Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 223 (2009) 62–74
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

Unconstrained derivative-free optimization by successive approximation☆

Árpád Bűrmen*, Tadej Tuma

University of Ljubljana, Faculty of Electrical Engineering, Tržaška 25, SI-1000 Ljubljana, Slovenia

Received 21 April 2006; received in revised form 20 December 2007

Abstract

We present an algorithmic framework for unconstrained derivative-free optimization based on dividing the search space in regions (partitions). Every partition is assigned a representative point. The representative points form a grid. A piecewise-constant approximation to the function subject to optimization is constructed using a partitioning and its corresponding grid. The convergence of the framework to a stationary point of a continuously differentiable function is guaranteed under mild assumptions. The proposed framework is appropriate for upgrading heuristics that lack mathematical analysis into algorithms that guarantee convergence to a local minimizer. A convergent variant of the Nelder–Mead algorithm that conforms to the given framework is constructed. The algorithm is compared to two previously published convergent variants of the NM algorithm. The comparison is conducted on the Moré–Garbow–Hillstrom set of test problems and on four variably-dimensional functions with dimension up to 100. The results of the comparison show that the proposed algorithm outperforms both previously published algorithms.

MSC: 65K05; 90C56

Keywords: Unconstrained minimization; Direct search; Successive approximation; Grid; Simplex

1. Introduction

Solving unconstrained optimization problems of the form

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

where $f : \mathbb{R}^n \to \mathbb{R}$ has received a lot of attention lately, in particular methods that search for local minima of f. Several different methods for solving such problems without using derivative information (direct search) were proposed in the past. These so-called direct search methods were despised by the optimization community at first [22] because most of them lacked mathematical analysis. In the past 20 years the advancements in computational

☆ The research was co-funded by the Ministry of Education, Science, and Sport (Ministrstvo za Šolstvo, Znanost in Šport) of the Republic of Slovenia through the programme P2-0246 Algorithms and optimization methods in telecommunications.
* Corresponding author.
E-mail address: arpadb@fides.fe.uni-lj.si (Á. Bűrmen).

0377-0427/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2007.12.017

capabilities and simulation techniques lead to many optimization problems where no derivatives of f are available. Consequently direct search methods became interesting for optimization practitioners.

The situation began to change with the advent of the multi-directional search by Torczon [20]. Its convergence theory was based on the fact that all visited points lie on successively finer grids. The convergence theory that followed [21] established the class of pattern search methods. Several well-known algorithms belong to that class, among others also the Hooke–Jeeves algorithm [11]. The developments continued by allowing larger flexibility in choosing the grid [8] and introducing a sufficient descent condition [7] which removes the requirement that the points must lie on a grid.

On the other side developments occurred on generalizing the convergence theory in the direction of nonsmooth functions (functions that are not continuously differentiable). The activities in this field started with the introduction of the generalized pattern search (GPS) [1] and the nonsmooth approach of Coope and Price [9]. GPS evolved into mesh-adaptive direct search (MADS) [2] where an asymptotically dense set of search directions is used. A very good overview which encompasses mostly the analysis for continuously differentiable functions is given in [12]. A somewhat older review of direct search methods can be found in [15]. Some of the above-mentioned convergence analyses are developed for constrained optimization algorithms [1,9,2,12].

This paper presents a framework for ensuring convergence to a local minimizer of continuously differentiable functions. The framework is based on the idea of grid restrainment from [4] where it was used with a very special form of a grid. The generalization presented here allows non-uniform grids provided that some simple requirements are satisfied. These requirements are equivalent to those imposed on the admissible sets in [9] (i.e. the intersection of a bounded set and the grid must always be finite). The division of $\mathbb{R}^n$ into regions (partitions) which define the behavior of the grid-restrainment operator can also be chosen in a very flexible manner.

Our framework is a byproduct of the search for simple convergent variants of the Nelder–Mead (NM) algorithm [18] which gave rise to the notion of grid restrainment. The effect of grid restrainment to successively finer grids can also be viewed from a different perspective. Instead of grid-restrained points we are working with increasingly finer piecewise-constant approximations to f. This interpretation leads to the successive approximation NM (SANM) algorithm. SANM requires less linear algebra operations than its predecessor, the grid-restrained NM (GRNM) algorithm [4], and is also faster.

The paper is divided as follows. First the background for analyzing our framework is developed. The framework is presented and its convergence is established under mild assumptions. Next a variant of the Nelder–Mead algorithm conforming to the presented framework is described. The algorithm is tested on the Moreé–Garbow–Hillstrom [17] test suite and on some multi-dimensional test problems with dimension ranging up 100. The results are compared to those obtained with the convergent simplex variants proposed in [19,5] and [4]. The variant [4] is shown to conform to the proposed framework. Finally the conclusions are given.

Notation. Vectors are denoted by lowercase letters and are assumed to be column vectors so that $\mathbf{x}^{\mathrm{T}}\mathbf{y}$ denotes the scalar product of $\mathbf{x}$ and $\mathbf{y}$. Matrices are denoted by uppercase letters e.g. $\mathbf{A}$. A_{ij} denotes jth element in the ith row of matrix $\mathbf{A}$. The corresponding lowercase letter with a superscript is reserved for matrix columns (e.g. $\mathbf{a}^i$). Set members are also denoted with a superscript. Members of a sequence $\{\mathbf{x}_k\}_{k=1}^{\infty}$ are denoted by a subscript (e.g. $\mathbf{x}_k$). Calligraphic uppercase letters are reserved for maps and sets. $\mathbb{R}$ and $\mathbb{Z}$ denote the set of real and integer numbers, respectively. Function $o(x)$ is such that $\lim_{x\downarrow 0} o(x)/x = 0$. $\mathcal{W}_r$ denotes an open ball with an arbitrary center and radius r. An open ball with radius r centered at $\mathbf{x}$ is denoted by $\mathcal{W}_r(\mathbf{x})$. The remaining notation is introduced in the text as needed.

2. Background

In the case of n-dimensional unconstrained optimization the search is conducted in $\mathbb{R}^n$.

Definition 1. Partitioning $\mathcal{P}(\mathbb{R}^n)$ divides $\mathbb{R}^n$ into a set of partitions P^i such that $\bigcup_i P^i = \mathbb{R}^n$ and $P^i \cap P^j \neq \emptyset$ iff $i = j$.

Now suppose that every partition P^i is assigned a representative point $\mathbf{p}^i \in P^i$. Let $\operatorname{diam}(P^i) = \max_{\mathbf{x},\mathbf{y}\in P^i} \|\mathbf{x}-\mathbf{y}\|$ denote the diameter of a partition.

Definition 2. A grid $\mathcal{G}(\mathbb{R}^n, \mathcal{P})$ is a one-to-one map between the set of representative points and the partitioning $\mathcal{P}(\mathbb{R}^n)$.

Lemma 3. *Suppose that for a given partitioning there exist $0 < \omega < \Omega$ such that for every partition P^i we can find a ball $\mathcal{W}_\omega \subseteq P^i$ and a ball $\mathcal{W}_\Omega$ such that $P^i \subseteq \mathcal{W}_\Omega$. Then for every compact set $\mathcal{C}$ the number of partitions P^i for which $P^i \cap \mathcal{C} \neq \emptyset$ is finite.*

Proof. Suppose that this is not true. Then there must exist an infinite set of partitions $\mathcal{S} = \{P^1, P^2, \ldots\}$ such that every member of this set has a non-empty intersection with $\mathcal{C}$. Since there exists $\Omega > 0$ such that every partition has a ball $\mathcal{W}_\Omega \supseteq P^i$ associated with it we can choose a compact set $\mathcal{C}_1 \supseteq \mathcal{C}$ for which $P^i \subseteq \mathcal{C}_1$ for all $P^i \in \mathcal{S}$.

Choose a finite subset of $\mathcal{S}$ and denote it $\mathcal{S}_1$. $\mathcal{C}$ cannot be a subset of $\bigcup_{P^i \in \mathcal{S}_1} P^i$ as then $\mathcal{S}$ would be finite. So there must exist some $P^{k_1} \notin \mathcal{S}_1$ and $P^{k_1} \in \mathcal{S}$ for which $P^{k_1} \cap \mathcal{C} \neq \emptyset$. Replace $\mathcal{S}$ with $\mathcal{S} \setminus \{P^{k_i}\}$ and choose a new $\mathcal{S}_1$. By repeating this reasoning we obtain an infinite sequence of partitions $\{P^{k_i}\}_{i=1}^{\infty}$ that have a non-empty intersection with $\mathcal{C}$ and lie in a compact set $\mathcal{C}_1$. Every partition from this sequence has a corresponding inscribed ball $\mathcal{W}_\omega \subseteq P^i$. The centerpoints of these balls constitute an infinite sequence within compact set $\mathcal{C}_1$ and the distance between any two points $\mathbf{x}_i, \mathbf{x}_j$ from this sequence is at least ω. Thus the sequence can have no limit points, which contradicts the compactness of $\mathcal{C}_1$. □

Definition 4. A grid-restrainment operator $\mathcal{R}_{\mathcal{G},\mathcal{P}}$ is a map $\mathbb{R}^n \to \mathcal{G}(\mathbb{R}^n, \mathcal{P})$ such that $\mathcal{R}_{\mathcal{G},\mathcal{P}}(\mathbf{x}) = \mathbf{p}^i$ iff $\mathbf{p}^i \in P^i$ and $\mathbf{x} \in P^i$.

