Enabling High-Dimensional Surrogate-Assisted Optimization by Using Sliding Windows

Bernhard Werth
bernhard.werth@fh-hagenberg.at
Heuristic and Evolutionary Algorithms Laboratory
University of Applied Sciences Upper Austria
Hagenberg, Austria
Institute for Formal Models and Verification
Johannes Kepler University
Linz, Austria

Erik Pitzer
erik.pitzer@fh-hagenberg.at
Heuristic and Evolutionary Algorithms Laboratory
University of Applied Sciences Upper Austria
Hagenberg, Austria

Michael Affenzeller
michael.affenzeller@fh-hagenberg.at
Heuristic and Evolutionary Algorithms Laboratory
University of Applied Sciences Upper Austria
Hagenberg, Austria
Institute for Formal Models and Verification
Johannes Kepler University
Linz, Austria

ABSTRACT

A major drawback of surrogate-assisted evolutionary algorithms is their limited ability to perform in high-dimensional scenarios. This paper describes a possible meta-algorithm scheme for the application of surrogate models to high-dimensional optimization problems. The main assumption of the proposed method is that for some of these expensive problems the nonlinear interactions between variables are sparse. If these interactions can be represented as a band matrix, they can be exploited by applying low-dimensional heuristic solvers in a sliding window fashion to the high-dimensional problem. A special type of composite test function is presented and the proposed meta-algorithm is compared against standard evolutionary algorithms.

CCS CONCEPTS

•**Theory of computation** → **Continuous optimization** ; •**Computing methodologies** → *Modeling and simulation;*

KEYWORDS

surrogate modeling, high dimensional heuristic optimization

ACM Reference format:
Bernhard Werth, Erik Pitzer, and Michael Affenzeller. 2017. Enabling High-Dimensional Surrogate-Assisted Optimization by Using Sliding Windows. In *Proceedings of GECCO '17 Companion, Berlin, Germany, July 15-19, 2017,* 8 pages.
DOI: http://dx.doi.org/10.1145/3067695.3082536

GECCO '17 Companion, Berlin, Germany

978-1-4503-4939-0/17/07…$15.00
DOI: http://dx.doi.org/10.1145/3067695.3082536

1 INTRODUCTION

In traditional heuristic optimization applications like the *travelling salesman*or the *knapsack* problem , evaluation of solution candidates can be performed efficiently. In the case of *simulation-based optimization*, where the evaluation function is itself a potentially costly computer simulation, the number of evaluations required for most heuristic algorithms to converge is often exceeding a feasible computational budget. The use of surrogate models to facilitate the search by reducing the number of required evaluations is therefore a quite sensible method. Some simulation-optimization combinations like *microscopic traffic simulation* [3] paired with the goal to optimize routes or departure times carry not only the burden of an expensive evaluation function but a high dimensional search space (hundreds up to a a few thousand parameters) as well. As the performance of surrogate assisted algorithms diminishes with such high dimensionality, new techniques are required to solve these problems. For the planning of individual vehicle routes it is a reasonable assumption, that vehicles starting at similar times are more likely to interact with each other than vehicles starting at very different points in time. Or in general, many time-based simulations might have late effects, that are not or hardly influenced by early effects. The proposed meta-algorithm tries to exploit such structures by optimizing a sorted parameter vector in a sliding window fashion.

The remainder of this paper is structured as follows: Section 2 gives a brief overview about the main surrogate assisted optimization approaches in scientific literature and enumerates some reasons why high dimensionality is detrimental there. Section 3 explains the concept of variable interaction or *epistasis* and how to detect it. Section 4 describes the proposed algorithm and in Section 5 some computational results on well known and newly introduced test functions are presented. Conclusions and closing remarks can be found in Section 6.

2 SURROGATE ASSISTED OPTIMIZATION

The use of surrogate models to approximate expensive black-box functions with the goal to enable gradient-free optimization is a well-established technique. Several types of models have been proposed to act as a *response surface model* emulating the real

function and therefore allowing the search to continue without having to perform an expensive evaluation. The most prominent ones are *polynomial regression* [22], *Radial Basis Functions* (RBFs) [30], Gaussian process models also called *kriging models* [14, 17], *support vector machines* and artificial neural networks [25, 39].

2.1 Literature Review

The proposed methods for using surrogates in heuristic optimization are manifold and can not be exhaustively described here. Wang et al. [35] found three major categories of surrogate modeling techniques:

- The *sequential* approach where only a single model is built using almost all the available computational budget, the resulting model is then explored with heuristic or exact optimization and the obtained optimum is then evaluated with the expensive black-box to prove its quality. The most notable approach of this category is the *response surface methodology* by Montgomery and Myers [22].
- The *adaptive* approach is mainly the iterative application of model building, selecting one or multiple points based on the model, and evaluating these points with the goal to adaptively sample towards the optimum. One of the most-used surrogate-based optimization algorithms of this category is the *efficient global optimization* (EGO) proposed by Jones et al. [14], which uses the *expected improvement* measure as an infill criterion to select the next sample point. Many variations of this algorithm using different infill criteria or parallelizations have since been presented [7, 37]. A number of such criteria designed for noisy black-boxes can be found in [26].
- The third approach described by Wang was the *direct sampling* approach that transforms the model directly into a probability density function instead of applying an optimizer to the model. However, to the best of our knowledge, this approach has not found broader attention in published literature.

