Pré-Publicações do Departamento de Matemática
Universidade de Coimbra
Preprint Number 19–45

TRUST-REGION METHODS FOR THE DERIVATIVE-FREE OPTIMIZATION OF NONSMOOTH BLACK-BOX FUNCTIONS

G. LIUZZI, S. LUCIDI, F. RINALDI AND L. N. VICENTE

ABSTRACT: In this paper we study the minimization of a nonsmooth black-box type function, without assuming any access to derivatives or generalized derivatives and without any knowledge about the analytical origin of the function nonsmoothness. Directional methods have been derived for such problems but to our knowledge no model-based method like a trust-region one has yet been proposed.

Our main contribution is thus the derivation of derivative-free trust-region methods for black-box type functions. We propose a trust-region model that is the sum of a max-linear term with a quadratic one so that the function nonsmoothness can be properly captured, but at the same time the curvature of the function in smooth subdomains is not neglected. Our trust-region methods enjoy global convergence properties similar to the ones of the directional methods, provided the vectors randomly generated for the max-linear term are asymptotically dense in the unit sphere. The numerical results reported demonstrate that our approach is both efficient and robust for a large class of nonsmooth unconstrained optimization problems. Our software is made available under request.

KEYWORDS: Nonsmooth optimization, derivative-free optimization, trust-region-methods, black-box functions.

1. Introduction

We develop a trust-region methodology for the derivative-free optimization of a possibly nonsmooth function without any knowledge about the source or form of its nonsmoothness. The objective function to be minimized is thus treated as a pure black box in the sense of only returning function values. Our goal is to develop an algorithm that is both efficient (in terms of the number of function evaluations taken to reach a meaningful stopping criterion) and rigorous (in terms of offering reasonable convergence properties). The derivative-free trust-region approach is a step towards efficiency as we know that it works well for smooth problems or even mildly nonsmooth

Received December 30, 2019.
Support for LNV was partially provided by FCT/Portugal under grants UID/MAT/00324/2019 (CMUC) and P2020 SAICTPAC/0011/2015.

ones, and we want to take advantage in our implementation of what are some of the best existing numerical practices known for these methods, since most nonsmooth problems, even the harder ones, exhibit large subdomains of smoothness.

However, the theory and practice of derivative-free trust-region methods have been developed for either smooth functions [10, 11] or for the sum or composition of a known nonsmooth function with a smooth, possibly vectorial black-box one [16, 18, 25, 30][1]. The trust-region models typically used are smooth, based on quadratic or radial basis functions. However, without any knowledge about the subdifferential of the function or access to its members, as it is the case in a pure black-box regime, the use of nonsmooth models based on a finite number of basis function elements or nonsmooth operators may not suffice to explore increasingly narrow cones of descent directions and render a trust-region algorithm convergent. Furthermore, in the theory of directional methods for the derivative-free optimization of black-box functions [3, 14, 38], it is proved the nonnegativity of generalized directional derivatives at certain limit points of the sequences of iterates along certain limiting directions. Such limiting directions cover the unit sphere if the algorithm directions are randomly generated in such a way that their support is the unit sphere for any subsequence of the iterates[2].

Hence, our trust-region models will have built-in the random generation of their linear terms. In a first, naive but simple approach this can be achieved by randomly generating the vector defining the linear term of a quadratic. Such models will however render the trust-region method inefficient as they are to some extent just adding a quadratic term to a directional-inspired linear one, without any attempt to explore the nonsmoothness of the function. We thus go a step further and propose a nonsmooth trust-region model by collecting a number of those randomly generated linear terms in a max-linear type model, adding to it the quadratic term for steady progress in the more

[1]An anonymous Referee has drawn our attention to the recent works [2, 22] (the latter one of trust-region type). However, both require the calculation of subgradients of approximate or nearby subdifferentials, therefore only applicable when the nonsmoothness of the objective function is known through some algebraic or composite form.

[2]Not all existing converging techniques for nonsmooth derivative-free optimization are based on random directions, an example being the use of the convex hull of (possibly randomly) sampled approximate gradients [4, 29, 21]. On the other hand, trust-region methods have also been developed based on probabilistic random models [6, 19] but for smooth problems.

smooth function subdomains. Our work is inspired by the bundle methodology [35] for nonsmooth optimization. By working with a sample set of points near the current iterate and appropriately using their function values, one forms the max-linear model in a way that the new vector randomly generated at each iteration (for the purpose of adding a new linear term to the model) will attempt to approximate in a certain way an element of the subdifferential at one of these sample points. Our numerical experiments have shown that such a methodology can lead to an efficient and particularly robust solver for the derivative-free optimization of nonsmooth functions.

In a way similar to the application of direct-search methods to nonsmooth functions, our convergence results state that the Clarke generalized derivative is nonnegative at any limit point of a subsequence of unsuccessful iterates, along any direction in the unit sphere, assuming some form of asymptotic density of the vectors randomly generated for the linear terms of the models. The Hessian of the quadratic term added to the max-linear model term does not have to be positive definite or semidefinite and, interestingly, not even necessarily bounded (as long as it does not grow faster than a certain negative power in $(-1, 0)$ of the trust-region radius).

The paper is organized in the following sections: In Section 2 we describe and analyze the basic version of our trust-region derivative-free algorithmic approach, based on a quadratic trust-region model with a randomly generated vector of linear coefficients. Then in Section 3 we introduce our random nonsmooth max-linear trust-region model and adapt the convergence analysis to this more advanced scenario. Section 4 describes the implementation of our basic and advanced algorithms and reports the numerical experiments conducted for a test of nonsmooth problems. Some conclusions and prospects of future work are outlined in Section 5. In terms of notation, all norms are Euclidean.

In this paper we consider an unconstrained minimization problem

$$\min_{x \in \mathbb{R}^n} \; f(x), \tag{1}$$

where the objective function, although possibly nonsmooth, will be assumed locally Lipschitz continuous whenever needed in the theory. Any type of first-order information (gradients or elements of subdifferentials) is considered unavailable or impractical to obtain.

Given a point $x \in \mathbb{R}^n$, at which the function is Lipschitz continuous in a neighborhood of, and a direction $d \in \mathbb{R}^n$, the Clarke generalized derivative [8] of f at x along d can be defined as

$$f^\circ(x;d) \;=\; \limsup_{y\to x, t\downarrow 0} \frac{f(y+td)-f(y)}{t} \;=\; \lim_{\epsilon\to 0} \sup_{(y,t):\|y-x\|\le\epsilon, 0<t\le\epsilon} \frac{f(y+td)-f(y)}{t}.$$

Later in the paper we will use the fact that $f^\circ(\cdot;\cdot)$ is Lipschitz continuous in its second argument (with Lipschitz constant equal to the one of f). Suppose again that f is Lipschitz near x. The Clarke generalized subdifferential of f at x can then be defined by

$$\partial f(x) \;=\; \{s \in \mathbb{R}^n : \; f^\circ(x;v) \ge v^\top s, \; \forall v \in \mathbb{R}^n\}.$$

Moreover, it can be proved that

$$f^\circ(x;d) \;=\; \max\{d^\top s : s \in \partial f(x)\}.$$

A point $x^* \in \mathbb{R}^n$ is (first order) Clarke stationary for problem (1) when $f^\circ(x^*;d) \ge 0$, for all $d \in \mathbb{R}^n$.

2. A basic trust region-type algorithm based on a smooth random model

We will see in this section that one way of endowing a trust-region method with the capacity to deal with nonsmooth black-box functions is by randomly generating the linear coefficients of the quadratic trust-region model. We suggest to generate the vector g_k of the linear coefficients of the quadratic model randomly in the unit sphere. Such a normalization tries to mimic to some extent the effect of a step of directional methods [3, 38], in the sense that a trust-region step of the form $\Delta_k g_k$, where Δ_k is the trust-region radius and $\|g_k\| = 1$, plays a similar role as the step of such directional methods. However, we consider also a quadratic term in the trust-region model to better approximate the curvature in smooth subdomains, and this is actually one of the theoretical challenges in this paper. So, at every iteration k of the `Basic DFO-TRNS` Algorithm, s_k is the solution of the trust-region subproblem

$$\begin{array}{ll} \min & f(x_k) + g_k^\top s + \dfrac{1}{2} s^\top B_k s \\ \text{s.t.} & \|s\|^2 \;\le\; \Delta_k^2, \end{array} \tag{2}$$

where $B_k \in \mathbb{R}^{n\times n}$ is a symmetric matrix built out from interpolation or regression on a sample set of points.

Algorithm `Basic DFO-TRNS` is quite simple. The step is accepted/rejected and the trust-region radius is nondecreased/decreased based on a ratio of actual versus predicted reductions. The only nonstandard aspect is the choice of the predicted reduction $\theta\|s\|^{1+p}$ used in place of the reduction achieved in the quadratic model from $s = 0$ to $s = s_k$. The reason for our choice lies in a convergence requirement, as we need the predicted reduction to behave like $o(\|s_k\|)$ in unsuccessful iterations to prove the nonnegativity of the Clarke generalized derivative along appropriate directions. Such an effect is similar to the use of a *forcing function* in directional methods [38]. Note that the use of a power of the norm of the step to replace the more traditional predicted reduction has been used before in trust-region methods [12].

Algorithm `Basic DFO-TRNS` `(Basic DFO Trust-Region Algorithm for Nonsmooth Problems)`

Initialization. Select $x_0 \in \mathbb{R}^n$, $\eta_1, \theta > 0$, $0 < \gamma_1 < 1 \leq \gamma_2$, $\Delta_0 > 0$, and $p > 0$.

For $k = 0, 1 \ldots$

Generate g_k randomly and densely on the unit sphere. Build a symmetric matrix B_k.

Let

$$
\begin{aligned}
s_k &\in \operatorname*{argmin}_{\|s\|^2 \leq \Delta_k^2} f(x_k) + g_k^\top s + \frac{1}{2} s^\top B_k s, \\
\rho_k &= \frac{f(x_k) - f(x_k + s_k)}{\theta \|s\|^{1+p}}.
\end{aligned}
$$

If $\rho_k \geq \eta_1$ **Then** set `SUCCESS` $\leftarrow$ `true`, $x_{k+1} \leftarrow x_k + s_k$, $\Delta_{k+1} \leftarrow \gamma_2 \Delta_k$,

Else set `SUCCESS` $\leftarrow$ `false`, $x_{k+1} \leftarrow x_k$, $\Delta_{k+1} \leftarrow \gamma_1 \Delta_k$.

