# TOWARDS A NEW DIRECT ALGORITHM. A TWO-POINTS BASED SAMPLING METHOD

CHITER LAKHDAR

ABSTRACT. The DIRECT algorithm was motivated by a modification to Lipschitzian optimization. The algorithm begins its search by sampling the objective function at the midpoint of an interval, where this function attains its lowest value, and then divides this interval by trisecting it. One of its weakness is that if a global minimum lies at the boundaries, which can never be reached, the convergence of the algorithm will be unecessary slow. We present a one-dimensional variante of the DIRECT algorithm based on another strategy of sudividing the search domain. It consists of decreasing the number of intervals and increasing the number of sampling points, by interverting the roles of dividing and sampling at some steps of the DIRECT algorithm, and thus, overcoming this disadvantage.

## 1. INTRODUCTION.

The original DIRECT "DIviding RECTangles" algorithm, developed by Donald. R. Jones et al. [10], was essentialy designed for finding the global minimum of a multivariate function. The algorithm was motivated to overcome some of the problems that standard Lipschitzian optimization encounters. The one-dimensional version is a modification of the Piyavskii-Shubert' algorithm, (see [11] and [12]). The algorithm adopts a center-sampling strategy. The objective function is evaluated at the midpoint of the domain, where a lower bound is constructed. The domain is then trisected and two new center points are sampled, so the center points will not be sampled again. At each iteration (dividing and sampling), DIRECT identifies intervals that contain the best function value found so far. A disadvantage of this algorithm is that the boundaries cannot be reached, consequently, the convergence to the global optimum will be slow if a minimizer lies at the boundary. In this paper we are interested with the one-dimensional version. We introduce a different way for sampling and dividing the search domain by inverting the roles of sampling and dividing. In fact, DIRECT evaluates the function at the center $c = \frac{a+b}{2}$ of an interval $[a, b]$, and then divides it into three subintervals, each one with a center. In our sampling method, the division (trisection) takes place at the evaluation of the objective function at two consecutive points : one-third and two-third of the interval, and the midpoint evaluation as a bisection. By this way, we save the property that the sampled points (1/3 and 2/3) are respectively at 2/3 and 1/3 of the left and right intervals. So we need two new points at each iteration. As a consequence, the points sampled by this method are closer to the boundaries. This fact predicts a fast convergence. Figure 1.1 is a comparison of the two sampling

*Date*: March 1st, 2005.

*Key words and phrases.* Global optimization, *DIRECT algorithm, weakly optimal interval, sampling method.*

methods. The paper is organized as follows. Section 2 is a short introduction to the DIRECT algorithm. Section 3 is the main part of this paper. We present the method in a general setting with giving some arguments. Note that the idea of potentially optimal used here is a weaker notion of optimality. This paper is of a theoretical interest. It provides a useful way for a generalization to the bidimensional and further to the multidimensional case which is of a more interest to show the practicability of this method

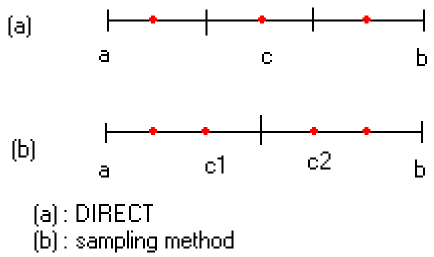


Fig. 1. A comparison with DIRECT of our subdivision and sampling method.

