



Controlled Self-organization for Steering Local Multi-objective Optimization in Virtual Power Plants

Jörg Bremer^(✉) and Sebastian Lehnhoff

University of Oldenburge, 26129 Oldenburg, Germany
{joerg.bremer,sebastian.lehnhoff}@uni-oldenburg.de

Abstract. The future smart grid has to be operated by rather small and hardly flexible energy resources. Such duty comprises different planning tasks. Virtual power plants powered by multi-agent control are seen as a promising aggregation scheme for coping with problem size and for gaining flexibility for distributed load planning. If agents are allowed to freely include local preferences into decision making the overall solution quality deteriorates significantly if no control mechanism is installed. We scrutinized this deterioration and propose an approach based on controlled self-organization to achieve an overall maximization of integrated local preferences while at the same time preserving global solution quality for grid control as much as possible. Some first results prove the applicability of the general approach. Further research directions and questions for future work are derived from these first results.

Keywords: Controlled self-organization · Smart grid · Decentralized multi-objective optimization

1 Introduction

The structure of the energy supply within the power grid is constantly changing. The future smart grid will basically consist of small, volatile and hardly controllable renewable decentralized energy resources (DER). In the long run, these small generation units will have to assume responsibility for all daily grid operation tasks and ancillary services. This can only be achieved when units pool together (most likely with controllable load and batteries) to gain flexibility and potential.

Virtual power plants (VPP) are a well-known instrument for aggregating and controlling DER [2]. A VPP comprises individually operated DER loosely coupled by some communication means and jointly orchestrated by some (decentralized) control algorithm [6]. Integration into current market structures recently also led to VPP systems that frequently re-configure themselves for a market and product specific alignment [27]. In general, VPP concepts for several purposes (commercial as well as technical) have already been developed. A use case

commonly emerging within VPP control is the need for scheduling the participating DER. Independently of the specific objective at hand, a schedule (course of energy generation) for each DER has to be found such that the schedule that finally is assigned to a DER is operable without violating any technical constraint [7]. For this paper we go with the example of scheduling for active power planning in day-ahead scenarios (not necessarily 24 h but for some given future time horizon). For large scale problems, distributed (usually agent based) approaches are currently discussed not least due to further advantages like ensured privacy issues. Some recent implementations are [13, 16, 32]. Distributed organization and self-organized control is also especially a characteristic of dynamic virtual power plants (DVPP) [27].

Some types of VPP specialize in predictive scheduling as operational service [26]. The goal of predictive scheduling is to select a schedule for each energy unit – with respect to a given search space of feasible schedules with respect to a future planning horizon – such that a global objective function (e.g. a target power profile for the VPP) is optimized. This target profile may be a schedule that is assigned to a VPP as a result of some trading action on an energy market. We consider this target schedule as already given for the rest of this contribution.

For solving this problem in a decentralized way, agent-based solutions have been developed. One approach based on a gossiping type of algorithm is given by COHDA – the combinatorial optimization heuristic for decentralized agents [3, 15].

The key concept of COHDA is an asynchronous iterative approximate best-response behavior, where each agent – representing a decentralized energy unit – reacts to updated information from other agents by adapting its own selected schedule with respect to local and global objectives. Different objectives are handled by scalarization into a single objective as weighted sum of objectives. As the global (main) goal is achieving a consensus on operation modes such that the market given energy schedule is delivered as agreed on (small deviations are acceptable), attention has to be paid to the result quality of this specific objective. In order to ensure a minimum solution quality for the main goal, control of the weighting of local objectives is needed.

From the perspective of individually operated decentralized energy resources it is desirable to maximize the weight of local preferences. As different participants in the VPP have different characteristics in their flexibilities and thus have different importance in achieving the main goal, individual maximum weight are possible for different participants. On the other hand, the maximum local weights should be assigned in a fair way, at least in the long run.

We propose to use the concept of controlled self-organization to steer the individual use of local preferences based on the current composition (individual flexibilities based on current operational state of different energy units) of the VPP and formulate the optimization problem that has to be solved for finding a set of best local weights.

The rest of the paper is organized as follows. We start with a recap of decentralized algorithms for the scrutinized problem and controlled self-organization

in general. We derive an architecture for controlling the local weights and present a first solution for the emerging optimization problem based on evolution strategies. Some results from a simulation study conclude the paper.

