

Towards fully Decentralized Multi-Objective Energy Scheduling

Jörg Bremer

Department of Computing Science
Carl von Ossietzky University
Oldenburg, Germany
joerg.bremer@uni-oldenburg.de

Sebastian Lehnhoff

Department of Computing Science
Carl von Ossietzky University
Oldenburg, Germany
sebastian.lehnhoff@uni-oldenburg.de

Abstract—Future demand for managing a huge number of individually operating small and often volatile energy resources within the smart grid is preponderantly answered by involving decentralized orchestration methods for planning and scheduling. Many planning and scheduling problems are of a multi-objective nature. For the single-objective case – e.g. predictive scheduling with the goal of jointly resembling a wanted target schedule – fully decentralized algorithms with self-organizing agents exist. We extend this paradigm towards fully decentralized agent-based multi-objective scheduling for energy resources e.g. in virtual power plants for which special local constraint-handling techniques are needed. We integrate algorithmic elements from the well-known S -metric selection evolutionary multi-objective algorithm into a gossiping-based combinatorial optimization heuristic that works with agents for the single-objective case and derive a number of challenges that have to be solved for fully decentralized multi-objective optimization. We present a first solution approach based on the combinatorial optimization heuristics for agents and demonstrate viability and applicability in several simulation scenarios.

I. INTRODUCTION

The upcoming smart grid gives rise to several multi-objective control tasks. Due to the expected huge number of distributed energy resources (DER) that have to be controlled, self-organized and decentralized algorithms are seen as the most promising solution. On the other hand, surprisingly low effort has been put in developing decentralized multi-objective approaches so far. Here, we go with the example of the predictive scheduling problem [1].

To enable small and individually operated energy devices to responsibly take over control tasks, pooling of different DER is necessary in order to gain enough potential and flexibility. An established concept for such pooling is the virtual power plant (VPP) [2], [3]. Orchestration of such groups of energy units is done by different scheduling procedures that frequently involve multi-objective optimization.

Predictive scheduling [1] describes an optimization problem for day-ahead planning of energy generation in VPPs, where the goal is to select a schedule for each energy unit – from an individual search space of feasible schedules with respect to a future planning horizon – such that as a global objective the distance to a target power profile for the VPP is minimized (e.g. a product from an energy market). Actually, this constitutes a multi-objective problem. Further objectives

like cost minimization, maximization of residual flexibility (for later planning periods) or environmental impact are usually to be achieved concurrently [4]. These goals are often conflicting.

So far, the problem is often reduced to the single objective case for proper solving; if applicable with a combination of different objectives to a single, weighted sum of objectives. We propose a fully decentralized multi-objective algorithm for this problem based on concepts from the combinatorial optimization heuristics for agents (COHDA) and the S -metric selection evolutionary multi-objective algorithm (SMS-EMOA). The goal is to derive a self-organization approach that results in autonomously acting agents that determine a Pareto front (or at least an approximation) without any central control.

The rest of the paper is organized as follows: We recap multi-objective optimization in general, a centralized solution to the predictive scheduling problem and the single objective approach to the decentralized solution. We define the set of challenges that have to be solved to make algorithms like COHDA multi-objective capable and present a first solution approach that integrates concepts from the SMS-EMOA. We conclude with evaluation results from a simulation study and deduce some further research questions.

II. RELATED WORK

Decentralized, multi-agent-based multi-objective optimization has so far not gained much attention; at least in the sense of jointly calculating the Pareto front of a given problem. Some approaches have been developed for tuning a multi-agent system or for internal multi-objective decisions. In [5] an example is given for the emergency response planning problem, [6] presents an elevator control. These approaches use centralized algorithms for the multi-objective part. [7] gives an example for multi-objective reinforcement learning. A decision model for objective relationships after intra-agent multi-objective solving is presented in [8].

On the other hand, some approaches have been developed for acceleration by distributing fitness evaluation in multi-objective optimization. An example for a general framework is given in [9]. But, this is not decentralized problem solving by local, agent-based decisions in a collaboration scenario, as we strive for.

A. Predictive scheduling

We here consider rather small, distributed electricity producers that are supposed to pool together with likewise distributed electricity consumers and prosumers (like batteries) in order to jointly gain more degrees of freedom in choosing load profiles. In this way, they become a controllable entity with sufficient market power. In order to manage such a pool of DER, the following distributed optimization problem has to be frequently solved: A partition of a demanded aggregate schedule has to be determined in order to fairly distribute the load among all participating DER. A schedule x is a real valued vector with each element x_i denoting the amount of energy generated or consumed during the i th time interval within the planning horizon. Optimality usually refers to local (individual cost) as well as to global (e.g. environmental impact) objectives in addition to the main goal: Resemble the wanted overall load schedule as close as possible.

In [10], a support vector decoder has been introduced to cope with individual constraints of different types of energy units. Constraints may be technically rooted like the state of charge of attached batteries or thermal buffer stores or be economically soft rooted or be due to individual preferences. The basic idea is to learn the enclosing envelope around the set of feasible schedules in data space and to derive a formal description that allows mapping any given schedule to or into the feasible regions. In this way solution repair and space mapping can be achieved. Such constraint handling technique is in general referred to as decoder [11], [12]. Formally, a decoder function γ with

$$\begin{aligned} \gamma : [0, 1]^d &\rightarrow \mathcal{F}_{[0,1]} \subseteq [0, 1]^d \\ x &\mapsto \gamma(x) \end{aligned} \quad (1)$$

transforms any given (maybe in-feasible) schedule (scaled to $[0, 1]^d$) into a feasible one. Thus, the scheduling problem is transformed into an unconstrained formulation when using a decoder:

$$\delta \left(\sum_{i=1}^d s_i \circ \gamma_i(x'_i), \zeta \right) \rightarrow \min, \quad (2)$$

where γ_i denotes the decoder of unit i that produces feasible schedules $x'_i \in [0, 1]^d$ and s_i scales these schedules entrywise to correct power values resulting in schedules that are operable by that unit. Technically, scaling can also be integrated into the decoding process by combining both functions. Thus, for the rest of the paper we refer with γ to a decoder function that maps an infeasible schedule into the feasible region and scales it appropriately to the rated power of the respective energy unit. Please note that this constitutes only a single objective solution and multi-objective scenarios so far have to combine different objectives to a single one by a weighted aggregation. Unfortunately, this is not possible in case of a mixture of global and local objectives.