A very well researched variant of surrogate-based optimization is enhancement of evolutionary algorithms with surrogate models by replacing some of the evaluations made by the algorithm with cheaper model predictions. In [19] a surrogate enhanced version of the well known covariance matrix adaption evolution strategy is described using a generation-based scheme where some generations are evaluated on the real fitness function and most others on the surrogate model. Contrasting generation based uses of surrogates are individual-based strategies [5] where in each generation only the most promising solutions undergo exact evaluation. Closely related to individual-based approaches is the idea of pre-selection by means of surrogates in evolutionary strategies [34].

More in depth classifications, surveys and literature reviews of surrogate-assited optimization techniques are available aplenty [10, 13, 31, 35]

2.2 Challenges of High Dimensionality

A well known Achilles heel of many optimization techniques employing surrogate models is their limited performance in high-dimensional search spaces which can be attributed to a number of factors [10].

- Some surrogate models like polynomial regression models require a minimum number of sample points to build which scales quadratically or worse with the dimensionality of the search space [36]
- Interpolation methods like Gaussian processes or radial basis function models may perform adequately when estimating fitness values between sample points (hence the term "interpolation") but loose a lot of their predictive power when estimating outside the convex hull of samples. With increasing dimensionality, the probability that any surrogate-assisted algorithm samples within such a convex hull diminishes.
- Even if enough sample points could be obtained to create reliable predictions, most models require the inversion of a covariance matrix which not only introduces the problem of numerical instability but scales worse than quadratically with the number of sample points used in terms of computation time [15].
- One can easily argue that optimization problems of high dimensionality are inherently more difficult to optimize than their low dimensional counterparts due to the exponential increase in search space size.
- Lastly, the construction of the surrogate model itself may not be cheap after all, since the often used Gaussian process models require a training effort that scales cubically with the number of evaluated sample points [28].

It is therefore imperative to develop new frameworks and approaches to allow optimization in budgeted high-dimensional scenarios.

3 EPISTASIS OF OBJECTIVE FUNCTIONS

Many non-surrogate-assisted optimization algorithms for high-dimensional real-valued problems like CCVIL [6] or DECC-2 [38] are based around the idea of separable fitness functions for which groups of variables exist that can be optimized separately and in the best case the size of all such groups is one. Such a function is called fully separable, satisfies Equation 1 and can be optimized dimension by dimension.

$$\begin{aligned} &\arg\min f(\vec{x}) = \\ &(\arg\min f(x_1, \cdots), \ldots, \arg\min f(\cdots, x_N)) \end{aligned} \tag{1}$$

In the context of genetic algorithms interactions between variables are equivalent to interactions between genes and can therefore be labeled with the biological term of *epistasis* [23, 29] which will be used for the remainder of this paper. The implication of such interactions is that the algorithm may need to optimize these variables together.

3.1 Detecting Variable Interactions

The established techniques to detect interaction between variables x_i and x_j are based on replacing exactly those variables with perturbed values x_i' and x_j' while keeping the rest of the candidate vector fixed and checking equations concerned with either monotony or linear separability [24, 33].

The monotony check in Equation 2 employs the idea that comparative relationships between points at x_i and x_i' may not change when both points are moved the same distance along the j-axis if x_i and x_j belong to two different groups. The *Linkage Identification by Nonlinearity Check* (LINC-R) (Equations 3 - 6) entertains the idea that the changes in x_i and x_j should additively accumulate if separability between both variables exists.

$$\begin{aligned} &f(\ldots, x_i, \ldots, x_j, \ldots,) < f(\ldots, x_i', \ldots, x_j, \ldots) \wedge \\ &f(\ldots, x_i, \ldots, x_j', \ldots,) > f(\ldots, x_i', \ldots, x_j', \ldots) \end{aligned} \tag{2}$$

$$\epsilon < |\Delta f_{ij} - (\Delta f_i + \Delta f_j)| \tag{3}$$

$$\Delta f_{ij} = f(\ldots, x_i', \ldots, x_j', \ldots) - f(\ldots, x_i, \ldots, x_j, \ldots) \tag{4}$$

$$\Delta f_i = f(\ldots, x_i', \ldots, x_j, \ldots) - f(\ldots, x_i, \ldots, x_j, \ldots) \tag{5}$$

$$\Delta f_j = f(\ldots, x_i, \ldots, x_j', \ldots) - f(\ldots, x_i, \ldots, x_j, \ldots) \tag{6}$$

It should be noted, that both separability checks assume deterministic noiseless fitness functions and random effects can easily disturb monotony or linearity assumptions. An alternative to the aforementioned approaches is used in [21] where a high-dimensional model representation comprised of second-order radial basis functions (RBF-HDMR) is constructed in a learning phase and the model coefficients are then used to identify linkage structures.

3.2 Effective Bandwidth

In the frame of *cooperative co-evolution* [27] problems are decomposed by any conceivable strategy and only a subset of variables is perturbed for a certain number of iterations. In the CCVIL algorithm [6], these variable groups are sets of variables previously determined in a learning stage via monotony interaction learning. A function with an interaction matrix as depicted in Figure 1 would therefore be separated in seven such subgroups (one group of size three, one of size two and five groups containing only a single variable). The size of the largest group defines the degree of separability of a function. Functions with interaction matrices similar to Figure 2 are treated as fully non-separable problems since there is no possibility to separate the variables into non-interacting groups and no information of the variable interaction learning step can be used. Nonetheless, it is apparent that some form of structure exists within the variable interactions in Figure 2. The variables are sorted in such a way that the variable interactions form a symmetric band matrix of band width one.