## 2. DIRECT algorithm overview.

In this section, we give a short description of the DIRECT algorithm. Here, only the one-dimensional DIRECT algorithm, which is of a particular interest, is described. For a more detailed description, the reader is reffered to [10]. In Lipschitzian optimization, it occurs frequently that the Lipschitz constant can not be determined or reasonable estimated. Many problems, in practice, may not even be Lipschitz continuous. If this Lipschitz constant can be estimated, and if this estimate is too low, the result may not be a minimum of the objective function, and if the choice is too large, the convergence of the an algorithm such as Shubert's ([11] and [12]) will be slow. The DIRECT algorithm was motivated by these two shortcomings of Lipschitzian Optimization. The 1-D DIRECT samples at the midpoints of the search intervals, by selecting potentially optimal intervals (definition 2.1) to be further explored. This strategy of selecting and dividing gives DIRECT its performance and fast convergence compared to other deterministic methods. As a young method, DIRECT is being enhanced with new ideas. Gablonsky et al.[5], [6], [7] and [8] studied the behaviour of DIRECT. But only few works were known for the convergence of the algorithm, [4]. The algorithm converges to the global optimal function value, if the objective function is continuous or at least continuous in the neighborhood of a global optimum. This could be assured since, as the number of iterations goes to infinity. An example of DIRECT applied to industrial problems can be found in [1], and [2]. DIRECT can be found at Matlab:

http://www4.ncsu.edu/~definkel/research/index.html, and Fortran:

http://www4.ncsu.edu/eos/users/c/ctkelley/www/iffco.html. DIRECT deal with the following unbound-constrained optimization problem.

$$\min_{x\in\Omega} f(x), \text{ where } f : IR^N \to IR \tag{2.1}$$

where

$$\Omega = \left\{ x \in IR^N : l \le x \le u \right\}.$$

$f$ is a Lipschitz continuous function on $\Omega$. The one-dimensional DIRECT algorithm is given in [5] and [10]. It can be described by the following steps. The first step in the algorithm is the initialization, it consists by evaluating $f$ at the midpoint of the initial interval. This value is taken as $f_{\min}$. This interval is then divided into three subintervals. The point sampled before will be a midpoint of the center subinterval. So, two new points are added at every step. In each iteration, where the pair divide-sample is refered as an iteration, new intervals are created by dividing old ones, and then the function is sampled at new centers of the new intervals. During an iteration, DIRECT will identifie potentially optimal intervals, see definition and figure below. DIRECT will sample $f$ at the centers of the newly created intervals. The algorithm will stop when a user-supplied budget of function evaluations is exhausted.

**Definition 2.1.** *Let $f_{\min}$ be the current best function value. An interval $i$ is said to be potentially optimal if there exists some rate of change $\widetilde{K} > 0$ such that*

$$f(c_i) - \widetilde{K}(b_i - a_i)/2 \leq f(c_j) - \widetilde{K}(b_j - a_j)/2, \forall j \tag{2.2}$$

$$f(c_i) - \widetilde{K}(b_i - a_i)/2 \leq f_{\min} - \epsilon\, |f_{\min}| \tag{2.3}$$

Condition (2.2) in definition expresses that the selected interval is in the lower right of the convex hull of the dots, while condition (2.3) forces the lower bound for the interval, to exceed the current best solution by a nontrivial amount. $\epsilon \geq 0$, is a balance parameter between local and global search. In some calculations $\epsilon$ is proposed between $10^{-3}$ and $10^{-7}$. See [3], [5], and [10]. Figure 2 is an illustration of definition (2.1). Potentially optimal intervals are on the lower-right convex hull. Each dot in the graph represent a set of intervals having the same lenght and equal function values at the centers. Note the effect of the balance parameter $\epsilon$. In this case, interval with the smallest function value at the center is not potentially optimal, that is, largest unexplored intervals are always chosen for sampling.

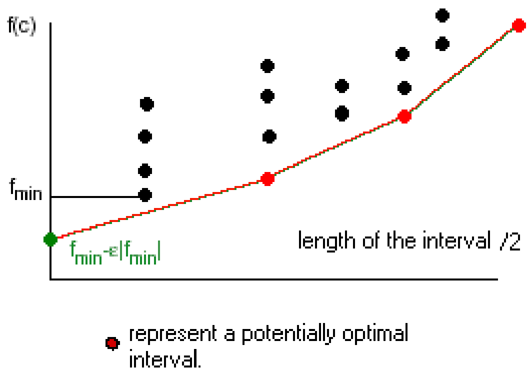


Fig. 2. Illustration of potentially optimal intervals

## 3. A two-points sampling method.

Let $f$ be a Lipschitz continuous function on $[a, b]$ with Lipschitz constant $K$, i.e.

