EVOLUTIONARY ALGORITHMS IN DISCRETE OPTIMISATION

Anton V. Eremeev, Colin R. Reeves,

Evolutionary algorithms (EAs)—an umbrella term for techniques such as genetic algorithms (GAs), genetic programming (GP) and evolution strategies (ES)—have become popular tools for attacking discrete optimisation problems, and there are many experimental papers that bear witness to the fact that they can be very effective. GAs in particular have been extensively used in OR (see [15] for a review), and have been the subject of some extravagant claims about their abilities to avoid local optima. However, theoretical developments have lagged some way behind, and it is the purpose of this paper to discuss some of what is known about their behaviour from a theoretical perspective. Much of this material is amplified in [17].

Definitions and some general results. Consider a maximisation problem: find $f^* = \max\{f(x) : x \in X\}$ where X is a finite solutions space. The minimisation problem would be treated analogously (the necessary changes are usually obvious).

Let $\Pi^t = (x_1^t, x_2^t, ..., x_N^t) \in X^N$ be the population of solutions in the EA on iteration t and let $\widehat{\Pi} \subseteq X$ denote the set of all solutions present in population Π. Usually the ordering of solutions in population is unimportant but the same solution may be present in several copies. The population Π^{t+1} is produced on the basis of Π^t using the following four randomised procedures.

The *selection* operator $s : X^N \to X^{N'}$ chooses N' copies of solutions called "parents" from the current population, i.e. $\Pi' = (x'_1, ..., x'_{N'}) = s\,(\Pi^t)$ is such that $\widehat{\Pi'} \subseteq \widehat{\Pi^t}$.

The *reproduction* operator $\rho : X^{N'} \to X^{N''}$ introduces some random changes to the "parents", producing N'' solutions called "offspring" in population $\Pi'' = (x''_1, ..., x''_{N''})$.

The *survival* operator $\sigma : X^N \times X^{N''} \to X^N$ determines which solutions from the population Π^t and from the "offspring" will be combined in Π^{t+1}, thus $\widehat{\sigma}\,(\Pi^t, \Pi'') \subseteq \widehat{\Pi^t} \cup \widehat{\Pi''}$.

In practice the selection and survival operators are designed to "favour" the solutions of better quality (greater objective function values in the case of maximisation problem). This may be done deterministically (e.g. in survival operators of the (μ, λ)-ES and $(\mu+\lambda)$-ES evolution strategies [14,24]) or randomly (e.g. in proportional or tournament selection in genetic algorithms [11,15]).

The EA starts with some randomly generated initial population and proceeds by iterating the random mapping $\Pi^{t+1} = \sigma\,(\rho\,(s\,(\Pi^t))\,, \Pi^t)$ until some termination condition is met. Such condition could be a limit on the number of total iterations, or a limit on the number of iterations without an improvement of the objective function value $f_t =$

Anton Valentinovich Eremeev, Omsk Branch of Sobolev Institute of Mathematics,
13, Pevtsov str., Omsk, 644099, Russia,
phone: +7 (3812) 236739, fax: +7 (3812) 234584, e-mail: eremeev@iitam.omsk.net.ru

Colin Richard Reeves, School of Mathematical & Information Sciences,
Coventry University, UK,
Priory Street, Coventry CV1 5FB, UK,
phone: 024 7688 8579, fax: 024 7688 8080, e-mail: C.Reeves@coventry.ac.uk

$\max_{i \le N} f(x_i^t)$ of the "best" solution in Π^t, or a requirement for f_t to reach a sufficient level. For convenience when talking about the theoretical analysis we will assume that the termination condition is never met, although practical experience shows that multiple independent restarts often yield a significant improvement in the performance of EAs.

