Simple Tools for Multimodal Optimization

Marc Schoenauer
Tao Inria, Lri
Umr Cnrs 8623
Paris-Sud F-91405 Orsay
Cedex
marc.schoenauer@inria.fr

Fabien Teytaud
Tao Inria, Lri
Umr Cnrs 8623
Paris-Sud F-91405 Orsay
Cedex
fabien.teytaud@lri.fr

Olivier Teytaud
Tao Inria, Lri
Umr Cnrs 8623
Paris-Sud F-91405 Orsay
Cedex
olivier.teytaud@inria.fr

ABSTRACT

We compare various approaches for multimodal optimization; we focus on comparing restart and more sophisticated approaches, and on the use of quasi-random numbers.

Categories and Subject Descriptors

G.1.6 [**Optimization**]: Global Optimization

General Terms

Theory

1. INTRODUCTION

In this section we (i) introduce multimodal optimization (ii) introduce quasi-random points (which will be used in our algorithms).

1.1 Multimodal optimization

Multimodal optimization (MMO) sometimes refers to looking for *one* optimum, and avoiding local minima. In this paper, MMO is the research of *all* global optima. Evolution Strategies (ES [17, 1]) are a classical tool for MMO, but there are not so many rigorous analyses of ES for MMO. We will here investigate ES for MMO both theoretically and experimentally. ES are a wide family of algorithms, and many other local search tools could be used instead of ES here; direct search methods [2] including simplex algorithms [21, 14], Hooke&Jeeves [8].

When working on ES-based MMO, the first point is the choice of the ES, before we add some tricks specifically for MMO: which algorithm, which population size λ, which selection rate μ/λ. We point out that our theoretical analysis is independent of the ES; this choice only matters for the experimental part. For choosing λ, it is known that in the MMO setting, λ large reduces the risk of poor local minima [22]. However, as the fitness functions considered in [18] and adopted here for the sake of comparison are locally simple, we use a simple ES, better than sophisticated variants in that case: the 1+1-ES with one-fifth rule [17]. Second, we have to choose how to adapt an ES to MMO. The main algorithms for MMO are as follows: clearing [16], including the modified clearing approach proposed in [18]; this is considered as a niching mechanism; clustering [25]; crowding

[3], including deterministic [12] and probabilistic [13] versions; sharing [5]; species conserving genetic algorithms [11]; and the classical restarts. The detailed presentation of all these algorithms is beyond the scope of this paper. They all depend on various parameters and are almost beyond theoretical analysis due to their complicated dynamics. We here propose some simple rules which can be analyzed efficiently and provide state of the art results. We will compare our results to those in the survey [18] and refer to [18] for a detailed description of all algorithms above. Importantly, we use a murder operator in our restart algorithms; this operator "kills" individuals too close to a previously found optimum, and it is analogous to a parameter used in the modified clearing which outperformed by far all other algorithms in the comparative study [18]. As this parameter is tuned for our testbed, it is of course an unfair comparison with algorithms which have no such parameter. Our conclusion is therefore not that our approach is better than all published approaches; our conclusion is just that we can design a restart algorithm, with theoretical guarantees, and proofs of superiority of the quasi-random restart over the random restart, and with state of the art performance with this context of a parameter d which essentially quantifies our prior knowledge on a minimal distance between two optima.

1.2 Quasi-random (QR) points

Quasi-random points provide a convergence as $1/n$ instead of $1/\sqrt{n}$ in numerical integration, within logarithmic factors. QR points are used in random search [15]; and later they we used in the initialization [9, 4] or in the mutations [20, 19] of evolution strategies. [20, 19] pointed out that "modern" QR points [10], using e.g. scrambling, are much better than old versions of quasi-random points. A quasi-random sequence is a sequence of points with some uniformity properties that will be recalled in this paper. Following [24, 19], all QR sequences in this paper are Halton sequences with random scrambling. Halton sequences[6] are the multidimensional extension of Van Der Corput's sequences[23]. We first define Van Der Corput's sequence. Consider p a prime number. The following procedure generates the nth element $vdc_{n,p} \in [0,1]$ of the Van Der Corput sequence in basis p: write n in basis p: $n = d_k d_{k-1} \ldots d_1$, i.e. $n = \sum_{i=0}^{k} d_i p^i$ with $d_i \in [[0, p-1]]$; define $v_{n,p} = 0.d_1 d_2 \ldots d_k$ in basis p, i.e. $v_{n,p} = \sum_{i=1}^{k} d_i p^{-i}$. For moderate values of n and large values of d, an improvement termed *scrambling* is often used; instead of $v_{n,p}$ as above, define $v_{n,p}$ as $v_{n,p} = 0.\pi(d_1)\pi(d_2)\ldots\pi(d_k)$ where π is some permutation of $[[0, p-1]]$ such that $\pi(0) = 0$ in order to ensure $\forall n, v_{n,p} \neq 0$.