Grid restrainment of an $\mathbf{x} \in \mathbb{R}^n$ results in a point $\mathcal{R}_{\mathcal{G},\mathcal{P}}(\mathbf{x})$ that more or less differs from $\mathbf{x}$ (grid-restrainment error). If the partition diameter has an upper bound this error also has an upper bound $\delta(\mathcal{G}, \mathcal{P}) = \max_{\mathbf{x} \in \mathbb{R}^n} \|\mathcal{R}_{\mathcal{G},\mathcal{P}}(\mathbf{x}) - \mathbf{x}\| \leq \max_i \operatorname{diam}(P^i)$.

Definition 5. $\tilde{f}(\mathbf{x})$ is a $(\mathcal{G}, \mathcal{P})$ approximation to $f(\mathbf{x})$ if $\tilde{f}(\mathbf{x}) = f(\mathcal{R}_{\mathcal{G},\mathcal{P}}(\mathbf{x}))$.

Note that the value of $\tilde{f}$ is constant across every partition.

Lemma 6. *Let $\tilde{f}(\mathbf{x})$ be a $(\mathcal{G}, \mathcal{P})$ approximation to $f(\mathbf{x})$ and let the requirements of Lemma 3 hold for $\mathcal{P}$. Suppose that an algorithm starts out with $\mathbf{x}_0$ and the level set $\mathcal{L} = \{\mathbf{x} : f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$ is compact. If the ith step of the algorithm produces $\mathbf{x}_i$ for which $\tilde{f}(\mathbf{x}_i) < \tilde{f}(\mathbf{x}_{i-1})$ then the algorithm terminates in a finite number of steps.*

Proof. Obviously $\tilde{f}(\mathbf{x}_i) < \tilde{f}(\mathbf{x}_0)$. Since the requirements of Lemma 3 hold and $\mathcal{L}$ is compact, there exists a compact set $\tilde{\mathcal{L}} \supseteq \{\mathbf{x} : \tilde{f}(\mathbf{x}) \leq \tilde{f}(\mathbf{x}_0)\}$. Lemma 3 states that $\tilde{\mathcal{L}}$ can be covered with a finite number of partitions. $\tilde{f}$ is constant over every one of these partitions by definition so the algorithm can choose between finitely many values of $\tilde{f}$ and subsequently there are only finitely many $\tilde{f}(\mathbf{x}_i)$ with $\tilde{f}(\mathbf{x}_i) < \tilde{f}(\mathbf{x}_0)$. □

The following 3 definitions establish the notions of locally Lipschitz function, Clarke generalized derivative, and strict differentiability [6,3].

Definition 7. A function f is locally Lipschitz around $\mathbf{x}$ if there exist $K > 0$ and $\delta > 0$ such that $|f(\mathbf{x}_1) - f(\mathbf{x}_2)| \leq K\|\mathbf{x}_1 - \mathbf{x}_2\|$ for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{W}_\delta(\mathbf{x})$.

Definition 8. Suppose that f is locally Lipschitz around $\mathbf{x}$. Then

$$f^\circ(\mathbf{x}; \mathbf{d}) = \lim_{\mathbf{y} \to \mathbf{x}} \sup_{t \downarrow 0} \frac{f(\mathbf{y} + t\mathbf{d}) - f(\mathbf{y})}{t} \tag{2}$$

is the Clarke generalized derivative [6].

Definition 9. A function f is strictly differentiable at $\mathbf{x}$ if there exists $\mathbf{w} \in \mathbb{R}^n$ and $\delta > 0$ such that $\lim_{\mathbf{y},\mathbf{z} \to \mathbf{x}, \mathbf{y} \neq \mathbf{z}} \frac{f(\mathbf{y}) - f(\mathbf{z}) - \mathbf{w}^{\mathrm{T}}(\mathbf{y} - \mathbf{z})}{\|\mathbf{y} - \mathbf{z}\|} = 0$ where $\mathbf{y}, \mathbf{z} \in \mathcal{W}_\delta(\mathbf{x})$. $\mathbf{w}$ is the strict derivative of f at $\mathbf{x}$.

If a function is strictly differentiable at $\mathbf{x}$ then it is also locally Lipschitz around $\mathbf{x}$ and $f^\circ(\mathbf{x}; \mathbf{d}) = \mathbf{w}^{\mathrm{T}}\mathbf{d}$. Continuous differentiability at $\mathbf{x}$ implies that $f^\circ(\mathbf{x}; \mathbf{d}) = (\nabla f(\mathbf{x}))^{\mathrm{T}}\mathbf{d}$ and results in f being strictly differentiable at $\mathbf{x}$.

Definition 10. A set $\mathcal{D} = \{\mathbf{d}^1, \mathbf{d}^2, \ldots, \mathbf{d}^r\}$ positively spans $\mathcal{A}$ if for every $\mathbf{x} \in \mathcal{A}$ there exists a set of nonnegative scalars α^i such that $\mathbf{x} = \sum_{i=1}^{r} \alpha^i \mathbf{d}^i$.

In other words vectors $\mathbf{x} \in \mathcal{A}$ are in the positive span of $\mathcal{D}$. The quality of a positive spanning set [14] can be expressed as

$$\epsilon(\mathcal{D}) = \min_{\mathbf{v}\in\mathbb{R}^n, \|\mathbf{v}\|\neq 0} \max_{\mathbf{d}\in\mathcal{D}} \frac{\mathbf{d}^{\mathrm{T}}\mathbf{v}}{\|\mathbf{d}\|\|\mathbf{v}\|} < 1 \tag{3}$$

$\epsilon(\mathcal{D}) > 0$ implies that $\mathcal{D}$ positively spans $\mathbb{R}^n$ (for proof see [4]).

Now suppose that the set $\tilde{\mathcal{D}}$ is obtained from $\mathcal{D}$ by restraining the members of $\mathcal{D}$ to grid $\mathcal{G}$ using partitioning $\mathcal{P}$. Let $\mathbf{d}^{\min}$ denote the shortest member of $\mathcal{D}$. Then (provided that no member of $\tilde{\mathcal{D}}$ has zero length)

$$\epsilon(\tilde{\mathcal{D}}) \geq \frac{\epsilon(\mathcal{D}) - \delta(\mathcal{G},\mathcal{P})/\|\mathbf{d}^{\min}\|}{1 + \delta(\mathcal{G},\mathcal{P})/\|\mathbf{d}^{\min}\|}. \tag{4}$$

See [4] for proof. If the members of $\mathcal{D}$ are obtained by subtracting two vectors (e.g. $\mathbf{d}^i = \mathbf{a}^i - \mathbf{b}^i$) the grid restrainment of vectors $\mathbf{a}^i$ and $\mathbf{b}^i$ results in the following estimate for $\tilde{\mathcal{D}}$.

$$\epsilon(\tilde{\mathcal{D}}) \geq \frac{\epsilon(\mathcal{D}) - 2\delta(\mathcal{G},\mathcal{P})/\|\mathbf{d}^{\min}\|}{1 + 2\delta(\mathcal{G},\mathcal{P})/\|\mathbf{d}^{\min}\|}. \tag{5}$$

The proof again goes along the lines of [4], except that the grid-restrainment error must be applied twice.

Definition 11. Finite set $\mathcal{D}_\infty \subseteq \mathbb{R}^n$ is a limit point of a sequence of sets $\{\mathcal{D}_k\}_{k=1}^\infty$ if there exists a subsequence $\{\mathcal{D}_{i_k}\}_{k=1}^\infty$ such that for every $\mathbf{d} \in \mathcal{D}_\infty$

$$\lim_{k\to\infty} \min_{\mathbf{d}'\in\mathcal{D}_{i_k}} \|\mathbf{d} - \mathbf{d}'\| = 0. \tag{6}$$

The following definition will simplify the proceedings for establishing the convergence theory of an algorithm.

Definition 12. A constellation $\mathcal{X}$ is an ordered set of $m+1$ points $\{\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^m\}$ where $m > 0$. Vector $\mathbf{x}^0$ is the origin and set $\mathcal{B}_\xi(\mathcal{X}) = \{\frac{\mathbf{x}^1-\mathbf{x}^0}{\xi}, \frac{\mathbf{x}^2-\mathbf{x}^0}{\xi}, \ldots, \frac{\mathbf{x}^m-\mathbf{x}^0}{\xi}\}$ is the ξ-basis of constellation $\mathcal{X}$. Note that a ξ-basis is not necessarily a linear basis for R^n.

By restraining every member of $\mathcal{X}_k$ to grid $\mathcal{G}_k$ using partitioning $\mathcal{P}_k$ we get a constellation denoted by $\tilde{\mathcal{X}}_k$. Let ϵ_k, $\tilde{\epsilon}_k$, and δ_k denote $\epsilon(\mathcal{B}_{\xi_k}(\mathcal{X}_k))$, $\epsilon(\mathcal{B}_{\xi_k}(\tilde{\mathcal{X}}_k))$, and $\delta(\mathcal{G}_k,\mathcal{P}_k)$, respectively.