As an example of EA let us consider the simple genetic algorithm [11] which has been intensively studied and exploited over three decades. This algorithm operates with populations of binary strings (traditionally called *genotypes*) thus we will assume $X = \{0,1\}^n$ ignoring the issues of representation of other types of solution spaces – the details can be found e.g. in [15]. Here the intermediate populations Π' and Π'' also consist of N solutions and N is even. In selection operator each element of "parent" population $s(\Pi^t)$ is drawn from Π^t independently of other elements with probability distribution assigning each solution x_i^t a probability proportional to $f(x_i^t)$, and often instead of the original objective function a composition of f with some monotonic function is helpful.

The reproduction operator consists here of *crossover* and *mutation* stages. In crossover stage random substrings are exchanged between sequential pairs of "parent" genotypes from Π' with constant *crossover probability* p_c and an intermediate population $(z_1, ..., z_N)$ is formed so that $P\{z_{2k-1} = x'_{2k-1}, z_{2k} = x'_{2k}\} = 1 - p_c$, and

$$P\{z_{2k-1} = (x'_{2k-1,1}, ..., x'_{2k-1,j}, x'_{2k,j+1}, ..., x'_{2k,n}),$$

$$z_{2k} = (x'_{2k,1}, ..., x'_{2k,j}, x'_{2k-1,j+1}, ..., x'_{2k-1,n})\} = p_c/(n-1)$$

for all $k = 1, ..., N/2, j = 1, ..., n-1$. After that in mutation stage each bit in $z_1, ..., z_N$ is flipped with a constant *mutation probability* p_m, which is usually chosen sufficiently small. Thus the random output of reproduction operator $\Pi'' = \rho(\Pi')$ is such that $P\{x''_{ij} = 1 - z_{ij}\} = p_m$, $P\{x''_{ij} = z_{ij}\} = 1 - p_m$, $i = 1, ..., N$, $j = 1, ..., n$. The survival operator is trivial: $\sigma(\Pi^t, \Pi'') = \Pi''$ and the initial population Π^0 is constructed randomly so that $P\{x_{ij}^0 = 1\} = P\{x_{ij}^0 = 0\} = 1/2$ and all these bits are chosen independently.

The following result due to A. Eiben, E. Aarts and K. van Hee (see e.g. [18]) gives a relatively mild sufficient condition for convergence of the EA to an optimum. Let random variable t^* denote the first population number such that $f_{t^*} = f^*$.

Proposition 1 *Let the operators of EA be such that* $P\{x \in \widehat{s}(\Pi)\} > 0$ *for any* $\Pi \in X^N, x \in \widehat{\Pi}$, *and* $P\{x \in \widehat{\sigma}(\Pi, \Pi'')\} > 0$ *for any* $\Pi \in X^N, \Pi'' \in X^{N''}, x \in \widehat{\Pi''}$. *Suppose also that for every* $x \in X$ *there exists a sequence* $y_1, y_2, ..., y_k$ *of solutions such that* $y_1 = x$, $f(y_k) = f^*$ *and for all* $i = 1, ..., k-1$ *if* $y_i \in \widehat{\Pi'}$ *then* $P\{y_{i+1} \in \widehat{\rho}(\Pi')\} > 0$.

Then for any initial population $P\{\exists t : t^* < t\} = 1$, *and if the survival operator is such that for any* Π, Π'' *holds* $\max\{f(x) : x \in \widehat{\sigma}(\Pi, \Pi'')\} = \max\{f(x) : x \in \widehat{\Pi} \cup \widehat{\Pi''}\}$ *then* $f_t \xrightarrow{t\to\infty} f^*$ *with probability 1.*

Application of Prop.1 to the simple GA shows that there the optimal solution is found almost surely in finite number of iterations but to guarantee that $P\{f_t \xrightarrow{t\to\infty} f^*\} = 1$ the survival operator needs to be changed (keeping the best solution from Π^t would suffice).

The properties ensured by Prop.1 are definitely desirable for the EAs; however, it says nothing about the speed of convergence to the optimum, which is crucial since the trivial complete enumeration method is also known to find the optimum in a finite number of iterations. Therefore it is important to consider the expected hitting time of an optimum $E[t^*]$ and the closely related probability $P\{t^* \le t\}$ for finite values of t (see e.g. [18]).

