Methodologies and Software for Derivative-free Optimization

A. L. Custódio [1] K. Scheinberg [2] L. N. Vicente [3]

March 14, 2017

[1]Department of Mathematics, FCT-UNL-CMA, Quinta da Torre, 2829-516 Caparica, Portugal (alcustodio@fct.unl.pt). Support for this author was provided by Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) under the project UID/MAT/00297/2013 (CMA) and the grant PTDC/MAT/116736/2010.

[2]Department of Industrial and Systems Engineering, Lehigh University, Harold S. Mohler Laboratory, 200 West Packer Avenue, Bethlehem, PA 18015-1582, USA (katyas@lehigh.edu). The work of this author is partially supported by NSF Grants DMS 10-16571, DMS 13-19356, AFOSR Grant FA9550-11-1-0239, and DARPA grant FA 9550-12-1-0406 negotiated by AFOSR.

[3]CMUC, Department of Mathematics, University of Coimbra, 3001-501 Coimbra, Portugal (lnv@mat.uc.pt). Support for this resarch was provided by FCT under grants PTDC/MAT/116736/2010 and UID/MAT/00324/2013.

37.1 ▪ Introduction

Derivative-Free Optimization (DFO) methods [53] are typically considered for the minimization/maximization of functions for which the corresponding derivatives are neither available for use, nor can be directly approximated by numerical techniques. Constraints may be part of the problem definition but, similar to the objective function, it is possible that their derivatives are not available. Problems of this type are common in engineering optimization, where the value of the functions is often computed by simulation and may be subject to statistical noise or other forms of inaccuracy. In fact, expensive function evaluations would prevent approximation to derivatives and, even when computed, noise would make such approximations less reliable. In the past couple of decades, intense research has resulted in robust and efficient DFO methods, accompanied by convergence theory and numerical implementations.

The purpose of the present work is to provide an overview of the main classes of state-of-the-art DFO methods, with a focus on the underlying ideas and on the respective classes of problems to which these methods are applicable. Only short descriptions of the methods and algorithms will be given, highlighting the motivational aspects that lead to their rigorous properties. We provide references to detailed algorithmic descriptions, theoretical results, and available software packages.

This chapter is structured around different problem features, rather than around classes of DFO methods as it was the case in [53]. Such a structure is more accessible to users of DFO as it directs the reader to the appropriate DFO algorithm suited for a given problem at hand.

Little notation or terminology needs to be introduced as the contents are given at a general level. However, we point out that by global convergence one means convergence to some form of stationarity regardless of the starting point. The vector norms will be ℓ_2 ones. The symbol $\mathcal{C}^k$ denotes the space of real n-dimensional functions whose derivatives are continuous up to the order k. The notation $\mathcal{O}(A)$ will mean a scalar times A, where the scalar does not depend on the iteration counter of the method under analysis (thus depending only on the problem or on algorithmic constants). The dependence of A on the dimension n of the problem will be made explicit whenever appropriate. The chapter is organized as follows. Section 37.2 covers unconstrained optimization. Bound and linearly constrained problems are addressed in Section 37.3. Section 37.4 is devoted to other types of problem constraints. Extensions to global optimization, multiobjective optimization, mixed integer problems, and some additional practical issues are briefly surveyed in Section 37.5.

37.2 ▪ Unconstrained optimization

37.2.1 ▪ Smooth functions

In this subsection we consider the unconstrained minimization of an objective function $f : \mathbb{R}^n \to \mathbb{R}$, at least once continuously differentiable and bounded from below (for which gradients are neither available for use, nor can be accurately approximated).

Sampling and modeling. At each iteration of a trust-region method [46], one

typically considers the minimization of a model $m_k(x_k+s) = f(x_k)+s^\top g_k+\frac{1}{2}s^\top H_k s$, in a region around the current iterate x_k, to obtain a trial point x_k+s_k. The region is frequently defined as a ball of the type $B(x_k;\Delta_k) = \{x_k + s \in \mathbb{R}^n : \|s\| \leq \Delta_k\}$, where Δ_k denotes the trust-region radius. The model m_k serves as a local approximation of the function, in particular of its curvature. The vector g_k can be set to $\nabla f(x_k)$ in the presence of first-order derivatives (similarly for H_k), but DFO trust-region methods are based on models built from sampling and some form of interpolation [133, 114, 47].

How well the model approximates the function is reflected by the ratio $\rho_k = [f(x_k) - f(x_k + s_k)] \,/\, [m_k(x_k) - m_k(x_k + s_k)]$. The algorithm proceeds by accepting the trial point $x_k + s_k$ when $\rho_k \geq \eta_0$ for some $\eta_0 > 0$. If $\rho_k < \eta_1$, with $\eta_1 \geq \eta_0$, then the quality of the model may be improved if not deemed sufficiently good, or, if the quality of the model is believed to be good, the trust-region radius is reduced since the step is then deemed to be too large. If x_k is non-stationary and m_k has good quality, the algorithm succeeds in accepting a trial point $x_k + s_k$ as a new iterate (at which the function value is improved) in a finite number of reductions of the trust-region radius Δ_k (see [53, Lemmas 10.6 and 10.17]).

In first-order approaches, the quality of a model is measured by its ability to provide accuracy similar to a first-order Taylor expansion:

$$\begin{array}{rcll} |f(y) - m_k(y)| & \leq & \kappa_f \, \Delta^2 & \\ \|\nabla f(y) - \nabla m_k(y)\| & \leq & \kappa_g \, \Delta & \forall y \in B(x_k;\Delta), \end{array}$$

where κ_f and κ_g are positive constants. Models that are $\mathcal{C}^1$ (with a Lipschitz continuous gradient) and satisfy the above bounds are called fully linear [50]. It was shown in [48] that a subsequence of the iterates generated by a model-based trust-region method drives the gradient to zero, under the condition that fully linear models are available when necessary. This result was further improved in [52] for the whole sequence of iterates, including the case where $\eta_0 = 0$, which means that any decrease in the function value is sufficient to accept a new point.

If convergence to second-order stationarity points is desired, then fully quadratic models [50] need to be considered. In this case the models should be $\mathcal{C}^2$ (with a Lipschitz continuous Hessian) and satisfy:

$$\begin{array}{rcll} |f(y) - m(y)| & \leq & \kappa_f \, \Delta^3 & \\ \|\nabla f(y) - \nabla m(y)\| & \leq & \kappa_g \, \Delta^2 & \\ \|\nabla^2 f(y) - \nabla^2 m(y)\| & \leq & \kappa_h \, \Delta & \forall y \in B(x;\Delta). \end{array}$$

Convergence to second-order stationary points is established in [52].

Building a (fully linear or fully quadratic) model based on a sample set raises questions related to the choice of the basis functions used in the model definition and to the geometry of the sample set. The use of polynomial models is quite attractive due to its simplicity, and in [50, 51] a first systematic approach to the subject of sampling geometry when using this class of functions was proposed (introducing the notion of Λ-poised sets, which is related to Lagrange polynomials and ensures fully linear or fully quadratic models). The strict need of controlling geometry or considering model-improvement steps was questioned in [70], where good numerical results were reported for an interpolation-based trust-region method (using complete quadratic models) which ignores the geometry of the sample sets. In [123] an example was given showing that geometry cannot be totally ignored and that some form of model improvement is necessary, at least when the size of the model gradient

becomes small (a procedure known as the criticality step, which then ensures that the trust-region radius converges to zero). In [123] an interpolation-based trust-region method was proposed which resorts to geometry-improving steps only when the model gradient is small. Global convergence for this method is the result of a self-correction property inherent in the combination of trust regions and polynomial interpolation models.