For the single objective case several solutions exist. In [13] an example for a centralized approach can be found, examples for decentralized approaches are given in [14]–[16]. A centralized multi-objective variant based on parallel

tempering that harnesses a decoder extension to co-encode different key performance indicators can be found in [17]. On the other hand, several approaches neglecting or relaxing individual constraint-handling can be found [2], [18].

For multi-objective optimization in general many approaches exist. In optimization problems with more than one and at least two conflicting objectives, Pareto optimization has become an appropriate means for solving [19]. As improving on one objective degrades each conflicting one, multi-objective optimization deals with finding a set of Pareto optimal solutions as trade-off between opposing solutions. Different algorithms have been designed to find an approximation to the Pareto-optimal set $M = \{x \in \mathbb{S} \mid \nexists x^* \in \mathbb{S} : x^* \prec x\}$ for a set of objective functions $f_{1,\dots,n} : \mathbb{S} \rightarrow \mathbb{R}$ defined on some search space \mathbb{S} [20]; and with $x \prec x^*$ denoting that x dominates x^* , i.e. all objective values of x are better than x^* . Different algorithms have been proposed [20]; among them are evolutionary algorithms [21], [22], genetic algorithms including the famous NSGA-II [23], or swarm-based approaches [24].

Predictive scheduling imposes some special needs on constraint handling to ensure that all local schedules are within the feasible phase-space of the individual energy resources [1], [25]. For constraint-handling in multi-objective optimization two general concepts are usually applied [26]. Either a penalty [27], [28] is added to each objective function degrading constraint violating solutions or the definition of Pareto-dominance is extended to take into account constraint violation [26], [29]. Introducing a penalty term changes the objective function and as in multi-objective optimization the impact on different objectives has to be balanced, a too weak set of penalties may lead to infeasible solutions whereas a too strong impact leads to poor distributions of solutions [26].

Nevertheless, all approaches for constraint integration so far need a closed form description of constraints. Constraints are given as a set of (possibly non-linear) in-equalities and equalities as well as a box-constraint demanding all parameters being from a specific range. In the smart grid domain, often no closed form descriptions of constraints are available. Such closed form description does not exist in decentralized energy resource scheduling that includes (at least in general) arbitrary unit types [30].

A first solution approach to hybridizing multi-objective optimization and decoder was given in [31], with a centralized approach based on SMS-EMOA.

B. SMS-EMOA

Using S -metric selection for evolutionary multi-objective algorithms has first been proposed by [22]. The S -metric is based on the hypervolume encapsulated by the set of non-dominated solutions and a reference point [21] and can thus be described as the Lebesgue measure Λ of the union of hypercubes defined by the reference point x_r and the set of non-dominated points m_i [19], [22]:

$$S(M) = \Lambda \left(\bigcup \{a_i \mid m \in M\} \right). \quad (3)$$

Algorithm 1 Basic algorithmic scheme of the SMS-EMOA (cf. [22]).

```

 $P^{(0)} \leftarrow \text{randomPopulation}()$ 
 $t \leftarrow 0$ 
while  $t < \text{max iterations}$  do
   $o \leftarrow \text{mutate} \circ \text{crossover}(P^{(t)})$ 
   $\{R_1, \dots, R_k\} \leftarrow \text{fast-nondominated-sort}(P^{(t)} \cup \{o\})$ 
   $p \leftarrow \arg \min_{s \in R_k} [\Delta_S(s, R_k)]$ 
   $P^{(t+1)} \leftarrow P^{(t)} \setminus \{p\}$ 
   $t \leftarrow t + 1$ 
end while

```

This metric constitutes an unary quality measure by mapping a solution set to a single value in \mathbb{R} : the size of the dominated space [32]. As it is desirable to have a large S -metric value for solution sets in multi-objective optimization, [32] first used this measure in a Simulated Annealing approach and [22] developed an evolution strategy (SMS-EMOA) based on this measure. Algorithm 1 shows the basic idea of SMS-EMOA. The algorithm repeatedly evolves a population of μ solutions. In each iteration, first a new solution is generated and added to the population. Subsequently, a selection process is started to find the worst individual in the solution which is then removed from the population. Thus, the number of individuals stays constant from a steady state perspective. Selection is done by first issuing a fast non-dominated sort after [23]. In this way, the Pareto fronts are ranked and from the front with the lowest rank the individual with the lowest contribution to the hypervolume (measured by the S -metric) is removed. This process is repeated until some stopping criterion – e. g. a number of maximum objective evaluations – is met.

In [31] the latter has already been hybridized with a decoder approach for flexibility modeling and constraint-handling in multi-objective energy management.

In general, two types of objective have to be considered. In [30] constraints have been identified on different locality levels. The same holds true for objectives in a VPP. Objectives on a global as well as on a local level have to be integrated. Objectives on a global level have to be achieved jointly. An example is given by the minimization of the deviation of the aggregated joint schedule from a given product schedule that has to be delivered as contracted. These objectives can only be achieved with joint effort. In contrast, local objectives like individual cost minimization are also to be integrated. Although evaluation can only be performed locally (individual cost), help from other to be able to choose a cheaper schedule is often necessary to achieve the goal. In the following we denote with f local and with F global objectives.

C. COHDA

In general, decentralized algorithms are considered advantageous in many fields of smart grid computation [33], [34]. For the case of predictive scheduling, [35] developed a decentralized algorithm for constrained combinatorial problems: The Combinatorial Optimization Heuristics for Distributed

Agents (COHDA). Combined with an appropriate abstraction for individual flexibilities [10]. So far, this fully decentralized approach works with a single objective function and integrates multiple objectives only by combining different objectives into a single one as weighted sum. We extended COHDA to a full-fledged decentralized multi-objective optimization algorithm.

COHDA has been designed as a fully distributed solution to the predictive scheduling problem (as distributed constraint optimization formulation) in smart grid management [36]. Each agent in the multi-agent system is in charge of controlling exactly one distributed energy resource (generator or controllable consumer) with procurement for negotiating the energy. All energy resources are drawn together to a virtual power plant and the controlling agents form a coalition that has to control the VPP in a distributed way.

