) If $x \in C$ does not solve (EP) and a minimum point $y_0(x)$ for $f(x, \cdot)$ over C exists, then there are $\bar{\alpha} > 0$ and $\bar{\beta} > \bar{\alpha}$ such that $y_\alpha(x) - y_\beta(x)$ is a descent direction for $\varphi_{\alpha,\beta}$ at x for all $\alpha \in (0, \bar{\alpha})$ and $\beta > \bar{\beta}$.

Proof. (a) Condition (11) implies that

$$\begin{aligned}&\langle \nabla \varphi_{\alpha,\beta}(x), y_\alpha(x) - y_\beta(x)\rangle = \\ &\quad = \langle \nabla_x f(x, y_\beta(x)) - \nabla_x f(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle + \\ &\quad\quad + \langle \beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle \\ &\quad \leq \langle \beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle.\end{aligned}$$

The optimality conditions satisfied by $y_\alpha(x)$ and $y_\beta(x)$ guarantee

$$\begin{aligned}&\langle \nabla_y f(x, y_\alpha(x)) + \alpha \nabla_y h(x, y_\alpha(x)), y_\beta(x) - y_\alpha(x)\rangle \geq 0, \\ &\langle \nabla_y f(x, y_\beta(x)) + \beta \nabla_y h(x, y_\beta(x)), y_\alpha(x) - y_\beta(x)\rangle \geq 0.\end{aligned} \tag{13}$$

Since partial derivatives of h are related to each other and $f(x, \cdot)$ is convex, (13) guarantees

$$\begin{aligned}&\langle \beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle = \\ &= \langle \alpha \nabla_y h(x, y_\alpha(x)) - \beta \nabla_y h(x, y_\beta(x)), y_\alpha(x) - y_\beta(x)\rangle \\ &\leq \langle \nabla_y f(x, y_\beta(x)) - \nabla_y f(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle \leq 0.\end{aligned}$$

(b) We have

$$\begin{aligned}&\langle \beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle = \\ &= f(x, y_\alpha(x)) - f(x, y_\beta(x)) - \alpha \langle \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle \\ &\quad + f(x, y_\beta(x)) - f(x, y_\alpha(x)) + \beta \langle \nabla_x h(x, y_\beta(x)), y_\alpha(x) - y_\beta(x)\rangle.\end{aligned} \tag{14}$$

The convexity of $f(x, \cdot)$, the relationships between partial derivatives of h and the optimality condition satisfied by $y_\beta(x)$ guarantee

$$\begin{aligned}&f(x, y_\alpha(x)) - f(x, y_\beta(x)) + \beta \langle \nabla_x h(x, y_\beta(x)), y_\beta(x) - y_\alpha(x)\rangle \geq \\ &\geq \langle \nabla_y f(x, y_\beta(x)), y_\alpha(x) - y_\beta(x)\rangle + \beta \langle \nabla_x h(x, y_\beta(x)), y_\beta(x) - y_\alpha(x)\rangle \\ &= \langle \nabla_y f(x, y_\beta(x)) + \beta \nabla_y h(x, y_\beta(x)), y_\alpha(x) - y_\beta(x)\rangle \geq 0.\end{aligned} \tag{15}$$

Therefore (14) and (15) imply

$$\begin{aligned}&\langle \beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle \leq \\ &\leq f(x, y_\alpha(x)) - f(x, y_\beta(x)) - \alpha \langle \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x)\rangle \\ &= -\varphi_\alpha(x) - f(x, y_\beta(x)) + \\ &\quad + \alpha \left[\langle \nabla_x h(x, y_\alpha(x)), y_\beta(x) - y_\alpha(x)\rangle - h(x, y_\alpha(x))\right].\end{aligned} \tag{16}$$

Since $x \in C$ does not solve *(EP)*, for any $\alpha \leq 1$ we have

$$\varphi_\alpha(x) \geq \varphi_1(x) > 0. \tag{17}$$

Since $x \in C$, then (8) and the continuity of f guarantee

$$\lim_{\beta \to +\infty} f(x, y_\beta(x)) = f(x, y_\infty(x)) = f(x, x) = 0. \tag{18}$$

The function $f(x, \cdot) + \alpha h(x, \cdot)$ is strongly convex, thus

$$\begin{aligned} f(x, y_0(x)) + \alpha h(x, y_0(x)) &\geq f(x, y_\alpha(x)) + \alpha h(x, y_\alpha(x)) + \\ + \langle \nabla_y f(x, y_\alpha(x)) + \alpha \nabla_y h(x, y_\alpha(x)), y_0(x) - y_\alpha(x) \rangle &+ \alpha\tau ||y_0(x) - y_\alpha(x)||^2 \geq \\ &\geq f(x, y_\alpha(x)) + \alpha\tau ||y_0(x) - y_\alpha(x)||^2, \end{aligned}$$

where the last inequality is due to the positiveness of h and the optimality condition (2) with $\sigma = \alpha$ and $z = y_0(x)$. Since $f(x, y_0(x)) \leq f(x, y_\alpha(x))$ holds by the choice of $y_0(x)$, then the inequality

$$||y_0(x) - y_\alpha(x)||^2 \leq \tau^{-1} h(x, y_0(x))$$

follows. Hence the sequence $\{y_\alpha(x)\}$ is bounded as $\alpha \to 0$, and moreover $y_\beta(x) \to x$ as $\beta \to +\infty$ by Lemma 2.2. Thus, the continuous differentiability of h guarantees

$$\lim_{\substack{\alpha \to 0 \\ \beta \to +\infty}} \alpha \left[\langle \nabla_x h(x, y_\alpha(x)), y_\beta(x) - y_\alpha(x) \rangle - h(x, y_\alpha(x)) \right] = 0. \tag{19}$$

Thanks to (16), (17), (18) and (19) we get that there exist $\bar{\alpha} > 0$ and $\bar{\beta} > \bar{\alpha}$ such that for all $\alpha \in (0, \bar{\alpha})$ and $\beta > \bar{\beta}$ we have

$$\langle \beta \nabla_x h(x, y_\beta(x)) - \alpha \nabla_x h(x, y_\alpha(x)), y_\alpha(x) - y_\beta(x) \rangle < 0, \tag{20}$$

hence $y_\alpha(x) - y_\beta(x)$ is a descent direction by inequality (12). □

When *(EP)* is a variational inequality, condition (12) with $h(x, y) = ||y - x||_2^2$ collapses to the one exploited in [20] (see equation (15) therein) while the right inequality in (12) reduces to Lemma 3.2 in [22].