Quadratic functions are particularly well suited to capture curvature [53]. In a context of expensive function evaluation, construction of a complete quadratic model, which requires $(n+1)(n+2)/2$ functions evaluations, could be unaffordable. A typical approach is to consider minimum Frobenius norm models, which are commonly built when at least $n+1$ sampling points are available for use, allowing at least to compute a fully linear model. Some variants minimize the Frobenius norm of the model Hessian [49], since the norm of the model Hessian is connected with the accuracy of the model. Other approaches, inspired by quasi-Newton methods, use a least updating minimum Frobenius norm strategy, by minimizing the difference between the current and the previous model Hessians [117]. The minimization of the ℓ_1-norm of the model Hessian has also been proposed to build accurate models from relatively small sample sets [30]. Inspired by the sparse solution recovery theory developed in compressed sensing, the underlying idea is to take advantage of the sparsity of the Hessian in cases where the sparsity structure is not known in advance. Algorithms to compute fully linear and fully quadratic models, in the context of polynomial interpolation or regression, can be found in [50, 51] (see also [53]).

An alternative to polynomial bases are radial basis functions (RBFs) [39, 115]. An RBF is defined by the composition of an univariate function and a function measuring the distance to a sample point. Thus, it is constant on a sphere and has a structure different from polynomials (more nonlinear; potentially more nonconvex). Models based on RBFs typically involve a linear polynomial tail and can be made fully linear. The use of RBFs in model-based trust-region methods was analyzed in [132].

Currently, several solvers implementing interpolation-based trust-region methods are available to the community. Quadratic polynomial models are in the heart of DFO [2] and NEWUOA [118] computational codes. In the first case, when the size of the sampling set is not large enough to build a complete quadratic interpolation model, minimum Frobenius norm models are computed. In contrast, NEWUOA [118] uses the least updating minimum Frobenius norm strategy, described above. Good numerical results on unconstrained problems were also reported for the BC-DFO code [76], an interpolation based trust-region method developed for bound constrained optimization (see Section 37.3 below). Models based on RBFs are implemented in ORBIT [131].

Sampling using simplex sets. In turn, direct-search methods use function values from sampling only to make algorithmic decisions, without explicit or implicit modeling of the function. However, the geometry of the sample sets continues to play a crucial role in the algorithmic design and convergence properties.

One possibility is to sample at the vertices of a simplex set, which are in number $n+1$, exactly as many points as required to build a fully linear model. The goal of each iteration in the well known Nelder-Mead algorithm [111] is to improve the worst vertex of a simplex, and for this purpose a number of operations are performed (reflection, expansion, outside contraction, inside contraction, and shrink). The various simplex operations allow the method to follow the curvature of the

function which explains its good performance in many problems.

However, all simplex operations but shrinks can deteriorate the simplex geometry (an evident example are the expansions), thus it becomes difficult to establish convergence for the original algorithm. In fact, a $\mathcal{C}^2$ strictly convex function has been constructed in [110] for $n = 2$ showing that the algorithm [111] fails to converge to the minimizer (by generating an infinite sequence of inside contractions). Convergence can be established for $n = 1$ (see [96] or Exercice 7 of Chapter 8 of [53]) and for $n = 2$ for functions where the Hessian is always positive definite and when no simplex expansions are allowed [95]. Modified variants have been proposed, yielding global convergence in $\mathbb{R}^n$, by including strategies like monitoring the simplex geometry and then possibly attempting a poll-type step (see below) together with using a sufficient decrease condition for accepting new points [127] (see the survey in [53]).

Numerical implementations of variants of the Nelder-Mead method can be found in [8] or in the Matrix Computation Toolbox [9] (see the function `NMSMAX`).

Sampling using positive spanning sets. Direct-search methods can also be of directional type, where the function is evaluated along directions in positive spanning sets [61]. (A positive spanning set (PSS) is a set of vectors that spans $\mathbb{R}^n$ with nonnegative coefficients.)

Typically, these methods evaluate the objective function at points of the form $x_k + \alpha_k d$, $d \in D_k$, where x_k represents the current iterate, α_k the current step size parameter, and D_k denotes a PSS. This procedure (called polling) is attempted with the goal of decreasing the current best function value. When only simple decrease is required, polling is successful if $f(x_k + \alpha_k d) < f(x_k)$, for some $d \in D_k$. Similarly to trust-region methods, several authors proposed the use of sufficient decrease strategies [105, 93], where success requires $f(x_k + \alpha_k d) < f(x_k) - \rho(\alpha_k)$, for some $d \in D_k$, and where $\rho(\cdot)$ represents a forcing function (namely a non-negative, non-decreasing function satisfying $\rho(t)/t \to 0$ when $t \to 0$). When no improvement is found, α_k is decreased. When polling is successful, α_k is kept constant or increased.

A property of a PSS essential for the minimization of a smooth function is that at least one of its vectors is a descent direction, regardless where the negative gradient is [55, 93]. Thus, unless the current iterate is already a first-order stationary point, the algorithm will succeed in finding a better point in a finite number of reductions of the step size. As in model-based trust-region methods, where the trust-region radius is guaranteed to converge to zero, in direct search a subsequence of step sizes will also converge to zero. In fact, imposing sufficient decrease promotes unsuccessful iterations with consequent reductions of the step size, and using the boundedness from below of the function, one can easily ensure convergence to zero for a subsequence of step sizes [93]. When only simple decrease is required, one has to implicitly keep a distance of the order of the step size among all iterates, and the typical way to achieve it is by generating PSSs such that all trial points lie in underlying integer lattices [64, 125, 22].

Using simple decrease and a finite number of PSSs through the iterations, it was proved in [125] that the gradient is driven to zero for a subsequence of the iterates. Such an algorithmic framework was improved, generalized, and analyzed in [22] and coined generalized pattern search (see also [13]). It was shown in [93] that an infinite number of PSSs can be used when sufficient decrease is imposed (an approach known as generating set search), as long as they are uniformly non-degenerate (meaning that their *cosine measure* [93] is bounded away from zero).

Polling can be opportunistic (when moving to the first point $x_k + \alpha_k d$ yielding the desired decrease) or complete (when the best of the points $x_k + \alpha_k d$ is taken in D_k and then compared with x_k). Doing complete polling leads to the convergence of the whole sequence of gradients to zero [93] (under the additional condition that the step size converges to zero, which occurs naturally when imposing sufficient decrease or if the step size is never increased). When polling is not complete, for instance when the first poll direction leading to descent is taken, the order of the poll directions has some influence in the numerical performance of the method (see [24, 60]).

Nowadays, several implementations of direct-search methods of directional type are available, such as DFL [1], HOPSPACK [5], NOMAD [10], and SID-PSM [12]. Even if most of these solvers offer additional features, polling is common to all of them.

