

Solving Multimodal Problems via Multiobjective Techniques with Application to Phase Equilibrium Detection

Mike Preuss

Günter Rudolph

Feelly Tumakaka

Abstract—For solving multimodal problems by means of evolutionary algorithms, one often resorts to multistarts or niching methods. The latter approach the question: ‘What is elsewhere?’ by an implicit second criterion in order to keep populations distributed over the search space. Induced by a practical problem that appears to be simple but is not easily solved, a multiobjective algorithm is proposed for solving multimodal problems. It employs an explicit diversity criterion as second objective. Experimental comparison with standard methods suggests that the multiobjective algorithm is fast and reliable and that coupling it with a local search technique is straightforward and leads to enormous quality gain. The combined algorithm is still fast and may be especially valuable for practical problems with costly target function evaluations.

I. INTRODUCTION

A multitude of methods from within and beyond *evolutionary computation* (EC) has been applied to real-valued multimodal optimization problems. These are generally considered the harder, the more basins of attraction they contain, and the less smooth the fitness landscape is. Additionally, a search space that extends over a large number of dimensions is said to complicate search for the desired global or good local optima (e.g. [1]).

However, in a real-world setting, even a low dimensional problem may turn out to be quite difficult. This can stem from different factors, one of which would be a very small extent of the basins that contain the sought optima. Fig. 1 visualizes the fitness landscape of an optimization problem which possesses this property. The application background will be detailed in §III, but for now it suffices to know that there are only two variables x_1 and x_2 , and that the desired optima (function values do not depend on variable order and are thus symmetric to the main diagonal, we minimize) are located near $(0.650, 0.001)$ and $(0.001, 0.650)$, respectively. It is easy to see that the appropriate basins are small; in the figure, they are hardly recognizable at all.

Another complicating factor would be uncertainty about the relative target function value of the sought optima. If it is not a priori known whether we are looking for global, or even the best local optima, there is no way around enumerating all existing optima and choose the ‘right’ solution out of these afterwards. Such difficulties may occur in cases where

it is not possible to integrate the whole available application specific knowledge into the established target function, i.e. if its value must be obtained by simulation and the existing simulation tool is not able to represent all important features of the real system.

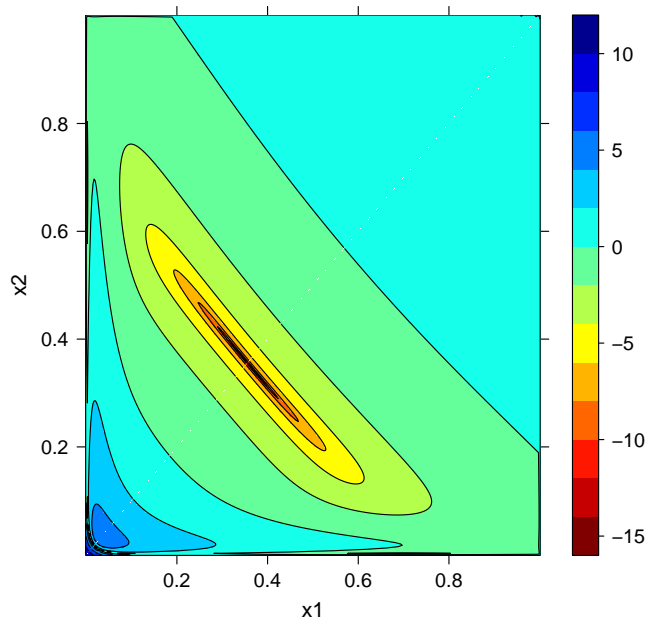


Fig. 1. 400×400 points grid sample of 2-dimensional example problem, minimization. Desired optima near $(0.650, 0.001)$ and $(0.001, 0.650)$.

Nevertheless, many different *evolutionary algorithms* (EAs) may be employed to tackle problems that exhibit both types of difficulties because they are able to detect several optima simultaneously or subsequently. The latter may be achieved by multistart approaches as sequential niching [2], whereas the former is established by means of diversity maintenance. That is, candidate solutions of the search populations are prevented from converging to the same region by implicitly or explicitly keeping them apart (see [3]). Prominent examples are crowding [4] and fitness sharing [5], and their successors. More recent approaches include, but are not limited to: UEGO [6], clearing [7], species conservation [8], clustering based niching [9], and cellular EA (CEA) [10]. Although there is no commonly accepted formal definition of what a niching method is (see [11]), most of these algorithms may be subsumed under the term *niching EA*. They all use the distance between candidate solutions (diversity) as implicit criterion which shall be maximized.

Mike Preuss: Algorithm Engineering, Department of Computer Science, University of Dortmund, email: mike.preuss@uni-dortmund.de .

Günter Rudolph: Algorithm Engineering, Department of Computer Science, University of Dortmund, email: guenter.rudolph@uni-dortmund.de .

Feelly Tumakaka: Thermodynamics, Department of Biochemical and Chemical Engineering, University of Dortmund, email: feelly.tumakaka@bci.uni-dortmund.de .

However, nothing prevents us from utilizing a diversity criterion directly. A step into this direction has been taken in the shifting balance GA [12]. But although it employs a separate diversity evaluation via subpopulation distance computation, it finally resorts to a single objective by weighting distance and target function values.