End If

End For

For convergence purposes, we require the model Hessian to satisfy the assumption below. We point out that such an assumption is weaker than what is considered in trust-region methods, where an upper bound on the norm of B_k is traditionally imposed. Our theory allows B_k to be unbounded as long as it is bounded by a negative power of the trust-region radius (which in turn will be proved to converge to zero). This negative power must lie in $(-1, 0)$ for the basic algorithm of this section and in $(-1/2, 0)$ for the advanced algorithm of the next section.

Assumption 2.1. *There exist $q \in (0,1)$, $m, M > 0$, such that: The maximal eigenvalue of B_k satisfies*

$$\lambda_{\max}(B_k) \;\leq\; M\Delta_k^{-q}.$$

When B_k has negative eigenvalues, its minimal eigenvalue satisfies

$$-\lambda_{\min}(B_k) \;\leq\; m\Delta_k^{-q}.$$

We start by proving that the trust-region radius goes to zero. It is typical to see the step size along directions or the trust-region radius converging to zero in derivative-free optimization [11].

Lemma 2.1. *Assume that f is bounded from below. Let Assumption 2.1 hold. Any sequence $\{\Delta_k\}$ of trust-region radii produced by Algorithm* `Basic DFO-TRNS` *is such that*

$$\lim_{k\to\infty} \Delta_k \;=\; 0.$$

Proof: Suppose, by contradiction, that $\{\Delta_k\}$ does not converge to zero. Then, there exists $\epsilon > 0$ such that $\#\{k : \Delta_k > \epsilon\} = \infty$. Because of the way Δ_k is updated we must have

$$\#\left\{k : \Delta_k > \epsilon,\, \Delta_{k+1} \geq \Delta_k\right\} \;=\; \infty,$$

in other words, there must exist an infinite number of iterations for which Δ_{k+1} is not decreased, and, for these iterations we have $\rho_k \geq \eta_1$.

If $\lambda_{\min}(B_k) \leq 0$, then we know from the well-known properties of trust-region subproblems that $\|s_k\| = \Delta_k$ (see [9]). If not, then either $\|s_k\| = \Delta_k$ or $s_k = -B_k^{-1}g_k$. In the latter case, from Assumption 2.1, $\|B_k^{-1}g_k\| \geq \lambda_{\min}(B_k^{-1})\|g_k\| \geq (1/M)\Delta_k^q$. As a result we obtain

$$\begin{aligned} f(x_k) - f(x_k + s_k) &\;\geq\; \eta_1\theta\|s_k\|^{1+p} \\ &\;\geq\; \eta_1\theta\min\{\|B_k^{-1}g_k\|, \Delta_k\}^{1+p} \\ &\;\geq\; \eta_1\theta\min\left\{\frac{\epsilon^q}{M}, \epsilon\right\}^{1+p}. \end{aligned}$$

This means that at each iteration where Δ_k is not decreased, f is reduced by a constant. Since f is bounded from below, the number of such iterations cannot be infinite, and hence we arrived at a contradiction. ∎

It is known that the solution of a trust-region subproblem with fixed data and B_k positive definite tends to a step along the negative gradient [13,

Figure 6.4.2] when the trust-region radius converges to zero. The property below may be seen as an expression of this behavior.

Property 2.1. *Any sequence* $\{(x_k, s_k, \Delta_k)\}$ *generated by Algorithm* `Basic DFO-TRNS` *is such that*

$$s_k \;=\; -\Delta_k D_k g_k,$$

with $D_k \in \mathbb{R}^{n\times n}$ *satisfying*

$$\lim_{k\to\infty} D_k \;=\; I.$$

We can show that our simple algorithm exhibits such a property even when subject to unbounded model Hessians. The result is proved under exact optimality of the solution of the trust-region subproblem (2) to avoid an even longer proof. However we could have assumed inexact optimality and deduce a similar result as long as the norm of the residual of the first-order necessary conditions of this subproblem is smaller than a certain power of the trust-region radius.

Proposition 2.1. *Let Assumption 2.1 hold. Assume also that all trust-region subproblems* (2) *are solved up to optimality. Then Algorithm* `Basic DFO-TRNS` *generates sequences* $\{(x_k, s_k, \Delta_k)\}$ *satisfying Property 2.1 (for k sufficiently large).*

Proof: From the first order necessary conditions for problem (2), we know that there exists $\sigma_k \geq 0$ such that

$$B_k s_k + 2\sigma_k s_k \;=\; -g_k, \quad \sigma_k(\|s_k\|^2 - \Delta_k^2) \;=\; 0, \quad \|s_k\|^2 \;\leq\; \Delta_k^2. \tag{3}$$

One can assume that $\|s_k\|^2 = \Delta_k^2$ for k sufficiently large. In fact, as mentioned before, only when B_k is positive definite and $\|-B_k^{-1}g_k\| < \Delta_k$ there is no solution at the boundary. From (3), such cases do render $s_k = -\Delta_k(\Delta_k B_k)^{-1}g_k$ (i.e., $D_k = (\Delta_k B_k)^{-1}$). However, they can only occur a finite number of times as, from Assumption 2.1, we have $\|B_k^{-1}g_k\| \geq (\Delta_k^{q-1}/M)\Delta_k$, and $\Delta_k \to 0$ thus Δ_k^{q-1}/M becomes eventually larger than 1. Hence, from the necessary conditions (3),

$$\sigma_k \;=\; -\frac{1}{2\Delta_k^2}\left(s_k^\top B_k s_k + g_k^\top s_k\right),$$

and

$$\left(B_k - \frac{1}{\Delta_k^2}\left(s_k^\top B_k s_k + g_k^\top s_k\right) I\right) s_k \;=\; -g_k. \tag{4}$$

The rest of the proof is now divided in two main parts.

(A) First we provide a lower and an upper bound for $\eta_k = -s_k^\top B_k s_k - g_k^\top s_k$. Since s_k solves problem (2), it yields a better model value than the step $-\Delta_k g_k$, which means

$$\frac{1}{2} s_k^\top B_k s_k + g_k^\top s_k \;\leq\; \frac{1}{2}\Delta_k^2 g_k^\top B_k g_k - \Delta_k g_k^\top g_k,$$

and, recalling that $\|g_k\| = 1$, we obtain

$$\frac{1}{2} s_k^\top B_k s_k + g_k^\top s_k \;\leq\; \frac{1}{2}\Delta_k^2 \lambda_{\max}(B_k) - \Delta_k. \tag{5}$$

Then, taking into account the fact that $\eta_k = -2(\frac{1}{2} s_k^\top B_k s_k + g_k^\top s_k) + g_k^\top s_k$ and using inequality (5), we can derive a lower bound on η_k from Assumption 2.1

$$\begin{aligned} \eta_k &\geq -\Delta_k^2 \lambda_{\max}(B_k) + 2\Delta_k + g_k^\top s_k \\ &\geq -M\Delta_k^{2-q} + 2\Delta_k - \Delta_k \\ &\geq -M\Delta_k^{2-q} + \Delta_k. \end{aligned} \tag{6}$$

The lower bound (6) guarantees that $-\eta_k = s_k^\top B_k s_k + g_k^\top s_k \neq 0$ for k sufficiently large, a fact that will later be used.

On the other hand, an upper bound on η_k can be derived also from Assumption 2.1 as follows (recall that $\|g_k\| = 1$)

$$\begin{aligned} \eta_k &= -s_k^\top B_k s_k - g_k^\top s_k \\ &\leq \max\{0, -\lambda_{\min}(B_k)\}\|s_k\|^2 + \|s_k\| \\ &\leq m\Delta_k^{-q}\Delta_k^2 + \Delta_k. \end{aligned} \tag{7}$$

(B) Knowing that $\eta_k = -s_k^\top B_k s_k - g_k^\top s_k$ is nonzero for k sufficiently large and using the generalized Sherman-Morrison-Woodbury formula, we obtain

$$\left(B_k + \frac{\eta_k}{\Delta_k^2} I\right)^{-1} \;=\; \frac{\Delta_k^2}{\eta_k} I - \frac{\Delta_k^2}{\eta_k} B_k \left(I + \frac{\Delta_k^2}{\eta_k} B_k\right)^{-1} \frac{\Delta_k^2}{\eta_k} I. \tag{8}$$

From (4) and (8), we can write

$$s_k \;=\; -\Delta_k D_k g_k$$

with

$$D_k \;=\; \frac{\Delta_k}{\eta_k} I - \frac{\Delta_k^2}{\eta_k} B_k \left(I + \frac{\Delta_k^2}{\eta_k} B_k\right)^{-1} \frac{\Delta_k}{\eta_k} I. \tag{9}$$

It remains to prove that $D_k \to I$ as $k \to \infty$, and for that we will use the lower and upper bounds derived in part (A) of the proof. In fact, using (6) and (7) and dividing by Δ_k lead us to

$$-M\Delta_k^{1-q} + 1 \ \le\ \frac{\eta_k}{\Delta_k} \ \le\ 1 + m\Delta_k^{1-q}$$

and from Lemma 2.1

$$\frac{\eta_k}{\Delta_k} \ \to\ 1.$$

Observe also that from this and Assumption 2.1, $\frac{\Delta_k^2}{\eta_k}B_k = \frac{\Delta_k}{\eta_k}(\Delta_k B_k) \to 0$. Finally, taking into account the formula (9), it results that $D_k \to I$ as $k \to \infty$, and the proof is complete. ∎

The next step in the analysis is to show that the Clarke generalized derivative is nonnegative along some limiting normalized trust-region step. Such a result corresponds to what has been obtained for directional methods along the so-called *refining directions*; see [3].

Lemma 2.2. *Assume that f is bounded from below. Let Assumption 2.1 hold. Let $\{(x_k, s_k, \Delta_k)\}$ be a sequence generated by Algorithm* `Basic DFO-TRNS`. *Let $L \subseteq K = \{k : \Delta_{k+1} < \Delta_k\}$ be an index set such that*

$$\lim_{k\in L, k\to\infty} x_k \ =\ x^* \quad \text{and}$$

$$\lim_{k\in L, k\to\infty} \frac{s_k}{\|s_k\|} \ =\ s^*.$$

Then $f^\circ(x^*; s^*) \ge 0$.