Lemma 13. *For a given sequence of constellations $\{\mathcal{X}_k\}_{k=1}^\infty$ choose a sequence of positive scalars $\{\xi_k\}_{k=1}^\infty$, a sequence of partitionings $\{\mathcal{P}_k\}_{k=1}^\infty$, and a sequence of grids $\{\mathcal{G}_k\}_{k=1}^\infty$. Suppose that there exist $\alpha > 0$ and $\Lambda > 0$ such that for every $\mathbf{b} \in \mathcal{B}_{\xi_k}(\mathcal{X}_k)$ and all $\mathcal{X}_k$*

$$\|\mathbf{b}\| \leq \Lambda, \tag{7}$$

$$\epsilon_k\|\mathbf{b}\| - 2\delta_k/\xi_k \geq \alpha \tag{8}$$

then there exists a limit point $\tilde{\mathcal{B}}_\infty$ of sequence $\{\mathcal{B}_{\xi_k}(\tilde{\mathcal{X}}_k)\}_{k=1}^\infty$ and every $\tilde{\mathcal{B}}_\infty$ positively spans $\mathbb{R}^n$.

Proof. The nature of grid restrainment implies bounds on $\tilde{\mathbf{b}} \in \mathcal{B}_{\xi_k}(\tilde{\mathcal{X}}_k)$ where $\mathbf{b}$ is the corresponding member of $\mathcal{B}_{\xi_k}(\mathcal{X}_k)$

$$\|\mathbf{b}\| - 2\delta_k/\xi_k \leq \|\tilde{\mathbf{b}}\| \leq \|\mathbf{b}\| + 2\delta_k/\xi_k. \tag{9}$$

Since $\epsilon_k\|\mathbf{b}\| < \|\mathbf{b}\|$ we can use (7) and (8) to obtain $2\delta_k/\xi_k < \Lambda - \alpha$ and

$$\alpha \leq \|\tilde{\mathbf{b}}\| < 2\Lambda - \alpha. \tag{10}$$

From (5), (7) and (8) we can write

$$\tilde{\epsilon}_k \geq \frac{\epsilon_k - 2\delta_k/(\xi_k\|\mathbf{b}^{\min}\|)}{1 + 2\delta_k/(\xi_k\|\mathbf{b}^{\min}\|)} = \frac{\|\mathbf{b}^{\min}\|\epsilon_k - 2\delta_k/\xi_k}{\|\mathbf{b}^{\min}\| + 2\delta_k/\xi_k} > \frac{\alpha}{2\Lambda - \alpha}. \tag{11}$$

Now since ξ_k and δ_k are both positive, we get

$$0 < \delta_k/\xi_k \leq \epsilon_k \|\mathbf{b}\| - \alpha \leq \epsilon_k \Lambda - \alpha < \Lambda - \alpha. \tag{12}$$

Eq. (10) guarantees the existence of limit point $\tilde{\mathcal{B}}_\infty$ and assures us that no limit point contains zero vectors. The latter is a consequence of (8) and does not require an additional lower bound λ on $\|\mathbf{b}\|$ as in [4]. From (11) and (12) it follows that $\tilde{\epsilon}_k > \alpha/(2\Lambda - \alpha) > 0$. This means that $\epsilon(\tilde{\mathcal{B}}_\infty) > 0$ and $\tilde{\mathcal{B}}_\infty$ positively spans $\mathbb{R}^n$. □

The following lemma is the basis for proving the convergence of our algorithmic framework.

Lemma 14. *Assume a sequence of points* $\{\mathbf{x}_k\}_{k=1}^\infty$, *a sequence of vectors* $\{\mathbf{b}_k\}_{k=1}^\infty$, *and a sequence of positive scalars* $\{\xi_k\}_{k=1}^\infty$ *converging to* $\mathbf{x}_\infty$, $\mathbf{b}_\infty$, *and* 0, *respectively. Let* f *be locally Lipschitz around* $\mathbf{x}_\infty$. *If* $f(\mathbf{x}_k + \xi_k\mathbf{b}_k) \geq f(\mathbf{x}_k) - o(\xi_k\|\mathbf{b}_k\|)$ *holds for every member of the sequence then the Clarke generalized derivative* $f^\circ(\mathbf{x}_\infty; \mathbf{b}_\infty)$ *is nonnegative.*

Proof. From the assumption on the function value at $\mathbf{x}_k$ we can write

$$\frac{f(\mathbf{y}_k + \xi_k\mathbf{b}_\infty) - f(\mathbf{y}_k) + f(\mathbf{y}_k) - f(\mathbf{x}_k)}{\xi_k} \geq -\frac{o(\xi_k\|\mathbf{b}_k\|)}{\xi_k} \tag{13}$$

where $\mathbf{y}_k = \mathbf{x}_k + \xi_k\mathbf{u}_k$ and $\mathbf{u}_k = \mathbf{b}_k - \mathbf{b}_\infty$. By taking the $\limsup_{k\to\infty}$ the right-hand side term vanishes.

$$\limsup_{k\to\infty} \frac{f(\mathbf{y}_k + \xi_k\mathbf{b}_\infty) - f(\mathbf{y}_k)}{\xi_k} + \limsup_{k\to\infty} \frac{f(\mathbf{x}_k + \xi_k\mathbf{u}_k) - f(\mathbf{x}_k)}{\xi_k} \geq 0. \tag{14}$$

Since $\mathbf{u}_k$ approaches 0 and f is Lipschitz continuous near $\mathbf{x}_\infty$, the second term vanishes. The first term is a lower bound for $f^\circ(\mathbf{x}_\infty; \mathbf{b}_\infty)$ which in turn must also be nonnegative. □

Lemma 15. *Suppose that* f *is strictly differentiable at* $\mathbf{x}$ *and* $f^\circ(\mathbf{x}; \mathbf{b}) \geq 0$ *for all* $\mathbf{b} \in \mathcal{B} = \{\mathbf{b}^1, \mathbf{b}^2, \ldots, \mathbf{b}^r\}$. *Then* $f^\circ(\mathbf{x}; \mathbf{v}) \geq 0$ *for all* $\mathbf{v}$ *that are in the positive span of* $\mathcal{B}$. *If* f *is continuously differentiable at* $\mathbf{x}$ *and* $\mathcal{B}$ *positively spans* $\mathbb{R}^n$, $\mathbf{x}$ *is a stationary point of* f.

Proof. Since $\mathbf{v}$ lies in the positive span of $\mathcal{B}$, it can be expressed as $\mathbf{v} = \sum_{i=1}^r \alpha^i\mathbf{b}^i$ where $\alpha^i \geq 0$. Strict differentiability at $\mathbf{x}$ implies that $f^\circ(\mathbf{x}; \mathbf{b}) = \mathbf{w}^{\mathrm{T}}\mathbf{b}$. By applying it to $f^\circ(\mathbf{x}; \mathbf{v})$ we get

$$f^\circ(\mathbf{x}; \mathbf{v}) = \mathbf{w}^{\mathrm{T}}\mathbf{v} = \sum_{i=1}^r \alpha^i\mathbf{w}^{\mathrm{T}}\mathbf{b}^i = \sum_{i=1}^r \alpha^i f^\circ(\mathbf{x}; \mathbf{b}^i) \geq 0. \tag{15}$$

For the second part ($\mathbf{x}$ being a stationary point of f): $\mathcal{B}$ positively spanning $\mathbb{R}^n$ implies that $f^\circ(\mathbf{x}; \mathbf{v}) = \mathbf{w}^T\mathbf{v} \geq 0$ for all $\mathbf{v} \in \mathbb{R}^n$, so $\mathbf{w}$ can only be 0. For continuously differentiable functions $\mathbf{w} = \nabla f(\mathbf{x})$ and $\mathbf{x}$ is a stationary point of f. □

The following algorithmic framework for unconstrained optimization will be the subject of discussion in the remainder of the paper.

Algorithm 1. Partitioning-based algorithmic framework

(1) Set $k := 1$. Choose an initial point $\mathbf{x}_1$ and a partitioning $\mathcal{P}_1$ with a set of representative points forming grid $\mathcal{G}_1$. Let $\tilde{f}_k$ be a $(\mathcal{G}_k, \mathcal{P}_k)$ approximation to $f : \mathbb{R}^n \to \mathbb{R}$.
(2) Evaluate $\tilde{f}_k$ for a finite set of points and let $\mathbf{x}'$ be the one with the lowest $\tilde{f}_k$ value.
(3) If $\tilde{f}_k(\mathbf{x}') < \tilde{f}_k(\mathbf{x}_k)$ set $\mathbf{x}_k := \mathbf{x}'$ and go back to step 2.
(4) Generate a constellation $\mathcal{X}_k$ comprising origin $\mathbf{x}_k^0 = \mathbf{x}_k$ and m additional points.
(5) If there exists $\mathbf{x}' \in \mathcal{X}_k$ for which $\tilde{f}_k(\mathbf{x}') < \tilde{f}_k(\mathbf{x}_k) - o(\|\mathbf{x}' - \mathbf{x}_k\|)$, set $\mathbf{x}_k := \mathbf{x}'$ and go back to step 2.
(6) Set $\mathbf{x}_{k+1} := \mathbf{x}_k$, choose a new partitioning $\mathcal{P}_{k+1}$ with a new set of representative points $\mathcal{G}_{k+1}$, and increment k. Go back to step 2.

