Journal of Global Optimization 9: 95–111, 1996.

Simple Linkage: Analysis of a Threshold-accepting Global Optimization Method

M. Locatelli
Dip. Scienze della Informazione — Università di Milano
via Comelico 39/41 — I-20139 Milano (Italy).
E-mail: locatelli@hermes.mc.dsi.unimi.it

F. Schoen *
Dip. Sistemi e Informatica — Università di Firenze
via di Santa Marta, 3 — I-50129 Firenze (Italy).
E-mail: schoen@ingfil.ing.unifi.it
www: www-dsi.ing.unifi.it/~schoen/home.html

Abstract. In a recent paper the authors introduced an infinite class of global optimization algorithms based upon random sampling from the feasible region and local searches started from selected sample points, based upon an acceptance/rejection criterion. All of the algorithms of that class possess strong theoretical properties.

Here we analyze a member of that family, which, although being significantly simpler to implement and more efficient than the well known Multi-Level Single-Linkage algorithm, enjoys the same theoretical properties. It is shown here that, with very high probability, our method is able to discover from which points Multi-Level Single-Linkage will decide to start local search.

Key words: Randomized algorithms, Multi Level Single Linkage, Local Searches

Introduction and basic definitions

In this paper we consider the simply bounded global optimization problem:

$$f^\star := \operatorname{glob}\max f(x) : x \in \Omega := [0,1]^d \subset \mathbb{R}^d$$

where f is a continuous function and $d > 0$ is an integer. We shall assume that a local optimization algorithm is available which, given a starting point, is capable of producing a local optimum. If $x^\star$ is a local optimum, the subset of points $\mathcal{A}(x^\star)$ characterized by the property that the local search, when started from any point in $\mathcal{A}(x^\star)$ will lead to the local optimum $x^\star$, is called the *region of attraction* of $x^\star$. We shall assume that f, besides continuity, possess all of the smoothness properties required by the local search employed. The region of attraction of the global optima is assumed to have non-null Lebesgue measure.

* corresponding author

We now briefly recall the basic definitions of both the proposed algorithm and of Multi Level Single Linkage (in short, MLSL).

Simple Linkage

The algorithm proposed in this paper is a member of a family, denominated "Threshold Random Linkage" (or, in short, TRL), first introduced and analyzed in (Locatelli and Schoen, 1995).

Here we briefly recall the definition and main properties of TRL. The algorithm proceeds by sequentially sampling points in Ω according to the uniform distribution. Let $X_1, \ldots, X_i$ be a uniform sample from $[0,1]^d$ available at step i. As soon as X_i is sampled, a decision is taken about whether or not to start a local search from that point. In particular, at step i, a local search is started from X_i if and only if

$$\min_{j<i}\{\|X_i - X_j\| : f(X_j) + \epsilon \geq f(X_i)\} \geq \alpha_i \tag{1}$$

where α_i is a threshold parameter, ϵ is a small positive constant and $\|\cdot\|$ denotes the Euclidean norm. In words, TRL will start a local search from the last sampled point if in a prescribed neighborhood there is no other sampled point with better (or just slightly worse) functional value.

In (Locatelli and Schoen, 1995) several theoretical properties of this algorithm were proven. We summarize here the main results:

- the global optimum will be observed after a finite number of iterations with probability one. By this we mean that, given any prefixed accuracy level $\eta > 0$, a sample point with function value not lower than $f^\star - \eta$ will eventually be found. We remark that this is a trivial result enjoyed, in particular, by all methods which are based upon a sample which assigns positive probability to every set with non null Lebesgue measure;

- when $i \rightarrow \infty$, the probability of starting a local search from X_i is asymptotically decreasing to 0 if and only if

$$\lim_{i\rightarrow\infty} \alpha_i i^{1/d} = \infty;$$

- when $i \rightarrow \infty$, if α_i does not tend to 0 "too fast", then the expected number of local searches started will be finite even if the algorithm is never stopped;

- for sufficiently great i, a local search will be started from X_i if and only if

$$\min_{j<i}\{\|X_i - X_j\|\} \geq \alpha_i. \tag{2}$$

In other words, for sufficiently large i, thanks to the continuity of f, the compactness of the feasible set and the presence of the parameter ϵ in (1), we can asymptotically neglect the effect of function values when deciding about the start of a local search.

We notice in passing that the introduction of ϵ in (1), while almost irrelevant from a computational point of view, greatly simplifies the theoretical analysis.

A possible choice for α_i, inspired from MLSL, is the following:

$$\alpha_i = \frac{1}{\sqrt{\pi}} \left(\sigma \Gamma(1 + d/2) \frac{\log i}{i} \right)^{1/d} \tag{3}$$

where σ is a constant to be chosen by the user. We adopt the convention of calling a TRL algorithm with threshold given as in (3) a "Simple Linkage" algorithm, or SL. Simple Linkage thus is just a particular member of the TRL family introduced and analyzed in the previously cited paper.

