

An Active-Set Evolution Strategy for Optimization with Known Constraints

Dirk V. Arnold^(✉)

Faculty of Computer Science, Dalhousie University,
Halifax, Nova Scotia B3H 4R2, Canada
dirk@cs.dal.ca

Abstract. We propose an evolutionary approach to constrained optimization where the objective function is considered a black box, but the constraint functions are assumed to be known. The approach can be considered a stochastic active-set method. It labels constraints as either active or inactive and projects candidate solutions onto the subspace of the feasible region that is implied by rendering active inequality constraints equalities. We implement the approach in a $(1 + 1)$ -ES and evaluate its performance using a commonly used set of test problems.

1 Introduction

Evolutionary algorithms are stochastic black box optimization strategies. They are commonly used in connection with optimization problems that do not admit a convenient mathematical representation of the objective, or if gradient estimates can be obtained only at a high cost or are necessarily inaccurate. Examples include scenarios where the evaluation of the quality of a candidate solution requires running a simulation model. In the context of constrained optimization with evolutionary algorithms, the constraint functions are often considered as black boxes as well. However, in many cases, including the case of bound constraints, it is not uncommon that the constraint functions are known and relatively inexpensive to evaluate. The objective of this paper is to develop a simple evolutionary algorithm for constrained optimization with known constraints.

Active-set methods are a common approach to solving constrained optimization problems [12]. They maintain a set of active inequality constraints and perform optimization in the subspace of the feasible region that is implied by rendering the active inequality constraints equalities. The algorithm we introduce in this paper can be considered a stochastic active-set approach implemented in a $(1 + 1)$ -ES¹. The step size of the $(1 + 1)$ -ES is commonly controlled using the 1/5th rule [13]. That rule can fail in the presence of small constraint angles (i.e., small angles between the gradient of the objective function and the normal vector of the constraint function) in cases as simple as a linear objective with a single linear constraint [2, 15]. Small constraint angles result in low success rates

¹ See Hansen et al. [7] for an introduction to evolution strategy terminology.

and thus a systematic reduction of the step size. Projection of infeasible candidate solutions onto the feasible region in connection with an active-set approach can potentially circumvent unwarranted decreases of the step size.

The remainder of this paper is organized as follows. In Sect. 2 we outline the class of optimization problems we strive to solve, formalize notation, and propose an active-set (1 + 1)-ES for optimization with known constraints. In Sect. 3, that algorithm is applied to commonly used test problems and its performance is discussed. Section 4 concludes.

2 Problem and Algorithm

We consider the problem of minimizing objective function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ subject to constraints

$$\begin{aligned} g_i(\mathbf{x}) &\leq 0 && \text{for } i \in \{1, \dots, l\} \\ h_j(\mathbf{x}) &= 0 && \text{for } j \in \{1, \dots, m\}. \end{aligned} \tag{1}$$

The active set $\mathcal{A}(\mathbf{x})$ of a (feasible) candidate solution \mathbf{x} is the set of indices of all those inequality constraints with $g_i(\mathbf{x}) = 0$. Assuming a single globally optimal solution \mathbf{x}^* to the optimization problem, we refer to the active set $\mathcal{A}(\mathbf{x}^*)$ of that solution as the optimal active set \mathcal{A}^* . We refer to the subspace of the search space where all equality constraints and active inequality constraints in $\mathcal{A}(\mathbf{x})$ are satisfied as equalities as the reduced search space at \mathbf{x} . We write n^* for the dimension of the reduced search space at the optimal solution \mathbf{x}^* .

Our active-set (1 + 1)-ES evolves a feasible candidate solution $\mathbf{x} \in \mathbb{R}^n$ to the optimization problem at hand, adapting the step size $\sigma \in \mathbb{R}_+$ using the 1/5th rule. Offspring candidate solutions are usually projected onto the reduced search space at the parent. However, with a certain probability the use of the active set is suspended, allowing to break out of the reduced search space. Adapting the step size of the algorithm only in those steps where the active inequalities are enforced as equalities prevents unwarranted decreases of the step size. A single iteration of the algorithm is described in Fig. 1.

