DOI: 10.15439/2021F60
ISSN 2300-5963 ACSIS, Vol. 25

Dynamic communication topologies for distributed heuristics in energy system optimization algorithms

Stefanie Holly
R&D Division Energy
OFFIS - Institute for Information Technology
Escherweg 2, 26121 Oldenburg, Germany
Email: stefanie.holly@offis.de

Astrid Nieße
R&D Division Energy
OFFIS - Institute for Information Technology
Escherweg 2, 26121 Oldenburg, Germany
Email: astrid.niesse@offis.de

***Abstract*—The communication topology is an essential aspect in designing distributed optimization heuristics. It can influence the exploration and exploitation of the search space and thus the optimization performance in terms of solution quality, convergence speed and collaboration costs – relevant aspects for applications operating critical infrastructure in energy systems. In this work, we present an approach for adapting the communication topology during runtime, based on the principles of simulated annealing. We compare the approach to common static topologies regarding the performance of an exemplary distributed optimization heuristic. Finally, we investigate the correlations between fitness landscape properties and defined performance metrics.**

I. Introduction

DISTRIBUTED heuristics are a promising field for current and future energy systems control and optimization tasks, and have been designed and evaluated in recent years on agent-based systems [1] [2] [3]. While conventional control systems – centralized or hierarchical in their control paradigm – perfectly fit to centralized generation and transmission systems, distributed renewable energy systems show properties that promote the application of distributed optimization systems: First, future energy systems can be regarded as complex systems of systems, sometimes framed as cyber-physical multi-energy systems, coupling communication systems, power, heat and gas systems. The resulting complexity of the solution space is the main motivation for heuristic distributed control and optimization [4]. Second, data availability as needed for centralized control typically is not given for end-user scenarios for privacy or regulatory reasons.

Distributed control and optimization systems often involve multiple energy units that decide locally and communicate with each other to solve global problems. For instance, software agents can represent flexible energy loads that cooperatively aggregate flexibility to provide load dispatch options for balancing markets or congestion management [5] [6] [7].

One major design aspect is the communication topology. This topology - usually modeled as a graph - determines which units exchange data directly. In energy system applications, the communication topology of multi-agent systems is often defined based on the topology of the underlying power grid (see e.g. [8] and [9]). This approach is limited to static topologies, and not reflecting algorithmic aspects. Considerable research has been conducted on distributed optimization in the power grid, in the area of control theory, where systematic mathematical approaches are used to design distributed controllers [10]. However, the type of problems that can be solved with such approaches is limited [11]. Distributed energy resources are very heterogeneous regarding forecast precision and flexibility potential. Agents have to consider more local constraints and be able to react flexibly to their environment. Thus, we consider algorithms that provide a framework for negotiation between different agents, but allow flexible local action. To the best of our knowledge, there has been little research on how the communication topology affects such distributed heuristics or how it can be optimally designed.

In [12] we showed that different communication topologies have an effect on the performance of the reflected algorithm class: Highly meshed topologies converged into good solutions reliably and quickly, but increased communication overhead and premature convergence. In contrast, results for sparsely meshed topologies were much less reliable. In the application domain of energy systems as critical infrastructures, this behavior is highly unwanted. We presume that dynamically adjusting the topology during runtime leads to a beneficial transition of exploration and exploitation of the search space for distributed heuristics.

In this contribution, we evaluate the effect of dynamic communication topology adaptation on a fully distributed optimization heuristic. To ensure scientific comprehensibility and reproducibility, standard optimization problems are taken for an extensive analysis of the approach. Furthermore, we analyse correlations between the performance of the distributed optimization algorithm and the fitness landscape characteristics of these benchmark functions with both static and dynamic overlay topologies using decision trees.

The rest of this contribution is structured as follows: In section II, an overview on the topic of communication topologies for distributed heuristics is presented, motivating the research gap. The dynamic topology adaptation scheme is presented in section III. The metrics used for the fitness landscape analysis are presented in section IV. In section V we set the scene for the experimental setup chosen to analyse the relevant correlations, followed by a discussion in section VI. We conclude our work with an outlook on future research directions.

II. Communication topologies for distributed heuristics

Distributed optimization heuristics are closely related to parallel cooperative metaheuristics [13]. In both types of heuristics, multiple distributed (meta)heuristics are interconnected and exchange information. Therefore, we consider the studies on the influence of exchange topologies in the area of parallel cooperative metaheuristics as relevant related work.

More precisely, we restrict our scope to asynchronous cooperative search strategies, i.e. several solvers run simultaneously (multi-search / distributed on algorithmic level) and cooperate with each other by asynchronously exchanging information. This type of parallel heuristics originates from the field of parallel computing. Therefore, the communication topology was mostly designed with respect to the hardware architecture (considering connections between processing units) leading to hypercube, ring or torus topologies [14]. For island model heuristics, i.e., heuristics where multiple instances of mostly population-based metaheuristics run in parallel exchanging individuals between their islands, a fully meshed topology is often chosen [15]. In addition, information is usually exchanged indirectly via a shared memory.

The effect of communication topologies on the performance of distributed optimization heuristics has been studied especially for the island-model, where the topology is often referred to as the migration topology. In most works, different topologies for a given parallel heuristic are studied on multiple benchmark problems and the topologies are ranked according to the achieved performance, which may involve different aspects [14], [16], [17], [18], [19]. Ruciński et al. investigated the effect of different migration topologies, including ring, cartwheel and hypercube topologies, on the performance of two different parallel global optimization algorithms cooperating via the island model [14]. They evaluated different topologies for both heuristics according to the performance obtained. Since the results varied widely, Ruciński et al. suggested that such studies be conducted in the future for other heuristics and with more problem instances.

Hijaze and Corne [16] analyzed how different topologies affect the performance of an asynchronous distributed evolutionary algorithm (EA). They evaluated the effect of the topologies on the performance of the algorithm using 30-dimensional target functions (Sphere, Rosenbrock, Schwefel, Rastrigin, Griewank, Ackley [20][21]). The success rate in finding the optimum was similar for all topologies, but better than for the standard single population EA (with equal total population size).

In their follow-up work in [17], they introduced an online adaptation of the migration scheme in which the migration probability was adjusted based on the progress of subpopulations on islands. With the adaptive scheme, optimal solutions were regularly found in less time and with a higher success rate, suggesting that a balance between exploration and exploitation can be achieved by dynamically adjusting the migration mechanisms.

In [18], Sanu and Jeyakumar conducted an empirical analysis on the performance of distributed differential evolution (DE) for varying migration topologies. They used various topologies (basic ring and ring variants, star, cartwheel, torus and mesh) and multiple benchmark functions (e.g. Sphere, Schwefel (1,2,3), Rosenbrock, Rastrigin) to investigate the impact of the topologies on the performance of an island model DE. They considered not only the convergence speed and solution quality based metrics, but also the computational effort, i.e., the number of function evaluations. They concluded that no single topology is suitable for all optimization problems and took a first step towards linking characteristics of the search spaces to the performance of the topologies by roughly categorizing the functions (modality and separability) and assigning the best performing topologies in each case.

