What's Hot in Evolutionary Computation

Tobias Friedrich,[1,2] **Frank Neumann**[2]

[1]Hasso Plattner Institute, Potsdam, Germany

[2]Optimisation and Logistics, The University of Adelaide, Adelaide, Australia

Abstract

We provide a brief overview on some hot topics in the area of evolutionary computation. Our main focus is on recent developments in the areas of combinatorial optimization and real-world applications. Furthermore, we highlight recent progress on the theoretical understanding of evolutionary computing methods.

Introduction

There are a large variety of high performing optimization methods available for convex optimization problems. However, many problems arising in important application domains such as supply chain management and planning pose non-convex problems. In particular, these problems are often noisy, stochastic and/or have dynamically changing constraints. This is where evolutionary computation methods are often applied as they have been shown to be highly successful when dealing with such difficult problem domains. We will highlight some important recent real-world application techniques of evolutionary computing, discuss recent developments in the area of combinatorial optimization, and finally report on recent theoretical studies on the foundations of evolutionary computing.

Real-World Applications

In the area of real-world applications, evolutionary algorithms have been used for a wide range of *planning tasks*. In particular, Unmanned Aircraft Vehicles (UAVs) have gained increasing attention. The evolutionary planner presented in (Perez-Carabaza et al. 2016) optimizes the target detection time taking into account the uncertainty of the target location and the UAV motion and sensory model. To do this, the approach uses a Bayesian approach to handle uncertainty and a kinematic model taking into account environmental effects. The planning process is carried out by splitting the planning task into subsequences and optimizing each subsequence by an evolutionary algorithm based on the ending state of the previous subsequence. Furthermore, evolutionary algorithms have been considered in (Ellefsen, Lepikson, and Albiez 2016) for 3D path planning. The authors investigate the problem of inspecting three dimensional structures

and use an evolutionary multi-objective algorithm to optimize the coverage degree and the energy usage at the same time. The energy usage is relatively simple to compute and the estimated coverage of a plan is the computationally expensive part as it has to take geometry of the 3D object (in this case a 3D Warehouse) into account.

Finally, the area of *bilevel optimization* has received a lot of attention in recent years. In (Lu, Deb, and Sinha 2016), a new approach for dealing with bilevel problems having uncertainties is presented. The authors study the interactions of uncertainties in the upper and lower level variables and show that their new approach performs well on a wide range of benchmark test case as well as a real-world Navy ship design problem.

Combinatorial Optimization

In the area of combinatorial optimization *multi-component problems* are a hot topic. Multi-component problems arise frequently in real-world applications such as supply chain management and the goal is to study the interactions arising from the combination of different combinatorial problems in a systematic way. A prominent problem which has gained significant attention during the last three years is the traveling thief problem (TTP). TTP combines the classical traveling salesperson problem (TSP) and the standard knapsack problem (KP) in a non-linear way. Given that both TSP and KP have been widely studied in the literature, this knowledge sets the basis for dealing with the interactions arising in TTP. Different heuristic approaches have been proposed for variants of TTP (Yafrani and Ahiod 2016; Chand and Wagner 2016; Mei, Li, and Yao 2014) and a large benchmark set to compare these algorithms has been established in (Polyakovskiy et al. 2014; Przybylek, Wierzbicki, and Michalewicz 2016). Recent investigations have considered the impact of the renting rate which connects the TSP and KP part of the problem (Wu, Polyakovskiy, and Neumann 2016).

Furthermore, the *features* and their impact on the performance of different evolutionary computation methods are an research area of high interest. Features occur in different areas. On the one hand, fitness landscape analysis has a long tradition in the area of evolutionary computation. Here one studies the landscape defined by search points and "natural" neighbourhoods of an underlying combinatorial opti-

mization problem in order to understand the performance of popular algorithms. Recent investigations have shown that fitness landscapes for combinatorial optimization problems often consists of several funnels in the landscape that makes search difficult. Clustering approaches for community detection have been investigated to detect funnels for NK landscapes (Herrmann, Ochoa, and Rothlauf 2016) and it has been shown that the number of clusters and the size of the cluster containing the optimal solution are an important factor for determining search difficulty of a landscape. For the traveling salesperson problem and the Chained Lin-Kernighan heuristic, a data driven approach to examine funnels has been presented (Ochoa and Veerapen 2016) which examines the landscape of various TSP instances.

