ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/332327952

Properties of the sign gradient descent algorithms

Article *in* Information Sciences · April 2019

DOI: 10.1016/j.ins.2019.04.012

CITATIONS
15

READS
1,186

3 authors:

Emmanuel Moulay
Université de Poitiers
122 PUBLICATIONS 3,299 CITATIONS
SEE PROFILE

Vincent Léchappé
Institut National des Sciences Appliquées de Lyon
35 PUBLICATIONS 471 CITATIONS
SEE PROFILE

Franck Plestan
Ecole Centrale de Nantes
236 PUBLICATIONS 7,686 CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Quantum dynamics View project

Project Implicit sliding mode control View project

All content following this page was uploaded by Emmanuel Moulay on 25 February 2020.

The user has requested enhancement of the downloaded file.

HAL
archives-ouvertes.fr

Properties of the sign gradient descent algorithms

Emmanuel Moulay, Vincent Léchappé, Franck Plestan

▶ To cite this version:

Emmanuel Moulay, Vincent Léchappé, Franck Plestan. Properties of the sign gradient descent algorithms. Information Sciences, Elsevier, 2019, 10.1016/j.ins.2019.04.012 . hal-02096241

HAL Id: hal-02096241

https://hal.archives-ouvertes.fr/hal-02096241

Submitted on 21 Feb 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Properties of the sign gradient descent algorithms

Emmanuel Moulay[a], Vincent Léchappé[b], Franck Plestan[c]

[a]XLIM (UMR CNRS 7252), Université de Poitiers, 11 bd Marie et Pierre Curie, 86073 Poitiers Cedex 9, France
[b]Laboratoire Ampère (UMR CNRS 5005), INSA de Lyon, 20 Avenue Albert Einstein, 69100 Villeurbanne, France
[c]École Centrale de Nantes-LS2N, UMR CNRS 6004, 1 Rue de la Noë, 44321 Nantes Cedex 3, France

Abstract

The aim of this article is to study the properties of the sign gradient descent algorithms involving the sign of the gradient instead of the gradient itself and first introduced in the RPROP algorithm. This article provides two results of convergence for local optimization, a first one for nominal systems without uncertainty and a second one for systems with uncertainties. New sign gradient descent algorithms including the dichotomy algorithm DICHO are applied on several examples to show their effectiveness in terms of speed of convergence. As a novelty, the sign gradient descent algorithms can allow to converge in practice towards other minima than the closest minimum of the initial condition making these algorithms suitable for global optimization as a new metaheuristic method.

Keywords: Gradient descent, discrete-time systems, optimization, metaheuristic, Lyapunov sequence.

1. Introduction

Gradient descent is one of the powerful local optimization algorithms [12, 14]. It is a first-order method involving only the gradient and is used in many applications as optimal control [2], video coding [33], localization [19] or robotics [40]. A fast gradient method is developed by Nesterov in [35] and used for instance for the model predictive control [41]. Moreover, an optimized gradient method is proposed in [27] and a gradient evolution algorithm is stated in [28]. By using the sign of the gradient instead of the gradient itself, the RPROP algorithm for backpropagation in artificial neural networks first stated by Riedmiller and Braun in [42] provides a new gradient descent algorithm. It has then been developed and used by many authors [1, 24, 25]. The use of the sign of the gradient instead of the gradient itself avoids the vanishing gradient problem in training artificial neural networks with gradient-based learning methods [37]. Stochastic gradient descent is an iterative gradient descent optimization algorithm used for minimizing a cost function written as the sum of differentiable functions, see for instance [48, Section 5.1.2] and [47]. It is used for example in machine learning [45, Chapter 14], [47], deep learning [30] and localization [50].

Discontinuous differential equations have been developed by Filippov [16] and Clarke [13] and used in automatic control for sliding mode control introduced by Utkin in the 70's for solving Lyapunov stabilization problems [49]. Then, sliding mode control has been developed by many authors [15, 17]. This method uses a discontinuous controller in order to force a continuous uncertain system to reach, in finite time and in spite of uncertainties and perturbations, a manifold called sliding surface, that is defined from the control objectives. Several extensions of sliding mode control have been proposed as higher order sliding mode control [31, 38] or adaptive sliding mode control [39].

Discrete-time systems involving continuous functions have been widely studied, see for instance [21, 22]. Discrete-time systems involving discontinuous functions have been first developed in the framework of sliding

Email addresses: emmanuel.moulay@univ-poitiers.fr (Emmanuel Moulay), vincent.lechappe@insa-lyon.fr (Vincent Léchappé), franck.plestan@ec-nantes.fr (Franck Plestan)

mode control for discrete-time systems in [4, 5, 44]. Then, a general Lyapunov theory of stability for this class of systems has been performed in [20] and applied to the nonlinear model predictive control in [20, 29].

In this article, the properties of the first-order gradient descent algorithm using the sign function, whose a first version is the RPROP algorithm, is studied by using the theory of discrete-time systems involving discontinuous functions. The algorithm is called the *sign gradient descent algorithm* in this article as proposed in [7, 32]. A first convergence result for local optimization is stated by using the same strategy as the one used for gradient descent in [11, Section 4.2.2], i.e. a Lyapunov sequence. Moreover, a second result of local convergence robustness is proved in case of uncertain data. The hybrid gradient descent algorithm is introduced as an extension of the classical gradient descent algorithm having a new degree of freedom brought by the sign gradient descent algorithm. As a novelty, the sign gradient descent algorithm allows to converge in practice towards other minima than the closest minimum of the initial condition making these algorithms usable for global optimization as a new metaheuristic method and this is illustrated with the new dichotomy algorithm DICHO and the old RPROP algorithm.

The article is organized as follows. The sign gradient descent algorithms are recalled in Section 2; results on local convergence and robustness are provided. Then, several applications are given in Section 3 showing that the sign gradient descent algorithms can be faster than classical gradient descent and allows to converge towards other minima then the closest minimum of the initial condition. Finally, a conclusion is addressed in Section 4.