The presented research can be summarized as follows: First, different communication topologies affect the performance of the various parallel metaheuristics. Second, the notion of performance is mainly limited to the achieved solution quality and convergence speed. Since the studies do not address spatially distributed systems, they usually do not examine the costs of collaboration, especially the resulting message traffic. Third, the design of communication topologies leads to different balances between exploration and exploitation of the search space. In some cases it is investigated how this balance can be improved by adjusting parameters like migration frequency or migration rate. However, the adaptation of the topology has not been treated as a distinct research topic for optimization problems with characteristics as be found in energy system applications. To our knowledge, a systematic approach regarding the above mentioned aspects including an in-depth fitness landscape analysis has not yet been conducted in this field.

III. Dynamic Topology Adaptation

As described in the previous section, the communication topologies of distributed optimization heuristics affect the degree of exploration and exploitation of the search space. Strongly meshed topologies lead to a high amount of information exchange between units (diversification). This leads to a fast convergence but bears the risk of a premature convergence into local optima. In contrast, with sparsely meshed topologies, the individual heuristics can evolve more independently. This leads to an intensified search (exploitation) in some areas of the search space. But in the worst case, these areas can be far away from the global optimum.

Many metaheuristics adjust parameters at runtime to allow a transition from exploration to exploitation. An example of this is the adjustment of the temperature parameter T for simulated annealing (SA) [22].

Since we want to achieve the same effect with dynamic topology adjustment, our approach is based on the principles the cooling process in SA. Just as SA starts with a high temperature, we start with a high number of connections in the communication topology. With cooling down, we reduce the number of connections. To model the cooling process, SA

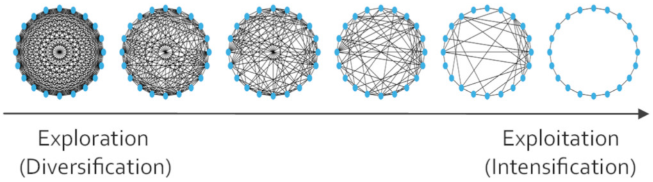


Fig. 1: Dynamic topology adaptation

uses a so-called cooling schedule. We therefore determine a so called "removal schedule". Table I shows the details of the specified analogy. The approach presented here starts with a fully meshed topology, transitions to small world intermediate stages by removing edges, and ends with a ring to exploit the most promising regions in the solution space. Fig. 1 illustrates this process.

In order to model this transition, more specifications are necessary. Let $G = (V, E)$ denote the bidirectional graph that represents the communication topology. V is the set of nodes, where each node is assigned to an agent and thus to one part of the distributed solver. E is the set of edges. An edge between two nodes indicates direct communication between the two agents assigned to the nodes. Agents can pass on information from their neighbors to other neighbors, which means that there is also indirect communication between unconnected nodes. The communication topology thus regulates the information dissemination in the distributed system.

A cooling schedule for SA is determined by the initial temperature T_0, the equilibrium state, i.e. the criterion that controls when the transition to the next temperature level occurs, and the cooling itself. Since the communication topology will start with a fully meshed bidirectional graph, the number of edges is defined as $|E_0| = \frac{n \cdot (n-1)}{2}$. Geometric functions are particularly popular to model cooling, whereas logarithmic functions are considered too slow for practical application, although they theoretically converge to a global optimum [13].

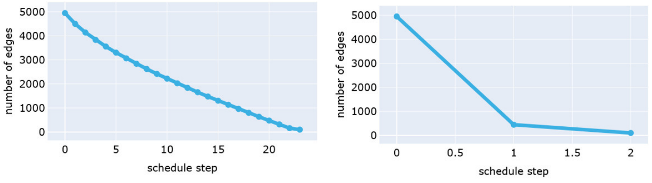
Considering that the initial number of edges is much smaller than usual starting temperatures, a slower reduction seems appropriate. A combination of linear, geometric and logarithmic reduction functions was chosen. Equation 1 displays the function that determines the number of edges δ at each schedule step.

$$\delta_{i+1} = |E_i| - \frac{|E_0|}{log(i)} \cdot \alpha, \quad with \;\; \alpha \in]0, 1] \tag{1}$$

The index i represents the index of the step in the reduction schedule. δ_i determines the number of edges that should remain in the reduction schedule step i. Thus the new communication topology graph is constructed such that

$$G_{i+1} = (V, E_{i+1}), \quad with \quad |E_{i+1}| = \delta_{i+1} \tag{2}$$

The parameter α controls how many steps the removal schedule includes. If it is close to 0, only a few edges are removed in each step, leading to a slow decrease of connectivity. If it equals 1, the number of edges is reduced in large steps, which leads to a rapid decrease in connectivity. The edges that are removed are selected randomly, ensuring that the final ring topology remains. The figures 2a and 2b show how the parameter α influences the granularity of the edge reduction schedule.



(a) $\alpha = 0.1$

(b) $\alpha = 1$

Fig. 2: Reduction schedule with 100 agents and different α

Finally, the transition criterion from one topology to the next must be specified. It defines when a transition from one schedule step to the next occurs. Common conditions for a temperature adaptation in SA are counting of iterations, acceptances, rejections or a combination of these [13]. In this case, a simple approach is used, where the number of local searches (equivalent to the number of iterations) since the last transition is counted. As soon as this number becomes larger than n, the topology is adjusted. Since the starting topology is a complete graph all agents start at the same time and each unit performs its local search at least once before the first adjustment. In later stages, this procedure can lead to some agents optimizing locally multiple times between topology adjustments and others not optimizing at all. This rapid transition was chosen since it showed the best results in preliminary experiments.

To evaluate this dynamic approach, we perform a systematic comparison of the performance of the approach in different parameterizations and several static topologies. For this purpose, we use a set of well-known benchmark functions. However, instead of considering them separately, we perform fitness landscape analysis to match the individual difficulties of the problems with the performance of the topologies and find possible correlations. Consequently, we first explain the metrics used in the following section before moving on to the experimental setup and the evaluation of the results.

IV. Fitness Landscape Analysis

As the no free lunch theorem states, no optimization heuristic can be superior to all others without regard to the problem [23]. Different communication topologies significantly influence the information propagation in a heuristic and thus the resulting optimization process. Consequently, for different problems, different topologies presumably lead to more advantageous behavior. We use various fitness landscape metrics to classify the objective functions to examine the relationships between problem characteristics and the performance of different topologies.