Proof: For each $k \in L$ we have from $\rho_k < \eta_1$ (unsuccess in the algorithm) that

$$f(x_k + s_k) - f(x_k) \ >\ -\eta_1\theta\|s_k\|^{1+p}$$

from which we obtain

$$\frac{f(x_k + \|s_k\|[s_k/\|s_k\|]) - f(x_k)}{\|s_k\|} \ >\ -\eta_1\theta\|s_k\|^p. \tag{10}$$

One can introduce s^* in the quotient of (10) as follows

$$\frac{f(x_k + \|s_k\|s^*) - f(x_k)}{\|s_k\|} - \frac{f(x_k + \|s_k\|s^*) - f(x_k + \|s_k\|[s_k/\|s_k\|])}{\|s_k\|} \tag{11}$$

and see that the second term in (11) is bounded by

$$\frac{f(x_k + \|s_k\| s^*) - f(x_k + \|s_k\| [s_k/\|s_k\|])}{\|s_k\|} \;\leq\; L_f^* \| [s_k/\|s_k\|] - s^* \|, \tag{12}$$

where L_f^* is the Lipschitz constant of f near x^*. From (10)–(12) we then have

$$\frac{f(x_k + \|s_k\| s^*) - f(x_k)}{\|s_k\|} \;>\; -\eta_1 \theta \|s_k\|^p - L_f^* \| [s_k/\|s_k\|] - s^* \|. \tag{13}$$

Then, using (13) and recalling Lemma 2.1 and the fact that $\|s_k\| \to 0$ when $\Delta_k \to 0$, we obtain

$$f^\circ(x^*; s^*) \;\geq\; \limsup_{k \in L, k \to \infty} \frac{f(x_k + \|s_k\| s^*) - f(x_k)}{\|s_k\|} \;\geq\; 0,$$

which completes the proof. ■

We can now assemble a final convergence result, which can be classified as *global* in the sense of not asking the starting point to be close to a solution. Essentially we know that the Clarke generalized derivative is nonnegative along a limiting trust-region step but we also know from Property 2.1 that trust-region steps tend to a step along the negative of g_k, which in turn can be asked to cover densely the unit sphere in some asymptotic sense.

Theorem 2.3. *Assume that f is bounded from below. Let Assumption 2.1 hold. Let Algorithm* `Basic DFO-TRNS` *satisfy Property 2.1. Let x^* be any limit point of $\{x_k\}$ and $K \subseteq \{k : \Delta_{k+1} < \Delta_k\}$ be a subset of indices such that*

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

If the subsequence $\{g_k\}_K$ is dense in the unit sphere, then x^* is stationary for problem (1).

Proof: We proceed by contradiction and assume that x^* is not stationary for problem (1). Then we know that a direction $\bar{g}$ exists such that $\|\bar{g}\| = 1$ and

$$f^\circ(x^*; -\bar{g}) \;<\; 0. \tag{14}$$

Since $\{g_k\}_K$ is dense in the unit sphere, we can extract a subset of iteration indices, which we call again K, such that

$$\lim_{k\in K, k\to\infty} x_k = x^*, \tag{15}$$

$$\lim_{k\in K, k\to\infty} g_k = \bar{g}, \tag{16}$$

$$\lim_{k\in K, k\to\infty} \Delta_k = 0. \tag{17}$$

Property 2.1 assures that $s_k = -\Delta_k D_k g_k$ with $D_k \to I$. Hence, it results that

$$\lim_{k\in K, k\to\infty} \frac{s_k}{\|s_k\|} = -\bar{g}.$$

Then, by Lemma 2.2, we have that

$$f^\circ(x^*; -\bar{g}) \geq 0,$$

which contradicts (14), thus concluding the proof. ∎

Remark 2.1. *The above result is also true when a step of the form $s_k = -\Delta_k g_k$ is taken, totally ignoring the quadratic term of the model.*

3. A trust-region algorithm based on a nonsmooth random model

3.1. A nonsmooth random model. Now we would like to modify the basic trust-region algorithm of Section 2 in such a way that the nonsmoothness of the objective function is better handled. Taking inspiration from bundle methods[3], we will define a new nonsmooth random model to replace the smooth random one.

[3]Bundle methods were developed to solve nonsmooth convex problems. The idea behind those methods is to approximate the objective function by means of a suitable underestimator. The use of a piecewise linear function (called *cutting plane model*) to handle the minimization of the original nonsmooth function was first proposed in [7, 24]. The main drawbacks of the cutting plane approach are the possible unboundedness of the approximating models and the slow convergence of the method. In order to overcome those issues, a stabilizing quadratic term is usually included in the approximation, see, e.g., [17, 26, 28] and references therein. Bundle methods have been combined with trust-region ones [37]. Other interesting approaches are tilted bundle methods [27], level bundle methods [32], bundle Newton methods [33], and generalized bundle methods [15]. When dealing with nonconvex problems, the model is not an underestimator anymore. Hence bundle methods need to be suitably modified in order to handle nonconvexity. Strategies like subgradient deletion rules and subgradient locality measures are usually implemented in order to avoid the difficulties caused by nonconvex functions (see, e.g., [35] for further details).

In a smooth setting, it is well-known that the trust-region model is given by the sum of a linear term and a quadratic one, namely

$$\bar{m}_k(s) + \frac{1}{2}s^\top B_k s \;=\; f(x_k) + \nabla f(x_k)^\top s + \frac{1}{2}s^\top B_k s.$$

A reasonable choice when considering nonsmoothness would be to replace the linear term $\bar{m}_k(s)$ by the following nonsmooth term

$$\bar{m}_k(s) \;=\; f(x_k) + f^\circ(x_k; s)$$

that is

$$\bar{m}_k(s) \;=\; \max_{\xi \in \partial f(x_k)} \left\{ f(x_k) + \xi^\top s \right\}. \tag{18}$$

Obviously, since the set $\partial f(x_k)$ is unknown, the above model cannot be used in practice.

Bundle methods overcome this difficulty by exploiting the information obtained on a set of points $\{y^j : j \in J_k\}$ approaching x_k, where J_k is an index set. In the case of convex optimization (f convex), these methods make approximations of the model (18) given by

$$\bar{m}_k(s) \;=\; \max_{j \in J_k} \left\{ f(y^j) + (\xi^j)^\top (x_k + s - y^j) \right\} \tag{19}$$

where $\xi^j \in \partial f(y^j), j \in J_k$. The model (19) is usually rewritten as

$$\bar{m}_k(s) \;=\; \max_{j \in J_k} \left\{ f(x_k) + (\xi^j)^\top s - \beta_k^j \right\}$$

where

$$\beta_k^j \;=\; f(x_k) + (\xi^j)^\top (y^j - x_k) - f(y^j)$$

represents the displacement related to the point y^j. This approach can be adapted to the nonconvex case by suitably modifying the expression of β_k^j in the following way:

$$\beta_k^j \;=\; \max\left\{0, f(x_k) - f(y^j) + (\xi^j)^\top (y^j - x_k) + \delta, \|y^j - x_k\|^2\right\},$$

where $\delta > 0$ is a parameter to be selected in the algorithm.

In a derivative-free context, we cannot even compute an element $\xi \in \partial f(y)$ for any sample point y. Hence, we need to somehow adapt the bundle approach to our derivative-free setting. The choice we made in this paper is to replace the information given by the subgradients ξ^j with the one obtained for a set of randomly generated normalized directions

$$G_k \;=\; \{g_i : \|g_i\| = 1, i \in I_k\},$$

where I_k is another index set. We first compute for each $(i, j) \in I_k \times J_k$, the displacements

$$\beta_k^{ij} \;=\; \max\Big\{0, f(x_k) - f(y_k^j) + (g_i)^\top (y_k^j - x_k) + \delta\|y_k^j - x_k\|^2\Big\} \tag{20}$$

Then it is possible to introduce the following model

$$\bar{m}_k(s) \;=\; \max_{i \in I_k}\Big\{ f(x_k) + (g_i)^\top s - \bar{\beta}_k^i \Big\} \tag{21}$$

where

$$\bar{\beta}_k^i \;=\; \max_{j \in J_k}\{\beta_k^{ij}\}. \tag{22}$$

Hence, while in bundle methods one selects a set of auxiliary points y^j and vectors $\xi^j \in \partial f(y^j)$, $j \in J_k$, to linearize the function and to somehow build an approximation of the subdifferential at x_k, in our derivative-free framework, since the elements of the subdifferential cannot be calculated, we randomly generate vectors g_i (say one per iteration) and build a linear term using a suitably chosen point y^{l_i} in our sample set (see (23) below). The rationale behind this strategy is that a direction g_i can be seen as an approximation of an element in $\partial f(y^{l_i})$, with y^{l_i} the point corresponding to the index giving the maximum displacement $\bar{\beta}_k^i$:

$$l_i \;\in\; \operatorname*{argmax}_{j \in J_k}\{\beta_k^{ij}\}. \tag{23}$$

An example can be seen in Figure 1. (For simplicity, in the figures, the scalar $\bar{\beta}_k^i$ is computed by setting $\delta = 0$. Also, in order to depict easily understandable examples in these figures, we used g_i's with absolute values different from 1.) Of course, when $\bar{\beta}_k^i = 0$ we might have that the direction g_i is a good approximation of an element of $\partial f(x_k)$ (like, e.g., the case of the line corresponding to g_3 in Figure 1). Summarizing, for each direction g_i, $i \in I_k$, we consider a linear term passing through a point y^{l_i} with $\partial f(y^{l_i})$ hopefully containing a subgradient close to the direction g_i. Then, according to (21), the model $\bar{m}_k(s)$ is the maximum of those linear functions over g_i, $i \in I_k$, see

Figure 2. An example for the nonconvex case is shown in Figures 3–4 (and here we set $\delta > 0$ when calculating the displacements).

In order to give some priority to the role of the new direction (generated at iteration k) in the proposed local model of the objective function, we perturb the parameters $\bar{\beta}_k^i$ in the following way:

$$\begin{aligned} \tilde{\beta}_k^i &= \bar{\beta}_k^i + \Delta_k^{1/2}, \qquad i \in I_k,\ i \neq k, && (24)\\ \tilde{\beta}_k^k &= \bar{\beta}_k^k. && (25) \end{aligned}$$

The modified displacements $\tilde{\beta}_k^i$, $i \in I_k$, are used to somehow penalize the linear terms corresponding to the directions $\{g_i : i \in I_k, i \neq k\}$ in the max function of the model (21).

Finally, the complete nonsmooth approximating model that we propose is the following:

$$m_k(s) \;=\; \max_{i \in I_k} \left\{ f(x_k) + (g_i)^\top s - \tilde{\beta}_k^i \right\} + \frac{1}{2} s^\top B_k s, \tag{26}$$

where $\tilde{\beta}_k^i$, $i \in I_k$, are defined according to (24) and (25), and B_k is a symmetric matrix built out from interpolation or regression on a sample set of points.