Theoretical Foundations

A major challenge to the theoretical analysis of evolutionary computing is developing a rigorous understanding of how they behave on important optimization problems. The role of *crossover* in evolutionary computation is still a major open problem in the theory of evolutionary algorithms. In some cases, it can be provably helpful for optimization obtaining quantifiable speed-ups on artificial test functions like JUMP and ONEMAX, and particular combinatorial optimization problems on graphs (Doerr, Happ, and Klein 2012; Doerr et al. 2013; Lehre and Yao 2011; Sudholt 2012; Dang et al. 2016a; 2016b). It has further been shown that increasing the effective recombination rate promotes the evolution of robustness (Friedrich, Kötzing, and Sutton 2016).

A major aspects in theoretical research of evolutionary computing has been the presence of *uncertainty*. Uncertain and dynamic problems are pervasive in practice, and practitioners often rely on heuristic techniques in these settings because classical tailored approaches often cannot cope with uncertain environments such as noisy objective functions and dynamically changing problems (Bianchi et al. 2009; Jin and Branke 2005).

It is therefore very important to understand the effect that different properties of uncertainty have on algorithm behavior. In *stochastic* optimization, the fitness of a candidate solution does not have a deterministic value, but instead follows some given (but fixed) noise distribution. Most research on stochastic optimization is concerned with the magnitude of noise (usually measured by the variance). There are a number of recent papers on the theoretical analysis of evolutionary computing in stochastic environments. For evolutionary algorithms, (Gießen and Kötzing 2014) analyzed the $(\mu + \lambda)$-EA on noisy ONEMAX and LeadingOnes and found that populations make the EA robust to specific distributions of prior and posterior noise, while (Dang and Lehre 2014) consider non-elitist EAs and give run time bounds in settings of partial information. Posterior noise from a Gaussian was considered in (Friedrich et al. 2015; Prügel-Bennett, Rowe, and Shapiro 2015) for various algorithms. Also the effect of different kinds of distributions has been studied (Friedrich et al. 2016).

In *dynamic* optimization, the problem at hand undergoes dynamic changes with respect to the objective function and/or the given constraints of the problem. Evolution-ary algorithms are able to adapt to dynamic changes and recompute new good solutions without having to start from scratch. Understanding this from a theoretical perspective and analyzing the time required for the recomputation after a dynamic change has happened is another important topic. Theoretical studies for dynamically changing problems have been carried out for classical example functions such as ONEMAX (Kötzing, Lissovoi, and Witt 2015) and the NP-hard makespan scheduling problem (Neumann and Witt 2015).

Another recently very popular concept are *self-adjusting* algorithms in discrete spaces. In continuous optimization it is obvious that static parameter choices are not very meaningful. This is why in continuous spaces several examples exist where adaptive parameter choices are required and well-studied (see e.g. (Auger and Hansen 2016; Jägersküpper 2008)). For the discrete domain, several empirical works exist that suggest an advantage of adaptive parameter updates (see e.g. (Eiben, Hinterding, and Michalewicz 1999; Karafotias, Hoogendoorn, and Eiben 2015) and (Eiben and Smith 2003, Chapter 8)). The first work formally showing an asymptotic gain over static parameter selection is the self-adjusting choice of the population size of the genetic algorithm proposed and analyzed in (Doerr and Doerr 2015). Furthermore, there are theoretical investigations of adaptive parameter choices in discrete optimization that show advantages of a fitness-dependent mutation rate (Böttcher, Doerr, and Neumann 2010; Doerr, Doerr, and Yang 2016b), a fitness-dependent population size (Doerr, Doerr, and Ebel 2015), a self-adjusting mutation rate (Doerr, Doerr, and Yang 2016a; Dang and Lehre 2016), a self-adjusting population size (Doerr and Doerr 2015), and a self-adjusting choice of the number of parallel evaluations in a parallel EA (Lässig and Sudholt 2011).