Boolean flag κ determines whether or not the active set of \mathbf{x} is enforced for offspring candidate solution \mathbf{y} . If κ is false, then the search proceeds in the reduced search space. If it is true, then the inequality constraints active at the parental candidate solution will be enforced as inequalities rather than as equalities. If the dimension of the reduced search space at \mathbf{x} is zero, then there is no use in enforcing the active constraints as they would repeatedly yield the same solution, and κ is thus set to true. Otherwise, the active inequality constraints are enforced as equality constraints with probability $1 - p$. Larger values of p decrease the likelihood that the algorithm will spend unproductive time in non-optimal reduced subspaces. However, once the optimal active set \mathcal{A}^* has been found, smaller values of p are useful as unproductive steps beyond the optimal reduced search space are avoided. We use $p = 0.2$ throughout.

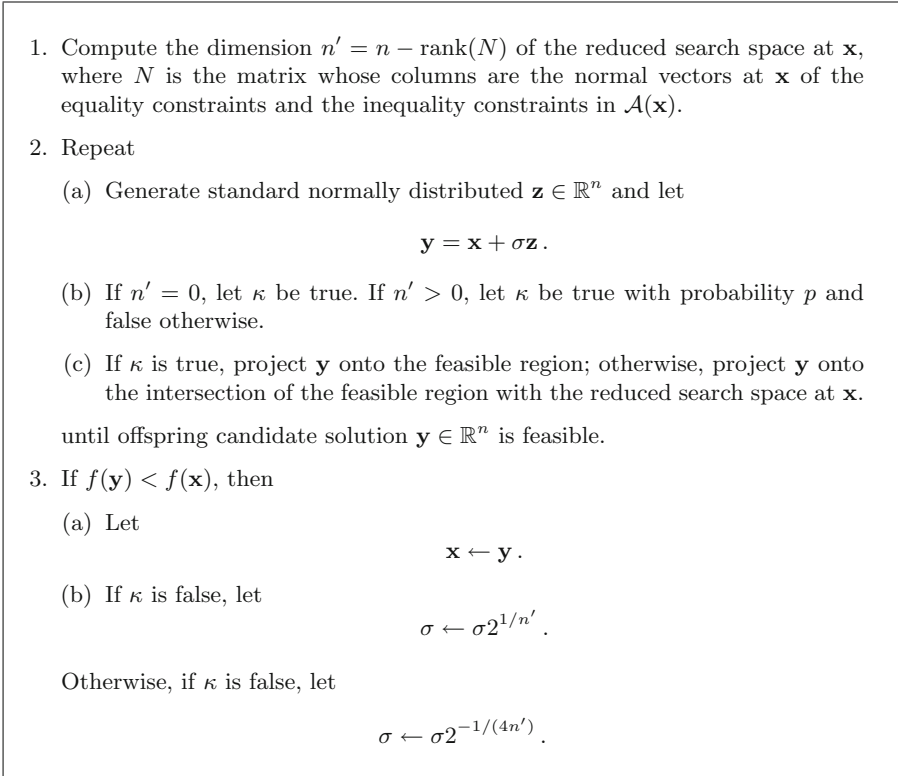


Fig. 1. Single iteration of the active-set (1 + 1)-ES.