We make the following assumptions for our framework.

A1. For every partitioning $\mathcal{P}_k$ there exist ω and Ω ($0 < \omega < \Omega$) such that for every partition P^i relations $\mathcal{W}_\omega \subseteq P^i$ and $P^i \subseteq \mathcal{W}_\Omega$ hold.

A2. There exists a sequence of scalars $\{\xi_k\}_{k=1}^\infty$ and $\Lambda > 0$ such that constellations $\mathcal{X}_k$ for which Algorithm 1 ends up at step 6 satisfy

$$\|\mathbf{b}\| \leq \Lambda \quad \forall \mathbf{b} \in \mathcal{B}_{\xi_k}(\mathcal{X}_k). \tag{16}$$

A3. There exists $\alpha > 0$ such that partitionings $\mathcal{P}_k$ and grids $\mathcal{G}_k$ for which Algorithm 1 ends up at step 6 satisfy

$$\epsilon_k \|\mathbf{b}\| - 2\delta_k/\xi_k \geq \alpha \quad \forall \mathbf{b} \in \mathcal{B}_{\xi_k}(\mathcal{X}_k). \tag{17}$$

A4. f is continuously differentiable with compact level sets.

Now we are prepared for our main result.

Theorem 16. *Suppose that ξ_k goes to 0 as k approaches infinity. Then assuming* A1–A4 *all limit points of sequence $\{\mathbf{x}_k\}_{k=1}^\infty$ (where $\mathbf{x}_k$ is collected at step 6 of Algorithm 1) are stationary points of f.*

Proof. Assumption A1 fulfills the requirements of Lemma 3. Since Algorithm 1 replaces $\mathbf{x}_k$ only if it decreases the value of $\tilde{f}(\mathbf{x}_k)$ and the level sets of f are compact, we can invoke Lemma 6 which ensures us that steps 2–5 are repeated finitely many times before step 6 is reached. This means that step 6 is visited infinitely many times. Therefore we can form infinite sequences of partitionings $\mathcal{P}_k$, grids $\mathcal{G}_k$, constellations $\mathcal{X}_k$, and constellation origins $\mathbf{x}_k^0$ collected at the beginning of step 6.

The sequence of grid-restrained constellation origins $\{\tilde{\mathbf{x}}_k^0\}_{k=1}^\infty$ (where $\tilde{\mathbf{x}}_k^0 = \mathcal{R}_{\mathcal{G}_k,\mathcal{P}_k}(\mathbf{x}_k^0)$) satisfies $f(\tilde{\mathbf{x}}_k^0) \leq f(\tilde{\mathbf{x}}_{k-1}^0)$. Care must be taken to make sure that $\tilde{f}(\mathbf{x}_k^0)$ does not change when a new partitioning with a new set of representative points is chosen in step 6. This can be achieved if the old and the new grid share a common representative point $\mathbf{p}$ for which $\mathcal{R}_{\mathcal{G},\mathcal{P}}(\mathbf{x}_k^0) = \mathbf{p}$. Since the level sets of f are compact the sequence $\{\tilde{\mathbf{x}}_k^0\}_{k=1}^\infty$ has at least one limit point $\tilde{\mathbf{x}}_\infty^0$.

Due to assumptions A2 and A3 the requirements of Lemma 13 are satisfied for sequences $\{\mathcal{X}_k\}_{k=1}^\infty$, $\{\xi_k\}_{k=1}^\infty$, $\{\mathcal{P}_k\}_{k=1}^\infty$, and $\{\mathcal{G}_k\}_{k=1}^\infty$. Therefore a limit point $\tilde{\mathcal{B}}_\infty$ of sequence $\{\mathcal{B}_{\xi_k}(\tilde{\mathcal{X}}_k)\}_{k=1}^\infty$ exists and all such limit points positively span $\mathbb{R}^n$.

Now choose a pair of limit points $\tilde{\mathbf{x}}_\infty^0$ and $\tilde{\mathcal{B}}_\infty$ and a subset of indices $\mathcal{K}$ such that the corresponding subsequence of $\{\tilde{\mathbf{x}}_k^0\}_{k=1}^\infty$ and $\{\mathcal{B}_{\xi_k}(\tilde{\mathcal{X}}_k)\}_{k=1}^\infty$ converges to $\tilde{\mathbf{x}}_\infty^0$ and $\tilde{\mathcal{B}}_\infty$, respectively. Replace all sequences with subsequences of themselves where $k \in \mathcal{K}$. This makes the proof valid for every limit point $\tilde{\mathbf{x}}_\infty^0$.

For any $\tilde{\mathbf{b}}_\infty \in \tilde{\mathcal{B}}_\infty$ we can form a sequence of vectors $\{\tilde{\mathbf{b}}_k\}_{k=1}^\infty$ where $\tilde{\mathbf{b}}_k \in \mathcal{B}_{\xi_k}(\tilde{\mathcal{X}}_k)$ such that it converges to $\tilde{\mathbf{b}}_\infty \in \tilde{\mathcal{B}}_\infty$. A2 asserts that $\delta_k \leq \xi_k(\epsilon_k \Lambda - \alpha)/2$. Together with (12) it means that the grid-restrainment error goes to 0 as ξ_k goes to 0.

The sequence of grid-restrained constellation origins $\{\tilde{\mathbf{x}}_k^0\}_{k=1}^\infty$ satisfies $f(\tilde{\mathbf{x}}_k^0 + \xi_k \tilde{\mathbf{b}}_k) \geq f(\tilde{\mathbf{x}}_k^0) + o(\xi_k \|\mathbf{b}_k\|)$. By looking at the proof of Lemma 14 we can see that it is still valid, even if we replace $o(\xi_k \|\tilde{\mathbf{b}}_k\|)$ with $o(\xi_k \|\mathbf{b}_k\|)$. Since the continuous differentiability of f implies local Lipschitz continuity we satisfy all requirements of Lemma 14 and conclude $f^\circ(\mathbf{x}_\infty; \tilde{\mathbf{b}}_\infty) \geq 0$.

Recall that $\tilde{\mathcal{B}}_\infty$ positively spans $\mathbb{R}^n$. The continuous differentiability of f and Lemma 15 result in $\nabla f(\tilde{\mathbf{x}}_\infty^0) = \mathbf{0}$. $\delta_k \to 0$ implies that $\|\tilde{\mathbf{x}}_k - \mathbf{x}_k\| \to 0$ and together with continuous differentiability of f result in $\|\nabla f(\mathbf{x}_k)\| \to 0$. □

3. The successive approximation simplex algorithm

3.1. The algorithm

The NM algorithm tries to find a local minimum of f by moving a polytope with $n + 1$ vertices (simplex) through the search space. The movement is achieved with simple geometric operations on the set of simplex vertices guided solely by their relative ordering according to the value of f. Its popularity is probably a consequence of its simplicity and the fact that for many practical optimization problems it performs astoundingly well. An overview of various modifications to the original NM algorithm can be found in [4].

Denote the simplex vertices with $\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^n$ and relabel them such that $f(\mathbf{x}^0) \leq f(\mathbf{x}^1) \leq \cdots \leq f(\mathbf{x}^n)$ holds. To simplify the notation f^i is used for $f(\mathbf{x}^i)$. The centroid of the n vertices with the lowest value of f is defined as

$\mathbf{x}^{\mathrm{cb}} = \frac{1}{n}\sum_{i=0}^{n-1}\mathbf{x}^i$. A candidate point for replacing $\mathbf{x}^n$ lies on the line defined by $\mathbf{x}^{\mathrm{cb}}$ and $\mathbf{x}^n$ and can be expressed as $\mathbf{x}(\gamma) = \mathbf{x}^{\mathrm{cb}} + \gamma(\mathbf{x}^{\mathrm{cb}} - \mathbf{x}^n)$.

Several candidate points for replacing $\mathbf{x}^n$ are examined. The candidates are $\mathbf{x}^{\mathrm{r}}$, $\mathbf{x}^{\mathrm{e}}$, $\mathbf{x}^{\mathrm{oc}}$, and $\mathbf{x}^{\mathrm{ic}}$ with the corresponding values of γ denoted by γ_{r}, γ_{e}, γ_{oc}, and γ_{ic}. In the literature they are usually referred to as the reflection, expansion, outer contraction, and inner contraction point. If none of the above-mentioned candidates is good enough to replace $\mathbf{x}^n$ the simplex is shrunk toward $\mathbf{x}^0$ using the formula $\mathbf{x}^0 + \gamma_{\mathrm{s}}(\mathbf{x}^i - \mathbf{x}^0)$ for $i = 1, 2, \ldots, n$.