The general theoretical results, specialized for SL, imply that, provided that $\sigma > 0$, the probability of starting a local search tends to 0. Moreover, it was proven in the cited paper that, if $\sigma > 1$ and if it is assumed that no local search is started from within a prescribed distance from the boundary of the feasible region, then the algorithm will perform a number of local searches whose expectation is finite even if the algorithm were never stopped. Although similar results could be proven even without assuming that local searches are never started very close to the boundary, here, in order to be able to make significant comparisons with MLSL, we will restrict the analysis to this special case.

Multi Level Single Linkage

For what concerns MLSL, a positive constant σ_M and an integer constant $N > 0$ are given, and sampling proceeds in batches. The decisions about starting or not local searches can be taken only after N points have been drawn — so that the decision epochs are N, $2N$, ... At each decision epoch, say hN, with $h \geq 1$, a threshold is computed

$$\delta_{hN} = \frac{1}{\sqrt{\pi}} \left(\sigma_M \Gamma(1 + d/2) \frac{\log hN}{hN} \right)^{1/d} . \tag{4}$$

Given this threshold, the whole sample of hN points is reconsidered and a local search is started from X_i, $i \in \{1 \ldots, hN\}$ if and only if

$$\min_{j \leq hN, j \neq i} \{ \|X_i - X_j\| : f(X_j) + \epsilon \geq f(X_i) \} \geq \delta_{hN} \tag{5}$$

which again, if hN is sufficiently large, reduces to

$$\min_{j\leq hN, j\neq i}\{\|X_i - X_j\|\} \geq \delta_{hN}. \qquad (6)$$

Our definition of MLSL differs from the original one proposed in (Rinnooy Kan and Timmer, 1987a; Rinnooy Kan and Timmer, 1987b) because of the presence of ϵ; with slightly stronger hypotheses on f and a slightly more involved analysis, we could have proven results very similar to those we present in this paper, without the necessity of modifying the acceptance/rejection criterion through the insertion of the parameter ϵ. However we feel that this way the theoretical analysis can be made in a simpler and more comprehensible way.

We also recall here that in the original definition of MLSL it was assumed that no local search is started from within a prescribed distance from the boundary of Ω. In (Locatelli and Schoen, 1995) we have shown how this hypothesis can be dropped. However, for sake of comparison, we will keep the hypothesis here, both for SL and for MLSL. We shall also assume that, letting $D(x, r)$ denote the d-dimensional hypersphere centered at x with radius r, then, at iteration i, all of the hyperspheres $D(X_j, \alpha_i)$, $j = 1 \ldots, i-1$ are entirely contained in Ω. This assumption, somewhat justified by the hypothesis of not letting any local search start from a point which is too near to the boundary, might be dropped at the expense of some more tedious computations. A similar hypothesis is made for MLSL. Again, the interested reader is referred to (Locatelli and Schoen, 1995) where the problem connected with the "boundary effect" is tackled.

Let us briefly denote with

$$Y_k^{(i)} := \min\{\|X_i - X_j\| : j \leq k, j \neq i\}$$

the random variable corresponding to the minimum distance between a sample point X_i and the first k points in the sample. Then a local search from X_i is started in SL if and only if

$$Y_i^{(i)} > \alpha_i$$

while it is started in MLSL from the same point if and only if

$$\exists\, h \geq \left\lceil \frac{i}{N} \right\rceil : Y_{hN}^{(i)} > \delta_{hN}.$$

In summary, the most evident differences between SL and MLSL are:

- SL samples sequentially ($N = 1$), while MLSL samples in batches;

- in SL a local search might be started *only* from the current point, with no need to re-consider previously sampled points; in MLSL, at each decision epoch, the whole sample is reconsidered (with great computational overhead in the computation of distances among all of the points in the sample) and local searches might be started also from points which have been sampled many iterations before.

1. Theoretical comparison between MLSL and SL

In this section a detailed analysis of the relative behaviour of the proposed algorithm versus MLSL will be carried out. Let us start with some terminology and basic notation.

We plan to compare the behaviour of the proposed algorithm and MLSL when the sample is the same. In other words, assuming the same sample has been drawn, do both algorithms perform the same decisions with respect to when to start a local search? In particular, will they both start local searches from the same sampled points?

Let us assume that there exists a constant $\beta > 1$ such that

$$\frac{\sigma_M}{\sigma} = \beta.$$

In the applications this is the most common situation: it has been proven in fact that a finite expected number of local searches in MLSL is obtained for $\sigma_M > 4$, while the same holds for SL when $\sigma > 1$. It is thus sensible, for comparison to choose, for example, $\beta = 4$. The case $\beta \leq 1$ might be analyzed by means techniques very similar to those employed in this paper.

Under our assumptions, it is easily seen that a local search will be started from X_i in MLSL at decision epoch h if and only if

$$Y_{hN}^{(i)} > \beta^{1/d}\alpha_{hN}.$$

Let us denote with S_i and M_i the events that, respectively, SL and MLSL decide to start a local search from X_i. We look for bounds on

$$P(S_i \mid M_i)$$

i.e. on the probability that a local search is started in SL given that, *sometimes after the i–th iteration* also MLSL decides to start a local search from the same point, and on

$$P(\neg S_i \mid \neg M_i),$$

which is the probability that, given that MLSL will never start a local search from a given sampled point, even SL will not.

