COMPUTATIONAL SCIENCE AND TECHNIQUES
Volume 1, Number 2, 2013, 184-192

Publisher: Klaipėda University
http://journals.ku.lt/index.php/CST
Online ISSN: 2029-9966

ON THE REDUCED-SET PARETO-LIPSCHITZIAN OPTIMIZATION

Jonas Mockus [1], Remigijus Paulavičius [2]
[1]Vilnius University, Institute of Mathematics and Informatics, System Analysis Department, Akademijos g. 4, LT-08663 Vilnius, jmockus@gmail.com
[2]Lithuanian University of Educational Sciences, Studentų g. 39, LT-08106 Vilnius, remigijus.paulavicius@mii.vu.lt

Abstract. A well-known example of global optimization that provides solutions within fixed error limits is optimization of functions with a known Lipschitz constant. In many real-life problems this constant is unknown. To address that a method called Pareto-Lipschitzian Optimization (PLO) was described that provides solutions within fixed error limits for functions with unknown Lipschitz constants. In this approach, a set of all unknown Lipschitz constants is regarded as multiple criteria using the concept of Pareto Optimality (PO).

In this paper, a new version of the Pareto-Lipschitzian Optimization method (PLOR) is proposed where a set of unknown Lipschitzian constants is reduced just to the minimal and maximal ones. In the both methods, partition patterns are similar to those of DIRECT. The difference is in the rules of sequential partitions defining non-dominated sets. In PLO, it includes all Pareto-Optimal sets defined by all Lipschitz constants. In PLOR, it considers just two elements corresponding to the maximal and minimal Lipschitz constant. in DIRECT, it selects a part of the Pareto-Optimal set which is determined by some heuristic parameter ε.

Keywords: Pareto optimality, Lipschitz functions, Reduced set, Global optimization

Introduction

Since, PLOR is reduced version of PLO the description of PLOR approximately follows the lines of the PLO description in (Mockus., 2011) with corresponding changes. In the following sections we focus mainly on comparison of PLOR and DIRECT, since of PLO and DIRECT was compared in (Mockus., 2011).

1. Worst Case Analysis

In terms of the decision theory the term "Worst Case Analysis" means that the method must retain the exactness or ε-exactness in all cases, including the worst one. To obtain the exact solution in the worst case, one may need many iterations, if the family of problems is large. An important advantage is a well-defined maximal deviation.

The well-known examples of Worst Case Analysis are optimization methods for the set of Lipschitz functions with known Lipschitz constants, see, e.g. (Evtushenko, 1985), (Figueira, Greco, & Ehrgott, 2004), (Paulavičius & Žilinskas, 2007), (Paulavičius, Žilinskas, & Grothey, 2010), (Pijavskij, 1972), (Shubert, 1972). Here the deviation can be defined in terms of the objective function:

$$\delta_f = |f(x) - f(x^*)| \leq \omega \parallel x - x^* \parallel, \tag{1}$$

where ω is the Lipschitz constant.

For a wider family of functions, such as Lipschitz functions with an unknown constant, only the deviation in terms of function arguments can be ensured

$$\delta_x = \parallel x - x^* \parallel. \tag{2}$$

In (Sukharev, 1971), the problem of global optimization for a family of Lipschitz functions with unknown Lipschitz constants is considered. In this case, the uniform grid on a compact feasible set is the optimal passive method, in the mini-max sense. The term "passive" means that all the points of observations (x_i) are determined at the start. The term "observation" denotes an evaluation of the objective function $f(x)$ at some fixed point x, and the term "mini-max" means minimization of the maximal deviation. Here the number of required observations will be exponentially increasing with the complexity of the problem. We define the complexity as the number of variables and the accuracy of solutions (Ko, 1991).

The contribution of this paper is a definition of the problem of optimization with reduced set of Lipschitz constants in terms of Pareto optimality and the Reduced Pareto-Lipschitzian Optimization (PLOR) algorithm to realize this approach. Furthermore, advantages and disadvantages of PLOR are compared to DIRECT algorithm (Jones, Perttunen, & Stuckman, 1993), which is a well-known active method for optimization of Lipschitz functions with unknown constants. To increase the efficiency of search, the DIRECT algorithm uses heuristic rules that depend on some manually chosen parameter ε. The DIRECT algorithm considers a subset of potentially optimal hyper-rectangles satisfying Definition 1.