$$|f(x) - f(y)| \leq K\,|x - y|\,\forall x, y \in [a, b] \tag{3.1}$$

We know that the concave function defined for any $y \in [a, b]$ by $f(y) - K\,|x - y|$, $x \in [a, b]$, is an underestimator of $f$. Let

$$c_1 = \frac{2a + b}{3}, and\ c_2 = \frac{a + 2b}{3}$$

be two consecutive points in $[a, b]$. By setting $y = c_1$, and $y = c_2$ in (3.1), then we get the following inequalities

$$f(x) \geq f(c_1) - K\,|x - c_1| \tag{3.2}$$

and

$$f(x) \geq f(c_2) - K\,|x - c_2| \tag{3.3}$$

The function $F(x) = \max\limits_{k=1,2} f(c_k) - L\,|x - c_k|\,,\ x \in [a, b]$ , formed by the intersection of (3.2) and (3.3) underestimates $f$, its minimum is a lower bound on the least value of $f$. The restriction of $F$ to the interval $[c_1,\ c_2]$, (called tooth) attains its minimal value (downward peak) given by

$$L^* = \frac{f(c_1) + f(c_2)}{2} - K\frac{c_2 - c_1}{2}$$

and it occurs at the point

$$x^* = \frac{c_1 + c_2}{2} + \frac{f(c_1) - f(c_2)}{2K}$$

There is two possibilities to define a potentially optimal intervals. We can restrict our lower bound on the center subinterval $[c_1, c_2]$. Since the peak where $f$ is minimal is not the lowest one (see [9]), we can not see wich interval can be selected to be further explored. As an important observation, we see that the downward peak is pointed in the direction of the minimum value at these two points, and the value of $x^*$ is close to the side where $f$ is minimal . Hence, for each tooth the peak point will be the center of the subinterval located in the projection of the top of the triangle onto the interval, notice that this triangle have two equal sides, cf. Figure 3 (a). An idea to see weather an interval is potentially optimal or not, is to choose a triangle in each iteration. The value of the domain point corresponding to the lowest peak is computed. The triangle with the lowest function value at this point is optimal. Note that if this function value dos not exist, the lowest value will be $\min(f(c_1), f(c_2))$. See the comments in the remark below.

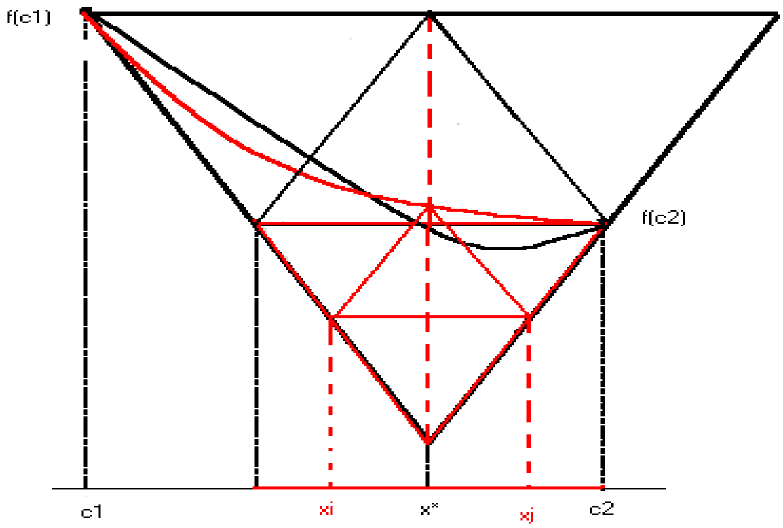


Fig. 3 (a). Function value of the point corresponding to the peak.