2. Sign gradient descent algorithms

First-of-all, some notations used in the sequel are introduced. Denoting $x = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$, the gradient of a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is the vector

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n} \right)^T .$$

A point $x_* \in \mathbb{R}^n$ is a critical point of f if $\nabla f(x_*) = 0$. Denote

$$\operatorname{sgn}(\nabla f(x)) = \left(\operatorname{sgn}\left(\frac{\partial f(x)}{\partial x_1} \right), \ldots, \operatorname{sgn}\left(\frac{\partial f(x)}{\partial x_n} \right) \right)^T$$

where sgn refers to the sign function defined by

$$\operatorname{sgn}(x) := \begin{cases} -1 & \text{if} \quad x < 0, \\ 0 & \text{if} \quad x = 0, \\ 1 & \text{if} \quad x > 0. \end{cases}$$

The euclidean norm is defined by $\|x\|^2 = x^T x$.

The gradient descent algorithm is a first-order local optimization method which intends to minimize a differentiable real function, *i.e.* it aims at solving the problem $\min_{x \in \mathbb{R}^n} f(x)$. Recall the definition of the gradient descent algorithm.

Definition 1. *Consider a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$. The gradient descent algorithm (GD) is defined by the following discrete-time system*

$$x_{k+1} = x_k - \gamma_k \nabla f(x_k), \qquad k \in \mathbb{N} \tag{1}$$

where $x_k \in \mathbb{R}^n$ is the state and $\gamma_k > 0$ the step size to be defined.

A critical point of f is an equilibrium of (1). There are several strategies for tuning the step size, see for instance [35, Section 1.2.3]. If the initial condition $x_0 \in \mathbb{R}^n$ is close to a local minimum and under additional assumptions on f, it is possible to prove the convergence of (1) towards the local minimum, see for instance

[35, Theorem 1.2.4] or [10, Subsection 2.3.2]. The GD is based on the fact that, if $\nabla f(x_k) \neq 0$, then the direction $d_k = -\gamma_k \nabla f(x_k)$ is a descent direction of f at x_k as

$$\langle \nabla f(x_k), d_k \rangle = -\gamma_k \, \|\nabla f(x_k)\|^2 < 0$$

or equivalently $f(x_k - \gamma_k \nabla f(x_k)) < f(x_k)$ for $\gamma_k > 0$ small enough. If f is not differentiable, it is possible to apply the subgradient method defined for instance in [8, Chapter 4].

Recall the sign gradient descent algorithm whose main interest lies in its simplicity and its speed of convergence with respect to the GD while being a first-order local optimization method.

Definition 2. *Consider a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$. The sign gradient descent algorithm (SGD) is defined by the following discrete-time system*

$$x_{k+1} = x_k - \gamma_k \, \operatorname{sgn}(\nabla f(x_k)), \qquad k \in \mathbb{N} \tag{2}$$

where $x_k \in \mathbb{R}^n$ is the state and $\gamma_k > 0$ the step size to be defined.

For tuning the step size γ_k, one can choose a sequence independent of $f(x_k)$ or dependent on $f(x_k)$ as proposed below in (10). Tuning the step size γ_k is crucial in practice. The adaptive sign gradient descent algorithm, first used in the RPROP algorithm [42], provides an automatic way for tuning the step size and also a strategy which can avoid local minima as shown in Subsection 3.2.

Definition 3. *Consider system* (2)*. If γ_k is defined by the discrete-time system*

$$\gamma_{k+1} = g(\gamma_k), \qquad k \in \mathbb{N} \tag{3}$$

with $g : \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$, then the algorithm (2)–(3) *is called adaptive sign gradient descent algorithm (ASGD).*

Considering $\nabla f(x_k) \neq 0$, and by using the descent direction $d_k = -\gamma_k \, \operatorname{sgn}(\nabla f(x_k))$, one has

$$\langle \nabla f(x_k), d_k \rangle = -\gamma_k \, |\nabla f(x_k)| < 0.$$

So, if $\gamma_{k+1} < \gamma_k$ for all $k \in \mathbb{N}$ and $\nabla f(x) = Cx$ is linear then the results in [26, 44] imply that the discrete-time system (2) converges towards a local minimum x_*. The speed of convergence of the SGD depends on the tuning of the step size γ_k. Special cases have been studied in [3], however this result is not enough general to be applied to the discrete-time system (2).

In the sequel, a convergence condition, first given for the GD in [11, Section 4.2.2], is stated for the SGD (2).

Theorem 1. *Suppose that f has a unique minimum x_* and satisfies $(x - x_*)^T \operatorname{sgn}(\nabla f(x)) > 0$ for all x in the domain of definition of f. Consider the SGD* (2) *and a sequence γ_k such that $\lim_{k \to +\infty} \gamma_k = 0$,*

$$0 < n\gamma_k < 2(x_k - x_*)^T \operatorname{sgn}(\nabla f(x_k)) \tag{4}$$

and

$$\gamma_k (x_k - x_*)^T \operatorname{sgn}(\nabla f(x_k)) \geqslant c\|x_k - x_*\|^\alpha, \qquad c > 0, \ \alpha > 0 \tag{5}$$

for all $k \in \mathbb{N}$. Then the sequence x_k given by the SGD (2) *satisfies $\lim_{k \to +\infty} x_k = x_*$.*

Proof. Consider the following Lyapunov sequence

$$V(x_k) = \|x_k - x_*\|^2.$$

By using (2) and (4), one gets

$$\begin{aligned} V(x_{k+1}) - V(x_k) &= \|x_{k+1} - x_*\|^2 - \|x_k - x_*\|^2 \\ &= \|x_k - \gamma_k \, \operatorname{sgn}(\nabla f(x_k)) - x_*\|^2 - \|x_k - x_*\|^2 \\ &= n\gamma_k^2 - 2\gamma_k (x_k - x_*)^T \operatorname{sgn}(\nabla f(x_k)) < 0. \end{aligned}$$

We deduce that $V(x_k)$ is decreasing. As $V(x_k)$ is decreasing and bounded from below by zero, the monotone convergence theorem given for instance in [6, Theorem 3.2] implies that $V(x_k)$ is convergent. As $\lim\limits_{k\to+\infty} \gamma_k = 0$ and $V(x_k)$ is convergent we deduce that

$$\lim_{k\to+\infty} \gamma_k (x_k - x_*)^T \operatorname{sgn}(\nabla f(x_k)) = \lim_{k\to+\infty} (x_k - x_*)^T (x_k - x_{k+1}) = 0.$$

By using (5), we conclude that $\lim\limits_{k\to+\infty} x_k = x_*$. □

Condition (4) implies that the sequence γ_k of the SGD (2) must be decreasing and the function g of the ASGD (3) must satisfy $g(x) < x$ for having the convergence. Condition (5) is required, because decreasing the step size too quickly could stop the convergence of the algorithm towards the minimum. However, Theorem 1 which is based on a Lyapunov sequence provides only a sufficient condition for the convergence of system (2). On the one hand, Conditions (4)–(5) are general conditions not too restrictive on f and its gradient, which is desirable. On the other hand, they involve the knowledge of the minimum x^* and can only be checked a posteriori in practice. On the contrary, Wolfe conditions, studied in [36, Subsection 3.1] for the GD algorithm and in [1] for the RPROP algorithm, are restrictive on f but they can be checked a priori in practice.