An agent in COHDA does not represent a complete solution as it is the case for instance in population-based approaches [37], [38]. Each agent represents a class within a multiple choice knapsack combinatorial problem [39]. Applied to predictive scheduling each class refers to the feasible region in the solution space of the respective energy unit. Each agent chooses schedules as solution candidate only from the set of feasible schedules that belongs to the DER controlled by this agent. This selection is done according to local constraints and to a given objective that usually reflects solely the distance (dissimilarity) of the sum of this selection and the schedules of all other agents to the given target schedule.

Each agent is connected with a rather small subset of other agents from the multi-agent system and may only communicate with agents from this limited neighborhood. The neighborhood (communication network) is defined by a small world graph [40]. As long as this graph is at least simply connected, each agent collects information from the direct neighborhood and as each received message also contains (not necessarily up-to-date) information from the transitive neighborhood, each agent may accumulate information about the choices of other agents and thus gains his own local belief of the aggregated schedule that the other agents are going to operate. With this belief each agent may choose a schedule for the own controlled energy unit in a way that the coalition is put forward best while at the same time own constraints are obeyed and own interests are pursued. Thus, we have a multi-objective optimization problem when deciding on the best schedule.

All choices for own schedules are rooted in incomplete knowledge and beliefs in what other agents do; gathered from received messages. The taken choice (together with the basis for decision-making that has been received with prior messages) is communicated to all neighboring agents and, in this way, knowledge is successively spread throughout the coalition without any central memory. Thus, COHDA is a type of gossiping algorithm [41].

Each information update results in recalculating the own best schedule contribution and spreading it to the direct neighbors. By and by all agents accumulate complete information and as soon as no agent is capable of offering a schedule

leading to a better solution, the algorithm converges and terminates. Convergence has been proved in [42].

More formally, each time an agent receives a message, three successive steps are conducted. First, during the perceive phase an agent a_j updates its own working memory κ_j with the received working memory κ_i from agent a_i . From the foreign working memory the objective of the optimization (i. e. the target schedule) is imported (if not already known) as well as the configuration that constitutes the calculation base of neighboring agent a_i . An update is conducted if the received configuration is larger or has achieved a better objective value, what is only directly possible with a single-objective. In this way, schedules that reflect the so far best choices of other agents and that are not already known in the own working memory are imported from the received memory.

During the decision phase agent a_j has to decide on the best choice for its own schedule based on the updated belief about the system state $\mathfrak{S}_k^{(a_j)}$. Index k indicates the age of the system state information. The agent knows from a subset of (or from all) other agents, which schedules they are going to operate (the system state $\mathfrak{S}(a_j)_k$). Thus, the schedule that fills the gap to the desired target schedule exactly can be easily identified. Due to operational constraints of the controlled DER, this optimal schedule can usually not be operated. In addition, other reasons might render some schedules largely unattractive due to high cost.

Because of this reason, each agent is equipped with a so called decoder that automatically maps the identified optimal schedule to a nearby feasible schedule that is operable by the DER and thus feasible. Based on a set of feasible schedules sampled from an appropriate simulation model for flexibility prediction [43], a decoder can e. g. be based directly on this set (by linearly searching the schedule with the smallest deviation) or be built by learning a support vector model after the approach of [10]. Both approaches have individual advantages and drawbacks regarding computational complexity, search space size and accuracy. Here, we used the support vector version for efficiency reasons.

As the whole procedure is based exclusively on local decisions, each agent decides privately which schedules are taken. Private interest and preferences can be included and all information on the flexibility of the local DER is kept private.

D. Challenges

The COHDA approach can be adapted to many different optimization problems as has been demonstrated e. g. in [3], [30], [44], [45]. Basically, solution encoding, objective evaluation and internal, local decision method have to be adapted to the problem at hand. On the other hand, adapting to the multi-objective case entails some additional challenges that have to be solved:

a) Solution representation: As the goal now is a Pareto front, each agent will have to manage a set of schedules (a set of own contributions to the joint set of schedules). In the multi-objective case, determining the best own selection will

no longer work by just determining the missing difference to the target and repairing it with a decoder.

b) Solution quality assessment: Each time an agent in COHDA decides on a new contribution to a solution (remember: an agent represents just the local contribution, not a full solution) the quality of the solution with the old contribution is compared to the one resulting from the new contribution. This assessment is usually done using the objective function evaluating both candidates. In the multi-objective case an agent represents a set of contributions to a set of solutions, thus it is not possible to compare an old solution directly with a new one using the objective.

This problem can be overcome by using measures that evaluate the quality of a set of solutions with regard to the Pareto front. When using concepts from SMS-EMOA, the achieved hypercube volume can be used.

c) Incomplete solutions: During the initial setting time of COHDA, solutions are incomplete by design. COHDA has been developed for predictive scheduling. One agent starts by issuing a schedule (as local solution contribution) for the own energy resource. At this point in time, a solution consists only of this single contribution (as if it was a VPP with just a single energy resource). For the single-objective predictive scheduling case this is admissible as such a solution is always worse regarding the single objective of resembling the wanted target schedule as close as possible. After some negotiation steps, more agents join in and finally a contribution from all agents are on hand. For the multi-objective case it cannot be guaranteed that solutions with incomplete contribution are worse than complete solutions. An example may be given by minimizing cost as objective. Cost dominated by primary energy would deteriorate the objective if more energy resources joint to contribute to the solution; contradicting a minimization.

Several solutions are possible.

- 1) The protocol could be altered and each agent could be requested to calculate an initial contribution in order to avoid incomplete solutions. Depending on the problem at hand these initial contributions might be nonsense as they have been determined without knowledge on the others' decisions.
- 2) A penalty term could be added to the objective in order to deteriorate solutions based on the number of agents that still have to join. In this case the agents would need knowledge on the number of agents that are in the group.
- 3) Solutions with a larger number of contributions are always considered better regardless of the evaluation result. This might not hold for all objectives and may lead to wrong convergence directions.