Definition 1. *Let S be the set of all hyperrectangles created by DIRECT after k iterations. Let c_i denote the center point of the ith hyperrectangle, and let d_i denote the distance from center point to the vertices. Let $\varepsilon > 0$ be a positive constant and f_{min} be the currently known best function value. A hyperrectangle $S_j \in S$ is said to be potentially optimal if there exists some rate-of-change constant $\tilde{K} > 0$ such that*

$$f(c_j) - \tilde{K}d_j \leq f(c_i) - \tilde{K}d_i, \quad \forall i \in S \tag{3}$$

$$f(c_j) - \tilde{K}d_j \leq f_{min} - \varepsilon|f_{min}|. \tag{4}$$

Different versions of DIRECT (Finkel & Kelley, 2006; Jones, Perttunen, & Stuckman, 1993), (Gablonsky & Kelley, 2001), (Jones, Perttunen, & Stuckman, 1993), (Sergeyev & Kvasov, 2006) regard different subsets of non-dominated decisions determined by the corresponding heuristics. Non-dominated decisions means that there exists no decision which is as good by all criteria and better by at least one criteria. Different criteria are defined by different Lipschitz functions. The non-dominated decisions are part of all Pareto ones. The main theoretical difference of DIRECT and Pareto-Lipschitzian (PL) algorithms is that PL considers all the Pareto Optimal decisions while DIRECT regards only a part of them with further filtering by some heuristic rule.

The difference between different versions of Pareto-Lipschitzian algorithm is as follows. PLO explores all PO decisions, and PLOR considers just two PO decisions defined by the minimal and maximal Lipschitz constants correspondingly. In both PLO and PLOR no heuristic parameters are applied. Exploration of all PO decisions is particularly suitable for parallel computations if the computing time of observations is sufficiently large. Otherwise, PLOR is preferable as a sequential search procedure.

2. Pareto-Optimal Approach: Dominant Analysis

The concept of Pareto optimality (see, e.g., (Figueira, Greco, & Ehrgott, 2004) (Miettinen, 1999) (Pardalos & Siskos, 1995)) is traditionally used in the cases where an objective is a vector-function $f_\omega(x),\ \omega \in \boxtimes$. Here $x \in D \subset R^K$ is the control parameter, ω is a component index of the vector-objective $f_\omega(x)$, and $\boxtimes$ is a set of all components ω.

Definition 2. *The decision* $x \in D$ *dominates the decision* $x^* \in D$*, if*

$$\begin{aligned} f_\omega(x) &\le f_\omega(x^*), \quad for\ all \quad \omega \in \boxtimes \\ f_\omega(x) &< f_\omega(x^*), \quad for\ at\ least\ one \quad \omega \in \boxtimes. \end{aligned} \qquad (5)$$

Here D is the decision space and $\boxtimes$ is the set of components ω.

Definition 3. *The decision* $x^* \in D$ *is called Pareto Optimal (PO)* [1]*, if there is no dominant decision* $x \in D$.

3. Reduced-Set Pareto-Lipschitzian Optimization (PLOR)

We explain the PO optimization of Lipschitz functions with just two (the minimal ant the maximal) constants by considering the following one-dimensional example.

Suppose that the interval $D = [a, b] \subset R$ is partitioned into intervals $[a_i, b_i],\ i = 1, \dots, I$ of lengths $l_i = b_i - a_i$ with midpoints $c_i = (b_i + a_i)/2$ and the values $f(c_i)$ of the function $f_\omega(x)$ are known only at the midpoints c_i. The unknown Lipschitz constants are regarded as different components of multiple criteria. The variables x are represented by the intervals $a_i \le\ x \le\ b_i$ and the function $f_\omega(x)$ is approximated by the lower bounds:

$$\mathrm{L_i(\omega) = f(c_i) - \omega\ l_i/2 \le f_\omega(x),\ a_i \le\ x \le b_i.} \qquad (6)$$

[1] Here we consider minimization, while in maximization the inequalities should be reversed.