Consider now the case where x_k is only known with an uncertainty ϵ_k. Indeed, if the data are given by measurements then they may have uncertainties due to their experimental features. It leads to the following uncertain sign gradient descent algorithm (USGD)

$$x_{k+1} = x_k + \epsilon_k - \gamma_k \ \operatorname{sgn}(\nabla f(x_k + \epsilon_k)), \qquad k \in \mathbb{N}. \tag{6}$$

One gets the following result

Theorem 2. *Suppose that f has a unique minimum x_* and for all x in the domain of definition of f there exists $\epsilon \in \mathbb{R}^n$ such that $(x + \epsilon - x_*)^T \operatorname{sgn}(\nabla f(x + \epsilon)) > 0$. Consider the USGD (6) and sequences $\gamma_k > 0$ and $\epsilon_k \in \mathbb{R}^n$ such that $\lim\limits_{k\to+\infty} \gamma_k = 0$, $\lim\limits_{k\to+\infty} \epsilon_k = 0$, $(x_k + \epsilon_k - x_*)^T \operatorname{sgn}(\nabla f(x_k + \epsilon_k)) > 0$,*

$$(x_k + \epsilon_k - x_*)^T \operatorname{sgn}(\nabla f(x_k + \epsilon_k)) - \sqrt{\Delta_k} < n\gamma_k < (x_k + \epsilon_k - x_*)^T \operatorname{sgn}(\nabla f(x_k + \epsilon_k)) + \sqrt{\Delta_k} \tag{7}$$

with

$$\Delta_k = \left((x_k + \epsilon_k - x_*)^T \operatorname{sgn}(\nabla f(x_k + \epsilon_k))\right)^2 - 2n\epsilon_k^T \left(x_k + \frac{\epsilon_k}{2} - x_*\right) > 0$$

and

$$\gamma_k (x_k + \epsilon_k - x_*)^T \operatorname{sgn}(\nabla f(x_k + \epsilon_k)) \geqslant c\|x_k - x_*\|^\alpha, \qquad c > 0,\ \alpha > 0 \tag{8}$$

for all $k \in \mathbb{N}$. Then, the sequence x_k given by the USGD (6) satisfies $\lim\limits_{k\to+\infty} x_k = x_*$.

Proof. Consider the following Lyapunov sequence

$$V(x_k) = \|x_k - x_*\|^2.$$

By using (6), one gets

$$\begin{aligned} V(x_{k+1}) - V(x_k) &= \|x_{k+1} - x_*\|^2 - \|x_k - x_*\|^2 \\ &= \|x_k + \epsilon_k - \gamma_k \ \operatorname{sgn}(\nabla f(x_k + \epsilon_k)) - x_*\|^2 - \|x_k - x_*\|^2 \\ &= n\gamma_k^2 - 2\gamma_k (x_k + \epsilon_k - x_*)^T \operatorname{sgn}(\nabla f(x_k + \epsilon_k)) + 2\epsilon_k^T \left(x_k + \frac{\epsilon_k}{2} - x_*\right). \end{aligned}$$

Consider $V(x_{k+1}) - V(x_k)$ as a second order polynomial in γ_k with a discriminant reading as $\Delta_k > 0$. It leads to $V(x_{k+1}) - V(x_k) < 0$ if and only if (7) is satisfied. So, we deduce that $V(x_k)$ is decreasing. As $V(x_k)$ is decreasing and bounded from below by zero, the monotone convergence theorem given for instance in [6, Theorem 3.2] implies that $V(x_k)$ is convergent. As $\lim_{k\to+\infty} \gamma_k = 0$, $\lim_{k\to+\infty} \epsilon_k = 0$ and $V(x_k)$ is convergent we deduce that

$$\lim_{k\to+\infty} \gamma_k (x_k + \epsilon_k - x_*)^T \operatorname{sgn}(\nabla f(x_k + \epsilon_k)) = \lim_{k\to+\infty} (x_k + \epsilon_k - x_*)^T (x_k + \epsilon_k - x_{k+1}) = 0.$$

By using (8), we conclude that $\lim_{k\to+\infty} x_k = x_*$. □

It is well known in control theory that sliding mode control has good robustness properties due to the use of the sign function, see for instance [46]; similar feature can be expected for the SGD in Theorem 2.

By choosing the following step size

$$\gamma_k = \gamma_{k,1} |\nabla f(x_k)| + \gamma_{k,2}, \qquad k \in \mathbb{N}$$

for (2) where $\gamma_{k,1}$ and $\gamma_{k,2}$ are two step size, one obtains hereafter the hybrid gradient descent algorithm which is the GD (1) with a new degree of freedom brought by the SGD (2).

Definition 4. *Consider a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$. The hybrid gradient descent algorithm (HGD) is defined by the following discrete-time system*

$$x_{k+1} = x_k - \gamma_{k,1} \nabla f(x_k) - \gamma_{k,2} \operatorname{sgn}(\nabla f(x_k)), \qquad k \in \mathbb{N} \tag{9}$$

where $x_k \in \mathbb{R}^n$ is the state and $\gamma_{k,1} > 0$, $\gamma_{k,2} > 0$ the step size to be defined.

3. Applications

The speed of convergence, given by the number of iterations, of the SGD algorithms satisfying conditions of Theorem 1 is usually better than the one of the GD algorithms and this has been highlighted for neural networks with the RPROP algorithm in [42] and [43, Subsection 8.3.3]. We recover this practical result on several examples for the different gradient descent algorithms (1), (2) and (9). Moreover, if we allow the initial step size γ_0 not to fulfill condition (4) of Theorem 1 then it is possible to converge towards other minima than the closest minimum of the initial condition. This new practical result allowing global optimization is highlighted in the examples below.