Theorem 3.1(a) guarantees that the directional derivative of $\varphi_{\alpha,\beta}$ at x along the direction $y_\alpha(x) - y_\beta(x)$ is not positive, but this is not enough for achieving descent along the direction. Indeed, this is not necessarily the case even when the gradient maps are strictly monotone since $y_\alpha(x) = y_\beta(x)$ may still occur. In fact, different directions have been exploited in [8, 13, 23, 24]. Anyway, according to Theorem 3.1(b), the search direction $y_\alpha(x) - y_\beta(x)$ is indeed a descent direction (therefore $y_\alpha(x) \neq y_\beta(x)$ must hold too) if x is feasible and provided that α and β are chosen, respectively, small and large enough. Notice that the assumption that a minimizer for $f(x, \cdot)$ exists is always satisfied whenever C is bounded or $f(x, \cdot)$ is coercive on C.

The above results provide the basic idea for a solution method: given α and β, the D-gap function $\varphi_{\alpha,\beta}$ is exploited until the search direction is no longer recognized as a descent direction, in which case a null step is performed while the parameters α and β are updated. An analogous idea was already exploited for gap functions in [1], but a substantial difference holds: the search direction $y_\alpha(x) - y_\beta(x)$ might be unfeasible, that is no stepsize might provide a feasible point moving away from the current iterate along the search direction. Since all the global minima of the D-gap functions $\varphi_{\alpha,\beta}$ are feasible, this is not a serious drawback: the search direction is exploited as long as a sufficient decrease condition (see (22) below) is satisfied even if unfeasible iterates are generated; when a sufficient decrease is no longer achieved, the current iterate is somehow replaced by a feasible point and the parameters α and β updated in such a way that the required decrease is lowered (see (21) below).

Algorithm

Step 0. Fix $\gamma, \eta \in (0,1)$, $\delta \in (0,\eta)$. Let $\{\alpha_k\}$ and $\{\varepsilon_k\}$ be two decreasing sequences going to zero, choose any $x^0 \in C$, $\beta_0 > \alpha_0$ and set $k = 1$.

Step 1. If $x^{k-1} \in C$ then set $z^0 = x^{k-1}$; else choose any $z^0 \in C$. Set $j = 0$. Choose $\beta_k \geq \beta_0$ such that

$$\varphi_{\alpha_k,\beta_k}(z^0)/(\beta_k - \alpha_k) \leq \varepsilon_k. \tag{21}$$

Step 2. Compute

$$y^j_{\alpha_k} = \arg\min\{\ f(z^j, y) + \alpha_k\, h(z^j, y)\ :\ y \in C\ \},$$
$$y^j_{\beta_k} = \arg\min\{\ f(z^j, y) + \beta_k\, h(z^j, y)\ :\ y \in C\ \}.$$

If $y^j_{\alpha_k} = z^j$ then STOP, else set $d^j = y^j_{\alpha_k} - y^j_{\beta_k}$.

Step 3. If

$$\langle \beta_k \nabla_x h(z^j, y^j_{\beta_k}) - \alpha_k \nabla_x h(z^j, y^j_{\alpha_k}), d^j \rangle \leq -\eta\, \varphi_{\alpha_k,\beta_k}(z^j)/(\beta_k - \alpha_k), \tag{22}$$

then compute the smallest $s \in \mathbb{N}$ such that

$$\varphi_{\alpha_k,\beta_k}(z^j + \gamma^s\, d^j) - \varphi_{\alpha_k,\beta_k}(z^j) \leq -\delta\, \gamma^s\, \varphi_{\alpha_k,\beta_k}(z^j)/(\beta_k - \alpha_k),$$

set $t_j = \gamma^s$, $z^{j+1} = z^j + t_j\, d^j$, $j = j + 1$ and goto Step 2

else set $x^k = z^j$, $k = k + 1$ and goto Step 1.

If the algorithm performs an infinite sequence of null steps, i.e., $k \to +\infty$, then α_k necessarily goes to 0 while β_k is not forced to go to infinity. Convergence to a solution of *(EP)* is achieved considering separately the case in which α_k actually goes to 0 from the case in which the parameters are updated a finite number of times.

Theorem 3.2. If f satisfies (11) and C is bounded, then either the algorithm stops at a solution of (EP) after a finite number of iterations, or it produces either a bounded sequence $\{x^k\}$ or a bounded sequence $\{z^j\}$ such that any of its cluster points solves (EP).

Proof. Lemma 2.2 guarantees that given any $z^0 \in C$ there exists a sufficiently large β_k such that (21) holds so that Step 1 is well-defined.

The line search procedure at step 3 is always finite. In fact, suppose by contradiction that there exist k and j such that

$$\varphi_{\alpha_k,\beta_k}(z^j + \gamma^s\, d^j) - \varphi_{\alpha_k,\beta_k}(z^j) > -\delta\, \gamma^s\, \varphi_{\alpha_k,\beta_k}(z^j)/(\beta_k - \alpha_k)$$

holds for all $s \in \mathbb{N}$. Taking the limit, we have

$$\langle \nabla \varphi_{\alpha_k,\beta_k}(z^j), d^j \rangle \geq -\delta\, \varphi_{\alpha_k,\beta_k}(z^j)/(\beta_k - \alpha_k).$$

On the other hand, Theorem 3.1 and condition (22) imply

$$\langle \nabla \varphi_{\alpha_k,\beta_k}(z^j), d^j \rangle \leq -\eta\, \varphi_{\alpha_k,\beta_k}(z^j)/(\beta_k - \alpha_k),$$

and thus

$$(\delta - \eta)\, \varphi_{\alpha_k,\beta_k}(z^j)/(\beta_k - \alpha_k) \geq 0,$$

which is not possible since $\delta < \eta$ and $\varphi_{\alpha_k,\beta_k}(z^j) > 0$.

If the algorithm stops at some z^j after a finite number of iterations, then the stopping criterion guarantees that z^j solves *(EP)* since it is a fixed point of the mapping y_{α_k}.

Now, suppose the algorithm produces an infinite sequence $\{z^j\}$ for some fixed k. Therefore, we can set $\alpha = \alpha_k$ and $\beta = \beta_k$ as these values don't change anymore. Since the sequence $\{\varphi_{\alpha,\beta}(z^j)\}$ is decreasing and the sublevel sets of $\varphi_{\alpha,\beta}$ are bounded (see [23]), then the sequence $\{z^j\}$ is bounded. Let z^* be any of its cluster points: taking the appropriate subsequence $\{z^{j_\ell}\}$, we have $z^{j_\ell} \to z^*$. By the continuity of the mappings y_α and y_β, $z^{j_\ell} \to z^*$ implies also $d^{j_\ell} \to d^* := y_\alpha(z^*) - y_\beta(z^*)$.