37.2.2 ▪ Non-smooth functions

In the presence of non-smoothness, the cone of descent directions can be arbitrarily narrow (see the example provided in [93, Page 441]). Thus, the use of a finite number of PSSs may not guarantee the existence of a descent direction among the poll vectors, and can cause stagnation of the optimization process. This fact was the main motivation for considering more general sets of directions [54] (see also [18], where the motivation arose from a practical context).

To rigorously avoid stagnation and guarantee some form of convergence (as defined below), poll vectors must therefore be asymptotically dense in the unit sphere. When simple decrease is used, all the generated trial points are required to belong to integer lattices and Mesh Adaptive Direct Search (MADS) [24] offers a framework to do so while using infinitely many directions (and taking them from PSSs if desired). If sufficient decrease is imposed, then the computation of new points is free of rules, and the set of poll directions could be simply randomly generated in the unit sphere [130] (an approach here denoted by RdDS).

In the absence of smoothness, convergence can be established by proving the non-negativity of some form of generalized directional derivatives at a limit point of the sequence of iterates and along all normalized directions. To do so, the authors in [22, 24] proposed the use of Clarke [44] analysis for locally Lipschitz continuous functions. As a consequence of using asymptotically dense sets of directions, a hierarchy of convergence results was derived in [24], depending on the level of non-smoothness present in the function. More recently, using Rockafellar generalized directional derivatives [121], the convergence results were extended to discontinuous functions [130]. Second-order results can be found in [14].

Simplex gradients [90] have been suggested as a possibility to define directions of potential descent. A simplex gradient can be regarded as the gradient of a particular linear interpolation model, requiring the evaluation of the function in a simplex (and its quality as an approximation to the gradient in the continuous differentiable case is analyzed in [90, 53]). Simplex gradients are also a possibility to approximate a direction in the Clarke subdifferential [44], defined for Lipschitz continuous functions as the set $\partial f(x) = \{\zeta \in \mathbb{R}^n : f^\circ(x; d) \geq \zeta^\top d \text{ for all } d \in \mathbb{R}^n\}$, where $f^\circ(x; d)$ represents the Clarke generalized directional derivative at x along d (and its quality as an approximation to such generalized gradients was analyzed in [56]).

In practice, non-smooth functions are frequently non-smooth compositions of

smooth functions. Lower-$\mathcal{C}^k$ functions [122], for instance, are characterized by being locally given as a maximum of $\mathcal{C}^k$ functions. Convex functions are lower-$\mathcal{C}^2$ [122]. Trivially, $f = \max\{f_1, \ldots, f_m\}$ is a lower-$\mathcal{C}^k$ function, provided that each f_i is $\mathcal{C}^k$. In [85] (see references therein) minmax problems of this form have been addressed, when the f_i's are $\mathcal{C}^1$ functions, by considering simplex gradients as approximations to generalized gradients in a line search approach. The general lower-$\mathcal{C}^2$ case was considered in [32], adapting ideas from convex non-smooth optimization.

Another possibility for optimizing a non-smooth function without derivatives is by approximating it by a family of smoothing functions (see [74, 92, 113]). The smoothing functions typically depend on a parameter, which must be then driven asymptotically, and may require prior knowledge of the non-smooth structure of the function.

Regarding numerical implementations, NOMAD [10, 97] is a reference for non-smooth unconstrained DFO using direct search. In this solver, two different instances are available to build the asymptotically dense sets of directions in the unit sphere fulfilling the integer lattice requirements, namely the probabilistic LT-MADS [24] and the deterministic ORTHOMADS [16].

37.2.3 ▪ Noisy functions

Simplex gradients are also used as search directions for the optimization of noisy functions. In implicit filtering [36], a (not too refined) line search is performed along a negative simplex gradient. A quasi-Newton scheme is then used for curvature approximation. Such ingredients equip the method in [36] to noisy problems in the hope that it can escape from spurious minimizers. A detailed description of the algorithm and corresponding convergence results can be found in the recent book [91]. A numerical implementation, called IFFCO, is available at [6].

In the presence of noise it is natural to consider least-squares regression techniques (see Chapter 4 in [53]) and use them in trust-region methods. However, when the level of noise is large, this type of models may over-fit the available data. In [89], assuming the knowledge of an upper bound for the level of noise present in function evaluations, it was suggested to relax the interpolation conditions using the corresponding bound. In [34] it was suggested instead to incorporate the knowledge about the noise level from each function evaluation in a weighted regression. When the level of noise is sufficiently small relatively to the trust radius, trust-regions methods based on weighted regression models retain global convergence to stationary points [34].

If the noise present in the function evaluation has a stochastic nature, then a simple possible approach would be to replicate function evaluations performed at each point, conferring accuracy to the estimation of the real corresponding function value. This procedure as been followed to adapt simplex type methods [20] and interpolation-based trust-region methods [62] to noisy optimization. Recently, in the context of direct search using PSSs [43], replication techniques were also applied to smooth and non-smooth functions computed by Monte Carlo simulation. Statistical based approaches, namely by using hypotheses tests, have also been suggested for providing confidence to the decision of accepting a new point, when using direct search in the presence of stochastic noise [126, 124].

37.2.4 ▪ Worst case complexity and global rates

The analysis of global convergence of algorithms can be complemented or refined by deriving worst case complexity (WCC) bounds for the number of iterations or function evaluations, an information which may be valuable in many practical instances. Derivative-free or zero-order methods have also been recently analyzed with the purpose of establishing their WCC bounds. As in gradient-based methods (see [112, 78, 40]), it was shown in [129] a WCC bound of $\mathcal{O}(\epsilon^{-2})$ for the number of iterations of direct-search methods (using PSSs and imposing sufficient decrease), when applied to a smooth, possibly non-convex function. Such type of bound translates into a sublinear global rate of $1/\sqrt{k}$ for the decay of the norm of the gradient. Note that these rates are called global since they are obtained independently of the starting point. In DFO it becomes also important to measure the effort in terms of the number of function evaluations: the corresponding WCC bound for direct search is $\mathcal{O}(n^2\epsilon^{-2})$. DFO trust-region methods achieve similar bounds and rates [73]. The authors in [41] have derived a better WCC bound of $\mathcal{O}(n^2\epsilon^{-3/2})$ for their adaptive cubic overestimation algorithm but using finite differences to approximate derivatives.

In the non-smooth case, using smoothing techniques, it was established a WCC bound of $\mathcal{O}\left((-\log(\epsilon))\epsilon^{-3}\right)$ iterations (and $\mathcal{O}\left(n^3(-\log(\epsilon))\epsilon^{-3}\right)$ function evaluations) for the zero-order methods in [73, 74, 113], where the threshold ϵ refers now to the gradient of a smoothed version of the original function and the size of the smoothing parameter. Composite DFO trust-region methods [73] can achieve $\mathcal{O}(\epsilon^{-2})$ when the non-smooth part of the composite function is known.