2 Related Work

In order to cope with the growing load planning and control complexity in the future smart grid, agent-based and self-organization approaches for problem solving are most promising [36]. Examples can already be found in [1, 9, 10, 30, 32]. As a use case for this paper we use the example of decentralized predictive scheduling [15].

The task of predictive scheduling is to plan energy production (e. g. for the next day) of a group of generators. In the future smart grid instead a large group of small distributed energy resources will have to be planned for appropriate dispatch instead – probably pooled together with controllable load and batteries for higher flexibility. Such group is commonly referred to as virtual power plant. In many scenarios such a group trades its flexibility on some energy market and is assigned a schedule from market that has to be operated. The target schedule usually comprises 96 time intervals of 15 min each with a given amount of energy (or equivalently mean active power) for each time interval, but might also be constituted for a shorter time frame by a given energy product that the VPP has to deliver. A schedule in this context is a real valued vector \mathbf{x} with each element \mathbf{x}_i denoting the respective amount of energy generated or consumed during the i -th time interval. It is the goal of the predictive scheduling to find then exactly one schedule for each energy unit such that

1. each schedule that is assigned to a specific energy resource can be operated by the respective energy unit without violating any hard technical constraint, and
2. the difference between the sum of all targets and a desired given market schedule is minimized.

A basic formulation of the scheduling problem is given by

$$\delta \left(\sum_{i=1}^m \mathbf{x}_i, \zeta \right) \rightarrow \min; \text{ s.t. } \mathbf{x}_i \in \mathcal{F}_i \forall U_i \in \mathcal{U}. \quad (1)$$

In Eq. (1) δ denotes an (in general) arbitrary distance measure for evaluating the difference between the aggregated schedule of the group and the desired target schedule ζ . [14] for example uses the Manhattan distance; in [4] also measures like excess supply minimization [11] have for example been integrated and tested. Throughout this paper, we will use the Euclidean distance $|\cdot|_2$.

\mathcal{F}_i denotes the feasible region of energy unit U_i . Feasibility of solution can be assured by using a decoder as constraint-handling technique. Such a decoder learns the individual set of feasible schedules of an energy unit and repairs infeasible solutions during optimization [8].

For solving this optimization tasks the fully decentralized combinatorial optimization heuristics for distributed agents (COHDA) has been developed [13, 15, 27]. An agent in COHDA does not represent a complete solution as it is the case for instance in population-based approaches. Each agent represents a class within a multiple choice knapsack combinatorial problem [20]. 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. 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 [35]. 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; what in turn – if not controlled – may lead to worse main goal quality.

A broadly used model for implementing intelligent agents has been developed by the Rational Agent Project at the Stanford Research Institute (<https://www.sri.com/>). In this architecture, each agent possesses beliefs about his environment, has a desired goal and access to a database with plans to achieve the goal. Due to the interplay of these beliefs, desires and intentions the architecture is known as BDI architecture.

Without a concrete database with plans, sometimes self-organization is the goal within multi-agent systems. Organic computing systems are highly dynamic and bundle a huge number of changing components; not necessarily agents [23]. Orchestration is not induced from the outside or by central control, but arises as emergent behavior [22]. This trait results in self-configuring, -adapting, and -healing and autonomous systems. Consequently, traditional tools and methods for design and analysis do no longer apply to such systems [34]. In order to introduce the advantages of classical closed loop control systems into the control of emergent systems, a specific observer/ controller architecture has been developed [33]. In this architecture the actual system is under observation of one or more observer components. These observers scrutinize and evaluate emergent behavior patterns inside the controlled system, aggregate information, and report to a controller component that decides based on user allowances and machine learning analysis of report history. In this way, a controlled self-organization is achieved by embedding the actual system into a control loop [25, 28].

Examples for implemented controlled self-organization for computer-based applications are given in [19, 29], but can also be found in chemistry [18] or quantum physics [31].

We want to use the concept of controlled self-organization to induce a control entity into the multi-agent systems that may observe and keep track of the impact of local optimization preferences and is capable to intervene by providing a vector for individually max values for the weights that the agents may use.