By contradiction, suppose that z^* does not solve *(EP)*, or equivalently $\varphi_{\alpha,\beta}(z^*) > 0$. By the step size rule we have

$$\varphi_{\alpha,\beta}(z^{j_\ell}) - \varphi_{\alpha,\beta}(z^{j_{\ell+1}}) \geq \delta\, t_{j_\ell}\, \varphi_{\alpha,\beta}(z^{j_\ell})/(\beta-\alpha) \geq 0.$$

Since $\{\varphi_{\alpha,\beta}(z^{j_\ell})\}$ is decreasing and bounded below by zero, we have

$$\lim_{\ell\to\infty} [\varphi_{\alpha,\beta}(z^{j_\ell}) - \varphi_{\alpha,\beta}(z^{j_{\ell+1}})] = 0,$$

and thus we get $\lim_{\ell\to\infty} t_{j_\ell} = 0$ since $\varphi_{\alpha,\beta}$ is continuous and $z^{j_\ell} \to z^*$.

Moreover, we have

$$\varphi_{\alpha,\beta}\left(z^{j_\ell} + t_{j_\ell}\,\gamma^{-1}\, d^{j_\ell}\right) - \varphi_{\alpha,\beta}(z^{j_\ell}) > -\delta\, t_{j_\ell}\,\gamma^{-1}\,\varphi_{\alpha,\beta}(z^{j_\ell})/(\beta-\alpha), \qquad \forall\, \ell \in \mathbb{N}.$$

The mean value theorem guarantees

$$\varphi_{\alpha,\beta}\left(z^{j_\ell} + t_{j_\ell}\,\gamma^{-1}\, d^{j_\ell}\right) - \varphi_{\alpha,\beta}(z^{j_\ell}) = \langle \nabla\varphi_{\alpha,\beta}(z^{j_\ell} + \theta_\ell\, t_{j_\ell}\,\gamma^{-1}\, d^{j_\ell}), t_{j_\ell}\,\gamma^{-1}\, d^{j_\ell}\rangle,$$

for some $\theta_\ell \in (0,1)$. Therefore, we have

$$\langle \nabla\varphi_{\alpha,\beta}(z^{j_\ell} + \theta_\ell\, t_{j_\ell}\,\gamma^{-1}\, d^{j_\ell}), d^{j_\ell}\rangle > -\delta\,\varphi_{\alpha,\beta}(z^{j_\ell})/(\beta-\alpha).$$

Since $\{d^{j_\ell}\}$ is bounded, taking the limit we get

$$\langle \nabla\varphi_{\alpha,\beta}(z^*), d^*\rangle \geq -\delta\,\varphi_{\alpha,\beta}(z^*)/(\beta-\alpha).$$

Theorem 3.1 and condition (22) imply

$$\langle \nabla\varphi_{\alpha,\beta}(z^{j_\ell}), d^{j_\ell}\rangle \leq -\eta\,\varphi_{\alpha,\beta}(z^{j_\ell})/(\beta-\alpha),$$

and thus

$$\langle \nabla\varphi_{\alpha,\beta}(z^*), d^*\rangle \leq -\eta\,\varphi_{\alpha,\beta}(z^*)/(\beta-\alpha)$$

follows just taking the limit. Hence, we get

$$(\delta-\eta)\,\varphi_{\alpha,\beta}(z^*)/(\beta-\alpha) \geq 0,$$

which is not possible since $\delta < \eta$ and $\varphi_{\alpha,\beta}(z^*) > 0$. Therefore, z^* solves *(EP)*.

Now, suppose that the algorithm produces an infinite sequence $\{x^k\}$. Since

$$0 \leq \varphi_{\alpha_k,\beta_k}(x^k)/(\beta_k-\alpha_k) \leq \varphi_{\alpha_k,\beta_k}(z^0)/(\beta_k-\alpha_k) \leq \varepsilon_k,$$

we have

$$\lim_{k\to\infty} \varphi_{\alpha_k,\beta_k}(x^k)/(\beta_k-\alpha_k) = 0. \tag{23}$$

Moreover, condition (22) is not satisfied at x^k, which reads

$$\begin{aligned} 0 \quad &\leq \langle \beta_k \nabla_x h(x^k, y_{\beta_k}(x^k)) - \alpha_k \nabla_x h(x^k, y_{\alpha_k}(x^k)), y_{\beta_k}(x^k) - y_{\alpha_k}(x^k)\rangle \\ &< \eta\,\varphi_{\alpha_k,\beta_k}(x^k)/(\beta_k-\alpha_k), \end{aligned}$$

where the left inequality is provided by (12). Thus, (23) implies

$$\lim_{k\to\infty} \langle \beta_k \nabla_x h(x^k, y_{\beta_k}(x^k)) - \alpha_k \nabla_x h(x^k, y_{\alpha_k}(x^k)), y_{\beta_k}(x^k) - y_{\alpha_k}(x^k)\rangle = 0. \tag{24}$$

The lower bound (5) implies

$$0 \leq h(x^k, y_{\beta_k}(x^k)) \leq \varphi_{\alpha_k,\beta_k}(x^k)/(\beta_k-\alpha_k),$$

and thus

$$\lim_{k\to\infty} h(x^k, y_{\beta_k}(x^k)) = 0$$

follows from (23). Since $h(x,\cdot)$ is strongly convex, we get

$$\begin{aligned} h(x^k, y_{\beta_k}(x^k)) &\geq h(x^k, x^k) + \langle \nabla_y h(x^k, x^k), y_{\beta_k}(x^k) - x^k \rangle + \tau \, \|y_{\beta_k}(x^k) - x^k\|^2 \\ &= \tau \, \|y_{\beta_k}(x^k) - x^k\|^2, \end{aligned}$$

and thus

$$\lim_{k\to\infty} \|y_{\beta_k}(x^k) - x^k\| = 0.$$

Since C is bounded, then also the sequence $\{x^k\}$ is bounded. Let x^* be any of its cluster points: taking an appropriate subsequence $\{x^{k_\ell}\}$, we have $x^{k_\ell} \to x^*$ and

$$\lim_{\ell\to\infty} y_{\beta_{k_\ell}}(x^{k_\ell}) = x^*.$$

Since $y_{\beta_{k_\ell}}(x^{k_\ell}) \in C$ for all $\ell \in \mathbb{N}$, we also have $x^* \in C$. On the other hand, we also have

$$\begin{aligned} &-f(x^{k_\ell}, y) - \alpha_{k_\ell} h(x^{k_\ell}, y) \leq \varphi_{\alpha_{k_\ell}}(x^{k_\ell}) \leq \\ &\leq \langle \beta_{k_\ell} \nabla_x h(x^{k_\ell}, y_{\beta_{k_\ell}}(x^{k_\ell})) - \alpha_{k_\ell} \nabla_x h(x^{k_\ell}, y_{\alpha_{k_\ell}}(x^{k_\ell})), y_{\beta_{k_\ell}}(x^{k_\ell}) - y_{\alpha_{k_\ell}}(x^{k_\ell}) \rangle + \\ &\quad -f(x^{k_\ell}, y_{\beta_{k_\ell}}(x^{k_\ell})) + \\ &\quad +\alpha_{k_\ell} \left[\langle \nabla_x h(x^{k_\ell}, y_{\alpha_{k_\ell}}(x^{k_\ell})), y_{\beta_{k_\ell}}(x^{k_\ell}) - y_{\alpha_{k_\ell}}(x^{k_\ell}) \rangle - h(x^{k_\ell}, y_{\alpha_{k_\ell}}(x^{k_\ell})) \right], \end{aligned}$$