The main results can be stated as follows:

THEOREM 1. *If $\beta > 1$ and $\sigma > 1$, then, for $i \to \infty$,*

$$P(S_i \mid M_i) \gtrsim \frac{\beta^{\beta\sigma} - 1}{\beta^{\beta\sigma} - 1 + \beta}$$

and

THEOREM 2. *If $\beta > 1, \sigma > 1, \sigma_M > 2$, then, for $i \to \infty$,*

$$P(\neg S_i \mid \neg M_i) \to 1.$$

Here the symbol $\gtrsim$ is used to denote "asymptotic minorization", i.e., $a_k \gtrsim b_k$ if and only if, for every $\epsilon > 0$ $\exists \bar{k}$ such that $k \geq \bar{k}$ implies $a_k \geq b_k - \epsilon$.

We now prove the first result.

Proof.

$$\begin{aligned} P(S_i \mid M_i) &= P(Y_i^{(i)} > \alpha_i \mid \exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN}) \\ &= \frac{P(Y_i^{(i)} > \alpha_i, \exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN})}{P(\exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN})} \end{aligned}$$

where h_0 is the first possible "decision point" for MLSL:

$$h_0 = \left\lceil \frac{i}{N} \right\rceil .$$

Using the result of lemma 3 in the Appendix, after which, if i is sufficiently large, then a local search is surely started in SL if it is not started "too late" in MLSL, we have:

$$\begin{aligned} P(S_i \mid M_i) &= \frac{P(Y_i^{(i)} > \alpha_i, \exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN})}{P(\exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN})} \\ &\geq \frac{P(Y_i^{(i)} > \alpha_i, \exists\, h, h_0 N \leq hN \leq \beta i : Y_{hN}^{(i)} > \delta_{hN})}{P(\exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN})} \\ &= \frac{P(\exists\, h, h_0 N \leq hN \leq \beta i : Y_{hN}^{(i)} > \delta_{hN})}{P(\exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN})} \\ &= \frac{P(Y_{h_0 N}^{(i)} > \delta_{h_0 N}) + \sum_{h=h_0+1}^{h_1} P(Y_{hN}^{(i)} > \delta_{hN}, Y_{jN}^{(i)} \leq \delta_{jN}, \forall\, j : h_0 \leq j < h)}{P(Y_{h_0 N}^{(i)} > \delta_{h_0 N}) + \sum_{h=h_0+1}^{\infty} P(Y_{hN}^{(i)} > \delta_{hN}, Y_{jN}^{(i)} \leq \delta_{jN}, \forall\, j : h_0 \leq j < h)} \end{aligned}$$

where $h_1 = \max\{h : hN \leq \beta i\} = \lfloor \beta i/N \rfloor$.

For sake of brevity let us denote by E_h the event

$$E_h := \{Y^{(i)}_{hN} > \delta_{hN}, Y^{(i)}_{(h-1)N} \leq \delta_{(h-1)N}\}.$$

Now, observing that

$$P(Y^{(i)}_{hN} > \delta_{hN}, Y^{(i)}_{jN} \leq \delta_{jN}, \forall j : h_0 \leq j < h) =$$
$$P(Y^{(i)}_{jN} \leq \delta_{jN}, \forall j : h_0 \leq j < h - 1 \mid E_h)P(E_h)$$

and that the first factor in the last expression can be bounded as in lemma 4 (see the Appendix), we have

$$P(S_i \mid M_i) \geq$$
$$\frac{P(Y^{(i)}_{h_0 N} > \delta_{h_0 N}) + \sum_{h=h_0+1}^{h_1} \frac{h_0}{h-1} P(E_h)}{P(Y^{(i)}_{h_0 N} > \delta_{h_0 N}) + \sum_{h=h_0+1}^{h_1} \frac{h_0}{h-1} P(E_h) + \sum_{h=h_1+1}^{\infty} P(E_h)} \geq$$
$$\frac{P(Y^{(i)}_{h_0 N} > \delta_{h_0 N}) + \frac{h_0}{h_1 - 1} \sum_{h=h_0+1}^{h_1} P(E_h)}{P(Y^{(i)}_{h_0 N} > \delta_{h_0 N}) + \frac{h_0}{h_1 - 1} \sum_{h=h_0+1}^{h_1} P(E_h) + \sum_{h=h_1+1}^{\infty} P(E_h)}$$