3 Controlling Local Objectives

Optimization problems with different (opposing) objectives constitute a multi-objective problem. In this case, optimality has to be defined by Pareto optimality; i. e. improvement on one objective cannot be achieved without deterioration on the other [24].

Thus, it seems immediately clear that in the case of predictive scheduling the solution quality for the main goal (objective of resembling the market schedule as close as possible) deteriorates if the agents give a too strong weighting to the local objectives (their local preferences).

We created a simulation of different co-generation plants [21] and a multi-agent system with one agent associated to each co-generation plant. The agents are capable of using the COHDA algorithm to conduct load planning in a decentralized way. Differing from the original algorithm the agents were allowed to use a weighted sum of two objectives for evaluating the solutions. One objective was for the global goal of achieving a close as possible resemblance of the sum of schedules to the wanted market schedule. The second objective allowed integrating local preferences. We use the example of maximizing the remaining flexibility of the energy unit for trading at the market later. For this purpose we defined

$$E_{S_d}(\mathbf{x}) = \frac{(0.5(\vartheta_{min} + \vartheta_{max}) - \vartheta_d)^2}{(\vartheta_{max} - \vartheta_{min})^2}. \quad (2)$$

E_{S_d} denotes the state of charge (SoC) error of the buffer store after operating d intervals of the schedule by taking into account the squared deviation of the resulting buffer temperature ϑ_d from the mean of the allowed temperature range $[\vartheta_{min}, \vartheta_{max}]$. In this way the remaining flexibility (to trade on some future market) for the controlled co-generation plant is maximized [5]. To this end, COHDA in the multi-agent system was equipped with an aggregated objective

$$w_j \cdot E_{S_d}(\mathbf{x}_j) + (1 - w_j) \cdot \delta \left(\sum_{i=0}^n \mathbf{x}_i, \zeta \right) \rightarrow \min \quad (3)$$

for optimizing the global objective of minimizing the deviation of the sum of all schedules $\sum_{i=0}^n \mathbf{x}_i$ (from agents 1 to n) from the desired market schedule ζ and the local goal of minimizing the deviation from the local mean buffer charge E_{S_d} at the same time. Each agent a_j may individually set the weight w_j for the own local objective individually.

First we tested the impact of different weights on the achieved resemblance to the market schedule. The mean absolute percentage error measure (MAPE) was

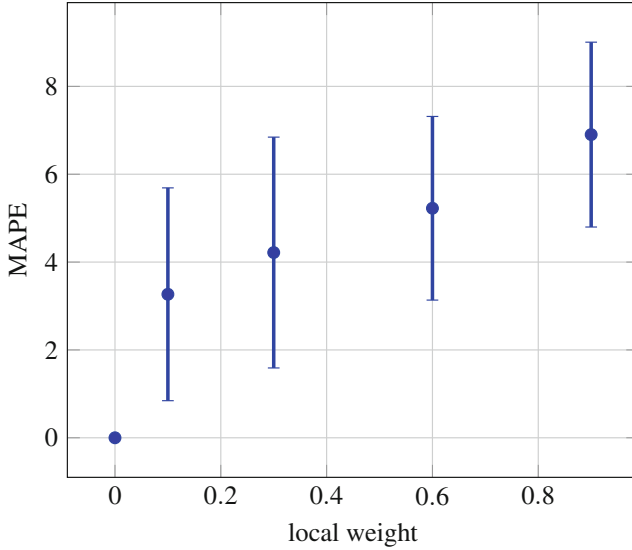


Fig. 1. Deterioration of the primary goal with varying (but identical for all agents) weights for the local (private) objective.

Table 1. Mean and best results for randomly sampled weights (different for all agents) for different normal distributions of w .

| $\mathcal{N}(\mu, \sigma)$ | Mean | Min. |
|----------------------------|-------|-------|
| (0.3,0.15) | 7.246 | 1.738 |
| (0.6,0.3) | 3.805 | 0.922 |

used in order to guarantee comparability between different scenarios. Figure 1 shows the result for a scenario with 10 co-generation plants and the same weight for all agents.

As expected the achieved main goal deteriorates if agents are allowed to concurrently include local (opposing) objectives. With a growing weight the result gets worse for the main goal. On the other hand, we scrutinized different weights and consider that different agents with different co-generation plants have different importance for the result within the group. Table 1 shows the result for two different spreads of the weights within the group. We tested 1000 different random combinations of weights for each group. As can be seen (by the growing standard deviation and the minimum results), there are combinations of weights that still result in good primary goal results. Thus, we can conclude that there is potential for finding good combinations of weights by an optimization approach.