3.2. A trust-region method. The new trust-region subproblem we propose to solve at each iteration is then

$$\begin{array}{ll} \min & m_k(s) \\ \text{s.t.} & \|s\|^2 \;\leq\; \Delta_k^2, \end{array} \tag{27}$$

where $m_k(s)$ is given by (26), which can then be equivalently stated as

$$\begin{array}{ll} \displaystyle\min_{s,\alpha} & \frac{1}{2} s^\top B_k s + \alpha \\ \text{s.t.} & (f(x_k) - \tilde{\beta}_k^i) + (g_i)^\top s \;\leq\; \alpha, \quad \forall\, i \in I_k, \\ & \|s\|^2 \;\leq\; \Delta_k^2. \end{array} \tag{28}$$

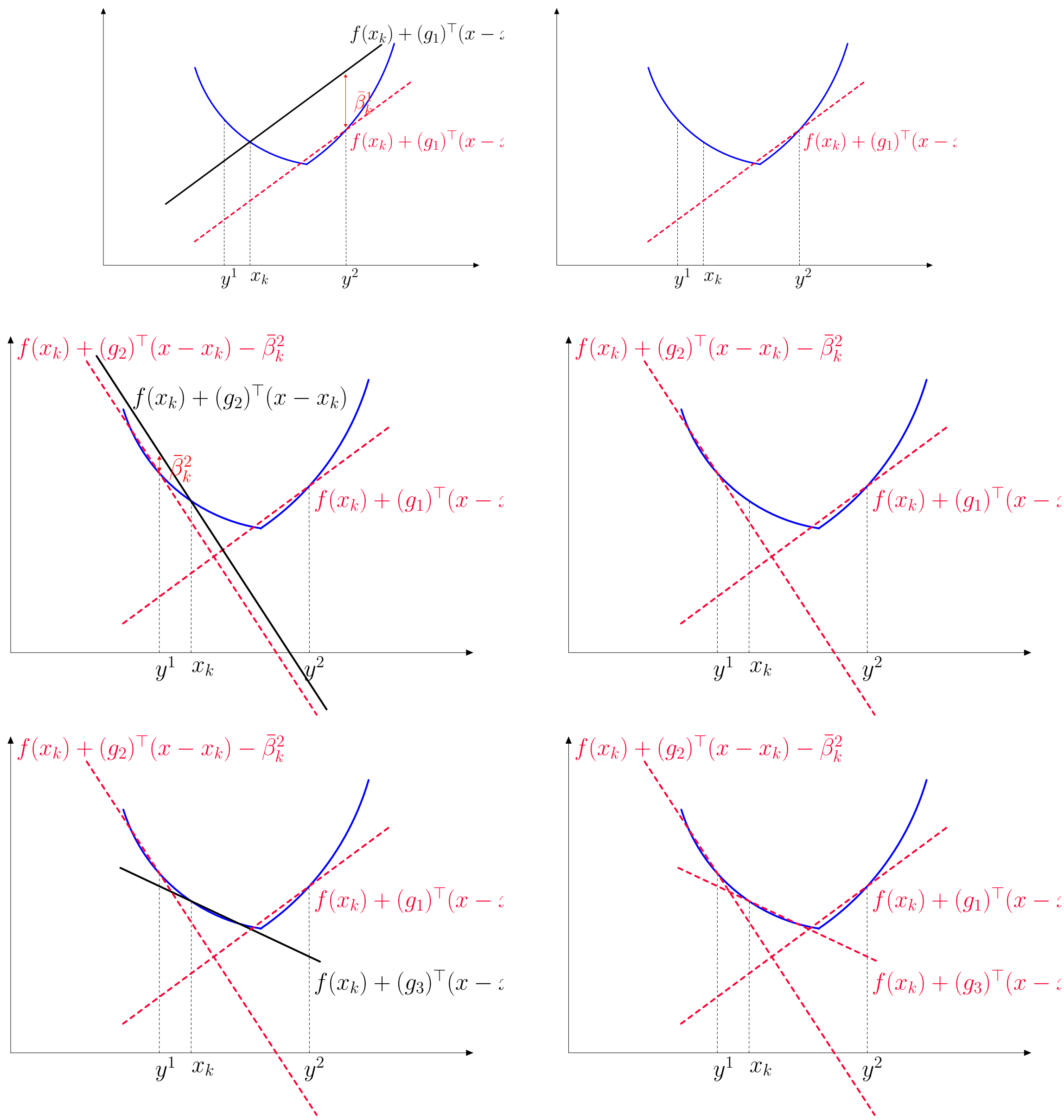

FIGURE 1. Construction of the nonsmooth model (convex case).

The first order necessary conditions for problem (28) require the existence of nonnegative Lagrange multipliers λ and σ such that

$$0 = B_k s + \sum_{i \in I_k} \lambda_i g_i + 2\sigma s \tag{29a}$$

$$0 = 1 - \sum_{i \in I_k} \lambda_i \tag{29b}$$

$$0 = \lambda_i \left((\tilde{\beta}_k^i - f(x_k)) - (g_i)^\top s + \alpha \right), \quad \forall\, i \in I_k \tag{29c}$$

$$0 \leq (\tilde{\beta}_k^i - f(x_k)) - (g_i)^\top s + \alpha, \quad \forall\, i \in I_k \tag{29d}$$

$$0 = \sigma(\|s\|^2 - \Delta_k^2), \quad \|s\|^2 \leq \Delta_k^2. \tag{29e}$$

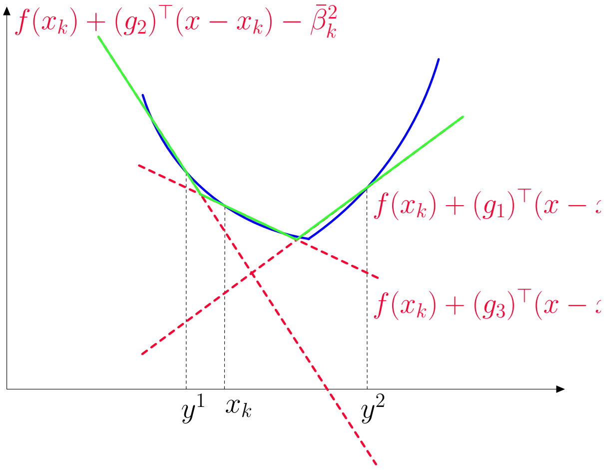

FIGURE 2. Nonsmooth model $\bar{m}_k(s)$ for the convex case (in green).

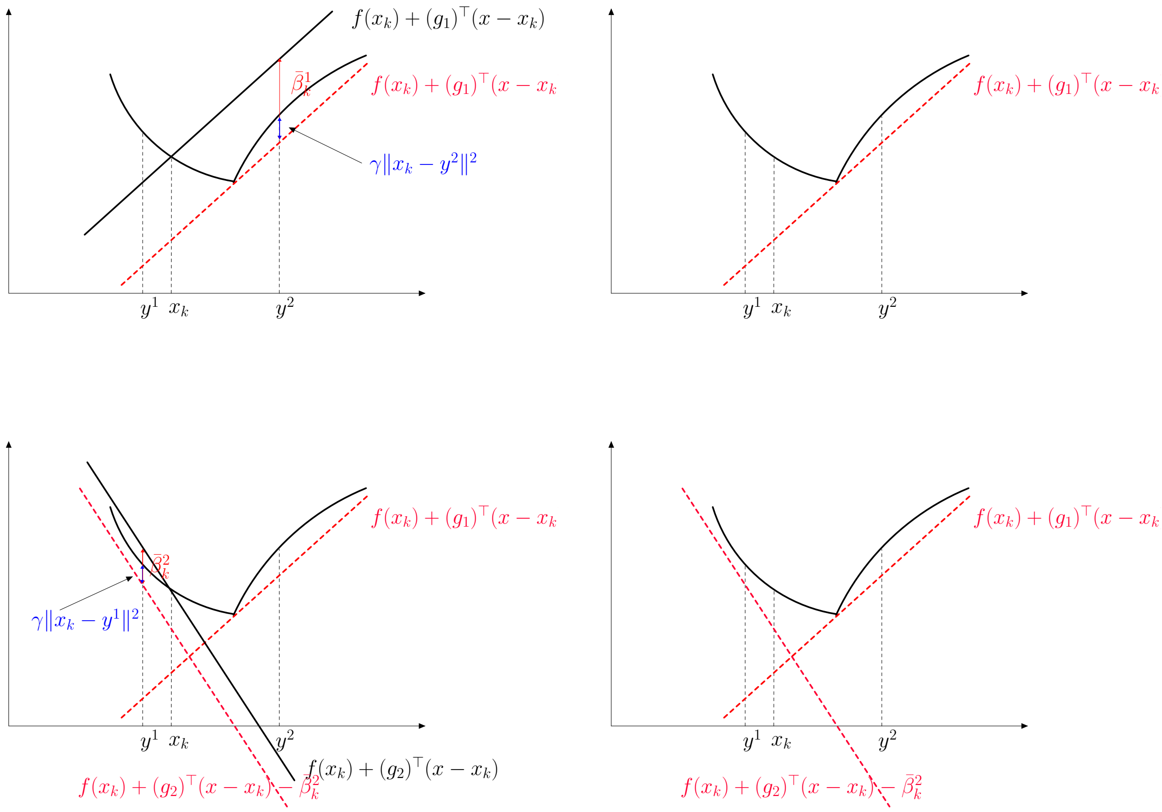

FIGURE 3. Construction of the nonsmooth model (nonconvex case).

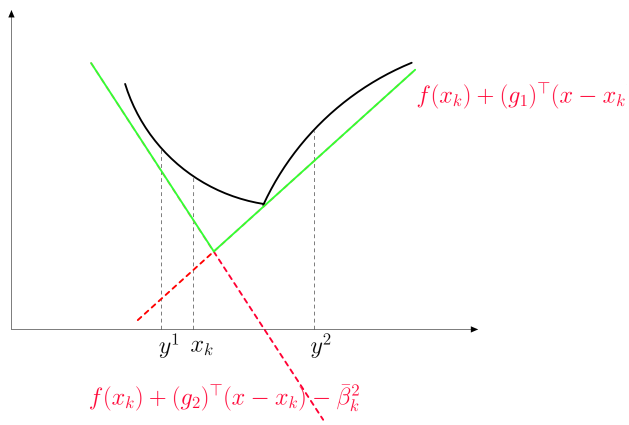

FIGURE 4. Nonsmooth model $\bar{m}_k(s)$ for the nonconvex case (in green).