Expression (6) shows that the lower bound of the interval i is increasing with $f(c_i)$ and decreasing with l_i for all ω. We compare the "quality" of different intervals by their lower bounds. For example, we say that the interval is better for a given ω, if its lower bound is lower.

Definition 4. *The interval* $i\colon a_i \le x \le b_i$ *that belongs to a compact set* $D \subset R$ *dominates the interval* $j\colon a_j \le x \le b_j$, *if*

$$L_i(\omega) \le L_j(\omega) \quad \text{for all } \omega \in \boxtimes, \tag{7}$$

$$L_i(\omega) < L_j(\omega), \text{ for at least one } \omega \in \boxtimes. \tag{8}$$

Definition 5. *The interval* $j\colon a_j \le x \le b_j$ *that belongs to the compact set* $D \subset R$ *is called Pareto Optimal (PO)* [2] *if there is no dominant interval* i *defined by (7) and (8).*

Assume that the minimal constant is $\omega = 0$. This constant provides that the interval with minimal $f(c_i)$ is in PO. The maximal constant is defined as a number at least as large that just the longest interval is to be included into the PO set. This theoretical framework provides that the reduced PO set includes just two intervals: one with the minimal observation, another with the longest interval. The only exception is the case when there are more then two intervals all with the minimal observed values and the maximal lengths. In such case PLOR select all of them. The formal rule to select the intervals i which belong to the reduced PO set I is as follows

$$i \in I \text{ if } f(c_i) \le f(c_j) \text{ or } l_i \ge l_k, \; j, k = 1, 2, \ldots \tag{9}$$

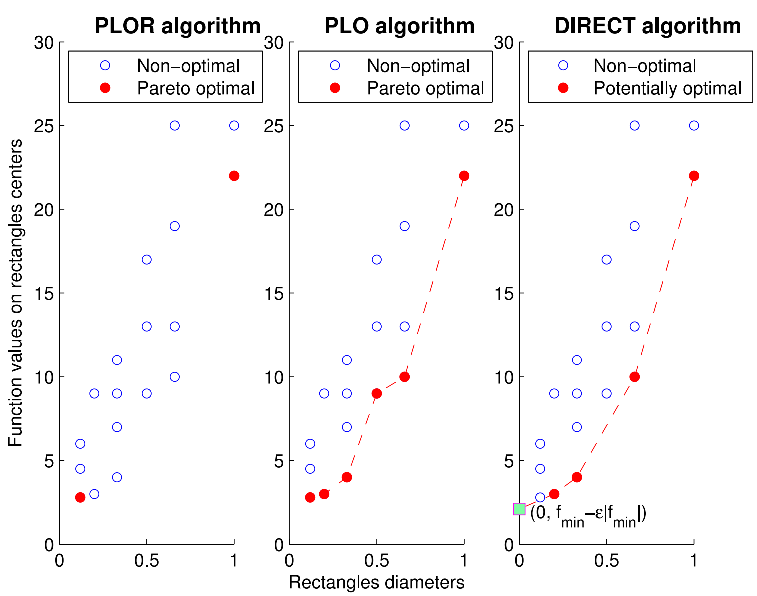

Figure 1. Visual comparison of selection Pareto Optimal intervals using reduced Pareto (PLOR) and Pareto (PLO) algorithms and selection of potentially optimal intervals using (DIRECT) algorithm.

[2] Here PO decisions are defined as indexes of PO intervals. In expression (??), PO decisions are expressed as continuous variables x.

Figure 1 illustrates the PLOR expression (9), the PLO Definitions 4, 5, and the DIRECT Definition 1. Each point on the graphs represents an interval[3], where the horizontal axis represents the distance from the center of the interval to one of its corners and on the vertical axis the value of the function f evaluated at the interval's center. The red (black, in black-white) circles represent intervals which are selected by the corresponding algorithms and will be divided in the next phase of the iteration of these algorithms.