Projection of \mathbf{y} onto the intersection of the feasible region with the reduced search space at \mathbf{x} is accomplished by minimizing function $d(\mathbf{w}) = \|\mathbf{w} - \mathbf{y}\|^2$ subject to constraints

$$\begin{aligned} g_i(\mathbf{w}) &\leq 0 && \text{for } i \in \{1, \dots, l\} \setminus \mathcal{A}(\mathbf{x}) \\ g_i(\mathbf{w}) &= 0 && \text{for } i \in \mathcal{A}(\mathbf{x}) \\ h_j(\mathbf{w}) &= 0 && \text{for } j \in \{1, \dots, m\}. \end{aligned}$$

When the use of the active set is suspended, projection onto the feasible region is accomplished by minimizing that some function, but subject to the original set of constraints from Eq. (1). Notice that minimization does not make use of the objective f of the original optimization problem and that thus the algorithm performs only a single evaluation of f per iteration. Minimization of d can be accomplished using any algorithm for constrained optimization. We use the active-set method implemented in `fmincon` in *Matlab*'s optimization toolbox. Step 2 involves a loop as minimization of d may fail to yield a feasible solution.

The update of σ in Step 3 of the algorithm employs the implementation of the 1/5th rule due to Kern et al. [9]. The step size is updated only in those iterations

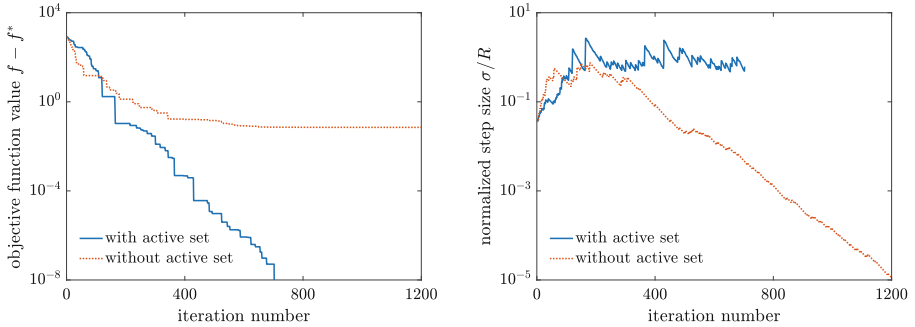


Fig. 2. Traces from runs of the active-set (1+1)-ES and a (1+1)-ES that projects infeasible candidate solutions onto the feasible region but does not enforce active inequalities as equalities applied to a 10-dimensional sphere with five mutually orthogonal linear inequality constraints active at the optimal solution. Shown are the evolution of the difference between $f(\mathbf{x})$ and the optimal objective function value f^* and the step size σ normalized by division by $R = \|\mathbf{x} - \mathbf{x}^*\|$.

where the search proceeds in the reduced search space, thus avoiding the issue of systematically decreasing σ when the use of the active set is suspended.

Throughout a run of the algorithm, we store the current active set. Constraints are added to the active set whenever a candidate solution is accepted for which those constraints are tight. Notice that it is straightforward from the output of `fmincon` to determine which constraints are tight by identifying those inequality constraints that have positive Lagrange multipliers. The active set is replaced in those iterations where a candidate solution replaces its parent that is generated with use of the active set suspended.

Figure 2 illustrates the advantage of the active-set based approach over a (1+1)-ES that simply projects infeasible candidate solutions onto the feasible region, without enforcing active inequality constraints as equalities, and that updates the step size in every iteration. We have conducted 21 runs of both strategies for objective function $f(\mathbf{x}) = \mathbf{x}^T \mathbf{x}$ with $n = 10$ and constraints $\mathbf{x}^T \mathbf{e}_i \geq 1$ for $i \in \{1, \dots, 5\}$, where \mathbf{e}_i is the unit vector in the direction of the i th coordinate axis. All runs are initialized to start at $\mathbf{x} = (9, \dots, 9)^T$ and with step size $\sigma = 1$. The active-set (1+1)-ES attained an objective function value within a factor of $(1 + 10^{-8})$ of the optimal objective function value $f^* = f(\mathbf{x}^*)$ in each of the runs; the run that required the median number of iterations is shown in the figure. It can be seen that the strategy converges linearly, and that after an initial increase, the step size σ is controlled to be approximately proportional to the distance R from the optimal solution \mathbf{x}^* . Not shown, in the run depicted, the five constraints become active between the 66th and the 100th iteration and remain active until the algorithm terminates. Without the use of the active set, none of the 21 runs obtained a solution with an objective function value within a factor of $(1 + 10^{-8})$ of f^* before terminating after 1,200 iterations. The corresponding trace shown in the figure is that of a random run. It can be

seen that not enforcing active inequalities as equalities and updating σ in every iteration eventually results in very small step sizes and slow progress.