We hereby propose the notion of an *epistasis bandwidth* that is defined as the minimum bandwidth of the variable interaction matrix of a problem over all possible permutations of the problem's variables.

$$bw(f(\vec{x})) = \min_{p \in P_N} bw(VIM(f(p(\vec{x})))) \tag{7}$$

$VIM(f)$ is the *Variable Interaction Matrix* of a specific function f
$bw(A) : R^{nxn} \rightarrow N$ is the bandwidth of a symmetric nxn matrix A

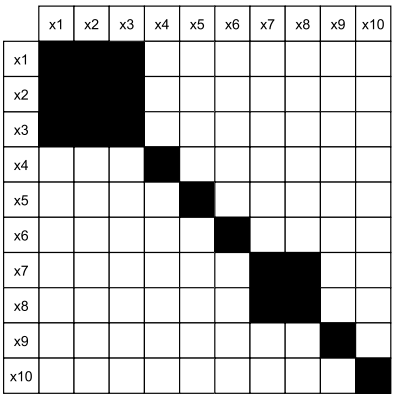

Figure 1: Variable interaction matrix of a 3-separable function

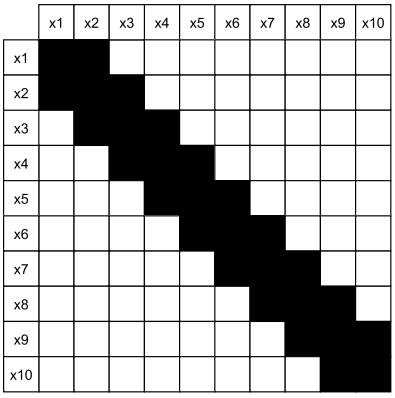

Figure 2: Variable interaction matrix of a non-separable function (bandwidth = 1)

P_N is the set of all permutations of length N. The authors of this paper assume that structures with bandwidth considerably smaller than N appear in many real world optimization tasks, especially simulation-based optimization scenarios where a natural ordering for a large number of variables exists e.g. time.

4 THE WINDOWED OPTIMIZATION APPROACH

The proposed (meta-)algorithm is using a sliding window approach to reduce the complexity of the optimization problem and continuously refine a single solution vector $\vec{m}$. In every iteration, a window of size w is placed on a random starting position $0 \leq p \leq N - w$ and moved forward and backwards by a step size $s < w$ in an alternating motion as depicted in Figure 3. For every window a reduced problem given in Equation 8 is solved by a sub-solver that hopefully profits from the reduced dimensionality. Surrogate assisted optimization approaches are amongst the most promising solvers to be used here as they have been proven to be quite powerful in lower

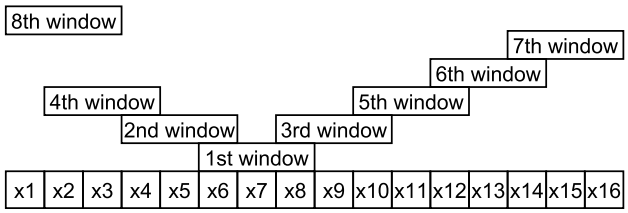

Figure 3: Movement of a sliding window with $w = 3$ and $s = 2$ on a 16-dimensional vector

dimensional spaces but scale poorly to high dimensional spaces.

$$\hat{f}_p(\vec{x}_w) = f(\vec{m}_{1:p-1}, \vec{x}_w, \vec{m}_{p+w:N}) \tag{8}$$

Here $\vec{x}_w$ denotes a partial solution vector of size w. While every potential optimization algorithm, be it surrogate assisted or not, could be used to solve the reduced problem, we use the well known "Efficient Global Optimization" (EGO) algorithm developed by Jones and colleagues [14] and the "Covariance Matrix Adaption Evolution Strategy" (CMA-ES) from Hansen and colleagues [11] as examples. Both algorithms face significant performance hits when confronted with high-dimensional problems and should therefore profit from solving only low-dimensional subproblems. After solving a reduced problem, the best found solution is used to update $\vec{m}$ by replacement.

Since w is assumed to be considerably smaller than N a linkage detection strategy can be used on the reduced problem without running into a minimal sampling effort of $O(N^2)$. The samples used to perform linkage detection can also be used as part of the initial sample population for the surrogate-assisted sub-solver.

After the iteration has ended (the movement of the window has reached both the end and the beginning of $\vec{m}$), one can use the obtained information about variable interactions to reduce the bandwidth of the variable interaction matrix by reordering the variables of f. While finding a reordering that minimizes the bandwidth of a sparse matrix is an $\mathcal{NP}$-hard problem, we find that the Cuthill-McKee-algorithm [8] performs adequately.

The interaction matrix A can additionally be used to dynamically update w. While one could set the new window size to $bw(A)$, using a sightly larger window might be beneficial since it increases the chance of detecting new variable interactions and takes more effects into account even if they are not detected. We propose updating w by Equation 9 since it al.

$$w' = bw(A) \cdot 2 - 1 \tag{9}$$

Algorithms 1 and 2 summarize the proposed method. Algorithm 1 describes the movement of the window along a solution vector, while Algorithm 2 performs epistasis detection, optimization of the current window and updates on the solution.