where the first inequality follows from the definition of φ_α while the second is actually (16). Taking the limit, thanks to (24) we get

$$-f(x^*, y) \leq 0 \qquad \forall\, y \in C,$$

i.e., x^* solves *(EP)*. □

Notice that convergence does not depend upon the way unfeasible iterates are replaced by feasible points during the null steps. A straightforward choice is to take one of the minimizers y_α or y_β computed at Step 2 during the last iteration. Another reasonable choice is to take the projection of the current iterate onto C, but it requires to solve a further optimization problem and it is therefore computationally expensive. Actually, it is also possible to not replace an unfeasible x^{k-1} by some feasible point and therefore set $z^0 = x^{k-1}$ all the same if the inequality

$$\varphi_{\alpha_k, \beta_{k-1}}(x^{k-1})/(\beta_{k-1} - \alpha_k) < \varepsilon_k$$

holds. In fact, Lemma 2.2 guarantees the existence of some β_k satisfying (21) also in this case.

In order to slow down the decrease of α_k towards 0, it is possible to keep it unchanged at a null step if the current D-gap function is still making enough progress towards 0, namely if $\varphi_{\alpha_{k-1}, \beta_{k-1}}(x^{k-1}) \leq \mu_{k-1}$ holds for some given sequence $\mu_k \downarrow 0$. If an infinite sequence of null steps is performed, either $\alpha_k \downarrow 0$ or $\alpha_k = \bar{\alpha}$ definitely for some $\bar{\alpha} > 0$ may occur. In the latter case convergence is guaranteed by (5): in fact, it guarantees both $\|y_{\beta_k}(x^k) - y_{\bar{\alpha}}(x^k)\| \to 0$ and $h(x^k, y_{\beta_k}(x^k)) \to 0$ so that any cluster point x^* of $\{x^k\}$ satisfies $y_{\bar{\alpha}}(x^*) = x^*$.

Furthermore, it is not necessary to fix the sequence $\{\varepsilon_k\}$ a priori before running the algorithm. Adaptive choices may be performed at each null step, for instance taking any ε_k such that

$$0 < \varepsilon_k \leq \sigma_k + \theta_k \varphi_{\alpha_{k-1}, \beta_{k-1}}(x^{k-1})/(\beta_{k-1} - \alpha_{k-1}) \tag{25}$$

where $\sigma_k \downarrow 0$ and $0 < \theta_k < \theta < 1$ for some given θ. Indeed, if an infinite sequence of null steps is performed, then the required condition $\varepsilon_k \to 0$ holds also in this case.

3.1 Convergence in the unbounded case

The boundedness of C is a key tool to achieve the convergence of the algorithm. Indeed, it is exploited to guarantee that the sequences $\{x^k\}$, $\{y_{\alpha_k}(x^k)\}$ and $\{y_{\beta_k}(x^k)\}$ as well as the sublevel sets

$$\{x \in \mathbb{R}^n \ : \ \varphi_{\alpha,\beta}(x) \leq \epsilon\} \tag{26}$$

are bounded. Actually, even if C is unbounded, the boundedness of the sublevel sets (26) alone is enough to achieve convergence, provided that the algorithm behaves in a such a way that $\{\beta_k\}$ is not bound to go to infinity.

Theorem 3.3. Suppose f satisfies (11) and the sublevel set (26) is bounded for any $\epsilon \geq 0$ and $0 < \alpha < \beta$. If the algorithm generates an infinite sequence $\{\beta_k\}$ bounded above, then the algorithm produces either a bounded sequence $\{x^k\}$ or a bounded sequence $\{z^j\}$ such that any of its cluster points solves (EP).

Proof. If the algorithm produces an infinite sequence $\{z^j\}$ for some fixed k, then we can set $\alpha = \alpha_k$ and $\beta = \beta_k$ as these values don't change anymore. The sequence $\{z^j\}$ is bounded since the sequence $\{\varphi_{\alpha,\beta}(z^j)\}$ is decreasing and the sublevel sets of $\varphi_{\alpha,\beta}$ are bounded. Therefore, the thesis follows just arguing as in the proof of Theorem 3.2.

If the algorithm produces an infinite sequence $\{x^k\}$, then the same arguments of the proof of Theorem 3.2 show that (23) holds. Furthermore, any $k \in \mathbb{N}$ satisfies $0 < \alpha_k \leq \alpha_0$ and $\beta_0 \leq \beta_k \leq \bar{\beta}$ for some $\bar{\beta} > 0$, hence the inequalities

$$0 \leq \varphi_{\alpha_0,\beta_0}(x^k)/\bar{\beta} \leq \varphi_{\alpha_k,\beta_k}(x^k)/\bar{\beta} \leq \varphi_{\alpha_k,\beta_k}(x^k)/(\beta_k - \alpha_k)$$

hold. Therefore, (21) implies $\varphi_{\alpha_0,\beta_0}(x^k) \to 0$. Since the sublevel sets of $\varphi_{\alpha_0,\beta_0}$ are bounded, the sequence $\{x^k\}$ is bounded. As a consequence, any cluster point x^* of $\{x^k\}$ satisfies $\varphi_{\alpha_0,\beta_0}(x^*) = 0$, i.e., x^* solves *(EP)*. □

The boundedness of the sublevel sets (26) is guaranteed if $\nabla_y f$ is Lipschitz continuous and $G(x) = \nabla_y f(x,x)$ is strongly monotone [8, Corollary 3.4] or if the mappings $\nabla_y f(\cdot, y)$ are strongly monotone uniformly in $y \in C$ [24, Theorem 4.1]. Instead of strong monotonicity, some kind of coercivity could be exploited. Indeed, a further result can be achieved relying on the condition

$$\exists\, y \in C \ \text{ s.t. } \lim_{\|x\| \to +\infty} f(x,y)/\|x\| = -\infty, \tag{27}$$

which implies the well-known coercivity condition [7]:

$$\exists\, r > 0,\ \exists\, y \in C \ \text{ with } \ \|y\| \leq r \ \text{ s.t. } f(x,y) < 0,\ \forall\, x \in C \ \text{ with } \ \|x\| > r.$$

As a consequence, it guarantees also the existence of solutions (see, for instance, [2]). Moreover, (27) holds whenever f is strongly monotone: in fact, there exists $\mu > 0$ such that the inequalities

$$\begin{aligned} f(x,y) &\leq -f(y,x) - \mu\|x-y\|^2 \\ &\leq -f(y,y) + \langle \nabla_y f(y,y), x-y \rangle - \mu\|x-y\|^2 \\ &\leq \|\nabla_y f(y,y)\|\|x-y\| - \mu\|x-y\|^2 \end{aligned}$$

hold for any $x, y \in \mathbb{R}^n$ (thanks to to strong monotonicity, the convexity of $f(x,\cdot)$ and the Cauchy-Schwarz inequality). Therefore, (27) follows immediately choosing any $y \in C$, since $\|x-y\|/\|x\| \to 1$ as $\|x\| \to +\infty$.