In this work, we suggest a more radical approach and introduce diversity in search space as an additional objective and treat the resulting combined problem by an *evolutionary multiobjective algorithm* (EMOA). The expected benefit is twofold:

- It enables placing solution candidates in basins that would otherwise go unnoticed due to their small size.
- We obtain a good overview of the available 'interesting' search space regions in a single run.

One may argue that the suggested method is doomed to fail for high-dimensional original problems. However, we do not make claims concerning such problems but instead only demonstrate its usefulness on a low-dimensional real-world problem.

In the next sections, we will first concretize our main goals (§II), followed by an introduction of the sample application (§III). We treat the example problem by simple multistart EAs (§IV) and introduce our multiobjective approach in §V, followed by hybridizations of multiobjective/local search algorithms for improved solution quality (§VI), and the conclusions.

II. AIMS AND METHODS

The main objective of this paper is to investigate if the newly suggested multiobjective approach to solve multimodal problems is feasible. Concerning efficiency and solution quality, it shall be at least competitive to well adapted naive multistart EAs, namely (1+1) and (μ, λ) *evolution strategies* (ESs). Furthermore, we want to relate its performance to the one measured for CEAs as a representative of EAs specialized on multimodal problems. It may be expected that multistart methods will outperform our multiobjective approach in terms of solution quality (approximation error). We thus additionally investigate if this anticipated advantage can be outrun by a 2-phase algorithm resulting from the hybridization of multiobjective with local search methods.

We perform all comparisons experimentally, and parameter values of the employed algorithms are adapted to the problem by means of the *sequential parameter optimization* (SPO) [13], a largely automated tuning method specially designed for nondeterministic optimization techniques. We measure performance by means of the *average evaluations to solution* (AES) and *mean best fitness* (MBF) indicators. Alternatively, the *success performance 2* (SP2) measure [14] could be used. However, it enforces stopping the optimization process at the first encounter of the optimum, resulting in variable run lengths. As this is undesirable from the viewpoint of the tuning method, we allow for repeated multistarts even after

the first hit, counting up the number of hits per run under a fixed run length.

The AES cannot be determined properly with an upper bound of evaluations for one run T , because T may still be too small to let runs terminate naturally. We therefore resort to estimating it from the number of hits h of the sought optimum over all repeats r for a given parametrization. Note that the approximation error introduced by this simplification shall be rather small for multistart techniques as the separate subruns are independent. Only the result obtained from the last subrun may be wrong because it ends prematurely.

$$\text{AES} \approx \overline{\text{AES}} = \frac{T \cdot r}{\sum_{i=1}^r h_i} \quad (1)$$

To avoid a possible zero denominator, we report the average number of hits $\varnothing h_i$, which is equivalent to $\frac{T}{\overline{\text{AES}}}$. As a side effect, our measure gets Poisson-distributed instead of geometrical.

As we undertake a first study, the obtained assessment of our approach is necessarily very rough. However, it shall only motivate or obviate further research in this direction. We therefore abstain from using many test problems and concentrate on only one interesting instance.

III. PHASE-EQUILIBRIA CALCULATION PROBLEMS

The knowledge of phase equilibria is required for the design and optimization of separation processes which are essential parts of typical chemical plants. The aim of *phase-equilibria calculation* (PEC)¹ is to quantitatively relate the variables (in particular, temperature T , pressure p , and composition x) which describe the state of equilibrium of two or more homogenous phases [15].

In any problem concerning the equilibrium distributions of some component i between two phases α and β , one must always begin with the equality of the chemical potential μ as

$$\mu_i^\alpha = \mu_i^\beta, \quad \forall i \quad (2)$$

To establish the relation of $\mu_i^{\alpha, \beta}$ to T , p , and x_i^α , it is convenient to introduce a certain auxiliary function such as fugacity coefficient $\varphi_i^\alpha(T, p, x_i^\alpha)$ which can be calculated by a thermodynamic model. Then, (2) can be rewritten

$$x_i^\alpha \cdot \varphi_i^\alpha = x_i^\beta \cdot \varphi_i^\beta, \quad \forall i \quad (3)$$

Typically, the calculation is performed at constant temperature and pressure, and the remaining concentrations x_i^α and x_i^β , respectively, are to be found.

Solving phase-equilibria problem according to (3) may lead to trivial solutions, i. e. $x_i^\alpha = x_i^\beta$, which are mathematically correct but have no physical meaning (except at the so-called

¹This is not a standard abbreviation, we use it only for simplification.

²We use the domain specific notation with upper index denoting different phases and lower index standing for separate substances.

critical demixing point). To avoid this, the initial guesses for the minimization procedure are not allowed to be too far away from the correct solutions, provided that the correct solutions are known.

In case of polymer solutions, initialization is very critical, because the concentration of the polymer in the solvent-rich phase can be in the magnitude of 10^{-20} , which is a numerical challenge for a simulation program [16]. Another difficulty rises as the number of components in the mixture increases, because the possible number of additional phases also increases. All of these point out the need for a robust algorithm to solve the phase-equilibria calculation for arbitrary number of components and phases, which is also applicable to polymer solutions.

As representative for a relatively simple test problem in the context of phase-equilibria calculation, we use a simple two-component mixture of water and pentanol, hereafter referred to as PEC1. This type of liquid-liquid equilibrium data are necessary for the design and optimization of liquid-liquid extractors and of decanters in distillation systems.