To achieve this, we propose the architecture depicted in Fig. 2. Following the approach of controlled self-organization, a control entity will be responsible to

interfere the self-organization process if agents choose local weights that lead to deteriorated results when trying to achieve the global goal of jointly operating a generation schedule that has been agreed on at some market. If the result quality falls below some given threshold, the weights are adjusted. In our case, we chose a mean absolute percentage error of 1%, meaning a mean deviation of 1% from the agreed energy delivery.

4 Results

For our evaluation, we use the famous co-variance matrix adaption evolution strategy (CMA-ES). CMA-ES is a well-known evolution strategy for solving black box problems and aims at learning lessons from previous successful evolution steps. New solution candidates are sampled from a multi variate normal distribution $\mathcal{N}(0, \mathbf{C})$ with covariance matrix \mathbf{C} which is adapted in a way that maximizes the occurrence of improving steps according to previously seen distributions for good steps. Sampling is weighted by a selection of solutions of the parent generation. In a way, a second order model of the objective function is exploited for structure information. A comprehensive introduction can for example be found in [12]. CMA-ES has a set of parameters that can be tweaked to some degree for a problem specific adaption. Nevertheless, default values applicable for a wide range of problems are available. We have chosen to set these values after [12] for our experiments.

For optimizing the weight vector \mathbf{w} we defined the following objective:

$$v \cdot \frac{|\mathbf{w}| - \sum_{i=1 \dots |\mathbf{w}|} w_i}{|\mathbf{w}|} + (1 - v) \cdot \overline{c(\mathbf{w})} + p(\mathbf{w}) \rightarrow \min \quad (4)$$

with

$$p(\mathbf{w}) = \sum \begin{cases} w_i^2, & w_i < 0 \\ (w_i - 1)^2, & w_i > 1 \\ 0, & 0 \leq w_i \leq 1 \end{cases} \quad (5)$$

and $c(\mathbf{w})$ denoting the mean error of COHDA simulations runs conducted with weight vector \mathbf{w} measured in MAPE. Function $p(\mathbf{w})$ introduces a penalty for weight values not in $[0, 1]$. In our simulation we set the weight v that balances minimization of 1-weight (thus maximizing the local weights) and minimization of the load scheduling error resulting from the weights to $v = 0.5$.

We used the simulation and multi-agent system presented in Sect. 3 and the architecture from Fig. 2. The control entity has access to the agents to set individual weights for the agents via a control interface. A COHDA optimization run can be started by the control entity. The multi-agent system then conducts COHDA autonomously, but the result again can be observed by the control entity (by subscribing to an observer interface).

Overall, the following control loop is established. The controller conducts CMA-ES with the objective of finding a good weight vector \mathbf{w} . During each iteration CMA-ES samples new candidates of \mathbf{w} . These candidates are evaluated

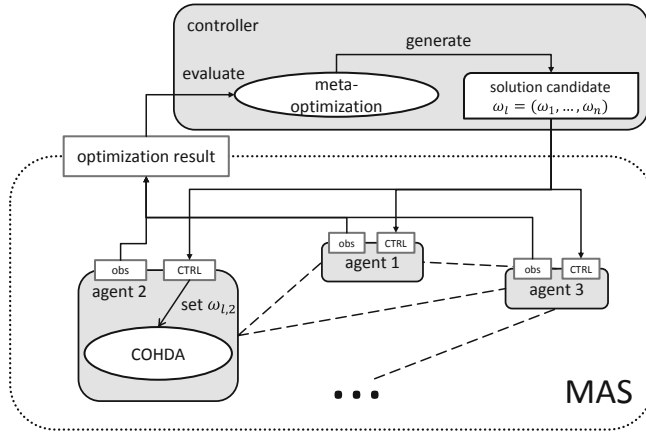


Fig. 2. Architecture for adjusting local weights in controlled self-organized global/local multi-objective energy scheduling.