In [112, Section 2.1.5] it is also shown that the gradient method achieves an improved WCC bound of $\mathcal{O}(\epsilon^{-1})$ if the function is convex and the solutions set is nonempty. Correspondingly, the global decaying rate for the gradient is improved to $1/k$. Due to convexity, the rate $1/k$ holds also for the error in function values. For derivative-free optimization, direct search [67] attains the $\mathcal{O}(\epsilon^{-1})$ bound ($\mathcal{O}(n^2\epsilon^{-1})$ for function evaluations) and the global rate of $1/k$ in the convex (smooth) case. As in the gradient method, direct search achieves an r-linear rate of convergence in the strongly convex case [67]. The analysis can be substantially simplified when direct search does not allow an increase in the step size (see [94]).

The factor of n^2 has been proved to be approximately optimal, in a certain sense, in the WCC bounds for the number of function evaluations attained by direct search (see [68]).

37.2.5 ▪ Models and descent of probabilistic type

The development of probabilistic models in [30] for DFO and the benefits of randomization for deterministic first-order optimization, led to the consideration of trust-region methods where the accuracy of the models is given with some positive probability [31]. It has been shown that provided the models are fully linear with a certain probability, conditioned to the prior iteration history, the gradient of the objective function converges to zero with probability one. In this trust-region framework, if $\rho_k \geq \eta_0 > 0$ and the trust-region radius is sufficiently small relatively to the size of the model gradient g_k, then the step is taken and the trust-region radius is possibly increased. Otherwise the step is rejected and the trust-region radius is decreased. It is shown in [31] that global convergence to second-order stationary points is also attainable almost surely.

Not surprisingly, one can define descent in a probabilistic way similarly as for fully linear models. A set of directions is *probabilistically descent* if at least one of them makes an acute angle with the negative gradient with a certain probability. Direct search based on probabilistic descent has been proved globally convergent with probability one [77]. Polling based on a reduced number of randomly generated directions (which can go down to two) satisfies the theoretical requirements [77] and can provide numerical results that compare favorable to the traditional use of PSSs.

It has been proved in [77] that both probabilistic approaches (for trust regions and direct search) enjoy, with overwhelmingly high probability, a gradient decaying rate of $1/\sqrt{k}$ or, equivalently, that the number of iterations taken to reach a gradient of size ϵ is $\mathcal{O}(\epsilon^{-2})$. Interestingly, the WCC bound in terms of function evaluations for direct search based on probabilistic descent is reduced to $\mathcal{O}(nm\epsilon^{-2})$, where m is the number of random poll directions [77].

Recently, in [42], it was proposed and analyzed a trust-region model-based algorithm for solving unconstrained stochastic optimization problems, using random models obtained from stochastic observations of the objective function or its gradient.

37.3 ▪ Bound and linearly constrained optimization

We now turn our attention to linearly constrained optimization problems in which $f(x)$ is minimized subject to $b \leq Ax \leq c$, where A is a $m \times n$ matrix and b and c are m-dimensional vectors. The inequalities are understood componentwise. In particular, if A is the identity matrix, then we have a bound constrained optimization problem. Again, we consider the derivative-free context, where it is not possible to evaluate derivatives of f.

Sampling along directions. In a feasible method, where all iterates satisfy the constraints, the geometry of the boundary near the current iterate should be taken into account when computing search directions (to allow for sufficiently long feasible displacements). In direct search this can be accomplished by computing sets of positive generators for tangent cones of nearby points, and then using them for polling. (A set of positive generators of a convex cone is a set of vectors that spans the cone with nonnegative coefficients.) If there are only bounds on the variables, such a scheme is ensured simply by considering all the coordinate directions [98]. For general non-degenerate linear constraints, there are schemes to compute such positive generators [100] (for the degenerate case see [17]). If the objective function is continuously differentiable, the resulting direct-search methods are globally convergent to first-order stationary points [100] (see also [93]), in other words, to points where the gradient is in the polar of the tangent cone, implying that the directional derivative is nonnegative for all directions in the tangent cone. Implementations are given in HOPSPACK [5] and PSwarm [11].

If the objective function is non-smooth, one has to use polling directions asymptotically dense in the unit sphere (for which there are two main techniques, either MADS [24] or RdDS [130]). We have seen that in unconstrained optimization global convergence is attained by proving that the Clarke generalized derivative is nonnegative at a limit point for all directions in $\mathbb{R}^n$ — which, in the presence of bounds/linear constraints, trivially includes all the directions of the tangent cone at the limit point. One can also think of hybrid strategies, combining positive generators and dense generation (see the algorithm CS-DFN [69] for bound constrained

optimization where the coordinate directions are enriched by densely generated ones when judged efficient).

Sampling and modeling. Active-set type approaches have also been considered in the context of trust-region methods for derivative-free bound constrained optimization. One difficulty is that the set of interpolation points may get aligned at one or more active bounds and deteriorate the quality of the interpolation set. In [76] an active-set strategy is considered by pursuing minimization in the subspace of the free (non-active) variables, circumventing such a difficulty and saving function evaluations from optimization in lower dimensional subspaces. The respective code is called BC-DFO [76].

In other strategies, all the constraints are included in the trust-region subproblem. This type of trust-region methods was implemented in the codes BOBYQA [119] (a generalization of NEWUOA [118] for bound constrained optimization) and DFO [2] (which also considers feasible regions defined by continuously differentiable functions for which gradients can be computed). Recently, extensions to linearly constrained problems have been provided in the codes LINCOA [120] and LCOBYQA [83].

37.4 ▪ Nonlinearly constrained optimization

Consider now the more general constrained problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in \Omega = \Omega_r \cap \Omega_{nr}. \end{array} \tag{37.1}$$

The feasible region of this problem is defined by relaxable and/or unrelaxable constraints. The non-relaxable constraints correspond to $\Omega_{nr} \subseteq \mathbb{R}^n$. Such constraints have to be satisfied at all iterations in an algorithmic framework for which the objective function is evaluated. Often they are bounds or linear constraints, as considered above, but they can also include hidden constraints (constraints which are not part of the problem specification/formulation and their manifestation comes in the form of some indication that the objective function could not be evaluated). In contrast, relaxable constraints, corresponding to $\Omega_r \subseteq \mathbb{R}^n$, need only to be satisfied approximately or asymptotically, and are often defined by algebraic inequality constraints.

Most of the globally convergent derivative-free approaches for handling nonlinearly constrained problems have been of direct search or line search type, and we summarize such activity next.

Unrelaxable constraints. Feasible methods may be the only option when all the constraints are unrelaxable ($\Omega_r = \mathbb{R}^n$). In addition they generate a sequence of feasible points, thus allowing the iterative process to be terminated prematurely with a guarantee of feasibility for the best point tested so far. This is an important feature in engineering design problems because the engineer does not want to spend a large amount of computing time and have nothing useful (i.e., feasible) to show for it. One way of designing feasible methods is by means of the barrier function (coined extreme barrier in [24])

$$f_{\Omega_{nr}}(x) \;=\; \begin{cases} f(x) & \text{if } x \in \Omega_{nr}, \\ +\infty & \text{otherwise.} \end{cases}$$

It is not necessary to evaluate f at infeasible points where the value of the extreme barrier function can be set directly to $+\infty$. Hidden constraints are fundamentally different because it is not known a priori if the point is feasible. Direct-search methods take action solely based on function value comparisons and are thus appropriate to use in conjunction with an extreme barrier function. In the context of direct-search methods of directional type for non-smooth functions, we have seen that there are two known ways of designing globally convergent algorithms (MADS [24] and RdDS [130]). In each case, one must use sets of directions whose union (after normalization if needed) is asymptotically dense in the unit sphere of $\mathbb{R}^n$. The resulting approaches are then globally convergent to points where the Clarke directional derivative is nonnegative along all directions in the (now unknown) tangent cone. An alternative to extreme barrier when designing feasible methods is the use of projections onto the feasible set, although this might require the knowledge of the derivatives of the constraints and be expensive or unpractical in many instances (see [106] for such an approach).