5 COMPUTATIONAL EXPERIMENTS

In order to create band-like epistasis structures, the strategy of overlapping lower-dimensional test functions was employed. In Figure 4 such a composite five-dimensional function comprising the well-known Ackley function with $d = 3$ and the Hartmann3 function [16] as components is schematically depicted . The Ackley

Algorithm 1: SlidingWindowOptimization

Data: Step size s, Window Size w, Problem f
initialize sparse variable interaction matrix for f
sample $\vec{m}$ uniformly random in each dimension
evaluate $\vec{m}$
while *computational budget not exhausted* **do**
  sample p uniformly random in $[1, N - w]$
  i=0
  while *feasible windows possible* **do**
    if $p - s \cdot i > 0$ **then**
      OptimizeWindow at position $p - s * i$
    if $p + s \cdot i \leq N - w$ **then**
      OptimizeWindow at position $p + s * i$
    i++;
  if $p\%s \neq 0$ **then**
    OptimizeWindow at position 0
  if $(N - p - w)\%s \neq 0$ **then**
    OptimizeWindow at position $N - w$
  Optional:
  construct Cuthill-McKee-Ordering
  reorder f and $\vec{m}$ by Cuthill-McKee-Ordering
  update w

Algorithm 2: OptimizeWindow

Data: Window position p, Window Size w, Problem f, Vector $\vec{m}$, initial sampling size s_i, adaptive sampling size s_a
construct $\hat{f}$ from f and $\vec{m}$
$\vec{s} = \vec{m}_{p:p+w-1}$
for *each Pair* (i, j) *with* $i \neq j$ *in window* **do**
  if *epistasis already determined* **then**
    continue
  Determine epistasis by perturbing $\vec{s}$ in dimensions i, j and evaluating the resulting vectors as described in LINC-R
Sample remaining points in initial sampling size according to preferred scheme (eg. Latin Hypercube)
Run preferred sub solver (eg. EGO) with s_a as computational budget
Update $\vec{m}$ by replacement if an improvement has been achieved.

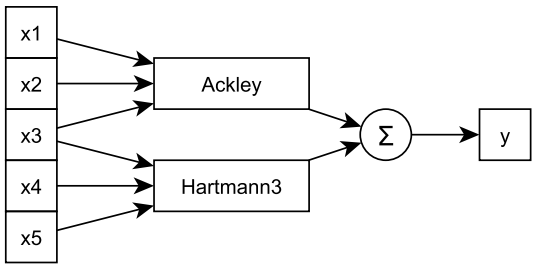

Figure 4: Overlapping test functions

Table 1: Test functions used as components in the composite test functions. All test functions are implemented as described in [2].

Function	Linear separable	Bounds
Ackley	no	$[-32.768, 32.768]$
Griewank	no	$[-600, 600]$
Levy	yes	$[-10, 10]$
Rastrigin	yes	$[-5.12, 5.12]$
Schwefel (Sine root)	yes	[-500,500]

function is applied to the first three parameters of the solution candidate and the Hartmann3 function to the last three parameters. Both functions' values are then summed to achieve one objective value.

In the experiments presented in this paper the five scalable test functions given in Table 1 have been used to construct three composite test functions (*ctf2*, *ctf3*, *ctf5*) for the proposed sliding window strategy.

The five component functions given in Table 1 were used with $d = 2, 3, 5$ in repeating alphabetical order. All of these test functions are multi-modal and scalable and allow for the individual components to have different characteristics. These composite test functions are themselves scalable and were tested with dimensions 100, 500 and 1000.

A number of transformations were applied to the individual component functions:

- **Scaling**: All of the test functions given below were linearly scaled to $[0, 1]^d$ in order to counteract the widely different bounding boxes.
- **Rotation**: In order to create components with epistasis from separable functions, random rotation matrices were created for each component by uniform sampling as described in [18].
- **Overlap** Similar to the schematic depiction in Figure 4, components were chosen to overlap at one parameter each time to create fully non-separable functions.

The optimal objective value for all component functions is zero but this is not the case for the composite function due to two effects. Firstly, while the Ackley, Griewank and Rastrigin functions have their optimum at the origin, the Levy function as well as the Schwefel (sine root) function have optimal function values at $(1, 1, 1, \ldots)$ and $(420.9687, 420.9687, \ldots)$ respectively, which creates a mixture of conflicting and non-conflicting components when overlapped. By creating conflicting components the optimal solution for each individual component can not be achieved simultaneously. Secondly by shifting and rotating the components, function values outside the usual bounding boxes can be achieved, which is mainly relevant for the Schwefel function which has several minima with function values less than zero outside its original bounding box.