In the sequel, constant and variable steps are used for the step size of the GD (1). The same rule can also be used for tuning the step size $\gamma_{k,1}$ of the HGD (9) which has always one more degree of freedom $\gamma_{k,2}$ than the GD (1). One chooses for the different gradient descent algorithms (1), (2) and (9) the maximum step size ensuring the maximum speed of convergence with a given initial condition x_0 and a given precision ε providing the stopping criterion of the algorithm. Finally, we will use a special ASGD for applications defined hereafter.

Definition 5. *The ASGD with the following geometric sequence $\gamma_0 > 0$ and $\gamma_{k+1} = \frac{\gamma_k}{2}$ is named the dichotomy algorithm (DICHO) after the dichotomy method. Moreover, we have $\gamma_k = \gamma_0 \cdot 0.5^k$.*

3.1. Polynomial scalar functions

Consider the following function

$$f_1(x) = x^4, \qquad x \in [-5, 5]$$

that is plotted on Figure 1. For all algorithms, the initial condition and the precision are taken equal to $x_0 = 4$ and $\varepsilon = 10^{-5}$ respectively. The speeds of convergence are provided in Table 1.

Let $p \in \mathbb{N}\backslash\{0\}$ be an even integer, $c_1, c_2 \in \mathbb{R}$ and consider the basic function

$$f(x) = (x + c_1)^p + c_2.$$

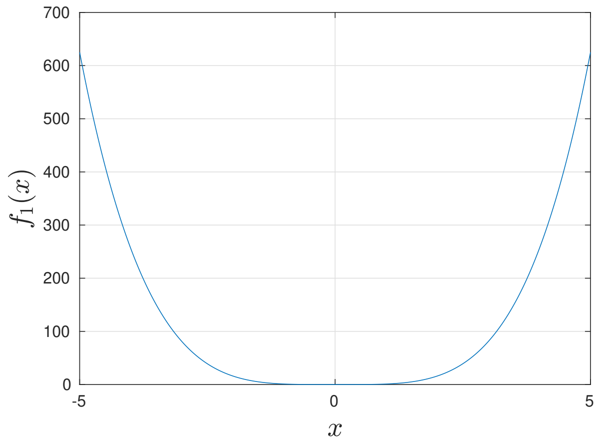


Figure 1: Function f_1

	GD	SGD	DICHO	HGD
Step size	$\gamma_k = 0.03$	$\gamma_k = 0.9^k$	$\gamma_0 = 5 \qquad \gamma_{k+1} = \frac{\gamma_k}{2}$	$\gamma_{k,1} = 0.03 \qquad \gamma_{k,2} = 0.5^k$
Number of iterations	2177	109	19	2178

Table 1: Speeds of convergence of gradient descent algorithms for f_1

There is a way for converging towards the minimum $x_* = -c_1$ in only one step. If we choose the step size

$$\gamma_k = p^{-\frac{1}{p-1}} |\nabla f(x_k)|^{\frac{1}{p-1}}, \qquad k \in \mathbb{N}$$

with $0 < \frac{1}{p-1} \leqslant 1$ for (2), it leads to the following discrete-time system

$$x_{k+1} = x_k - p^{-\frac{1}{p-1}} |\nabla f(x_k)|^{\frac{1}{p-1}} \operatorname{sgn}(\nabla f(x_k)) = -c_1, \qquad k \in \mathbb{N}. \tag{10}$$

The continuous function $x \mapsto |x|^\alpha \operatorname{sgn}(x)$ with $0 < \alpha < 1$ has the property to render continuous systems as finite time stable [9]. This is also the case for the discrete-time system (10) which is finite time convergent after the first step $k = 1$. However, this strategy can only be used if p is known but not c_1 and c_2.

3.2. A non convex scalar function

In this subsection, a comparison of the different gradient descent algorithms has been performed for the following non convex function

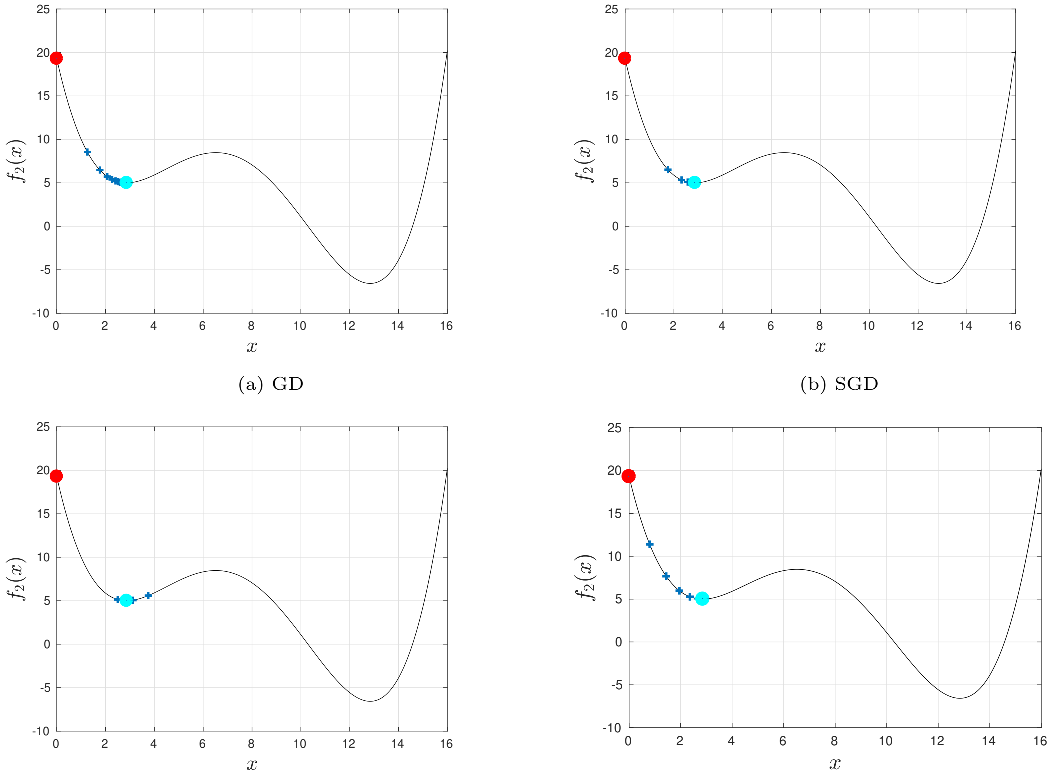
$$f_2(x) = 0.0131x^4 - 0.3881x^3 + 3.644x^2 - 12.55x + 19.29, \qquad x \in [0, 16].$$

which has a local minimum in $x_{*1} = 2.8621$ and a global minimum in $x_{*2} = 12.84$ (see Figure 2). For all the algorithms, the initial condition and the precision are taken equal to $x_0 = 0$ and $\varepsilon = 10^{-5}$ respectively. The speeds of convergence are provided in Table 2.