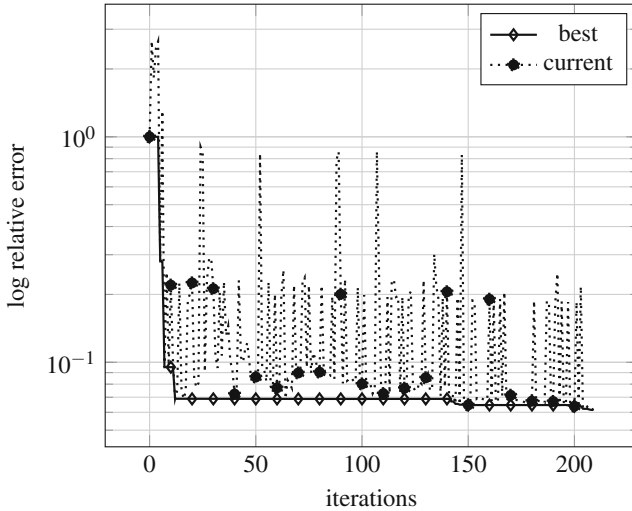


Fig. 3. Example convergence profile for a scenario with 5 co-generation plants.

by sending the respective weight values to the respective agents and starting COHDA. The result schedules of COHDA are collected, summed up and compared with the market schedule ζ (this is repeated 5 times for each candidate w). The mean error from COHDA serves for evaluating the candidate weights w . In this way, good combinations of weights can be found for individually steering the weighting of local preferences in the self-organizing load planning process of the agents.

Figure 3 shows the convergence behavior of a first result for a scenario with 5 co-generation plants. Figure 3 shows another example with 10 units. Please note

Table 2. Mean, min., and and best optimization results (mean MAPE achieved with the resulting weight vector \mathbf{w}) for different values of ϵ (threshold for result improvements as termination condition).

| ϵ | Mean | | Min | Max |
|------------|--------|--------|---------|--------|
| 0.1 | 1.7311 | 1.0840 | 0.0043 | 3.6410 |
| 0.01 | 1.3607 | 1.1656 | 0.0029 | 3.5628 |
| 0.001 | 0.9803 | 0.8854 | 0.00079 | 2.9657 |

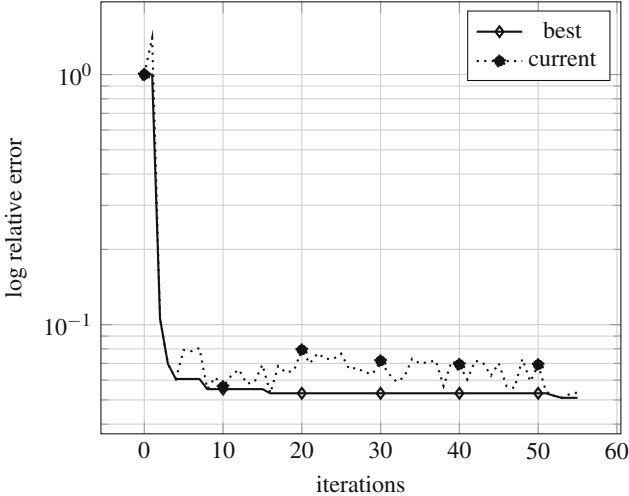


Fig. 4. Example convergence profile for a scenario with 10 co-generation plants.

that the primary axis denotes iterations. In each iteration a population of size 8 (for the 5 units scenario) and 10 for the large scenario has to be evaluated respectively. Nevertheless, convergence is promising.

Table 2 shows some statistics on the result quality for the 10 units scenario for different ϵ -values for convergence checks (mean improvement of two succeeding iterations). A moderate ϵ seems to be sufficient to push the quality of the main goal back below 1%, what is rather small compared with forecast error e. g. in photovoltaics production as a prerequisite for the initial market trading [17]. Overall, the results are promising enough to justify further research and improvements (Fig. 4).

5 Conclusion

Integrating local intentions and preferences into distributed scheduling as local objectives inevitably deteriorates the global solution quality of the primary goal. Examples were given using the predictive scheduling use case in the smart grid

domain. We demonstrated that depending on the current situation this deterioration can be mitigated by an appropriate distribution of weights for local objectives. This task constitutes a meta-optimization problem that can be solved by an architecture for controlled self-organization that is introduced into the multi-agent system responsible for planning. First results are promising. Nevertheless, for a quick reaction of the proposed system further research regarding better performance and convergence behavior of the used meta-optimization algorithm are necessary.