**Remark 3.1.** *One possibility to specify which interval would be selected to be further sampled can be done by the following idea: Suppose that we are only concerned with the center interval* $[c_1, c_2]$ *as shown in figure 3 (a). A distinguish point is the peak point* $x^*$ *which is a center of a specific subinterval* $[x_i, x_j]$ *found by the projection of an horizental line as illustrated in figure 3 (a). Note that the vale of* $f(x^*)$ *lies in a small triangle or in a big one. Otherwise, this subset may be left out from further consideration. A lower bound for* $f(x^*)$ *can be found by the intersection of the two lines*

$$y_l^U = f(x^*) + K(x - x^*)$$

*and*

$$y_r^U = f(x^*) - K(x - x^*)$$

*respectively with the lines*

$$y_l^L = f(c_1) + K(c_1 - x)$$

*and*

$$y_r^L = f(c_2) + K(x - c_2)$$

*which gives respectively the points*

$$x_i = \frac{f(c_1) - f(x^*)}{2K} + \frac{(x^* + c_1)}{2}$$

*and*

$$x_j = \frac{f(x^*) - f(c_2)}{2K} + \frac{(x^* + c_2)}{2}$$

*A definition of potentially optimal intervals can be done by selecting subintervals with the lowest value of* $f(x^*)$*, plus an analogous condition to (2.2) using the weighting parameter* $\epsilon$*. It is natural to define a variante of a potentially optimal interval using the above observations. We leave the details to the interested reader.*

Notice that $L^*$ and $x^*$ can be expressed by means of $a$ and $b$. The second version of optimality is the one used in the original DIRECT. The lowest value the function $f(c_1) - K|x - c_1|$ defined on the interval $[a, x^*]$ can attain is at the endpoint $a$ . This minimum value is $f(c_1) + K(a - c_1)$, which can also be expressed by

$$f(c_1) - K(\frac{b-a}{3}) \tag{3.4}$$

Notice that this value is the same at $c_2$ since $c_1$ is the midpoint of the interval $[a, c_2]$. The same observation can be done with $f(c_2) - K|x - c_2|$. The minimum value occurs at the endpoint $b$. This lower bound is then

$$f(c_2) - K(\frac{b-a}{3}) \tag{3.5}$$

The minimum value for (3.4) and (3.5) is a minimum for $F$. These lower bounds can be considered as stronger bounds. Figure 3 (b) shows this. So, strong bounds leads to weak optimality and weak bounds gives strong optimality. Now, we divide the ineterval $[a, b]$ into two subintervals by bisecting it. The point one-third becomes two-third for the left interval, and the one with two-third becomes one-third for the right interval. So we need two new points for each interval. What can we preserve by the use of this strategy ? as it is shown in figure 1, after an iteration some points sampled by the original DIRECT are also sampled in this method. Note that the boudaries in (a) are sample points in (b).

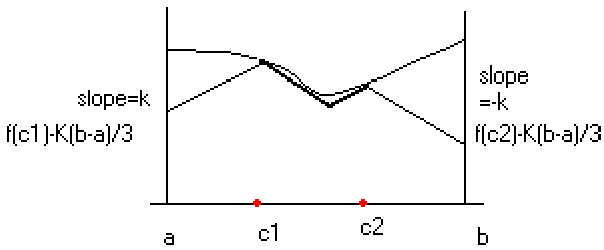


Fig. 3 (b). Lower bound with two points sampling

Next, we define which interval to choose for further subdivision and sampling. Suppose that we have partioned the interval into $[a_i, b_i]$ intervals, each one with two consecutive points $c_i$ and $c_{i+1}$. The interval with the lowest value at $c_i$ or $c_{i+1}$ will be chosen. A similar definition to (2.1) is given by:

**Definition 3.1.** *Let $f_{\min}$ be the best value of the objective function found so far. An interval $i$ is said to be optimal potentially optimal if there exists some $\widetilde{K} > 0$ such that*

$$\min\left\{f(c_i) - \widetilde{K}(b_i - a_i)/3, f(c_{i+1}) - \widetilde{K}(b_i - a_i)/3\right\}$$

$$\leq \min\left\{f(c_j) - \widetilde{K}(b_j - a_j)/3, f(c_{j+1}) - \widetilde{K}(b_j - a_j)/3\right\}, \forall j, \text{ and} \tag{3.6}$$

$$\min\left\{f(c_i) - \widetilde{K}(b_i - a_i)/3, f(c_{i+1}) - \widetilde{K}(b_i - a_i)/3\right\} \leq f_{\min} - \epsilon|f_{\min}| \tag{3.7}$$