A fitness landscape, as presented e.g. in [24], is defined by the search space X, containing all possible solutions of the problem, connected according to a defined distance measure,

TABLE I: Comparison of the modeling of SA cooling and the connection reduction of the communication topology

	cooling schedule	removal schedule
definition of	temperature for each step of the SA algorithm	number of edges in the communication topology for each step
initialization parameter	T_0: initial temperature	$\|E_0\| = \delta_0$: initial number of edges
equilibrium state	number of iterations at a temperature	number of local optimizations at a topology configuration
adaptation	cooling: decrease of the temperature	decrease of edges in the communication topology

and the fitness function $f : X \to R$. A fitness landscape for a continuous problem, often uses euclidean distance measures and thus can be described as a landscape with the search space as the bottom floor and the landscape surface being elevated according to the values of the fitness function f [13]. Analogous to a geographical landscape, fitness landscapes can have peaks, valleys, plains, canyons, cliffs, plateaus, basins, etc. Investigating these landscape characteristics provides clues as to how difficult it is to find an optimum, i.e., the highest mountain peak (maximization) or the lowest valley (minimization). Fig. 3 shows 3-D plots of benchmark functions demonstrating some manifestations of such landscape features.

Various metrics have been proposed in literature. We limit the scope to metrics that can be used for continuous search spaces and that can be normalized, since we want to be able to compare different functions. In [24], Sun et al. distinguished some basic features of fitness landscapes and argued that for proper characterization, these features must be covered when selecting a set of metrics. These features include:

- Dimensionality
- Separability
- Ruggedness, Smoothness and Neutrality
- Modality
- Deception and Evolvability

The dimensionality of the problems is a selectable parameter in the experimental setup and therefore known. Furthermore, separability is a well-known property of the benchmark functions. For the other characteristics mentioned, suitable metrics must be selected. We discuss our choice in the following.

A. Ruggedness, smoothness and neutrality

The characteristics of ruggedness and smoothness concern the quantity and distribution of the local optima in the search space. The fitness differences in a neighborhood can be large (rugged), small (smooth), or barely present (neutral). Each of these surface shapes presents different challenges for optimization algorithms. In [25] Malan and Engelbrecht adapted the entropy based measure for ruggedness that was first proposed by Vassilev et al. [26] for continuous fitness landscapes. The information theoretic technique is based on a random walk through the search space. The random walk is represented as a string with respect to the information stability measure ϵ. If the magnitude of the difference between two fitness values is less than ϵ, they are considered to be equivalent. The string representation is obtained as follows:

$$S_i(\epsilon) = \begin{cases} -1, & \text{if } f_i - f_{i-1} < -\epsilon \\ 0, & \text{if } |f_i - f_{i-1}| \leq \epsilon \\ 1, & \text{if } f_i - f_{i-1} > \epsilon \end{cases} \tag{3}$$

The entropy value is calculated for the resulting string $S_i(\epsilon)$ according to Equation 4:

$$H(\epsilon) = -\sum_{p \neq q} P_{[pq]} log_6 P_{[pq]} \tag{4}$$

where $P_{[pq]}$ is the frequency of occurrence of the block $[pq]$ in $S_i(\epsilon)$ with $p, q \in \{-1, 0, 1\}$. This entropy is calculated with various ϵ between 0 and ϵ_{max}, which is the value at which the resulting string consists only of zeros. The maximum of all attained entropy measures is taken as the final result [25]. This entropy measure reflects the information content of the random walk. This is naturally high for a rugged landscape, whereas a smoother landscape has a smaller entropy value.

Depending on the stepsize of the random walk, ruggedness can be viewed on different scales. We follow Malan's suggestion and compute the metric once based on random walks with a maximum step size of 1% of the search space and once with 10%. The resulting metrics FEM_{micro} and respectively FEM_{macro} (First Entropic measure as defined by [27]) are values in $[0, 1]$ and reflect the relationship between ruggedness and neutrality on micro an macro scale.

Similarly to the FEM a second entropy measure can be calculated that estimates the smoothness of the function rather than the ruggedness. Based on the string $S_i(\epsilon)$ the entropy of smooth blocks, i.e. two consecutive characters with the same sign, is calculated as follows:

$$h(\epsilon) = -\sum_{p = q} P_{[pq]} log_3 P_{[pq]} \tag{5}$$

We apply the same approach as for FEM by calculating $h(\epsilon)$ with different values for ϵ and keeping the maximum of all entropy values as SEM (Second Entropy Measure). The SEM_{micro} and SEM_{macro} are again values in $[0, 1]$, but refer to the interaction of smoothness and neutrality of the landscape [27].

B. Modality

The *modality* of a function corresponds to the number of local optima. Modality and ruggedness are closely related, since a rugged landscape may also include many local optima. In [26], Vassilev et al. also proposed a metric to quantify the modality of the random walk encoded by Equation 3. A new string $S'(\epsilon)$ is constructed by removing all zeros from $S(\epsilon)$ and reducing sequences of equal characters to one character. Thus, $S'(\epsilon)$ contains only information that is essential with respect to modality. The resulting modality measure is called partial information content (PIC) and is given by

$$PIC(\epsilon) = \frac{\mu}{n} \tag{6}$$

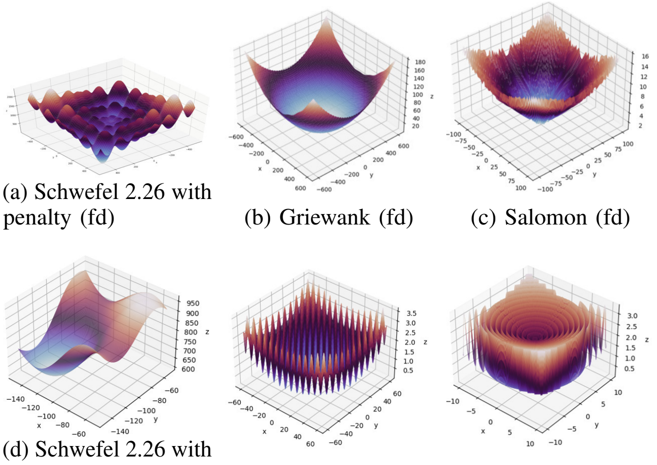
where n is the length of $S(\epsilon)$ and μ the length of $S'(\epsilon)$. If the random walk encountered a landscape with high modality, the length of $S'(\epsilon)$ is almost the same as that of $S(\epsilon)$, resulting in $PIC(\epsilon)$ being close to or equal to one. If the path is flat or only leading in one direction $PIC(\epsilon)$ tends to or is equal to zero. For PIC, we use the same procedure as for FEM and SEM and perform random walks with varying step sizes to look at modality at different scales. Accordingly, we use PIC_{micro} and PIC_{macro} as metrics.