The two variables $x1$ and $x2$ denote x_i^α and x_i^β , respectively. They correspond to the concentrations of water in the water-rich phase (for the larger of the two) and in the pentanol-rich phase (for the smaller one). Under the assumption that $x1 > x2$, and the $x1$ and α stand for water and $x2$ and β for pentanol, we have $x1 = x_1^\alpha$, and $x2 = x_1^\beta$.

For this two-component problem, two equations of type (3) have to be satisfied, resulting in two error values e_1 and e_2 . A feasible solution to the problem shall exhibit errors below 10^{-10} due to practical requirements. In the following, we aggregate e_1 and e_2 into a single target function value by using the sum of squares (4), which is to be minimized.

$$f(x1, x2) = e_1^2 + e_2^2 \quad (4)$$

Additionally, we define a criterion for detecting the search points in the basin of attraction of the sought optimum (5). As the exact basin is not known, it is estimated from a grid sample.

$$\sqrt{(x1 - 0.65)^2 + (x2 - 0.001)^2} \leq 0.07 \wedge f(x1, x2) \leq 0.01 \quad (5)$$

If not otherwise stated, we model the PEC1 problem at a temperature of 90°C, for which the sought optimum is located near the coordinates (0.650, 0.001). As system properties change with temperature and pressure, the pursued optimum also moves through the search space. Table I depicts approximate solutions for different temperatures and constant pressure of 1.0132 bar.

IV. TREATMENT WITH STANDARD EAS

As can be seen from Fig.1, the predominant difficulty any optimization method must overcome in order to solve PEC1 is to find the ‘right’ basin of attraction. Different evolutionary algorithms possess different abilities to handle this global

TABLE I
THE SOUGHT OPTIMUM OF PCE1 AT DIFFERENT TEMPERATURES

| variable | 40°C | 60°C | 90°C |
|----------|------------|------------|------------|
| $x1$ | 0.74698 | 0.7097 | 0.65084 |
| $x2$ | 0.00020913 | 0.00038142 | 0.00082809 |

search. Most approaches fall in the categories of diversity maintenance or multistart, or both, as previously stated. However, parallelization of search in time or space gradually decreases local search performance if the effort is kept constant, so that at some point, hybrids of EAs with local search methods may get interesting. We thus concentrate on the basin finding step first, and, for the time being, leave it open whether optimization within a basin is done with another algorithm. To achieve an overview over the performance of standard EAs on the basin finding part of the PEC1 problem, we apply (1+1) and (μ, λ) evolution strategies (ESs), and CEAs with Watts-Strogatz small-world model connections on top of a ring topology as utilized in [17] (these are called swn-CEA in the following). Recent investigations indicate that multistart methods are often at least competitive to more complex EAs [18], so that all three algorithms are enhanced with a simple multistart mechanism, triggered by a number of generations without improvement. For the CEAs, this resembles a number of loosely coupled (1+1)-EAs that are all started anew if they all fail to improve during a certain number of iterations. As only global search capabilities are of interest here, we also initiate a restart whenever the ‘right’ basin is found.

The available PEC1 simulator is relatively fast and allows for ≈ 20 to 50 target function evaluations per second on a modern PC. But then, in a real-world situation, engineers dealing with these problems want them solved in minutes at most. Consequently, we can afford only around 10^4 simulator calls.

A. Experiment 1: Are naive (multistart) EAs or CEAs efficient in detecting the desired optimum of the PEC1 problem?

Pre-experimental planning: First experiments indicate that without multistart, the desired optimum is very rarely attained for (1+1) and (μ, λ) -ES as well as for CEAs. A manually parametrized (stagnation stop $s = 20$, $\sigma_{\text{init}} = 0.1$, $\tau = 0.1$, one mutation strength self-adaptation after Schwefel [19]) multistart (1+1)-ES detects the optimum 20 times in 10 runs of 20000 evaluations each. Its estimated AES is thus 10000. We therefore determine the run length for the SPO runs to this number of evaluations as we expect that the tuned variants will perform even better.

Task: Compare the performance of well parametrized simple hillclimbing (1+1)-EA and (μ, λ) -EA in terms of the number of hits h of the basin of the predefined optimum. As our data is Poisson distributed and most likely not normal ($h < 9$), we employ the Mann-Whitney U-test to detect non-equality (p-value < 0.05) between the best performing variants of the three algorithms.

TABLE II
SPO ALGORITHM DESIGN

| LHDini | budget | repeats | merge |
|----------------|--------|---------|-------|
| 100×4 | 1000 | 4 : 30 | mean |

Setup: The parameters of all three algorithms are tuned by means of the SPO method, using the design given in Tab. II, with only one new parameter set tested in every iteration. The number of repeats is successively increased up to 30 until the new and the current best configuration can be separated at a p-value below 0.05 by a random permutation test as described in [20]. The resulting best configurations and the medium configurations from the initial *latin hypercube design* (LHD) are validated with 30 repeats, respectively. Wherever appropriate, 2-point crossover is applied to the object variables, and intermediate recombination to the mutation strengths. All runs are stopped after $T = 10000$ evaluations. Limits for the tuned parameters are given in Tab. III.