References

1. Awad, A., German, R.: Self-organizing smart grid services. In: 2012 Sixth International Conference on Next Generation Mobile Applications, Services and Technologies, pp. 205–210. IEEE (2012)
2. Awerbuch, S., Preston, A.M. (eds.): The Virtual Utility: Accounting, Technology & Competitive Aspects of the Emerging Industry, Topics in Regulatory Economics and Policy, vol. 26. Kluwer Academic Publishers, Berlin (1997)
3. Boyd, S.P., Ghosh, A., Prabhakar, B., Shah, D.: Gossip algorithms: design, analysis and applications. In: Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 3, pp. 1653–1664 (2005)
4. Bremer, J., Rapp, B., Jellinghaus, F., Sonnenschein, M.: Tools for teaching demand-side management. In: Wohlgemuth, V., Page, B., Voigt, K. (eds.) Environmental Informatics and Industrial Environmental Protection - 23rd International Conference on Informatics for Environmental Protection, EnviroInfo2009, pp. 455–463 (2009)
5. Bremer, J.: Constraint-Handling mit Supportvektor-Dekodern in der verteilten Optimierung. Ph.D. thesis (2015). <http://oops.uni-oldenburg.de/2336/>
6. Bremer, J., Lehnhoff, S.: A decentralized PSO with decoder for scheduling distributed electricity generation. In: Squillero, G., Burelli, P. (eds.) EvoApplications 2016. LNCS, vol. 9597, pp. 427–442. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31204-0_28
7. Bremer, J., Rapp, B., Sonnenschein, M.: Encoding distributed search spaces for virtual power plants. In: IEEE Symposium Series on Computational Intelligence 2011 (SSCI 2011), Paris, France, April 2011
8. Bremer, J., Sonnenschein, M.: Constraint-handling for optimization with support vector surrogate models - a novel decoder approach. In: Filipe, J., Fred, A.L.N. (eds.) Proceedings of the 5th International Conference on Agents and Artificial Intelligence (ICAART 2013), 15–18 February 2013, Barcelona, Spain, vol. 2, pp. 91–100. SciTePress (2013)
9. Coll-Mayor, D., Picos, R., García-Moreno, E.: State of the art of the virtual utility: the smart distributed generation network. *Int. J. Energy Res.* **28**(1), 65–80 (2004)
10. Wedde, H.F., Lehnhoff, S., Rehtanz, C., Krause, O.: Intelligent agents under collaborative control in emerging power systems. *Int. J. Eng. Sci. Technol.* **2**, 45–59 (2010)
11. Futia, C.A.: Excess supply equilibria. *J. Econ. Theory* **14**(1), 200–220 (1977)
12. Hansen, N.: The CMA evolution strategy: a tutorial. Technical report (2011). www.lri.fr/~hansen/cmatutorial.pdf