To also analyze the performance of the proposed approach also on more common functions, the well known Rastrigin and sphere functions were used as examples for separable scenarios and the Rosenbrock function given in Equation 10 as a special case of a non-separable function with an epistasis band width of two which is caused by the term $(x_{i+1} - x_i^2)^2$.

$$y(\vec{x}) = \sum_{i=1}^{d-1} \left(100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2 \right) \tag{10}$$

The windowed optimization approach was compared against a evolution strategy (ES) with multiple mutation and recombination operators [4], a genetic algorithm extension using an age layered population structure (ALPS) [12] and the Self Adaptive Segregative Genetic Algorithm with Simulated Annealing Aspects (SASEGASA) [1] as these three algorithms were the best performing ones on the newly introduced composite test functions. For the sub solver in the windowed approach the afore mentioned CMAES and EGO algorithms were used (WO-CMAES and WO-EGO). While a direct comparison of the windowed and original versions of the surrogate-assisted algorithms would be of interest, directly applying EGO on a problem with even a few dozen parameters and several thousand evaluations would have been infeasible.

As an initial window size for the WO-CMAES and WO-EGO algorithms $w_{WO-CMAES} = 10$ and $w_{WO-EGO} = 5$ with a a budget of $s_{a,WO-CMAES} = 2000$ and $s_{a,WO-EGO} = 80$ evaluations for each window have been chosen in preliminary experiments as trade offs between wasting too many evaluations on a single window and preemptively stopping a sub-solver before convergence. EGO's 80 evaluations were spent on 50 samples for initial sampling and epistasis detection and 30 iterations maximizing the expected improvement with a single evaluation each.

The total number of evaluations for each algorithm was defined as $k = 1000 \cdot d$ when comparing conventional algorithms similar to [20] and other papers where surrogate assisted algorithms were compared. However we feel that such a big multiplication factor is rarely feasible with large values of d therefore $k = 100 * d$ for comparisons with the surrogate assisted algorithm. A questionable feature of some optimization algorithms is their convergence towards the middle of the feasible bounding box even if no information to guide the search to this region exists. To alleviate an unfair advantage stemming from the fact that most test functions have their global optimum at or in the vicinity of this center, all test problems were shifted by a random vector as it is common practice in the CEC competition for large scale optimization and similar evolutionary competitions [32]. Each combination of algorithm and problem was evaluated 10 times with the same shift vector and rotation matrix.

Algorithm 3: DisturbOrdering

```
Data: Problem f
s⃗ = m⃗_{p:p+w−1}
for int i in 2...n do
    u = randomUniform(0, 1)
    if u < 0.5 then
        swap parameters i and i − 1 of f
```

Figure 5 depicts the box-and-whiskers-plots of achieved objective values for the 500-dimensional variants of the test problems for all tested algorithms that do not use surrogate modeling. As can be seen, the windowed CMAES (WO-CMAES) is almost on par with the

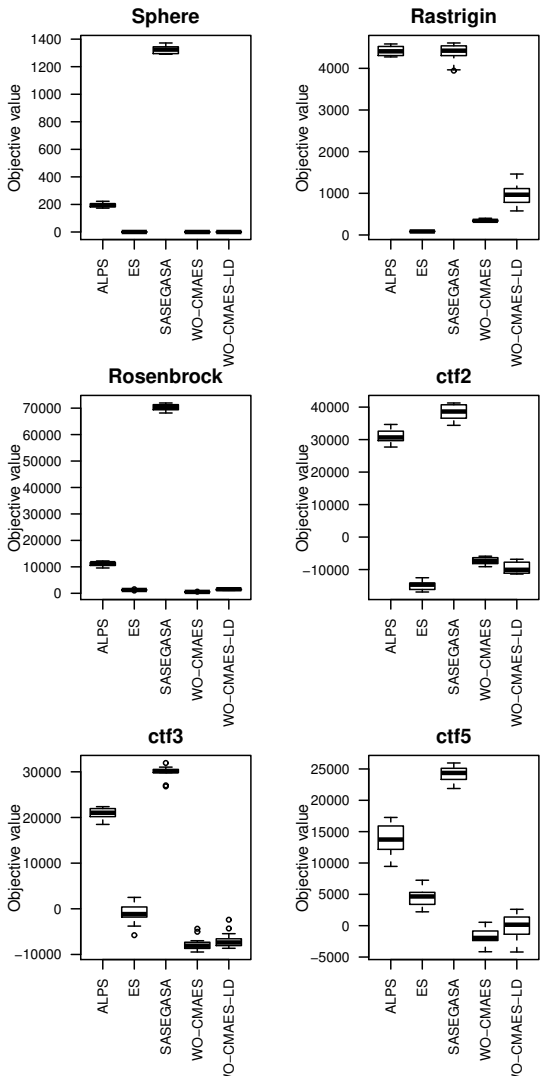

Figure 5: Performance of non-surrogate algorithms on 500 dimensions.