For some objective functions there seems to be no issue at least if the number of agents is low enough compared with the number of network connections in between them and if the agents join in quick enough. For the simulations conducted with the first approach proposed here, after an initial deterioration of the solution quality a convergence could be observed towards

better solutions. Improved versions should integrate one of the afore mentioned approaches to guarantee convergence.

d) *Localization of objectives*: In standard COHDA each agent knows the global objective function. In multi-objective COHDA all global objective functions might also be known by all agents. On the other hand, not all objectives can be calculated by using the schedules of other agents directly, as it is the case in predictive scheduling. Calculating individual costs for example requires knowledge on private cost factors of other energy resources. Such factors are usually not known publicly nor communicated. Thus essential information for calculating the objectives is missing in a fully decentralized scenario.

In [46], an extension to the decoder approach has been proposed that is capable of annotating individual schedules with performance indicators. In [47] an ontology has been presented to capture and reliably interpret these information in a decentralized scenario. In the first approach presented here, this issue is currently neglected and only objectives that can be calculated without further information are used.

e) *Convergence detection*: In standard COHDA the solution converges to a single solution and eventually all agent represent the same solution. In the multi-objective case, all solution sets converge towards the same Pareto front and eventually all agents represent (an approximation) to the same Pareto front, but with probably different solution sets. Whether this is a problem or not highly depends on the specific problem at hand.

III. A FIRST APPROACH

A. Implementation

We implemented a fully decentralized multi-objective approach for predictive scheduling by extending the COHDA algorithm. First, we had to define the solution format. A solution to the overall problem (Eq. 2) is – in the multi-objective case – a set of sets of schedules for the virtual power plant; each one consisting of a schedule for the respective energy resource in the group:

$$\mathfrak{X} = \{\mathbf{X}_1, \dots, \mathbf{X}_n\}, \quad (4)$$

with

$$\mathbf{X}_k = (x_{ij}) \in \mathbb{R}^{m \times d} \quad (5)$$

where x_{ij} denotes the mean real power of energy resource i during the j th time period. Thus each row of the matrix \mathbf{X}_k represents a schedule for the respective energy resource. This solution set \mathfrak{X} is determined in a way that it approximates the Pareto front. In this way, each agent holds and negotiates on a set of schedules for the own energy resource.

Let $\{\mathbf{x}_1^{(a_j)}, \dots, \mathbf{x}_n^{(a_j)}\}$ denote the set of local schedules (for the own, controlled device) that is negotiated by agent a_j . Let κ_j be the current working memory of agent a_j (updated by an incoming message; cf. II-C). Let

$$X_O = \{\mathbf{x}_1^{(a_1)}, \dots, \mathbf{x}_n^{(a_1)}\}, \dots, \{\mathbf{x}_1^{(a_{m-1})}, \dots, \mathbf{x}_n^{(a_{m-1})}\} \in \kappa_j \quad (6)$$

be the currently known schedule selections (local solution candidates) from all the other agents $a_1, \dots, a_{m-1} \in \mathcal{A} \setminus a_j$. Basically, this is the system state belief $\mathfrak{S}(a_j)_k$ without the agent's own contribution from decision k :

$$X_O = \mathfrak{S}(a_j)_k \setminus \{\mathbf{x}_1^{(a_j)}, \dots, \mathbf{x}_n^{(a_j)}\}. \quad (7)$$

Now the procedure (performed by agent a_j) for deciding on the own schedule selection is as follows: The sum of schedules of the other agents is calculated as

$$\mathcal{O} = \{\mathbf{O}, \dots, \mathbf{O}_n\} \quad (8)$$

with

$$\mathbf{O}_i = \sum_{\mathbf{x}_i \in X_O} \mathbf{x}_i. \quad (9)$$

Now a solution of the MAS (cf. Eq. 4) to the joint Problem can be represented as

$$\left[\begin{array}{c} \mathbf{O}_1 \\ \mathbf{x}_1^{(a_j)} \end{array} \right], \dots, \left[\begin{array}{c} \mathbf{O}_n \\ \mathbf{x}_n^{(a_j)} \end{array} \right] \quad (10)$$

with $\mathbf{x}_1^{(a_j)}, \dots, \mathbf{x}_n^{(a_j)}$ being the decision variables of the local problem of deciding on the best local schedules under the assumption that the other agents' schedules are operated as communicated.

Solution candidate $\mathbf{x}_1^{(a_j)}, \dots, \mathbf{x}_n^{(a_j)}$ is initialized randomly and evolved for some iterations towards the Pareto front. For each evolution step one randomly chosen schedule $\mathbf{x}_k \in \mathbf{x}_1^{(a_j)}, \dots, \mathbf{x}_n^{(a_j)}$ is mutated to \mathbf{x}'_k by adding a Gaussian delta $r \in \mathcal{N}(0, 1)$ to one randomly chosen element of \mathbf{x}_k . As crossover operator, uniform crossover is applied. Please note, as an agent can only decide on its own schedules, mutation and crossover may not be applied to other agents' schedules from \mathcal{O} . Then, the agent performs a fast non-dominated sort on $\mathcal{O} \cup \left[\begin{array}{c} \mathbf{O}_k \\ \mathbf{x}'_k \end{array} \right]$.

For applying the fast non-dominance-sort as introduced in [48], from the worst front the worst individual (solution with the lowest hypercube contribution) is removed. For sorting and selecting the worst individual by S -metric selection, the dominance of solutions has to be determined by using the objective functions. For this purpose, we extend the definition of dominance by integrating the decoder set:

$$\mathbf{y} \prec \mathbf{y}^* \equiv \forall i = 1, \dots, d: y_i < y_i^* \quad (11)$$

in order to keep track of the individual (technical) constraints of the energy resources by setting

$$\mathbf{y} = \left(\begin{array}{c} f_{1,1}(\gamma_1(\mathbf{x}_1)) + \dots + f_{1,n}(\gamma_n(\mathbf{x}_n)) \\ f_{2,1}(\gamma_1(\mathbf{x}_1)) + \dots + f_{2,n}(\gamma_n(\mathbf{x}_n)) \\ \vdots \\ f_{m,1}(\gamma_1(\mathbf{x}_1)) + \dots + f_{m,n}(\gamma_n(\mathbf{x}_n)) \\ F_1(\mathbf{M}) \\ \vdots \\ F_\ell(\mathbf{M}) \end{array} \right) \quad (12)$$

as introduced in [31]. In this way, variations of the previous solutions in the solution set are produced by applying variation