13. Hinrichs, C., Bremer, J., Sonnenschein, M.: 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)
14. Hinrichs, C., Lehnhoff, S., Sonnenschein, M.: A decentralized heuristic for multiple-choice combinatorial optimization problems. In: Helber, S., et al. (eds.) Operations Research Proceedings 2012. ORP, pp. 297–302. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-00795-3_43
15. Hinrichs, C., Sonnenschein, M.: A distributed combinatorial optimisation heuristic for the scheduling of energy resources represented by self-interested agents. *Int. J. Bio-Inspired Comput.* **10**, 69 (2017)
16. Kamphuis, R., Warmer, C., Hommelberg, M., Kok, K.: Massive coordination of dispersed generation using powermatcher based software agents. In: 19th International Conference on Electricity Distribution, May 2007
17. Kramer, O., Gieseke, F.: Short-term wind energy forecasting using support vector regression. In: Corchado, E., Snášel, V., Sedano, J., Hassanien, A.E., Calvo, J.L., Ślęzak, D. (eds.) SOCO 2011. AINSC, pp. 271–280. Springer, Berlin Heidelberg, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19644-7_29
18. Lehn, J.M.: Towards complex matter: supramolecular chemistry and self-organization. *Eur. Rev.* **17**(2), 263–280 (2009)
19. Liu, L., Thanheiser, S., Schmeck, H.: A reference architecture for self-organizing service-oriented computing. In: Brinkschulte, U., Ungerer, T., Hochberger, C., Spallek, R.G. (eds.) ARCS 2008. LNCS, vol. 4934, pp. 205–219. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78153-0_16
20. Lust, T., Teghem, J.: The multiobjective multidimensional knapsack problem: a survey and a new approach. *CoRR* abs/1007.4063 (2010)
21. Meunier, F.: Co-and tri-generation contribution to climate change control. *Appl. Thermal Eng.* **22**(6), 703–718 (2002)
22. Mittal, S., Rainey, L.: Harnessing emergence: the control and design of emergent behavior in system of systems engineering. In: Proceedings of the Conference on Summer Computer Simulation, pp. 1–10. Society for Computer Simulation International (2015)
23. Müller-Schloer, C., Schmeck, H., Ungerer, T.: Organic Computing—A Paradigm Shift for Complex Systems. Springer Science & Business Media, Heidelberg (2011)
24. Ngatchou, P., Zarei, A., El-Sharkawi, A.: Pareto multi objective optimization. In: Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems, pp. 84–91. IEEE (2005)
25. Nieße, A.: Verteilte kontinuierliche Einsatzplanung in Dynamischen Virtuellen Kraftwerken. Dissertation, Carl v. Ossietzky Universität, Oldenburg (2015)
26. Nieße, A., Beer, S., Bremer, J., Hinrichs, C., Lünsdorf, O., Sonnenschein, M.: Conjoint dynamic aggregation and scheduling methods for dynamic virtual power plants. In: Ganzha, M., Maciaszek, L.A., Paprzycki, M. (eds.) Proceedings of the 2014 Federated Conference on Computer Science and Information Systems. *Annals of Computer Science and Information Systems*, vol. 2, pp. 1505–1514. IEEE (2014)
27. Nieße, A., Beer, S., Bremer, J., Hinrichs, C., Lünsdorf, O., Sonnenschein, M.: Conjoint dynamic aggregation and scheduling for dynamic virtual power plants. In: Ganzha, M., Maciaszek, L.A., Paprzycki, M. (eds.) Federated Conference on Computer Science and Information Systems - FedCSIS September 2014, Warsaw, Poland (2014)
28. Nieße, A., Bremer, J., Sonnenschein, M.: Continuous scheduling. In: Smart Nord - Final Report, pp. 69–76. Hartmann GmbH, Hannover (2015)

29. Nieße, A., Tröschel, M.: Controlled self-organization in smart grids. In: Proceedings of the 2016 IEEE International Symposium on Systems Engineering (ISSE), pp. 1–6. IEEE (2016)
30. Nikonowicz, L.B., Milewski, J.: Virtual power plants - general review: structure, application and optimization. *J. Power Technol.* **92**(3) (2012). <http://papers.itc.pw.edu.pl/index.php/JPT/article/view/284/492>
31. Percec, V., et al.: Transformation from kinetically into thermodynamically controlled self-organization of complex helical columns with 3D periodicity assembled from dendronized perylene bisimides. *J. Am. Chem. Soc.* **135**(10), 4129–4148 (2013)
32. Ramchurn, S.D., Vytelingum, P., Rogers, A., Jennings, N.R.: Agent-based homeostatic control for green energy in the smart grid. *ACM Trans. Intell. Syst. Technol.* **2**(4), 35:1–35:28 (2011)
33. Richter, U., Mnif, M., Branke, J., Müller-Schloer, C., Schmeck, H.: Towards a generic observer/controller architecture for organic computing. In: Jahrestagung, G.I. (1) LNI, vol. 93, pp. 112–119. GI (2006)
34. Steghöfer, J.-P., et al.: Trustworthy organic computing systems: challenges and perspectives. In: Xie, B., Branke, J., Sadjadi, S.M., Zhang, D., Zhou, X. (eds.) ATC 2010. LNCS, vol. 6407, pp. 62–76. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16576-4_5
35. Watts, D., Strogatz, S.: Collective dynamics of ‘small-world’ networks. *Nature* **393**(6684), 440–442 (1998)
36. Wu, F., Moslehi, K., Bose, A.: Power system control centers: past, present, and future. *Proc. IEEE* **93**(11), 1890–1908 (2005)