Unfortunately, for the class of NP-hard problems, EAs with polynomally bounded $E[t^*]$ are unlikely to exist: for the EAs and other randomised algorithms the following "folklore" result (see e.g. [5]) can be derived from the definitions available in [3,13].

Proposition 2 *Let P_{opt} be an NP-hard NP optimisation problem and the approximation of P_{opt} within a factor r is NP-hard. If $NP \not\subseteq BPP$, then for any polynomial $h(\cdot)$ no randomised algorithm gives r-approximate solutions to all instances $\mathbf{x}$ of P_{opt} in polynomial time with probability greater than $1/h(|\mathbf{x}|)$, where $|\mathbf{x}|$ is the length of input $\mathbf{x}$.*

Note that the relation between classes NP and BPP is one of the open questions in complexity theory. The conjecture $NP \not\subseteq BPP$ is widely believed to be true, and it is also equivalent to conjecture $NP \neq RP$ [12].

Evolutionary algorithms for specific problems. In view of Prop.2 it is natural that the polynomial upper bounds on the expected optimisation time are only known for the EAs applied to polynomially solvable problem classes, e.g. some cases of the pseudo-Boolean functions [10,18,24], some families of the set cover [6] and the knapsack [10] problems etc. Several classes of these problems with exponential upper and lower bounds on $E[t^*]$ may be found in [10,18,24] as well. An instructive example of problems family where the usage of crossover in GA provably decreases $E[t^*]$ from exponential to polynomial is suggested by T.Jansen and I.Wegener (see e.g. [24]).

A number of randomised algorithms successfully used in complexity theory may also be considered as very simple instances of the EAs with multiple restarts, e.g. the algorithm of C. Papadimitriou with quadratic expected running time for the 2-SAT problem (see [13]) and the algorithm of U. Schöning for the k-SAT problem [19] (the later algorithm has an exponential upper bound on expected complexity). A new interesting approach to the analysis of EAs is based on rapidly mixing Markov chains associated with some EAs: in case in stationary distribution the probability that Π^t contains an optimum is non-negligible, the fast convergence of such chains allows to find an optimum efficiently and with high probability [21].

Another promising approach is aimed at the analysis of average-case complexity of the EAs given a probability distribution on the set of problem inputs. An example of such analysis is the minimum graph bisection problem with random graphs drawn from the so-called "planted bisection" model for which it has been theoretically shown [7] that there is little difference between the simple evolutionary algorithm (1+1)-ES and the more sophisticated EAs such as the Metropolis algorithm, and a problem-specific heuristic [4].

Often the EAs are used in situations where the information on quantitative parameters of the input problem (except for its dimensionality) is accessible only through evaluations of objective function in the sample points. This raises the issue of evaluation and comparison of different algorithms in such a "black box optimisation" scenario, which is currently being intensively studied (see e.g. [20,24]).

Dynamical system model and the local optima structure. Several workers (see e.g. [1]) have pointed out that in the GA on Boolean solutions space $X = \{0,1\}^n$ mutation is strongly connected to the local optima structure defined by Hamming distance: the effect of mutation at typical rates is that the Hamming distance between "offspring" and "parents" is usually 1; in some cases a distance of 2 will occur fairly frequently, but normally the probability of a greater distance is very small. However, the effect of crossover is markedly different: the offspring generated by crossover may be a long way from their parents in terms of Hamming distance. Nevertheless, while there are exceptions, functions that are easy for local search with a Hamming distance neighbourhood are generally fairly easy for GAs with crossover, while the converse often also seems to be true. The question remains as to why this is the case.

The last few years have seen significant progress in developing and understanding a formal model of simple GA and its generalisations as a dynamical system [22] (for brevity the simple GA is called GA in what follows). This work is highly important in its own right, but it turns out that the behaviour of the dynamical systems model is closely related to the local optima structure of the problem for binary search spaces.