C. Deception and evolvability

A *deceptive* landscape provides information that can guide an optimization algorithm away from the global optimum, towards local optima. Which properties of a function are deceptive depends on the algorithm - in the case of distributed algorithms, perhaps also on the communication topology. One possibly deceiving characteristic is the presence of funnels. A funnel is a cluster of local optima that forms a global basin shape [28]. The dispersion metric of Lunacek et al. [29] provides insight into the global topology of fitness functions and thus indirectly allows estimation of the presence of funnels. In [28], Malan and Engelbrecht proposed a normalized version to allow comparison of functions with different domain sizes. To compute the dispersion metric, a random sample $\mathscr{S}$ of length n is drawn that is uniformly distributed over the search space. From this sample $\mathscr{S}$, a subset $\mathscr{S}^*$ is determined that contains the best points by fitness values. To make functions with different domain sizes comparable, the position vectors of $\mathscr{S}^*$ are normalized in such a way that the search space is scaled to [0,1]. In addition, a comparison sample $\mathscr{C}$ also of size is sampled uniformly across the search space. Let $disp(\mathscr{S})$ be the average pairwise distance between normalized positions in the sample $\mathscr{S}$. Then the dispersion metric DM is defined as follows:

$$DM = disp(\mathscr{S}^*) - disp(\mathscr{C}) \tag{7}$$

Fig. 3: Selection of benchmark functions as 3-D plot on the full domain (fd) or on a 1 % section of the domain (1%d)



(a) Schwefel 2.26 with penalty (fd)

(b) Griewank (fd)

(c) Salomon (fd)

(d) Schwefel 2.26 with penalty (1%d)

(e) Griewank (1%d)

(f) Salomon (1%d)

Thus, the metric quantifies how far points with high fitness values are away from each other compared to a large uniform random sample. It yields values in the range of [-1,1]. A low value ($DM < 0$) indicates a single funnel landscape with an underlying unimodal structure. A high value ($DM > 0$) indicates a multi-funnel landscape and underlying multimodal structure. Fig. 3 shows 3-d plots of three benchmark functions once in their full domain and once in a one-percent section of their domain, respectively. The penalized Schwefel 2.26 function is very rugged on the macro scale (Fig. 3a), but much less so on the micro scale (Fig. 3d). The function has a high value in the dispersion metric, which is also consistent with its global multi funnel shape. In contrast, both Griewank and Salomon have a small value in the dispersion metric, corresponding to their global single funnel shapes. While Salomon is highly rugged on macro and micro scale with a slight increase on micro scale, Griewank is much more rugged on micro scale than on the macro level. This effect actually decreases in higher dimensions, making the function "easier" to solve [30].

Table II shows an overview of the applied fitness landscape metrics, including a brief summary. The results obtained for the metrics for the benchmark functions are summarized in a dedicated repository[1].

dimension	dimension, here also equal to number of agents
separability	boolean that indicates if variables of a function are independent
FEM_{macro} FEM_{micro}	first entropy based measure of *ruggedness* on macro and micro scale
SEM_{macro} SEM_{micro}	second entropy based measure of *smoothness* on macro and micro scale
PIC_{micro} PIC_{macro}	measure of partial information content concerning *modality* on macro and micro scale
DM	dispersion metric that quantifies distances between good solutions and thus indicates the presence of funnels

TABLE II: Overview of applied metrics

V. Methodology

The goal of the experimental study is to investigate if a dynamic topology adaptation approach outperforms static topologies. We use multiple benchmark functions, all scalable and multi-modal, and the distributed optimization heuristic presented in subsection V-A.

Furthermore, we examine correlations between the properties of the benchmark functions and the performance of different

[1] https://github.com/sholly-offis/Deta

topologies in the defined performance dimensions. In doing so, we also consider different parameterizations of the dynamic approach. Thereby, we hope to find clues that will help in the further development of the dynamic approach and ultimately lead to the parameterization of dynamic topology adaptation in such a way that it provides a tailored solution to a problem. In the following we first give a short introduction to the chosen example heuristic and then elaborate on the experimental setup.

A. Distributed Optimization Algorithm

COHDA is a combinatorial optimization heuristic for distributed agents, and was developed for the self-organized scheduling of distributed energy resources in virtual power plants [31]. The heuristic can be classified as a system realizing a gossiping protocol based on strictly defined communication and knowledge integration rules. In [32], Bremer et al. adapted COHDA to find the global minimum of a real valued objective function. An agent a_i is responsible for only one value x_i from a continuous search space. It performs its local optimization to minimize the global objective function, by adapting its own choice of x_i while considering the choices of other agents $x_j, j \neq i$ as temporarily fixed. Agents send update messages to their neighbors - as defined by the communication topology - to pass on new information from their neighbors or to inform about their own changes in the value selection. Depending on well-defined convergence conditions, COHDA has been proven to always converge at least to a local optimum [31]. Parts of these conditions are related to the topology, and thus have to be reflected here: The chosen topology has to be connected, irreflexive, and symmetric. As a consequence, the topology adaptation developed in this work has to guarantee these characteristics in all intermediate stages to not sacrifice convergence.

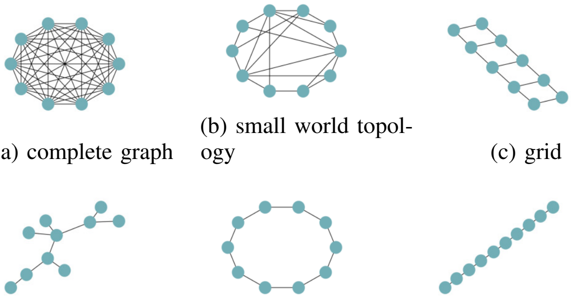
B. Experimental Setup

The experiments are preformed with agent systems in two different sizes, namely 50 and 100. In the applied setup, the system size is equal to the dimension of the objective functions, as each agent is responsible for choosing one solution variable. A set of 13 different benchmark functions is used as underlying synthetical problem instances. They include Ackley, a scalable version of Eggholder, Griewank, Happy Cat [33], Rana, Rastrigin [21], Rosenbrock, Salomon, Schaffer F6, Qing, Schwefel 2.26 and a penalized Version of Schwefel 2.2.6 which was introduced in [12]. Unless otherwise stated, the definitions are taken from [20]. Definitions, domains and global minima of the benchmark functions are listed in a dedicated repository[2].

The search process in the solution space defined by each benchmark function is performed using COHDA and different topologies. This involves complete graphs, ring-, tree-, small-world-, path-, and grid- topologies. Fig. 4 shows these topology types for 10 agents.

For each system size (50 and 100) each topology type is created, including a randomized setting with 5 different seeds and optimization runs and 10 different starting seeds. As a result of this setting, 50 different setups are examined for each combination of the number of agents and topology type. The same is done for the dynamic approach with different values for $\alpha \in [0.1, 0.3, 0.7, 1]$. [3] A total of 1000 optimization runs were performed per benchmark function (600 with static topologies and 400 with the dynamic approach). Therefore, the following evaluations are based on a total of 13,000 optimization runs.