	GD	SGD	DICHO	HGD
Step size	$\gamma_k = 0.1$	$\gamma_k = 0.8^k$	$\gamma_0 = 5 \qquad \gamma_{k+1} = \frac{\gamma_k}{2}$	$\gamma_{k,1} = 0.1 \qquad \gamma_{k,2} = 0.5^k$
Number of iterations	47	51	18	38

Table 2: Speeds of convergence of gradient descent algorithms for f_2

The trajectories are represented on Figure 2 where the red circle is the starting point, the cyan circle is the optimum and the blue crosses are intermediate states.



(a) GD (b) SGD

(c) DICHO (d) HGD

Figure 2: Trajectories of gradient descent algorithms for f_2 with a precision of $\varepsilon = 10^{-5}$

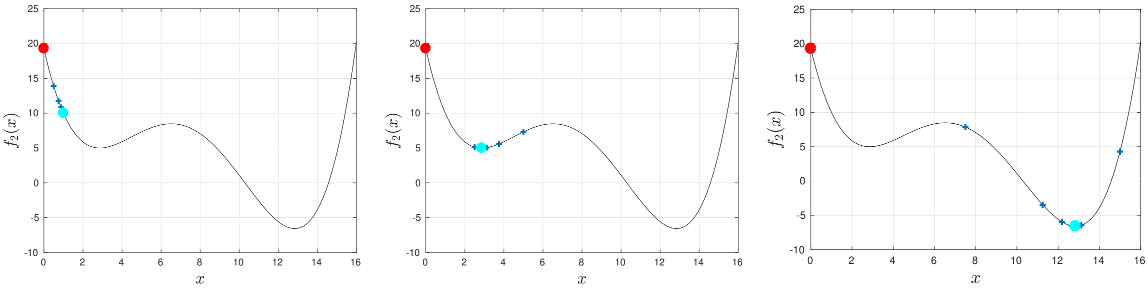
Notice that several initial conditions have been tested and it is observed that the HGD is always faster in terms of number of iterations than the GD. This highlights the effect of the new degree of freedom $\gamma_{k,2}$ brought by the discontinuous sign function. Note that the optimum found by the different algorithms depends on the initial condition: for $x_0 = 0$, all the algorithms converge to $x_{*1} = 2.8620$.

Compare now the influence of the initial step size γ_0 on the DICHO. The analysis of Table 3 is the following:

- when γ_0 is too small then the DICHO converges but not to a minimum (see Figure 3a);
- when γ_0 is not large then the DICHO converges to the local minimum x_{*1} (see Figure 3b);
- when γ_0 is large enough then the DICHO converges to the global minimum x_{*2} even for γ_0 very large (see Figure 3c).

We observe the chattering phenomenon on Figure 3b and Figure 3c when the states oscillate on both sides of the equilibrium point. This phenomenon is well known in sliding mode control theory for continuous systems [17]. Notice that, in the context of control systems, chattering can be damageable for the closed-loop system performances. However, in the current context, this phenomenon has no negative effect.

As far as the speed of convergence is concerned, γ_0 does not play a crucial role to reduce the number of iterations for the DICHO but choosing γ_0 sufficiently large can allow to converge towards the global minimum by avoiding the local minimum (see Table 3). For this, we allow γ_0 not to fulfill condition (4) of Theorem 1 and then it is possible for the DICHO to converge towards other minima than the closest minimum of the initial condition.



(a) $\gamma_0 = 1$ (b) $\gamma_0 = 10$ (c) $\gamma_0 = 30$

Figure 3: Trajectories of the DICHO for f_2 with a precision of $\varepsilon = 10^{-5}$

γ_0	0.1	1	3	10	20	100	1000
Number of iterations	13	16	18	19	20	23	26
Minimum	✗	✗	x_{*1}	x_{*1}	x_{*2}	x_{*2}	x_{*2}

Table 3: Influence of the initial step size on the DICHO for f_2

From Table 4, one can see that the constant step size of the GD needs to be sufficiently small to ensure the convergence towards the closest minimum x_{*1} and that varying step size cannot ensure convergence of the GD. It implies that the GD can only converge towards the local minimum closest to the initial condition rendering this method usable for local optimization only. This is a key difference with the DICHO which can be used for global optimization.

γ_k	0.01	0.1	1	10	$0.1 \cdot 0.5^k$	$1 \cdot 0.5^k$	$3 \cdot 0.5^k$
Number of iterations	406	47	125	5	15	15	8
Minimum	x_{*1}	x_{*1}	x_{*1}	✗	✗	✗	✗

Table 4: Influence of the step size on the GD for f_2

Remark 1. *If we consider the ASGD* (3) *with a step size γ_k of the form $\gamma_k = \gamma_0 \cdot q^k$ with $0 < q < 1$, the previous analysis shows that:*
– the parameter γ_0 determines the research domain for the minimum; a large γ_0 implies a large research domain, without having a real impact on the speed of convergence;
– the parameter q determines the precision of the research for the global minimum; a parameter q close to 0 implies a high precision whereas a parameter q close to 1 implies a low precision, and has an impact on the speed of convergence.
By restarting the ASGD algorithm with several initial step size γ_0, we obtain a new metaheuristic method [18] allowing to find the global minimum of a function.