4. Extension to several dimensions

We define the length l_i of the closed K-dimensional interval $[a_i^k, b_i^k] \subset [a^k, b^k]$, $k = 1, \ldots, K$ as

$$l_i = \sqrt{\sum_k (l_i^k)^2}. \tag{10}$$

where $l_i^k = b_i^k - a_i^k$. The observation points c_i are in the middle $c_i^k = (b_i^k + a_i^k)/2$, $k = 1, \ldots, K$,

4.1. Sampling

The definition of PO intervals depends on ω which is unknown, so the rational strategy of sampling (choosing the points where the objective function is to be evaluated) is to investigate both the PO intervals.

To retain a symmetry, instead of picking one side of the PO interval, two additional observations are made in the middle of two additional intervals, produced by dividing the initial PO interval along the longest dimension into three equal parts. After the division, the new two PO intervals are defined by the condition (9).

Figure 2 illustrates first four iterations of the PLOR algorithm by a two-dimensional Shubert (Hedar, 2005) example. Each iteration $i = 1, \ldots, n$ includes the following tasks:

- The basic task is to make observations (calculations of $f(c_i)$ at fixed c_i). The dots in the figure show the observation points, the accompanying numbers are function values on these points.
- An auxiliary task consists of four parts:
  - Definition of the PO intervals. The red (different shade, in gray) indicate PO rectangles in the current iteration by expression (9).
  - Creating new intervals by splitting the current PO intervals along the longest dimension. Figure 2 illustrates a specific situation when PLOR selects more than two identical intervals with equal function values and lengths.

[3] The general term "interval", that means an interval in the k-dimensional space ($k = 1, \ldots, K$), is used while explaining formulas. The term "rectangle" explains pictures better.

- Defining new observation points in the middle of the new intervals.
- Keeping the current best observation which will be accepted as the solution at the end of the optimization process.

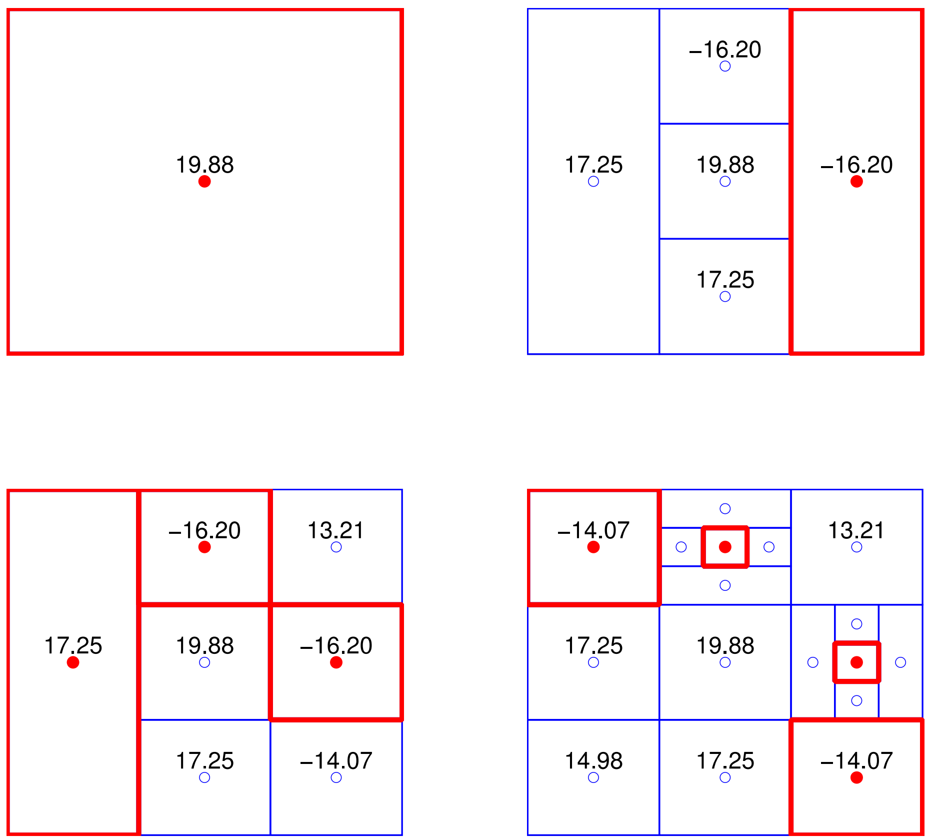

Figure 2. First four iterations of partitioning and selection of Pareto optimal intervals using plor algorithm on Shubert test problem.