Relaxable constraints. In the case where there are no unrelaxable constraints (rather than those of the type $b \leq Ax \leq c$), one can use a penalty term by adding to the objective function a measure of constraint violation multiplied by a penalty parameter, thus allowing starting points that are infeasible with respect to the relaxable constraints. In this vein, an approach based on an augmented Lagrangian method was suggested (see [99]), considering the solution of a sequence of subproblems where the augmented Lagrangian function takes into account only the nonlinear constraints and is minimized subject to the remaining ones (of the type $b \leq Ax \leq c$). Each problem can then be approximately solved using an appropriate DFO method such as a (directional) direct-search method. This application of augmented Lagrangian methods yields global convergence results to first-order stationary points of the same type of those obtained under the presence of derivatives. In [65] a more general augmented Lagrangian setting is studied, where the problem constraints imposed in the subproblems are not necessarily of linear type.

In turn, algorithms for inequality constrained problems, based on smooth and non-smooth penalty functions were developed and analyzed in [101, 103, 69], imposing sufficient decrease and handling bound/linear constraints separately, proving that a subset of the set of limit points of the sequence of iterates satisfy the first-order necessary conditions of the original problem. Numerical implementations can be found in the DFL library [1].

Filter methods from derivative-based optimization [72] have also been used in the context of relaxable constraints in DFO. In a simplified way, these methods treat a constrained problem as a bi-objective unconstrained one, considering as goals the objective function and a measure of the constraints violation, but giving priority to the latter one. Typically a restoration procedure is considered to compute nearly feasible points. A first step along this direction in DFO was suggested in [23], for direct-search methods using a finite number of PSSs. The filter approach in [63] (where an envelope around the filter is used as a measure of sufficient decrease) guarantees global convergence to a first-order stationary point. Inexact restoration methods from derivative-based optimization [107] have also been applied to DFO, again algorithms alternating between restoration and minimization steps. In [108] an algorithm is proposed for problems with 'thin' constraints, based on relaxing feasibility and performing a subproblem restoration procedure. Inexact restoration has been applied in [38] to optimization problems where derivatives of

the constraints are available for use, thus allowing derivative-based methods in the restoration phase.

Relaxable and unrelaxable constraints. The first general approach to consider both relaxable and unrelaxable constraints is called progressive barrier [25]. It allows the handling of both types of constraints, by combining MADS for unrelaxable constraints with non-dominance filter-type concepts for the relaxable constraints (see the consequent developments in [27]). An alternative to progressive barrier has been proposed in [79], handling the relaxable constraints by means of a merit function instead of a filter, and using RdDS for the unrelaxable ones. The merit function and the corresponding penalty parameter are only used in the evaluation of an already computed step. An interesting feature of these two approaches is that constraints can be considered relaxable until they become feasible whereupon they can be transferred to the set of unrelaxable constraints. Both of them exhibit global convergence properties.

Model-based trust-region methods. On the model-based trust-region side of optimization without derivatives, nonlinear constraints have been considered mostly in implementations and in a relaxable mode.

Two longstanding software approaches are COBYLA [116] (where all the functions are modeled linearly by interpolation), see also [37], and DFO [49] (where all the functions are modeled quadratically by interpolation).

Another development avenue has been along composite-step based SQP [46, Section 15.4]. Here one models the objective function by quadratic functions and the constraints by linear ones. The first approach has been proposed in [45] and [33], using, respectively, filters and merit functions for step evaluation.

More recently, a trust-funnel method (where the iterates can be thought as flowing towards a critical point through a funnel centered on the feasible set; see [75]) was proposed in [109] for the particular equality constrained case. Another approach (and implementation code NOWPAC) has been proposed in [29] for equalities and inequalities and inexact function evaluations.

37.5 ▪ General extensions

In real life applications, it is often the case that the user can supply a starting point for the optimization process and that some (local) improvement over the provided initialization may already fulfill the original goals. Nevertheless, there are situations where global minimizers are requested and/or good initial guesses are unknown. Extensions of DFO to global optimization try to cope with such additional difficulties. One possibility is to partition the feasible region into subdomains, which are locally explored by a DFO procedure in an attempt to identify the most promising ones. DIRECT [88] and MCS [87] follow this approach being the latter enhanced by local optimization based on quadratic polynomial interpolation (see the corresponding codes in [71] and [7]). An alternative is to multistart different instances of a DFO algorithm from distinct feasible points. Recently, in the context of direct search, it was proposed to merge the different starting instances when sufficiently close to each other [57] (see the corresponding code GLODS [4]). Heuristics have been tailored to global optimization without derivatives, and an example providing interesting numerical results are evolution strategies like CMA-ES [84] (for which a modified version is capable of globally converging to stationary points [66]).

DFO algorithms can be equipped with a search step for the purpose of improving their local or global performance (such type of steps are called *magical* in [46]). The paper [35] proposed a search-poll framework for direct search, where a search step is attempted before the poll step. A similar idea can be applied to model-based trust-region algorithms [80]. The search step is optional and does not interfere in the global convergence properties of the underlying methods. Surrogate models (see [104, Section 3.2] and [53, Section 12]) can be built and optimized in a search step such as in [59] for quadratics or in [19] for RBFs. Other possibilities for its use include the application of global optimization heuristics [128, 21]. See the various solvers [10, 11, 12].

Parallelizing DFO methods is desirable in the presence of expensive function evaluations. The poll step of direct search offers a natural parallelization by distributing the poll directions among processors [86]. Asynchronous versions of this procedure [82] are relevant in the presence of considerably different function evaluation times. Several codes [5, 10, 11] offer parallel modes. Subspace decomposition in DFO is also attractive for parallelization and surrogate building [26, 81].

The extension of DFO methods to problems involving integer or categorical variables has also been considered. The methodologies alternate between a local search in the continuous space and some finite exploration of discrete sets for the integer variables. Such discrete sets or structures could be fixed in advance [15] or be adaptively defined [102]. Implementations are available in NOMAD [10] and DFL library [1], respectively.

Multiobjective optimization has also been the subject of DFO. A common approach to compute Pareto fronts consists of aggregating all the functions into a single parameterized one, and it has been done in DFO (see [28] and references therein). In [58] the concept of Pareto dominance was used to generalize direct search to multiobjective DFO without aggregation. Implementations are available in the codes NOMAD [10] and DMS [3], respectively.

Bibliography

[1] DFL. http://www.dis.uniroma1.it/~lucidi/DFL.

[2] DFO. http://www.coin-or.org/projects.html.

[3] DMS. http://www.mat.uc.pt/dms.

[4] GLODS. http://ferrari.dmat.fct.unl.pt/personal/alcustodio/GLODS.htm.

[5] HOPSPACK. https://software.sandia.gov/trac/hopspack/wiki.