Fig. 4: Overview of reflected static communication topologies


(a) complete graph
(b) small world topology
(c) grid
(d) random tree
(e) ring
(f) path graph

To compare the performance, four performance dimensions are evaluated:

- solution quality: error measure
- speed of convergence: time required to converge
- communication traffic: number of messages exchanged between agents
- computational effort: number of local searches preformed by all agents

Each of the performance dimensions is normalized per system size and benchmark function. We differentiate the four performance dimensions when determining the best topology for a benchmark function. To be the best topology in a performance dimension, a topology must achieve the lowest possible values for this measure. Therefore, we consider the mean in each case. For some topologies however, the spread of values is extremely large, so that despite a good mean value, the risk of an unfavorable outlier is much higher than for other topologies. Again, in the application domain of critical infrastructure optimization, this is unfavorable. The sum of mean and standard deviation is used as an additional measure (using the notion *mps* – mean plus standard deviation), to reflect this aspect.

To relate the performance of the topologies to the characteristics of the benchmark problems, fitness landscape metrics described in section IV are computed for each benchmark function using random walks of length 1000 and the mean of 30 runs is taken as measure as proposed by [28]. Finally, we train decision trees using the CART algorithm [34] implemented by scikit-learn [35]. We use them to determine which metrics can be employed to distinguish benchmark functions and assign them to the best-performing topologies.

[2] https://github.com/sholly-offis/Deta

[3] $\alpha = 0.5$ was discarded in this presentation for reasons of brevity as it was not superior to the other values.

VI. Results and Discussion

In the following, we first examine the differences in the solution quality, using decision trees generated on the basis of the fitness landscape analysis. Additionally, we want to identify correlations between the properties of the fitness landscapes and the performance of different topologies and parameterizations of the dynamic adaptation approach.

In the second part of the evaluation, we examine the other performance dimensions. Since these cannot be reasonably analyzed on their own, individual functions are analyzed as representatives in order to investigate the overall performance of the different topologies and the trade-offs between the performance dimensions.

A. Decision tree-based performance analysis

To determine which topology is best for a given function and dimension for a performance indicator, minimum mean and mps are evaluated in each case. For simplicity, additional collective topology categories were introduced when several topologies were equally good. All topologies whose mean and mps do not deviate more than 1% from the minimum values are assigned to a best list. If the best list contains more than three entries, single topologies are replaced by collective categories. The label *highly meshed* is assigned if only highly meshed topologies are in the given class (complete, small world, grid, dynamic). *Weakly meshed* accordingly summarizes ring, tree, and path graph topologies. Other combinations are labeled as *various*. Note that if $dynamic$ is displayed without a value for α, multiple values for α have performed equally well. In order to classify what a decision boundary for a particular landscape metric means, the distribution of the metric in the set of benchmark functions must be considered. For this purpose, Table III shows the respective minima, maxima, mean values and the limits for the upper and lower quartiles.

	min	Q_1	median/Q_2	Q_3	max
DM	-0.39	-0.28	-0.2	-0.01	0.05
FEM_{macro}	0.69	0.72	0.87	0.88	0.89
FEM_{micro}	0.2	0.4	0.66	0.76	0.9
SEM_{macro}	0.51	0.53	0.57	0.68	0.74
SEM_{micro}	0.52	0.65	0.73	0.78	0.8
PIC_{macro}	0.35	0.38	0.61	0.66	0.7
PIC_{micro}	0.07	0.14	0.32	0.46	0.74

TABLE III: Distribution of fitness landscape metrics

First, we focus on the achieved solution quality in terms of the normalized error. Fig. 5 shows the resulting decision tree as sankey diagram for the mean error and Fig. 6 the tree illustrating the error mps. The thickness of the branches indicates the number of benchmark functions that belong to a path. On the right, the best topologies or topology categories are shown with the functions they contain. To facilitate the discussion of the results, individual end nodes or clusters of end nodes were numbered. These are discussed below.

1) functions with below average micro ruggedness (FEM_{micro}) and small dispersion (DM), i.e. good points in the search space are close to each other; These function are easy to solve, thus many topologies provide good results
2) Sargan and Happy Cat function; The Sargan function has the highest values for macro smoothness (SEM_{macro}). The Happy Cat function is also very smooth. For both functions complete graph and the dynamic approach perform similarly, while all other topologies rank far behind.
3) functions with below average micro ruggedness but higher dispersion; slightly more difficult than (1), thus weakly meshed topologies are outperformed.
4) functions with above average micro ruggedness (FEM_{micro}), above average macro smoothness (SEM_{macro}) and low to medium modality on macro scale (PIC_{macro}) are solved best by $dynamic_{\alpha=1}$ topology. Solomon (50) has a very high micro ruggedness. If only the mean error is considered, the ring topology performs best. However, if the reliability is also considered (see Fig. 6), the $dynamic_{\alpha=1}$ topology performs better. The ring topology is obviously very well suited for the intensive exploitation of landscape areas with high micro ruggedness and lower macro smoothness, but an initial exploration push, as in the dynamic approach, helps to guide the search more reliably into good regions.
5) functions with above average micro ruggedness and macro smoothness in the lower quartile; dynamic topology adaptation with different values for α achieves the best results there;
6) functions with above average micro ruggedness but very low smoothness on micro scale but not on macro scale; Considering only the mean error, the grid and tree topologies perform best, but their advantage is very small, so for the mps value for the Ackley function, many topologies perform equally well and Eggholder (100) is best and most reliably optimized with the dynamic topology with $\alpha = 0.7$.

Note that functions for which $dynamic_{\alpha=1}$ topology performs best, have the highest values for the dispersion metric. A DM larger than -0.037 seems to be a good indicator for a class of functions with multi-funnel shapes for which the dynamic approach with large values for α achieves the best results. Fitness landscapes with high dispersion, high modality, high ruggedness and low smoothness less strongly meshed topologies are advantageous, which often makes the dynamic approach with large values for α favorable. For landscapes with lower modality, less dispersion, a less rugged surface, and more smooth sections, more strongly meshed topologies tend to have an advantage, and the dynamic approach with small values for α also often performs well. More analysis on this will be needed in future work.