Given a solution s_k of problem (28), along with its associated multipliers λ, we further consider the following auxiliary subproblem

$$\begin{aligned} \min \quad & \tilde{m}_k(s) = f(x_k) + \tilde{g}_k^\top s + \frac{1}{2} s^\top B_k s \\ \text{s.t.} \quad & \|s\|^2 \leq \Delta_k^2, \end{aligned} \tag{30}$$

where

$$\tilde{g}_k = \sum_{i \in I_k} \lambda_i g_i \quad \text{with} \quad \sum_{i \in I_k} \lambda_i = 1, \quad \text{and} \quad \lambda_i \geq 0, \ \forall i \in I_k, \tag{31}$$

is a convex linear span of the randomly generated vectors. The first order necessary conditions (29) show that a solution s_k of problem (28) satisfies the first order necessary conditions (3) with g_k replaced by $\tilde{g}_k$, thus satisfying the first order necessary conditions for (30). In addition, if s_k solves problem (28), then from the second order necessary conditions, we know that

$$\begin{bmatrix} d_s \\ d_\sigma \end{bmatrix}^\top \begin{bmatrix} B_k + 2\sigma & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{bmatrix} \begin{bmatrix} d_s \\ d_\sigma \end{bmatrix} \geq 0$$

for all (d_s, d_σ) in the cone of these conditions. Furthermore, if we look at the inequalities or equalities that define this cone *and* are associated with the constraints indexed by I_k, they are either of the form $(g_i)^\top d_s - d_\alpha \leq 0$ or $(g_i)^\top d_s - d_\alpha = 0$ which do not constrain d_s. Hence, we have that $d_s^\top (B_k + 2\sigma I) d_s \geq 0$ for all d_s in the cone of the second order necessary conditions associated *only* with the constraint $\|s\|^2 \leq \Delta_k^2$. The conclusion is that s_k also satisfies the second order necessary conditions for (30) with g_k replaced by $\tilde{g}_k$. Since the first and second order necessary conditions are sufficient for any trust-region subproblem with a spherical trust-region constraint (regardless of the sign of eigenvalues of B_k), we conclude that a solution of (28) is a solution of (30). This fact has repercussions in our algorithmic design and convergence analysis.

We now can define a modified version of the `Basic DFO-TRNS` Algorithm using this more sophisticated model that will enable us to better handle nonsmoothness in the objective function. The detailed scheme is reported as Algorithm `Advanced DFO-TRNS`. We need to consider that the points in the sample set $\{y_k^j : \; j \in J_k\}$ used to build the linear terms of the model verify

$$\|y_k^j - x_k\| \;\leq\; \gamma\Delta_k, \quad \forall j \in J_k, \tag{32}$$

for all iterations k of our algorithm, and for some suitably chosen $\gamma > 0$. Furthermore, the advanced algorithm reverts to the basic (in the sense of using a quadratic model) when the norm of the convex linear span vector $\tilde{g}_k$ in (31) becomes too small (relative to the trust-region radius). For simplicity the condition we test is $\|\tilde{g}_k\| < \bar{\epsilon}\Delta_k^{1/2}$ but one could use any power of Δ_k with exponent in $(0, 1)$ (see the comment below made before Assumption 3.1).

3.3. Convergence analysis. The convergence analysis of the advanced algorithm follows the same lines as the basic version. The main difference is the use of the convex linear span vector $\tilde{g}_k$, see (31), instead of the normalized g_k. However, note that from the logic of the algorithm, $\tilde{g}_k$ can be bounded below as follows:

$$\|\tilde{g}_k\| \;\geq\; \min\{1, \bar{\epsilon}\Delta_k^{\frac{1}{2}}\}. \tag{33}$$

A similar assumption as in the basic algorithm is imposed in the model Hessians. In this case the exponent q is restricted to $(0, 1/2)$. This has to do with the test $\|\tilde{g}_k\| < \bar{\epsilon}\Delta_k^{1/2}$ in the advanced algorithm. If we had considered a parameter $r \in (0, 1)$ and asked instead for $\|\tilde{g}_k\| < \bar{\epsilon}\Delta_k^{r}$, then the exponent q in the assumption below would had been restricted to $(0, 1 - r)$.

Assumption 3.1. *There exist $q \in (0, 1/2)$, $m, M > 0$, such that: The maximal eigenvalue of B_k satisfies*

$$\lambda_{\max}(B_k) \;\leq\; M\Delta_k^{-q}.$$

Algorithm `Advanced DFO-TRNS` `(Advanced DFO Trust-Region Algorithm for Nonsmooth Problems)`

Initialization. Select $x_0 \in \mathbb{R}^n$, $\eta_1, \theta, \gamma > 0$, $0 < \gamma_1 < 1 \leq \gamma_2$, $\bar{\epsilon} > 0$, $\Delta_0 > 0$, and $p > 0$. Set $G_0 = \emptyset$.

For $k = 0, 1 \ldots$

Generate g_k randomly and densely on the unit sphere. Consider a sample set of points
satisfying (32). Build a symmetric matrix B_k.

Set $G_k = G_{k-1} \cup \{g_k\}$.

Let s be a solution for subproblem (28) for this G_k, and λ the associate multipliers.

Let $\tilde{g}_k = \sum_{i \in I_k} \lambda_i g_i$, where I_k is the index set corresponding to G_k.

If $\|\tilde{g}_k\| < \bar{\epsilon} \Delta_k^{\frac{1}{2}}$ **Then**

Reset $G_k = \{g_k\}$.

Let s be a solution for subproblem (28) for this reset G_k.

End If

Set $s_k = s$ and

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{\theta \|s_k\|^{p+1}}.$$

If $\rho_k \geq \eta_1$ **Then** set `SUCCESS` $\leftarrow$ `true`, $x_{k+1} \leftarrow x_k + s_k$, $\Delta_{k+1} \leftarrow \gamma_2 \Delta_k$,

Else set `SUCCESS` $\leftarrow$ `false`, $x_{k+1} \leftarrow x_k$, $\Delta_{k+1} \leftarrow \gamma_1 \Delta_k$.

End If

End For

When B_k has negative eigenvalues, its minimal eigenvalue satisfies

$$-\lambda_{\min}(B_k) \leq m \Delta_k^{-q}.$$

Again, one first proves that the trust-region radius converges to zero.

Lemma 3.1. *Assume that f is bounded from below. Let Assumption 3.1 hold. Any sequence $\{\Delta_k\}$ of trust-region radii produced by Algorithm* `Advanced DFO-TRNS` *is such that*

$$\lim_{k \to \infty} \Delta_k = 0.$$

Proof: The proof is similar to the one of Lemma 2.1, and the only difference is the use of $\tilde{g}_k$ instead of g_k. We thus only need to redo the algebraic part of the proof:

$$\begin{aligned} f(x_k) - f(x_k + s_k) &\geq \eta_1 \theta \|s_k\|^{1+p} \\ &\geq \eta_1 \theta \min\{\|B_k^{-1} \tilde{g}_k\|, \Delta_k\}^{1+p} \\ &\geq \eta_1 \theta \min\left\{\frac{\epsilon^q}{M}\|\tilde{g}_k\|, \epsilon\right\}^{1+p} \\ &\geq \eta_1 \theta \min\left\{\frac{\epsilon^q}{M}\min\{1, \bar{\epsilon}\epsilon^{\frac{1}{2}}\}, \epsilon\right\}^{1+p}, \end{aligned}$$

where the last inequality follows from (33). ■

Then we show that the algorithm generates steps that tend to a step that is now along the convex linear span vector $\tilde{g}_k$.

Property 3.1. *Any sequence* $\{(x_k, s_k, \Delta_k)\}$ *generated by Algorithm* `Advanced DFO-TRNS` *is such that*

$$s_k \;=\; -\Delta_k D_k \frac{\tilde{g}_k}{\|\tilde{g}_k\|},$$

with $D_k \in \mathbb{R}^{n\times n}$ *satisfying*

$$\lim_{k\to\infty} D_k \;=\; I.$$

As in Proposition 2.1, the following result is proved under exact optimality of the solution of the trust-region subproblem (27), but we could have assumed inexact optimality and deduce a similar result as long as the norm of the residual of the first-order necessary conditions of this subproblem is smaller than a certain power of the trust-region radius.

Proposition 3.1. *Assume that f is bounded from below. Let Assumption 3.1 hold. Assume also that all trust-region subproblems* (27) *are solved up to optimality. Then Algorithm* `Advanced DFO-TRNS` *generates sequences* $\{(x_k, s_k, \Delta_k)\}$ *satisfying Property 3.1 (for k sufficiently large).*

Proof: The proof follows the line of thought of Proposition 2.1. The main difference is that the step s_k solves now the modified trust-region subproblem (30), where $\tilde{g}_k$ takes the place of g_k. The same calculations take us to

$$\left(B_k + \frac{\eta_k}{\Delta_k^2} I\right) s_k \;=\; -\tilde{g}_k, \tag{34}$$

where now $\eta_k = -s_k^\top B_k s_k - \tilde{g}_k^\top s_k$. As in Proposition 2.1, the rest of the proof is divided in two main parts.

(A) When deriving the lower and upper bounds on η_k we can no longer use the fact that $\tilde{g}_k$ is normalized but rather that it satisfies the bound (33).

Since s_k solves problem (30), it yields a better model value than the step $-\Delta_k \tilde{g}_k / \|\tilde{g}_k\|$, which means

$$\frac{1}{2} s_k^\top B_k s_k + \tilde{g}_k^\top s_k \;\leq\; \frac{1}{2} \Delta_k^2 \tilde{g}_k^\top B_k \tilde{g}_k / \|\tilde{g}_k\|^2 - \Delta_k \tilde{g}_k^\top \tilde{g}_k / \|\tilde{g}_k\|,$$

and we obtain

$$\frac{1}{2} s_k^\top B_k s_k + \tilde{g}_k^\top s_k \;\leq\; \frac{1}{2} \Delta_k^2 \lambda_{\max}(B_k) - \Delta_k \|\tilde{g}_k\|. \tag{35}$$

Then, taking into account the fact that $\eta_k = -2(\frac{1}{2} s_k^\top B_k s_k + \tilde{g}_k^\top s_k) + \tilde{g}_k^\top s_k$ and using inequality (35), we can derive a lower bound on η_k from Assumption 3.1

$$\begin{aligned} \eta_k &\geq -\Delta_k^2 \lambda_{\max}(B_k) + 2\Delta_k \|\tilde{g}_k\| + \tilde{g}_k^\top s_k \\ &\geq -M \Delta_k^{2-q} + 2\Delta_k \|\tilde{g}_k\| - \Delta_k \|\tilde{g}_k\| \\ &\geq -M \Delta_k^{2-q} + \Delta_k \|\tilde{g}_k\|. \end{aligned} \tag{36}$$

The bound (36) ensures that η_k is nonzero for k sufficiently large.