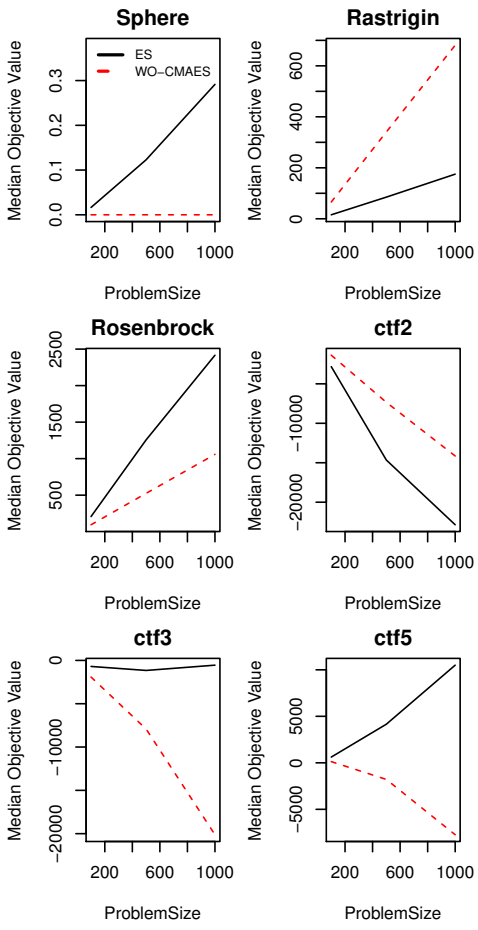

Figure 6: Difference in performance of ES and WO-CMAES

regular ES on problems with low or no epistasis and outperforms both GA-based approaches, at least for this selected set of problems. The differences between the WO-CMAES variants with and without linkage detection (WO-CMAES-LD) can be attributed to the fact that all problems are perfectly sorted in terms of epistasis and therefore the -LD variant is effectively wasting evaluations. For the sake of visibility the GA-based algorithms ALPS and SASEGASA are not displayed in the remaining figures.

The direct comparison between ES and WO-CMAES over problems of different dimensionality given in Figure 6 displays an interesting trend. While for the problems with low or no epistasis the difference in performance between both algorithms scales linearly with the dimension of the problem regardless of which algorithm actually performs better, the difference in performance changes nonlinearly for problems with more epistasis (*ctf5*) in favor of WO-CMAES. A possible explanation would be that the chance for ES-operators, that update multiple parameters at the same time, to randomly choose parameters belonging to the same component decreases with dimensionality in these scenarios. The downward trend of the composite test functions is an effect of including the Schwefel sine-root function as a component, as it contains several optima with objective values below zero that can be reached due to the rotation of the component. As more Schwefel components are present in the higher dimensional composite test functions the minimal achievable value decreases.

In order to simulate an imperfect and a wrongly chosen initial sorting order for the optimization parameters, the test functions ctf2, ctf3 and ctf5 are subjected to two types of reordering. To simulate sorting order completely unrelated to their interaction a random permutation is used (e.g. *Shuffled_ctf2*). For simulating a sup-par ordering of the problems, Algorithm 3 is used to create a slightly disturbed permutation (e.g. *Disturbed_ctf2*). As can be seen in Figure 7 where the results of the WO-CMAES are compared with all three types of shuffling on the 500 dimensional composite test functions, having a viable initial ordering is pivotal in a positive and negative sense for the success of the windowed optimization approach. On

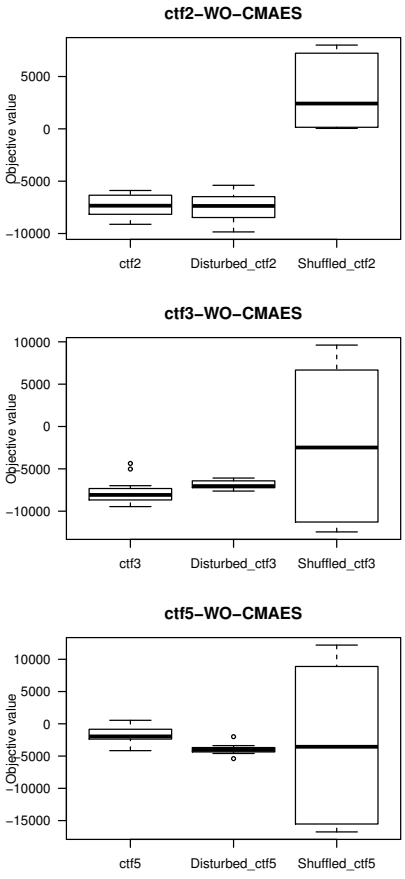

Figure 7: Difference in performance of WO-CMAES on different degrees of shuffling

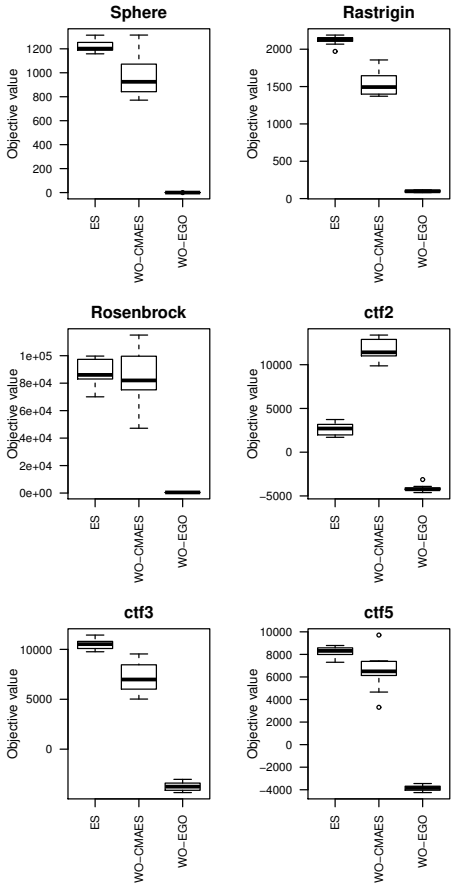

Figure 8: Performance of Windowed EGO algorithms on 100 dimensions compared to the best non-surrogate contenders

some occasions the shuffled versions of a problem allow for better fitness values than their perfectly aligned counterparts. This is an indication that problems can be constructed in a way that misleads the search and optimizing one component can prevent the algorithm to detect the deceptiveness of another component. In the shuffled cases such a misleading behavior is avoided, since components are distributed along the parameter vector. It could also be argued that such a misleading behavior is less likely the larger the ratio between window size and component size is.