Condition (27) guarantees the boundedness of the sublevel sets (26) if paired with the condition

$$\limsup_{\|x\| \to +\infty} \|\nabla_y f(x,x)\|/\|x\| < +\infty, \tag{28}$$

which, roughly speaking, requires that the gradient of $f(x,\cdot)$ at x does not grow faster than x itself as $\|x\| \to +\infty$.

Proposition 3.1. If f satisfies (27) and (28), then the sublevel set (26) is bounded for any $\epsilon \geq 0$ and $0 < \alpha < \beta$.

Proof. Ab absurdo, suppose there exists a sequence $\{x^k\}$ such that $\|x^k\| \to \infty$ and $\varphi_{\alpha,\beta}(x^k) \leq \epsilon$ for some fixed $\epsilon \geq 0$ and $0 < \alpha < \beta$.

The uniform strong convexity of $h(x, \cdot)$ and inequality (5) provide the following inequalities

$$\tau\,(\beta - \alpha)\,\|x^k - y_\beta(x^k)\|^2 \leq (\beta - \alpha)\,h(x^k, y_\beta(x^k)) \leq \varphi_{\alpha,\beta}(x^k) \leq \epsilon,$$

which guarantee $\|x^k - y_\beta(x^k)\| \leq \sqrt{\epsilon/[\tau\,(\beta - \alpha)]} := M$. Hence, the bound $\|y_\beta(x^k)\| \leq M + \|x^k\|$ follows as well.

Let $y \in C$ be provided by (27). The following chain of inequalities holds

$$\begin{aligned}
f(x^k, y_\beta(x^k)) - f(x^k, y) &\leq \langle \nabla_y f(x^k, y_\beta(x^k)), y_\beta(x^k) - y\rangle \\
&\leq \beta\,\langle \nabla_y h(x^k, y_\beta(x^k)), y - y_\beta(x^k)\rangle \\
&\leq \beta\,\|\nabla_y h(x^k, y_\beta(x^k))\|\,\|y - y_\beta(x^k)\| \\
&\leq \beta\,L\,\|x^k - y_\beta(x^k))\|\,\|y - y_\beta(x^k)\| \\
&\leq \beta\,L\,M\,\|y - y_\beta(x^k)\| \\
&\leq \beta\,L\,M\,(\|y\| + M + \|x^k\|).
\end{aligned}$$

The first inequality is due to the convexity of $f(x, \cdot)$, the second follows from the optimally condition (2), the third is the Cauchy-Schwarz inequality, and the forth follows from the uniform Lipschitz continuity of the functions $\nabla_y h(x^k, \cdot)$ taking into account that $\nabla_y h(x^k, x^k) = 0$. Therefore, the inequality

$$f(x^k, y_\beta(x^k)) - f(x^k, y) \leq 2\beta\,L\,M\|x^k\| \tag{29}$$

holds whenever k is large enough (precisely, whenever $\|x^k\| \geq \|y\| + M$). Furthermore, the convexity of $f(x, \cdot)$ and the Cauchy-Schwarz inequality imply

$$\begin{aligned}
f(x^k, y_\beta(x^k)) &\geq \langle \nabla_y f(x^k, x^k), y_\beta(x^k) - x^k\rangle \\
&\geq -\|\nabla_y f(x^k, x^k)\|\,\|y_\beta(x^k) - x^k\| \\
&\geq -M\,\|\nabla_y f(x^k, x^k)\|.
\end{aligned}$$

Therefore, (28) guarantees $f(x^k, y_\beta(x^k)) \geq -\hat{M}\|x^k\|$ for some $\hat{M} > 0$ whenever k is large enough. Consequently, (27) implies

$$[f(x^k, y_\beta(x^k)) - f(x^k, y)]/\|x^k\| \geq -\hat{M} - f(x^k, y)/\|x^k\| \to +\infty,$$

contradicting (29). □

Notice that the so-called linear equilibrium problem, that is *(EP)* with

$$f(x, y) = \langle Px + Qy + r, y - x\rangle \tag{30}$$

for some $r \in \mathbb{R}^n$ and some $P, Q \in \mathbb{R}^{n\times n}$ where Q is positive semidefinite, fulfills the growth condition (28) since $\nabla_y f(x, x) = (P + Q)x + r$. Furthermore, it fulfills also the coercivity condition (27) if P is positive definite, since $f(\cdot, y)$ turns out to be quadratic and strongly concave.

Proposition 3.1 is neither weaker nor stronger than Theorem 4.1 of [24] and Corollary 3.4 of [8]. Though the uniform strong monotonicity of the mappings $\nabla_y f(\cdot, y)$ implies the strong monotonicity of f (see [5, Theorem 3.1 b)]) and hence condition (27), no other assumption is required by Theorem 4.1 unlike the above Proposition 3.1. The Lipschitz continuity of $\nabla_y f$ implies condition (28), but the strong monotonicity of $G(x) = \nabla_y f(x, x)$ and condition (27) are independent of each other (the former is stronger than the latter for variational inequalities, vice versa it is weaker for linear equilibrium problems).

4 Numerical results

Some preliminary tests have been run to analyse the sensitivity of the algorithm with respect to its parameters. Afterwards, another set of numerical tests has been run to compare it with other algorithms which exploit D-gap functions. The algorithms have been implemented in MATLAB 7.10.0. The built-in functions `fmincon` and `quadprog` from the Optimization Toolbox were exploited to evaluate the D-gap functions $\varphi_{\alpha,\beta}$ and to compute $y_\alpha(x)$ and $y_\beta(x)$, choosing the regularizing bifunction $h(x,y) = \|y-x\|_2^2/2$.

4.1 Preliminary tests

We tested the algorithm on some noncooperative games with quadratic payoffs. Each player i has a set of feasible strategies $K_i \subseteq \mathbb{R}^{n_i}$ and aims at maximizing an utility function which depends also on the strategies of the other players, namely $f_i : C \to \mathbb{R}$ with $C = K_1 \times \cdots \times K_N$ where N is the number of players. Finding a Nash equilibrium amounts to solving *(EP)* with the Nikaido-Isoda aggregate bifunction:

$$f(x,y) = \sum_{i=1}^{N} \left[f_i(x) - f_i(x(y_i)) \right],$$

where $x(y_i)$ denotes the vector obtained from x by replacing x_i with y_i (see, for instance, [2, 6]).