On the other hand, an upper bound on η_k can be derived also from Assumption 3.1 as follows

$$\begin{aligned} \eta_k &= -s_k^\top B_k s_k - \tilde{g}_k^\top s_k \\ &\leq \max\{0, -\lambda_{\min}(B_k)\} \|s_k\|^2 + \Delta_k \|\tilde{g}_k\| \\ &\leq m \Delta_k^{-q} \Delta_k^2 + \Delta_k \|\tilde{g}_k\|. \end{aligned} \tag{37}$$

(B) The application of the generalized Sherman-Morrison-Woodbury formula,

$$\left(B_k + \frac{\eta_k}{\Delta_k^2} I \right)^{-1} \;=\; \frac{\Delta_k^2}{\eta_k} I - \frac{\Delta_k^2}{\eta_k} B_k \left(I + \frac{\Delta_k^2}{\eta_k} B_k \right)^{-1} \frac{\Delta_k^2}{\eta_k} I,$$

and (34) allow us to write

$$s_k \;=\; -\Delta_k D_k \frac{\tilde{g}_k}{\|\tilde{g}_k\|},$$

with

$$D_k \;=\; \frac{\|\tilde{g}_k\| \Delta_k}{\eta_k} I - \frac{\|\tilde{g}_k\| \Delta_k}{\eta_k} B_k \left(I + \frac{\Delta_k^2}{\eta_k} B_k \right)^{-1} \frac{\Delta_k^2}{\eta_k} I. \tag{38}$$

It remains to prove that $D_k \to I$ as $k \to \infty$. Using the bounds (36) and (37) derived in part (A) of the proof and dividing by $\Delta_k\|\tilde{g}_k\|$ we arrive at

$$-\frac{M\Delta_k^{1-q}}{\|\tilde{g}_k\|}+1 \;\leq\; \frac{\eta_k}{\Delta_k\|\tilde{g}_k\|} \;\leq\; 1+\frac{m\Delta_k^{1-q}}{\|\tilde{g}_k\|}.$$

From this, Lemma 3.1, (33), and $q \in (0, 1/2)$,

$$\frac{\eta_k}{\Delta_k\|\tilde{g}_k\|} \;\to\; 1.$$

From this and Assumption 3.1, and again using (33) and $q \in (0, 1/2)$,

$$\frac{\Delta_k^2}{\eta_k}B_k \;=\; \frac{\Delta_k\|\tilde{g}_k\|}{\eta_k}\frac{B_k\Delta_k}{\|\tilde{g}_k\|} \;\to\; 0.$$

Finally taking into account (38), it results that $D_k \to I$ as $k \to \infty$, and the proof is complete. ■

The proof that the Clarke generalized derivative is nonnegative along limiting trust-region steps is verbatim the one of Lemma 2.2.

Lemma 3.2. *Assume that f is bounded from below. Let Assumption 3.1 hold. Let $\{(x_k, s_k, \Delta_k)\}$ be a sequence generated by Algorithm* `Advanced DFO-TRNS`*. Let $L \subseteq K = \{k : \Delta_{k+1} < \Delta_k\}$ be an index set such that*

$$\lim_{k\in L, k\to\infty} x_k \;=\; x^* \quad \textit{and}$$

$$\lim_{k\in L, k\to\infty} \frac{s_k}{\|s_k\|} \;=\; s^*.$$

Then $f^\circ(x^*; s^*) \geq 0$.

The final global convergence is given below. Its proof follows the main argument of the proof of Theorem 2.3.

Theorem 3.3. *Assume that f is bounded from below. Let Assumption 3.1 hold. Let Algorithm* `Advanced DFO-TRNS` *satisfy Property 3.1. Let x^* be any limit point of $\{x_k\}$ and $K \subseteq \{k : \Delta_{k+1} < \Delta_k\}$ be a subset of indices such that*

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

If the subsequence $\{g_k\}_K$ is dense in the unit sphere, then x^* is stationary for problem (1).

Proof: We proceed by contradiction and assume that x^* is not stationary for problem (1). Then we know that a direction $\bar{g}$ exists such that $\|\bar{g}\| = 1$ and $f^\circ(x^*; -\bar{g}) < 0$. Since $\{g_k\}_K$ is dense in the unit sphere, we can extract a subset of iteration indices, which we call again K, such that (15)–(17) hold. Property 3.1 assures that $s_k = -\Delta_k D_k \tilde{g}_k / \|\tilde{g}_k\|$ with $D_k \to I$.

Considering the definitions of $\tilde{\beta}_k^i$ given in (24) and (25), we can write the expression of the constraints in model (28) as follows:

$$(g_i)^\top s - \bar{\beta}_k^i - \Delta_k^{1/2} \le \alpha - f(x_k), \quad \text{for all } i \in I_k,\ g_i \neq g_k, \tag{39}$$

$$(g_k)^\top s - \bar{\beta}_k^k \qquad\quad \le \alpha - f(x_k). \tag{40}$$

Now, taking into account (20) and (22), the Lipschitz continuity of function f, and the inequalities given in (32), we conclude that there exists a constant $c > 0$ such that

$$\bar{\beta}_k^i \ \le\ c\Delta_k, \quad \forall i \in I_k.$$

This inequality, $\bar{\beta}_k^i \ge 0$, and the constraint $\|s\| \le \Delta_k$ allow us to give, for k sufficiently large, an upper bound to the left hand side of constraints (39) and a lower bound to the left hand side of constraint (40). In particular, for k sufficiently large, there exist $0 < \theta_1 < \theta_2 < 1$ such that (recall that each g_i has norm 1)

$$\begin{aligned} (g_i)^\top s - \bar{\beta}_k^i - \Delta_k^{1/2} &\le \ \Delta_k - \Delta_k^{1/2} \le -\theta_2 \Delta_k^{1/2}, && \text{for all } i \in I_k,\ g_i \neq g_k, \\ (g_k)^\top s - \bar{\beta}_k^k &\ge -(1+c)\Delta_k \qquad \ge -\theta_1 \Delta_k^{1/2}. && \end{aligned}$$

Thus, for k sufficiently large, we conclude that the constraints (39) are not active and that the only active constraint is the one related to g_k, namely constraint (40). This implies, by taking into account the necessary conditions in (29), that

$$\begin{aligned} \lambda_k^i &= 0, \qquad \text{for all } i \in I_k,\ i \neq k, \\ \lambda_k^k &= 1. \end{aligned}$$

We have thus $\tilde{g}_k = g_k$, for k sufficiently large, and Property 3.1 assures $s_k = -\Delta_k D_k g_k$ with $D_k \to I$, and hence

$$\lim_{k \in K, k \to \infty} \frac{s_k}{\|s_k\|} \ = \ -\bar{g}.$$

Then, by Proposition 3.1, we have that $f^\circ(x^*; -\bar{g}) \ge 0$, which is a contradiction, and the proof is concluded. ■

Remark 3.1. *It becomes evident from the proof of Theorem 3.3 that Property 3.1 is not really needed after all, but rather Property 2.1 for k sufficiently large. However, Property 3.1 is the foundational one for the advanced algorithm, and it allows us to consider scenarios where the density of the directions can be guaranteed for $\tilde{g}_k$ instead of for g_k.*

4. Implementation and numerical results

In this section we report numerical results obtained by our Basic and Advanced DFO-TRNS algorithms and compare them against the performance of NOMAD v 3.8.1 [1, 31], a state-of-the-art software for nonsmooth derivative-free optimization. NOMAD has been run using its default parameter settings. The performance of the different solvers was assessed on 51 well-known nonsmooth unconstrained problems with dimensions between 10 and 30 variables. The complete problem list with the corresponding references is reported in Table 1. We notice that our benchmark includes some composite nonsmooth problems with specific structure (e.g., the outer function is the max operator or the ℓ_1 norm) and more general nonsmooth problems constructed either by chaining and extending existing nonsmooth problems or by including some nonsmoothness in existing smooth problems (e.g., by replacing a variable x_i with $|x_i|$).

We ran all versions of the tested solvers giving a budget of 10000 function evaluations. Such a number of function evaluations seems a reasonable choice in practice given the nonsmoothness and the dimensions of the problems.

Performance of different derivative-free solvers on different problems can be analyzed using data and performance profiles [36]. Specifically, let S be a set of solvers and P a set of problems. For each $s \in S$ and $p \in P$, let $t_{p,s}$ be the number of function evaluations required by solver s on problem p to satisfy the condition

$$f(x_k) \;\leq\; f_L + \tau(f(x_0) - f_L),$$

where $0 < \tau < 1$ and f_L is the best objective function value achieved by any solver in S on problem p. Then, performance and data profiles of solver s are the following functions

$$\begin{aligned}
\rho_s(\alpha) &= \frac{1}{|P|}\left|\left\{p \in P : \frac{t_{p,s}}{\min\{t_{p,s'} : s' \in S\}} \leq \alpha\right\}\right|, \\
d_s(\kappa) &= \frac{1}{|P|}\left|\{p \in P : t_{p,s} \leq \kappa(n_p + 1)\}\right|,
\end{aligned}$$

name	dimension	reference
wong2	10	[34]
polak2	10	[34]
maxquad	10	[34]
gill	10	[34]
maxq	{10,20,30}	[23]
l1hilb	{10,20,30}	[23]
lq	{10,20,30}	[23]
cb3	{10,20,30}	[23]
cb32	{10,20,30}	[23]
af	{10,20,30}	[23]
brown	{10,20,30}	[23]
mifflin2	{10,20,30}	[23]
crescent	{10,20,30}	[23]
crescent2	{10,20,30}	[23]
polak3	11	[34]
osborne2	11	[34]
steiner2	12	[34]
shelldual	15	[34]
wong3	20	[34]
maxl	20	[34]
maxql	20	[34]
watson	20	[34]
wild1	20	[36]
wild2	20	[36]
wild3	20	[36]
wild19	20	[36]
wild11	20	[36]
wild16	20	[36]
wild20	20	[36]
wild15	20	[36]
wild21	20	[36]

TABLE 1. Problems used in the numerical experiments.

where n_p is the dimension of problem p. Comparisons were carried out for values of the tolerance parameter τ in $\{10^{-3}, 10^{-5}, 10^{-6}\}$.