The Halton sequence generalizes the Van Der Corput sequence to dimension D by using one different prime number per dimension. Consider p_i the ith prime number. Then, h_n, the nth element of a Halton sequence in dimension D, is $h_n = (v_{n,p_1}, v_{n,p_2}, \ldots, v_{n,p_D}) \in [0,1]^D$. The scrambled-Halton sequence used in this paper is the use of a randomly drawn permutation for each $i \in [[1,D]]$; more sophisticated versions exist. The Nth Hammersley point set[7] is $\{\text{HAMM}_{N,1}, \text{HAMM}_{N,2}, \ldots, \text{HAMM}_{N,N}\}$ where $\text{HAMM}_{N,n}$ is defined by $\text{HAMM}_{N,n} = ((n-1)/N, v_{n,p_1}, v_{n,p_2}, \ldots, v_{n,p_{D-1}})$.

2. CONCLUSION

The first conclusion is the superiority of quasi-random restarts over random restarts in nearly all tested settings. The second conclusion is that we could not find any superiority of clearing or modified clearing approaches over simple restarts; however, we have had to take care of removing restarts close to previously found optima. This conclusion is limited to sequential optimization; it might be the case that clearing and modified clearing benefit from parallelization more than the simple restart approaches.

3. REFERENCES

[1] H.-G. Beyer. *The Theory of Evolution Strategies.* Springer, Heidelberg, 2001.

[2] A. Conn, K. Scheinberg, and L. Toint. Recent progress in unconstrained nonlinear optimization without derivatives. *Mathematical Programming*, 79:397–414, 1997.

[3] K. A. De Jong. *An analysis of the behavior of a class of genetic adaptive systems.* PhD thesis, 1975.

[4] A. Georgieva and I. Jordanov. A hybrid meta-heuristic for global optimisation using low-discrepancy sequences of points. *Computers and Operations Research, - special issue on hybrid metaheuristics*, In press.

[5] D. E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodalfunction optimization. In J. J. Grefenstette, editor, *ICGA*, pages 41–49. Lawrence Erlbaum Associates, 1987.

[6] J. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2:84–90, 1960.

[7] J.-M. Hammersley. Monte carlo methods for solving multivariable problems. *Ann. New York Acad. Sci.*, 86:844–874, 1960.

[8] R. Hooke and T. A. Jeeves. Direct search solution of numerical and statistical problems. *Journal of the ACM, Vol. 8, pp. 212-229*, 1961.

[9] S. Kimura and K. Matsumura. Genetic algorithms using low-discrepancy sequences. In *GECCO*, pages 1341–1346, 2005.

[10] P. L'Ecuyer and C. Lemieux. *Recent Advances in Randomized Quasi-Monte Carlo Methods*, pages 419 – 474. Kluwer Academic, 2002.

[11] J.-P. Li, M. E. Balazs, G. T. Parks, and P. J. Clarkson. A species conserving genetic algorithm for multimodal function optimization. *Evol. Comput.*, 10(3):207–234, 2002.

[12] S. W. Mahfoud. *Niching methods for genetic algorithms.* PhD thesis, Champaign, IL, USA, 1995.

[13] O. J. Mengshoel and D. E. Goldberg. Probabilistic crowding: Deterministic crowding with probabilisitic replacement. In W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 1, pages 409–416, Orlando, Florida, USA, 13-17 July 1999. Morgan Kaufmann.

[14] J. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal 7*, pages 308–311, 1965.

[15] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods.* 1992.

[16] A. Pétrowski. A clearing procedure as a niching method for genetic algorithms. In *International Conference on Evolutionary Computation*, pages 798–803, 1996.

[17] I. Rechenberg. *Evolutionstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution.* Fromman-Holzboog Verlag, Stuttgart, 1973.

[18] G. Singh and D. Kalyanmoy Deb. Comparison of multi-modal optimization algorithms based on evolutionary algorithms. In *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 1305–1312, New York, NY, USA, 2006. ACM.

[19] O. Teytaud. When does quasi-random work?. In G. Rudolph, T. Jansen, S. M. Lucas, C. Poloni, and N. Beume, editors, *PPSN*, volume 5199 of *Lecture Notes in Computer Science*, pages 325–336. Springer, 2008.

[20] O. Teytaud and S. Gelly. DCMA: yet another derandomization in covariance-matrix-adaptation. In *GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 955–963, New York, NY, USA, 2007. ACM.

[21] V. Torczon. On the convergence of the multidirectional search algorithm. *SIAM journal on Optimization*, 1(1):123–145, 1991.

[22] F. van den Bergh and A. Engelbrecht. Cooperative learning in neural networks using particle swarm optimizers, 2000.

[23] J. G. van der Corput. Verteilungsfunktionen. *Proc. Ned. Akad. v. Wet., 38:813-821*, 1935.

[24] B. Vandewoestyne and R. Cools. Good permutations for deterministic scrambled halton sequences in terms of l2-discrepancy. *Computational and Applied Mathematics*, 189(1,2):341:361, 2006.

[25] X. Yin and N. Germay. A fast genetic algorithm with sharing scheme using cluster analysis methods in multimodal function optimization. In R. F. Albrecht, N. C. Steele, and C. R. Reeves, editors, *Artificial Neural Nets and Genetic Algorithms*, pages 450–457, Wien, 1993. Springer Verlag.