From lemma 5 we obtain

$$P(Y^{(i)}_{h_0 N} > \delta_{h_0 N}) \sim (h_0 N)^{-\beta\sigma}$$

for $h_0 \to \infty$, while, from lemma 6, we have

$$P(E_h) \sim \frac{\beta\sigma \log h}{h^{\beta\sigma+1} N^{\beta\sigma}}.$$

Using the bounds and the asymptotic results of lemma 7 and the fact that, for $i \to \infty$, $h_0 \sim i/N$ and $h_1 \sim \beta i/N$, we finally obtain

$$P(S_i \mid M_i) \gtrsim$$
$$\frac{(h_0 N)^{-\beta\sigma} + \frac{h_0}{h_1 - 1} \frac{\beta\sigma}{N^{\beta\sigma}} \sum_{h=h_0+1}^{h_1} \frac{\log h}{h^{\beta\sigma+1}}}{(h_0 N)^{-\beta\sigma} + \frac{h_0}{h_1 - 1} \frac{\beta\sigma}{N^{\beta\sigma}} \sum_{h=h_0+1}^{h_1} \frac{\log h}{h^{\beta\sigma+1}} + \frac{\beta\sigma}{N^{\beta\sigma}} \sum_{h=h_1+1}^{\infty} \frac{\log h}{h^{\beta\sigma+1}}} \sim$$

$$\begin{aligned}
&\frac{i^{-\beta\sigma} + \frac{1}{\beta N^{\beta\sigma}}\left(\frac{\log i/N}{(i/N)^{\beta\sigma}} - \frac{\log \beta i/N}{(\beta i/N)^{\beta\sigma}}\right)}{i^{-\beta\sigma} + \frac{1}{\beta N^{\beta\sigma}}\left(\frac{\log i/N}{(i/N)^{\beta\sigma}} - \frac{\log \beta i/N}{(\beta i/N)^{\beta\sigma}}\right) + \frac{1}{N^{\beta\sigma}}\frac{\log \beta i/N}{(\beta i/N)^{\beta\sigma}}} \sim \\
&\frac{1 + \frac{1}{\beta N^{\beta\sigma}}\left(\log i - \frac{\log i}{\beta^{\beta\sigma}}\right)}{1 + \frac{1}{\beta N^{\beta\sigma}}\left(\log i - \frac{\log i}{\beta^{\beta\sigma}}\right) + \frac{\log i}{\beta^{\beta\sigma}}} \sim \\
&\frac{\beta^{\beta\sigma} - 1}{\beta^{\beta\sigma} - 1 + \beta}
\end{aligned}$$

which concludes the proof.

We now prove the second main result, which is straightforward.

Proof. When $i \to \infty$, we have

$$\begin{aligned}
P(\neg S_i \mid \neg M_i) &= P(Y_i^{(i)} \leq \alpha_i \mid Y_{hN}^{(i)} \leq \delta_{hN} \, \forall\, h \geq h_0) \\
&= \frac{1 - P(S_i \bigcup M_i)}{1 - P(M_i)} \\
&\geq \frac{1 - P(S_i) - P(M_i)}{1 - P(M_i)} \to 1
\end{aligned}$$

Here we used the fact that, if $\sigma > 0$ and if $\sigma_M > 2$ then the probability of starting a local search tend to 0 both for SL and for MLSL.

The most important implication of these two theorems is that we can substantially simplify MLSL by using SL. In fact the decision of starting a local search with MLSL is a complex one, based upon the consideration of the whole sample and the analysis of chains of points hopefully belonging to the region of attraction of the same local optimum; it has thus been shown that with very high probability the analysis of such chains, at least asymptotically, leads to the same decisions which are taken disregarding the chain effect.

Looking back to the origin of MLSL, we may now ask ourselves which was the rationale behind the choice of proceedings in batches of N samples. For what concerns the "chain effect", the choice of N is irrelevant; however, due to the fact that the threshold used in MLSL decreases to 0, points which have been assigned to a chain in an early stage, might become un-assigned later on. The analysis of the computational overhead caused by this situation lead to the introduction of batch sampling. We have proven here that SL, in some sense, simulates the chain effect of MLSL without having the necessity of looking back to previously "chained" points. This has the positive effect of letting us sample sequentially ($N = 1$) and, most important, not revising previously assigned links. This combined effect enables to avoid the decision about sample size, N, which usually has quite a relevant effect on computational efficiency.

Analyses similar to the one presented here can be carried on in order to bound the probability that MLSL starts a local search when SL does; moreover it is possible also to generalize the results to the case in which $\beta \leq 1$. Indeed, from Theorem 1 it is seen that when $\beta \approx 1$ the lower bound on the probability of starting a local search from a point with SL given that MLSL does, tends to 0. As the bound is quite tight, this seems to imply that the similarity between SL and MLSL is achieved only for sufficiently large values of β. This is indeed confirmed by the analysis of the case $\beta \leq 1$ where it can be shown that this probability tends to 0.

1.1. Computational experiments

A few numerical experiments have been carried out in order to see whether the asymptotic results reported in this paper can be actually observed in finite time. We thus compared the behaviour of MLSL with $\sigma_M = 4$ and SL with $\sigma = 2$ by running both algorithms on the well known 2-dimensional penalized Shubert function (see, e.g., (Lucidi and Piccioni, 1989)):

$$\begin{aligned} f(x_1, x_2) &= -\prod_{i=1}^{2} \sum_{j=1}^{5} (j \cos((j+1)x_i + j)) \\ &\quad -0.5((x_1 + 1.42513)^2 + (x_2 + 0.80032)^2) \end{aligned}$$

with $x_1, x_2 \in [-10, 10]$. This function has 760 local optima and a single global optimum. While not being a particularly difficult test function for global optimization methods which can use functional structure (like, e.g., bounds on the derivatives), it is quite a challenging one for methods exclusively based on sampling and local searches.