Values of γ satisfy the following requirements

$$0 < \gamma_{\mathrm{r}} < \gamma_{\mathrm{e}}, \quad \gamma_{\mathrm{e}} > 1, \quad 0 < \gamma_{\mathrm{oc}} < 1, \quad -1 < \gamma_{\mathrm{ic}} < 0, \quad 0 < \gamma_{\mathrm{s}} < 1. \tag{18}$$

In the original paper by Nelder and Mead [18] the following values were proposed: $\gamma_{\mathrm{r}} = 1$, $\gamma_{\mathrm{e}} = 2$, and $\gamma_{\mathrm{oc}} = -\gamma_{\mathrm{ic}} = \gamma_{\mathrm{s}} = 0.5$. Algorithm 2 is the summary of the original Nelder–Mead algorithm as stated by [13]. This algorithm differs slightly from the original version in [18] where several ambiguities are present.

The initial simplex can be chosen randomly or by using some predefined rules. Next f is evaluated at the simplex vertices upon which iterations of Algorithm 2 are repeated until some stopping condition is satisfied.

Algorithm 2. One iteration of the NM algorithm:

(1) Order the simplex.
(2) Evaluate $f^{\mathrm{r}} = f(\mathbf{x}^{\mathrm{r}})$. If $f^{\mathrm{r}} < f^0$ evaluate $f^{\mathrm{e}} = f(\mathbf{x}^{\mathrm{e}})$.
    If $f^{\mathrm{e}} < f^{\mathrm{r}}$ replace $\mathbf{x}^n$ with $\mathbf{x}^{\mathrm{e}}$, otherwise replace it with $\mathbf{x}^{\mathrm{r}}$.
(3) If $f^0 \leq f^{\mathrm{r}} < f^{n-1}$, replace $\mathbf{x}^n$ with $\mathbf{x}^{\mathrm{r}}$.
(4) If $f^{n-1} \leq f^{\mathrm{r}} < f^n$, evaluate $f^{\mathrm{oc}} = f(\mathbf{x}^{\mathrm{oc}})$.
    If $f^{\mathrm{oc}} \leq f^n$ replace $\mathbf{x}^n$ with $\mathbf{x}^{\mathrm{oc}}$.
(5) If $f^n \leq f^{\mathrm{r}}$, evaluate $f^{\mathrm{ic}} = f(\mathbf{x}^{\mathrm{ic}})$.
    If $f^{\mathrm{ic}} \leq f^n$, replace $\mathbf{x}^n$ with $\mathbf{x}^{\mathrm{ic}}$.
(6) If $\mathbf{x}^n$ was not replaced, shrink the simplex toward $\mathbf{x}^0$.

In the remainder of the paper we assume box-shaped partitions of the form $P = \{\mathbf{x} = [x^1, x^2, \ldots, x^n] : y^i - \Delta^i/2 \leq x^i < y^i + \Delta^i/2\}$ where $\mathbf{y} = [y^1, y^2, \ldots, y^n]$ denotes the representative point and $\boldsymbol{\Delta} = [\Delta^1, \Delta^2, \ldots, \Delta^n]$ is the grid density. The following grid will be assumed to be $\mathcal{G} = \{\mathbf{y} : \mathbf{y} = \mathbf{z} + \sum_{i=1}^n N^i \mathbf{e}^i \Delta^i, N^i \in \mathbb{Z}\}$ where $\mathbf{e}^i$ denotes the ith unit vector and $\mathbf{z}$ is the grid origin. For such a combination of partitioning and grid $\delta = \|\boldsymbol{\Delta}\|/2$.

In SANM algorithm (see Algorithm 3) we replace function f which is used in the original NM algorithm with a sequence of approximations $\{\tilde{f}_i\}_{i=1}^{\infty}$ over gradually finer partitionings. The original values of the simplex scaling coefficients were used except for $\gamma_{\mathrm{e}} = 1.2$ and $\gamma_{\mathrm{s}} = 0.25$. The value of γ_{e} is from [4] where it was found that this value improves the algorithm's performance compared to the original $\gamma_{\mathrm{e}} = 2$. The acceptance criterion for contraction steps is more strict than in the original NM algorithm ($\tilde{f}^{\mathrm{oc}} < \tilde{f}^{\mathrm{n}}$ and $\tilde{f}^{\mathrm{ic}} < \tilde{f}^{\mathrm{n}}$).

The algorithm starts by constructing a simplex around the initial point $\mathbf{x}_0$. The vertices of the initial simplex consist of the initial point $\mathbf{x}^0$ and n additional points obtained by perturbing the individual coordinates of $\mathbf{x}^0$ by 5% or 0.00025 if the respective coordinate value is zero. Let $x^{i,j}$ denote the jth component of vector $\mathbf{x}^i$. The initial grid origin is at $\mathbf{x}_0$ and the initial grid scaling can be expressed as $\Delta^j = \frac{1}{10}\max_{i=1,2,\ldots,n}|x^{i,j} - x^{0,j}|$.

Most of the time step 1 is being executed. The original NM algorithm is considered to fail if none of the four trial points ($\mathbf{x}^{\mathrm{r}}$, $\mathbf{x}^{\mathrm{e}}$, $\mathbf{x}^{\mathrm{oc}}$, and $\mathbf{x}^{\mathrm{ic}}$) is accepted.

One of the reasons why the original NM algorithm fails is the simplex limiting to an $n' < n$-dimensional object. When this happens the search becomes more and more confined to a linear subspace of $\mathbb{R}^n$. The algorithm's progress slows down and ultimately results in inner or outer contraction step not being accepted. Let $\tilde{f}^{\mathrm{best}}$ denote the $\tilde{f}$ value at the best simplex point when this happens.

After a failed contraction step the simplex is checked for degeneracy (Algorithm 3, step 1). Let $\mathbf{v}^i = \mathbf{x}^i - \mathbf{x}^0$, $i = 1, 2, \ldots, n$ denote the simplex side vectors and c some positive constant. Keeping the interior angles bounded away from 0 implies the following:

$$|\det V| = |\det[\mathbf{v}^1, \mathbf{v}^2, \ldots, \mathbf{v}^n]| \geq c^n \prod_{i=1}^{n} \|\mathbf{v}^i\|. \tag{19}$$

If (19) is violated the simplex is reshaped (step 3 of Algorithm 3). The reshape keeps the best point intact and changes the remaining n points. Simplex side vectors are ordered so that $\|\mathbf{v}^1\| \geq \|\mathbf{v}^2\| \geq \cdots \geq \|\mathbf{v}^n\|$ and matrix $\mathbf{V} = [\mathbf{v}^1, \mathbf{v}^2, \ldots, \mathbf{v}^n]$ is constructed. The matrix is factored using QR decomposition ($\mathbf{V} = \mathbf{QR}$). The resulting orthogonal basis $\mathcal{D} = \{\mathbf{d}^1, \mathbf{d}^2, \ldots, \mathbf{d}^n\}$ is obtained from

$$\mathbf{d}^i = \operatorname{sign}(R_{ii}) \max\left(\lambda n^{1/2}\|\Delta\|, \min(|R_{ii}|, \Lambda n^{1/2}\|\Delta\|)\right) \mathbf{q}^i,$$

where $\operatorname{sign}(x)$ is -1 for $x < 0$ and 1 otherwise. Our implementation uses $c = 10^{-6}$, $\lambda = 2$, and $\Lambda = 2^{52}$.

The absolute value of the determinant in (19) can be calculated incrementally like in [19,5]. This is due to the fact that no grid restrainment is performed when a new point is accepted into the simplex (as opposed to [4]). The new value is obtained by multiplying the old value with γ_r, γ_e, γ_oc, γ_ic, or $(\gamma_\mathrm{e} - \gamma_\mathrm{r})/\gamma_\mathrm{r}$ when $\mathbf{x}^\mathrm{r}$, $\mathbf{x}^\mathrm{e}$, $\mathbf{x}^\mathrm{oc}$, $\mathbf{x}^\mathrm{ic}$, or $\mathbf{x}^\mathrm{pe}$ is accepted, respectively. $\mathbf{x}^\mathrm{pe}$ denotes the pseudo-expand point [19,5] defined as $\mathbf{x}^\mathrm{pe} = \mathbf{x}^0 + (\gamma_e/\gamma_r - 1)(\mathbf{x}^0 - \mathbf{x}^\mathrm{cw})$ where $\mathbf{x}^\mathrm{cw} = 1/n \sum_{i=1}^n \mathbf{x}^i$. When the trial steps are shrinked, the determinant is multiplied by γ_s^n. The only time the determinant needs to be calculated from scratch is at the start of the algorithm and at every reshape. But since the reshaped simplex is orthogonal the absolute value of the determinant can be obtained as the product of simplex side lengths. The simplified evaluation of the determinant reduces the number of linear algebra operations compared to the GRNM algorithm where a QR decomposition is performed every time the determinant is evaluated.

The optional reshape is followed by the so-called pseudo-expand step [19,5] (step 4 of Algorithm 3). The best point $\mathbf{x}^0$ is treated as a result of a successful reflection step which should be followed by an expansion step resulting in the pseudo-expand point $\mathbf{x}^\mathrm{pe}$. The pseudo-expand point replaces the best point of the simplex if $\tilde{f}^\mathrm{pe} < \tilde{f}^0$. If any of the points in the resulting simplex is better than $\tilde{f}^\mathrm{best}$ the algorithm continues with the NM algorithm.