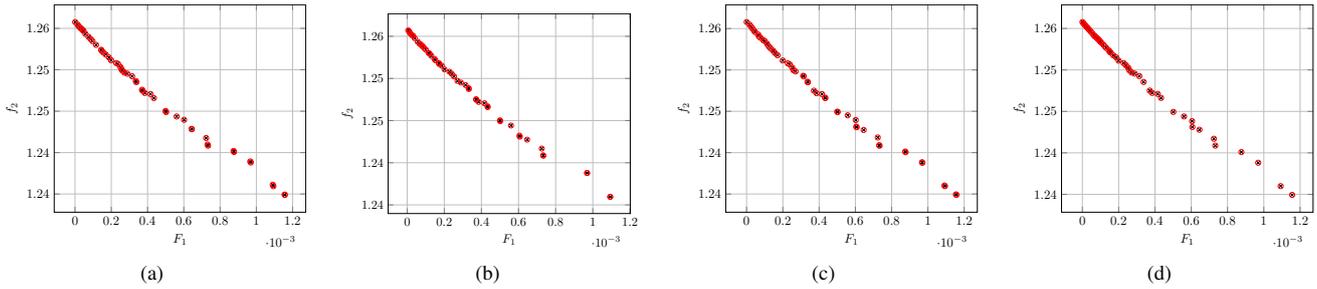


Fig. 1. Resulting solution sets of the 4 agents of the small scenario depicting the individually approximated Pareto fronts.

operators to the genotype. In this way the selection and crossover operators merely have to obey an easy to integrate box constraint that ensures that each value of a solution candidate is kept within $[0, 1]$ if max power is scaled to 1 (corresponding to 100% rated power). No further constraints have to be integrated. Thus, the problem formulation can be regarded as constraint-free. Constraint-handling is introduced by using decoder functions that abstract from individual capabilities or technical constraints of energy units. The set of decoders ensures that selection is done on feasible solutions only and thus that the solution set approaches a Pareto front without any knowledge about controlled energy units.

The last step in the COHDA process requires comparing the result achieved in the previous round with the current achievement. If the new one is better, it is communicated to the neighboring agents, else it is discarded and the old is kept without communicating any achievement. In the single objective case, the achieved objective values can be compared directly. This is not possible in the multi-objective case as the result constitutes a set of solutions. Thus, we decided to compare both results using the hypervolume between a reference point and the solution set as rather usual in multi-objective algorithms. For fast calculation of exact hypervolumes we applied the WFG (walking fish group) algorithm [49].

With these settings we addressed all challenges identified in section II-D to constitute a fully decentralized, agent-based determination of the Pareto front of a joint multi-objective problem.

B. Results

For our evaluation we simulated different virtual power plants consisting of different co-generation plants. The model has already been used and evaluated in different projects, e. g. [13], [46], [50], [51]. We started with a rather small setting of four agents and 96-dimensional schedules resulting in a 384-dimensional search space which has already been evaluated to be highly multi-modal and ragged [52]. As goal, two objectives were set: F_1 denotes the deviation of the joint schedule from the desired target schedule ($\|\cdot\|_2$) and f_2 equalizes the run of the co-generation plants by minimizing peak loads:

$$f_2 = \sum_{\mathbf{x}_i \in \mathbf{X}} \sum_{j=1}^d (x_{ij} - \mu)^2, \quad (13)$$

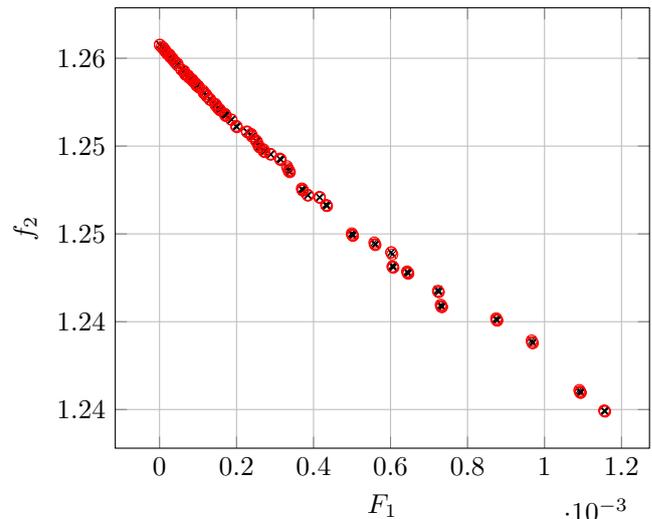


Fig. 2. Combination of local Pareto fronts of agents from solution Fig. 1. Solutions are marked with a cross; non-dominated solutions are marked by a circle. In this example, no solution is dominated by a solution from another agent.

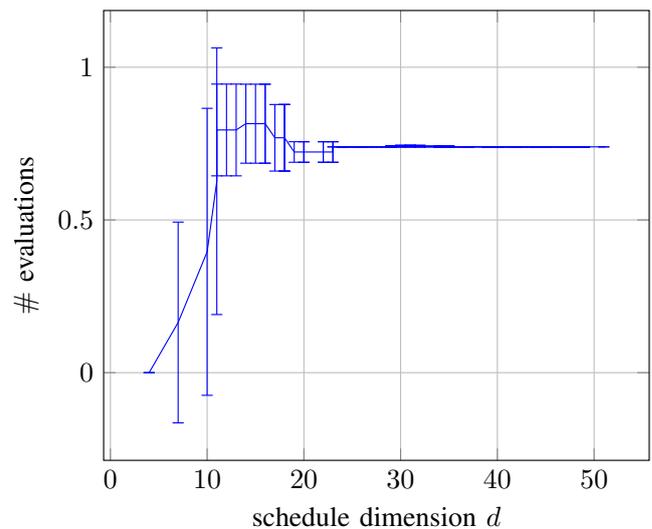


Fig. 3. Convergence and inter-agent variability (error bars) of the small scenario. Only the first 50 (out of ~ 1700) measuring points are depicted.