B. Examination of further performance dimensions

For the analysis of the other performance dimensions, an isolated consideration of the decision trees is not suitable, since there are usually correlations between good and poor

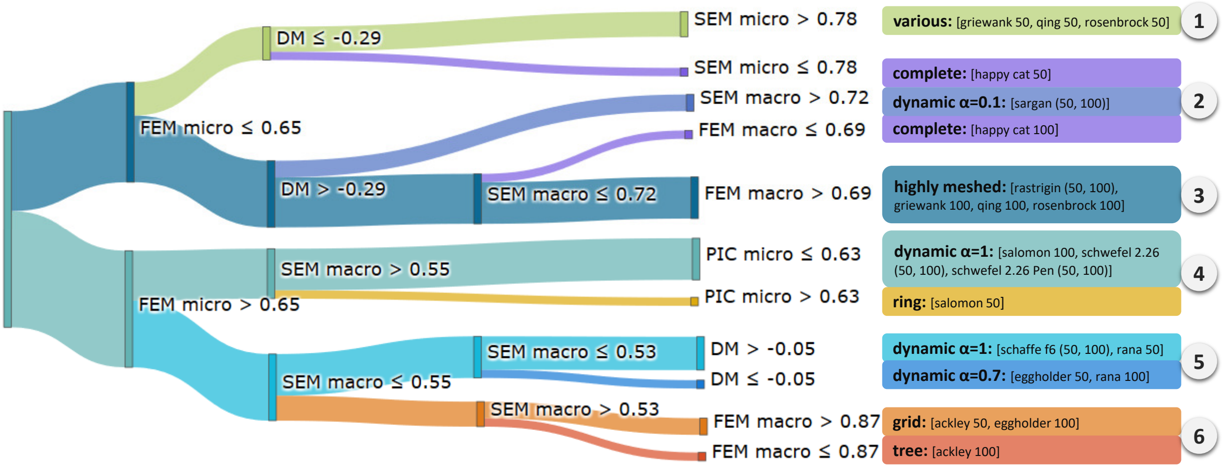

DM ≤ -0.29
FEM micro ≤ 0.65
DM > -0.29
SEM macro ≤ 0.72
FEM micro > 0.65
SEM macro > 0.55
SEM macro ≤ 0.55
SEM macro ≤ 0.53
SEM macro > 0.53
SEM micro > 0.78
SEM micro ≤ 0.78
SEM macro > 0.72
FEM macro ≤ 0.69
FEM macro > 0.69
PIC micro ≤ 0.63
PIC micro > 0.63
DM > -0.05
DM ≤ -0.05
FEM macro > 0.87
FEM macro ≤ 0.87
various: [griewank 50, qing 50, rosenbrock 50]
1
complete: [happy cat 50]
dynamic α=0.1: [sargan (50, 100)]
complete: [happy cat 100]
2
highly meshed: [rastrigin (50, 100), griewank 100, qing 100, rosenbrock 100]
3
dynamic α=1: [salomon 100, schwefel 2.26 (50, 100), schwefel 2.26 Pen (50, 100)]
ring: [salomon 50]
4
dynamic α=1: [schaffe f6 (50, 100), rana 50]
dynamic α=0.7: [eggholder 50, rana 100]
5
grid: [ackley 50, eggholder 100]
tree: [ackley 100]
6

Fig. 5: Decision tree for mean error

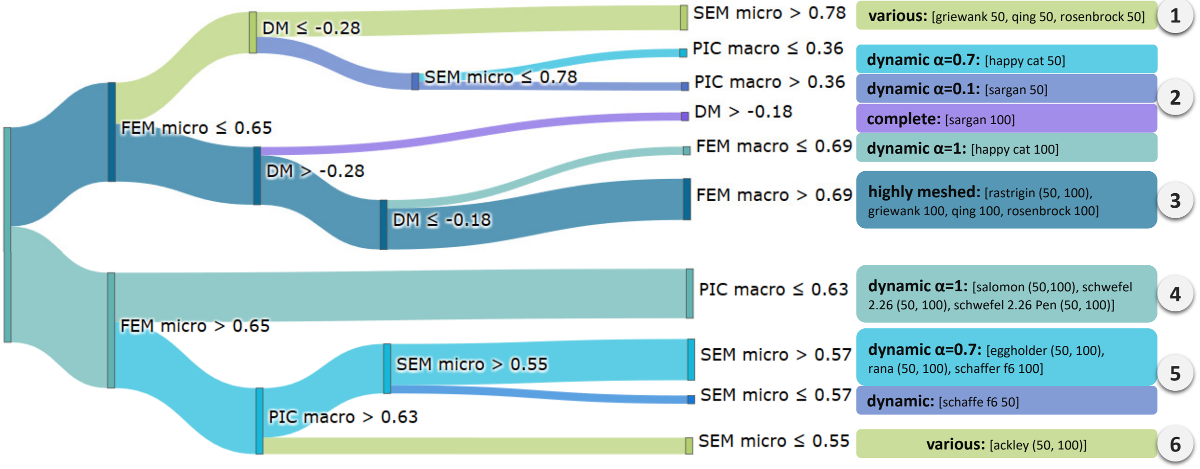

DM ≤ -0.28
SEM micro ≤ 0.78
FEM micro ≤ 0.65
DM > -0.28
DM ≤ -0.18
FEM micro > 0.65
SEM micro > 0.55
PIC macro > 0.63
SEM micro > 0.78
PIC macro ≤ 0.36
PIC macro > 0.36
DM > -0.18
FEM macro ≤ 0.69
FEM macro > 0.69
PIC macro ≤ 0.63
SEM micro > 0.57
SEM micro ≤ 0.57
SEM micro ≤ 0.55
various: [griewank 50, qing 50, rosenbrock 50]
1
dynamic α=0.7: [happy cat 50]
dynamic α=0.1: [sargan 50]
complete: [sargan 100]
dynamic α=1: [happy cat 100]
2
highly meshed: [rastrigin (50, 100), griewank 100, qing 100, rosenbrock 100]
3
dynamic α=1: [salomon (50,100), schwefel 2.26 (50, 100), schwefel 2.26 Pen (50, 100)]
4
dynamic α=0.7: [eggholder (50, 100), rana (50, 100), schaffer f6 100]
dynamic: [schaffe f6 50]
5
various: [ackley (50, 100)]
6

Fig. 6: Decision tree for mps error

performance in several dimensions. For example, the heuristic may have terminated after a short time, but converged to a local optimum. Or only very few messages are sent, but only small parts of the search space are explored, which makes the solution quality a matter of luck. Therefore, we examine the results for the penalized Schwefel 2.26 function, Rana, and Griewank in more detail below.

Fig. 7 shows the results obtained for Schwefel 2.26 with penalty for a system size of 100 agents. Each of the four performance dimensions considered is presented in a separate row. For each static topology and the dynamic approach with $\alpha = 1$ a violin plot is displayed, which shows the distribution with an internal box plot surrounded by a density plot. The top row shows the distributions of the normalized errors. For this function, the dynamic approach performs with the highest solution quality in terms of mean value and outliers. Compared to the strongly meshed topologies (complete, small world and grid graph), the computational effort depicted in row two of Fig. 7 is slightly increased. This seems reasonable, since the topology converges to a ring topology after a short time and therefore performs more computations (with the design objective of these computations being in a promising region of the search space). A similar picture is obtained for the emerging communication traffic, row three in Fig. 7. However, the complete graph generates much more communication effort. In terms of convergence time, the dynamic approach shows slower convergence compared to the topologies that are consistently strongly meshed.