3.3. The two dimensional Rosenbrock's function

The Rosenbrock's function reads as

$$f_3(x) = (1 - x_1)^2 + 100\,(x_2 - x_1^2)^2, \qquad x = (x_1, x_2) \in [-2, 2] \times [-1, 3]$$

and has a global minimum in $x_* = [1, 1]^T$. It is plotted on Figure 4 and used here to compare the efficiency of the different gradient descent algorithms. For all the algorithms, the initial condition and the precision are taken equal to $x_0 = [2, 0]^T$ and $\varepsilon = 10^{-5}$ respectively. The speeds of convergence are given in Table 5.

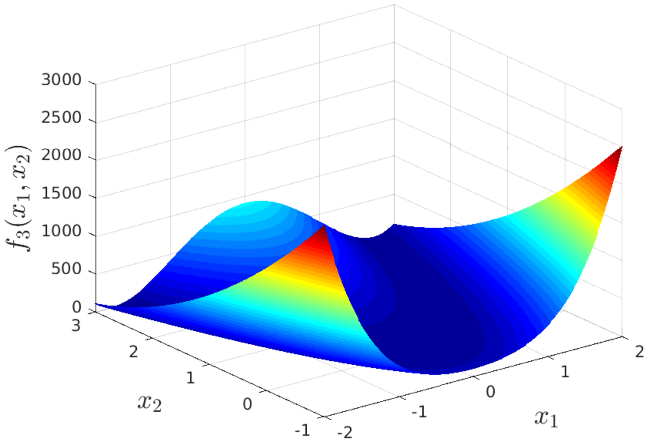


Figure 4: Rosenbrock's function f_3

	GD	DICHO		HGD	
Step size	$\gamma_k = 0.5^k$	$\gamma_0 = 3$	$\gamma_{k+1} = \frac{\gamma_k}{2}$	$\gamma_{k,1} = 0.001$	$\gamma_{k,2} = 0.5^k$
Number of iterations	8270	18		7360	

Table 5: Speeds of convergence of gradient descent algorithms for f_3

The trajectories of the gradient descent algorithms are plotted on Figure 5 where the red circle is the starting point, the cyan circle is the global minimum x_* and the blue crosses are intermediate states. The advantage of the DICHO algorithm is clear since it allows to reduce the number of iterations by almost 500 in comparison with the GD. Finally, if we suppose there are uncertainties of the form $\epsilon_k = \epsilon_0 \cdot 0.5^k$ on the values x_k then the USGD (6) with $\gamma_k = 0.5^k$ converges for all $0 < \epsilon_0 \leqslant 0.05$.

3.4. The DICHO algorithm and the multivariable Rastrigin's function

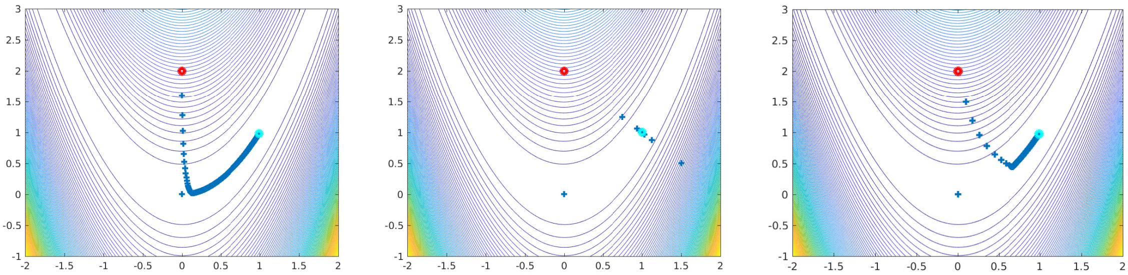
In order to provide an example in large dimension, we consider the nonlinear multivariable Rastrigin's function defined by

$$f_4(x) = 10n + \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i)\right), \qquad x = (x_1, \ldots, x_n) \in [-5.12, 5.12]^n. \tag{11}$$

In the case of $n = 2$, the function is represented on Figure 6. Due to the high number of local minima, we know that the GD is not able to find the global minimum which is known to be at $x = 0$ with $f_4(0) = 0$ [34]. The simulation results for large dimension $n = 100000$ and precision $\varepsilon = 10^{-5}$ are given in Tables 6 and 7 for different initial conditions. It can be seen that the GD algorithm never converges even to a local minimum. On the contrary, the DICHO algorithm always converges to a minimum which is the global minimum if the initial condition x_0 belongs to $[-0.5, 0.5]$ and a local minimum otherwise.

	GD			DICHO	
Step size	$\gamma_k = 0.001$	$\gamma_k = 0.01$	$\gamma_k = 0.1$	$\gamma_0 = 0.8$	$\gamma_{k+1} = \frac{\gamma_k}{2}$
Minimum	✗	✗	✗	global	
Number of iterations	✗	✗	✗	24	

Table 6: Comparison between the GD and DICHO for $n = 100000$ and $x_0 \in [-0.5, 0.5]^n$



(a) GD (b) DICHO (c) HGD

Figure 5: Trajectories of gradient descent algorithms for f_3 with a precision of $\varepsilon = 10^{-5}$

	GD			DICHO
Step size	$\gamma_k = 0.001$	$\gamma_k = 0.01$	$\gamma_k = 0.1$	$\gamma_0 = 0.8$ $\gamma_{k+1} = \frac{\gamma_k}{2}$
Minimum	✗	✗	✗	local
Number of iterations	✗	✗	✗	24

Table 7: Comparison between the GD and DICHO for $n = 100000$ and $x_0 \notin [-0.5, 0.5]^n$

3.5. The RPROP algorithm and the Himmelblau's function

The RPROP algorithm is an ASGD algorithm where $\gamma(k)$ is denoted $\Delta^{(k)}$ and defined in [42, Equation (4)]. It is mentioned in [42] that $\gamma(0) = \Delta^{(0)}$ has no influence on the speed of convergence of the RPROP algorithm. This result has also been observed on the example of Subsection 3.2 with DICHO. However, we will see that the initial step size $\Delta^{(0)}$ has an influence on the convergence of the RPROP algorithm if the function to study has several minima.

The Himmelblau's function defined by

$$f_5(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2, \qquad x = (x_1, x_2) \in [-5, 5] \times [-5, 5] \tag{12}$$

is plotted on Figure 7 and has 4 minima denoted M1, M2, M3, M4 on Figure 8.