[6] IMPLICIT FILTERING. http://www4.ncsu.edu/~ctk/iffco.html.

[7] MCS. http://www.mat.univie.ac.at/~neum/software/mcs.

[8] NELDER-MEAD SIMPLEX. http://www4.ncsu.edu/~ctk/matlab_darts.html.

[9] NELDER-MEAD SIMPLEX. http://www.maths.manchester.ac.uk/~higham/mctoolbox.

[10] NOMAD. http://www.gerad.ca/nomad.

[11] PSwarm. http://www.norg.uminho.pt/aivaz/pswarm.

[12] SID-PSM. http://www.mat.uc.pt/sid-psm.

[13] M. A. Abramson. Second-order behavior of pattern search. *SIAM J. Optim.*, 16:515–530, 2005.

[14] M. A. Abramson and C. Audet. Convergence of mesh adaptive direct search to second-order stationary points. *SIAM J. Optim.*, 17:606–619, 2006.

[15] M. A. Abramson, C. Audet, J. W. Chrissis, and J. G. Walston. Mesh adaptive direct search algorithms for mixed variable optimization. *Optim. Lett.*, 3:35–47, 2009.

[16] M. A. Abramson, C. Audet, J. E. Dennis, Jr., and S. Le Digabel. OrthoMADS: A deterministic MADS instance with orthogonal directions. *SIAM J. Optim.*, 20:948–966, 2009.

[17] M. A. Abramson, O. A. Brezhneva, J. E. Dennis Jr., and R. L. Pingel. Pattern search in the presence of degenerate linear constraints. *Optim. Methods Softw.*, 23:297–319, 2008.

[18] P. Alberto, F. Nogueira, H. Rocha, and L. N. Vicente. Pattern search methods for user-provided points: Application to molecular geometry problems. *SIAM J. Optim.*, 14:1216–1236, 2004.

[19] Le Thi Hoai An, A. I. F. Vaz, and L. N. Vicente. Optimizing radial basis functions by D.C. programming and its use in direct search for global derivative-free optimization. *TOP*, 20:190–214, 2012.

[20] E. J. Anderson and M. C. Ferris. A direct search algorithm for optimization with noisy function evaluations. *SIAM J. Optim.*, 11:837–857, 2001.

[21] C. Audet, V. Béchard, and S. Le Digabel. Nonsmooth optimization through mesh adaptive direct search and variable neighborhood search. *J. Global Optim.*, 41:299–318, 2008.

[22] C. Audet and J. E. Dennis Jr. Analysis of generalized pattern searches. *SIAM J. Optim.*, 13:889–903, 2002.

[23] C. Audet and J. E. Dennis Jr. A pattern search filter method for nonlinear programming without derivatives. *SIAM J. Optim.*, 14:980–1010, 2004.

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

[25] C. Audet and J. E. Dennis Jr. A progressive barrier for derivative-free nonlinear programming. *SIAM J. Optim.*, 20:445–472, 2009.

[26] C. Audet, J. E. Dennis Jr., and S. Le Digabel. Parallel space decomposition of the mesh adaptive direct search algorithm. *SIAM J. Optim.*, 19:1150–1170, 2008.

[27] C. Audet, J. E. Dennis Jr., and S. Le Digabel. Globalization strategies for mesh adaptive direct search. *Comput. Optim. Appl.*, 46:193–215, 2010.

[28] C. Audet, G. Savard, and W. Zghal. A mesh adaptive direct search algorithm for multiobjective optimization. *European J. Oper. Res.*, 204:545–556, 2010.

[29] F. Augustin and Y. M. Marzou. NOWPAC: A provably convergent nonlinear optimizer with path-augmented constraints for noisy regimes. Technical Report arXiv:1403.1931v1, 2014.

[30] 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.

[31] 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.

[32] H. H. Bauschke, W. L. Hare, and W. M. Moursi. A derivative-free comirror algorithm for convex optimization. *Optim. Methods Softw.*, 2015, to appear.

[33] F. V. Berghen. *CONDOR: A Constrained, Non-Linear, Derivative-Free Parallel Optimizer for Continuous, High Computing Load, Noisy Objective Functions*. PhD thesis, Université Libre de Bruxelles, 2004.

[34] S. C. Billups, J. Larson, and P. Graf. Derivative-free optimization of expensive functions with computational error using weighted regression. *SIAM J. Optim.*, 23:27–53, 2013.

[35] A. J. Booker, J. E. Dennis Jr., P. D. Frank, D. B. Serafini, V. Torczon, and M. W. Trosset. A rigorous framework for optimization of expensive functions by surrogates. *Structural and Multidisciplinary Optimization*, 17:1–13, 1998.

[36] D. M. Bortz and C. T. Kelley. The simplex gradient and noisy optimization problems. In J. T. Borggaard, J. Burns, E. Cliff, and S. Schreck, editors, *Computational Methods in Optimal Design and Control, Progress in Systems and Control Theory*, volume 24, pages 77–90. Birkhäuser, Boston, 1998.

[37] Ruud Brekelmans, L. Driessen, H. Hamers, and D. den Hertog. Constrained optimization involving expensive function evaluations: A sequential approach. *European J. Oper. Res.*, 160:121–138, 2005.

[38] L. F. Bueno, A. Friedlander, J. M. Martínez, and F. N. C. Sobral. Inexact restoration method for derivative-free optimization with smooth contraints. *SIAM J. Optim.*, 23:1189–1213, 2013.

[39] M. D. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, Cambridge, 2003.

[40] C. Cartis, N. I. M. Gould, and Ph. L. Toint. On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization. *SIAM J. Optim.*, 20:2833–2852, 2010.

[41] C. Cartis, N. I. M. Gould, and Ph. L. Toint. On the oracle complexity of first-order and derivative-free algorithms for smooth nonconvex minimization. *SIAM J. Optim.*, 22:66–86, 2012.

[42] R. Chen, M. Menickelly, and K. Scheinberg. Stochastic optimization using a trust-region method and random models. Technical Report ISE 15T-002, Dept. Industrial and Systems Engineering, Lehigh University, 2015.

[43] X. Chen and C. T. Kelley. Sampling methods for objective functions with embedded Monte Carlo simulations. Technical report, 2014.

[44] F. H. Clarke. *Optimization and Nonsmooth Analysis*. John Wiley & Sons, New York, 1983. Reissued by SIAM, Philadelphia, 1990.

[45] B. Colson. *Trust-Region Algorithms for Derivative-Free Optimization and Nonlinear Bilevel Programming*. PhD thesis, Département de Mathématique, FUNDP, Namur, 2003.

[46] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust-Region Methods*. MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2000.

[47] A. R. Conn and Ph. L. Toint. An algorithm using quadratic interpolation for unconstrained derivative free optimization. In G. Di Pillo and F. Gianessi, editors, *Nonlinear Optimization and Applications*, pages 27–47. Plenum Publishing, New York, 1996.

[48] A. R. Conn, K. Scheinberg, and Ph. L. Toint. On the convergence of derivative-free methods for unconstrained optimization. In M. D. Buhmann and A. Iserles, editors, *Approximation Theory and Optimization, Tributes to M. J. D. Powell*, pages 83–108. Cambridge University Press, Cambridge, 1997.