**Remark 3.2.** *Note that the optimality in the above definition differs from the one in definition 2.1. In fact, another way to reformulate this definition is to show that an interval [a, b] is weakly optimal if there exists a strongly optimal subinterval. Figure 4 gives an illustration of this fact. An analogue definition of potentially strong optimal intervals is the following*

$$max\{f(c_i) - \widetilde{K}(b_i - a_i)/3, f(c_{i+1}) - \widetilde{K}(b_i - a_i)/3\}$$

$$\leq \min\left\{f(c_j) - \widetilde{K}(b_j - a_j)/3, f(c_{j+1}) - \widetilde{K}(b_j - a_j)/3\right\}$$

*Note also that the second condition is the same as in definition 2.1.*

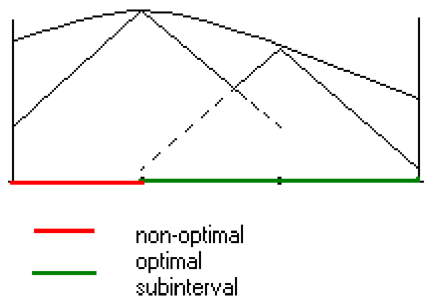


Fig. 4. Potentially optimal subinterval

We now formally state the one-dimensional algorithm for the second version.

3.1. **Univariate algorithm.** Step 1. Set $m = 1$ (evaluation number), $[a_1, b_1] = [l, u]$ $c_1 = (2a_1 + b_1)/3, c_1' = (a_1 + 2b_1)/3$, and evaluate $f(c_1)$, $f(c_1')$. Set $f_{\min} = \min(f(c_1), f(c_1'))$. Let $t = 0$ (iteration counter).

Step 2. Identify the set $S$ of all potentially optimal intervals.

Step 3. Select any interval $i \in S$. with two points $c_i$ and $c_i'$.

Step 4. Let $\delta = (b_i - a_i)/2$, and set $c_{m+1} = c_i' - \delta = c_{m+2} - \delta$, and $c'_{m+2} = c_i + \delta = c'_{m+1} + \delta$. evaluate $f(c_{m+1})$ and $f(c'_{m+2})$,i.e., evaluate $\min(f(c_{m+1}), f(c'_{m+1}))$, and $\min(f(c_{m+2}), f(c'_{m+2}))$. Then update $f_{\min}$.

Step 5. In the partition, we have the left subinterval $[a_{m+1}, b_{m+1}] = [a_i, a_i + \delta]$, with two successive points $c_{m+1}$ and $c'_{m+1}$ and the right subinterval $[a_{m+2}, b_{m+2}] = [b_{m+1}, b_{m+2}] = [a_i + \delta, b_i]$, with two successive points $c_{m+2}$ and $c'_{m+2}$. Then modify interval $i$ to be the subinterval with the lowest function value $f(c_i)$ or $f(c_i')$. i.e., the left or the right subinterval.

Set $m = m + 2$.

Step 6. Set $S = S - \{i\}$ . If $S \neq \emptyset$, go to step 3.

Step 7. Set $t = t + 1$ . If $t <$ iteration number, then stop. Otherwise, go to step 2.

**Remark 3.3.** *Step 4 can be done in the following way: if we set* $\delta = (b_i - a_i)/6$,*and set* $c_{m+1} = c_i - \delta = c'_{m+1} - \delta$, *and* $c'_{m+2} = c_{m+2} + \delta = c_i' + \delta$. *evaluate* $f(c_{m+1})$ *and* $f(c'_{m+2})$,*i.e., evaluate* $\min(f(c_{m+1}), f(c'_{m+1}))$, *and* $\min(f(c_{m+2}), f(c'_{m+2}))$. *Then* *update* $f_{\min}$. *Updating* $f_{\min}$ *for each interval is done by* $f_{\min} = \min(f(c_{m+1}), f(c'_{m+1}))$ *and* $\min(f(c_{m+1}), f(c'_{m+1}))$.