The practical implementation of DFO-TRNS makes use of a quadratic term $\frac{1}{2}s^\top B_k s$. In turn, building B_k requires maintaining a sample set, say Y_k. Here we followed the smooth derivative-free trust-region approach of [5, Section 5], where B_k is built out of minimum Frobenius norm models using a sample set that starts by $2n+1$ points x_0, $x_0 + \Delta_0 e_i$, $x_0 - \Delta_0 e_i$, where e_i is the i-th column of the identity matrix of order n (even though the approach computes coefficients for both linear and quadratic basis terms, those ones related to the linear part are simply thrown away in our case).

At each iteration, the trial point $x_k + s_k$ is added to the sample set until it reaches a cardinality of $(n+1)(n+2)/2$, after which the sample point farthest away from the new iterate is discarded to give room to the new one.

It is this very same sample set that is normally used to build the max-linear terms of our nonsmooth trust-region models in the Advanced version of DFO-TRNS, except for the following modification. If the points in the sample set Y_k are too close to the current best point, more specifically, if the cardinality of the set

$$\left\{y \in Y_k : \ \|y - x_k\| \leq \tilde{\Delta}_k, \ \|y - x_k\| > 10^{-7}\right\},$$

with $\tilde{\Delta}_k = \min\{\Delta_k, 10\}$, is less than 2, we then define a completely new set

$$Y_k \ = \ \{y_i = x_k + d_i, \ i = 1, \ldots, p\} \cup \{x_k\},$$

where $p = \max\{3, n/3\}$ and $d_i \in \mathbb{R}^n$ are vectors with $\|d_i\| = \tilde{\Delta}_k/2$ generated by suitably scaling the vectors of the pseudorandom Halton sequence [20].

Moreover, in the practical implementation of our algorithms, we have included a weight $0 \leq \omega \leq 1$ in the quadratic term of the models, that is we replaced $\frac{1}{2}s^\top B_k s$ with $\frac{\omega}{2}s^\top B_k s$ in Problems (2) and (27). In such a way, the users of our solver can tune it to the degree of nonsmoothness of their problems. The other parameters of our method were set to the following values: $\eta_1 = 10^{-8}$, $\gamma_1 = 1/10$, $\gamma_2 = 10/9$, $\Delta_0 = 1$, $p = 0.1$, and $\theta = 10^{-3}$. Furthermore, in the displacements β_k^{ij} reported in (20), we used a value of $\delta = 10^{-5}$.

As a preliminary test, we ran Basic DFO-TRNS using models without the quadratic term (i.e., $\omega = 0$) and we naturally compared it against NOMAD with the option *disable models*. In this way we are comparing a trust-region approach that uses a linear model with random first-order term against a directional type direct-search method with random directions in the poll step (and no search step included). In Figure 5, we report performance and data profiles related to the comparison between Basic DFO-TRNS with $\omega = 0.0$ and NOMAD without models. As we can easily see, our Basic DFO-TRNS solver yields good results when compared to NOMAD without models, especially when the required tolerance is small enough.

In Figure 6, we report the profiles for Advanced DFO-TRNS when varying the value of the ω parameter in $\frac{\omega}{2}s^\top B_k s$. It can be observed that reducing the value of the parameter does not always improve the performance of the algorithm. The results are comparable but still the best result seems to

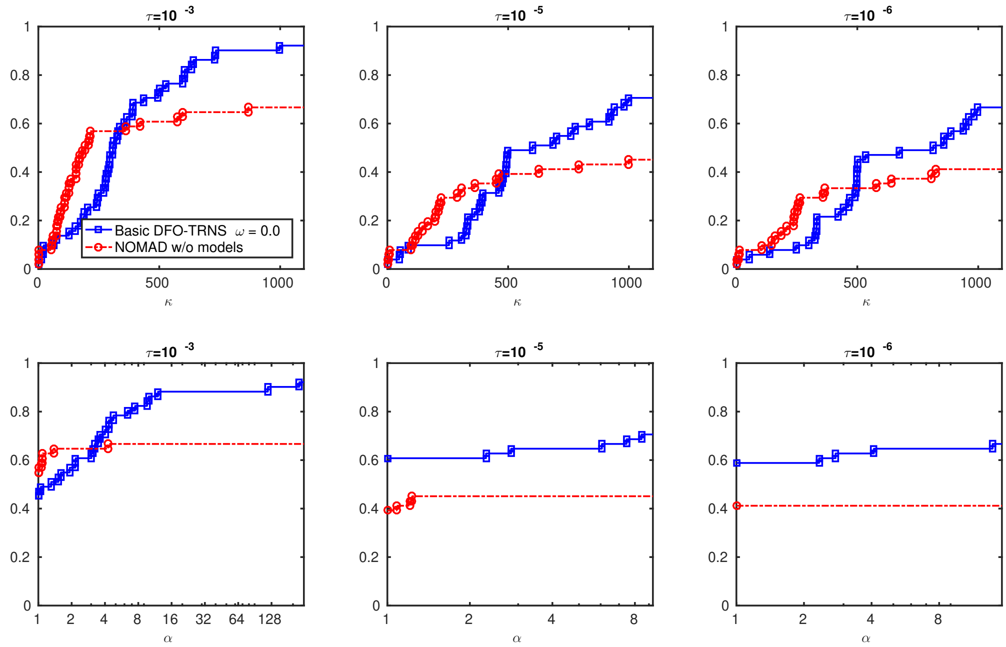

FIGURE 5. Comparison between Basic DFO-TRNS with $\omega = 0.0$ in $\frac{\omega}{2}s^\top B_k s$ and NOMAD without models.

be obtained when $\omega = 1$, indicating that the inclusion of a good level of smoothness in the models could lead to some improvement in performance.

Another experiment is reported to highlight the importance of incorporating a max-linear term in the trust-region models, i.e., the potential improvement obtained when passing from the Basic to the Advanced DFO-TRNS algorithm. We report in Figure 7 a comparison of these two versions for $\omega = 1$ from where this improvement is clearly visible.

Finally, we hence compare, on Figure 8, Advanced DFO-TRNS with $\omega = 1$ and NOMAD when this includes a search step consisting of the minimization of quadratic models. The results seem to indicate that our trust-region approach is again competitive with NOMAD, especially when the required tolerance is small enough.

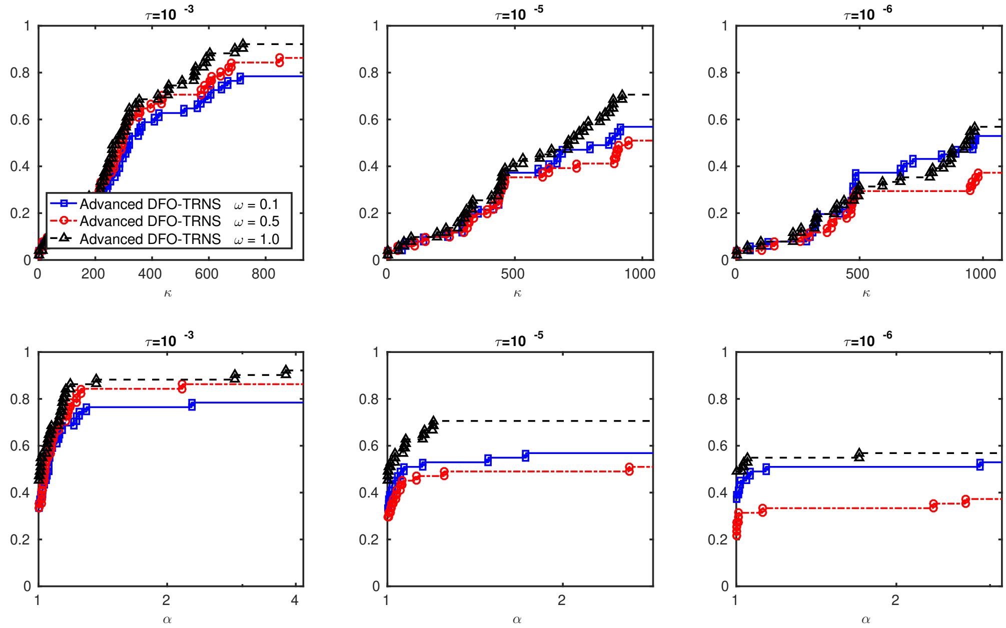

FIGURE 6. Comparison of Advanced DFO-TRNS versions with different ω in $\frac{\omega}{2} s^\top B_k s$.

5. Conclusions

In this paper it was developed for the first time a convergent trust-region methodology for nonsmooth derivative-free optimization, when no information whatsoever is available about the origin of the nonsmoothness of the objective function. The trust-region models considered are nonsmooth, of the max-linear type, where each linear term in the max-linear model attempts to approximate an element of the subdifferential in a nearby sampling point. A quadratic term can also be considered in the trust-region model for the purpose of improving numerical performance. Interestingly, we have shown that the Hessian matrix of this quadratic term can be unbounded as long as it does not go to infinity faster than the inverse of a negative power of the trust-region radius.

A number of open future questions deserve attention. The most pressing issue is the development of a derivative-free trust-region solver capable of dealing well with large problems. In fact, quadratic interpolation/regression models or similar require significant storage and linear algebra effort as the

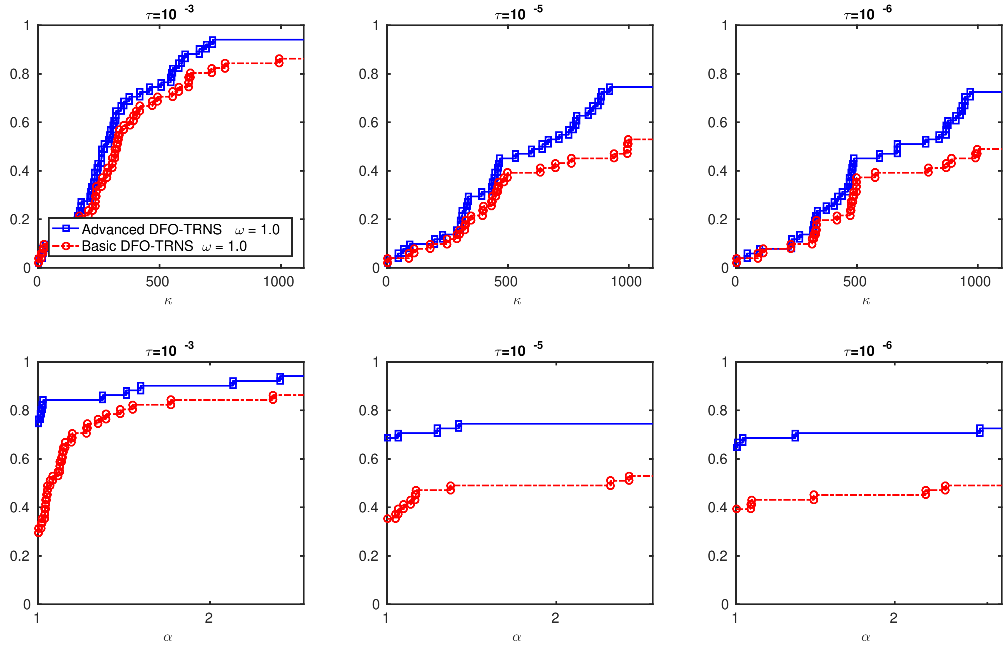

FIGURE 7. Comparison between Basic and Advanced DFO-TRNS with $\omega = 1$ in $\frac{\omega}{2} s^\top B_k s$.

model Hessians are typically dense. Other generalizations include the extension to the constrained case, in particular to linear constraints, and to stochastic objective functions. But there are also some interesting open questions related to the analysis presented in this paper. It would be pertinent to study what can be said when the objective function is itself of the max-linear type. Another question is related to the development of a convergence rate, which does not seem likely to be observed when the directions are densely generated in the unit sphere, but still it would be interesting to derive some quantitative probabilistic argument on the rate of progress.