Results/Visualization: The best found configurations for all three algorithms (SPO best) are depicted in Tab. IV, together with the median configuration of the initial LHD as representative of an ‘average’ parameter setting. Figures 2 and 3 show the parameter settings of all SPO tested configurations, for the (1+1)-EA and the sw-n-CEA, respectively. They are split into 3 equally sized groups according to quality.

Observations: Tab. IV indicates that (1+1)-EA and sw-n-CEA perform a lot better than the (μ, λ) -EA. Furthermore, none of the initial LHD parameter settings for the latter lead to any detection of the sought basin. Whether an average (1+1) parameter setting performs reasonably well, this is not so for the sw-n-CEA. However, after tuning, it is able to detect the important region approximately as often as an average (1+1)-EA. Figures 2 and 3 exhibit interesting similarities between successful parameter settings of (1+1)-EA and sw-n-CEA:

TABLE III

EA ALGORITHM DESIGNS: POPULATION SIZE, SELECTION PRESSURE, LEARNING RATE, INITIAL MUTATION STRENGTH, STAGNATION STOP, REWIRING PROBABILITY

| EA | μ | $\frac{\lambda}{\mu}$ | τ | σ_{init} | s | β |
|------------------|-------|-----------------------|--------|------------------------|--------|---------|
| (1+1) | (1) | (1) | 0 : 1 | 0.01:0.5 | 10:100 | — |
| (μ, λ) | 5:150 | 3:10 | 0 : 1 | 0.01:0.5 | 1:20 | — |
| sw-n-CEA | 5:150 | (1) | 0 : 1 | 0.01:0.5 | 1:20 | 0:0.2 |

TABLE IV

EA BEST DETECTED ALGORITHM DESIGNS (SEE TAB. III)

| μ | $\frac{\lambda}{\mu}$ | τ | σ_{init} | s | β | $\mathcal{O}h_i$ |
|--|-----------------------|--------|------------------------|-----|---------|------------------|
| (1+1), top: median LHD, bottom: best SPO | | | | | | |
| (1) | (1) | 0.635 | 0.385 | 61 | — | 4.13 |
| (1) | (1) | 0.18 | 0.350 | 10 | — | 7.50 |
| (μ, λ) , top: median LHD, bottom: best SPO | | | | | | |
| 13 | 6.89 | 0.125 | 0.066 | 11 | — | 0.00 |
| 7 | 3.24 | 0.020 | 0.493 | 18 | — | 0.53 |
| sw-n-CEA, top: median LHD, bottom: best SPO | | | | | | |
| 77 | (1) | 0.075 | 0.110 | 7 | 0.035 | 0.33 |
| 5 | (1) | 0.033 | 0.500 | 1 | 0.140 | 4.00 |

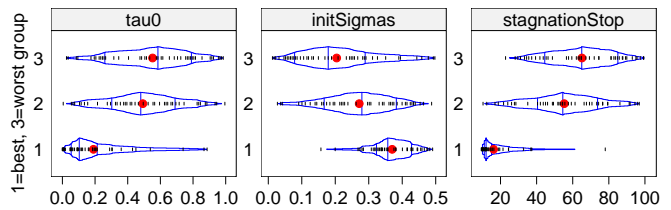


Fig. 2. Parameter effect split plots of the (1+1), depicting all configurations tested by SPO, separated into 3 groups of equal size. Parameters: learning rate, initial mutation strength, and stagnation stop (number of unsuccessful generations).

The initial mutation strengths (step sizes) are driven towards the upper limit of 0.5, and the stagnation stop towards the lower limit (10 and 1, respectively). Lower rewiring (only CEA) and learning rates have a tendency to enable better performance, and small population sizes (only CEA) are obviously better than larger ones.

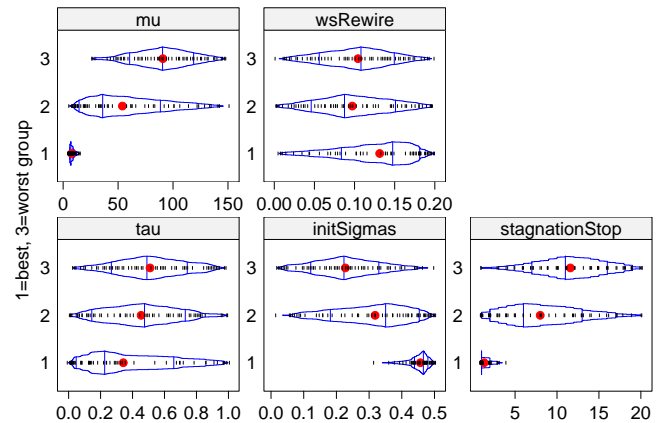


Fig. 3. Parameter effect split plots for the sw-n-CEA, all SPO-tested configurations, separated into 3 groups of equal size. Parameters: population size, rewiring rate for the small-world connectivity, learning rate, initial mutation strength, and stagnation stop.

Discussion: Mann-Whitney U-tests between the tuned and the median LHD variant of each of the three algorithms as depicted in Tab. IV return p-values below 10^{-4} in all cases. That is, the best found configurations perform significantly better than their average untuned counterparts. As the reported differences are quite large, this is well in accordance with intuition. In the same manner, we can safely reject all hypotheses which pairwise assume equal distributions behind the best variants. Here, p-values are even lower ($< 10^{-5}$), meaning that all visually detected differences are also statistically significant. This indicates that the (1+1)-EA performs better than the sw-n-CEA, which is in turn better suited for basin finding on the PEC1 problem than the (μ, λ) -EA.