In Figure 8 the performance of the WO-EGO algorithm is compared with all other algorithms on the 100 dimensional versions of the test functions. As expected the surrogate algorithm easily outperforms the conventional algorithms. The weak performance of WO-CMAES in this scenario can be attributed to the fact that with the low number of function evaluations allowed, the sliding window was not moved along the whole solution candidate so that some parameters remained completely untouched after their initial random generation.

6 CONCLUSIONS

In the submitted work a fairly simple but new scheme for dividing high-dimensional problems with some structure in their variable interactions has been presented. The proposed sliding window approach seems to be especially well-suited for surrogate assisted optimization and allows the reuse of existing surrogate-assisted heuristic solvers.

Future work on this approach needs to incorporate a more elaborate strategy on how to distribute the computational budget amongst the solution parameters as for most problems some parameters will have significantly more relevance in terms of quality than others. Additionally, a balance needs to bee struck between spending more evaluations on optimizing a specific window and trying to do multiple passes along the solution vector.

It is certainly imaginable that problems with an apparent band matrix structure exist, that are not susceptible to the proposed deconstruction scheme. However, as already stated in [9], real world problems often have some structure that can (and should) be exploited. When applied to time-based simulation scenarios like traffic simulation, the main strength of the proposed technique compared with other divide and conquer algorithms like cooperative

co-evolution is the possibility to further reduce the time required for optimization by snapshotting or simulating only parts of the whole problem. A reduction that is nearly impossible with the random parameter groups created by cooperative co-evolution.

Lastly, it is noteworthy to mention that information about variable interactions and potential separations into independent subproblems might have additional merit, because they allow a glimpse of deeper understanding the problem and might even carry semantic importance. Overall, the use of problem decomposition in combination with surrogate-assisted optimization appears to be a promising combination as it alleviates some of the challenges surrogate models face with increasing dimensionality without having to construct specialized types of models or training methods.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Gerald Ostermayer and the team of the *Research Group Networks and Mobility*. The authors would also like to thank the anonymous reviewers for their valuable comments and helpful suggestions. This work was supported by the European Union through the European Regional Development Fund (EFRE; further information on IWB/EFRE is available at www.efre.gv.at)

REFERENCES

[1] Michael Affenzeller and Stefan Wagner. 2004. SASEGASA: A new generic parallel evolutionary algorithm for achieving highest quality results. *Journal of Heuristics* 10, 3 (2004), 243–267.

[2] Michael Affenzeller and Stefan Wagner. 2005. Offspring selection: A new self-adaptive selection scheme for genetic algorithms. In *Adaptive and Natural Computing Algorithms*. Springer, 218–221.

[3] Christian Backfrieder, Christoph F Mecklenbrauker, and Gerald Ostermayer. 2013. TraffSim–A Traffic Simulator for Investigating Benefits Ensuing from Intelligent Traffic Management. In *Modelling Symposium (EMS), 2013 European*. IEEE, 451–456.

[4] Hans-Georg Beyer and Hans-Paul Schwefel. 2002. Evolution strategies–A comprehensive introduction. *Natural computing* 1, 1 (2002), 3–52.

[5] J. Branke and C. Schmidt. 2005. Faster convergence by means of fitness estimation. *Soft Computing* 9, 1 (2005), 13–20. DOI:http://dx.doi.org/10.1007/s00500-003-0329-4

[6] Wenxiang Chen, Thomas Weise, Zhenyu Yang, and Ke Tang. 2010. Large-scale global optimization using cooperative coevolution with variable interaction learning. In *International Conference on Parallel Problem Solving from Nature*. Springer, 300–309.

[7] Clément Chevalier and David Ginsbourger. 2013. Fast computation of the multi-points expected improvement with applications in batch selection. In *International Conference on Learning and Intelligent Optimization*. Springer, 59–69.

[8] Elizabeth Cuthill and James McKee. 1969. Reducing the bandwidth of sparse symmetric matrices. In *Proceedings of the 1969 24th national conference*. ACM, 157–172.

[9] Stefan Droste, Thomas Jansen, and Ingo Wegener. 1999. Perhaps not a free lunch but at least a free appetizer. In *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation-Volume 1*. Morgan Kaufmann Publishers Inc., 833–839.

[10] Alexander IJ Forrester and Andy J Keane. 2009. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences* 45, 1 (2009), 50–79.

[11] Nikolaus Hansen and Andreas Ostermeier. 1996. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*. IEEE, 312–317.

[12] Gregory S Hornby. 2006. ALPS: the age-layered population structure for reducing the problem of premature convergence. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 815–822.

[13] Yaochu Jin. 2005. A comprehensive survey of fitness approximation in evolutionary computation. *Soft computing* 9, 1 (2005), 3–12.

[14] Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* 13, 4 (1998), 455–492.

[15] EJ Kansa and YC Hon. 2000. Circumventing the ill-conditioning problem with multiquadric radial basis functions: applications to elliptic partial differential equations. *Computers & Mathematics with applications* 39, 7-8 (2000), 123–137.