3 Evaluation and Discussion

We evaluate the active-set (1 + 1)-ES by applying it to the test problems g01 through g11 gathered by Michalewicz and Schoenauer [11] and summarized by Liang et al. [10]. We initialize the parental candidate solution by uniformly randomly sampling a point in the bound constrained search space and then projecting it onto the feasible region using the same approach as described for offspring candidate solutions in Sect. 2. The initial step size is set to $\sigma = 0.2 \min\{u_i - l_i \mid i = 1, \dots, n\}$, where the l_i and u_i are the lower and upper bounds of the search space in dimension i for the respective problem. A run of the algorithm is terminated and considered a success if a candidate solution \mathbf{x} with an objective function value $f(\mathbf{x}) < (1 + \epsilon)f^*$ is found, where f^* is the objective function value of the optimal solution to the problem. It is considered unsuccessful if after 1,200 iterations (and thus as many evaluations of the objective function) no solution satisfying the termination criterion has been found. We refer to ϵ as the target accuracy.

Table 1. Test function properties and results.

	g01	g02	g03	g04	g05	g06	g07	g08	g09	g10	g11
Dimension n	13	20	10	5	4	2	10	2	7	8	2
Reduced dimension n^*	0	19	9	0	1	0	4	2	5	2	1
Success rate:											
$\epsilon = 10^{-4}$	1.0	0.0	1.0	1.0	1.0	1.0	0.98	0.52	1.0	1.0	1.0
$\epsilon = 10^{-8}$	1.0	0.0	1.0	1.0	1.0	1.0	0.99	0.53	1.0	1.0	1.0
Median number of function evaluations:											
$\epsilon = 10^{-4}$	49	—	458	22	36	3	411	123	302	117	27
$\epsilon = 10^{-8}$	46	—	865	24	84	4	684	229	616	243	106

We have conducted 100 runs of the algorithm for each test problem and target accuracies $\epsilon \in \{10^{-4}, 10^{-8}\}$ ². Table 1 summarizes the results. For eight of the eleven test problems, the globally optimal solution was found to the desired accuracy in all 100 runs conducted. The three exceptions are as follows:

² The `ConstraintTolerance` parameter of `fmincon` is set to its default value of 10^{-6} for the runs with target accuracy $\epsilon = 10^{-4}$. For target accuracy $\epsilon = 10^{-8}$ we used `ConstraintTolerance` 10^{-9} instead as some runs for problem g03 terminate unsuccessfully if the default accuracy is used.

- For **g02** not a single run successfully located the globally optimal solution. Problem **g02** has a very large number of local minima, and the search space is such that the likelihood of starting in the basin of attraction of the global optimum is near zero. A stochastic hill climber, such as the $(1 + 1)$ -ES, will almost always converge toward a merely local minimum.
- A small number of unsuccessful runs are observed for problem **g07**, even though the problem is unimodal. The search space is ten-dimensional, with six inequality constraints active at the optimal solution. In some runs, the upper bound constraint for variable x_8 is included in the active set of the algorithm at some point during the run. The likelihood of escaping the point that is optimal for this active set in one of the steps where the use of the active set is suspended can be observed to be no higher than 5% for a large range of values of the step size. It may thus take hundreds of iterations before the upper bound constraint is rendered inactive, and either reaching the iteration limit or the step size becoming so small that the limits of numerical accuracy are reached prevents successfully escaping the non-optimal reduced search space.
- Problem **g08** has four locally optimal solutions. No constraints are active at the globally optimal one of those. The $(1 + 1)$ -ES converges to one of the merely locally optimal solutions with a likelihood of just under one half. Conducting multiple runs of the algorithm would allow locating the globally optimal solution with high probability.