On a first glance, the tuned parameter settings may seem surprising. However, they are optimized for enabling as many restarts as possible. Interestingly, with an initial stepsize of around 0.5, we nearly arrive at random search. Self-adaptation of mutation strengths is almost switched off as

would be the case for $\tau = 0$. Adaptation of well performing (1+1)-EA into a rapid multistarter is relatively fast (in terms of the tuning effort) as it has only three parameters instead of 5 and its population size is already set ‘right’.

The swn-CEA ‘emulates’ a rapid multistarter by reducing its population size to a minimum. Within such small populations, modifying the rewiring rate does not entail much change in algorithm behavior, as topology diameters are also very small. The remaining parameters follow the same tendency as for the (1+1)-EA, resulting in a very globally oriented and very stochastic search process.

This is exactly the feature the (μ, λ) -EA is missing: It can not adapt into a rapid multistarter because it requires at least $5 \times 3 = 15$ evaluations per iteration, whether the swn-CEA needs only 5 in a minimal configuration.

Concluding, one can state that for the basin finding phase, successful algorithms shall put small weight on search paths, but detect stagnation as fast as possible and initiate search at a new location. With nearly 600 restarts in 10000 evaluations for the best (1+1)-EA, an average search path is shorter than 20 steps. In effect, a restart is done whenever success probabilities drop below a certain level. We conjecture that it is even profitable to erroneously stop search at a good location if some ‘unlucky’ mutations occur. A disadvantage of this technique may be that it is presumably difficult to cluster the final solutions obtained from the restarts as they may still be far away from the corresponding optimum.

V. A NEW APPROACH: EMOAS AS MULTIMODAL EAS

Experiment 1 has shown that it is possible, but difficult to obtain a search point within the basin of attraction of the sought optimum by means of multistart EAs. This holds even if diversity maintenance schemes are applied as is the case for the swn-CEA. Additionally, one ends up with a set of search points that is fairly scattered over the available space. It would thus be difficult to determine which of the many search paths lead to the same optimum, so that some sort of clustering has to be applied.

To overcome this difficulty, we suggest to employ a multiobjective optimization algorithm, namely an EMOA, and use a diversity-based criterion next to the original target function. Note that in contrast to measuring pairwise distances between search points as is common in niching techniques, our second criterion shall only depend on the position of a single individual, without taking the rest of the population into account. Any dependency on other individuals would lead to changes in the second’s criterion values of large parts of the population if one or more individuals are moved due to application of search operators. This would prevent fast approximation of the Pareto front as the true front itself then moves, according to the movement of the population. We thus switch from relative to absolute distances.

In case of our two-variable PEC1 problem, such an absolute criterion is easy to achieve: The distance is obtained from $|x_1 - x_2|$, and it can be converted to a similarity value by

subtracting it from 1, which is the maximum of this term for $x_1, x_2 \in [0, 1]$. This transformation enables to minimize on this criterion (6) as well as on the original target function. Note that it makes sense to reward individuals with small similarities as we know in advance that for the trivial solution of PEC1, x_1 always equals x_2 .

$$f_{\text{sim}}(x_1, x_2) = 1 - |x_1 - x_2| \quad (6)$$

Several evolutionary multiobjective optimization algorithms are available, i.e. NSGA2 and SPEA2. In principle, each of them could be applied. However, we choose the SMS-EMOA [21] as it is known to perform reasonably well even in case of very limited function evaluation budgets. For the same reason, we employ a small population size (20). On the one hand, this is motivated by our need for fast progress towards the Pareto Front. On the other hand, it is desirable to obtain a small approximate Pareto Set with at least one search point within the sought basin.

The task for the EMOA is to filter out search points that lead to the same optimum. It is presumed to work nearer to the basin level than to the local search level, or, stated differently, as a global search method. This is exactly what we demand from the multistart methods in §IV.

In the following experiment, we investigate the SMS-EMOA with the original target function and the additional similarity criterion (6) under similar conditions as applied to the multistart methods in experiment 1.

A. Experiment 2: Are EMOAs capable of reliably detecting points in the sought basin of attraction?

Pre-experimental planning: First test runs lead to the conclusion that *evolution strategy* (ES)-like search operators lead to unsatisfactory results as they require much longer runs to achieve the same level of S-metric (hypervolume) values than standard NSGA-2 operators SBX and PM. With these, it was found that tuning has only little effect on performance. We therefore do not apply SPO here. The critical run length seems to be at around 2000 evaluations, which are needed to obtain S-metric values larger than 0.9. Further analysis reveals that this seems to be a good indicator for placement of individuals in vicinity of the sought optimum.

Task: Within each run, the EMOA shall put at least one individual of the final approximated Pareto front into the sought basin with probability near 1.

Setup: The SMS-EMOA is run 30 times for $T = 2500$ evaluations. The population size is 20, we employ crossover and mutation operators SBX and PM with distribution indices $\eta_c = 15$ and $\eta_m = 20$, as common for NSGA-2. Hits of the sought basin are detected via (5), tested against all individuals of the final population.

Results/Visualization: In 30 runs, 31 of the obtained individuals fulfill the basin condition (5), sometimes up to three per run. 80 % deliver at least one point in the basin, 6 do not. However, the final populations of the unsuccessful runs all

contain individuals that are very near to the basin. Figure 4 depicts 3 example final populations.