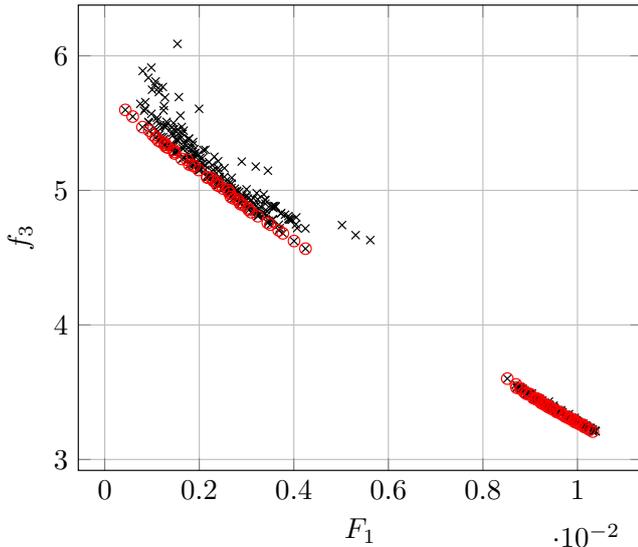


Fig. 4. Consolidated Pareto fronts from the large example. Non-dominated solutions are additionally marked with a circle. As COHDA is a heuristic, this example has obviously not completely reached a common front approximation. Maybe, a post-processing that removes dominated solutions could improve the approach significantly.

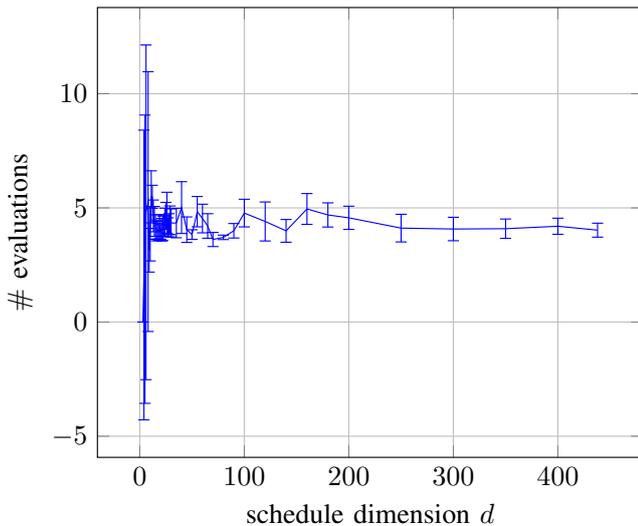


Fig. 5. Convergence and inter-agent variability (error bars) of the large scenario.

with μ being the mean power over the whole planning horizon. In this way, the variance in power levels is minimized. Figure 1 shows an example of the resulting Pareto front approximations of the individual 4 agents. The evolutionary part during the decision phase has been run for 500 iterations each. A cross marks an individual for the solution set (50 schedules per agent) and a circle marks a non-dominated solution from the front. Note, that solution here denotes already a solution (joint schedule) for the predictive scheduling problem. Even if all schedules from the individual result sets of the agents are plotted jointly in a single plot, all solutions are non-dominated;

see Figure 2. Thus, all solutions that are individually generated by the agents stem from the same Pareto front approximation, although there are slight differences in the solution sets (regarding distribution on the front) as can be visually seen in Figure 1. The agents approximate the same front but exhibit differences in the individual solution sets.

Some statistics on this scenario with different numbers of evolutionary iterations are given in Table I.

As the proposed approach is still a heuristic, results are as yet not that perfect with growing problem size. Figure 4 shows another example from a scenario with 50 agents approximating a 50 schedule solution set each.

Here another objective has been tested: Achieve a desired state of charge (SOC) for some of the thermal buffer stores that are attached to the co-generation plants.

$$f_3 = \sum \delta(\text{SOC}_i, \text{soc}(x_i)). \quad (14)$$

In this larger scenario some of the solutions in the joint solution set are still dominated by the solutions from others, even though 5000 iterations have been conducted during decision phases. Obviously, the number of necessary iterations quickly grows with problem size: deteriorating needed negotiation time.

On the other hand, the mean hypervolume (as quality measure for a solution) converges quite early to an acceptable value. Figure 3 shows the convergences of the process that led to result 1. Here, the mean (so long achieved) hypervolume of all agents is measured at discrete points in time from the concurrently asynchronously running multi-agent system. Error bars show the inter-agents variance. At the same time the variance among all agents is determined. Depicted are only the first iterations, not the full process. Depending on the specific use case at hand, it might thus be possible to stop the negotiation at an earlier stage with a still acceptably good solution. Some more investigations will be necessary here. Figure 5 shows the situation for the second case.

TABLE I
PERFORMANCE INDICATORS FOR BOTH TEST SCENARIOS. THIS TIME f_2 WAS USED FOR BOTH.

indicator	4 agents	50 agents
hypervolume	0.401 ± 0.213	0.533 ± 0.258
best F_1	0.103 ± 0.110	0.083 ± 0.006
best f_2	1.785 ± 0.213	1.745 ± 0.142
# messages	2191.6 ± 2407.1	679.6 ± 516.6
# decisions	1064.1 ± 1166.6	328.4 ± 245.3

Mean achieved results for 100 runs each are depicted in Table I. The achieved hypervolume in the larger scenario larger due to the better aggregated schedule (F_1). This observation is consistent with the single-objective case and rooted in the higher flexibility of larger VPPs. The number of exchanged messages and decisions decreases significantly with growing scenario size probably allowing for more complex decision routines of the agents in future improvements.

IV. CONCLUSION

For many applications within the smart grid scheduling domain, multi-objective optimization problems have to be solved. As for scalability reasons decentralized (agent-based) algorithms are seen as a promising solution, multi-objective capability has to be integrated into these methods. At the same time, proper constraint-handling is indispensable for acceptance.

With the approach demonstrated here, we integrated multi-objective capabilities taken from SMS-EMOA into fully decentralized energy scheduling.

Applicability and effectiveness of the proposed approach have been demonstrated by simulations. Nevertheless, some questions remain for further research: Replacing the decoder decision by a more complex optimization based decision method entails the need for tuning additional parameters; e. g. the (probably adaptive) number of iterations during each decision procedure. Maybe a substantial acceleration could be reached with additional convergence detection methods. Designing mutation, crossover and selection is also still subject to specialized improvements. Moreover, an integration of indicators into the communicated solution data format could improve privacy as local objectives (and thus their calculation details) would no longer be publicly known. Nevertheless, this first approach demonstrated the general feasibility of fully decentralized multi-objective optimization.