In this model, a *population vector* p is introduced which has length $|X|$ and is one whose k-th component is the proportion of the population Π consisting of the solution $y_k \in X, k = 0, ..., |X|-1$ (numbered in some standard order). Given the current population Π^t with population vector p^t a function $G(p^t)$ produces a vector in $R^{|X|}$ whose k-th component is the probability that the solution y_k will be found in population Π^{t+1}.

The result of M.Vose [22] shows that as $N \to \infty$, with probability converging to 1, the sequence of population vectors $p^0, p^1, ..., p^t$ converges to the sequence $p^0, G(p^0), ..., G^t(p^0)$ for any finite t. In this connection it is helpful to consider the infinite-population GA because on each iteration of such a GA the vector p^{t+1} is a deterministic function $G(p^t)$ of the current population vector.

It turns out that the fixed points of G, i.e. such population vectors p that $p = G(p)$ are crucial in analysis of GA. It should be remarked that for a crossover-only GA (i.e. assuming $p_m = 0$), potentially every solution corresponds to a fixed point: just start the process in a uniform population with multiple copies of the same solution and it will stay there. However, not all fixed points are *stable*, in the sense that a small perturbation will start the population on a trajectory that will ultimately lead away from it.

The stability status of a particular fixed point p for the infinite-population GA can be determined by looking at the eigenvalues associated with the differential of G at that point (see e.g. [23]). If all these eigenvalues are in the interior of unit disk, p is called stable. Such stable points are *attractors* in the sense that from almost all initial population vectors p^0 the infinite-population GA converges to one of them [22].

In [23] M.Vose and A.Wright conjecture that in the case of crossover-only infinite-population GAs all fixed points are uniform population vectors, i.e. a single component has the value 1 and all others are zero. The attractors are really populations, but if the Vose-Wright conjecture holds, they are uniform populations, so we can identify them with individual solutions. This amplifies the importance of the following result linking the GA attractors and the local optima structure where the local optimum is a solution that does not have a neighbour with greater objective function value in Hamming distance 1.

Proposition 3 [23] *For a crossover-only infinite-population GA, each uniform attractor consists of multiple copies of some local optimum in a Hamming distance 1 neighbourhood.*

In other words, the number of (Hamming distance 1) local optima is an upper bound on the number of attractors. In principle, in a GA with mutation the attractors will not coincide exactly with those in the crossover-only GA, but the argument given in [23] shows that these fixed points are fundamental to GA performance even when mutation is included.

While Prop.3 gives some comfort to arguments of the type "we should use GAs because they avoid local optima", the comfort is small. In [16] C.Reeves has shown that it is rather "difficult" for a Hamming local optimum not to be a GA attractor too. Experimental analysis has also revealed that there is a strong identity between attractors and local optima, but other questions remain unanswered: for example, is there any relationship between properties of a basin of attraction of an attractor in the dynamical systems model and those of a basin in the Hamming neighbourhood structure? Do the eigenvalues of an attractor convey any information about basin size?

Discussion. The present state-of-the-art in analysis of the EAs is characterised by a large gap between the modest set of provable results and the rich encouraging experimental data. Most of the theoretical findings are limited in applicability either due to their abstractness or due to the narrow specification to a particular family of problems. This situation is gradually improving yet there is a need for intermediate kind of studies rigorously explaining the reasons of success or failure of a particular type of the EAs on a particular set of non-trivial problems and justifying the usage of certain procedures for it. Such analysis might involve further elaboration of theoretical bounding techniques like those in [2,5,7,10,21,24] as well as development of methodology for measurement of problems difficulty for EAs in terms of quickly computable characteristics (see e.g. [8,9]).

Acknowledgment This research was supported in part by a grant from the Royal Society.