[49] A. R. Conn, K. Scheinberg, and Ph. L. Toint. A derivative free optimization algorithm in practice. In *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, Missouri, September 2-4*, 1998.

[50] A. R. Conn, K. Scheinberg, and L. N. Vicente. Geometry of interpolation sets in derivative free optimization. *Math. Program.*, 111:141–172, 2008.

[51] A. R. Conn, K. Scheinberg, and L. N. Vicente. Geometry of sample sets in derivative free optimization: Polynomial regression and underdetermined interpolation. *IMA J. Numer. Anal.*, 28:721–748, 2008.

[52] 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.

[53] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-Free Optimization*. MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2009.

[54] I. D. Coope and C. J. Price. On the convergence of grid-based methods for unconstrained optimization. *SIAM J. Optim.*, 11:859–869, 2001.

[55] I. D. Coope and C. J. Price. Positive basis in numerical optimization. *Comput. Optim. Appl.*, 21:169–175, 2002.

[56] A. L. Custódio, J. E. Dennis Jr., and L. N. Vicente. Using simplex gradients of nonsmooth functions in direct search methods. *IMA J. Numer. Anal.*, 28:770–784, 2008.

[57] A. L. Custódio and J. F. A. Madeira. GLODS: Global and Local Optimization using Direct Search. *J. Global Optim.*, 62:1–28, 2015.

[58] A. L. Custódio, J. F. A. Madeira, A. I. F. Vaz, and L. N. Vicente. Direct multisearch for multiobjective optimization. *SIAM J. Optim.*, 21:1109–1140, 2011.

[59] A. L. Custódio, H. Rocha, and L. N. Vicente. Incorporating minimum Frobenius norm models in direct search. *Comput. Optim. Appl.*, 46:265–278, 2010.

[60] A. L. Custódio and L. N. Vicente. Using sampling and simplex derivatives in pattern search methods. *SIAM J. Optim.*, 18:537–555, 2007.

[61] C. Davis. Theory of positive linear dependence. *Amer. J. Math.*, 76:733–746, 1954.

[62] G. Deng and M. C. Ferris. Adaptation of the UOBYQA algorithm for noisy functions. In L. F. Perrone, F. P. Weiland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, editors, *Proceedings of the 2006 Winter Simulation Conference*, pages 312–319, 2006.

[63] J. E. Dennis Jr., C. J. Price, and I. D. Coope. Direct search methods for nonlinearly constrained optimization using filters and frames. *Optim. Eng.*, 5:123–144, 2004.

[64] J. E. Dennis Jr. and V. Torczon. Direct search methods on parallel machines. *SIAM J. Optim.*, 1:448–474, 1991.

[65] M. A. Diniz-Ehrhardt, J. M. Martínez, and L. G. Pedroso. Derivative-free methods for nonlinear programming with general lower-level constraints. *Comput. Appl. Math.*, 30:19–52, 2011.

[66] Y. Diouane, S. Gratton, and L. N. Vicente. Globally convergent evolution strategies. *Math. Program.*, 2015, to appear.

[67] M. Dodangeh and L. N. Vicente. Worst case complexity of direct search under convexity. *Math. Program.*, 2015, to appear.

[68] M. Dodangeh, L. N. Vicente, and Z. Zhang. On the optimal order of worst case complexity of direct search. *Optim. Lett.*, 2015, to appear.

[69] 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.

[70] G. Fasano, J. L. Morales, and J. Nocedal. On the geometry phase in model-based algorithms for derivative-free optimization. *Optim. Methods Softw.*, 24:145–154, 2009.

[71] D. E. Finkel. *DIRECT Optimization Algorithm User Guide*, 2003. http://www4.ncsu.edu/ definkel/research/index.html.

[72] R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. *Math. Program.*, 91:239–269, 2002.

[73] R. Garmanjani, D. Júdice, and L. N. Vicente. Trust-region methods without using derivatives: Worst case complexity and the non-smooth case. Technical Report 15-03, Dept. Mathematics, Univ. Coimbra, 2015.

[74] R. Garmanjani and L. N. Vicente. Smoothing and worst-case complexity for direct-search methods in nonsmooth optimization. *IMA Journal of Numerical Analysis*, 33:1008–1028, 2013.

[75] N. I. M. Gould and Ph. L. Toint. Nonlinear programming without a penalty function or a filter. *Math. Program.*, 122:155–196, 2010.

[76] S. Gratton, Ph. L. Toint, and A. Troeltzch. An active-set trust-region method for derivative-free nonlinear bound-constrained optimization. *Optim. Methods Softw.*, 21:873–894, 2011.

[77] S. Gratton, C. W. Royer, L. N. Vicente, and Z. Zhang. Direct search based on probabilistic descent. *SIAM J. Optim.*, 2015, to appear.

[78] S. Gratton, A. Sartenaer, and Ph. L. Toint. Recursive trust-region methods for multiscale nonlinear optimization. *SIAM J. Optim.*, 19:414–444, 2008.

[79] S. Gratton and L. N. Vicente. A merit function approach for direct search. *SIAM J. Optim.*, 24:1980–1998, 2014.

[80] S. Gratton and L. N. Vicente. A surrogate management framework using rigorous trust-region steps. *Optim. Methods Softw.*, 29:10–23, 2014.

[81] S. Gratton, L. N. Vicente, and Z. Zhang. A subspace decomposition framework for nonlinear optimization. Technical report, in preparation.

[82] J. D. Griffin, T. G. Kolda, and R. M. Lewis. Asynchronous parallel generating set search for linearly-constrained optimization. *SIAM J. Sci. Comput.*, 30:1892–1924, 2008.

[83] E. A. E. Gumma, M. H. A. Hashim, and M. Montaz Ali. A derivative-free algorithm for linearly constrained optimization problems. *Comput. Optim. Appl.*, 57:599–621, 2014.

[84] N. Hansen, A. Ostermeier, and A. Gawelczyk. On the adaptation of arbitrary normal mutation distributions in evolution strategies: The generating set adaptation. In L. Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms, Pittsburgh*, pages 57–64, 1995.

[85] W. Hare and J. Nutini. A derivative-free approximate gradient sampling algorithm for finite minimax problems. *Comput. Optim. Appl.*, 56:1–38, 2013.

[86] P. Hough, T. G. Kolda, and V. Torczon. Asynchronous parallel pattern search for nonlinear optimization. *SIAM J. Sci. Comput.*, 23:134–156, 2001.

[87] W. Huyer and A. Neumaier. Global optimization by multilevel coordinate search. *J. Global Optim.*, 14:331–355, 1999.

[88] D. Jones, C. Perttunen, and B. Stuckman. Lipschitzian optimization without the Lipschitz constant. *J. Optim. Theory Appl.*, 79:157–181, 1993.

[89] A. Kannan and S. M. Wild. Obtaining quadratic models of noisy functions. Technical Report ANL/MCS-P1975-1111, Argonne National Laboratory, 2011.

[90] C. T. Kelley. *Iterative Methods for Optimization.* SIAM, Philadelphia, 1999.

[91] C. T. Kelley. *Implicit Filtering.* Software Environments and Tools. SIAM, Philadelphia, 2011.