Figure 3 shows histograms of the number of objective function evaluations required to solve problems **g01** and **g03** through **g11** to both target accuracies. The ranges of the histograms are such that all successful runs are included. No data are shown for **g02** as no successful runs were observed for that problem. It can be seen that the histograms for **g01**, **g04**, and **g06** differ fundamentally from those for the other problems in that there is little difference between the data for $\epsilon = 10^{-4}$ and $\epsilon = 10^{-8}$. This is due to the dimension of the optimal reduced search space being zero. In that case, solving for the solution at the intersection of the active constraints yields the optimal solution (up to the limits of numerical accuracy). For the remaining problems, the gap between the histograms for $\epsilon = 10^{-4}$ and $\epsilon = 10^{-8}$ is due to the need for the $(1 + 1)$ -ES to search a non-zero reduced search space, with a larger discrepancy for those cases where the dimension of that space is large.

Figure 4 shows traces from runs requiring the median number of iterations to reach target accuracy $\epsilon = 10^{-8}$ for test problems **g04** and **g10**. The former is an example of a problem where the dimension of the optimal reduced search space is zero; for the latter, that dimension is $n^* = 2$. Plotted against the iteration number are the difference between the objective function value $f(\mathbf{x})$ and the optimal objective function value f^* as well as the step size σ of the algorithm. Those iterations where the algorithm suspends the use of the active set (i.e., κ is true) are marked with small circles.

In the run on **g04**, the evolution strategy generates the optimal active set \mathcal{A}^* in iteration 24, at which point it terminates. Between iterations 18 and 21, the use