In our test we chose to consider 3 players, each of them controlling 2 variables ($n_i = 2$) in the following intersection of a box and a ball

$$K_i = [-5,5]^2 \cap B\left(0, 5(1+\sqrt{2})/2\right)$$

in order to maximize the following type of quadratic utility function

$$f_i(x) = \frac{1}{2}\langle x_i, A_{ii}x_i\rangle + \sum_{\substack{j=1\\ j\neq i}}^{N} \langle x_i, A_{ij}x_j\rangle + \langle b_i, x_i\rangle,$$

where the squared matrices $A_{11}, \dots, A_{NN}$ are symmetric and negative semidefinite while $A_{ij}^T = -A_{ji}$ for all $i \neq j$. In this setting, the key assumption (11) of the algorithm is satisfied. In fact, we have

$$\nabla_x f(x,y) = Dx - Sy + b,$$

where

$$D = \begin{pmatrix} A_{11} & 0 & \dots & 0 \\ 0 & A_{22} & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & \dots & A_{NN} \end{pmatrix}, \quad S = \begin{pmatrix} 0 & A_{21}^T & \dots & A_{N1}^T \\ -A_{21} & 0 & \dots & A_{N2}^T \\ \vdots & & \ddots & \vdots \\ -A_{N1} & -A_{N2} & \dots & 0 \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ \vdots \\ \vdots \\ b_N \end{pmatrix},$$

and therefore

$$\langle \nabla_x f(x,y) - \nabla_x f(x,z), y-z\rangle = -\langle y-z, S(y-z)\rangle = \langle y-z, S(y-z)\rangle = 0$$

holds for any y and any z since S is a skew-symmetric matrix. Thus, the mapping $\nabla_x f(x,\cdot)$ is monotone, but it is not strictly/strongly monotone and the algorithms from [8, 13, 24, 25] can not be exploited.

Instances have been produced relying on the generator of uniformly distributed pseudorandom numbers of MATLAB to choose the coefficients of the utility functions f_i and the starting point of the algorithm. In particular, $A_{ii} = -B_i\, B_i^T$ while A_{ij} with $i \neq j$ are taken from the matrix $(B - B^T)/2$, where $B_i \in \mathbb{R}^{2\times 2}$ and $B \in \mathbb{R}^{6\times 6}$ are matrices with pseudorandom elements drawn from the uniform distribution on $[0,1]$; similarly, the components of the vectors b_i are uniform pseudorandom values in the range $[0,5]$. Finally, the starting point is the Euclidean projection on the ball $B\left(0, 5(1+\sqrt{2})/2\right)$ of a vector whose components are uniform pseudorandom values in the range $[-5,5]$.

At step 1, we set $z^0 = y_{\alpha_k}$ for the y_{α_k} computed at previous iteration whenever x^{k-1} is not feasible, and we took

$$\beta_k = \min\{\beta'_i : \ \beta'_i \geq \beta_{k-1} \text{ and } \beta'_i \text{ satisfies (21)}\},$$

for a given increasing sequence $\{\beta'_i\}$ which goes to $+\infty$. The value 10^{-2} was used as the threshold for the stopping criterion at step 2, more precisely the algorithm stopped whenever $\|y_{\alpha_k} - z^j\|_\infty \leq 10^{-2}$. A preliminary set of tests on random instances suggested to set the parameters of the algorithm in following way: $\gamma = 0.4$, $\delta = 0.4$, $\eta = 0.9$, $\alpha_k = 1/3^k$, $\varepsilon_k = 1/3^k$, $\beta'_i = 99 + 3^i$.

Afterwards, computational tests have been carried out to show the behaviour of the algorithm with different values of the parameters. First, we ran the algorithm for different choices of the parameters γ, δ, η and different kinds of sequences $\{\beta'_i\}$ on a set of 100 random instances. Results with respect to different values of γ, δ and η are given in Tables 1 and 2: each row reports the average number of iterations, null steps, number of updates of β which have been performed and the average number of optimization problems which have been solved for each instance. The results suggest that the choice of these 3 parameters does not have a relevant impact on the performance of the algorithm.

Table 1: $\delta = 0.4$, $\eta = 0.9$, $\alpha_k = 1/3^k$, $\varepsilon_k = 1/3^k$, $\beta'_i = 99 + 3^i$

γ	iterations	null steps	β updates	opt. pbs
0.1	19.91	1.78	1.10	39.86
0.2	19.36	1.80	1.13	38.75
0.3	19.36	1.78	1.10	38.30
0.4	19.24	1.78	1.10	37.84
0.5	19.30	1.79	1.11	39.08
0.6	19.24	1.78	1.10	37.86
0.7	19.28	1.78	1.10	37.94
0.8	19.81	1.78	1.10	39.32
0.9	19.64	1.78	1.10	39.14

Table 2: $\gamma = 0.4$, $\alpha_k = 1/3^k$, $\varepsilon_k = 1/3^k$, $\beta'_i = 99 + 3^i$

δ	η	iterations	null steps	β updates	opt. pbs
0.2	0.3	20.61	1.90	1.22	40.82
0.2	0.5	20.19	1.90	1.22	39.52
0.2	0.7	19.95	1.90	1.22	39.05
0.2	0.9	19.81	1.90	1.22	38.87
0.4	0.5	20.16	1.90	1.22	40.19
0.4	0.7	19.94	1.90	1.22	39.28
0.4	0.9	19.79	1.90	1.22	38.94
0.6	0.7	19.97	1.90	1.22	39.76
0.6	0.9	19.78	1.90	1.22	38.98
0.8	0.9	19.80	1.90	1.22	39.42

Table 3 reports the performance of the algorithm when different sequences $\{\beta'_i\}$ are chosen. The results show that the exponential growth provides a better performance than the quadratic growth with

Table 3: $\gamma = 0.4$, $\delta = 0.4$, $\eta = 0.9$, $\alpha_k = 1/3^k$, $\varepsilon_k = 1/3^k$

β'_i	iterations	null steps	β updates	opt. pbs
$2 + i^2$	23.89	4.16	9.69	53.87
$10 + i^2$	24.40	4.34	8.97	54.09
$100 + i^2$	20.00	2.01	3.79	41.92
$1 + 3^i$	23.18	3.66	4.04	48.96
$9 + 3^i$	23.23	3.93	3.62	47.37
$99 + 3^i$	19.40	1.96	1.21	38.19

respect to all the considered indicators. Higher values of β'_0 produce better results both in the exponential and quadratic case.

To test the algorithm when null steps do occur, we ran it for different sequences $\{\alpha_k\}$ and $\{\varepsilon_k\}$ on a set of 100 random instances in which at least 1 null step is performed. Table 4 shows that $\{\varepsilon_k\}$ impacts on the performance of the algorithm more than $\{\alpha_k\}$ and that exponentially decreasing sequences seem to be the best choice for both parameters.

Table 4: $\gamma = 0.4$, $\delta = 0.4$, $\eta = 0.9$, $\beta'_i = 99 + 3^i$.