Acknowledgements

References

Auger, A., and Hansen, N. 2016. Linear convergence of comparison-based step-size adaptive randomized search via stability of markov chains. *SIAM Journal on Optimization* 26(3):1589–1624.

Bianchi, L.; Dorigo, M.; Gambardella, L.; and Gutjahr, W. 2009. A Survey on Metaheuristics for Stochastic Combinatorial optimization. *Natural Computing* 8:239–287.

Böttcher, S.; Doerr, B.; and Neumann, F. 2010. Optimal fixed and adaptive mutation rates for the LeadingOnes problem. In *Proc. of PPSN'10*, 1–10. Springer.

Chand, S., and Wagner, M. 2016. Fast heuristics for the multiple traveling thieves problem. In *Proc. of GECCO'16*, 293–300. ACM.

Dang, D.-C., and Lehre, P. K. 2014. Evolution under partial information. In *Proc. of GECCO '14*, 1359–1366. ACM.

Dang, D.-C., and Lehre, P. K. 2016. Self-adaptation of mutation rates in non-elitist populations. In *Proc. of PPSN'16*, 803–813. Springer.

Dang, D.-C.; Friedrich, T.; Krejca, M. S.; Kötzing, T.; Lehre, P. K.; Oliveto, P. S.; Sudholt, D.; and Sutton, A. M. 2016a. Escaping local optima with diversity mechanisms and crossover. In *Proc. GECCO'16*, 645–652. ACM.

Dang, D.-C.; Lehre, P. K.; Friedrich, T.; Kötzing, T.; Krejca, M. S.; Oliveto, P. S.; Sudholt, D.; and Sutton, A. M. 2016b. Emergence of diversity and its benefits for crossover in genetic algorithms. In *Proc. PPSN'16*, 890–900. Springer.

Doerr, B., and Doerr, C. 2015. Optimal parameter choices through self-adjustment: Applying the 1/5-th rule in discrete settings. In *Proc. of GECCO'15*, 1335–1342. ACM.

Doerr, B.; Johannsen, D.; Kötzing, T.; Neumann, F.; and Theile, M. 2013. More effective crossover operators for the all-pairs shortest path problem. *Theor. Comput. Sci.* 471:12–26.

Doerr, B.; Doerr, C.; and Ebel, F. 2015. From black-box complexity to designing new genetic algorithms. *Theor. Comput. Sci.* 567:87–104.

Doerr, B.; Doerr, C.; and Yang, J. 2016a. k-bit mutation with self-adjusting k outperforms standard bit mutation. In *Proc. of PPSN'16*, 824–834. Springer.

Doerr, B.; Doerr, C.; and Yang, J. 2016b. Optimal parameter choices via precise black-box analysis. In *Proc. of GECCO'16*, 1123–1130. ACM.

Doerr, B.; Happ, E.; and Klein, C. 2012. Crossover can provably be useful in evolutionary computation. *Theor. Comput. Sci.* 425:17–33.

Eiben, A. E., and Smith, J. E. 2003. *Introduction to Evolutionary Computing*. Springer.

Eiben, A. E.; Hinterding, R.; and Michalewicz, Z. 1999. Parameter control in evolutionary algorithms. *IEEE Trans. on Evol. Comp.* 3:124–141.

Ellefsen, K. O.; Lepikson, H. A.; and Albiez, J. C. 2016. Planning inspection paths through evolutionary multi-objective optimization. In *Proc. of GECCO'16*, 893–900. ACM.

Friedrich, T.; Kötzing, T.; Krejca, M. S.; and Sutton, A. M. 2015. The benefit of recombination in noisy evolutionary search. In *Proc. of ISAAC'15*, 140–150. Springer.