A total of 10 runs of both algorithms were performed, each run being based upon uniform samples of 10,000 points; both algorithms were run using the same sample. No stopping rule was used, except the termination at the 10,000–th iteration. Needless to say, both algorithms did actually discover the global optimum well before the 10,000 iterations limit.

In table I the statistics collected in these experiments are reported. The column headings in the table have the following meaning: in the first two columns we report the total number of local searches performed by SL and by MLSL; in the third column the total number of sampled points from which *both* SL *and* MLSL started a local search; in

Table I. Computational results

Run	SL	MLSL	both	$P(S\|M)$	$P(\neg S\|\neg M)$
1	569	395	331	83.80%	97.52%
2	576	374	311	83.16%	97.25%
3	580	373	301	80.70%	97.10%
4	594	401	328	81.80%	97.23%
5	619	384	327	85.16%	96.96%
6	641	390	339	86.92%	96.86%
7	586	375	309	82.40%	97.12%
8	647	389	340	87.40%	96.81%
9	578	401	328	81.80%	97.40%
10	588	383	322	84.07%	97.23%
Average	597.8	386.5	323.6	83.73%	97.15%
Std.dev.	27.89	10.54	12.89		

the last two columns we report respectively the computed percentage of times SL started a local search given that MLSL did the same and the percentage of times SL did not start a local search given that MLSL also did not started one. If we recall that, with the values chosen for the parameters in these experiments, we have $\beta = 2$ and $\sigma = 2$, from the theoretical analysis we deduce that, asymptotically, the probability of starting a local search in SL given that it was started in MLSL should be at least 15/17, or 88.24%. The similarity between the asymptotic bound and the observed one is striking. As an illustration, in figure 1, the points used as starting points for both MLSL and SL in the first of the ten experiments are represented. A small square in the negative quadrant corresponds to the global optimum.

Obviously not very much can be concluded by such a limited experiment; however it is quite surprising to notice that the bounds we obtained using asymptotic considerations and, perhaps most significantly, disregarding function values, are extremely close to the observed ones. This fact surely deserves a deeper analysis. For what concerns computational times, our first implementation of MLSL, using no special data structure for storing distances, was extremely penalizing for this algorithm; a revision of our MLSL code lead to a drastic reduction in the computational overhead caused by distance computation. On a SUN Sparc Station, the ten experiments required 0.79 CPU seconds for SL and 1.23 CPU seconds for MLSL. In order to appreciate this difference in performance, it can be observed that, while the ratio of CPU times required by SL and MLSL is approximately 64%, the analogous one for the total number of local searches is roughly 155%. In other

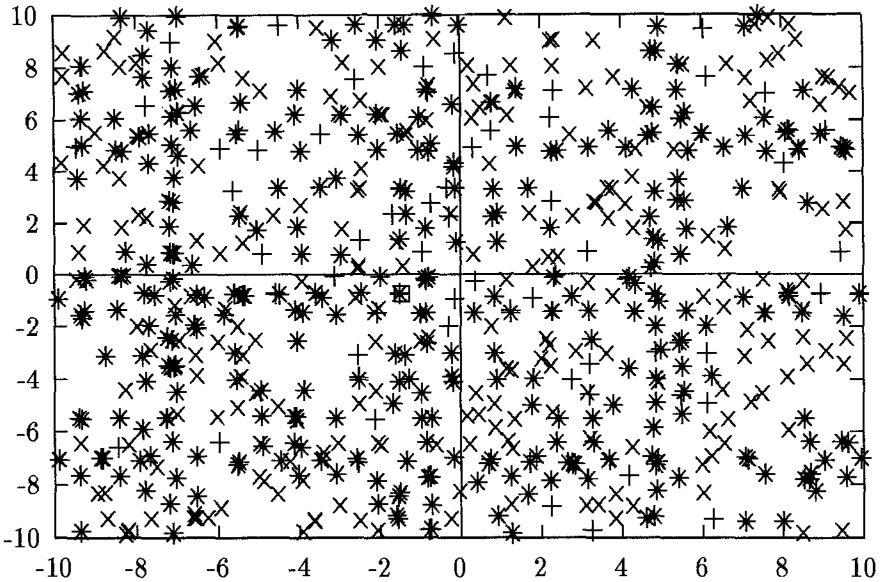

Figure 1. Local searches started from MLSL (+) and by SL (×)