α_k	ε_k	iterations	null steps	β updates	opt. pbs
$1/(1 + k^2)$	$1/(1 + k^2)$	135.51	105.47	4.36	173.43
$1/3^k$	$1/(1 + k^2)$	135.64	105.42	4.33	171.28
$1/(1 + k^2)$	$1/3^k$	31.20	8.40	5.43	63.81
$1/3^k$	$1/3^k$	29.76	8.40	5.42	60.80

Finally, we tested the adaptive rule (25) for ε_k at step 1 taking precisely the upper bound, namely

$$\varepsilon_k = \sigma_k + \theta_k \varphi_{\alpha_{k-1},\beta_{k-1}}(x^{k-1})/(\beta_{k-1} - \alpha_{k-1}),$$

for different sequences $\{\sigma_k\}$ and $\{\theta_k\}$. Table 5 shows that the number of iterations and the number of optimization problems significantly decrease as the rate of convergence of $\{\sigma_k\}$ increases, and actually the best results are achieved for $\{\sigma_k\} \equiv 0$. The impact of $\{\theta_k\}$ seems to be less relevant, anyway notice that the non-adaptive rule ($\{\theta_k\} \equiv 0$) provides worse results than the best choices for the adaptive rule.

4.2 Comparison with other D-gap algorithms

Two other algorithms rely on D-gap functions [8, 13, 24, 25]. Actually, they are both based on the minimization of a single D-gap function $\varphi_{\alpha,\beta}$ for some fixed values of α and β. Another meaningful difference with the algorithm of this paper is that they exploit search directions other than $y_\alpha(x^k) - y_\beta(x^k)$.

The first algorithm (see [8, 13]) performs an inexact line search along the direction $y_\alpha(x^k) - y_\beta(x^k) + \rho s(x^k)$ for some suitable fixed $\rho > 0$, where the additional term $s(x^k) = \alpha[x^k - y_\alpha(x^k)] - \beta[x^k - y_\beta(x^k)]$ is needed to guarantee descent without changing α and β. Since it was the first method to be developed, it will be referred to as the "basic algorithm".

The other algorithm (see [24, 25]) tries to exploit the same direction $d^k = y_\alpha(x^k) - x^k$ which is used by the algorithms based on gap functions (see, for instance, [2]). If $x^k + d^k$ provides a large enough improvement of the value of the D-gap function, it is taken as the new iterate; otherwise, an inexact line

Table 5: $\gamma = 0.4$, $\delta = 0.4$, $\eta = 0.9$, $\alpha_k = 1/3^k$, $\beta'_i = 99 + 3^i$.

σ_k	θ_k	iterations	null steps	β updates	opt. pbs
$1/(1+k^2)$	0	135.77	105.56	4.31	171.56
$1/(1+k^2)$	0.5	182.81	150.64	4.28	218.84
$1/(1+k^2)$	$1/[1+(k+1)^2]$	136.14	105.92	4.32	171.95
$1/(1+k^2)$	$1/3^{k+1}$	135.77	105.56	4.31	171.56
$1/3^k$	0	29.99	8.41	5.41	61.42
$1/3^k$	0.5	32.81	9.18	4.85	64.22
$1/3^k$	$1/[1+(k+1)^2]$	30.09	8.44	5.44	61.62
$1/3^k$	$1/3^{k+1}$	29.99	8.41	5.41	61.42
0	0.5	18.06	1.00	6.00	41.12
0	$1/[1+(k+1)^2]$	18.06	1.00	7.00	42.12
0	$1/3^{k+1}$	18.06	1.00	7.09	42.21

search along either d^k or $-\nabla\varphi_{\alpha,\beta}(x^k)$ is performed. Since the algorithm combines together features of both the gap and D-gap function approaches, it will be referred to as the "hybrid algorithm".

In order to converge to a solution of *(EP)* both algorithms require the boundedness of the sublevel sets of $\varphi_{\alpha,\beta}$ (see also Section 3.1). In addition, the basic algorithm requires that the mappings $\nabla_x f(x,\cdot)$ are strongly monotone and Lipschitz continuous uniformly with respect to $x \in \mathbb{R}^n$, while the hybrid algorithm requires that the mappings $\nabla_x f(x,\cdot)$ are strictly monotone. As a consequence, the noncooperative games of the previous subsection can not be used as test problems to compare the three algorithms, since the mappings $\nabla_x f(x,\cdot)$ are monotone but neither strictly nor strongly monotone.

We tested the algorithms on the so-called linear equilibrium problems, that is *(EP)* with f given by (30) for some $r \in \mathbb{R}^n$ and some matrices $P, Q \in \mathbb{R}^{n\times n}$ such that Q is positive semidefinite. Asking for $P^T - Q$ to be positive definite guarantees the desired properties. In fact, the equality

$$\nabla_x f(x,y) - \nabla_x f(x,z) = (P^T - Q)(y-z)$$

guarantees that the mappings $\nabla_x f(x,\cdot)$ are uniformly strongly monotone and Lipschitz continuous, with the minimum eigenvalue of the symmetric part of $P^T - Q$ and $\|P^T - Q\|$ providing the corresponding moduli of uniformity μ and L.

Instances have been produced relying on the generator of uniformly distributed pseudorandom numbers of MATLAB to choose P, Q and r. In particular, $Q = A\,A^T$ and $P = Q + a\,B\,B^T + b\,I + c\,(S - S^T)$, where A, B and S are matrices with pseudorandom elements drawn from the uniform distribution on $[0,1]$ and the parameters a, b and c have been exploited to control μ and L. Finally, the components of r are uniform pseudorandom values in the range $[-1,1]$.

Test have been made considering $C = [-5,5]^n$ and taking a vector with uniform pseudorandom components in the range $[-5,5]$ as the starting point, while the same stopping criterion of the previous subsection has been exploited for all the three algorithms. A preliminary set of tests has been run to set the parameters. Afterwards, we ran each algorithm on a set of 1000 random instances for given values of μ and L. When an algorithm did not stop before solving 1000 optimization problems, we considered it a failure.

Tables 6 and 7 report the performances of the algorithms on instances with $n = 5$ and $n = 10$, respectively. Each row corresponds to a choice of μ and L and reports the percentage of failures and the average number of optimization problems required by a single instance. Both tables show that our algorithm performs better than the two others when the modulus of strong monotonicity of $\nabla_x f(x,\cdot)$ is close to zero, while it behaves at least comparably with them in the other situations.

Table 6: $n = 5$.