The algorithm stops when a point $\overline{x}$ is generated such that

$$\begin{aligned} &\frac{f(\overline{x})-f^*}{|f^*|} \leq 10^{-4}, \quad f^* \neq 0 \\ &f(\overline{x}) \leq 10^{-4} \qquad f^* = 0, \end{aligned} \tag{11}$$

or the number of generated intervals exceeds 500000. Here $f(\overline{x})$ is the value of a test function in Table 1, at some fixed vector X of optimization variables.

4.2. Convergence

It follows from (9) that the set of PO intervals includes the longest interval. Thus, the longest intervals will be divided into three equal parts until they reach the error limit

$$\max_{i=1,\ldots,n} \ 1/2 \ l_i \leq \varepsilon_l(n). \tag{12}$$

This limit is reached after the finite number of partitions, since $[a_i^k, b_i^k] \subset [a^k, b^k],\ k = 1,\ldots,K$. That proves the following proposition: For any $\varepsilon_l > 0$ there exists a number n_ε such that $\varepsilon_l(n) \leq \varepsilon_l$, if $n \geq n_\varepsilon$.

5. Experimental calculations, preliminary results

In this section, the efficiency of the proposed algorithm PLOR is compared with the well-known DIRECT algorithm. A list of used test problems is presented in Table 1. These test problems were investigated by (Jones, Perttunen, & Stuckman, 1993) and (Hedar, 2005). The global minimum of the Griewank function is in the center of the feasible set, so the DIRECT algorithm would find it in the first iteration. Therefore, we slightly changed the feasible region, for meaningful comparison.

Table 1. Description of test problems.

Function name	n	D	No. of global minimizers	Global minimum
Ackley	2	$[-15,30]^2$	1	0.000
Branin	2	$[-5,10] \times [0,15]$	3	0.398
Easom	2	$[-100,100]^2$	1	−1.000
Goldstein-Price	2	$[-2,2]^2$	1	3.000
Griewank	2	$[-600,500]^2$	1	0.000
Michalewics-2	2	$[0,\pi]^2$	1	−1.801
S-H. Camel B.	2	$[-3,3] \times [-2,2]$	2	−1.032
Shubert	2	$[-10,10]^2$	18	−186.831
Hartman-3	3	$[0,1]^3$	1	−3.863
Shekel-5	4	$[0,10]^4$	1	−10.153
Shekel-7	4	$[0,10]^4$	1	−10.403
Shekel-10	4	$[0,10]^4$	1	−10.536
Michalewics-5	5	$[0,\pi]^5$	1	−4.688
Hartman-6	6	$[0,1]^6$	1	−3.322

In Table 2, the proposed PLOR, and the original DIRECT algorithms are compared. The efficiency is defined as the number of function evaluations (n.f.eval.) until algorithm generates a trial point $\bar{x}$ such that inequality (11) is satisfied.

The results show that PLOR is very competitive with DIRECT since, PLOR was more efficient in 11 problems out of 14 with no help from manually adjustable parameters which are common in many well-known algorithms.

Concluding remarks

A theoretical contribution of this paper is the development of the Reduced-Set

Pareto-Lipschitz Optimization (PLOR) which exploits advantages of the sequential optimization of the problem and provides the algorithm without adjustable parameters. It is important as compared with other methods and convenient for users.

An extensive computer simulation with various test functions may reveal additional aspects of the proposed algorithm and that would be an interesting new investigation. The present description of PLOR can serve as a basis for discussions on the possibilities and limitations of the Pareto optimality criteria for various applications of the Lipschitzian optimization to functions with unknown Lipschitz constants.

Table 2. Comparison of PLOR and DIRECT.