REFERENCES

- [1] M. Sonnenschein, O. Lünsdorf, J. Bremer, and M. Tröschel, "Decentralized control of units in smart grids for the support of renewable energy supply," *Environmental Impact Assessment Review*, no. 0, pp. –, 2014, in press.
- [2] Ł. B. Nikonowicz and J. Milewski, "Virtual power plants – general review: structure, application and optimization." *Journal of Power Technologies*, vol. 92, no. 3, 2012.
- [3] A. Nieße, S. Beer, J. Bremer, C. Hinrichs, O. Lünsdorf, and M. Sonnenschein, "Conjoint dynamic aggregation and scheduling for dynamic virtual power plants," in *Federated Conference on Computer Science and Information Systems - FedCSIS 2014, Warsaw, Poland*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 9 2014.
- [4] J. Bremer and M. Sonnenschein, "Automatic reconstruction of performance indicators from support vector based search space models in distributed real power planning scenarios," in *Informatik 2013, 43. Jahrestagung der Gesellschaft für Informatik e.V. (GI), Informatik angepasst an Mensch, Organisation und Umwelt, 16.-20. September 2013, Koblenz*, ser. LNI, M. Horbach, Ed., vol. 220. GI, 2013, pp. 1441–1454.
- [5] G. Narzisi, V. Mysore, and B. Mishra, "Multi-objective evolutionary optimization of agent-based models: An application to emergency response planning," in *Computational Intelligence*. IASTED/ACTA Press, 2006, pp. 228–232.
- [6] Y. Gu, "Multi-objective optimization of multi-agent elevator group control system based on real-time particle swarm optimization algorithm," *Engineering*, vol. 04, no. 07, pp. 368–378, 2012.
- [7] "Adaptive multi-objective reinforcement learning with hybrid exploration for traffic signal control based on cooperative multi-agent framework," *Engineering Applications of Artificial Intelligence*, vol. 29, pp. 134 – 151, 2014.
- [8] A. illah Mouaddib, M. Boussard, and M. Bouzid, "Towards a formal framework for multi-objective multi-agent planning," in *In Proc. of the 6th Int. Conf. on Autonomous Agents and Multiagent Systems*, 2007, pp. 801–808.
- [9] L. T. Bui, H. A. Abbass, and D. Essam, "Local models—an approach to distributed multi-objective optimization," *Comput. Optim. Appl.*, vol. 42, no. 1, pp. 105–139, Jan. 2009.
- [10] J. Bremer and M. Sonnenschein, "Constraint-handling for optimization with support vector surrogate models - A novel decoder approach," in *ICAART 2013 - Proceedings of the 5th International Conference on Agents and Artificial Intelligence, Volume 2, Barcelona, Spain, 15-18 February, 2013*, J. Filipe and A. L. N. Fred, Eds. SciTePress, 2013, pp. 91–100.
- [11] C. A. Coello Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 11-12, pp. 1245–1287, Jan. 2002.
- [12] S. Koziel and Z. Michalewicz, "Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization," *Evol. Comput.*, vol. 7, pp. 19–44, 03 1999.
- [13] C. Hinrichs, J. Bremer, S. Martens, and M. Sonnenschein, "Partitioning the data domain of combinatorial problems for sequential optimization," in *FedCSIS*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2016, pp. 551–559.
- [14] J. Bremer and M. Sonnenschein, "A distributed greedy algorithm for constraint-based scheduling of energy resources," in *Federated Conference on Computer Science and Information Systems - FedCSIS 2012, Wroclaw, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 1285–1292.
- [15] A. Nieße, C. Hinrichs, J. Bremer, and M. Sonnenschein, "Local Soft Constraints in Distributed Energy Scheduling," in *5th International Workshop on Smart Energy Networks & Multi-Agent Systems, Proceedings of the 2016 Federated Conference on Computer Science and Information Systems*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., Gdansk, 2016.
- [16] A. Nieße and M. Tröschel, "Controlled self-organization in smart grids," in *Proceedings of the 2016 IEEE International Symposium on Systems Engineering (ISSE)*. IEEE, 2016, pp. S. 1–6.
- [17] J. Bremer and M. Sonnenschein, "Parallel tempering for constrained many criteria optimization in dynamic virtual power plants," in *2014 IEEE Symposium on Computational Intelligence Applications in Smart Grid, CIASG 2014, Orlando, FL, USA, December 9-12, 2014*. IEEE, 2014, pp. 51–58.
- [18] C.-S. Karavas, G. Kyriakarakos, K. Arvanitis, and G. Papadakis, "A multi-agent decentralized energy management system based on distributed intelligence for the design and control of autonomous polygeneration microgrids," *Energy Conversion and Management*, vol. 103, 10 2015.
- [19] C. A. Coello Coello, G. B. Lamont, and D. A. V. Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems (Genetic and Evolutionary Computation)*. Berlin, Heidelberg: Springer-Verlag, 2006.
- [20] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 32 – 49, 2011.
- [21] E. Zitzler and L. Thiele, "Multiobjective optimization using evolutionary algorithms — a comparative case study," in *Parallel Problem Solving from Nature — PPSN V*, A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 292–301.
- [22] M. Emmerich, N. Beume, and B. Naujoks, "An emo algorithm using the hypervolume measure as selection criterion," in *Evolutionary Multi-Criterion Optimization*, C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 62–76.
- [23] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr 2002.
- [24] J. J. Durillo, J. García-Nieto, A. J. Nebro, C. A. Coello, F. Luna, and E. Alba, "Multi-objective particle swarm optimizers: An experimental comparison," in *Proceedings of the 5th International Conference on Evolutionary Multi-Criterion Optimization*, ser. EMO '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 495–509.
- [25] J. Bremer, B. Rapp, and M. Sonnenschein, "Support vector based encoding of distributed energy resources' feasible load spaces," in *IEEE PES Conference on Innovative Smart Grid Technologies Europe, Chalmers Lindholmen, Gothenburg, Sweden*, 2010.
- [26] F. Snyman and M. Helbig, "Solving constrained multi-objective optimization problems with evolutionary algorithms," in *Advances in Swarm Intelligence*, Y. Tan, H. Takagi, Y. Shi, and B. Niu, Eds. Cham: Springer International Publishing, 2017, pp. 57–66.
- [27] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1994.