For Rana, the overall impression is similar to Schwefel 2.26 with penalty. The dynamic approach usually performs best and thus needs more time, computational effort and message exchange than other strongly meshed topologies. If the main concern is convergence speed or low communication traffic, the small-word or grid topology are preferable as they provide a good trade-off. For functions with lower modality, dispersion, and ruggedness, such as Rastrigin or Griewank, where many topologies succeed in finding the global optimum, the complete and dynamic graphs usually converge the fastest and have the lowest computational cost. However, they also generate the largest amount of messages, with the dynamic approach

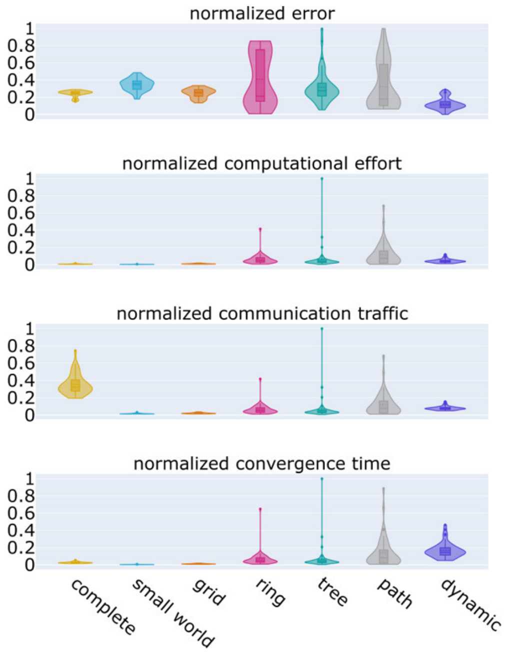
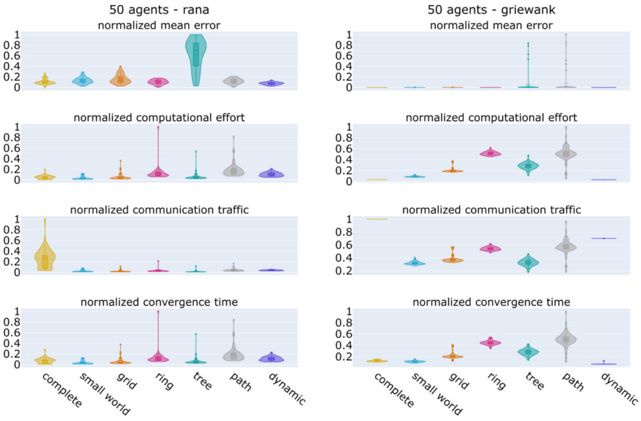


Fig. 7: Results for penalized Schwefel 2.26 (100 agents, $dynamic_{\alpha=1}$)

Fig. 8: Performance results for Rana and Griewank (50 agents, $dynamic_{\alpha=1}$)



(a) Rana

(b) Griewank

performing slightly better. Fig. 8b shows these relationships exemplary for the Griewank function.

To sum up the presented results: In our evaluations, easy to explore fitness landscapes, i.e. functions with uni-modal single funnel shapes and a hardly rugged surface, an arbitrary fairly meshed topology leads reliably to the global optimum. Depending on whether lower message volume or lower computational costs are preferred, small world or the dynamic approach with large α are sufficient. For fitness landscapes with multi-modal and multi-funnel shapes, the dynamic approach with a fast reduction schedule shows advantages regarding solution quality with moderate resource consumption.

VII. Conclusion and Outlook

Distributed optimization heuristics are a suitable approach to handle the increased complexity in cyber-physical multi-energy systems. The communication between the distributed entities must be carefully designed to ensure reliable behavior of a heuristic and thus its suitability for applications in critical infrastructures like energy systems. The communication topology defines which entities exchange information and therefore affects solution quality, convergence speed and collaboration costs. A dynamic approach to topology adaptation during runtime has been presented, based on the principles of simulated annealing. This approach was evaluated against several static topologies using a distributed optimization heuristic to optimize a set of well-known benchmark functions. In addition, we conducted a fitness landscape analysis and trained decision trees to derive correlations between problem properties and performance of the communication topologies.

The main findings can be summarized as follows:

- Functions with small ruggedness at micro level can be considered as "simple". Moreover, if they have a very small DM, i.e., a unimodal shape, many topologies will find the global optimum. Otherwise, more meshed topologies are advantageous.
- When low micro ruggedness is combined with high smoothness or very low macro ruggedness, strongly meshed topologies such as the full graph or the dynamic approach with small values for α, i.e., slowly decreasing connectivity, outperform other topologies.
- The dynamic approach with large values for α excels in providing superior solution quality for functions with high dispersion and thus a difficult multi-funnel landscape. With respect to the cost of cooperation, the dynamic approach is competitive for both easy and hard to explore problems.

Regarding the application domain of distributed control in cyber-physical energy systems, search spaces will have to be analyzed in detail. Given the complexity and non-linearity of the resulting system of systems though, high modality and rugged surfaces seem to be characteristic for many use cases with distributed energy resources, controllable loads and storage systems [36]. So far, communication topologies for distributed heuristics have been selected mostly independently from solution space characteristics. Our work is a first step towards a more systematical selection and dynamic adaptation of communication topologies in this regard.

In future work, we plan to further enhance the dynamic approach. One aspect of this is a intelligent selection of the connections to be removed, by analysing the graph-theoretical characteristics of the intermediate topologies. Furthermore, non-monotonic reduction schedules could be beneficial for some problems, i.e., schedules in which the number of edges can also increase again under certain conditions. The starting topology may be varied as well, since starting with complete graphs is not always advantageous. Overall, with all these enhancements, we aim to be able to optimally select these parameters in the future depending on the problem characteristics and prioritization of the performance dimensions. The resulting parametrizable dynamic topology adaptation will be evaluated on real world problems in the energy domain.

Acknowledgment

We would like to express our sincere gratitude to our college Martin Tröschel, who provided us with advice and support, especially in all data science aspects.

References

[1] P. Vrba, V. Mařík, P. Siano, P. Leitão, G. Zhabelova, V. Vyatkin, and T. Strasser, "A Review of Agent and Service-Oriented Concepts Applied to Intelligent Energy Systems," *IEEE Transactions of Industrial Informatics*, vol. 10, no. 3, pp. 1890–1903, 2014.