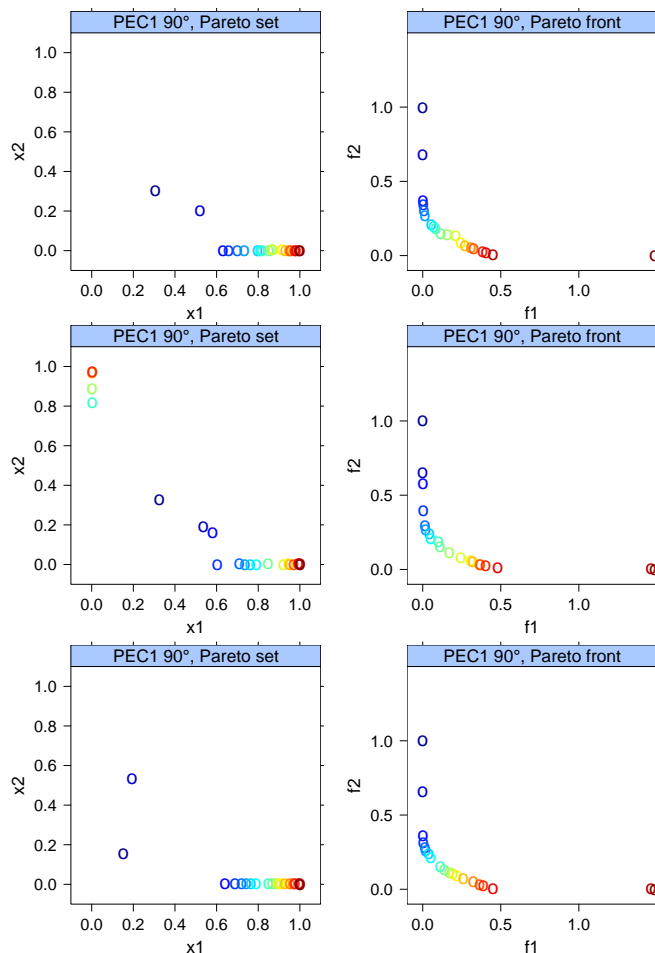


Fig. 4. Three of 30 randomly picked final populations, left column: Pareto sets, right: Pareto fronts. The interesting region is around $x_1 = 0.65$.

Observations: Two interesting facts are recorded. Firstly, the final populations appear to be very homogenous, as can be seen in Fig. 4. Secondly, it often happens that individuals of the final front are placed on both sides of the sought basin, none of them coming too near, as if driven apart by each other. Additionally, it shall be noted that very few individuals of the final fronts point to the trivial solution. Usually, only one of each front is recognizable located near to it.

Discussion: We can state that the SMS-EMOA is able to detect ‘good’ points with high probability. A direct comparison to the multistart methods is not possible with the available data as their runs are much longer. However, it is obvious that the multiobjective approach is at least competitive to them, with the additional advantage of returning only a small result set. It may well be possible to increase the EMOA performance even further by taking a closer look at its parameters, or at the modeling of the second objective. For the multistart methods, this is unlikely, unless new search operators are introduced.

As it frequently happens that the interesting region is surrounded by two or more individuals with none of them

getting really close, one may reconsider the basin condition (5); it may be too strict. However, this condition has been set up as a heuristic only. Determining the real value of the returned points is only possible by additionally applying a local search method afterwards.

VI. A HYBRID EMOA/LOCAL SEARCH APPROACH

The idea is obvious and its realization straightforward. As the EMOA delivers a set of nearly Pareto-optimal points that represent relatively low error in phase equilibrium approximation over a range of distances to the trivial solutions, it is still necessary to decide for each of these points whether it is close to a nontrivial solution. For this purpose we deploy a deterministic direct search algorithm, namely the Hooke and Jeeves [22] method, to each solution in decision space of the EMOA. If the local search approaches a trivial solution it is aborted. This rule fires if $|x_1 - x_2| < 0.01$. Otherwise the method shall get stuck somewhere in the search space $[0, 1] \times [0, 1]$. In this case only solutions with singleobjective function value less than 10^{-10} are candidates for the nontrivial solutions of the phase equilibrium detection problem.

A. Experiment 3: Does the hybrid approach with MOEA and local search lead to a reliable detection and accurate localization of the nontrivial solutions of the phase equilibrium detection problem?

Pre-experimental planning: It must be precluded that the effect of hybridization to be observed is actually caused by the direct search method used in the second phase of the hybrid algorithm. As a consequence, we have run the Hooke/Jeeves method in a multistart manner 500 times with starting points sampled independently from a uniform distribution with support $[0, 1]^2$. We applied the decision rules described above for distinguishing the outcomes of the local search. Table V summarizes the results of our experiments for temperatures at 40° , 60° , and 90° Celsius. Notice that the Hooke/Jeeves search method is parameterized to stop if the step size is less than 10^{-10} . Therefore it may get stuck with considerable distance to an optimum.