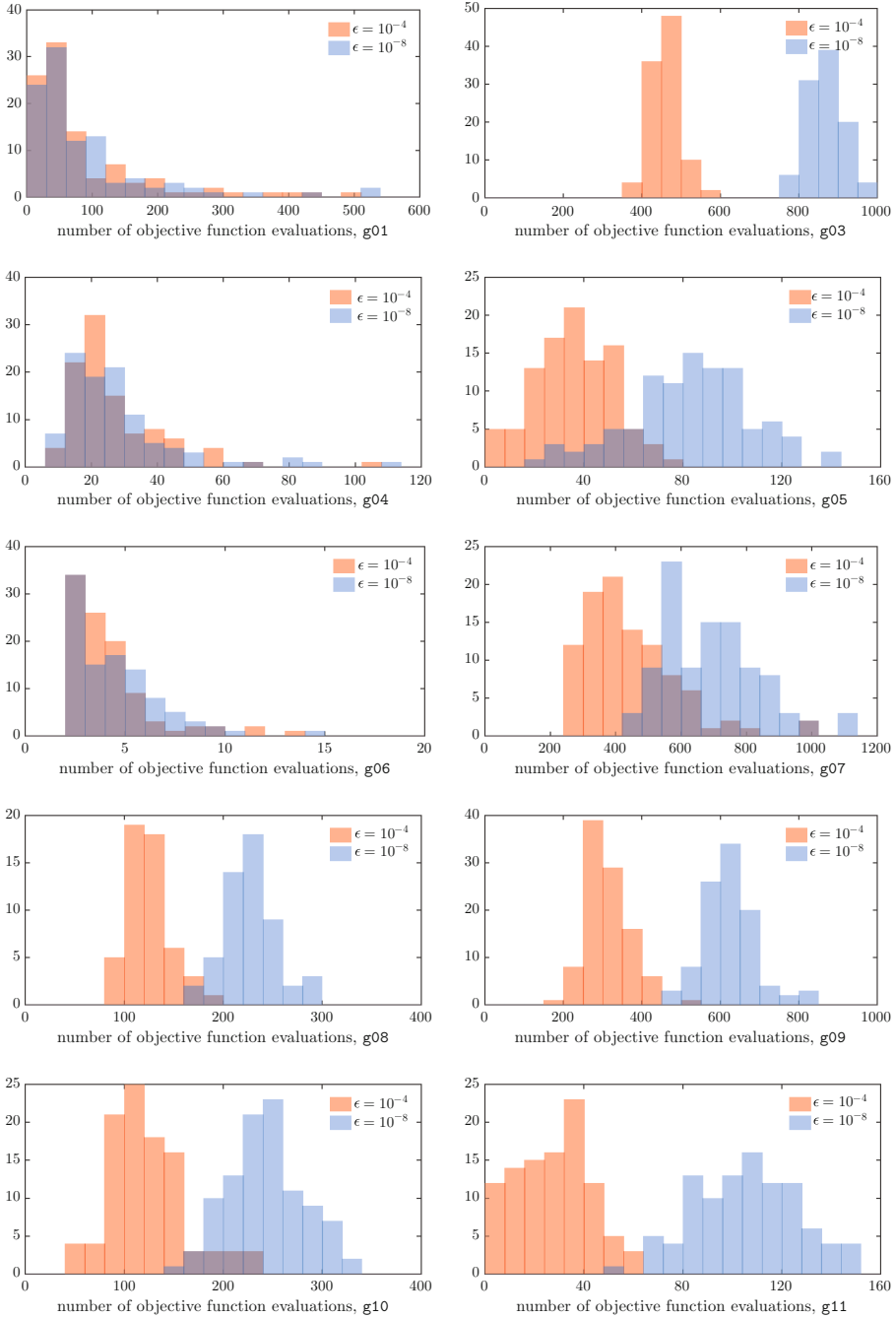


Fig. 3. Histograms showing the number of objective function evaluations required to solve problems g01 and g03 through g11.

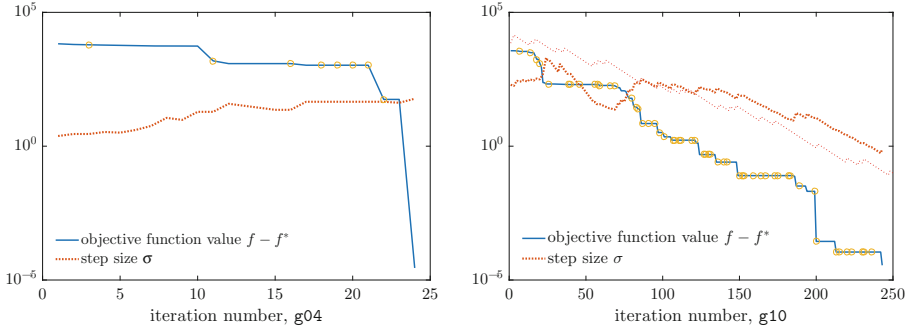


Fig. 4. Traces from runs requiring the median number of iterations to reach target accuracy $\epsilon = 10^{-8}$ for problems **g04** and **g10**. The evolution of both the difference between $f(\mathbf{x})$ and the optimal objective function value f^* and the step size σ are shown. The small circles mark those iterations where the use of the active set is suspended. The thin dotted line in the plot for **g10** shows the evolution of the step size in a typical run of the $(1 + 1)$ -ES on an unconstrained, two-dimensional sphere function.

of the active set is suspended in each step as it is such that the dimension of the reduced search space is zero. The step size σ largely increases through most of the run.