words, SL was roughly twice as fast as MLSL, even if it started a much higher number of local searches. From this we deduce that the overhead caused by the necessity of looking back to previously sampled points, which is peculiar of MLSL, is quite relevant. In our implementation of MLSL the sample was organized as a linked ordered list, with the current value of the nearest neighbor distance associated to each sample point. Using this organization, when the k–th point is sampled, a total number of $k-1$ distances have to be evaluated; we need in fact to compute not only the distance between the current point and sample points with better function values, but also the distance between the current point and worse ones, in order to possibly update the distance information of these ones. Thus, if at stage h, that is after hN points have already been sampled, a batch of N new points is sampled, the total number of distances to be computed is $O(hN^2)$. If we consider that SL requires the nearest neighbor distance to be computed only from the most recently sampled point, in the worst case we observe that the total number of computation required is the same as in MLSL. However two important differences make the practical behaviour of SL much better than MLSL: first of all, MLSL requires, at each step, exactly $O(hN^2)$ distances to be evaluated; SL requires the same order of magnitude only in the worst case, which corresponds to the case in which all of the most recent N sampled points are "negative records", that is each of them has the lowest functional value up to the time it is sampled. It is evident that this event is very unlikely. Moreover, in order to be able to correctly implement MLSL, the exact nearest neighbor distance has to be stored with each sampled point; in SL, however, as soon as a sample point whose distance to the current one is lower than the current threshold is found, we may stop and decide not to start a local search, thus avoiding many useless distance calculations. It should be noticed that, as the probability of starting a local search goes to 0, the savings in computations due to this shortcut become more and more sensible. Unfortunately an exact estimate of the average overhead in MLSL an SL seems to be very difficult unless unrealistically simple models of the objective function are assumed. Moreover different data structures might significantly change our estimate of the computational overhead. It is a current research issue to try to develop an efficient data structure for the implementation of SL. Details will appear elsewhere.

2. A generalization of Simple Linkage

The analysis carried out in the preceding section makes it possible to conclude that, provided that $\beta > 1$, SL will very often start local searches from the points chosen by MLSL. However, as β approaches 1, the two algorithms tend to become quite different. In this section we introduce a generalized SL algorithm in which the threshold parameter σ is not constant, but becomes 1 in the limit. The purpose of this modification is to analyze the relative behaviour of MLSL and SL when their characteristic parameters tend to be close each other.

Let us modify the definition of SL by substituting the threshold parameter σ with a sequence σ_i such that $\sigma_i < \sigma_M$ and $\lim_i \sigma_i = \sigma_M$; thus, defining $\beta_i := \sigma_M / \sigma_i$, we obtain $\beta_i \downarrow 1$.

It is immediate to prove an analogue of theorem 1 for this case, the only modification required being that, in lemma 3, it should be assumed that β_i goes to 1 "not too fast"; in particular, it is necessary that, at least for sufficiently large values of i,

$$\beta_i \geq \frac{N-1}{i} + 1$$

Provided this assumption is made, it is trivial to derive the following result:

$$P(S_i \mid M_i) \gtrsim \frac{\beta_i^{\sigma_M} - 1}{\beta_i^{\sigma_M} - 1 + \beta_i}$$

From this it can be observed that, when $\beta_i \downarrow 1$, the lower bound on the probability of starting local searches in SL given that MLSL did the same, tends to 0. This obviously does not imply that the probability goes to 0, even if, being the minorizations in the proofs quite stringent, it is quite likely that this will be effectively the case. Thus it seems that, as $\beta_i \downarrow 1$, the decisions taken by SL will be more and more different from those of MLSL. It is in this case instructive to look also at bounds on the event that MLSL starts local searches when SL does. A crude minorization is obtained as follows:

$$\begin{aligned} P(M_i \mid S_i) &= \frac{P(Y_i^{(i)} > \alpha_i, \exists\, h \geq h_0 : Y_{hN}^{(i)} > \delta_{hN})}{Y_i^{(i)} > \alpha_i} \\ &\geq \frac{P(Y_{h_0N}^{(i)} > \delta_{h_0N})}{P(Y_i^{(i)} > \alpha_i)} \\ &\sim \frac{(h_0N)^{-\sigma_M}}{i^{-\sigma_i}} \\ &\sim \exp\left(-\sigma_M(1-\frac{1}{\beta_i})\log i\right) \end{aligned}$$

This minorization shows that the relative behaviour of MLSL and SL is influenced by the speed of convergence of β_i to 1. In particular, if the speed is sufficiently high, then, with probability 1, MLSL will start local searches from all the points used by SL as starting points. On the other hand, by slowing down the speed of convergence of β_i, we obtain bounds going to 0, thus permitting less and less searches to be started from the same points.

Thus we may conclude observing that, if the parameter of SL is strictly lower than that of MLSL, then SL will perform many local searches and, in particular, most of the local searches performed by MLSL. On the other side, if the threshold parameter of SL is made closer and closer to that of SL at a sufficiently high speed, then the reverse property holds: MLSL will perform more local searches than SL, and, in particular, it will start local searches from all of the points from which SL did the same. An intermediate situation occurs when the speed of convergence of σ_i to σ_M is slow.