[16] Jack PC Kleijnen, Wim van Beers, and Inneke Van Nieuwenhuyse. 2012. Expected improvement in efficient global optimization through bootstrapped kriging. *Journal of global optimization* 54, 1 (2012), 59–73.

[17] Daniel G Krige. 1951. A statistical approach to some basic mine valuation problems on the Witwatersrand. *Journal of the Southern African Institute of Mining and Metallurgy* 52, 6 (1951), 119–139.

[18] Carlos A León, Jean-Claude Massé, and Louis-Paul Rivest. 2006. A statistical model for random rotations. *Journal of Multivariate Analysis* 97, 2 (2006), 412–430.

[19] Ilya Loshchilov, Marc Schoenauer, and Michele Sebag. 2012. Self-adaptive surrogate-assisted covariance matrix adaptation evolution strategy. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation*. ACM, 321–328.

[20] Ilya Loshchilov, Marc Schoenauer, and Michèle Sebag. 2013. Intensive surrogate model exploitation in self-adaptive surrogate-assisted cma-es (saacm-es). In *Proceedings of the 15th annual conference on Genetic and evolutionary computation*. ACM, 439–446.

[21] Sedigheh Mahdavi, Mohammad Ebrahim Shiri, and Shahryar Rahnamayan. 2014. Cooperative co-evolution with a new decomposition method for large-scale optimization. In *Evolutionary Computation (CEC), 2014 IEEE Congress on*. IEEE, 1285–1292.

[22] Douglas C Montgomery and Raymond H Myers. 1995. Response surface methodology: process and product optimization using designed experiments. *A Wiley-Interscience Publications* (1995).

[23] Jason H Moore. 2005. A global view of epistasis. *Nature genetics* 37, 1 (2005), 13–14.

[24] Masaharu Munetomo and David E Goldberg. 1999. Identifying linkage groups by nonlinearity/non-monotonicity detection. In *Proceedings of the genetic and evolutionary computation conference*, Vol. 1. 433–440.

[25] Jooyoung Park and Irwin W Sandberg. 1991. Universal approximation using radial-basis-function networks. *Neural computation* 3, 2 (1991), 246–257.

[26] Victor Picheny, Tobias Wagner, and David Ginsbourger. 2013. A benchmark of kriging-based infill criteria for noisy optimization. *Structural and Multidisciplinary Optimization* 48, 3 (2013), 607–626.

[27] Mitchell A Potter and Kenneth A De Jong. 2000. Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary computation* 8, 1 (2000), 1–29.

[28] Carl Edward Rasmussen. 2004. Gaussian processes in machine learning. In *Advanced lectures on machine learning*. Springer, 63–71.

[29] Colin R Reeves and Christine C Wright. 1995. Epistasis in Genetic Algorithms: An Experimental Design Perspective.. In *ICGA*. 217–224.

[30] Rommel G Regis and Christine A Shoemaker. 2005. Constrained global optimization of expensive black box functions using radial basis functions. *Journal of Global optimization* 31, 1 (2005), 153–171.

[31] Songqing Shan and G Gary Wang. 2010. Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. *Structural and Multidisciplinary Optimization* 41, 2 (2010), 219–241.

[32] Ke Tang, Xın Yáo, Ponnuthurai Nagaratnam Suganthan, Cara MacNish, Ying-Ping Chen, Chih-Ming Chen, and Zhenyu Yang. 2007. Benchmark functions for the CECfi2008 special session and competition on large scale global optimization. *Nature Inspired Computation and Applications Laboratory, USTC, China* (2007), 153–177.

[33] Masaru Tezuka, Masaharu Munetomo, and Kiyoshi Akama. 2004. Linkage identification by nonlinearity check for real-coded genetic algorithms. In *Genetic and Evolutionary Computation Conference*. Springer, 222–233.

[34] Holger Ulmer, Felix Streichert, and Andreas Zell. 2003. Evolution strategies assisted by Gaussian processes with improved preselection criterion. In *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, Vol. 1. IEEE, 692–699.

[35] G Gary Wang and Songqing Shan. 2007. Review of metamodeling techniques in support of engineering design optimization. *Journal of Mechanical design* 129, 4 (2007), 370–380.

[36] Liqun Wang, Songqing Shan, and G Gary Wang. 2004. Mode-pursuing sampling method for global optimization on expensive black-box functions. *Engineering Optimization* 36, 4 (2004), 419–438.

[37] Song Xiao, Mihai Rotaru, and Jan K Sykulski. 2013. Adaptive weighted expected improvement with rewards approach in kriging assisted electromagnetic design. *IEEE Transactions on Magnetics* 49, 5 (2013), 2057–2060.

[38] Zhenyu Yang, Ke Tang, and Xin Yao. 2007. Differential evolution for high-dimensional function optimization. In *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*. IEEE, 3523–3530.

[39] W. Yao, X.Q. Chen, Y.Y. Huang, and M. van Tooren. 2014. A surrogate-based optimization method with RBF neural network enhanced by linear interpolation and hybrid infill strategy. *Optimization Methods and Software* 29, 2 (2014), 406–429. DOI:http://dx.doi.org/10.1080/10556788.2013.777722 arXiv:http://dx.doi.org/10.1080/10556788.2013.777722