Friedrich, T.; Kötzing, T.; Krejca, M. S.; and Sutton, A. M. 2016. Graceful scaling on uniform versus steep-tailed noise. In *Proc. of PPSN'16*, 761–770. Springer.

Friedrich, T.; Kötzing, T.; and Sutton, A. M. 2016. On the robustness of evolving populations. In *Proc. of PPSN'16*, 771–781. Springer.

Gießen, C., and Kötzing, T. 2014. Robustness of populations in stochastic environments. In *Proc. of GECCO '14*, 1383–1390. ACM.

Herrmann, S.; Ochoa, G.; and Rothlauf, F. 2016. Communities of local optima as funnels in fitness landscapes. In *Proc. of GECCO'16*, 325–331. ACM.

Jägersküpper, J. 2008. Oblivious randomized direct search for real-parameter optimization. In *Proc. of ESA'08*, 553–564. Springer.

Jin, Y., and Branke, J. 2005. Evolutionary Optimization in Uncertain Environments—A Survey. *IEEE Trans. on Evol. Comp.* 9:303–317.

Karafotias, G.; Hoogendoorn, M.; and Eiben, A. 2015. Parameter control in evolutionary algorithms: Trends and challenges. *IEEE Trans. on Evol. Comp.* 19:167–187.

Kötzing, T.; Lissovoi, A.; and Witt, C. 2015. (1+1) EA on generalized dynamic OneMax. In *Proc. of FOGA'15*, 40–51. ACM.

Lässig, J., and Sudholt, D. 2011. Adaptive population models for offspring populations and parallel evolutionary algorithms. In *Proc. of FOGA'11*, 181–192. ACM.

Lehre, P. K., and Yao, X. 2011. Crossover can be constructive when computing unique input–output sequences. *Soft Computing* 15(9):1675–1687.

Lu, Z.; Deb, K.; and Sinha, A. 2016. Finding reliable solutions in bilevel optimization problems under uncertainties. In *Proc. of GECCO'16*, 941–948. ACM.

Mei, Y.; Li, X.; and Yao, X. 2014. On investigation of interdependence between sub-problems of the TTP. *Soft Computing* 1–16.

Neumann, F., and Witt, C. 2015. On the runtime of randomized local search and simple evolutionary algorithms for dynamic makespan scheduling. In *Proc. of IJCAI'15*, 3742–3748. AAAI Press.

Ochoa, G., and Veerapen, N. 2016. Additional dimensions to the study of funnels in combinatorial landscapes. In *Proc. of GECCO'16*, 373–380. ACM.

Perez-Carabaza, S.; Besada-Portas, E.; Orozco, J. A. L.; and de la Cruz, J. M. 2016. A real world multi-uav evolutionary planner for minimum time target detection. In *Proc. of GECCO'16*, 981–988. ACM.

Polyakovskiy, S.; Bonyadi, M. R.; Wagner, M.; Michalewicz, Z.; and Neumann, F. 2014. A comprehensive benchmark set and heuristics for the traveling thief problem. In *Proc. of GECCO'14*, 477–484. ACM.

Prügel-Bennett, A.; Rowe, J. E.; and Shapiro, J. 2015. Runtime analysis of population-based evolutionary algorithm in noisy environments. In *Proc. of FOGA'15*, 69–75.

Przybylek, M. R.; Wierzbicki, A.; and Michalewicz, Z. 2016. Multi-hard problems in uncertain environment. In *Proc. of GECCO'16*, 381–388. ACM.

Sudholt, D. 2012. Crossover speeds up building-block assembly. In *Proc. of GECCO'12*, 689–696. ACM.

Wu, J.; Polyakovskiy, S.; and Neumann, F. 2016. On the impact of the renting rate for the unconstrained nonlinear knapsack problem. In *Proc. of GECCO'16*, 413–419. ACM.

Yafrani, M. E., and Ahiod, B. 2016. Population-based vs. single-solution heuristics for the travelling thief problem. In *Proc. of GECCO'16*, 317–324. ACM.