TABLE V

PERFORMANCE OF HOOKE/JEEVES ALGORITHM

| Temperature (in Celsius) | #runs to trivial solution | #runs to nontrivial solution | #runs got stuck | relative success frequency |
|--------------------------|---------------------------|------------------------------|-----------------|----------------------------|
| 40° | 494 | 4 | 2 | 0.008 |
| 60° | 494 | 4 | 2 | 0.008 |
| 90° | 482 | 14 | 4 | 0.028 |

Evidently, the success probabilities are much too small for deploying such an approach in practice. The hybrid method should have a significantly higher success rate.

Task: For each approximated Pareto set of the EMOA, apply the Hooke/Jeeves direct search method for each point of the set. If necessary, apply a statistical test to check for significant differences in the success rates if compared to the performance of the multistart variant documented in table V.

Setup: We do not use parameter tuning at this point. Simply deploy some standard implementation of the Hooke/Jeeves method (step size contraction factor $\rho = 0.5$, initial step size $s = 0.001$) that is stopped as described above. The EMOA is run with population size of 20 individuals and a maximum number of 2500 function evaluations in total. The remaining parameters are identical to the parameters chosen in the previous section.

Results/Visualization: The EMOA was run 25 times and the resulting solution set of each run was fed to the Hooke/Jeeves direct search method in a multistart fashion. The results are summarized in table VI.

TABLE VI
AVERAGE PERFORMANCE OF HYBRID ALGORITHM

| temperature (in Celsius) | #runs to trivial solution | #runs to nontrivial solution | #runs got stuck | relative success frequency |
|--------------------------|---------------------------|------------------------------|-----------------|----------------------------|
| 40° | 11.2 ± 2.7 | 7.0 ± 3.1 | 1.8 ± 2.3 | 0.35 ± 0.15 |
| 60° | 8.5 ± 3.1 | 10.0 ± 3.3 | 1.5 ± 1.7 | 0.50 ± 0.17 |
| 90° | 3.8 ± 2.2 | 15.1 ± 2.9 | 1.1 ± 1.6 | 0.76 ± 0.15 |

For example, at temperature 90° Celsius on average 15.1 of the 20 individuals of each EMOA run were 'transformed' to nontrivial solutions of the phase equilibrium problem. In other words: The preprocessing step with the EMOA has generated starting points for the Hooke/Jeeves multistart method such that the success probability is about $15.1/20 \approx 76\%$. For sake of a fair comparison, we have to take into account the 2500 function evaluations of the EMOA in the preprocessing step. Here, we take the most pessimistic point of view by assuming that these function evaluations have been used for runs to a trivial solution. Since such a run requires 138.0 function evaluations on average (see second column of last row in table VIII), we regard the 2500 function evaluations of the preprocessing step as $2500/138.0 < 19$ runs to a trivial solution. As a consequence, our conservative estimator of the success probability is $15.1/(20+19) > 38.7\%$. Table VII provides an overview regarding the improvement of the success probability.

TABLE VII
IMPROVEMENT OF SUCCESS PROBABILITY

| temperature (Celsius) | Success Probability | | |
|-----------------------|---------------------------|------------------------------|-------------------------------|
| | Hooke&Jeeves (Multistart) | Hybrid Approach (EMOA + H&J) | Hybrid Approach (pessimistic) |
| 40° | 0.008 | 0.350 | 0.212 |
| 60° | 0.008 | 0.500 | 0.303 |
| 90° | 0.028 | 0.756 | 0.387 |

Observations: The runtime of the Hooke/Jeeves method significantly depends on the solution finally obtained (see table VIII). This behavior is caused by the effective stopping criterion for runs heading to a trivial solution. The effort required to locate a nontrivial solution to the desired accuracy is a order of magnitude higher. An effective stopping criterion for runs to nontrivial solutions might be useful. For example, we can stop such a run if we recognize that it is heading to a nontrivial solution that has been accurately localized already.

Discussion: Using the multiobjective approach as a preprocessing step to obtain promising starting points for some

TABLE VIII
AVERAGE NUMBER OF FUNCTION EVALUATIONS OF HYBRID ALGORITHM

| temperature (in Celsius) | #FEs to trivial solution | #FEs to nontrivial solution | #FEs until stuck |
|--------------------------|--------------------------|-----------------------------|------------------|
| 40° | 199.0 | 2018.5 | 2101.9 |
| 60° | 195.0 | 2173.5 | 2384.6 |
| 90° | 138.0 | 1476.5 | 4313.6 |

local search method has boost the success probability from 2.8 % to 75.6 % (or 38.7 %, depending on the point of view) when considering the test problem with 90° Celsius. Needless to say, this is a significant improvement. For lower temperatures the success probability does not increase that dramatically. This behavior is probably caused by the location of the optima: The lower the temperature, the closer moves the optimum to the border of the search space.

VII. CONCLUSIONS

The phase equilibrium detection problem has revealed the limitations of the standard optimization approaches. Although it is possible to find nontrivial solutions to the problem by multistart techniques the success probabilities for locating a nontrivial solution are too low such that the effort required is prohibitive for practical use. In such a case the integration of domain knowledge into the solution approach is inevitable. Here, we formulated the application-specific knowledge about the process in a second objective function and we used a multiobjective evolutionary algorithm to generate promising starting points for a local search method that is applied in a multistart manner. The experimental results obtained so far clearly indicate a significant improvement with respect to standard approaches. We are confident that the approach presented here can be developed to a technique suited for practical use also for more complicated problems.