If no reshape happened at step 3 a reshape takes place at step 7. What follows is a loop that repeatedly tries $2n$ trial steps around the best point in order to find a point that is better than $\mathbf{x}^0$. n points are tried in one pass of the loop. Since the reshaped simplex is orthogonal the $2n$ steps form a positive spanning set (a maximal positive basis [10]) for $\mathbb{R}^n$.

If the $2n$ points fail to produce descent, the trial steps are shrinked by γ_s (this step is similar to the shrink step in the original NM algorithm). When the trial steps become short compared to the grid-restrainment error the grid and the partitioning are refined. This implies a new (finer) approximation to f. Care must be taken to make sure that $\tilde{f}(\mathbf{x}^0)$ does not change as a result of the refinement. This is achieved by choosing the new grid origin ($\mathbf{z}$) at $\mathcal{R}_{\mathcal{G}^\mathrm{old},\mathcal{P}^\mathrm{old}}(x^0)$ and moving $\mathbf{x}^0$ to $\mathbf{z}$. Here $\mathcal{G}^\mathrm{old}$ and $\mathcal{P}^\mathrm{old}$ denote the grid and the partitioning before refinement.

Let $\mathbf{d}^\mathrm{min}$, $\mathbf{d}^{\mathrm{min},i}$, and z^i denote the shortest trial step vector, its components, and the components of the grid origin, respectively. The components of the new grid scaling vector are obtained as $\Delta^i = \frac{1}{128\lambda n} \max(|d^{\mathrm{min},i}|, \frac{\|\mathbf{d}^\mathrm{min}\|}{n^{1/2}})$. Due to the finite precision of floating point representation there is an inherent grid present. When a component of the grid scaling vector reaches the precision of the floating point representation ($\Delta^i < \max(\tau_\mathrm{r}|z^i|, \tau_\mathrm{a})$) grid restrainment is no longer applied to the ith component of vectors and the inherent floating point grid takes over. Constants τ_r and τ_a are the relative and the absolute precision. The SANM algorithms uses $\tau_\mathrm{r} = 2^{-52}$ and $\tau_\mathrm{a} = 10^{-100}$, respectively (for 64-bit IEEE floating point $\tau_\mathrm{r} \geq 2^{-52}$ and $\tau_\mathrm{a} \geq 10^{-323}$).

When a trial step produces descent with respect to $\mathbf{x}^0$ the loop is abandoned, a new simplex is formed, and the algorithm returns to step 1.

The stopping condition is based on the size of the simplex and the range of the function values that correspond to the simplex points. Let $v^{i,j}$ and $x^{0,j}$ denote the jth component of v^i and x^0. The algorithm is stopped when $\max_{i=1,2,\ldots,n} |\tilde{f}^i - \tilde{f}^0| < \max(\beta_\mathrm{f}, \beta_\mathrm{r}|\tilde{f}^0|)$ and $\max_{i=1,2,\ldots,n} |v^{i,j}| < \max(\beta_\mathrm{x}, \beta_\mathrm{r}|x^{0,j}|)$ for $j = 1, 2, \ldots, n$. In the implementation the following values were used: $\beta_\mathrm{r} = 10^{-16}$, $\beta_\mathrm{x} = 10^{-9}$, and $\beta_\mathrm{f} = 10^{-16}$.

Algorithm 3. Successive approximation simplex algorithm:

(1) Repeat iterations of the original NM algorithm without shrink steps and with modified acceptance criteria for contraction points. Instead of $f(\mathbf{x})$ use its $(\mathcal{G}, \mathcal{P})$ approximation $\tilde{f}(\mathbf{x})$. When an iteration not replacing $\mathbf{x}^n$ (NM failure) is encountered, go to step 2.

(2) $\mathbf{x}^\mathrm{best} = \arg\min_{\mathbf{x}\in\{\mathbf{x}^0,\mathbf{x}^1,\ldots,\mathbf{x}^n\}} \tilde{f}(\mathbf{x})$ and $\tilde{f}^\mathrm{best} = \tilde{f}(\mathbf{x}^\mathrm{best})$.

(3) If the simplex shape violates (19), reshape it by forming an orthogonal basis $\mathcal{D} = \{\mathbf{d}^1, \mathbf{d}^2, \ldots, \mathbf{d}^n\}$ subject to $\lambda n^{1/2}\|\mathbf{\Delta}\| \leq \|\mathbf{d}^i\| \leq \Lambda n^{1/2}\|\mathbf{\Delta}\|$ for all $i = 1, 2, \ldots, n$. Construct a simplex comprising $\mathbf{x}^0$ and $\mathbf{x}^i = \mathbf{x}^0 + \mathbf{d}^i$ where $i = 1, 2, \ldots, n$, and evaluate $\tilde{f}$ at the new simplex vertices.

(4) Order the simplex and evaluate $\tilde{f}$ at the pseudo-expand point to obtain $\tilde{f}^{\mathrm{pe}} = \tilde{f}(\mathbf{x}^{\mathrm{pe}})$. If $\min(\tilde{f}^{\mathrm{pe}}, \tilde{f}^0, \tilde{f}^1, \ldots, \tilde{f}^n) \geq \tilde{f}^{\mathrm{best}}$ go to step 7
(5) If $\tilde{f}^{\mathrm{pe}} < \tilde{f}^0$ replace $\mathbf{x}^0$ with $\mathbf{x}^{\mathrm{pe}}$.
(6) Go to step 1.
(7) If a reshape happened at step 3 set $l = 1$, otherwise set $l = 0$ and reshape the simplex now.
(8) Repeat the following steps.
  (a) If $l > 0$, reverse vectors $\mathbf{d}^i$.
  (b) If $l \geq 2$ and $l \bmod 2 = 0$
    • Shrink vectors $\mathbf{d}^i$ by a factor of $0 < \gamma_{\mathrm{s}} < 1$.
    • If $\|\mathbf{d}^{\min}\| < \lambda n^{1/2}\|\boldsymbol{\Delta}\|$ choose the new grid origin $\mathbf{z}$ at $\mathcal{R}_{\mathcal{G},\mathcal{P}}(\mathbf{x}^0)$, set $\mathbf{x}^0$ to $\mathbf{z}$, and refine the partitioning $\mathcal{P}$ and the grid $\mathcal{G}$ (implies a new $\tilde{f}$).
  (c) Evaluate $\tilde{f}$ at $\mathbf{x}^0 + \mathbf{d}^i$ for $i = 1, 2, \ldots, n$.
  (d) Set $l = l + 1$.
  Until stopping condition is satisfied or $\min_{\mathbf{d}\in\mathcal{D}} \tilde{f}(\mathbf{x}^0 + \mathbf{d}) < \tilde{f}(\mathbf{x}^0)$.
(9) Construct a new simplex comprising $\mathbf{x}^0$ and $\mathbf{x}^0 + \mathbf{d}^i$ where $i = 1, 2, \ldots, n$.
(10) If stopping condition is satisfied finish, else go to step 1.

Theorem 17. *Suppose that the stopping condition is removed from Algorithm 3 and infinite numerical precision is available. Then for a continuously differentiable f with compact level sets the algorithm converges to a set of points $\mathbf{x}$ for which $\|\nabla f(\mathbf{x})\| = 0$.*

Proof. All we need to do is prove that Algorithm 3 adheres to the framework specified by Algorithm 1, assumptions A1–A3 are satisfied, and ξ_k goes to 0. Assumption A4 is satisfied by theorem's requirements.

Steps 1–6 correspond to steps 2–3 of the framework. The rest corresponds to steps 4–6. The set $\{\mathbf{x}^0, \mathbf{x}^0 + \mathbf{d}^1, \mathbf{x}^0 - \mathbf{d}^1, \mathbf{x}^0 + \mathbf{d}^2, \mathbf{x}^0 - \mathbf{d}^2, \ldots\}$ is constructed in two consecutive iterations of loop at step 8 of Algorithm 3 before the grid and the partitioning are refined. This set is the constellation that leads to step 6 of the framework.

The nature of the partitioning (uniform box-shaped partitions) makes sure that assumption A1 is satisfied. Since $\mathcal{D}$ is a linear orthogonal basis for $\mathbb{R}^n$ the set $\{\mathbf{d}^1, -\mathbf{d}^1, \mathbf{d}^2, -\mathbf{d}^2, \ldots\}$ positively spans $\mathbb{R}^n$ with $\epsilon = 1/n^{1/2}$. Since $\lambda n^{1/2}\|\Delta\| \leq \|\mathbf{d}\| \leq \Lambda n^{1/2}\|\Delta\|$. We satisfy assumption A2 by choosing $\xi = n^{1/2}\|\Delta\|$. A2 also holds when the basis is shrunk in the loop at step 8.

Since $\mathcal{D}$ is an orthogonal linear basis $\epsilon_k = n^{-1/2}$. The grid-restrainment error is $\delta = \|\Delta\|/2$. We can deduce $\epsilon_k\|\mathbf{b}\| - 2\delta/\xi = (\|\mathbf{b}\| - 1)/n^{1/2} \geq (\lambda - 1)/n^{1/2} = \alpha$ and see that assumption A3 is satisfied for all $\lambda > 1$.