[92] K. C. Kiwiel. A nonderivative version of the gradient sampling algorithm for nonsmooth nonconvex optimization. *SIAM J. Optim.*, 20:1983–1994, 2010.

[93] T. G. Kolda, R. M. Lewis, and V. Torczon. Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Rev.*, 45:385–482, 2003.

[94] J. Konečný and P. Richtárik. Simple complexity analysis of simplified direct search. Technical Report arXiv:1410.0390v2, School of Mathematics, University of Edinburgh, November, 2014.

[95] J. C. Lagarias, B. Poonen, and M. H. Wright. Convergence of the restricted Nelder-Mead algorithm in two dimensions. *SIAM J. Optim.*, 22:501–532, 2012.

[96] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright. Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM J. Optim.*, 9:112–147, 1998.

[97] S. Le Digabel. Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm. *ACM Trans. Math. Software*, 37:44:1–44:15, 2011.

[98] R. M. Lewis and V. Torczon. Pattern search algorithms for bound constrained minimization. *SIAM J. Optim.*, 9:1082–1099, 1999.

[99] R. M. Lewis and V. Torczon. A globally convergent augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds. *SIAM J. Optim.*, 12:1075–1089, 2002.

[100] R. M. Lewis and Virginia Torczon. Pattern search methods for linearly constrained minimization. *SIAM J. Optim.*, 10:917–941, 2000.

[101] G. Liuzzi and S. Lucidi. A derivative-free algorithm for inequality constrained nonlinear programming via smoothing of an ℓ_∞ penalty function. *SIAM J. Optim.*, 20:1–29, 2009.

[102] G. Liuzzi, S. Lucidi, and F. Rinaldi. Derivative-free methods for bound constrained mixed-integer optimization. *Comput. Optim. Appl.*, 53:505–526, 2012.

[103] G. Liuzzi, S. Lucidi, and M. Sciandrone. Sequential penalty derivative-free methods for nonlinear constrained optimization. *SIAM J. Optim.*, 20:2614–2635, 2010.

[104] M. Locatelli and F. Schoen. *Global Optimization: Theory, Algorithms, and Applications*. MOS-SIAM Series on Optimization. SIAM, Philadelphia, 2013.

[105] S. Lucidi and M. Sciandrone. On the global convergence of derivative-free methods for unconstrained optimization. *SIAM J. Optim.*, 13:97–116, 2002.

[106] S. Lucidi, M. Sciandrone, and P. Tseng. Objective-derivative-free methods for constrained optimization. *Math. Program.*, 92:37–59, 2002.

[107] J. M. Martínez and E. A. Pilotta. Inexact restoration algorithms for constrained optimization. *J. Optim. Theory Appl.*, 104:135–163, 2000.

[108] J. M. Martínez and F. N. C. Sobral. Constrained derivative-free optimization on thin domains. *J. Global Optim.*, 56:1217–1232, 2013.

[109] Ph. R. Sampaio and Ph. L. Toint. A derivative-free trust-funnel method for equality-constrained nonlinear optimization. *Comput. Optim. Appl.*, 61:25–49, 2015.

[110] K. I. M. McKinnon. Convergence of the Nelder-Mead simplex method to a nonstationary point. *SIAM J. Optim.*, 9:148–158, 1998.

[111] J. A. Nelder and R. Mead. A simplex method for function minimization. *Comput. J.*, 7:308–313, 1965.

[112] Y. Nesterov. *Introductory Lectures on Convex Optimization*. Kluwer Academic Publishers, Dordrecht, 2004.

[113] Y. Nesterov. Random gradient-free minimization of convex functions. Technical Report 2011/1, CORE, 2011.

[114] M. J. D. Powell. A new algorithm for unconstrained optimization. In J. B. Rosen, O. L. Mangasarian, and K. Ritter, editors, *Nonlinear Programming*. Academic Press, New York, 1970.

[115] M. J. D. Powell. The theory of radial basis function approximation in 1990. In W. A. Light, editor, *Advances in Numerical Analysis, Vol. II: Wavelets, Subdivision Algorithms and Radial Basis Functions*, pages 105–210. Oxford University Press, Cambridge, 1992.

[116] M. J. D. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In S. Gomez and J.-P. Hennart, editors, *Advances in Optimization and Numerical Analysis, Proceedings of the Sixth Workshop on Optimization and Numerical Analysis, Oaxaca, Mexico*, volume 275 of *Math. Appl.*, pages 51–67. Kluwer Academic Publishers, Dordrecht, 1994.

[117] M. J. D. Powell. Least Frobenius norm updating of quadratic models that satisfy interpolation conditions. *Math. Program.*, 100:183–215, 2004.

[118] M. J. D. Powell. Developments of NEWUOA for minimization without derivatives. *IMA J. Numer. Anal.*, 28:649–664, 2008. `http://en.wikipedia.org/wiki/NEWUOA`.

[119] M. J. D. Powell. The BOBYQA algorithm for bound constrained optimization without derivatives. Technical Report DAMTP 2009/NA06, University of Cambridge, 2009. `http://en.wikipedia.org/wiki/BOBYQA`.

[120] M. J. D. Powell. On fast trust region methods for quadratic models with linear constraints. Technical Report DAMTP 2014/NA02, University of Cambridge, 2014. `http://en.wikipedia.org/wiki/LINCOA`.

[121] R. T. Rockafellar. Generalized directional derivatives and subgradients of nonconvex functions. *Can. J. Math.*, 32:257–280, 1980.

[122] R. T. Rockafellar and R. J-B. Wets. *Variational Analysis*. Springer, Berlin, 1998.

[123] K. Scheinberg and Ph. L. Toint. Self-correcting geometry in model-based algorithms for derivative-free unconstrained optimization. *SIAM J. Optim.*, 20:3512–3532, 2010.

[124] T. A. Sriver, J. W. Chrissis, and Mark A. Abramson. Pattern search ranking and selection algorithms for mixed variable simulation-based optimization. *European J. Oper. Res.*, 198:878–890, 2009.

[125] V. Torczon. On the convergence of pattern search algorithms. *SIAM J. Optim.*, 7:1–25, 1997.

[126] M. W. Trosset. On the use of direct search methods for stochastic optimization. Technical Report TR00-20, CAAM Technical Report, Rice University, 2000.

[127] P. Tseng. Fortified-descent simplicial search method: A general approach. *SIAM J. Optim.*, 10:269–288, 1999.

[128] A. Ismael F. Vaz and L. N. Vicente. A particle swarm pattern search method for bound constrained global optimization. *J. Global Optim.*, 39:197–219, 2007.

[129] L. N. Vicente. Worst case complexity of direct search. *EURO Journal on Computational Optimization*, 1:143–153, 2013.

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

[131] S. M. Wild, R. G. Regis, and C. A. Shoemaker. ORBIT: Optimization by radial basis function interpolation in trust-regions. *SIAM J. Sci. Comput.*, 30:3197–3219, 2008.

[132] S. M. Wild and C. Shoemaker. Global convergence of radial basis function trust region derivative-free algorithms. *SIAM J. Optim.*, 21:761–781, 2011.

[133] D. Winfield. *Function and Functional Optimization by Interpolation in Data Tables*. PhD thesis, Harvard University, USA, 1969.