References

[1] M. A. Abramson, C. Audet, G. Couture, J. E. Dennis, Jr., S. Le Digabel, and C. Tribes. The NOMAD project. Software available at `https://www.gerad.ca/nomad/`.

[2] C. Audet and W. hare. Model-based methods in derivative-free nonsmooth optimization. Technical Report G-2018-34, Les Cahiers du GERAD, HEC Montréal, 2018.

[3] C. Audet and J. E. Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. *SIAM J. Optim.*, 17:188–217, 2006.

[4] A. M. Bagirov, B. Karasözen, and M. Sezer. Discrete gradient method: Derivative-free method for nonsmooth optimization. *J. Optim. Theory Appl.*, 137:317–334, 2008.

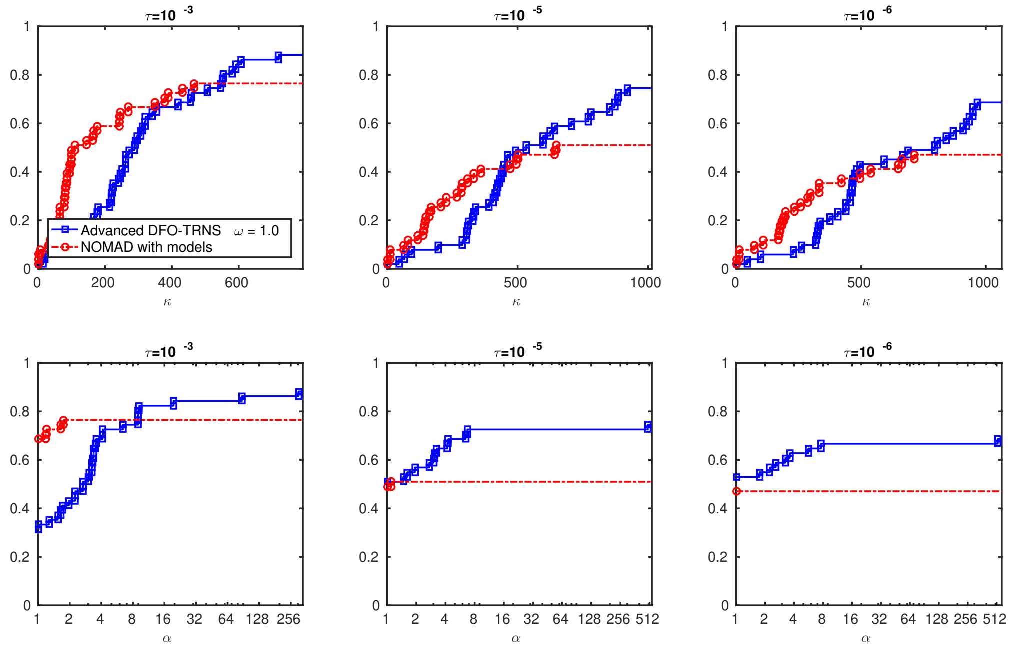

FIGURE 8. Comparison between Advanced DFO-TRNS with $\omega = 1$ in $\frac{\omega}{2} s^\top B_k s$ and NOMAD with models.

[5] A. S. Bandeira, K. Scheinberg, and L. N. Vicente. Computation of sparse low degree interpolating polynomials and their application to derivative-free optimization. *Math. Program.*, 134:223–257, 2012.
[6] A. S. Bandeira, K. Scheinberg, and L. N. Vicente. Convergence of trust-region methods based on probabilistic models. *SIAM J. Optim.*, 24:1238–1264, 2014.
[7] E. W. Cheney and A. A. Goldstein. Newton's method for convex programming and Tchebycheff approximation. *Numer. Math.*, 1:253–268, 1959.
[8] F. H. Clarke. *Optimization and Nonsmooth Analysis.* John Wiley & Sons, New York, 1983. Reissued by SIAM, Philadelphia, 1990.
[9] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust-Region Methods.* MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2000.
[10] A. R. Conn, K. Scheinberg, and L. N. Vicente. Global convergence of general derivative-free trust-region algorithms to first and second order critical points. *SIAM J. Optim.*, 20:387–415, 2009.
[11] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-Free Optimization.* MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2009.
[12] F. E. Curtis, D. P. Robinson, and M. Samadi. A trust region algorithm with a worst-case iteration complexity of $o(\epsilon^{3/2})$ for nonconvex optimization. *Math. Program.*, 162:1–32, 2017.
[13] J. E. Dennis Jr. and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice–Hall, Englewood Cliffs, (republished by SIAM, Philadelphia, in 1996, as Classics in Applied Mathematics, 16), 1983.

[14] G. Fasano, G. Liuzzi, S. Lucidi, and F. Rinaldi. A linesearch-based derivative-free approach for nonsmooth constrained optimization. *SIAM J. Optim.*, 24:959–992, 2014.
[15] A. Frangioni. Generalized bundle methods. *SIAM J. Optim.*, 13:117–156, 2002.
[16] R. Garmanjani, D. Júdice, and L. N. Vicente. Trust-region methods without using derivatives: Worst case complexity and the non-smooth case. *SIAM J. Optim.*, 26:1987–2011, 2016.
[17] M. Gaudioso and M. F. Monaco. A bundle type approach to the unconstrained minimization of convex nonsmooth functions. *Math. Program.*, 23:216–226, 1982.
[18] G. N. Grapiglia, J. Yuan, and Y. Yuan. A derivative-free trust-region algorithm for composite nonsmooth optimization. *Comp. Appl. Math.*, 35:475–499, 2016.
[19] S. Gratton, C. W. Royer, L. N. Vicente, and Z. Zhang. Complexity and global rates of trust-region methods based on probabilistic models. *IMA J. Numer. Anal.*, 38:1579–1597, 2018.
[20] J. H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numer. Math.*, 2:84–90, 1960.
[21] W. Hare and J. Nutini. A derivative-free approximate gradient sampling algorithm for finite minimax problems. *Comput. Optim. Appl.*, 56:1–38, 2013.
[22] N. Hoseini and S. Nobakhtian. A new trust region method for nonsmooth nonconvex optimization. *Optimization*, 67:1265–1286, 2018.
[23] N. Karmitsa. Test problems for large-scale nonsmooth minimization. *Reports of the Department of Mathematical Information Technology. Series B, Scientific computing*, 2007.
[24] J. E. Kelley. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8:703–712, 1960.
[25] K. Khan, J. Larson, and S. M. Wild. Manifold sampling for optimization of nonconvex functions that are piecewise linear compositions of smooth components. *SIAM J. Optim.*, 28:3001–3024, 2018.
[26] K. C. Kiwiel. Proximity control in bundle methods for convex nondifferentiable minimization. *Math. Program.*, 46:105–122, 1990.
[27] K. C. Kiwiel. A tilted cutting plane proximal bundle method for convex nondifferentiable optimization. *Oper. Res. Lett.*, 10:75–81, 1991.
[28] K. C. Kiwiel. *Methods of descent for nondifferentiable optimization*, volume 1133. Springer, Berlin, 2006.
[29] K. C. Kiwiel. A nonderivative version of the gradient sampling algorithm for nonsmooth nonconvex optimization. *SIAM J. Optim.*, 20:1983–1994, 2010.
[30] J. Larson, M. Menickelly, and S. M. Wild. Manifold sampling for L1 nonconvex optimization. *SIAM J. Optim.*, 26:2540–2563, 2016.
[31] S. Le Digabel. Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm. *ACM Trans. Math. Softw.*, 37:1–15, 2011.
[32] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov. New variants of bundle methods. *Math. Program.*, 69:111–147, 1995.
[33] L. Lukšan and J. Vlček. A bundle-Newton method for nonsmooth unconstrained minimization. *Math. Program.*, 83:373–391, 1998.
[34] L. Lukšan and J. Vlček. Test problems for nonsmooth unconstrained and linearly constrained optimization. *Technická zpráva*, 798, 2000.
[35] M. Mäkelä. Survey of bundle methods for nonsmooth optimization. *Optim. Methods Softw.*, 17:1–29, 2002.
[36] J. J. Moré and S. M. Wild. Benchmarking derivative-free optimization algorithms. *SIAM J. Optim.*, 20:172–191, 2009.
[37] H. Schramm and J. Zowe. A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results. *SIAM J. Optim.*, 2:121–152, 1992.

[38] L. N. Vicente and A. L. Custódio. Analysis of direct searches for discontinuous functions. *Math. Program.*, 133:299–325, 2012.

G. Liuzzi
Istituto di Analisi dei Sistemi ed Informatica "A.Ruberti", Consiglio Nazionale delle Ricerche, Via dei Taurini 19, 00185 Rome, Italy
E-mail address: giampaolo.liuzzi@iasi.cnr.it

S. Lucidi
Dipartimento di Ingegneria Informatica Automatica e Gestionale, "Sapienza" Università di Roma, Via Ariosto 25, 00185 Rome, Italy
E-mail address: lucidi@dis.uniroma1.it

F. Rinaldi
Dipartimento di Matematica "Tullio Levi-Civita", Università di Padova, Via Trieste 63, 35121 Padova, Italy
E-mail address: rinaldi@math.unipd.it

L. N. Vicente
Department of Industrial and Systems Engineering, Lehigh University, 200 West Packer Avenue, Bethlehem, PA 18015-1582, USA and Centre for Mathematics of the University of Coimbra (CMUC), Portugal
E-mail address: lnv@mat.uc.pt