Conclusions

In this paper an analysis of the similarities between Multi-Level Single-Linkage and a new threshold-accepting method for global optimization has been carried out. The most important result of this analysis is that, by means of suitable tuning of the characteristic parameters of the two methods, their computational behaviour may be made very similar. In particular, both the probability that the new algorithm, SL, starts or do not start a local search from a sampled point, given that MLSL does the same, asymptotically is very near to 1. Thus SL can be seen as a simulator of MLSL which avoids looking back to previously sampled point as possible candidates for starting a local search. Worst case analysis of the savings caused by this significant difference between SL and MLSL does not enable to discriminate between the two approaches. However simple considerations on the average behaviour of the two methods permit to conclude that in practice the overhead in SL will be significantly lower than in MLSL. A few computational experiments confirm this analysis.

As a concluding remark, we would like to add that in this paper we have shown how it is possible to obtain theoretical as well as computational results very close to those of MLSL with a simpler algorithm, SL; obviously our main aim is to find a global optimization algorithm which is both theoretically and practically reliable and efficient. Thus future research will concentrate on the optimal tuning of the parameter of SL; moreover, as SL is just a particular member of a larger family of global optimization algorithms, the question naturally arises of identifying one or a few methods which are in some sense the best ones. This is still an open research issue which we hope to be able to address in the future.

Appendix

LEMMA 3. *Let $h_0 := \lceil i/N \rceil$; if $\beta > 1$, $i > \frac{N}{\beta - 1}$ and $\exists\, h$ integer with $h_0 N \leq hN \leq \beta i$ such that*

$$Y_{hN}^{(i)} > \delta_{hN}$$

then

$$Y_i^{(i)} > \alpha_i$$

Proof. From the assumption made on i it follows that the index set

$$\{h : h_0 N \leq hN \leq \beta i\}$$

is non empty. Then it immediately follows that if there exists an $h \geq h_0$ such that

$$Y_{hN}^{(i)} > \delta_{hN}$$

then, recalling that $Y_j^{(i)}$ is non-increasing in j and exploiting the relationship between the thresholds of MLSL and SL, we obtain

$$\begin{aligned} Y_i^{(i)} &\geq Y_{hN}^{(i)} \\ &> \beta^{1/d} \alpha_{hN} \end{aligned}$$

$$\begin{aligned} &= \frac{1}{\sqrt{\pi}}\left(\beta\sigma\Gamma(1+d/2)\frac{\log hN}{hN}\right)^{1/d} \\ &\geq \frac{1}{\sqrt{\pi}}\left(\beta\sigma\Gamma(1+d/2)\frac{\log \beta i}{\beta i}\right)^{1/d} \\ &\geq \frac{1}{\sqrt{\pi}}\left(\sigma\Gamma(1+d/2)\frac{\log i}{i}\right)^{1/d}. \end{aligned}$$

LEMMA 4. *If* $h > h_0 + 1$ *then*

$$P(Y_{jN}^{(i)} \leq \delta_{jN}, \forall j : h_0 \leq j < h-1 \mid E_h) \geq \frac{h_0}{h-1}$$

where

$$E_h := \{Y_{hN}^{(i)} > \delta_{hN}, Y_{(h-1)N}^{(i)} \leq \delta_{(h-1)N}\}$$

Proof. The event E_h corresponds to the event that no point, among the first hN sampled ones, is placed in $D(X_i, \delta_{hN})$, while at least one out of the first $(h-1)N$ points falls in $D(X_i, \delta_{(h-1)N}) \setminus D(X_i, \delta_{hN})$. Let us denote by $\tilde{E}$ the event:

$$\{\exists j \leq h_0 : X_j \in D(X_i, \delta_{(h-1)N}) \setminus D(X_i, \delta_{hN})\}$$

Then, recalling that both $Y_{(\cdot)}^{(i)}$ and $\delta_{(\cdot)}$ are non increasing, we have

$$\begin{aligned} \tilde{E} \Rightarrow\ & Y_{h_0 N}^{(i)} \leq \delta_{(h-1)N} \\ & \Downarrow \\ Y_{jN}^{(i)} \ \leq\ & \delta_{jN} \quad \forall j : h_0 \leq j < h-1 \end{aligned}$$

so that

$$P(Y_{jN}^{(i)} \leq \delta_{jN}, \forall j : h \leq j < h-1 \mid E_h) \geq P(\tilde{E} \mid E_h)$$

The last term corresponds to the probability that a point in the sample falls in a prescribed set given that at least one of the first $(h-1)N$ points falls there. Being the sample i.i.d., given any region with positive volume, each point has the same chance of any other of falling in such a region. Thus this probability cannot be less than $h_0/h - 1$.

It can be observed that the last bound cannot be improved upon. In fact, direct computation of the required probabilities enables to show that $P(\tilde{E} \mid E_h) \sim h_0/h$.

LEMMA 5.

$$P(Y_k^{(i)} > \delta_k) \sim k^{-\beta\sigma}$$

for $k \to \infty$.