ACKNOWLEDGMENT

This work was supported in part by the DFG via the Collaborative Research Center on Computational Intelligence (SFB 531) at the University of Dortmund. Mike Preuss gratefully acknowledges DFG support via project no. 252441, "Mehrkriterielle Struktur- und Parameteroptimierung verfahrenstechnischer Prozesse mit evolutionären Algorithmen am Beispiel gewinnorientierter unscharfer destillativer Trennprozesse".

REFERENCES

- [1] A. A. Törn and A. Žilinskas, *Global Optimization*, ser. Lecture Notes in Computer Science. Springer, 1989, vol. 350.
- [2] D. Beasley, D. R. Bull, and R. R. Martin, "A sequential niche technique for multimodal function optimization," *Evolutionary Computation*, vol. 1, no. 2, pp. 101–125, 1993. [Online]. Available: citeseer.ist.psu.edu/beasley93sequential.html
- [3] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*. Berlin, Heidelberg: Springer, 2003.
- [4] K. A. De Jong, "An analysis of the behavior of a class of genetic adaptive systems." Ph.D. dissertation, University of Michigan, 1975.
- [5] D. E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in *Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application*. Mahwah, NJ, USA: Lawrence Erlbaum Associates, Inc., 1987, pp. 41–49.

- [6] M. Jelasity, "Uego, an abstract niching technique for global optimization," in *PPSN*, ser. Lecture Notes in Computer Science, A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, Eds., vol. 1498. Springer, 1998, pp. 378–387.
- [7] A. Pétrowski, "A clearing procedure as a niching method for genetic algorithms," in *Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC '96)*, Nagoya, T. Fukuda, T. Furuhashi, and D. B. Fogel, Eds. Piscataway NJ: IEEE Press, 1996, pp. 798–803.
- [8] J.-P. Li, M. E. Balazs, G. T. Parks, and P. J. Clarkson, "A species conserving genetic algorithm for multimodal function optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 207–234, 2002.
- [9] F. Streichert, G. Stein, H. Ulmer, and A. Zell, "A clustering based niching method for evolutionary algorithms," in *Genetic and Evolutionary Computation – GECCO-2003*, E. Cantú-Paz, Ed. Berlin: Springer-Verlag, 2003, pp. 644–645.
- [10] M. Tomassini, *Spatially Structured Evolutionary Algorithms Artificial Evolution in Space and Time*, ser. Natural Computing Series. Berlin: Springer, 2005.
- [11] M. Preuss, "Niching Prospects," in *Bioinspired Optimization Methods and their Applications (BIOMA 2006)*, B. Filipic and J. Silc, Eds. Jozef Stefan Institute, Ljubljana, Slovenia, 2006, pp. 25–34.
- [12] F. Oppacher and M. Wineberg, "The shifting balance genetic algorithm: Improving the GA in a dynamic environment," in *Proc. Genetic and Evolutionary Computation Conf. (GECCO 1999)*, Orlando FL, W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, Eds., vol. 1. San Francisco CA: Morgan Kaufmann, 1999, pp. 504–510.
- [13] T. Bartz-Beielstein, *Experimental Research in Evolutionary Computation – The New Experimentalism*, ser. Natural Computing Series. Berlin: Springer, 2006.
- [14] A. Auger and N. Hansen, "Performance Evaluation of an Advanced Local Search Evolutionary Algorithm," in *Proc. 2005 Congress on Evolutionary Computation (CEC'05)*, B. McKay et al., Eds. Piscataway NJ: IEEE Press, 2005.
- [15] G. d. A. E. Prausnitz J. M., Lichtenthaler R. N., *Molecular Thermodynamics of Fluid Phase Equilibria*. New Jersey: Prentice-Hall, 1986.
- [16] S. Behme, "Thermodynamik von Polymersystemen bei hohen Drücken," Dissertation, Technische Universität Berlin, 2000, (in German).
- [17] M. Giacobini, M. Preuss, and M. Tomassini, "Effects of Scale-Free and Small-World Topologies on Binary Coded Self-Adaptive CEA," in *Evolutionary Computation in Combinatorial Optimization, 6th European Conference, EvoCOP 2006, Proceedings*, ser. Lecture Notes in Computer Science. Berlin: Springer, 2006, pp. 86–98.
- [18] M. Preuß, B. Naujoks, and G. Rudolph, "Pareto set and EMOA behavior for simple multimodal multiobjective functions," Universität Dortmund, Interner Bericht des Sonderforschungsbereichs 531 *Computational Intelligence* CI–205/06, April 2006.
- [19] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies: A comprehensive introduction," *Natural Computing*, vol. 1, no. 1, pp. 3–52, 2002.
- [20] T. Bartz-Beielstein and M. Preuss, "Considerations of budget allocation for sequential parameter optimization (spo)," in *Workshop on Empirical Methods for the Analysis of Algorithms, Proceedings*, L. P. et al., Ed., Reykjavik, Iceland, 2006, pp. 35–40.
- [21] M. Emmerich, N. Beume, and B. Naujoks, "An EMO algorithm using the hypervolume measure as selection criterion," in *Evolutionary Multi-Criterion Optimization (EMO)*, C. A. C. Coello et al., Eds. Springer, Berlin, 2005, pp. 62–76.
- [22] R. Hooke and T. A. Jeeves, "Direct search solution of numerical and statistical problems," *Journal of the ACM*, vol. 8, pp. 212–229, 1961.