Function name	**PLOR**		**DIRECT**	
	$f(\overline{x})$	*n.f.eval*	$f(\overline{x})$	*n.f.eval.*
Ackley	0.00005646	**649**	0.00005646	705
Branin	0.39790038	**85**	0.39790038	195
Easom	-0.99998998	**32833**	-0.99998998	32845
Goldstein-Price	3.00009038	**85**	3.00009038	191
Griewank	0.00007949	60231	0.00007949	**7099**
Michalewics-2	-1.80127241	**55**	-1.80127241	69
S-H. Camel B.	-1.03162357	**269**	-1.03162357	285
Shubert	-186.72153725	**1641**	-186.72153725	2967
Hartman-3	-3.86245215	**111**	-3.86245215	199
Shekel-5	-10.15234984	6857	-10.15234984	**155**
Shekel-7	-10.40196762	**133**	-10.40196762	145
Shekel-10	-10.53539008	**133**	-10.53539008	145
Michalewics-5	-4.53765578	500000	-4.6872130	**13537**
Hartman-6	-3.32207380	**311**	-3.32207380	571

References

Evtushenko, Y. G. (1985). Numerical optimization techniques. New York: Optimization software, Inc.

Figueira, J., Greco, S., & Ehrgott, M. (2004). Multiple Criteria Decision Analysis:State of the Art Surveys. Berlin: Springer.

Finkel, D., & Kelley, C. (2006). Additive Scaling and the DIRECT Algorithm. Journal of Global Optimization , 36, 597-608.

Gablonsky, J., & Kelley, C. (2001). A Locally-Biased form of the DIRECT Algorithm. Journal of Global Optimization , 21, 27-37.

Hedar, A. (2005). Hedar, A. Retrieved from http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO.htm

Jones, D., Perttunen, C., & Stuckman, B. (1993). Lipschitzian Optimization Without the Lipschitz Constant. Journal of Optimization Theory and Application, , 79, 157-181.

Ko, K.-I. (1991). Complexity Theory of Real Functions. Boston: Birkhauser.

Miettinen, K. (1999). Nonlinear Multiobjective Optimization. Boston: Kluwer Academic Publishers.

Mockus., J. (2011). On the Pareto-Lipschitzian Optimization. Informatica , 22, 521–536.

Pardalos, P., & Siskos, Y. (1995). A Historical Perspective. In Advances in Multi-criteria Analysis. Kluwer Academic Publishers.

Paulavičius, R., & Žilinskas, J. (2007). Analysis of different norms and corresponding Lipschitz constants for global optimization in multidimensional case. Information Technology and Control , 36 (4), 383-387.

Paulavičius, R., Žilinskas, J., & Grothey, A. (2010). Investigation of selection strategies in branch and bound algorithm with simplicial partitions and combination of Lipschitz bounds. Optimization Letters , 4 (2), 173-183.

Pijavskij, S. A. (1972). An Algorithm for finding the absolute extremum of function. Computational Mathematics and Mathematical Physics , 57-67.

Sergeyev, Y., & Kvasov, D. (2006). Global search based on efficient diagonal partitions and a set of Lipschitz constants. SIAM Journal on Optimization , 16 (3), 910-937.

Shubert, G. (1972). A sequential method seeking the global maximum of function. SIAM Journal on Numerical Analysis , 9, 379-388.

Sukharev, A. (1971). On optimal strategies of search of extremum. Computational Mathematics and Mathematical Physics , 910-924.

APIE REDUKUOTĄ PARETO-LIPŠICO OPTIMIZAVIMĄ

Jonas Mockus, Remigijus Paulavičius

Santrauka

Straipsnyje pasiūlyta ir ištirta nauja Pareto-Lipšico optimizavimo versija (PLOR). Parodyti PLOR privalumai lyginant su originalia Pareto-Lipšico optimizavimo versija (PLO) bei su plačiai žinomu DIRECT algoritmu.

Pagrindiniai žodžiai: Pareto optimalumas, Lipschitz funkcijos, sumažintas duomenų rinkinyas, globali optimizacija