Proof. Let us denote with $\mu(\cdot)$ the Lebesgue measure. Then, for any $r > 0$, we have

$$\begin{aligned} P(Y_k^{(i)} > r) &= P(\|X_j - X_i\| > r\,\forall j \leq k, j \neq i) \\ &= (1 - \mu\left(D(X_i, r)\right))^{k-1} \\ &\sim \exp\left(-k\mu\left(D(X_i, r)\right)\right) \end{aligned} \tag{7}$$

under the assumption that $r \to 0$. Recalling the choice of the threshold value of MLSL, we have immediately

$$\mu\left(D(X_i, \delta_k)\right) = \beta\sigma\frac{\log k}{k}$$

so that

$$P(Y_k^{(i)} > \delta_k) \sim k^{-\beta\sigma}$$

when $k \to \infty$.

LEMMA 6.

$$P\left(Y_{hN}^{(i)} > \delta_{hN}, Y_{(h-1)N}^{(i)} \leq \delta_{(h-1)N}\right) \sim \frac{\beta\sigma\log h}{N^{\beta\sigma}h^{\beta\sigma+1}}$$

Proof. Given two integers r, s with $s > r$, we have:

$$\begin{aligned} P\left(Y_s^{(i)} > \delta_s, Y_r^{(i)} \leq \delta_r\right) &= P\left(\|X_i - X_j\| > \delta_s \forall j \leq s, j \neq i,\right. \\ &\qquad \left. \exists \bar{\jmath} \leq r, \bar{\jmath} \neq i : \|X_i - X_{\bar{\jmath}}\| \leq \delta_r\right) \\ &= P\left(\delta_s < Y_r^{(i)} \leq \delta_r, X_{r+1}, \ldots, X_s \notin D(X_i, \delta_s)\right). \end{aligned}$$

Using stochastic independence and the asymptotic expression (7), we obtain that the last quantity is equal to

$$\left(P(Y_r^{(i)} \leq \delta_r) - P(Y_r^{(i)} \leq \delta_s)\right)\left(1 - \mu\left(D(X_i, \delta_s)\right)\right)^{s-r+1} \sim$$
$$\left(\exp\left(-\beta\sigma r\frac{\log s}{s}\right) - \exp\left(-\beta\sigma\log s\right)\right)\exp\left(-(s - r + 1)\beta\sigma\frac{\log s}{s}\right)$$

Letting now $r = (h-1)N$ and $s = hN$, and observing that, for $h \to \infty$,

$$\exp\left(-(N+1)\beta\sigma\frac{\log hN}{hN}\right) \sim 1$$

we obtain

$$P\left(Y_{hN}^{(i)} > \delta_{hN}, Y_{(h-1)N}^{(i)} \leq \delta_{(h-1)N}\right) \sim$$
$$\exp\left(-\beta\sigma(h-1)N\frac{\log hN}{hN}\right)\left(-\exp\left(-\beta\sigma\log hN\right)\right) =$$

$$\exp\left(-\beta\sigma(h-1)\frac{\log hN}{h}\right)\left(1-\exp\left(-\beta\sigma\frac{\log hN}{h}\right)\right)\sim$$
$$\beta\sigma(hN)^{-\beta\sigma}\frac{\log h}{h}.$$

LEMMA 7. *If* $r > 1$ *then, for* $i \to \infty$,

$$\sum_{k=i}^{\infty}\frac{\log k}{k^r}\sim\frac{\log i}{(r-1)i^{r-1}}.$$

Proof. The general term in the summation is, at least for i sufficiently great, decreasing. Thus we can use the following inequalities:

$$\int_{i+1}^{\infty}\frac{\log x}{x^r}\,dx\le\sum_{k=i+1}^{\infty}\frac{\log k}{k^r}\le\int_{i}^{\infty}\frac{\log x}{x^r}\,dx.$$

Integrating by parts we readily obtain the following bound:

$$\frac{\log(i+1)}{(r-1)(i+1)^{r-1}}+\frac{1}{(r-1)^2 i^{r-1}}\le\sum_{k=i+1}^{\infty}\frac{\log k}{k^r}\le\frac{\log i}{(r-1)i^{r-1}}+\frac{1}{(r-1)^2 i^{r-1}}$$

from which the thesis follows.

Acknowledgements

The authors would like to express their thanks to the referee, whose comments helped to improve the presentation.

This research was partially supported by CNR contract n. 94.00149.01 and MURST project “Metodi di Ottimizzazione nella Ricerca Operativa”.

References

Locatelli, M. and Schoen, F.: 1995, ‘Random Linkage: a family of acceptance/rejection algorithms for global optimisation’, Technical Report **156-96**, DSI, University of Milano, *submitted for publication*, available through the www page of the 2^{nd} author.

Lucidi, S. and Piccioni, M.: 1989, ‘Random Tunneling by Means of Acceptance-Rejection Sampling for Global Optimization’. *Journal of Optimization Theory and Applications* **62**, pp. 255–276

Rinnooy Kan, A.H.G. and Timmer, G.T.: 1987, ‘Stochastic Global Optimization Methods. Part I: Clustering Methods’, *Mathematical Programming* **39**, pp. 27–56

Rinnooy Kan, A.H.G. and Timmer, G.T.: 1987, ‘Stochastic Global Optimization Methods. Part II: Multi Level Methods’, *Mathematical Programming* **39**, pp. 57–78