Recall that the components of the new grid scaling vector are chosen as $\Delta^i = \frac{1}{128\lambda n}\max(|d^{\min,i}|, \frac{\|\mathbf{d}^{\min}\|}{n^{1/2}})$. From $|\mathrm{d}^{\min,i}| \leq \|\mathbf{d}^{\min}\|$ we can estimate that $\Delta^{\mathrm{new},i} \leq \|\mathbf{d}^{\min}\|/(128\lambda n)$. The grid is refined when $\|\mathbf{d}^{\min}\| < \lambda n^{1/2}\|\Delta\|$. This means that $\Delta^{\mathrm{new},i} < \|\Delta\|/(128n^{1/2})$ and $\|\Delta^{\mathrm{new}}\| < \|\Delta\|/128$ which in the end result in $\|\Delta\| \to 0$. □

A short note may be appropriate here. Remember from (12) that $\alpha < \Lambda$ must hold. Since $\alpha = (\lambda - 1)/n^{1/2}$ and $\lambda < \Lambda$ this is true, indeed.

3.2. The GRNM simplex algorithm in the proposed framework

The GRNM algorithm [4] also conforms to the presented framework. The part that guarantees GRNM's convergence is the equivalent of loop at step 8 of Algorithm 3. The only difference is that in GRNM grid restrainment does not affect $\mathbf{x}^0$ (since $\mathbf{x}^0$ always lies on the grid). The proposed convergence analysis is still valid. If, however, we consider that grid restrainment is applied only to one endpoint of a vector (using Eq. (4) instead of Eq. (5)) assumption A3 can be relaxed to $\epsilon_k\|\mathbf{b}\| - \delta_k/\xi_k \geq \alpha$.

3.3. Results of numerical testing

The sufficient descent-based simplex algorithm (SDNM) [19,5], the GRNM algorithm [4], and Algorithm 3 (SANM) were implemented in MATLAB R14. The Moré–Garbow–Hillstrom set of test functions [17] was used for algorithm evaluation. Besides these functions the standard quadratic and McKinnon [16] functions were also used. The starting simplex was chosen in the same manner as in [19,5], except for the McKinnon (alt.) function where

Table 1
Comparison of SDNM, GRNM, and SANM on the Moré–Garbow–Hillstrom set of test problems

Function	n	SDNM		GRNM		SANM	
		NF	Minimum	NF	Minimum	NF	Minimum
Rosenbrock	2	285	1.39058e−17	517	1.79285e−17	538	8.556045e−20
Freudenstein and Roth	2	217	48.9843	274	48.9843	333	48.9843
Powell badly scaled	2	969	4.23980e−25	1 245	1.87891e−25	1 464	4.44784e−25
Brown badly scaled	2	498	7.99797e−17	595	4.45581e−17	579	2.32382e−12
Beale	2	191	2.07825e−18	183	1.13556e−18	151	0.00000
Jennrich and Sampson	2	157	124.362	149	124.362	228	124.362
McKinnon	2	426	−0.250000	380	−0.250000	231	−0.250000
McKinnon (alt)	2	351	−0.250000	210	−0.250000	103	−0.250000
Helical valley	3	342	9.83210e−16	591	1.64083e−16	497	5.67580e−19
Bard	3	1 134	17.4287	427	8.21488e−3	407	8.21488e−3
Gaussian	3	194	1.12793e−8	252	1.12793e−8	244	1.12793e−8
Meyer	3	2 801	87.9459	7 269	87.9459	4 066	87.9459
Gulf research	3	529	5.44511e−19	955	2.92451e−21	937	2.90829e−23
Box	3	478	8.70459e−21	923	1.91130e−20	498	1.60807e−20
Powell singular	4	1045	6.73509e−26	1 280	3.43198e−25	2 104	2.35132e−32
Wood	4	656	2.57400e−16	1 177	2.50092e−17	1 102	7.14988e−19
Kowalik and Osborne	4	653	3.07506e−4	566	3.07506e−4	638	3.07506e−4
Brown and Dennis	4	603	85 822.2	620	85 822.2	683	85 822.2
Quadratic	4	440	2.15350e−17	427	2.82657e−17	301	0.00000
Penalty (1)	4	1 848	2.24998e−5	1 596	2.24998e−5	2 837	2.24998e−5
Penalty (2)	4	4 689	9.37629e−6	2 274	9.37629e−6	3 137	9.37629e−6
Osborne (1)	5	1 488	5.46489e−5	1 766	5.46489e−5	1 798	5.46490e−5
Brown almost linear	5	648	1.08728e−18	769	4.03372e−18	1 062	1.43221e−20
Biggs EXP6	6	4 390	1.16131e−20	2 877	1.12896e−20	3 399	1.26952e−23
Extended Rosenbrock	6	3 110	1.35844e−14	2 345	9.06455e−18	1 977	1.94036e−19
Brown almost linear	7	1 539	1.51163e−17	1 473	4.83079e−18	1 446	9.24764e−20
Quadratic	8	1 002	8.07477e−17	1 124	1.96893e−16	1 189	5.24081e−19
Extended Rosenbrock	8	5 314	3.27909e−17	2 996	1.50285e−17	4 637	7.96193e−20
Variably dimensional	8	2 563	1.24784e−15	2 634	7.66228e−16	2 988	5.28934e−18
Extended Powell	8	7 200	6.43822e−24	7 014	1.63762e−25	6 586	3.06964e−29
Watson	9	5 256	1.39976e−6	5 394	1.39976e−6	6 266	1.39976e−6
Extended Rosenbrock	10	7 629	2.22125e−16	6 208	1.77981e−17	8 611	1.95807e−19
Penalty (1)	10	9 200	7.08765e−5	11 514	7.08765e−5	8 479	7.08765e−5
Penalty (2)	10	32 768	2.93661e−4	31 206	2.93661e−4	28 753	2.93661e−4
Trigonometric	10	2 466	2.79506e−5	1 521	1.49481e−16	2 116	2.79506e−5
Osborne (2)	11	6 416	0.0401377	3 263	0.0401377	3 723	0.0401377
Extended Powell	12	20 076	1.11105e−20	12 846	5.51619e−28	12 146	2.08459e−20
Quadratic	16	2 352	1.41547e−16	3 639	4.70425e−16	3 349	1.38958e−18
Quadratic	24	4 766	1.21730e−15	6 067	4.06413e−16	7 065	8.37327e−19

McKinnon's initial simplex, which causes the original NM algorithm to fail, was used. The results of the testing are in Table 1. NF denotes the number of function evaluations. The best (lowest) function value obtained by the algorithms is also listed.

The results were compared to the results of the sufficient descent-based algorithm [19,5] (SDNM) and the grid-restrained algorithm [4] (GRNM). For all test functions SANM found a stationary point of the test function. For the Bard 3D function SDNM and SANM found different stationary points. Similar happened for GRNM and SANM on the trigonometric function. The functions were excluded from the comparisons.

When SANM and SDNM are compared there were 15 functions for which algorithms obtained the same final function value. SDNM was better on 5 functions and SANM on 18 functions. If we compare the number of function evaluations, SDNM was better on 22 problems and SANM on 16.

If we consider both the number of function evaluations and the final function value we can say, that Algorithm A outperforms Algorithm B on some test problem if it obtains the same or better final function value with less function evaluations, or if it obtains a better final function value with the same number of evaluations. According

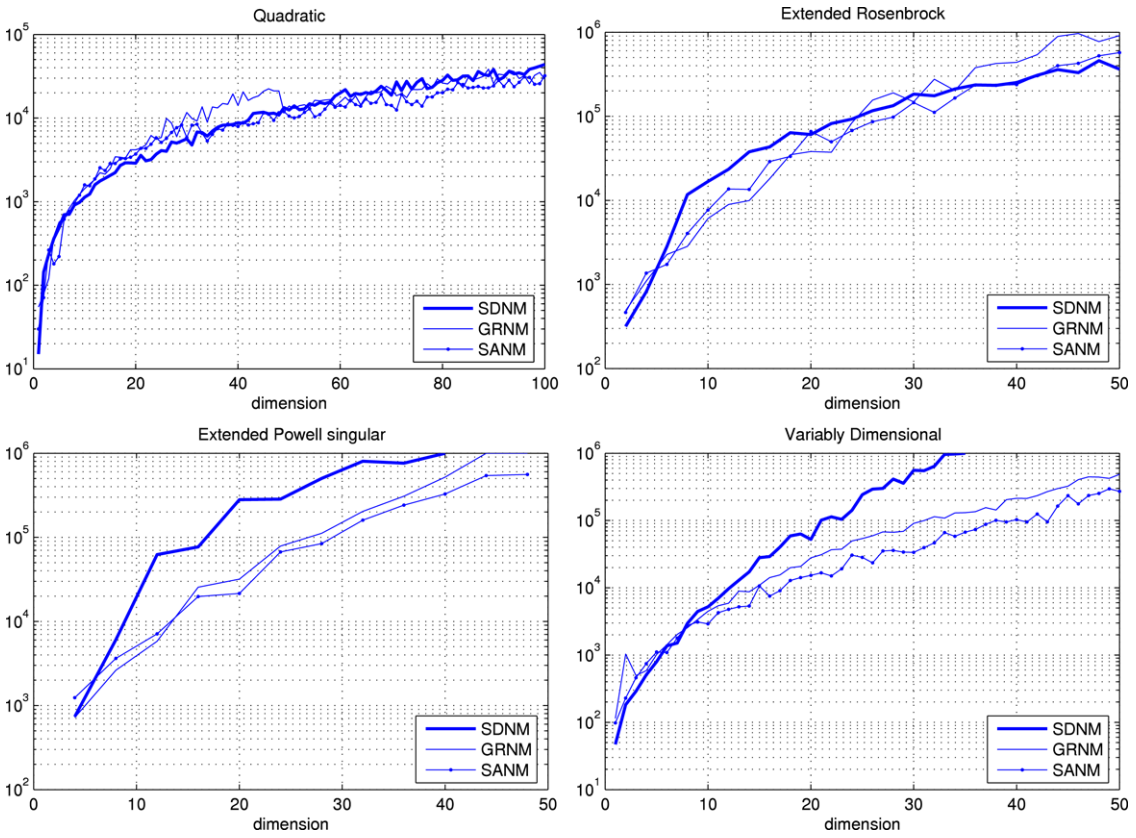

Fig. 1. Number of function evaluations for different problem dimensions.

to this criterion SDNM outperformed SANM on 11 test functions and SANM outperformed SDNM on 14 test functions.