μ	L	our algorithm % fail	opt. pbs	basic algorithm % fail	opt. pbs	hybrid algorithm % fail	opt. pbs
0.001	0.01	0.4	78.86	90.1	579.39	40.9	80.97
0.001	0.05	0.4	77.32	4.0	258.00	36.1	78.46
0.001	0.1	0.3	77.18	0.0	103.71	30.3	78.87
0.01	0.1	0.3	73.80	1.6	342.00	17.5	72.73
0.01	0.5	0.0	62.21	0.0	58.49	2.8	58.28
0.01	1	0.0	56.81	0.0	53.78	1.1	52.77
0.1	0.2	0.0	45.57	2.2	240.51	0.0	43.41
0.1	0.5	0.0	41.77	0.0	64.74	0.0	39.73
0.1	1	0.0	37.79	0.0	36.13	0.0	35.67
0.3	0.5	0.0	23.93	0.0	70.14	0.0	22.06
0.3	1	0.0	22.55	0.0	61.63	0.0	20.65
0.3	1.5	0.0	21.73	0.0	20.92	0.0	19.71
0.5	0.6	0.0	16.77	0.0	44.31	0.0	14.93
0.5	1	0.0	16.36	0.0	32.42	0.0	14.42
0.5	1.5	0.0	15.94	0.0	31.74	0.0	13.77

Table 7: $n = 10$.

μ	L	our algorithm % fail	opt. pbs	basic algorithm % fail	opt. pbs	hybrid algorithm % fail	opt. pbs
0.001	0.01	0.8	92.67	99.8	956.00	52.3	91.41
0.001	0.05	0.4	92.20	15.4	522.63	53.2	85.70
0.001	0.1	0.2	91.08	0.1	174.27	53.0	84.60
0.01	0.1	0.5	85.34	24.6	630.38	33.9	84.60
0.01	0.5	0.3	78.32	0.0	73.77	5.8	75.92
0.01	1	0.0	73.05	0.0	68.89	1.1	69.06
0.1	0.2	0.0	51.06	31.6	673.82	0.0	47.32
0.1	0.5	0.0	48.47	0.6	220.35	0.0	44.58
0.1	1	0.0	45.47	2.2	94.32	0.0	41.46
0.3	0.5	0.0	26.91	0.0	211.62	0.0	22.95
0.3	1	0.0	26.08	0.0	73.65	0.0	22.15
0.3	1.5	0.0	25.44	0.0	34.47	0.0	21.39
0.5	0.6	0.0	19.59	0.0	77.62	0.0	15.31
0.5	1	0.0	19.25	0.0	25.26	0.0	15.03
0.5	1.5	0.0	18.99	0.0	31.80	0.0	14.66

Acknowledgements. The authors wish to thank the two anonymous referees and the associate editor for the careful reviews and valuable comments, which helped them to improve the paper.

References

[1] Bigi, G., Castellani, M., Pappalardo, M.: A new solution method for equilibrium problems. Optim. Methods Softw. 24, 895–911 (2009)

[2] Bigi, G., Castellani, M., Pappalardo, M., Passacantando, M.: Existence and solution methods for equilibria. European J. Oper. Res. 227, 1–11 (2013)

[3] Bigi, G., Passacantando, M.: Gap functions and penalization for solving equilibrium problems with nonlinear constraints. Comput. Optim. Appl. 53, 323–346 (2012)

[4] Bigi, G., Passacantando, M.: Descent and penalization techniques for equilibrium problems with nonlinear constraints. J. Optim. Theory Appl., doi: 10.1007/s10957-013-0473-7.

[5] Bigi, G., Passacantando, M.: Twelve monotonicity conditions arising from algorithms for equilibrium problems. Optim. Methods Softw., doi: 10.1080/10556788.2014.900552.

[6] Blum, E., Oettli, W.: From optimization and variational inequalities to equilibrium problems. Math. Student 63, 123–145 (1994)

[7] Brézis, H., Nirenberg, L., Stampacchia, G.: A remark on Ky Fan's minimax principle. Boll. Un. Mat. Ital. 6, 293–300 (1972)

[8] Cherugondi, C.: A note on D-gap functions for equilibrium problems. Optimization 62, 211–226 (2013)

[9] Chadli, O., Konnov, I.V., Yao, J.C.: Descent methods for equilibrium problems in a Banach space. Comput. Math. Appl. 48, 609–616 (2004)

[10] Di Lorenzo, D., Passacantando, M., Sciandrone, M.: A convergent inexact solution method for equilibrium problems. Optim. Methods Softw. (2013) doi: 10.1080/10556788.2013.796376.

[11] Kanzow, C., Fukushima, M.: Theoretical and numerical investigation of the D-gap function for box constrained variational inequalities. Math. Program. 83, 55–87 (1998)

[12] Konnov, I.V., Ali, M.S.S.: Descent methods for monotone equilibrium problems in Banach spaces. J. Comput. Appl. Math. 188, 165–179 (2006)

[13] Konnov, I.V., Pinyagina, O.V.: D-gap functions for a class of equilibrium problems in Banach spaces. Comput. Methods Appl. Math. 3, 274–286 (2003)

[14] Konnov, I.V., Pinyagina, O.V.: A descent method with inexact linear search for nonsmooth equilibrium problems. Comput. Math. Math. Phys. 48, 1777–1783 (2008)

[15] Mastroeni, G.: Gap functions for equilibrium problems. J. Global Optim. 27, 411–426 (2003)

[16] Muu L.D., Oettli, W.: Convergence of an adaptive penalty scheme for finding constrained equilibria. Nonlinear Anal. 18, 1159-1166 (1992)

[17] Peng, J.-M.: Equivalence of variational inequality problems to unconstrained minimization. Math. Program. 78, 347–355 (1997)

[18] Peng, J.-M., Fukushima, M.: A hybrid Newton method for solving the variational inequality problem via the D-gap function. Math. Program. 86, 367–386 (1999)

[19] Peng, J.-M., Kanzow, C., Fukushima, M.: A hybrid Josephy-Newton method for solving box constrained variational inequality problems via the D-gap function. Optim. Methods Softw. 10, 687–71 (1999)

[20] Solodov, M.V., Tseng, P.: Some methods based on the D-gap function for solving monotone variational inequalities. Comput. Optim. Appl. 17, 255–277 (2000)

[21] Sun, D., Fukushima, M., Qi, L.: A computable generalized Hessian of the D-gap function and Newton-type methods for variational inequality problems. In: Ferris, M.C., Pang, J.-S. (eds.) Complementarity and variational problems, pp. 452–473. SIAM, Philadelphia (1997)

[22] Yamashita, N., Taji, K., Fukushima, M.: Unconstrained optimization reformulations of variational inequality problems. J. Optim. Theory Appl. 92, 439–456 (1997)

[23] Zhang, L., Han, J.Y.: Unconstrained optimization reformulations of equilibrium problems. Acta Math. Sin. (Engl. Ser.) 25, 343–354 (2009)

[24] Zhang, L., Wu, S.-Y.: An algorithm based on the generalized D-gap function for equilibrium problems. J. Comput. Appl. Math. 231, 403–411 (2009)

[25] Zhang, L., Wu, S.-Y., Fang, S.-C.: Convergence and error bound of a D-gap function based Newton-type algorithm for equilibrium problems. J. Ind. Manag. Optim. 6, 333–346 (2010)