[2] S. Ramchurn, P. Vytelingum, A. Rogers, and N. Jennings, "Agent-based homeostatic control for green energy in the smart grid," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, 2011. [Online]. Available: http://dx.doi.org/10.1145/1989734.1989739

[3] M. Sonnenschein, C. Hinrichs, A. Nieße, and U. Vogel, "Supporting Renewable Power Supply through Distributed Coordination of Energy Resources," in *ICT Innovations for Sustainability*, 1st ed., L. M. Hilty and B. Aebischer, Eds. Berlin: Springer, Cham, 2015, vol. 1, pp. 387–404.

[4] A. Nieße, M. Tröschel, and M. Sonnenschein, "Designing Dependable and Sustainable Smart Grids – How to Apply Algorithm Engineering to Distributed Control in Power Systems," *Environmental Modelling & Software*, 2013.

[5] J. Hu, A. Saleem, S. You, L. Nordström, M. Lind, and J. Østergaard, "A multi-agent system for distribution grid congestion management with electric vehicles," *Engineering Applications of Artificial Intelligence*, vol. 38, pp. 45–58, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197614002577

[6] J. Lai, X. Lu, A. Monti, and R. W. de Doncker, "Agent-based voltage regulation scheme for active distributed networks under distributed quantized communication," in *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, 2019, pp. 5941–5946.

[7] A. Nieße, N. Ihle, S. Balduin, M. Postina, M. Tröschel, and S. Lehnhoff, "Distributed ledger technology for fully automated congestion management," *Energy Informatics*, vol. 1, no. 1, p. 22, 2018. [Online]. Available: https://doi.org/10.1186/s42162-018-0033-3

[8] K. Kok, C. Warmer, R. Kamphuis, P. Mellstrand, and R. Gustavsson, "Distributed Control in the Electricity Infrastructure," in *Proceedings of the International Conference on Future Power Systems*, 2005.

[9] S. Lehnhoff, O. Krause, C. Rehtanz, and H. F. Wedde, "Distributed Autonomous Power Management," *at - Automatisierungstechnik*, vol. 3, pp. 167 – 179, 2011.

[10] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A survey of distributed optimization and control algorithms for electric power systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.

[11] Y. Rizk, M. Awad, and E. W. Tunstel, "Decision making in multiagent systems: A survey," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 3, pp. 514–529, 2018.

[12] S. Holly and A. Nieße, "On the effects of communication topologies on the performance of distributed optimization heuristics in smart grids," in *INFORMATIK 2020*, R. H. Reussner, A. Koziolek, and R. Heinrich, Eds. Gesellschaft für Informatik, Bonn, 2021, pp. 783–794.

[13] E.-G. Talbi, *Metaheuristics: From Design to Implementation*. John Wiley & Sons, 2009, vol. 74.

[14] M. Ruciński, D. Izzo, and F. Biscani, "On the impact of the migration topology on the island model," *Parallel Computing*, vol. 36, no. 10-11, pp. 555–571, 2010.

[15] T. Crainic, "Parallel metaheuristics and cooperative search," in *Handbook of Metaheuristics*. Springer, 2019, pp. 419–451.

[16] M. Hijaze and D. Corne, "An investigation of topologies and migration schemes for asynchronous distributed evolutionary algorithms," in *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*. IEEE, 2009, pp. 636–641.

[17] ——, "Distributed evolutionary algorithm topologies with adaptive migration schemes," in *2011 IEEE Congress of Evolutionary Computation (CEC)*. IEEE, 2011, pp. 608–615.

[18] M. Sanu and G. Jeyakumar, "Empirical performance analysis of distributed differential evolution for varying migration topologies," *International Journal of Applied Engineering Research*, vol. 10, no. 5, pp. 11–919, 2015.

[19] T.-C. Wang, C.-Y. Lin, R.-T. Liaw, and C.-K. Ting, "Empirical analysis of island model on large scale global optimization," in *2019 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2019, pp. 342–349.

[20] J. Momin and X.-S. Yang, "A literature survey of benchmark functions for global optimization problems," *Int. Journal of Mathematical Modelling and Numerical Optimisation*, vol. 4, no. 2, pp. 150–194, 2013.

[21] X. Li, K. Tang, M. N. Omidvar, Z. Yang, K. Qin, and H. China, "Benchmark functions for the cec 2013 special session and competition on large-scale global optimization," *gene*, vol. 7, no. 33, p. 8, 2013.

[22] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM computing surveys (CSUR)*, vol. 35, no. 3, pp. 268–308, 2003.

[23] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE transactions on evolutionary computation*, vol. 1, no. 1, pp. 67–82, 1997.

[24] Y. Sun, S. K. Halgamuge, M. Kirley, and M. A. Munoz, "On the selection of fitness landscape analysis metrics for continuous optimization problems," in *7th International Conference on Information and Automation for Sustainability*. IEEE, 2014, pp. 1–6.

[25] K. M. Malan and A. P. Engelbrecht, "Quantifying ruggedness of continuous landscapes using entropy," in *2009 IEEE Congress on evolutionary computation*. IEEE, 2009, pp. 1440–1447.

[26] V. K. Vassilev, T. C. Fogarty, and J. F. Miller, "Information characteristics and the structure of landscapes," *Evolutionary computation*, vol. 8, no. 1, pp. 31–60, 2000.

[27] ——, "Smoothness, ruggedness and neutrality of fitness landscapes: from theory to application," *Advances in Evolutionary Computing: Theory and Applications*, p. 3, 2002.

[28] K. M. Malan and A. P. Engelbrecht, "Ruggedness, funnels and gradients in fitness landscapes and the effect on pso performance," in *2013 IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 963–970.

[29] M. Lunacek and D. Whitley, "The dispersion metric and the cma evolution strategy," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, 2006, pp. 477–484.

[30] M. Locatelli, "A note on the griewank test function," *Journal of global optimization*, vol. 25, no. 2, pp. 169–174, 2003.

[31] C. Hinrichs and M. Sonnenschein, "A distributed combinatorial optimisation heuristic for the scheduling of energy resources represented by self-interested agents." *IJBIC*, vol. 10, no. 2, pp. 69–78, 2017.

[32] J. Bremer and S. Lehnhoff, "An agent-based approach to decentralized global optimization-adapting cohda to coordinate descent," in *International Conference on Agents and Artificial Intelligence*, vol. 2. SCITEPRESS, 2017, pp. 129–136.

[33] H.-G. Beyer and S. Finck, "Happycat–a simple function class where well-known direct search algorithms do fail," in *International conference on parallel problem solving from nature*. Springer, 2012, pp. 367–376.

[34] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[36] A. Nieße, J. Bremer, and S. Lehnhoff, "On local minima in distributed energy scheduling." in *FedCSIS (Position Papers)*, 2017, pp. 61–68.