In the run on **g10**, the algorithm arrives at the optimal active set \mathcal{A}^* , which consists of the six constraints active at the global optimum of the problem, at iteration 70. The active set remains stable after this, and optimization effectively proceeds in a reduced subspace of dimension $n^* = 2$, with the exception of those steps where the use of the active set is suspended, but which have no further effect on the sequence of successful candidate solution generated. It can be seen that the step size σ largely decreases as the search in the two-dimensional subspace progresses. Comparison with the thin dotted curve, which shows the rate of decrease of the step size of a $(1 + 1)$ -ES on an unconstrained, two-dimensional sphere function, shows a similar rate of linear convergence.

A comparison of the performance of the active-set $(1 + 1)$ -ES with that of other approaches to constrained evolutionary optimization is not straightforward as both initialization conditions and termination criteria often differ for approaches found in the literature. More significantly, most other approaches consider the constraint functions as black boxes and are thus not easily able to project candidate solutions onto the feasible region (or subspaces thereof). That said, some useful points of comparison do exist:

- The numbers of objective function evaluations required by the active covariance matrix adaptation based approach by Arnold and Hansen [3] to solve four of the problems³ to target accuracy 10^{-8} range from 308 for **g06** to 3,976 for **g10**. The corresponding figures from Table 1 range from 4 for **g06** to 684

³ Multimodal problems and those with equality constraints were not considered in that paper, leaving only problems **g06**, **g07**, **g09**, and **g10**.

for **g07**. It is important to keep in mind though that the active-set approach introduced here assumes knowledge of the constraint functions, whereas the algorithm from [3] considers the constraint functions as black boxes.

- The algorithm by Takahama and Sakai [16], which performed best among all entries submitted to the *CEC 2006 Special Session on Constrained Real-Parameter Optimization*, assumes knowledge of gradient vectors and thus does not treat the constraint functions as black boxes. It requires median numbers of objective function evaluations ranging from 1,182 for **g08** to 105,799 for **g10**. The algorithm does successfully solve **g02**.
- Bagheri et al. [6] propose SACOBRA, a self-adaptive variant of the surrogate based COBRA algorithm by Regis [14], which solves most of the eleven problems (though not **g02**, and others only with limited accuracy) with fewer than 500 objective function evaluations. SACOBRA considers the constraint functions as black boxes and is thus applicable when the constraints are not known. It requires more function evaluations than the active-set (1 + 1)-ES for a number of those problems where the dimension of the optimal reduced search space is low. However, it appears to often converge faster where that dimension is not very small. This latter advantage is a consequence of the smooth nature of the test problems, which admit polynomial surrogate models that make it possible to converge superlinearly. As shown by Teytaud and Gelly [17], as a comparison-based algorithm that does not use objective function values other than in comparisons, the active-set (1 + 1)-ES cannot exhibit super-linear convergence.

4 Conclusions

To conclude, we have proposed an active-set (1+1)-ES for constrained numerical optimization with known constraints. The algorithm usually generates offspring candidate solutions constrained to the reduced subspace at their parents, but with a fixed probability samples offspring that do not necessarily fall into that space, thus allowing it to render active constraints inactive. Key to the functioning of the algorithm is to adapt the step size only in those iterations where the offspring are constrained to the reduced search space. The algorithm can be implemented in a few lines of *Matlab* code and performs very well when compared with related work.

It is of interest to apply the proposed active-set approach in evolutionary algorithms other than the (1 + 1)-ES. The $(\mu/\mu, \lambda)$ -ES is an evolutionary algorithm less sensitive to noise and ruggedness of the objective than the (1 + 1)-ES as it is capable of proceeding with larger steps. Restart variants of that algorithm [5] are commonly used for multimodal optimization problems and may exhibit improved performance for problems **g02** and **g08**. However, using the active-set approach with cumulative step size adaptation is less than straightforward, and it has been seen that care has to be taken when integrating constraint handling techniques with step size adaptation [1, 8]. Also of interest is the problem of employing the active-set approach in combination with other constraint

handling techniques, such as augmented Lagrangian methods [4], in case only a proper subset of the constraints is known explicitly. Finally, it is of interest to employ the active-set approach in evolutionary algorithms that use surrogate models of the objective function in order to reduce the time spent on optimization in reduced search spaces.