- [28] A. Smith and D. Coit, *Handbook of Evolutionary Computation*. Department of Industrial Engineering, University of Pittsburgh, USA: Oxford University Press and IOP Publishing, 1997, ch. Penalty Functions, p. Section C5.2.
- [29] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: John Wiley & Sons, Inc., 2001.
- [30] C. Hinrichs, J. Bremer, and M. Sonnenschein, "Distributed Hybrid Constraint Handling in Large Scale Virtual Power Plants," in *IEEE PES Conference on Innovative Smart Grid Technologies Europe (ISGT Europe 2013)*. IEEE Power & Energy Society, 2013.
- [31] J. Bremer and S. Lehnhoff, "Hybridizing s-metric selection and support vector decoder for constrained multi-objective energy management," in *Hybrid Intelligent Systems*, A. M. Madureira, A. Abraham, N. Gandhi, and M. L. Varela, Eds. Cham: Springer International Publishing, 2020, pp. 249–259.
- [32] M. Fleischer, "The measure of pareto optima applications to multi-objective metaheuristics," in *Evolutionary Multi-Criterion Optimization*, C. M. Fonseca, P. J. Fleming, E. Zitzler, L. Thiele, and K. Deb, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 519–533.
- [33] A. Nieße, S. Lehnhoff, M. Tröschel, M. Uslar, C. Wissing, H. J. Apperath, and M. Sonnenschein, "Market-based self-organized provision of active power and ancillary services: An agent-based approach for smart distribution grids," in *Complexity in Engineering (COMPENG)*, 2012, June 2012, pp. 1–5.
- [34] S. D. Ramchurn, P. Vytelingum, A. Rogers, and N. R. Jennings, "Putting the 'smarts' into the smart grid: A grand challenge for artificial intelligence," *Commun. ACM*, vol. 55, no. 4, pp. 86–97, Apr. 2012.
- [35] C. Hinrichs, S. Lehnhoff, and M. Sonnenschein, "A Decentralized Heuristic for Multiple-Choice Combinatorial Optimization Problems," in *Operations Research Proceedings 2012*. Springer, 2014, pp. 297–302.
- [36] C. Hinrichs, M. Sonnenschein, and S. Lehnhoff, "Evaluation of a Self-Organizing Heuristic for Interdependent Distributed Search Spaces," in *International Conference on Agents and Artificial Intelligence (ICAART 2013)*, J. Filipe and A. L. N. Fred, Eds., vol. Volume 1 – Agents. SciTePress, 2013, pp. 25–34.
- [37] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.
- [38] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, Nov. 2007.
- [39] T. Lust and J. Teghem, "The multiobjective multidimensional knapsack problem: a survey and a new approach," *CoRR*, vol. abs/1007.4063, 2010.
- [40] D. Watts and S. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, no. 393, pp. 440–442, 1998.
- [41] J. Liu, B. Anderson, M. Cao, and A. Morse, "Analysis of accelerated gossip algorithms," *Automatica*, vol. 49, no. 4, pp. 873–883, 4 2013.
- [42] C. Hinrichs, "Selbstorganisierte Einsatzplanung dezentraler Akteure im Smart Grid," Ph.D. dissertation, Department for Computing Science, 2014. [Online]. Available: <http://oops.uni-oldenburg.de/1960/>
- [43] J. Bremer and M. Sonnenschein, "Sampling the search space of energy resources for self-organized, agent-based planning of active power provision," in *27th International Conference on Environmental Informatics for Environmental Protection, Sustainable Development and Risk Management, EnviroInfo 2013, Hamburg, Germany, September 2-4, 2013. Proceedings*, ser. Berichte aus der Umweltinformatik, B. Page, A. G. Fleischer, J. Göbel, and V. Wohlgemuth, Eds. Shaker, 2013, pp. 214–222.
- [44] A. Nieße and M. Sonnenschein, "Using grid related cluster schedule resemblance for energy rescheduling - goals and concepts for rescheduling of clusters in decentralized energy systems," in *SMARTGREENS*, B. Donnellan, J. F. Martins, M. Helfert, and K.-H. Krempels, Eds. SciTePress, 2013, pp. 22–31.
- [45] M. Sonnenschein, C. Hinrichs, A. Nieße, and U. Vogel, "Supporting renewable power supply through distributed coordination of energy resources," in *ICT Innovations for Sustainability*, ser. Advances in Intelligent Systems and Computing, L. M. Hilty and B. Aebischer, Eds. Springer International, 2015, vol. 310, pp. 387–404.
- [46] J. Bremer and M. Sonnenschein, "Automatic reconstruction of performance indicators from support vector based search space models in distributed real power planning scenarios," in *Informatik 2013, 43. Jahrestagung der Gesellschaft für Informatik e.V. (GI), Informatik angepasst an Mensch, Organisation und Umwelt, 16.-20. September 2013, Koblenz*, ser. LNI, M. Horbach, Ed., vol. 220. GI, 2013, pp. 1441–1454.
- [47] J. Bremer, "Ontology based description of der's learned environmental performance indicators," in *Proceedings of the 1st International Conference on Smart Grids and Green IT Systems – SmartGreens 2012*, B. Donnellan, J. P. Lopes, J. Martins, and J. Filipe, Eds. Porto, Portugal: SciTePress, 04 2012, pp. 107–112.
- [48] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii," in *Parallel Problem Solving from Nature PPSN VI*, M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. J. Merelo, and H.-P. Schwefel, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 849–858.
- [49] R. Lyndon While, L. Bradstreet, and L. Barone, "A fast way of calculating exact hypervolumes," *IEEE Trans. Evolutionary Computation*, vol. 16, pp. 86–95, 02 2012.
- [50] J. Neugebauer, O. Kramer, and M. Sonnenschein, "Classification cascades of overlapping feature ensembles for energy time series data," in *Proceedings of the 3rd International Workshop on Data Analytics for Renewable Energy Integration (DARE'15)*. Springer, 2015.
- [51] J. Bremer and S. Lehnhoff, *Decentralized Surplus Distribution Estimation with Weighted k-Majority Voting Games*. Cham: Springer International Publishing, 2017, pp. 327–339.
- [52] A. Nieße, J. Bremer, and S. Lehnhoff, "On local minima in distributed energy scheduling," in *Position Papers of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3-6, 2017.*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 12.