## 4. CONCLUSION

The DIRECT algorithm is a deterministic sampling method designed for bound constrained Lipschitz problems in a small dimensions. Different ideas have been proposed to overcome some of its weakness. Our research was motivated by the above comments in the introduction and the fact that DIRECT preserves centers of potentially optimal intervals in the sens that these points become centers of new interval, so they will not be sampled again. By analogy, in the proposed method, the point one-third become two-third, and vice-versa for each selected interval. Some points sampled by the proposed method are also centers of intervals for the original DIRECT. Boudaries in (a) are considered as sampled points in (b) as shown in figure 1. The proposed method predicts a fast convergence as seen in [10], formula 9. The convergence will be fast if we set $2^{-k}$, in place of $3^{-k}$, i.e.,

$$d = \frac{\left[j 2^{-2(k+1)} + (n-j) 2^{-2k}\right]^{1/2}}{3}$$

Here, $d$ is the distance from the points (one-third, and two-third) to the boundaries. $d$ approaches zero when the number of subdivisions approaches infinity. There is some optimism regarding to perspective of our approach, the multidimensional generalization seems promising. Two-dimensional version can be proposed further and tests will be carried out in the framework of this model. Also, this paper is in its first version, it may be subject to some modifications.

## References


[1] M. Bjorkman and K. Holmstrom. *Global Optimization Using the DIRECT Algorithm in Matlab1.* Center for Mathematical Modeling Department of Mathematics and Physics. Malardalen University, Sweden. AMO - Advanced Modeling and Optimization Vol. 1, Num. 2, 1999.

[2] R.G. Carter, J.M. Gablonsky, A. Patrick, C.T. Kelley, and O.J. Eslinger. *Algorithms for noisy problems in gas transmission pipeline optimization.* Optimization and Engineering, 2:139-157, 2002.

[3] D. E. Finkel, *DIRECT Optimization Algorithm User Guide.* Center for Research in Scientific Computation. North Carolina State University, Raleigh, march 2003.

[4] D. E. Finkel, and C. T. Kelley, *Convergence Analysis of the DIRECT algorithm.* North Carolina State University, Center for Research in Scientific Computation and Department of Mathematics, Raleigh, version of july 14, 2004.

[5] J.M. Gablonsky, *An implementation of the DIRECT algorithm.* Technical Report, Center for Research in Scientific Computation, North Carolina State University, august 1998.

[6] J.M. Gablonsky, *DIRECT version 2.0 userguide.* Technical Report, CRSC-TR01-08, Center for Research in Scientific Computation, North Carolina State University, april 2001.

[7] J.M. Gablonsky, *Modifications of the DIRECT Algorithm.* PhD Thesis. North Carolina State University, Raleigh, North Carolina, 2001.

[8] J.M. Gablonsky and C.T. Kelley. *A locally-biased form of the DIRECT algorithm.* Journal of Global Optimization, 21:27-37, 2001.

[9] P. Hansen, B. Jaumard, and Shi-Hui Lu. *On Timonov's Algorithm for Global Optimization of Univariate Lipschitz Functions.* Journal of Global Optimization, 1:37-46, 1991.

[10] D.R. Jones, C.D. Perttunen, and B.E. Stuckman. *Lipschitzian optimization without the lipschitz constant.* Journal of Optimization Theory and Application, 79:157, October 1993.

[11] A. Piyavskii. *An algorithm for finding the absolute extremum for a function.* USSR Comput. Mathem. and Mathem. Phys., 12, No.4, 888-896. 1972.

[12] B. Shubert. *A sequential method seeking the global maximum of a function.* SIAM J. Numer. Anal., 9:379-388, 1972.

Laboratoire de Mathématiques Fondamentale et Numérique, Faculté des Sciences. Université Ferhat-abbas. 19000 Sétif. Algerie.
*E-mail address*: chiterl@univ-setif.dz