The RPROP$^-$ algorithm recalled in [23] has been implemented and tested for different values of the initial step size γ_0 with the initial condition $x_0 = [0, 0]^T$ and the precision $\varepsilon = 10^{-5}$. We have $x_1 = \gamma_0 \cdot [1, 1]^T$. The results are displayed on Figure 8. One sees that the RPROP$^-$ algorithm converges to the four different minima depending on the values of the initial step size γ_0. Table 8 sums up the different convergence results with respect to γ_0. It shows that the RPROP algorithm can be used as a new metaheuristic method.

γ_0	minimum
4	M1
5	M2
7	M3
8	M4

Table 8: Different minima achieved with different values of the initial step size γ_0

4. Conclusion

In this article, the first-order gradient descent algorithm involving the sign of the gradient, called *sign gradient descent algorithm*, is developed. To facilitate the tuning of the step size, the adaptive sign gradient descent algorithm is introduced. Moreover, the hybrid gradient descent algorithm is defined and it brings

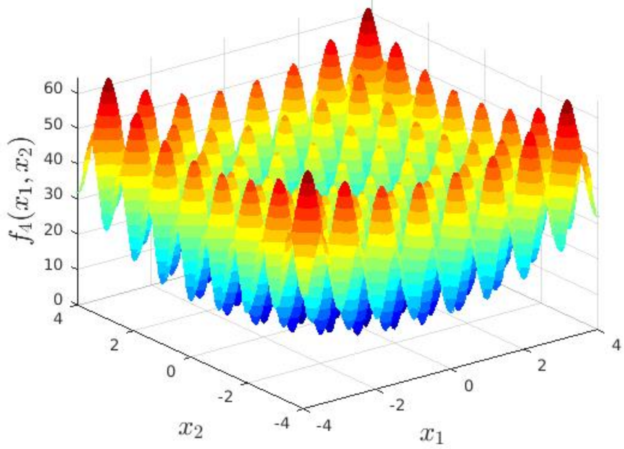


Figure 6: Rastrigin's function f_4

an additional degree of freedom for tuning classical gradient descent. Two results of convergence for local optimization are provided and several examples are treated. The sign gradient descent algorithms can be faster than classical gradient descent algorithm. Moreover, they can allow to reach other minima than the closest minimum of the initial condition making these algorithms usable for global optimization.

Acknowledgements

The authors would like to that Emmanuel Kravitzch from University of Poitiers for pointing out an issue in Theorem 1 and Theorem 2 which has been fixed in this postprint version of our article "Properties of the sign gradient descent algorithms" published in *Information Sciences* (volume 492, pages 29-39) in 2019.

References

References

[1] Aristoklis D Anastasiadis, George D Magoulas, and Michael N Vrahatis. New globally convergent training scheme based on the resilient propagation algorithm. *Neurocomputing*, 64:253–270, 2005.

[2] Aliasghar Arab and Alireza Alfi. An adaptive gradient descent-based local search in memetic algorithm applied to optimal controller design. *Information Sciences*, 299:117–142, 2015.

[3] Bijnan Bandyopadhyay and Deepak Fulwani. High-performance tracking controller for discrete plant using nonlinear sliding surface. *IEEE Transactions on Industrial Electronics*, 56(9):3628–3637, 2009.

[4] Giorgio Bartolini, Antonella Ferrara, and Vadim I Utkin. Adaptive sliding mode control in discrete-time systems. *Automatica*, 31(5):769–773, 1995.

[5] Andrzej Bartoszewicz. Discrete-time quasi-sliding-mode control strategies. *IEEE Transactions on Industrial Electronics*, 45(4):633–637, 1998.

[6] Richard Beals. *Analysis: an introduction.* Cambridge University Press, 2004.

[7] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Anima Anandkumar. Compression by the signs: distributed learning is a two-way street. In *6th International Conference on Learning Representations*, pages 1–6, 2018.

[8] Dimitri P Bertsekas, Angelia Nedic, and Asuman E Ozdaglar. *Convex analysis and optimization.* Athena Scientific, 2003.

[9] Sanjay P Bhat and Dennis S Bernstein. Finite-time stability of continuous autonomous systems. *SIAM Journal on Control and Optimization*, 38(3):751–766, 2000.

[10] Joseph-Frédéric Bonnans, Jean Charles Gilbert, Claude Lemaréchal, and Claudia A Sagastizábal. *Numerical optimization: theoretical and practical aspects.* Universitext. Springer, 2006.

[11] Léon Bottou. Online learning and stochastic approximations. In *On-line learning in neural networks*, volume 17 of *Publications of the Newton Institute*, pages 9–42. Cambridge University Press, 2009.

[12] Stephen Boyd and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

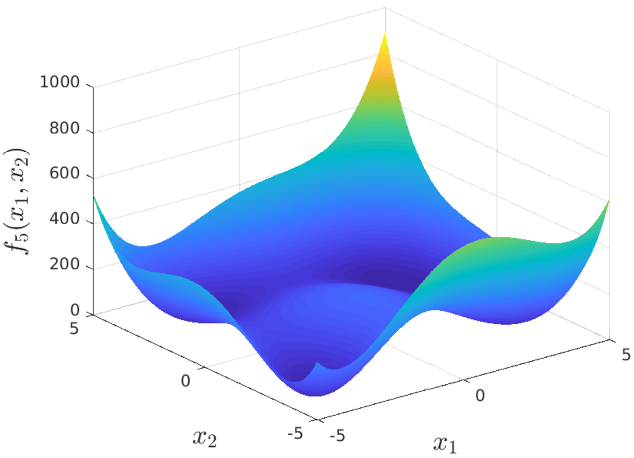


Figure 7: Himmelbau's function f_5

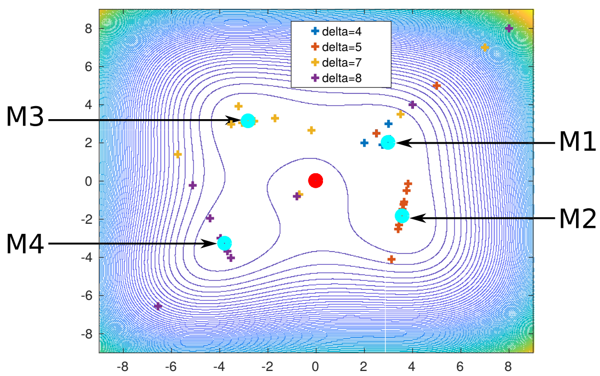


Figure 8: Convergence of the RPROP$^-$ algorithm for f_4 with different values of the initial step size γ_0

[13] Francis H Clarke, Yuri S Ledyaev, Ronald J Stern, and Peter R Wolenski. *Nonsmooth analysis and control theory*, volume 178 of *Graduate Texts in Mathematics*. Springer, 1998.