REFERENCES

1. L.Altenberg (1997) *Fitness distance correlation analysis: an instructive counter-example.* In Proc. of 7-th International Conference on Genetic Algorithms, Morgan Kaufmann, San Francisco, CA, 57-64.
2. D.Aldous, U.U.Vazirani (1994) *"Go with the Winners" Algorithms.* In Proc. of FOCS 1994, 492-501.
3. G.Ausiello, M.Protasi (1995) *Local Search, Reducibility and Approximability of NP-Optimization Problems.* Information Processing Letters, **54**, 73-79.
4. R.B.Boppana (1987) *Eigenvalues and Graph Bisection: an Average Case Analysis.* In Proc. of the 28th IEEE Symposium on Foundations of Computer Science, 280-285.
5. P.A.Borisovsky, A.V.Eremeev (2001) *On Performance Estimates for Two Evolutionary Algorithms.* In Proc. of EvoWorkshops 2001. LNCS **2037**, 161-171.
6. P.A.Borisovsky, A.V.Eremeev (2002) *Ob Odnom Algoritme Sluchainogo Poiska.* This volume (in Russian).
7. T.Carson, R.Impagliazzo (2001) *Hill-Climbing Finds Random Planted Bisections.* In Proc. of SODA 2001, 903-909.
8. A.V.Eremeev, C.R.Reeves (2002) *Non-parametric Estimation of Properties of Combinatorial Landscapes.* In S.Cagnoni, J.Gottlieb, E.Hart, M.Middendorf and G.Raidl (Eds.) *Applications of Evolutionary Computing*, Springer-Verlag, Berlin, pp. 31-40.
9. J.Garnier, L.Kallel (2001) *How to Detect All Maxima of a Function ?* Proc. of the Second EVONET Summer School on Theoretical Aspects of Evolutionary Computing (Anvers, 1999), Springer, Berlin, 343-370.

10. J.He, X.Yao (2001) *Drift Analysis and Average Time Complexity of Evolutionary Algorithms.* Artificial Intelligence **127**, 57-85.
11. J.Holland (1975) *Adaptation in natural and artificial systems.* University of Michigan Press, Ann Arbor, MI.
12. K.Ko (1982) *Some Observations on the Probabilistic Algorithms and NP-Hard Problems.* Information Processing Letters, **14**, 39-43.
13. R.Motwani, P.Raghavan (1995) *Randomized Algorithms.* Cambridge University Press.
14. I.Rechenberg (1994) *Evolutionsstrategie'94.* Formann-Holzboog Verlag, Stuttgart.
15. C.R.Reeves (1997) *Genetic Algorithms for the Operations Researcher.* INFORMS Journal on Computing. **9** (3), 231-250.
16. C.R.Reeves (2002) *The Crossover Landscape and the Hamming Landscape for Binary Search Spaces.* Manuscript submitted for publication.
17. C.R.Reeves and J.E.Rowe (2002) *Genetic Algorithms: Principles and Perspectives*, Kluwer, Norwell, MA.
18. G.Rudolph (1998) *Finite Markov Chain Results in Evolutionary Computation A Tour d'Horizon.* Fundamenta Informaticae. **35** (1-4), 67-89.
19. U.Schöning (1999) *A Probabilistic Algorithm for k-SAT and Constraint Satisfaction Problems.* In Proc. of 40th IEEE Symposium on Foundations of Computer Science. 410-414.
20. C.Schumacher (2000) *Black Box Search - Framework and Methods.* PhD. thesis, The University of Tennessee, Knoxville.
21. P.Vitanyi (1996) *A Discipline of Evolutionary Progrmming.* In Proc. of 7th International Workshop on Algorithmic Learning Theory, Lecture Notes in Artificial Intelligence, **1160**, 67-82.
22. M.D.Vose (1999) *The Simple Genetic Algorithm: Foundations and Theory*, MIT Press, Cambridge, MA.
23. M.D.Vose, A.H.Wright (1995). *Stability of vertex fixed points and applications.* In D.Whitley and M.Vose (Eds.) Foundations of Genetic Algorithms 3, Morgan Kaufmann, San Mateo, CA, 103-114.
24. I.Wegener (2001) *Theoretical Aspects of Evolutionary Algorithms.* In Proc. of ICALP 2001, LNCS **2076**, 64-78.