Acknowledgements. This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. Arnold, D.V.: Resampling versus repair in evolution strategies applied to a constrained linear problem. *Evol. Comput.* **21**(3), 389–411 (2013)
2. Arnold, D.V., Brauer, D.: On the behaviour of the (1+1)-ES for a simple constrained problem. In: Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N. (eds.) PPSN 2008. LNCS, vol. 5199, pp. 1–10. Springer, Heidelberg (2008)
3. Arnold, D.V., Hansen, N.: A (1 + 1)-CMA-ES for constrained optimisation. In: Genetic and Evolutionary Computation Conference – GECCO, pp. 297–304. ACM Press (2012)
4. Arnold, D.V., Porter, J.: Towards an augmented Lagrangian constraint handling approach for the (1+1)-ES. In: Genetic and Evolutionary Computation Conference – GECCO, pp. 249–256. ACM Press (2015)
5. Auger, A., Hansen, N.: A restart CMA evolution strategy with increasing population size. In: IEEE Congress on Evolutionary Computation – CEC, pp. 1769–1776. IEEE Press (2005)
6. Bagheri, S., Konen, W., Emmerich, M., Bäck, T.: Solving the G-problems in less than 500 iterations: improved efficient constrained optimization by surrogate modeling and adaptive parameter control (2015). [arXiv:1512.09251](https://arxiv.org/abs/1512.09251)
7. Hansen, N., Arnold, D.V., Auger, A.: Evolution strategies. In: Kacprzyk, J., Pedrycz, W. (eds.) Springer Handbook of Computational Intelligence, pp. 871–898. Springer, Berlin (2015)
8. Hellwig, M., Arnold, D.V.: Comparison of constraint handling mechanisms for the (1, λ)-ES on a simple constrained problem. *Evol. Comput.* **24**(1), 1–23 (2016)
9. Kern, S., Müller, S.D., Hansen, N., Büche, D., Ocenasek, J., Koumoutsakos, P.: Learning probability distributions in continuous evolutionary algorithms – a comparative review. *Nat. Comput.* **3**(1), 77–112 (2004)
10. Liang, J.J., Runarsson, T.P., Mezura-Montes, E., Clerc, M., Suganthan, P.N., Coello Coello, C.A., Deb, K., Problem definitions and evaluation criteria for the CEC special session on constrained real-parameter optimization. Technical report, Nanyang Technological University, Singapore (2006)
11. Michalewicz, Z., Schoenauer, M.: Evolutionary algorithms for constrained parameter optimization problems. *Evol. Comput.* **4**(1), 1–32 (1996)
12. Nocedal, J., Wright, S.: Numerical Optimization, 2nd edn. Springer, Berlin (2006)
13. Rechenberg, I.: Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Friedrich Frommann Verlag, Stuttgart (1973)
14. Regis, R.G.: Constrained optimization by radial basis function interpolation for high-dimensional expensive black-box problems with infeasible initial points. *Eng. Optim.* **46**(2), 218–243 (2014)

15. Schwefel, H.-P.: Numerical Optimization of Computer Models. Wiley, Hoboken (1981)
16. Takahama, T., Sakai, S.: Constrained optimization by the ϵ constrained differential evolution with gradient-based mutation and feasible elites. In: IEEE World Congress on Computational Intelligence – WCCI, pp. 308–315. IEEE Press (2006)
17. Teytaud, O., Gelly, S.: General lower bounds for evolutionary algorithms. In: Runarsson, T.P., Beyer, H.-G., Burke, E.K., Merelo-Guervós, J.J., Whitley, L.D., Yao, X. (eds.) PPSN 2006. LNCS, vol. 4193, pp. 21–31. Springer, Heidelberg (2006)