[14] John E Dennis Jr and Robert B Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*, volume 16 of *Classics in Applied Mathematics*. SIAM, 1996.

[15] Christopher Edwards and Sarah Spurgeon. *Sliding mode control: theory and applications*. CRC Press, 1998.

[16] Aleksej Fedorovič Filippov. *Differential equations with discontinuous righthand sides: control systems*, volume 18 of *Mathematics and its Applications*. Springer, 1988.

[17] Leonid Fridman, Jaime Moreno, and Rafael Iriarte. *Sliding modes after the first decade of the 21st century*, volume 412 of *Lecture Notes in Control and Information Sciences*. Springer, 2011.

[18] Michel Gendreau and Jean-Yves Potvin. *Handbook of metaheuristics*, volume 146 of *International Series in Operations Research & Management Science*. Springer, 2010.

[19] Giorgio Grisetti, Cyrill Stachniss, Slawomir Grzonka, and Wolfram Burgard. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. In *Robotics: Science and Systems*, volume 3, page 9, 2007.

[20] Lars Grüne and Jürgen Pannek. *Nonlinear model predictive control: theory and algorithms*. Communications and Control Engineering. Springer, 2011.

[21] Guoxiang Gu. *Discrete-Time Linear Systems: Theory and Design with Applications*. Control Engineering. Springer, 2012.

[22] Wassim M Haddad and VijaySekhar Chellaboina. *Nonlinear dynamical systems and control: a Lyapunov-based approach*. Princeton University Press, 2008.

[23] Christian Igel and Michael Hüsken. Improving the Rprop learning algorithm. In *Proceedings of the Second International Symposium on Neural Computation*, pages 115–121, 2000.

[24] Christian Igel and Michael Hüsken. Empirical evaluation of the improved rprop learning algorithms. *Neurocomputing*,

50:105–123, 2003.
[25] Yaochu Jin. *Multi-objective machine learning*, volume 16 of *Studies in Computational Intelligence*. Springer, 2006.
[26] Okyay Kaynak and Ahmet Denker. Discrete-time sliding mode control in the presence of system uncertainty. *International Journal of Control*, 57(5):1177–1189, 1993.
[27] Donghwan Kim and Jeffrey A Fessler. Optimized first-order methods for smooth convex minimization. *Mathematical Programming*, 159(1-2):81–107, 2016.
[28] Ren-Jieh Kuo and Ferani E Zulvia. The gradient evolution algorithm: A new metaheuristic. *Information Sciences*, 316:246–265, 2015.
[29] Mircea Lazar, Maurice Heemels, Siep Weiland, and Alberto Bemporad. Stabilizing model predictive control of hybrid systems. *IEEE Transactions on Automatic Control*, 51(11):1813–1818, 2006.
[30] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436, 2015.
[31] Arie Levant. Higher-order sliding modes, differentiation and output-feedback control. *International journal of Control*, 76(9-10):924–941, 2003.
[32] Yan Liao, Ning Deng, Huaqiang Wu, Bin Gao, Qingtian Zhang, and He Qian. Weighted synapses without carry operations for RRAM-based neuromorphic systems. *Frontiers in neuroscience*, 12:167, 2018.
[33] Lurng-Kuo Liu and Ephraim Feig. A block-based gradient descent search algorithm for block motion estimation in video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(4):419–422, 1996.
[34] Heinz Mühlenbein, M Schomisch, and Joachim Born. The parallel genetic algorithm as function optimizer. *Parallel Computing*, 17(6-7):619–632, 1991.
[35] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87 of *Applied Optimization*. Springer, 2004.
[36] Jorge Nocedal and Stephen J Wright. *Nonlinear Equations*. Springer Series in Operations Research and Financial Engineering. Springer, second edition, 2006.
[37] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *30th International Conference on Machine Learning*, pages 1310–1318, Atlanta, USA, 2013.
[38] Franck Plestan, Emmanuel Moulay, Alain Glumineau, and Thibault Cheviron. Robust output feedback sampling control based on second-order sliding mode. *Automatica*, 46(6):1096–1100, 2010.
[39] Franck Plestan, Yuri Shtessel, Vincent Bregeault, and Alex Poznyak. New methodologies for adaptive sliding mode control. *International Journal of Control*, 83(9):1907–1919, 2010.
[40] Nathan Ratliff, Matt Zucker, J Andrew Bagnell, and Siddhartha Srinivasa. CHOMP: Gradient optimization techniques for efficient motion planning. In *IEEE International Conference on Robotics and Automation*, pages 489–494, 2009.
[41] Stefan Richter, Colin Neil Jones, and Manfred Morari. Computational complexity certification for real-time mpc with input constraints based on the fast gradient method. *IEEE Transactions on Automatic Control*, 57(6):1391–1403, 2012.
[42] Martin Riedmiller and Heinrich Braun. A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In *IEEE International Conference on Neural Networks*, pages 586–591, 1993.
[43] Raúl Rojas. *Neural networks: a systematic introduction*. Artificial Intelligence. Springer, 2013.
[44] Sam Z Sarpturk, Yorgo Istefanopulos, and Okyay Kaynak. On the stability of discrete-time sliding mode control systems. *IEEE Transactions on Automatic Control*, 32(10):930–932, 1987.
[45] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.
[46] Yuri Shtessel, Christopher Edwards, Leonid Fridman, and Arie Levant. *Sliding mode control and observation*. Birkhäuser, Control Engineering. Springer, 2014.
[47] Krzysztof Sopyła and Paweł Drozda. Stochastic gradient descent with Barzilai–Borwein update step for SVM. *Information Sciences*, 316:218–233, 2015.
[48] James C Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons, 2005.
[49] Vadim Utkin. *Sliding modes in control and optimization*. Communications and Control Engineering. Springer, 1992.
[50] David Valiente, Arturo Gil, Lorenzo Fernández, and Óscar Reinoso. A modified stochastic gradient descent algorithm for view-based slam using omnidirectional images. *Information Sciences*, 279:326–337, 2014.

View publication stats