On the other hand if we compare GRNM and SANM, we get the same final function value on 15 problems, GRNM is better on 4 problems and SANM on 19 problems. The result regarding the number of function evaluations is 19–19. GRNM outperforms SANM on 9 problems and SANM outperforms GRNM on 17 problems.

Judging from the two comparisons we conclude that SANM performs better than GRNM and SDNM on the test suite.

To compare the performance for problems of increasing dimensionality we tested all three algorithms on four variably-dimensional problems: the standard quadratic function (up to $n = 100$), the extended Rosenbrock's function, extended Powell's singular function, and variably-dimensional function (up to $n = 50$). All these functions have a global minimum where $f = 0$. The optimization was stopped when f reached values below 10^{-14} or when the number of function evaluations exceeded 10^6. The results (number of function evaluations) are plotted in Fig. 1.

For quadratic problems of lower dimension SDNM was slightly faster than the remaining two algorithms. For $n < 50$ the worst performance was exhibited by the GRNM algorithm. For problems of higher dimension ($n > 50$) GRNM performs similarly as SDNM and SANM is slightly faster than SDNM. On the extended Rosenbrock function GRNM and SANM outperform SDNM for problems with $n \leq 20$. For higher dimensions GRNM is slower and SANM performs roughly the same and SDNM. On the extended Powell function GRNM and SANM significantly outperform SDNM. SANM is slightly faster than GRNM. The situation is similar for the variably-dimensional function.

In general SANM performs better than GRNM on the four variably-dimensional test problems. We assume that this is due to the grid restrainment directly affecting the simplex shape in GRNM. Restraining the simplex vertices to a grid can significantly change the longest side of the simplex (which in general follows the function's gradient), especially when $\|\Delta\|$ is of the same order of magnitude as the simplex side length.

Based on the comparison of the Moré–Garbow–Hillstrom test suite results and the performance for higher-dimensional functions (Fig. 1) we conclude that the proposed SANM performs better than SDNM and GRNM.

4. Conclusion

We presented an algorithmic framework for unconstrained derivative-free optimization based on successive approximation. The framework relies on the grid restrainment and search space partitioning to ensure that a subsequence of iterates exists that converges to a local minimizer of a continuously differentiable function. The search space is divided into partitions that must satisfy some mild assumptions. Every partition has a representative point. The representative points form a grid which can be non-uniform in shape. If the partitions are bound in size from above and below the grid has a finite intersection with any compact set.

We defined a grid-restrainment operator that maps every point of a partition to the respective representative point. The norm of the difference between the original and the mapped point (grid-restrainment error) has an upper bound not greater than the partition diameter. The partitioning and the grid are used to construct a piecewise-constant approximation to function f.

The finiteness of the number of partitions that have a non-empty intersection with a given compact set guarantees that the search always reaches a point where no further progress can be made if the grid and the partitioning are not refined. We prove that a subsequence of points at which the refinement must happen, converges to some stationary point of f.

We defined a convergent variant of the Nelder–Mead (NM) algorithm within the proposed framework (SANM). In place of function f the algorithm utilizes its piecewise-constant approximation corresponding to the current grid and partitioning. This makes it possible to omit the grid restrainment of simplex vertices and simplifies the algorithm compared to the grid-restrained NM algorithm (GRNM) [4] since the determinant of the simplex sides can be calculated in a very straightforward manner. It also reduces the number of linear algebra operations compared to GRNM. We showed that GRNM conforms to the presented framework.

The proposed algorithm was tested on the Moré–Garbow–Hillstrom (MGH) set of test problems and on some multi-dimensional test problems. The results were compared to the results of the GRNM algorithm and the sufficient descent NM algorithm (SDNM) [19,5]. The proposed algorithm outperformed GRNM and SDNM.

We attribute the performance advantage of SANM over GRNM to the fact that SANM does not apply grid restrainment to simplex vertices. Grid restrainment can significantly change the direction of the longest simplex side which in general points in a descent direction. The effect of grid restrainment is most pronounced when the simplex side length is comparable to the grid-restrainment error.

The proposed framework is a powerful tool for defining new provably convergent optimization algorithms. Within its scope it is possible to easily upgrade the existing well-established heuristics that lack a convergence theory (like the Nelder–Mead algorithm) into new algorithms that can guarantee convergence to a local minimizer.

References

[1] C. Audet, J.E. Dennis Jr., Analysis of generalized pattern searches, SIAM J. Optim. 11 (2003) 859–869.
[2] C. Audet, J.E. Dennis Jr., Mesh adaptive direct search algorithms for constrained optimization, SIAM J. Optim. 17 (2006) 188–217.
[3] J.M. Borwein, A.S. Lewis, Convex Analysis and Nonlinear Optimization: Theory and Examples, Springer, 2000.
[4] Á. Bűrmen, J. Puhan, T. Tuma, Grid restrained Nelder–Mead algorithm, Comput. Optim. Appl. 34 (2006) 359–375.
[5] D. Byatt, A convergent variant of the Nelder-Mead algorithm, Master's Thesis, Mathematics and Statistics Department, University of Canterbury, Christchurch, NZ, 2000.
[6] F.H. Clarke, Optimization and Nonsmooth Analysis, Wiley, New York, 1983.
[7] I.D. Coope, C.J. Price, Frame based methods for unconstrained optimization, J. Optim. Theory Appl. 107 (2000) 261–274.
[8] I.D. Coope, C.J. Price, On the convergence of grid-based methods for unconstrained optimization, SIAM J. Optim. 11 (2001) 859–869.
[9] C.J. Price, I.D. Coope, Frames and grids in unconstrained and linearly constrained optimization: A nonsmooth approach, SIAM J. Optim. 14 (2003) 415–438.
[10] C. Davis, Theory of positive linear dependence, Amer. J. Math. 76 (1954) 733–746.
[11] R. Hooke, T.A. Jeeves, Direct search solution of numerical and statistical problems, J. ACM 8 (1961) 212–229.
[12] T.G. Kolda, R.M. Lewis, V. Torczon, Optimization by direct search: New perspectives on some classical and modern methods, SIAM Rev. 45 (2003) 385–482.
[13] J.C. Lagarias, J.A. Reeds, M.H. Wright, P.E. Wright, Convergence properties of the Nelder-Mead simplex method in low dimensions, SIAM J. Optim. 9 (1998) 112–147.

[14] R.M. Lewis, V.J. Torczon, Rank ordering and positive bases in pattern search algorithms, Tech. Report 96-71, ICASE NASA Langley Research Center, Hampton, VA, USA, 1996.
[15] R.M. Lewis, V.J. Torczon, M.W. Trosset, Direct search methods: Then and now, J. Comput. Appl. Math. 124 (2000) 191–207.
[16] K.I.M. McKinnon, Convergence of the Nelder-Mead simplex method to a non-stationary point, SIAM J. Optim. 9 (1998) 148–158.
[17] J.J. Moré, B.S. Garbow, K.E. Hillstrom, Testing unconstrained optimization software, ACM Trans. Math. Software 7 (1981) 17–41.
[18] J.A. Nelder, R. Mead, A simplex method for function minimization, Comput. J. 7 (1965) 308–313.
[19] C.J. Price, I.D. Coope, D. Byatt, A convergent variant of the Nelder-Mead algorithm, J. Optim. Theory Appl. 113 (2002) 5–19.
[20] V.J. Torczon, Multi-directional search: A direct search algorithm for parallel machines, Ph.D. Thesis, Department of Mathematical Sciences, Rice University, Houston, TX, USA, 1989.
[21] V.J. Torczon, On the convergence of pattern search methods, SIAM J. Optim. 7 (1997) 1–25.
[22] M.H. Wright, Direct search methods: Once scorned, now respectable, in: Griffiths D.F. (Ed.), Numerical Analysis, 1995, Addison Wesley Longman, Edinburgh Gate, Harlow, 1996, pp. 191–208.